



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Diseño e implementación de Informes estructurados para el diagnóstico y seguimiento del Neuroblastoma y Glioma Pontino Intrínseco Difuso

TRABAJO FIN DE MÁSTER

Máster Universitario en Ingeniería Informática

Autor: Chichell Ruiz, David

Tutor: Segrelles Quilis, José Damián

Cotutora externa: Jimenez Pastor, Ana María

Curso 2019/2020

Resumen

El uso de las nuevas tecnologías en la toma de decisiones en el ámbito de la medicina es un factor clave para el desarrollo de un sistema sanitario robusto. La cantidad de información relativa a pacientes que se maneja actualmente necesita de un tratamiento y una organización que permita que el flujo de información entre distintos centros hospitalarios sea rápido y eficaz, siguiendo unos estándares internacionales que proporcionen una validez global de las interpretaciones que se realicen (interoperabilidad semántica). En este TFM, se propone el desarrollo de un módulo que genere en base a los datos recolectados de pacientes de Neuroblastoma y Glioma Pontino Intrínseco Difuso, informes estructurados que organicen esta información siguiendo el estándar DICOM. La codificación de términos, la semántica y el uso de ontologías son factores que han sido necesarios para obtener una correcta organización de la información. El módulo permite que los profesionales sanitarios generen un informe estructurado en estándar DICOM, visible desde las estaciones de trabajo de los sanitarios, que recoge información de diagnóstico, tratamiento y seguimiento en el que apoyarse para realizar la toma de decisiones.

Palabras clave: Informática médica, DICOM, ontologías, sanidad, informe estructurado

Resum

L'ús de les noves tecnologies per a la millora de la presa de decisions en l'àmbit de la medicina és un factor clau per al desenvolupament d'un sistema sanitari robust. La quantitat d'informació relativa a pacients necessita d'un tractament i una organització que permeti que el flux d'informació entre diferents centres hospitalaris siga ràpid i eficaç, seguint uns estàndards internacionals que proporcionen una validesa global de les interpretacions que es realitzen (interoperabilitat semantica). En aquest TFM es proposa el desenvolupament d'un mòdul que genere sobre la base de les dades recol·lectades de pacients de Neuroblastoma i Glioma Pontí Intrínsec Difús informes estructurats que organitzin aquesta informació seguint l'estàndard DICOM. La codificació de termes, la semàntica i l'ús d'ontologies són factors que han sigut necessaris per a obtindre una correcta organització de la informació. El mòdul permet que els professionals sanitaris generen un informe estructurat en estàndard DICOM, visible des de les estacions de treball dels sanitaris, que recull informació de diagnòstic, tractament i seguiment en el qual recolzar-se per a realitzar la presa de decisions.

Paraules clau: Informàtica mèdica, DICOM, ontologies, sanitat, informe estructurat

Abstract

Use of new technologies to improve decision-making in medicine is a key factor for the development of a robust healthcare system. The amount of information related to patients requires treatment and an organization that allows the information flow between different hospital centers to be fast and effective,

following international standards providing global validation about interpretations (semantic interoperability). The development of a module that generates structured reports following DICOM standard based on the collected data from Neuroblastoma and Diffuse Intrinsic Pontine Glioma patients is proposed. Term codification, semantics and use of ontologies are factors that have been necessary to obtain a correct organization of the information. The module allows healthcare professionals to generate a report structured in the DICOM standard, visible from the healthcare workstations, which collects diagnostic, treatment and follow-up information to lean on for decision-making.

Key words: Medical informatics, DICOM, ontologies, health, structured report

Índice general

Índice general	v
Índice de figuras	vii
Índice de tablas	viii

1	Introducción	1
1.1	Motivación	1
1.2	Objetivos	2
1.3	Metodología	3
1.4	Estructura de la memoria	4
2	Estado del arte	5
2.1	Estándares para los informes estructurados	5
2.1.1	DICOM	7
2.1.2	HL7	11
2.2	Terminologías estándar	13
2.2.1	SNOMED CT	14
2.2.2	RadLex	16
2.2.3	Otras	17
2.3	Uso de informes estructurados en otros proyectos	18
2.3.1	TRENCADIS y CVIMO	18
2.3.2	DICOM SR en redes de telemedicina	18
2.4	Crítica al estado del arte y propuesta	19
3	Análisis del problema	21
3.1	Metodología actual	21
3.2	Estructura de la información	23
4	Diseño de la solución	27
4.1	Tecnologías empleadas	27
4.1.1	La librería DCMTK	27
4.1.2	La librería nlohmann/JSON	28
4.2	Arquitectura del informe estructurado	30
5	Desarrollo del módulo	35
5.1	Instalación de las librerías y creación del proyecto en Windows	35
5.2	Instalación de las librerías y creación del proyecto en Ubuntu	39
5.3	Cabecera del informe estructurado	41
5.4	Nodo raíz y título del informe	45
5.5	Contenido y verificaciones del DICOM-SR	46
6	Despliegue del módulo	53
6.1	Herramientas de despliegue	53
6.2	Uso de Docker	55
7	Conclusiones	59

7.1 Relación del trabajo desarrollado con los estudios cursados	60
8 Trabajos futuros	63
Bibliografía	65

Apéndices

A Terminología propia del proyecto	67
B Dockerfile	71
C Tabla de relación de términos NB	73

Índice de figuras

2.1	Estructura de un informe estructurado [1]	8
2.2	Árbol con referencias entre nodos [2]	9
2.3	Anidamiento de clases en SNOMED CT	15
2.4	Información del término	15
2.5	Cuadro de mandos del Sistema Integrado de Telemedicina del Estado de Santa Catarina	19
3.1	Selección del proyecto en la plataforma PRIMAGE	22
3.2	Lista de estudios creados en la plataforma PRIMAGE	22
3.3	Campos del formulario de paciente de la plataforma PRIMAGE	23
3.4	Objeto JSON marcando las cuatro áreas de información del paciente	24
4.1	Resultado de búsqueda del término neuroblastoma	30
4.2	ID del término neuroblastoma	31
5.1	Paquetes de código fuente necesarios para la instalación	36
5.2	Directorio poblado tras instalación	37
5.3	Conjunto de módulos de cabecera	42
5.4	Informe estructurado de NB en el visor DICOM	44
5.5	Fragmento de cabecera del objeto DICOM-SR	45
5.6	Detalle del nodo raíz y sus atributos en el objeto DICOM	46
5.7	Detalle del nodo raíz en HTML	46
5.8	Inserción de un ítem	47
5.9	Extracto del informe estructurado con los componentes encuadrados	49
5.10	Extracto del informe estructurado con los componentes encuadrados	50
5.11	Inserción de un nodo de tipo CODE a raíz de un booleano	50
5.12	Número de alteraciones en la plataforma PRIMAGE	51
5.13	Implementación número de alteraciones	52
5.14	Resultado de la implementación del número de alteraciones	52
6.1	Arquitectura de Kubernetes	55
6.2	Imagen creada tras la ejecución del dockerfile	57
6.3	Contenedor de la imagen creada	57

Índice de tablas

2.1	Métricas de SNOMED CT en junio de 2020 [3]	14
2.2	Métricas de RadLex en junio de 2020 [4]	16
4.1	Pruebas para medir la conformidad [4]	28
4.2	Pruebas para medir el rendimiento [4]	29
5.1	Atributos del módulo paciente	42
5.2	Atributos del módulo estudio	43
5.3	Atributos del módulo documento	43

CAPÍTULO 1

Introducción

1.1 Motivación

El presente Trabajo Final de Máster (TFM) se enmarca en un proyecto médico traslacional europeo de cuatro años de duración (actualmente activo en su segundo año), dentro de los campos de investigación de imagen médica, inteligencia artificial y cáncer infantil. El proyecto en cuestión, es el proyecto PRIMAGE¹ (*PR*edictive *In-silico* *M*ultiscale *A*nalytics to support cancer personalized *di*agnosis and prognosis, *E*mpowered by imaging biomarkers)[5] [6]. Investigadores y colaboradores de PRIMAGE buscan y ofrecen el desarrollo de una plataforma abierta en la nube para facilitar la toma de decisiones de dos cánceres pediátricos considerados enfermedades raras. Estos son el Neuroblastoma (NB), que es el tumor sólido más frecuentes durante los cinco primeros años de vida [7], y el Glioma Pontino Intrínseco Difuso (DIPG), la principal causa de muerte relacionada con tumores cerebrales en niños.

Los principales desarrollos que se están realizando en PRIMAGE son módulos de análisis para la extracción de biomarcadores de imagen, simulaciones del desarrollo tumoral a diferentes escalas y visualizaciones de predicciones basadas en algoritmos de *machine-learning*, con el fin de que estas sean integradas en los flujos clínicos hospitalarios de diagnóstico, pronóstico, tratamiento, y seguimiento del DIPG y NB.

Uno de los puntos críticos para el desarrollo de dichas herramientas son la obtención y estandarización de la información a recopilar desde las bases de datos hospitalarias. Por ello, los principales socios clínicos implicados en el proyecto (sociedad europea de oncología pediátrica, dos biobancos de imágenes y varias unidades de oncología pediátrica de diferentes hospitales europeos), ponen a disposición de PRIMAGE sus bases de datos, a la vez que investigan procesos de extracción, anonimización, curación, estandarización y procesado de dicha información, con el objeto de contribuir a la construcción de una infraestructura de información compartida a nivel europeo que nutra a los socios tecnológicos de datos para el desarrollo y validación de las herramientas comentadas. El proyecto PRIMAGE necesita de un entorno multidisciplinar para avanzar y llevar a cabo el proyecto. Entre los socios sanitarios se encuentra el Hospital Universitario y Politécnico La Fe de Valencia; el *St. Anna Children's Cancer Research Institute* de Viena,

¹<https://www.primageproject.eu/>

Austria; el hospital universitario de Colonia, Alemania y el hospital universitario de Pisa, Italia. Estos centros pertenecen a la red europea de investigación en NB y DIPG donde se otorga acceso a los socios de PRIMAGE a los conjuntos de datos existentes sobre los pacientes de estas enfermedades.

Entre los grupos de investigación y las entidades privadas que participan en el proyecto PRIMAGE se encuentra la empresa *Quantitative Imaging Biomarkers in Medicine* (QUIBIM) de Valencia; el Instituto de Instrumentación para Imagen Molecular y el departamento de Ingeniería Mecánica de la Universidad Politécnica de Valencia (UPV); la empresa Chemotargets y *Matical Innovation* todas entidades españolas. El departamento de ingeniería informática de la universidad de Constanza, Alemania; el *Medical Imaging Technologies* (Medexpim) de Francia; la universidad de *Sheffield* en el Reino Unido, el grupo Ansys encargado de simulaciones y diseños 3D en Francia; la Universidad de Ciencias y Tecnologías de Cracovia, Polonia y el departamento de Ingeniería Industrial de la Universidad de Bolonia en Italia.

En este contexto, el presente TFM se ubica específicamente en los procesos de estandarización de la información de los datos clínicos, con el objeto de conseguir una interoperabilidad semántica plena entre los socios tanto clínicos como tecnológicos. El TFM es cotutorizado por el tutor de la UPV perteneciente al Instituto de Instrumentación para Imagen Molecular y por la tutora externa perteneciente a la empresa QUIBIM debido a que, como miembros del mismo proyecto colaboran en la creación de la plataforma web para la recogida de datos de los pacientes diagnosticados de NB o DIPG a través de una serie de formularios (e-forms). La empresa QUIBIM se centra en la aplicación de técnicas de inteligencia artificial y modelos computacionales avanzados sobre imágenes radiológicas para medir los cambios producidos por una lesión o por un tratamiento farmacológico, ofreciendo información cuantitativa adicional basada en algoritmos que complementa la información cualitativo de la radiología, mientras que los grupos de investigación y departamentos citados anteriormente pertenecientes a la UPV se encargan de la infraestructura en la nube del proyecto PRIMAGE. Los usuarios con acceso a esta plataforma desarrollada cuentan con una infraestructura que permite añadir estudios de pacientes a través de los mencionados formularios y compartirlos con los demás participantes. Esta información recogida en los formularios será la base de nuestro proyecto de conversión para la creación de plantillas de informes estructurados siguiendo el estándar DICOM.

