

# **Especificación de requisitos de software cpPlugins 3.0**

# cpPlugins 3.0

**Integrantes:**

**Pablo Andrés Ariza Luna  
Carlos Orlando Barón León  
Santiago Chaustre Perdomo  
Andrés Felipe Cocunubo Quintero  
Ailín Ana María Rojas Bohórquez**

**Director:**

**Leonardo Flórez Valencia PhD.**

**Asesor:**

**Jaime Andrés Pavlich Mariscal PhD.**

**Mayo 2019**

## Historial de cambios

Fecha	No. Sección	Nombre de la sección	Responsable
16/05/2019	-	Estructura general del documento, títulos.	Andrés Cocunubo
18/05/2019	5	Atributos del producto	Pablo Ariza, Santiago chaustre
18/05/2019	4	Requisitos del producto	Andrés Cocunubo, Ailín Rojas
18/05/2019	3.1	Perspectiva del producto	Santiago Chaustre
18/05/2019	3.3	Suposiciones y restricciones	Carlos Barón
18/05/2019	1	Introducción	Carlos Barón
18/05/2019	3.2	Interfaces con el sistema	Andrés Cocunubo

# cpPlugins 3.0

## Tabla de contenidos

Historial de cambios .....	2
Tabla de contenidos .....	3
Lista de tablas .....	5
1    Introducción .....	6
1.1    Propósito .....	6
1.2    Alcance .....	6
2    Glosario .....	7
3    Descripción global .....	8
3.1    Perspectiva de producto .....	8
3.2    Interfaces del sistema .....	8
3.3    Suposiciones y restricciones .....	8
3.3.1    Suposiciones .....	8
3.3.2    Restricciones .....	9
4    Requisitos del producto .....	10
4.1    Especificación de requisitos .....	10
4.1.1    Requisitos funcionales .....	10
4.1.2    Requisitos no funcionales .....	11
4.2    Priorización de requisitos .....	11
4.2.1    Priorización requisitos funcionales .....	11
4.2.2    Priorización requisitos no funcionales .....	12
5    Atributos del producto .....	13
5.1    Rendimiento .....	13
5.2    Usabilidad .....	13
5.3    Portabilidad .....	13
5.4    Interoperabilidad .....	13
5.5    Seguridad .....	13
5.6    Confiabilidad .....	13
5.7    Atributos Relevantes .....	13
6    Anexos .....	14
6.1    Plan de administración de software de cpPlugins 3.0 .....	14
6.2    Requisitos de cpPlugins 3.0 .....	14
7    Referencias .....	15

# cpPlugins 3.0

## Lista de tablas

Tabla 1-Estimación de requisitos .....	10
Tabla 2 - Priorización requisitos funcionales .....	12
Tabla 3- Priorización requisitos no funcionales .....	12

# cpPlugins 3.0

# 1 Introducción

## 1.1 Propósito

El objetivo de este documento es establecer los requisitos de cpPlugins 3.0, indicando los aspectos que influyan en estos y su respectiva descripción. Para cumplir con este objetivo, se expone el contexto en el que se encuentra cpPlugins 3.0, mostrando los beneficios que trae su desarrollo. En seguida, se muestran los mecanismos de comunicación que tiene cpPlugins 3.0 tanto con sus usuarios como con sistemas externos. El documento continúa estableciendo todos los aspectos que afectan directa o indirectamente los requisitos establecidos. También, se listan formalmente los requisitos funcionales y no funcionales y se realiza la priorización de estos. Finalmente, se describen los atributos de calidad que debe tener el sistema para su correcto funcionamiento.

Este documento es importante debido a que establece formalmente las características y funcionalidades que espera el cliente del sistema. De esta manera, el documento busca conseguir claridad durante el desarrollo del producto y un punto de referencia a la hora de validar el sistema con el cliente.