1.2 Objetivos

El objetivo general del presente TFM es el desarrollo e integración de un módulo *software* para generar informes estructurados en base a los formularios (e-forms) definidos en el proyecto PRIMAGE para la recolección de datos clínicos de pacientes NB y DIPG de los diferentes hospitales involucrados. Los informes estructurados seguirán el estándar DICOM (*Digital Imaging and Communications in Medicine*), detallado en el capítulo 2, y se construirá en base a los datos introducidos por los profesionales clínicos a través de la plataforma PRIMAGE, con el fin de que estos puedan ser descargados y consultados por todos los investigadores

del proyecto para el desarrollo de herramientas que ayuden en el pronóstico y diagnóstico de NB y DIPG.

Para la lograr la consecución del objetivo general se identifican los siguientes subobjetivos altamente marcados por la metodología empleada:

- Seleccionar herramientas para la implementación de los informes estructurados mediante el estándar DICOM-SR. Esto incluye tanto el Entorno de desarrollo integrado como las librerías a utilizar.
- Diseño de plantillas DICOM-SR a partir de los formularios (e-form) ya existentes en la plataforma de PRIMAGE. En dicho diseño se definirá la jerarquía de los datos, así como la identificación de los términos utilizados en terminologías estándares, tanto de los conceptos de los campos utilizados, como de los valores que pueden tener dichos campos.
- Implementar la codificación de los informes en base a las plantillas diseñadas. Dicha implementación se traducirá en un modulo software integrable en la plataforma actual de PRIMAGE.
- Verificar que la codificación de los informes existentes en el proyecto PRIMAGE a DICOM-SR es válida y correcta.
- Desplegar el módulo de creación de informes para su integración en la plataforma PRIMAGE.

1.3 Metodología

Debido a que se trata del desarrollo de un módulo muy concreto de un sistema de información, la metodología empleada a lo largo del proyecto ha seguido las mismas etapas del ciclo de vida de un sistema de información[8]. Es por esto por lo que para el buen seguimiento del proyecto se han concretado las siguientes fases:

1. Análisis: en esta fase se procede a la identificación de necesidades y las condiciones que debe cumplir el módulo a desarrollar.
2. Diseño: representación de la organización del módulo en el sistema íntegro, así como características y funcionalidades extraídas de la fase anterior.
3. Implementación: desarrollo del módulo del sistema.
4. Verificación: ejecución del módulo del sistema para verificar que genera correctamente el informe estructurado.
5. Despliegue: despliegue del módulo para la integración en la plataforma

1.4 Estructura de la memoria

Atendiendo a lo descrito en la sección de motivación, los objetivos especificados y la metodología descrita, el documento ha sido organizado como se especifica a continuación. El capítulo dos plasma una revisión del estado del arte de las terminologías, estándares actuales y proyectos relacionados con la temática del trabajo, así como también la propuesta del proyecto en este ámbito. El capítulo tres describe la problemática a la que nos enfrentamos, se describe la problemática a resolver mediante el desarrollo del proyecto. En el capítulo cuatro se presenta el diseño de la solución y el primer resultado obtenido. Se exponen las tecnologías empleadas acompañadas de análisis de tecnologías similares y justificación de la elección de la tecnología seleccionada para el proyecto. El capítulo cinco describe el procedimiento que se ha seguido para desarrollar el módulo, desde la instalación del entorno hasta la programación del informe estructurado y el resultado obtenido de la ejecución del módulo. En el capítulo seis se mencionan algunas herramientas de despliegue, así como el procedimiento seguido en nuestro proyecto y las tecnologías empleadas. El capítulo siete presenta las conclusiones obtenidas y la relación del trabajo con los estudios cursados. Finalmente, en el capítulo ocho se indican las futuras líneas de trabajo.

CAPÍTULO 2

Estado del arte

En esta sección se presenta el estado del arte actual del contexto del proyecto. En primer lugar se hará un repaso de los distintos tipos de informes existentes, se repasarán las ventajas y desventajas del uso de informes estructurados y se presentarán los estándares sanitarios más utilizados para la generación de informes estructurados. En segundo lugar se presentarán las terminologías estandarizadas más usadas, dónde consultarlas, qué información ofrecen y cómo usarlas en nuestro proyecto para organizar y codificar la información. En tercer lugar se hará un estudio del uso de informes estructurados en otros proyectos de ámbito similar, así como de otros estudios donde se haga uso de estándares mencionados. Finalmente se hará un repaso de los proyectos y presentaremos la solución propia para las carencias y diferencias identificadas entre el actual TFM y anteriores proyectos.

2.1 Estándares para los informes estructurados

El día a día de la medicina hace un uso intensivo de numerosos informes basados en la recolección de datos de pacientes, pruebas realizadas sobre el mismo, resultados, actuaciones, etc. El ámbito sanitario es uno de los campos que más uso hace de la información debido a que su funcionamiento se basa exclusivamente en la información recogida por los profesionales de la salud. En base a esta información el paciente puede ser atendido y el sistema sanitario puede avanzar e ir desarrollándose. Añadiendo que se trata de un área regulada, fundamentalmente en el ámbito de la gestión pública, y muy fragmentada debido a la naturaleza del Sistema Nacional de Salud (SNS). Esta obtiene competencias en los distintos territorios del país pudiendo existir protocolos, estándares o actuaciones diversas entre distintos centros hospitalarios. La fragmentación se ve aumentada si comparamos además los sistemas sanitarios de distintos países. El conjunto de reglas comunes es tan superfluo que no existe una coordinación común de estandarización[9].

Existen intentos de estandarización de casos muy concretos que llegan de mano de proyectos como PRIMAGE, donde distintos organismos, estados y empresas de países diferentes deben coordinarse y establecer un marco común de actuaciones. Todo ello con el objetivo de poder llevar a cabo un mismo proyecto y lograr una estandarización de una pequeña unidad. En nuestro caso, nos centra-

mos en la implementación de un informe estructurado que recoja la información esencial de los pacientes de NB o DIPG en cada una de las etapas de la enfermedad, pudiendo así estandarizar y recoger en un documento único las pruebas de carácter diagnóstico, los tratamientos, el seguimiento de la enfermedad y los resultados obtenidos de cada una de estas pruebas, y poder seguir así un protocolo estándar de atención a este tipo de pacientes.

Es muy común que los informes a día de hoy sean texto libre enmarcado en un formato estándar muy débil donde se hace uso de abreviaturas y algunos códigos de pruebas más comunes. Pero con el paso de los años, está creciendo el interés por el desarrollo de informes estructurados, sobre todo en el área de la imagen médica. Actualmente podemos reconocer tres niveles de informes estructurados[10]:

1. Informes estructurados simplemente con encabezamiento donde se indican datos personales del paciente.
2. Informes estructurados detallados, donde se incluyen resultados de pruebas, así como agrupamiento por categorías: descripción de pruebas sobre distintos órganos o equipamiento usado para la realización de la prueba.
3. Informes estructurados estandarizados que requieren de un uso común del lenguaje, así como un consenso de uso de léxico de ontologías estándar.

Este último nivel de estandarización de los informes estructurados, crea la cohesión del mismo pero resulta mucho más complejo de implementar. Por ello, siempre es conveniente realizar un análisis de las ventajas y las desventajas que pueden florecer a la hora de implementar este tipo de soluciones. Por esto se recogen las siguientes ventajas e inconvenientes sobre el uso de informes estructurados para el análisis de imagen médica.

Ventajas del uso de informes estructurados

- Se trata de documentos que cualquier radiólogo puede interpretar donde se hace uso de un léxico cohesionado, se minimiza sustancialmente la ambigüedad en los términos y se evita la confusión y los errores médicos.
- Ayuda a realizar el análisis de una manera más sencilla para constituir una herramienta que facilite la toma de decisiones.
- El resultado es un informe completo y estructurado donde se deben especificar todos los hallazgos pertinentes. El uso de una metodología y una estructura de trabajo ayuda a que aspectos menos significativos queden en el olvido.

Desventajas del uso de informes estructurados

- Las personas suelen demostrar una resistencia al cambio.
- Se requiere tiempo para realizar la asignación de los resultados de todas las variables en juego.

- Un alto número de variables puede derivar en que el contenido sea muy difícil de resumir o no sea posible proporcionar a primera vista una impresión general de la enfermedad.

Actualmente existen múltiples estándares que formalizan la estructura y creación de los informes estructurados, así como las características que deben cumplir las herramientas que se encarguen de la implementación práctica de los mismos. En este trabajo vamos a mostrar los dos estándares más reconocidos a nivel mundial y que se usan ampliamente en el ámbito de la imagen médica. Estos son DICOM y *Health Level Seven International (HL7)*.

2.1.1. DICOM

DICOM es un estándar nacido para proporcionar un protocolo de comunicación entre sistemas a la hora del manejo de información en el ámbito sanitario y de investigación. Estas reglas se hacen necesarias debido a que la interconexión entre la gran multitud de sistemas médicos que existen es compleja. Esto se debe a la especificidad de los sistemas que se han ido adaptando y desarrollando conforme la medicina se ha ido especializando en las distintas áreas que abarca[11]. En el estándar queda definido el formato de los archivos, la información necesaria para la identificación del paciente, además de la cabecera del objeto DICOM. Gracias a estas especificaciones todas las pruebas de imagen seguirán el mismo formato y llevarán asociada los mismos campos de información, independientemente del sistema o paciente.

Algunos de los servicios que ofrece el estándar son los siguientes:

- *DICOM Storage*: protocolos para el almacenamiento de imágenes.
- *DICOM Query & Retrieve*: protocolos para la búsqueda y recuperación de imágenes.
- *DICOM Print*: protocolos de impresión de imagen.
- *DICOM SR*: protocolos para los informes estructurados.

Este último servicio es el que nos interesa debido a que nuestro proyecto busca la implementación de los mismos, así que seguiremos la parte del estándar que nos conviene para codificar el informe.

En esta parte del estándar DICOM se han especificado las reglas para la correcta codificación, transmisión y almacenamiento del documento. DICOM SR especifica un documento compuesto por texto que puede tener enlaces a otra información como puedan ser imágenes, coordenadas espaciales o temporales, o unidades de medida para dosis administradas o mediciones de órganos y tumores. La forma en la que se estructura, y el uso de codificaciones basadas en terminologías estándar, permite la comprensión semántica de los datos en las historias clínicas actuales. La codificación nos permite que sea leído por quien sea, gracias a que el código significa lo mismo independientemente del país desde donde se esté leyendo, teniendo la certeza de que nos estamos refiriendo a lo mismo. También el uso de codificaciones nos permite realizar análisis y minería de datos con

los informes estructurados, ya que las etiquetas nos permiten hacer extracciones de documentos que se adecuen a nuestros parámetros de búsqueda. Pudiendo así por ejemplo, obtener todos los informes relativos a una patología, con una sintomatología específica o de un grupo de población concreto [12].

Los servicios de DICOM SR nos permiten introducir códigos estandarizados o propios que representen cada uno de los elementos de información que constituyen el informe. Además, establece los elementos que agrupan la información codificada en los denominados *Information Objects Definition* (IODs), que pueden ser de múltiples tipos: *Basic Text SR*, *Enhanced SR*, *Comprehensive SR*[2]. A modo de representación gráfica, es muy usual representar los informes estructurados en forma de árbol, donde tenemos un nodo raíz del documento del cual colgarán todos los demás nodos del árbol. Esta estructura resulta muy acertada ya que DICOM SR se introdujo para definir la semántica entre los objetos. DICOM SR no se centra en imagen pura, busca establecer el significado semántico de los distintos elementos de información que constituyen el informe, además esta jerarquía de árbol nos permite darle el significado deseado, pudiendo agrupar y desagrupar en diversos conjuntos de nodos que tengan relación y sean coherentes entre sí.

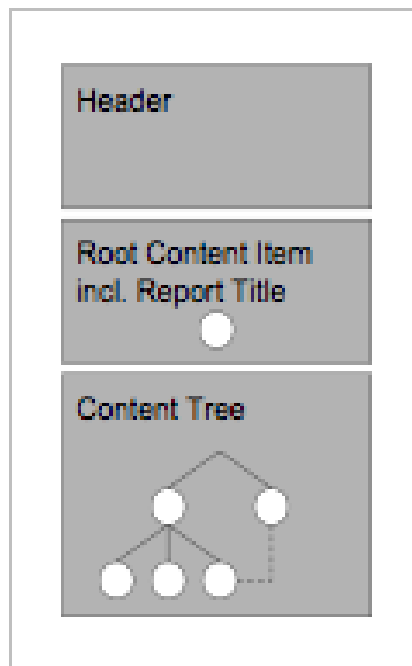


Figura 2.1: Estructura de un informe estructurado [1]

En la figura 2.1 podemos observar lo resaltado en el párrafo anterior. El documento constaría de tres elementos claves diferenciados. La cabecera, el nodo raíz y los nodos intermedios y finales.

La cabecera suele emplear los módulos necesarios para incluir los datos que permitan identificar el documento. Se distinguen 11 módulos pertenecientes a 5 áreas dependiendo de la información incluida. Estas áreas abarcan información del paciente, información del estudio, información de la serie, información de equipamiento e información del documento. No siempre se usaran todos los módulos ni todas las áreas en la cabecera, dependiendo del contexto será necesario incluir una información u otra. En el apartado de implementación se ex-

plicará cómo se ha desarrollado esta cabecera para los informes estructurados implementados[2].

El *root content item* empaqueta el título del documento. A partir de este nodo todos sus descendientes conforman lo que se ha denominado en la figura 2.1 como *content tree* (contenido del documento). El árbol generado debe ser único. No pueden relacionarse nodos de distintos árboles porque pertenecerían a informes estructurados diferentes y no es posible. Toda la información relativa al informe debe estar en un solo informe. Sí que se puede dar el caso de aparecer relaciones entre nodos del árbol, que si no son con sus antecesores o predecesores, formarían un grafo acíclico dirigido como se observa en la figura 2.2. Estos ciclos aparecen cuando nodos de ramas distintas tienen alguna relación entre sí.

A modo de ejemplo se muestra como se desglosa la búsqueda de un hallazgo maligno (si lo hay y si es maligno). El primer nivel se divide entre hallazgo (*mass*) y maligno (*malignancy*). El ítem "hallazgo" contiene distintas propiedades, para ser precisos tiene cuatro, de las cuales tres de ellas siempre aparecen y una de ellas aparece si el nodo paralelo a "masa" (*malignancy*) es evaluado a verdadero. Esta evaluación indica que se ha encontrado un hallazgo maligno, por lo que deberemos añadir una propiedad más al cuerpo para que indique los datos relacionados con la malignidad.

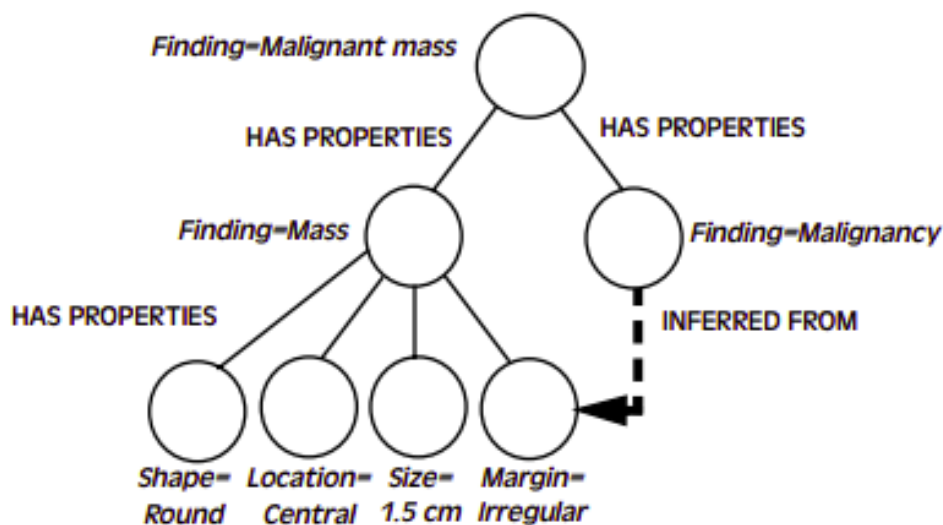


Figura 2.2: Árbol con referencias entre nodos [2]

Atendiendo a un artículo publicado por Rita Noumeir, profesora del departamento de ingeniería eléctrica en el *Natural Sciences and Engineering Research Council of Canada* (NSERC), se presenta a continuación una serie de ventajas que presenta DICOM SR, y que se han tenido en cuenta a la hora de evaluar el uso del mismo en el proyecto para que se adaptase a las necesidades del mismo [12].

- Consistencia de la información de la cabecera: debido a que el SR comparte la misma cabecera que las imágenes, la aplicación que genera el SR copia la información del encabezado de la imagen. Esto asegura la exactitud y consistencia de la información del paciente y del estudio.

- Archivado, transferencia y administración: el SR se puede transferir entre varias estaciones de trabajo de visualización y archivo DICOM como las imágenes. Además, se puede almacenar junto con las imágenes. Se puede exportar en un CD, un DVD o un archivo a largo plazo. Se queda con las imágenes dentro del mismo estudio. Este punto es muy útil en nuestro proyecto ya que deberá permanecer en el entorno sanitario para tener acceso al mismo y no sabemos con certeza las copias de seguridad que se realizan en cada hospital, así que es muy interesante que el propio objeto DICOM se comporte como un fichero y pueda moverse libremente.
- Referencias a versiones anteriores del mismo documento: además de hacer referencia a evidencias actuales y anteriores, la cabecera del informe puede incluir referencias a documentos anteriores, como informes anteriores o provisionales. Este enlace entre objetos DICOM permite retroceder a versiones anteriores del documento en el caso de que haya equivocaciones o queramos hacer una trazabilidad de la evolución del paciente.
- Semántica codificada: la información se encuentra en los nodos del árbol que se forma. Cada nodo tiene un nombre descrito por un código, que define su significado. En consecuencia, se adjunta una semántica a cada elemento de información. Por ejemplo, si tenemos un nodo con la información 'Número de ciclos: 3', significa que '3' es la medida del 'número de ciclos', que se especifica inequívocamente con un código definido para 'Número de ciclos'.
- Partes estructuradas y textuales: es posible crear un DICOM SR con un contenido que se compone de dos partes principales: una estructura detallada que puede ser procesada automáticamente por una aplicación de software, y una parte que puede estar compuesta por una o varias secciones de texto para ser leídas por un humano. La generación de estas dos secciones puede ser completamente transparente para el usuario. A partir de la misma información de entrada, ambas partes se pueden generar automáticamente. Este punto es interesante para nuestro proyecto debido a que en algunos campos se requiere que el profesional escriba algo libre, es decir, no debe seleccionar una opción de las existentes en una lista que se le ofrezca. Podemos estar hablando de casos en los que por ejemplo la causa de fallecimiento no se encuentra entre las que con más frecuencia se producen en el ámbito.

DICOM es un estándar cuyo uso es generalizado en el ámbito sanitario debido al dinamismo de sus versiones. Se trata de un estándar actualizado que se va adaptando a las nuevas necesidades del área, incluyéndose en cada nueva versión componentes y servicios adicionales que consiguen suplir carencias de versiones anteriores. Esta rápida adaptación al entorno es posible a su diseño modular y poco conexionado. Esta estructura permite añadir nuevos módulos fácilmente o editar ya existentes sin tener repercusiones en los módulos ya desarrollados. La constante actualización unida a la gran adaptación de DICOM a los servicios hospitalarios, tanto a nivel técnico como de servicios, hacen que sea un estándar interesante y útil que conviene utilizar para seguir unas pautas ampliamente reconocidas a nivel mundial, y sacar el máximo partido a nuestros servicios o desarrollos en cualquier otra organización distinta a la nuestra. Hacemos así que

la información entre centros pueda fluir correctamente siguiendo las especificaciones del estándar.

2.1.2. HL7

Para el establecimiento de nuevos estándares capaces de establecer pautas para la buena integración de los sistemas de salud existe la organización sin ánimo de lucro HL7. Se trata de una organización reconocida mundialmente fundada a finales de los años 80 que cuenta con participantes de 50 países distintos. Su organización se diversifica entre las distintas ramas propias de cada país, actuando como sedes independientes pero integradas en la propia organización. Podemos decir que HL7 garantiza la interoperabilidad entre sistemas de salud haciendo hincapié en cómo se organiza la información y en cómo esta se comunica entre las dos partes.

Los estándares que se agolpan bajo esta organización van destinados a la concreción de protocolos y métodos para el intercambio de información entre las aplicaciones del ámbito sanitario. Es por esto por lo que el 7 indica la capa 7 en el modelo OSI, porque es donde se ofrece el acceso a servicios a través de aplicaciones, así como las definiciones de los protocolos que se usan para intercambiar información [13]. Los cinco estándares primarios que se denominan de HL7 son: HL7 V2, HL7 V3, CDA (*Clinical Document Architecture*), HL7 FHIR (*Fast Healthcare Interoperability*) y CCOW (*Clinical Context Object Workgroup*). A continuación describiremos muy brevemente cada uno de ellos, y para finalizar la subsección analizaremos más en profundidad CDA, ya que está relacionado con la estructura y la semántica de los informes estructurados.

- HL7 V2: se trata de un estándar de mensajes para pautar el intercambio de información entre sistemas, es uno de los estándares de interoperabilidad más usado en el mundo. Los mensajes más frecuentemente usados bajo ese estándar son los de órdenes (ORM), resultados (ORU) y gestión de pacientes (ADT).
- HL7 V3: fue formalizado para intentar cubrir los aspectos negativos de la anterior versión. En la versión 2, dado que es una versión muy flexible, surgía la problemática de que ante un pequeño cambio para actualizarla, se requería un gran estudio previo y multitud de cambios para que esa modificación siguiera integrándose con las demás partes del estándar.
- CDA: es el estándar que se encarga de estandarizar la estructura y el contenido de los informes. Tras esta lista entraremos a analizarlo más en profundidad.
- HL7 FHIR: se trata de una combinación entre HL7 V2, HL7 V3 y CDA para el intercambio de información. Puede estructurarse tanto en formato *Extensible Markup Language* (XML) como en *JavaScript Object Notation* (JSON).
- CCOW: estándar con el objetivo de facilitar la integración de aplicaciones haciendo uso de técnicas de unificación y sincronización de los datos del usuario que se encuentren distribuidos entre distintos sistemas en una única interfaz de usuario.

CDA

Como hemos dicho anteriormente, HL7/CDA es el estándar que se encarga de los documentos clínicos y cuyo objetivo es agilizar el traspaso de información en informes entre pacientes y centros sanitarios.

Según el estándar, los documentos deben tener estas seis características [14]:

- **Persistencia:** un documento clínico debe continuar inalterado por un período de tiempo definido por los requisitos locales y reglamentarios.
- **Administración:** un documento clínico debe ser mantenido por una persona u organización encargada de su cuidado.
- **Potencial de autenticación:** un documento clínico es un conjunto de información que está destinado a ser legalmente autenticado, debe tener validez legal.
- **Integridad:** la autenticación de un documento clínico se aplica al conjunto y no se aplica a partes del documento sin el contexto completo del documento.
- **Legibilidad humana:** un documento clínico debe ser legible por personas.

Los documentos que se rigen bajo CDA deben componerse de dos campos principales similares a los existentes en el estándar DICOM. En primer lugar consta de una cabecera donde se debe incluir toda la información identificativa del documento, aquella que pueda ser objeto de filtrado para poder encontrar el documento. Entre los campos que podemos encontrar en la cabecera, se encuentra el título, la fecha, el autor, el paciente o las autorizaciones que permiten el documento. Tras esta cabecera encontramos el cuerpo del documento que puede estar definido en tres niveles, ordenados de mayor a menor flexibilidad en cuanto a estructuración. Estos niveles son:

- **Nivel 1:** se corresponde a documentos que no contengan contenido estructurado, es decir, son documentos que a su vez llevan anexados otros archivos o documentos, es el nivel más flexible.
- **Nivel 2:** encontramos contenido que puede estar estructurado en niveles donde algunas de las secciones que se tratan pueden estar codificadas, este es el nivel medio de flexibilidad.
- **Nivel 3:** el contenido está totalmente estructurado y codificado, es el nivel más rígido.