## 1.2 Alcance

cpPlugins 3.0 es una aplicación web para el diseño de flujos de procesamiento de datos médicos. Dentro del diseño de estos flujos de datos, se encuentran funcionalidades del producto como subir al sistema diferentes formatos de archivos de datos médicos, que se encuentren en diferentes discos (locales o en la nube), mostrar todos los algoritmos de procesamiento o visualización que se puedan realizar con el sistema, otorgar al usuario la capacidad de diseñar un grafo con los componentes descritos para representar un flujo de procesamiento, permitir al usuario la posibilidad de visualizar e interactuar con los datos del flujo en diferentes etapas del procesamiento, y el poder ejecutar parcialmente o por completo su flujo de datos.

Entre otros aspectos, es de gran importancia que cpPlugins 3.0 sea altamente usable, de manera que cualquier persona, sin conocimientos en programación, pueda diseñar su propio flujo de procesamiento de datos médicos, y así mismo pueda extender la funcionalidad de los algoritmos presentados por medio de su diseño. Además, se debe brindar al usuario una respuesta rápida durante el diseño y a la hora de ejecutar cualquier procesamiento. Adicionalmente se debe tener acceso al sistema en diferentes plataformas.

## 2 Glosario

- **Deadline:** la hora o el día en que algo debe ser terminado [1].
- **Drag and drop:** técnica de arrastrar y soltar, acción de mover objetos con el ratón de una ventana a otra o entre partes de una misma ventana [2].
- **Funcionalidad:** término que hace referencia a un algoritmo o filtro que se aplica a un dato médico estructurado.
- **Pipeline:** transformación de un flujo de datos en un proceso comprendido por varias fases secuenciales, siendo la entrada de cada una la salida de la anterior [3].
- **Procedimiento:** conjunto de funcionalidades que conforman un pipeline.

# cpPlugins 3.0

## 3 Descripción global

### 3.1 Perspectiva de producto

cpPlugins3.0, pretende ser una ampliación de la familia de cpPlugins, extendiendo las funcionalidades existentes en el sistema mediante un aplicativo web, que permita a los diferentes usuarios, realizar procesamiento y visualización de datos médicos estructurados gracias a la facilidad de diseño de pipelines de manera gráfica, en cualquier equipo de cómputo sin importar sus recursos técnicos tales como hardware o software. También añade un factor diferencial a la arquitectura, planteando un modelo cliente/servidor, ofreciendo servicios web para comunicación con el cliente.

La propuesta de proyecto nace principalmente de dos factores clave, el primero de ellos es la portabilidad, ya que en la versión existente de cpPlugins es necesario contar con un entorno de ejecución específico para lograr el funcionamiento de esta. El segundo factor está relacionado con la usabilidad, ya que actualmente cuenta con una interfaz de línea de comandos, la cual no es sencilla de utilizar para el público objetivo al que está enfocado el aplicativo.

### 3.2 Interfaces del sistema

Interfaz	Descripción
Usuario	El usuario podrá interactuar con el sistema ingresando a través de un navegador, estos pueden ser Google Chrome o Mozilla Firefox. Allí, el usuario con ayuda del ratón y la pantalla usará el sistema drag and drop con el que cuenta la plataforma.
Software	Puesto que cpPlugins está escrito en C++, se utilizará el framework JNI para la interacción entre el software existente y el que será implementado. Para cargar la información a la plataforma se usarán servicios de la nube o archivos locales del computador donde se encuentre el usuario. Algunos ejemplos de servicios de nube son OneDrive, Google Drive, Mega, entre otros.
Hardware	Teniendo en cuenta que la plataforma es web es necesario ciertos componentes mínimos de red, para el correcto funcionamiento, estos son el cable ethernet o WiFi y tarjeta de red.
Comunicación	Se harán peticiones HTTP tanto para enviar como para recibir información del servidor, siguiendo estándares de seguridad, ya que la información es de carácter privado.

### 3.3 Suposiciones y restricciones

En esta sección se presentan todos los factores que afectan o restringen los requisitos del sistema.

#### 3.3.1 Suposiciones

Las suposiciones son:

- Los usuarios de cpPlugins 3.0 no cuentan con fundamentos de programación.
- Los usuarios de cpPlugins 3.0 cuentan con conocimientos básicos acerca del procesamiento de datos médicos.