A continuación se muestra un extracto de lo que podría ser un documento estandarizado en CDA.

```
<?xml version="1.0"?>
<!DOCTYPE levelone PUBLIC "-//HL7//DTD CDA Level One 1.0//EN"
"levelone_1.00dtd">
<levelone>
```

```

<clinical_document_header>
  <id EX="a123" RT="2.16.840.1.113883.3.933"/>
  <set_id EX="B" RT="2.16.840.1.113883.3.933"/>
  <version_nbr V="2"/>
  <document_type_cd V="11488-4" S="LOINC"
    DN="Consultation note"/>
  <origination_ddtm V="2000-04-07"/>
  <confidentiality_cd ID="CONF1" V="N" S="HL7_Confidentiality"/>
  <confidentiality_cd ID="CONF2" V="R" S="HL7_Confidentiality"/>
  <document_relationship>
    <document_relationship.type_cd V="RPLC"/>
    <related_document>
      <id EX="a234" RT="2.16.840.1.113883.3.933"/>
      <set_id EX="B" RT="2.16.840.1.113883.3.933"/>
    </related_document>
  </document_relationship>
  ...

```

A modo de conclusión indicar que gracias a este estándar, HL7 ha logrado introducirse en el campo de la definición de la estructura y la semántica de los documentos clínicos. Este primer estándar ha logrado la estandarización de las notas clínicas comunes, como la historia y los exámenes físicos, los resúmenes de alta y las notas de progreso. Deliberadamente tiende a omitir ciertas semánticas avanzadas y complejas, tanto para fomentar una implementación amplia como para dar tiempo a que esta semántica compleja se elimine. Se espera que las futuras versiones de CDA incluyan capas más profundas de la arquitectura que codifiquen una semántica más rica y que el estándar actual sirva como un trampolín, permitiendo a los usuarios implementar estas versiones progresivamente.

2.2 Terminologías estándar

La organización estructural en forma de árbol del documento de las distintas variables a evaluar otorga beneficios, pero no debemos olvidar que DICOM-SR se introdujo para dar un significado a cada uno de esos ítem de información, y aquí es donde entran en juego las terminologías, las cuales nos aportan la consistencia necesaria para poder codificar la información de una forma efectiva. Según el estándar DICOM, una entrada o nodo del árbol debe codificarse de la siguiente forma:

- *Code value*: el código necesario para especificar el término.
- *Coding scheme designator*: la terminología usada de la que proviene el código anterior para definir el término.
- *Code meaning*: descripción del término que se está definiendo. Esta descripción es legible y debe ser entendida y procesada mediante el lenguaje natural.

Con estas directrices, imaginemos que queremos codificar el órgano llamado corazón. A modo de ejemplo podríamos tener una terminología que se llamase -termOrganos- y dentro de esta terminología tuviésemos el código 23 correspondiendo a la descripción -Corazón-. En este caso siguiendo el estándar DICOM podríamos codificar este nodo de la siguiente forma (23, termOrganos, Corazón). Donde tenemos una terna que representa con cada una de las entradas explicadas anteriormente.

Como hemos dicho en la sección anterior, estas terminologías pueden estar estandarizadas o podemos crear la terminología propia que consideremos, si algún término no se encuentra estandarizado en ninguna terminología estándar ya existente. Las terminologías más usadas son SNOMED-CT (*Systematized Nomenclature of Medicine - Clinical Terms*) y LOINC también del inglés (*Logical Observation Identifier Names and Codes*). Sin embargo hay muchas otras que existen y de las cuales se ha hecho uso en este proyecto. A continuación detallamos las dos terminologías que más peso tienen en nuestro proyecto que son SNOMED-CT y RadLex (*Radiology Lexicon*).

2.2.1. SNOMED CT

Cuando hablamos de SNOMED CT estamos hablando de una terminología que recoge léxico clínico y que es considerada como la terminología con mayor extensión, mejor exactitud y mayor consideración a nivel mundial. Como muestra de ello en la tabla 2.1, se recogen un conjunto de métricas que están disponibles en BioPortal¹, una web que indexa las ontologías disponibles para que el usuario pueda realizar una búsqueda de los términos a codificar. El léxico cifrado en ella se puede usar para codificar, rescatar, comunicar y realizar un estudio de los datos clínicos para mejorar la toma de decisiones en base a una representación adecuada de la información. Está estructurada por términos, descripciones de términos y relaciones entre los mismos. Agrupados en grupos y subclases permiten una búsqueda eficaz de elementos relacionados. El fin es representar la información clínica con el mayor rigor posible [15].

Número de clases	357 533
Individuales	0
Propiedades	218
Profundidad máxima	27
Número máximo de hijos	1173
Número medio de hijos	4
Clases con un único hijo	41 784
Clases con más de 25 hijos	2880
Clases sin definición	352 128

Tabla 2.1: Métricas de SNOMED CT en junio de 2020 [3]

La estructuración que ofrece SNOMED CT es sencilla. Una estructura por clases que heredan de otras clases y que a su vez contienen hijos con propiedades

¹<https://bioportal.bioontology.org>

de sus antecesores. En la figura 2.3 se puede observar un esquema de la organización. En este caso vemos como el término *Animal virus*, está contenido dentro de *Virus*, que a su vez está contenido en *Organism*.

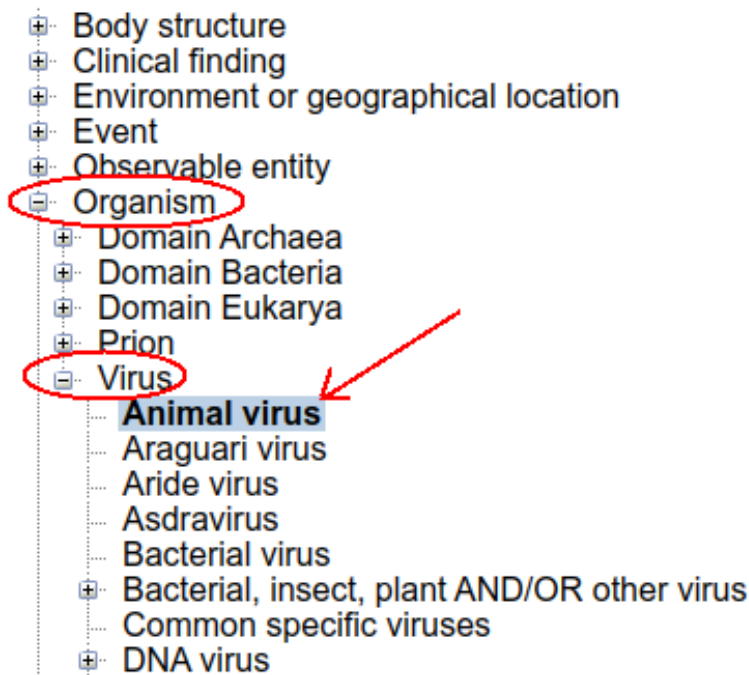


Figura 2.3: Anidamiento de clases en SNOMED CT

Una vez hemos encontrado el término que nos interesa, la información que nos resultará útil a la hora de desarrollar el informe estructurado será: el *preferred name*, el identificador del término y la clase a la que pertenece. Esta información se encuentra en la ontología en la forma en que se muestra en la imagen 2.4. En el apartado de desarrollo del módulo se indicará como integrar esta información en el código haciendo uso de las librerías apropiadas.

Preferred Name	Animal virus
Synonyms	Animal virus (organism)
ID	http://purl.bioontology.org/ontology/SNOMEDCT/50287002
Active	1
altLabel	Animal virus (organism)

Figura 2.4: Información del término

Para finalizar este apartado, indicar que BioPortal ofrece una lista de los proyectos registrados en el propio portal por los investigadores en los que se está haciendo uso de las distintas terminologías. En el caso de SNOMED-CT, algunos de estos proyectos se presentan a continuación.

- *Educational Medical Materials annotator*: se trata de un proyecto llevado a cabo en la universidad de Warwick (Inglaterra) para el estudio de las relaciones entre el material médico educativo que está distribuido en Internet.

- *Epidemic Marketplace*: este proyecto se realiza en la facultad de ciencias de la universidad de Lisboa (Portugal). Consiste en una plataforma de integración donde los datos epidemiológicos y los recursos relacionados se almacena, gestionan y ponen a disposición, con el objetivo de fomentar la colaboración.
- *Integrative Genomics*: proyecto de la universidad estatal de Ohio (Estados Unidos de América) que desarrolla una base de datos híbrida de gráficos y ontología en un contenedor llamado ontografía. El objetivo es aprovechar las mejores características de las ontologías e integrarlas con los datos más apropiados de las bases gráficas de datos.

2.2.2. RadLex

Esta terminología ha sido diseñada por la *Radiological Society of North America* (RSNA) animada por los beneficios que aporta una utilización de un lenguaje común entre radiólogos para comunicar resultados de diagnósticos. RadLex conforma un conjunto de términos para ser usados en la confección de informes radiológicos, en la toma de decisiones, explotación de datos, en el ámbito educativo y en la investigación. La creación de esta terminología estandarizada ha tenido el apoyo del *National Institute of Biomedical Imaging and Bioengineering* (NIBIB) y el *cancer Biomedical Informatics Grid* (caBIG) [16].

Al igual que sucede con SNOMED-CT, está estructurada por términos, descripciones de términos y relaciones entre los mismos. Agrupados en grupos y subclases permiten una búsqueda eficaz de elementos relacionados cuyo fin es representar la información clínica con el mayor rigor posible.

Número de clases	46 636
Individuales	23 922
Propiedades	52
Profundidad máxima	20
Número máximo de hijos	780
Número medio de hijos	3
Clases con un único hijo	695
Clases con más de 25 hijos	94
Clases sin definición	42 903

Tabla 2.2: Métricas de RadLex en junio de 2020 [4]

Como se puede ver en la tabla 2.2, esta terminología contiene un conjunto de términos más reducido, no es tan amplio como SNOMED-CT y esto se debe al área que abarca cada una. Mientras SNOMED-CT contiene términos usados en cualquier área de la medicina, RadLex se limita únicamente a la radiología. Es por esto, por lo que para este proyecto es interesante utilizarla, debido a que muchos términos eran demasiado específicos como para encontrarlos en SNOMED CT, pero sin embargo como son de uso habitual en el área de imagen médica hemos podido encontrarlos sin problemas en RadLex.

Para finalizar este apartado se ofrece una lista de los proyectos registrados en el propio Bioportal por los investigadores en los que se está haciendo uso de las distintas terminologías. En el caso de RadLex, algunos de estos proyectos se presentan a continuación.

- *AdaptEHR*: se trata de un proyecto llevado a cabo en la universidad de California (Estados Unidos de América) con el objetivo de construir un marco de trabajo que permite la integración de datos de diferentes fuentes en el registro del paciente y, posteriormente, la presentación en función del contenido de los informes clínicos de los pacientes y las necesidades de información del usuario.
- *Mobile RadLex*: este proyecto ha sido realizado por un usuario independiente. Consiste en un navegador móvil de RadLex para el teléfono Android. Permite búsquedas de términos en RadLex de forma similar a la que ofrece BioPortal.
- *Neural ElectroMagnetic Ontologies*: proyecto de colaboración entre la universidad de Oregón y la universidad estatal de Georgia (Estados Unidos de América) que desarrolla ontologías y herramientas basadas en los términos recogidos para las pruebas de electroencefalografía y magnetoencefalografía. Estos recursos se utilizarán para apoyar la representación, clasificación y metaanálisis de datos electromagnéticos cerebrales.

2.2.3. Otras

En esta subsección hablaremos de otras terminologías que resultan de interés para el proyecto ya que la información con la que nosotros trabajaremos, que será explicada en el capítulo 3, no es proveniente de un único campo de la medicina. En este proyecto, en la elaboración de los e-forms, se trabaja con resultados de imágenes médicas, pruebas de laboratorio, pruebas moleculares, clasificación de estadios de cáncer según distintos organismos internaciones, etc. Para estandarizar todo el vocabulario empleado en todo este amalgama de pruebas y relaciones de distintas áreas de la salud se deberá recurrir a otras terminologías diferentes a Radlex o SNOMed-CT que se presentan a continuación.

Para la información relativa a los estudios moleculares recurriremos a LOINC (*Logical Observation Identifiers Names and Codes*). LOINC contiene terminología estándar para identificar pruebas de laboratorio como la medición del colesterol en sangre, orina, hemoglobina, etc. [17].

Para la estandarizaciones de términos relacionados con medicamentos o tratamientos haremos uso de MedDRA (*Medical Dictionary for Regulatory Activities*). Es una terminología usada en la industria farmacéutica en los procesos de investigación preproducción y en las fases de postproducción en la farmacovigilancia [18].

Finalmente, para términos específicos relacionados directamente con la oncología acudiremos a la ontología creada por el *National Cancer Institute Thesaurus* (NCIT), donde se recoge vocabulario para el cuidado clínico, de investigación, así como actividades administrativas y de clasificación de pacientes [19].

2.3 Uso de informes estructurados en otros proyectos

2.3.1. TRENCADIS y CVIMO

TRENCADIS (*Towards a gRid ENvironment to proCess and shAre DIcom objectS*) es un proyecto desarrollado por el Instituto de Aplicaciones de las Tecnologías de la Información y de las Comunicaciones Avanzadas (ITACA) de la Universidad Politécnica de Valencia (UPV). Se trata de un *middleware* que permite una gestión eficiente y segura del intercambio de información (imágenes médicas o informes estructurados) entre los distintos centros hospitalarios. El proyecto en sí no se ocupa concretamente de la creación de estos informes estructurados que generará el módulo que se pretende desarrollar en este proyecto, pero se centra en el montaje de una infraestructura para poder crear módulos que gestionen el intercambio de información [20]

Este proyecto se integró con CVIMO (*Valencian Cyberinfrastructure of Oncological Medical Images*) para desarrollar un banco de imágenes médicas que pudiera ser consultado por diversos centros hospitalarios de la Comunidad Valenciana. Estas imágenes se indexaban mediante informes estructurados que eran almacenados en bases de datos relacionales [21].

2.3.2. DICOM SR en redes de telemedicina

Este proyecto se centra en el diseño de informes estructurados siguiendo el estándar DICOM SR para satisfacer la demanda de redes de telemedicina a gran escala. Para llevar a cabo este proyecto, se ha desarrollado un vocabulario estructurado específico en el contexto de la cardiología de acuerdo con los estándares y directrices de la Sociedad Brasileña de Cardiología (SBC).[22]

La implantación de estos informes estructurados se ha llevado a cabo sobre el Sistema Integrado de Telemedicina del Estado de Santa Catarina en el sur de Brasil. Se trata de una red que conecta a 401 instituciones de atención médica en 291 municipios en todo el estado y su expansión sigue avanzando. Entre los distintos servicios que ofrece se encuentra el diagnóstico a distancia, segundas opiniones, informes en tiempo real y colaborativos, educación continua, entre otros. En esta red se computan alrededor de unos 75 000 exámenes médicos al mes, y los usuarios tienen acceso a imágenes DICOM y estudios de señal a través de plataformas de software especialmente desarrolladas basadas en la web y para teléfonos inteligentes a las que se accede a través de un portal integrado que permite visualización, análisis, discusión y redacción de informes de los exámenes. Todos los servicios son accedidos vía web como se muestra en la figura 2.5.



Figura 2.5: Cuadro de mandos del Sistema Integrado de Telemedicina del Estado de Santa Catarina

El proyecto se centró en la presentación del informe estructurado (visualización e impresión), la interfaz de usuario y la estructura de datos utilizada para almacenar el informe, así como el algoritmo responsable de explotar los datos.

En el modelo propuesto, los médicos no hacen uso de la interfaz únicamente para escribir el informe en un estilo de texto plano natural, sino también tienen la posibilidad de vincular y adjuntar uno o más descriptores al informe final. Se comenta también que para la mayoría de los informes estructurados se dividen en tres campos predefinidos: Descripción del estudio, Resultados y Conclusión. Al final, la información adicional que se desea introducir bajo un formato no establecido se adjunta automáticamente al archivo DICOM SR.

2.4 Crítica al estado del arte y propuesta

Tras el análisis de los proyectos anteriores, así como de los proyectos que se han mencionado para cada una de las terminologías, se ha visto que el proyecto que más se asemeja al campo en el que estamos trabajando es el proyecto TREN-CADIS. La idea difiere en que TREN-CADIS propone una infraestructura para integrar imágenes médicas e informes estructurados que están almacenados en texto plano entre distintos centros hospitalarios, mientras que nuestra propuesta consiste en un módulo que permita la creación de objetos DICOM-SR, que seguirán una jerarquía según el estándar DICOM, en una plataforma ya existente que de por sí ya integra y da accesibilidad a la información de los distintos pacientes de NB y DIPG a los distintos centros que participan en el proyecto PRIMAGE.

El valor añadido de este TFM es que a nivel de transferencia de información entre entidades de distintos países no se han encontrado proyectos que traten de estandarizar informes estructurados.

CAPÍTULO 3

Análisis del problema

En este capítulo haremos un análisis de la problemática a la que nos enfrentamos y cómo con la consecución de los objetivos que se han definido al inicio del documento podemos dar una solución a estos problemas. En primer lugar se presentará la metodología que se usa actualmente, presentaremos visualmente la plataforma del proyecto PRIMAGE elaborada por QUIBIM y haremos un repaso por algunos de los campos del formulario web que se emplea para realizar la recolección de datos de pacientes. En segundo lugar veremos cómo se guarda esta información y estudiaremos la estructura y el formato que se utiliza para tener la información necesaria que nos permita cumplir con los objetivos relativos a la selección de herramientas para la implementación de informes estructurados y el diseño de plantillas DICOM-SR a partir de los formularios existentes.

3.1 Metodología actual

Como se ha comentado, actualmente el proyecto PRIMAGE cuenta con una plataforma web para la recogida de datos de los pacientes diagnosticados de NB o DIPG a través de una serie de formularios (e-forms). Esta plataforma cuenta con un apartado que permite introducir datos clínicos asociados a los distintos pacientes. Este contiene toda la información relativa al paciente dado, al diagnóstico, al tratamiento y al seguimiento de la enfermedad. Para esta tarea de recolección cuentan con un formulario web (e-forms) donde los especialistas de los hospitales colaboradores deben ir introduciendo la información de los campos de estudio que resultan de interés.

Actualmente, tras la autenticación en la plataforma se accede a un menú que te permite seleccionar el proyecto sobre el que deseas trabajar. En la figura 3.1, vemos como nos ofrece la opción de trabajar con pacientes de NB, de DIPG o ambos.

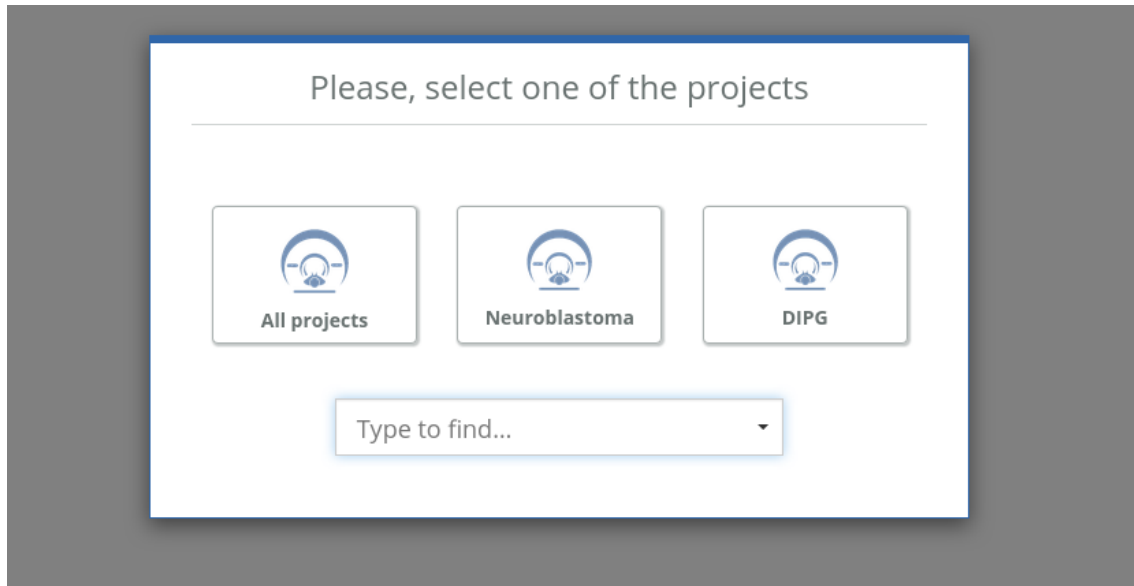


Figura 3.1: Selección del proyecto en la plataforma PRIMAGE

Una vez se ha seleccionado el proyecto sobre el que queremos trabajar, pasaremos a la pantalla que se muestra en la figura 3.2. En esta sección tenemos todos los estudios que los clínicos han ido introduciendo. En el caso de querer incluir algún paciente nuevo, en el menú de la izquierda tendríamos la opción de crear un nuevo paciente. Como se ve, cada paciente va identificado por un código que se le da a la hora de crearlo y que se empleará para reidentificar al paciente en el proceso de psudonimización del mismo. En ningún momento se utiliza el nombre del paciente para garantizar el cumplimiento de las leyes estatales y europeas relativas a la protección de datos.

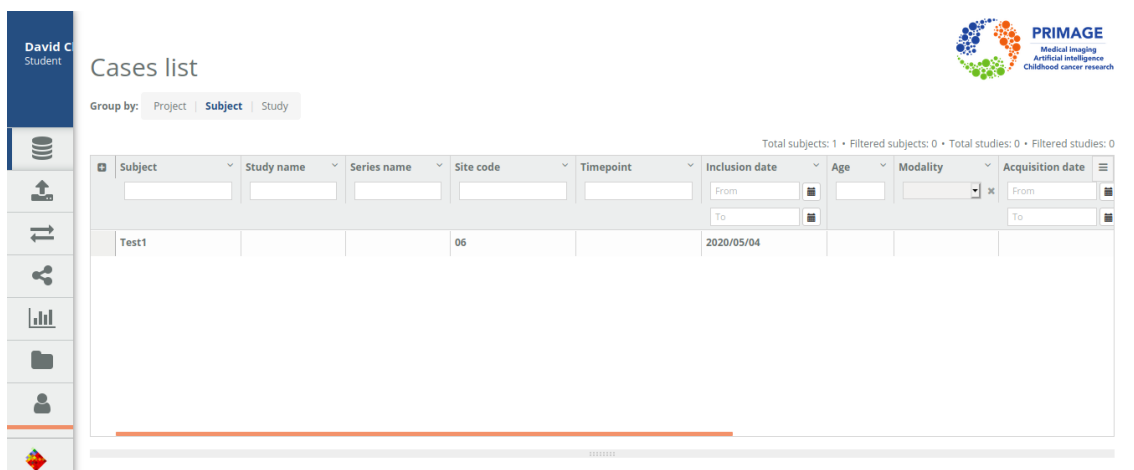


Figura 3.2: Lista de estudios creados en la plataforma PRIMAGE

Una vez creado el paciente, el usuario ya puede empezar a introducir la información correspondiente haciendo uso de un formulario como se muestra en la figura 3.3. Vemos como este formulario contiene un control de validación donde algunos de los campos son estrictamente necesarios, y no será posible cerrar y validar el formulario si no se completan todas las variables de obligada asignación.

Finalmente la plataforma permite realizar una descarga de los datos en formato JSON para poder trabajar con ellos.