- Los usuarios de cpPlugins 3.0 cuentan con conocimientos acerca de la forma y estructura de los datos médicos.
- Los usuarios que accedan al sistema cuentan con una conexión a internet superior a 100 Mb.
- Para su primera versión, el servidor de cpPlugins 3.0 se encuentra centralizado.
- Para su primera versión, el sistema no brinda servicio a un gran número de usuarios.
- La aplicación existente (cpPlugins) posee una implementación estable.
- Se tiene acceso al código fuente de cpPlugins.

### **3.3.2 Restricciones**

#### **3.3.2.1 Hardware**

- El aplicativo del cliente debe poder ser ejecutado sin problemas en computadores personales de no muy alto costo.
- Los servidores que ejecuten cpPlugins deben poseer un rendimiento superior, de manera que puedan acabar tareas de procesamiento en cortos periodos de tiempo.

#### **3.3.2.2 Software**

- Los servidores que ejecuten cpPlugins deben contar con un sistema operativo GNU/Linux.
- La visualización e interacción de datos médicos realizada por el cliente, debe ser elaborada con vtk.js.

# cpPlugins 3.0

## 4 Requisitos del producto

A continuación, se presentan los requisitos del producto recolectados por el grupo de trabajo, para detalles del proceso de requisitos (ver Anexo 1. SPMP\_cpPlugins3.0, sección 5.1.1 Identificación de requisitos).

### 4.1 Especificación de requisitos

Esta sección comprende la descripción de los requisitos de software (ver Anexo2. Requisitos.xlsx, primera hoja “Especificación”), los cuales han sido analizados, clasificados, verificados y priorizados por el equipo de trabajo. Así mismo, se realizó una estimación de tiempo para cada requisito funcional, con el objetivo de medir la dificultad de implementación y poder distribuir las tareas en el equipo de trabajo.

Para medir la dificultad, se utilizó una escala de tres niveles en donde cada uno de estos se relaciona con un intervalo de tiempo. A continuación, en la tabla 1. Estimación de requisitos, se ilustra la relación del tiempo con los niveles de dificultad.

Dificultad	Tiempo requerido
Baja	1-3 días
Media	3-5 días
Alta	5-8 días

*Tabla 1-Estimación de requisitos*

La estimación de cada requisito es el resultado de la votación de los miembros del equipo.

#### 4.1.1 Requisitos funcionales

Esta sección, contiene la lista de requisitos funcionales que describen las características que el sistema debe satisfacer para ser aceptado por el cliente.

- **RF-01:** El sistema debe cargar imágenes médicas de sistemas externos.
- **RF-02:** El sistema debe cargar imágenes médicas de forma local.
- **RF-03:** El sistema debe presentar los pipelines de procesamiento diseñados por el usuario.
- **RF-04:** El sistema debe presentar en pantalla datos médicos estructurados.
- **RF-05:** El sistema debe ejecutar funcionalidades específicas del pipeline.
- **RF-06:** El sistema debe ejecutar todo el procedimiento descrito en el pipeline.
- **RF-07:** El sistema debe desplegar las funcionalidades de manera ordenada por categorías.
- **RF-08:** El sistema debe proveer widgets para la interacción del usuario con los datos y el pipeline del sistema.
- **RF-09:** El sistema debe proveer las operaciones de creación, lectura, actualización, deshacer operación anterior y rehacer operación eliminada dentro del pipeline.
- **RF-10:** El sistema debe crear pipelines a partir de un archivo descriptor.
- **RF-11:** El sistema debe almacenar pipelines en diferentes fuentes.

#### 4.1.2 Requisitos no funcionales

A continuación, se presenta la lista de requisitos no funcionales que se identificaron de acuerdo con los atributos de calidad que el sistema debe proveer al ser implementado (**ver sección 5. Atributos del producto**) y restricciones a tener en cuenta.