Figura 3.3: Campos del formulario de paciente de la plataforma PRIMAGE

Introducida la plataforma PRIMAGE de la que se dispone para la introducción de datos, lo que se quiere implementar es una opción similar a la descargar JSON, pero que en vez de obtener un objeto JSON con la información que se ha ido rellenando, se obtenga un objeto DICOM-SR correspondiente a un informe estructurado. De esta forma los profesionales sanitarios podrían descargar esa información de los pacientes y les sería presentada en sus propios equipos en forma de informe, pudiendo así tener mejor organizada la información dado que sigue un estándar. DICOM-SR empaqueta los informes en objetos similares que pueden ser visualizados en equipos hospitalarios y podemos prescindir de las hojas de cálculo o los documentos que se usan actualmente para guardar la información en estos centros.

3.2 Estructura de la información

Tras la inserción de toda la información relativa al paciente haciendo uso del formulario web de la plataforma de PRIMAGE, la información es almacenada en una base de datos no relacional, MongoDB. MongoDB es un sistema de base de datos NoSQL orientado al almacenamiento de datos en documentos JSON, siendo totalmente contrario a las bases de datos relacionales que optan por una estructura relacional de tablas para guardar la información.

JSON

El uso de este formato empezó a ser significativo en el año 2000, sin embargo, la especificación oficial no llegó hasta el 2013. Es un formato empleado para el intercambio de datos estructurados en forma de texto como si fuera un objeto *JavaScript* (JS). Esta estructura facilita el procesamiento de la información, debido a que su conversión a objetos JS resulta directa.

{

```

"obj_collection": [
  {
    "id" : "value0"
    "key1": "value1",
    "key2": "value2"
  },
  {
    "id" : "valueX"
    "key1": "value1",
    "key2": "value2"
  },
]
}

```

El objeto JSON con el que trabajamos, se divide desde la raíz en cuatro ramas correspondientes a la información del paciente (*patient_data*), diagnóstico (*diagnosis*), tratamiento y evaluación (*treatment_and_evaluation*) y seguimiento (*follow_up*). En la figura 3.4 hemos usado un visor JSON para agilizar el análisis de la información. Como se puede observar, se trata de una colección de objetos que a su vez está compuesto por otros objetos que contienen información referente a cada una de las ramas anteriormente mostradas. Si representamos el objeto JSON en forma de árbol, vemos como las distintas ramas tienen profundidades diferentes. Por ejemplo, la parte de información del paciente acaba con el valor de la variable *sex*, pero sin embargo en la sección laboratorio, que se encuentra anidada en diagnóstico, la profundidad es mayor.

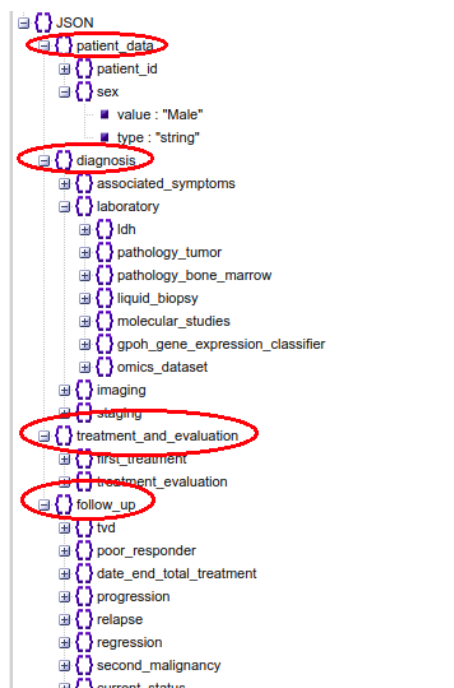


Figura 3.4: Objeto JSON marcando las cuatro áreas de información del paciente

Para finalizar el capítulo de análisis podemos establecer que en base a la plataforma actual y a cómo está dispuesta la información de los datos que deben

aparecer en los informes estructurados, se requiere la implementación e integración de un módulo en la misma plataforma que genere un objeto DICOM-SR, de forma que pueda ser visualizado y almacenado en los dispositivos médicos a partir de los fichero JSON correspondientes a los e-form introducidos.

CAPÍTULO 4

Diseño de la solución

En este capítulo se describe el proceso de diseño de la solución. En primer lugar se presentarán las tecnologías que serán empleadas para el desarrollo. Entre las tecnologías analizaremos las librerías necesarias para conseguir una implementación que cumpla con el estándar DICOM y las librerías necesarias para trabajar con la forma en la que está estructurada la información. En segundo lugar presentaremos la arquitectura del informe estructurado, donde veremos la información necesaria para codificar los términos extraídos y el uso de los distintos tipos de variables empleados. Al finalizar este capítulo tendremos la información suficiente para haber cumplido con el objetivo primero relacionado con la selección de herramientas para la implementación de informes estructurados y los primeros resultados del proyecto que permitirán cumplir con el objetivo segundo sobre el diseño de las plantillas DICOM-SR.

4.1 Tecnologías empleadas

4.1.1. La librería DCMTK

Para el desarrollo del proyecto se ha optado por DICOM-Toolkit (DCMTK). Este es un conjunto de librerías con funcionalidades que permiten implementar soluciones informáticas que cumplan con el estándar DICOM. Al igual que el estándar DICOM, DCMTK ofrece la implementación de distintos servicios como son la creación y conversión de imágenes a archivos DICOM, la gestión de las comunicaciones en red de imágenes médicas, así como la gestión de almacenamiento en servidores de los distintos objetos DICOM que fueran implementados. Se trata de *software* abierto que se encuentra a disposición de todo el mundo para su uso y ha sido desarrollado en C y C++ [23].

La librería fue desarrollada por el *Oldenburger OFFIS - Institut für Informatik* que se encuentra en la localidad alemana de Oldenburg. El centro divide sus vías de investigación en energía, salud, transporte e industria. Este proyecto nace fruto de la rama sanitaria con el objetivo de crear tecnologías estándares que permitan la integración de los sistemas independientemente de la localización, y para ello plantean el seguimiento de estándares como DICOM o HL7 enmarcados en el *framework Integrating the Healthcare Enterprise (IHE)* [24].

DCMTK se ha expuesto y presentado en diversas conferencias de DICOM para proporcionar servidores de listas de trabajo y almacenamiento de imágenes. Actualmente existen sistemas desarrollados con esta librería por hospitales y empresas de todo el mundo para una amplia variedad de propósitos, como son herramienta para pruebas, componentes en proyectos de investigación, prototipos o productos comerciales.

El software DCMTK se puede compilar en Windows y en una amplia gama de sistemas operativos Unix, incluidos Linux, Solaris, FreeBSD, OpenBSD, MacOS X y NetBSD. Se proporcionan todos los *scripts* de configuración necesarios y los archivos *MAKE* del proyecto.

4.1.2. La librería nlohmann/JSON

Como se ha comentado en la sección 3.2. La fuente de datos con la que trabajamos es un objeto JSON del cual deberemos acceder a los valores de cada una de las claves que sean necesarias. Para trabajar de una manera directa con esta estructura de datos vamos a hacer uso de la librería nlohmann/JSON para C++ para el correcto análisis de objetos JSON.

Para realizar la elección de esta librería hemos analizado los resultados de un *benchmark* que evalúa el rendimiento de 41 bibliotecas C/C++ de código abierto con capacidades para analizar y generar objetos de tipo JSON. Las características de rendimiento vienen medidas en función de la velocidad, la memoria empleada y el número de líneas de código de la librería [25].

La librería nlohmann/JSON obtiene un 96% de puntuación media, lo que le sitúa en la cuarta posición de la clasificación por detrás de otras librerías como RapidJSON_FullPrec¹, Configuru² y taocpp/json³. A continuación, se recoge la recopilación de las pruebas que se ejecutan para valorar cada una de las librerías.

Benchmark	Descripción
<i>Parse Validation</i>	Identificar si un JSON es correcto o incorrecto
<i>Parse Double</i>	66 objetos JSON con valores decimales en un array son analizados
<i>Parse String</i>	9 objetos JSON con valores de tipo cadena en un array son analizados
<i>Roundtrip</i>	27 objetos JSON compactos (en una línea) son analizados y desenrollados

Tabla 4.1: Pruebas para medir la conformidad [4]

¹<https://github.com/Tencent/rapidjson>

²<https://emilk.github.io/Configuru/html/index.html>

³<https://github.com/taocpp/json>

Benchmark	Descripción
<i>Parse</i>	Analizar en memoria el objeto JSON en estructura de árbol (DOM)
<i>Stringify</i>	Serializar DOM en JSON compactos
<i>Prettify</i>	Serializar DOM en JSON desenrollados
<i>Statistics</i>	Contar número de tipos de JSON y número total de elementos en el DOM
<i>Sax Round-trip</i>	Generación de eventos a partir de objeto JSON
<i>Sax Statistics</i>	Generación de eventos para extraer estadísticas.
<i>Code size</i>	Tamaño del ejecutable en <i>bytes</i>

Tabla 4.2: Pruebas para medir el rendimiento [4]

Los parámetros que se han medido para todas y cada una de las pruebas son el tiempo en milisegundos, el consumo de memoria en bytes para generar la estructura resultado, el pico de consumo de memoria durante el proceso y la cantidad de memoria asignada al proceso.

Entre los resultados publicados en la web podemos encontrar que la librería *nlohmann* no es la más rápida, sigue siendo más lenta que las 3 opciones con nota media de 96 %, pero destaca en consumo de memoria. Desde el punto de vista de nuestro proyecto se considera más importante este aspecto, ya que tenemos una gran cantidad de variables y queremos que nuestro módulo consuma poca memoria ya que carece de sentido mejorar la velocidad mínimamente para sacrificar memoria cuando nuestra aplicación no necesita ser extremadamente rápida.

La media de velocidad de *nlohmann* se sitúa en 72 milisegundos mientras que *taocpp/json* se sitúa en 41 y *RapidJSON_FullPrec* en 16. La media de ganancia con respecto a estas dos sería de unos 44 milisegundos, dato que no compensa si comparamos el consumo de memoria. La media de consumo de memoria de *nlohmann* es de 9.9 MB mientras que *taocpp/json* se sitúa en 11.5 MB, *Configuru* con 20 MB y *RapidJSON_FullPrec* en 5MB. *Nlohmann* quedaría segunda en cuanto a consumo de memoria 5 MB por encima de la mejor. Se debe tener en cuenta también que el desarrollador del *benchmark* es el mismo desarrollador que el de la librería ganadora y aunque se intenta que el desarrollo del *benchmark* sea lo más objetivo y justo, los procedimientos, las pruebas y los datos han sido escogidos por el mismo desarrollador, es por eso que anima a probar nuevos conjuntos de datos y nuevas pruebas para comparar resultados. Ante esta situación y por las razones que se listan a continuación se ha decidido de hacer de *nlohmann* la librería para tratar objetos JSON en nuestro proyecto.

Esta librería ha sido elegida por las siguientes características:

- **Sintaxis:** esta librería recuerda al lenguaje de programación Python, el método de acceso a las claves y valores del objeto JSON es intuitivo, no se utiliza un código farragoso o poco estructurado difícil de comprender.
- **Integración:** a diferencia con *DCMTK*, esta librería es muy simple en cuanto a instalación. Únicamente contiene un archivo *json.hpp* que puede ser importado desde el código del desarrollo y funcionar directamente tras la compilación.
- **Memoria:** debido a que en el proyecto se utiliza una gran cantidad de variables e información, necesitamos que con cada una de las asignaciones a

variables que hacemos no consumamos muchos recursos, ya que es un aspecto a tener en cuenta para no quedarse sin memoria y provocar un fallo de ejecución.

4.2 Arquitectura del informe estructurado

El primer paso a realizar tras el análisis de la información con la que tenemos que trabajar es definir una estructura y plasmarla de forma visual para que podamos reproducir las relaciones entre los nodos del árbol que deben generarse para ensamblar el objeto DICOM-SR. Para ello tenemos 338 variables que se recogen en el formulario web. Estas variables pueden ser asignadas de forma abierta o cerrada, es decir, hay variables en las que el valor asignado será una entrada de texto libre y otras en las que la asignación se hará haciendo uso de *checkboxes*, *radio button* o listas desplegadas con opciones predeterminadas. En función del tipo de entrada debemos seleccionar un tipo de dato para asignarlo.

El resultado de esta primera fase es presentado en el anexo C. El anexo presenta unas tablas donde se especifican las secciones y las variables que servirán para generar el informe estructurado con la siguiente estructura:

```
NombreVariabe [Terminología, ID, TipoDeDato]
```

En primera instancia, lo que se ha realizado es identificar y codificar todos los términos (tanto variables, como valores posibles) en base a terminologías estándar cuyos valores estuvieran registradas para poder rellenar los campos Terminología e ID. Para ello se ha hecho uso del buscador de clases delBioportal⁴. Este buscador nos permite ingresar un término y ofrecernos una lista de las terminologías que lo recogen como se puede ver en la figura 4.1.

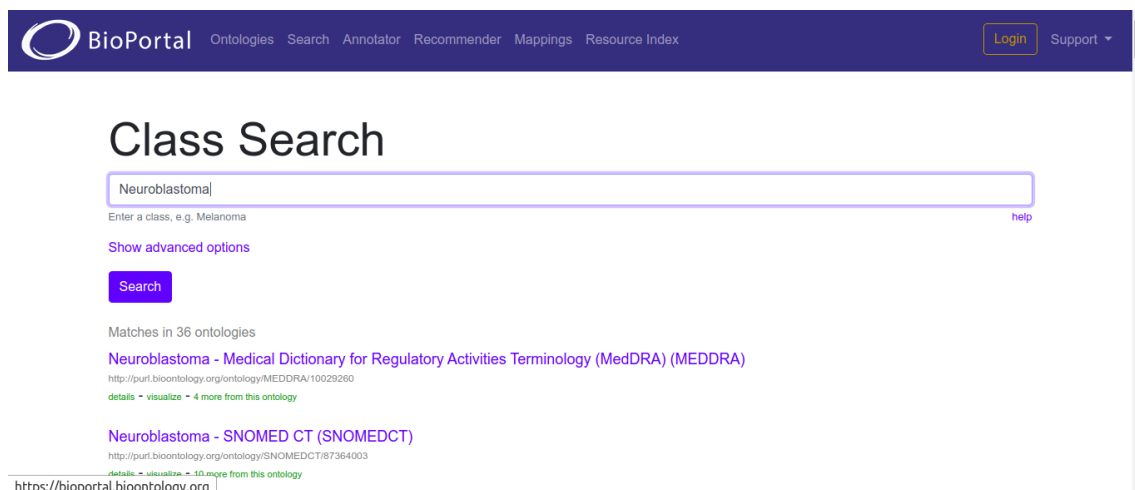


Figura 4.1: Resultado de búsqueda del término neuroblastoma

Una vez hemos obtenido que el término que nos interesa debemos rescatar el identificador del mismo (figura 4.2). Este código es necesario indicarlo junto

⁴<https://bioportal.bioontology.org>

con la terminología empleada para que a la hora de referirnos a una variable, independientemente del idioma, acudamos a esa terminología y sepamos exactamente a qué nos estamos refiriendo. El uso de la semántica otorga un significado a la información, conseguimos catalogar las variables como si de un diccionario se tratase y en donde podemos consultar qué código pertenece a qué término.


Details	Visualization	Notes (0)	Class Mappings (107)	
Preferred Name	Neuroblastoma			
Synonyms	Neuroblastoma (morphologic abnormality)			
ID	http://purl.bioontology.org/ontology/SNOMEDCT/87364003			
Active	1			
	Neuroblastoma (morphologic abnormality)			
altLabel	Central neuroblastoma			
	Sympathicoblastoma			
	NB - Neuroblastoma			

Figura 4.2: ID del término neuroblastoma

Uno de los problemas más comunes, es que tengamos un nombre de variable o un valor que queramos codificar que no esté registrado en ninguna terminología. Ante esta situación, se ha optado por crear una terminología propia para el proyecto: PrimageNB. Esta terminología recoge aspectos concretos sobre el neuroblastoma o conjuntos de indicadores que son usados por los profesionales sanitarios, pero que no están recogidos por ser demasiado específicos. La terminología cuenta con 153 términos y puede consultarse en el anexo A.

Hasta aquí tendríamos dos de los tres campos de la tríada que va ligada a cada variable. El último campo a rellenar sería el tipo de dato. Para ello debemos hacer un repaso de los distintos tipos de datos que ofrece DICOM SR, ya que en base a estos podremos realizar el desarrollo haciendo uso de la librería DCMTK [2]. Los tipos que recoge el estándar se presentan a continuación.

- Contenedores: CONTAINER
- Texto plano: TEXT
- Entradas codificadas: CODE
- Valores numéricos: NUM
- Nombres de persona: PNAME
- Fecha, hora o fecha y hora: DATETIME
- Identificadores únicos: UIDREF
- Referencias a objetos de tipo imagen: IMAGE
- Referencias a diagramas de ondas: WAVEFORM
- Referencias a otros objetos ensamblados: COMPOSITE

- Coordenadas espaciales: SCOORD
- Coordenadas temporales: TCOORD

Dada la naturaleza de nuestras variables nos centraremos en los tipos CONTAINER, TEXT, CODE y DATETIME.

CONTAINER

Las variables de tipo CONTAINER realmente no tienen un valor asignado. Sirven para contener a otras variables. En nuestro caso estos contenedores servirán para agrupar las variables de las distintas fases por las que pasa un paciente, así como también tendremos contenedores para agrupar resultados relativas a distintos campos como son estudios de imagen o estudios moleculares *ConceptName*. El uso de contenedores va condicionado a que debe tener nodos hijos, ya que si no, no tendría sentido su implementación. Una representación gráfica podría ser la siguiente:

```
CONTAINER ----> ConceptName
```

TEXT

Las variables de tipo TEXT contienen texto plano. Sirven para insertar un texto libre cuyo formato no está especificado. En nuestro proyecto serán usadas cuando las asignaciones de variables no se correspondan a valores de tipo CODE.

```
TEXT ----> ConceptName ----> Text Value
```

```
//Especificación de una variable de tipo TEXT
```

```
<TEXT:(209001,99PMP,"Finding")="Large, irregular mass">
```

CODE

Las variables de tipo CODE contienen un valor único codificado. Sirven para insertar valores código previamente definidos donde debe especificarse la terminología, el ID y el valor. En nuestro proyecto serán usadas cuando las asignaciones de variables se consideren cerradas y sigan un patrón repetitivo a lo largo del informe.

```
CODE ----> ConceptName ----> Code Value
```

```
//Especificación de una variable de tipo CODE
```

```
<CODE:(209104,99PMP,"Anatomic Site")=(T-04000,SNM3,"Breast")>
```

DATETIME

Las variables de tipo DATETIME contienen un valor de una hora en una fecha determinada. Sirven para insertar marcas de tiempo. En nuestro proyecto serán usadas para marcar el momento exacto de creación del informe, así como la fecha y hora exactas de las pruebas que lleven adjunta la hora de realización.


```
DATETIME ----> ConceptName ----> DT Value

//Especificación de una variable de tipo CODE
<DATETIME: (209701,99PMP, "Admitted")="200006100913">
```

Con la presentación de tipos, las terminologías y los identificadores se ha obtenido como primer resultado la tabla codificada que se adjunta en el anexo C. Se trata de una tarea costosa ya que la codificación de términos debe realizarse de forma manual y la cantidad de información que se maneja de cada paciente es muy elevada. Sin embargo aporta grandes beneficios ya que al tener toda la información codificada, el informe estructurado que se genere contendrá una gran cantidad de metainformación muy útil en trabajos futuros donde se puedan realizar búsquedas indexadas de los informes en base a los apartados y valores de las variables. Además, la codificación facilita el procesado e integración en los procesos de entrenamientos de modelos de inteligencia artificial tales como *machine learning* o *deep learning* como son los utilizados en el proyecto PRIMAGE.

CAPÍTULO 5

Desarrollo del módulo

En este capítulo se presenta la forma de desarrollar el módulo solución al problema. En primer lugar se explicará cómo generar un proyecto CMake en un entorno de trabajo donde contemos con las librerías necesarias completamente instaladas y accesibles desde el proyecto creado, el proceso de creación se ha detallado tanto para Windows como para Linux. En segundo lugar se presentará cómo realizar la cabecera del informe estructurado, hablaremos de la carga del objeto JSON y la información identificativa del documento estructurado. En tercer lugar se presenta cómo codificar e implementar el nodo raíz del documento. Finalmente se presentará el desarrollo del grueso del contenido del informe, se hará un repaso por los principales métodos de la librería DCMTK que han sido utilizados, se presentarán extractos de informes estructurados generados para comprobar la validez y la estructuración del mismo y se comentan algunos puntos críticos a la hora de realizar la implementación. Al final de este capítulo habremos conseguido cumplir con el tercer objetivo marcado relativo a la implementación del informe estructurado y el objetivo cuarto relacionado con la verificación de la creación de esos informes estructurados en formato DICOM.

5.1 Instalación de las librerías y creación del proyecto en Windows

Las características del equipo en el que se ha desarrollado el módulo son las siguientes:

- Sistema operativo: Windows 10 de 64 bits.
- Versión DCMTK: 3.6.5
- Versión Visual Studio: 2017
- Versión CMake: 3.17.0

En primer lugar es necesario obtener el código fuente de la librería que puede ser encontrado en la web de DICOM@OFFIS¹. Una vez descargado el código



¹<https://dicom.offis.de/dcmtk.php.en>

deberemos tener instalado en nuestra máquina el programa CMake². CMake está diseñado para tener un control del proceso de compilación de un proyecto. Las ventajas de CMake es que nos permite confeccionar nuestra propia configuración de compilación con unas pocas líneas de código, también nos permite elegir entre compilaciones para distintas versiones de procesadores (32 o 64 bits). Por último se ha utilizado la versión 2017 de Visual Studio como entorno de desarrollo.


Ohloh Analysis

DCMTK 3.6.5 - source code and documentation (2019-10-28)

DCMTK can be downloaded as a gzip compressed tar archive or as a ZIP archive. The contents of the two archives are identical.

	dcmk-3.6.5.tar.gz 6,332K	DCMTK 3.6.5 source code and documentation
	dcmk-3.6.5.zip 9,082K	DCMTK 3.6.5 source code and documentation

The developer documentation can also be viewed [online](#) or downloaded separately.

	dcmk-3.6.5-html-help.zip 24,212K	DCMTK 3.6.5 documentation in HTML format
---	---	--

DCMTK makes use of the following external libraries for zlib compression, TIFF and PNG creation as well as XML document processing.

Figura 5.1: Paquetes de código fuente necesarios para la instalación

Una vez tengamos el código fuente de la librería DCMTK descargado, debemos proceder a extraerlo en un directorio y ejecutar CMake que nos generará un proyecto en Visual Studio para poder instalar la librería en nuestro equipo. Al ejecutar CMake nos pedirá dos rutas, una de ellas será dónde hemos descomprimido el código y la otra dónde queremos que se compile el proyecto. Una vez ejecutado este paso, nos precargará una serie de configuraciones de las cuales únicamente deberemos marcar `BUILD_SHARED_LIBS` y `DCMTK_WITH_DOXYGEN`, lo demás debe quedar como está.

- `BUILD_SHARED_LIBS`: nos permite crear el proyecto con las librerías que hemos descargado anteriormente. Estas librerías forman parte del proyecto, así que de no marcar esta opción no sería posible el funcionamiento del mismo.
- `DCMTK_WITH_DOXYGEN`: genera la documentación de la librería

Debemos tener en cuenta y anotar para el futuro la ruta que se indica en la variable `CMAKE_INSTALL_PREFIX`, ya que será la ruta donde se van a instalar todas las funcionalidades de DCMTK de las cuales luego necesitaremos acceder para formar nuestro proyecto de aplicación.