- **Rendimiento**
  - **RNF-01:** El sistema debe enviar y recibir datos médicos estructurados en no más de 10sg
- **Usabilidad**
  - **RNF-05:** El usuario no debe gastar más de 4 horas en aprender a utilizar el sistema.
  - **RNF-09:** El sistema debe ser desarrollado utilizando guías de diseño para las interfaces de usuario.
- **Portabilidad**
  - **RNF-06:** El aplicativo web debe ejecutarse en diferentes sistemas operativos y navegadores.
- **Interoperabilidad**
  - **RNF-07:** El sistema debe intercambiar información con plataformas de alojamiento de archivos en la nube.
- **Seguridad**
  - **RNF-03:** El sistema debe garantizar la confidencialidad de los datos médicos estructurados.
- **Confiabilidad**
  - **RNF-02:** El sistema debe garantizar la integridad de los datos médicos estructurados.
  - **RNF-04:** El sistema debe garantizar la disponibilidad de los datos médicos estructurados.
- **Mantenibilidad**
  - **RNF-08:** El sistema debe ser desarrollado utilizando las guías de programación establecidas.

#### 4.2 Priorización de requisitos

La priorización de los requisitos se realizó utilizando el método de urgencia y valor de negocio, este método consiste en realizar la valorización del requerimiento a través de dos escalas, la primera es la urgencia o tiempo y la segunda es el valor de negocio. Cada escala consiste en cinco rangos, en donde cada uno determina características del requerimiento. Para más detalle acerca de este método ver [4].

##### 4.2.1 Priorización requisitos funcionales

De acuerdo con los resultados obtenidos, se tomaron como prioritarios aquellos requisitos funcionales que obtuvieron un rango **CRÍTICO** o **IMPORTANTE**.

Los críticos deben ser desarrollados lo más pronto posible y los importantes deben implementarse enseguida de los críticos.

De esta manera, los requisitos funcionales prioritarios son:

ID	Descripción	Rango
RF-01	El sistema debe cargar imágenes médicas de sistemas externos.	CRÍTICO

<b>RF-02</b>	El sistema debe cargar imágenes médicas de forma local.	CRÍTICO
<b>RF-03</b>	El sistema debe presentar los pipelines de procesamiento diseñados por el usuario.	CRÍTICO
<b>RF-04</b>	El sistema debe presentar en pantalla datos médicos estructurados.	CRÍTICO
<b>RF-06</b>	El sistema debe ejecutar todo el procedimiento descrito en el pipeline.	CRÍTICO
<b>RF-09</b>	El sistema debe proveer las operaciones de creación, lectura, actualización, deshacer operación anterior y rehacer operación eliminada dentro del pipeline.	CRÍTICO
<b>RF-10</b>	El sistema debe crear pipelines a partir de un archivo descriptor.	IMPORTANTE

*Tabla 2 - Priorización requisitos funcionales*

#### 4.2.2 Priorización requisitos no funcionales

Al igual que los funcionales, se tomaron como prioritarios aquellos requisitos funcionales que obtuvieron un rango **CRÍTICO** o **IMPORTANTE**.

En la siguiente tabla se muestran los requisitos no funcionales prioritarios:

ID	Descripción	Rango
<b>RNF-03</b>	El sistema debe garantizar la confidencialidad de los datos médicos estructurados.	IMPORTANTE
<b>RNF-05</b>	El usuario no debe gastar más de 4 horas en aprender a utilizar el sistema.	CRÍTICO
<b>RNF-06</b>	El aplicativo web debe ejecutarse en diferentes sistemas operativos y navegadores.	IMPORTANTE
<b>RNF-09</b>	El sistema debe ser desarrollado utilizando guías de diseño para las interfaces de usuario	IMPORTANTE
<b>RNF-03</b>	El sistema debe garantizar la confidencialidad de los datos médicos estructurados.	IMPORTANTE
<b>RNF-05</b>	El usuario no debe gastar más de 4 horas en aprender a utilizar el sistema.	CRÍTICO
<b>RNF-06</b>	El aplicativo web debe ejecutarse en diferentes sistemas operativos y navegadores.	IMPORTANTE

*Tabla 3- Priorización requisitos no funcionales*

## 5 Atributos del producto

### 5.1 Rendimiento

El rendimiento general de la aplicación depende de los algoritmos. Por lo tanto, el tiempo de respuesta de estos varía.

Para la comunicación se tiene un *deadline* de tres de minutos para cargar la información al servidor. Para la descarga de información proveniente del servidor se tiene un *deadline* de 10 segundos.

### 5.2 Usabilidad

La usabilidad de la aplicación se encuentra en la interfaz *drag and drop* ofrecida. Facilita el diseño de los pipelines al mostrar los recursos brindados, los cuales son filtros, widgets, funciones y procedimiento. También, mostrará de manera visual que puertos de destino puede ser conectados con los puertos de origen de cada componente.

Finalmente, cada componente del pipeline tendrá una documentación mínima que se mostrará al permanecer con el ratón sobre este.

### 5.3 Portabilidad

La portabilidad de la aplicación se asegura al desarrollar el cliente con tecnologías web. De este modo lo necesario para la ejecución es el acceso a internet y tener un navegador web.

### 5.4 Interoperabilidad

El servidor debe tener la capacidad de obtener los datos médicos estructurados tanto de almacenamiento interno como del almacenamiento en la nube.

### 5.5 Seguridad

Debido a la clasificación de la información, es de vital importancia brindar un medio seguro para transmitir los datos médicos estructurados entre la máquina del usuario y el servidor, de tal manera que la información se mantenga íntegra y no sea accedida por individuos u entidades externas al dueño de esta.

### 5.6 Confiabilidad

Puesto que la aplicación procesa datos médicos estructurados y los resultados de un pipeline puede afectar el diagnóstico de los pacientes, es importante asegurar la integridad y disponibilidad de los datos médicos estructurados fruto del pipeline.

### 5.7 Atributos Relevantes

Los atributos más importantes del producto son la portabilidad y la usabilidad. Esto se debe a que según lo planteado en 1.2 Alcance la aplicación debe ser capaz de ser usada por médicos y personas que no saben programar.

La seguridad tiene un rango importante porque al usar imágenes médicas y el estándar DICOM se debe tener una seguridad mínima en el traspaso de la información.

La interoperabilidad y rendimiento no tienen relevancia, porque el primero da un valor adicional al negocio, es decir, no es crítico para el negocio. Por último, como se mencionó en 5.1 Rendimiento, los tiempos de ejecución de los algoritmos es variante. Por lo tanto, no se puede asegurar que todas las peticiones de los clientes se ejecuten en menos de un tiempo especificado.

Finalmente, al trabajar sobre lo realizado en versiones anteriores de cpPlugins la confiabilidad se ha venido mejorando, por ello la nueva versión posee una prioridad baja.

## **6 Anexos**

### **6.1 Plan de administración de software de cpPlugins 3.0**

Archivo Word que contiene la información correspondiente al SPMP de cpPlugins 3.0. [Anexos\1. SPMP\\_cpPlugins3.0.docx](#)

### **6.2 Requisitos de cpPlugins 3.0**

Archivo Excel que lista tanto los requisitos funcionales, como los no funcionales de cpPlugins 3.0. Además, contiene la priorización de estos. [Anexos\2.Requisitos.xlsx](#)

# cpPlugins 3.0

## 7 Referencias

- [1] «Significado de DEADLINE en el Diccionario Cambridge inglés». [En línea]. Disponible en: <https://dictionary.cambridge.org/es/diccionario/ingles/deadline>. [Accedido: 19-may-2019].
- [2] M. León, *Diccionario de Informatica, Telecomunicaciones y Ciencias Afines/Dictionary of Computing, Telecommunications, and Related Sciences: Ingles-Espanol/Spanish-English*. Ediciones Díaz de Santos, 2004.
- [3] «MODELO DE ARQUITECTURA PIPELINE - Electrónica analogica y digital». [En línea]. Disponible en: <https://www.electrontools.com/Home/WP/2018/04/18/modelo-de-arquitectura-pipeline/>. [Accedido: 05-may-2019].
- [4] Michael, «How to Easily Prioritize Your Agile Stories», *Michael Lant*, 21-may-2010. .

# cpPlugins 3.0