²<https://cmake.org>

Una vez se han configurado estos dos parámetros, debemos generar desde CMake el proyecto que quedará como fichero con extensión `sln` dentro de la carpeta que le hemos indicado. Una vez tenemos este fichero (que es la solución del proyecto de instalación), debemos abrirlo con Visual Studio en modo administrador³ y compilar la solución entera (esto llevará unos minutos). Finalmente en la parte derecha de nuestra interfaz de Visual Studio veremos una librería llamada `INSTALL` que deberemos compilar y ejecutar para que en la ruta indicada (Figura 5.2) en la variable `CMAKE_INSTALL_PREFIX` se instalen las distintas funcionalidades de DCMTK.

```
Directorio de C:\Program Files (x86)\DCMTK\bin
27/03/2020 18:14 <DIR> .
27/03/2020 18:14 <DIR> ..
27/03/2020 18:07      103.936 cda2dcm.exe
27/03/2020 18:01      393.216 charls.dll
27/03/2020 18:08     1.772.544 cmr.dll
27/03/2020 18:07      122.368 dcm2json.exe
27/03/2020 18:07      100.352 dcm2pdf.exe
27/03/2020 18:05      232.448 dcm2pnm.exe
27/03/2020 18:03      149.504 dcm2xml.exe
27/03/2020 18:06      128.512 dcmcjpeg.exe
27/03/2020 18:05      114.176 dcmcjpls.exe
27/03/2020 18:03      118.272 dcmconv.exe
27/03/2020 18:07      105.472 dcmcrle.exe
27/03/2020 18:03     3.447.808 dcndata.dll
27/03/2020 18:04     1.944.064 dcndata_tests.exe
27/03/2020 18:06      106.496 dcmdjpeg.exe
```

Figura 5.2: Directorio poblado tras instalación

Hasta aquí ya habríamos cubierto los pasos que se deben realizar para instalar la librería DCMTK en el equipo donde vamos a desarrollar. Con estos pasos lo que conseguimos es que el nuevo proyecto que se creará para desarrollar el módulo pueda acceder a las librerías necesarias de DCMTK. Para instalar la librería para parsear objetos JSON debemos seguir el mismo procedimiento que con DCMTK (indicar ruta donde hemos descomprimido y ruta de compilación) y simplemente hacer clic en generar, ya que no requiere realizar ningún cambio de parámetros de configuración. Haciendo esto ya tendremos instaladas ambas librerías.

Para crear este nuevo proyecto de aplicación necesitaremos que en la ruta `C:/Users/Source/Repos/` se cree un directorio que incluya un archivo `.cxx` que contendrá el código `main` del módulo a desarrollar y un fichero de texto que debe ser nombrado como `CMakeLists.txt` que contendrá la información necesaria para crear y compilar un proyecto CMake. El formato CXX es usado para desarrollos en lenguaje C++. Así como se ha hecho uso de CXX, se podría haber usado la extensión CPP de la misma forma tal como se ha usado en la versión de preparación del entorno en Ubuntu. Debemos destacar que el directorio que debemos crear para almacenar estos dos ficheros debe encontrarse en la ruta especificada,

³Es muy importante el modo administrador porque desde este proyecto vamos a instalar las funcionalidades de DCMTK

es muy importante esta ruta ya que de estar en otra ruta distinta el proyecto no podrá tener libre acceso a la librerías instaladas de DCMTK y nlohmann.

En el fichero *CMakeLists.txt* deben plasmarse los aspectos de compilación para construir el proyecto en base a los archivos que se encuentran en el directorio. Este fichero deberá contener el nombre del proyecto, un enlace a las librerías que se requieran, configuraciones del compilador de C++ y nombre del ejecutable. A continuación se muestra el contenido del fichero *CMakeLists.txt* que se ha utilizado para la construcción del proyecto.

```
PROJECT(nbdicomsr)
cmake_minimum_required(VERSION 3.16)
SET(DCMTK_DIR "C:\\Program\ Files\ (x86)\\DCMTK")
SET(JSON_DIR "C:\\Program\ Files\ (x86)\\nlohmann_json")

find_package(nlohmann_json 3.2.0 REQUIRED)

# settings for Microsoft Visual C++ 6
SET(CMAKE_C_FLAGS "/nologo /W3 /GX /Gy /YX /MD /MDd")
SET(CMAKE_C_FLAGS_DEBUG "/MTd /Z7 /Od /MD /MDd")
SET(CMAKE_C_FLAGS_RELEASE "/MT /O2 /MD /MDd")
SET(CMAKE_CXX_FLAGS "/nologo /W3 /GX /Gy /YX /MD /MDd")
SET(CMAKE_CXX_FLAGS_DEBUG "/MTd /Z7 /Od /MD /MDd")
SET(CMAKE_CXX_FLAGS_RELEASE "/MT /O2 /MD /MDd")

ADD_DEFINITIONS(-D_REENTRANT)

INCLUDE_DIRECTORIES(${DCMTK_DIR}/include)
LINK_DIRECTORIES(${DCMTK_DIR}/lib)
LINK_DIRECTORIES(${DCMTK_DIR}/bin)

INCLUDE_DIRECTORIES(${JSON_DIR}/include)
LINK_DIRECTORIES(${JSON_DIR}/lib)

ADD_EXECUTABLE(nbdicomsr nbdicomsr)
TARGET_LINK_LIBRARIES(nbdicomsr netapi32 wsock32
ofstd dcndata dcmsr nlohmann_json::nlohmann_json)
```

Pasamos a comentar el contenido del mismo. En primer lugar debemos especificar el nombre del proyecto con la palabra PROJECT. También resulta obligatorio especificar la versión mínima requerido de CMake para general el proyecto. A continuación se deben indicar los argumentos con los que se va a configurar y generar el proyecto CMake (conjunto de SET). Por último especificaremos las librerías de DCMTK que se van a enlazar en el proyecto (TARGET_LINK_LIBRARIES).

A raíz de este punto ya se nos crea un archivo .sln que contiene la solución de Visual Studio que podremos abrir para desarrollar nuestro proyecto.

5.2 Instalación de las librerías y creación del proyecto en Ubuntu

Las características del equipo en el que se ha desarrollado el módulo son las siguientes:

- Sistema operativo: Ubuntu 18.04 de 64 bits.
- Versión DCMTK: 3.6.5
- Versión CodeLite: 10.0.0
- Versión CMake: 3.15.4

En primer lugar, tal como se ha explicado en la sección 5.1, es necesario obtener el código fuente de la librería. Una vez descargado el código deberemos tener instalado en nuestra máquina el programa CMake haciendo uso de los siguientes comandos:

```
sudo apt install build-essential libssl-dev
wget https://github.com/Kitware/CMake/releases/
      download/v3.15.4/cmake-3.15.4.tar.gz
tar -zxvf cmake-3.15.4.tar.gz
cd cmake-3.15.4
./bootstrap
make
sudo make install
```

Por último se ha utilizado la versión 10.0.0 de Codelite⁴ como entorno de desarrollo.

Una vez tengamos el código fuente de la librería DCMTK descargado, debemos proceder a extraerlo en un directorio y ejecutar los siguientes comandos en el directorio donde hemos extraído la librería:

```
mkdir dcmtk-3.6.5-build
cd dcmtk-3.6.5-build
cmake ../dcmtk-3.6.5
make -j8
sudo make install
```

Debemos tener en cuenta y anotar para el futuro que la ruta por defecto de instalación es `/usr/local`. Ruta donde se encontrarán todas las funcionalidades de DCMTK necesarias para formar nuestro proyecto.

En este momento es necesario también instalar la librería que nos permitirá parsear el json. Para ello debemos descargar el fichero desde su repositorio de GitHub⁵. Descomprimir los archivos y proceder como anteriormente en el caso de dcmtk para que se instalen las dependencias en la máquina:

⁴<https://codelite.org>

⁵<https://github.com/nlohmann/json>

```

mkdir nlohmann-build
cd nlohmann-build
cmake ../nlohmann-build
make
sudo make install

```

Finalizadas las instalaciones de ambas librerías, a continuación debemos generar en un directorio cualquiera un archivo `Main.cpp`, el fichero `CMakeLists.txt` que será un tanto distinto respecto a la creación en Windows, un directorio `src` para contener el código fuente y un script propio que crearemos para crear nuestro *workspace* en Codelite.

En el fichero `CMakeLists.txt` deben plasmarse los aspectos de compilación para construir el proyecto en base a los archivos que se encuentran en el directorio. Este fichero deberá contener el nombre del proyecto, un enlace a las librerías que se requieran y nombre del ejecutable. A continuación se muestra el contenido del fichero `CMakeLists.txt` que se ha utilizado para la construcción del proyecto.

```

PROJECT (NBStrucuredReport)
cmake_minimum_required (VERSION 3.5)

SET (CMAKE_CXX_FLAGS "${CMAKE_CXX_FLAGS} -Wall -Werror -std=c++14")

set (source_dir "${PROJECT_SOURCE_DIR}/src/")
file (GLOB source_files "${source_dir}/*.cpp")

# specify DCMtk's (default) installation directory
SET(DCMtk_DIRECTORY "/usr/local")
SET(JSON_DIRECTORY "/usr/local")

FIND_PACKAGE(DCMtk REQUIRED)

FIND_PACKAGE(nlohmann_json 3.7.0 REQUIRED)

INCLUDE_DIRECTORIES(${DCMtk_DIRECTORY}/include)
LINK_DIRECTORIES(${DCMtk_DIRECTORY}/lib)
LINK_DIRECTORIES(${DCMtk_DIRECTORY}/bin)

INCLUDE_DIRECTORIES(${JSON_DIRECTORY}/include)
LINK_DIRECTORIES(${JSON_DIRECTORY}/lib)

# declare executable and link required libraries
ADD_EXECUTABLE(NBStrucuredReport Main.cc)
TARGET_LINK_LIBRARIES(NBStrucuredReport ofstd dcldata
dcmsr dcldata nlohmann_json::nlohmann_json)

```

Los comandos empleados son prácticamente los mismos que para la creación del proyecto en Windows. Lo único que difiere es que no es necesario apuntar las opciones para el compilado de Visual Studio ya que usaremos otro entorno

de desarrollo y que es necesario apuntar al directorio `src` con el archivo `main` del proyecto. Para crear el *workspace* desarrollaremos el siguiente script al que deberemos darle permisos de ejecución una vez implementado para que se ejecute.

```
#!/bin/sh
cmake -G "CodeLite - Unix Makefiles" -DCMAKE_BUILD_TYPE=Debug
```

Con este código lo que conseguimos es lanzar el programa CMake para que nos compile el proyecto y cree nuestro *workspace* que podremos abrir desde CodeLite para programar nuestra aplicación.

5.3 Cabecera del informe estructurado

A partir de esta sección vamos a pasar a hablar de la implementación pura del informe estructurado. El código del módulo desarrollado, así como los archivos de configuración de compilación del proyecto y el informe estructurado generado se encuentran en un repositorio propio público en GitHub⁶. Como se ha indicado durante la construcción del proyecto, el árbol a generar se realizará en un archivo *Main* que contiene un método principal que al ser ejecutado irá rellenando el valor de las variables que hemos identificado a partir de los valores del objeto JSON que contiene la información del formulario web de PRIMAGE.

En primer lugar, antes de crear el documento DICOM SR deberemos cargar el objeto JSON haciendo uso de la librería que hemos instalado para este propósito. El modo de carga consiste en hacer uso del operador *ifstream* para cargar el JSON combinado con el operador *nlohmann::json* que nos permitirá acceder a los pares clave-valor del objeto precargado:

```
using json = nlohmann::json;
std::ifstream i("example_3.json");
json j;
i >> j;
```

Una vez cargado el objeto, debemos crear un documento que será nuestro informe estructurado y adjuntarle los campos de cabecera que hayamos estimado. Para ello DICOM ofrece una serie de módulos a implementar en la propia cabecera (figura 5.3). En nuestro caso utilizaremos los módulos de la clase paciente, el módulo general de la clase estudio y el módulo general de la clase documento. Los atributos que se pueden emplear en cada uno de estos módulos son numerosos (pueden consultarse en DICOM NEMA⁷ accediendo a las referencias de cada uno). Ante tal cantidad se ha decidido hacer una selección en base a la información recogida del paciente indicada con la palabra *header* en el anexo C y los atributos predeterminados que la propia librería DCMTK inserta sobre la identificación del documento e información sobre el informe estructurado.

⁶<https://github.com/dachirui/NBStructuredReport>

⁷http://dicom.nema.org/dicom/2013/output/chtml/part20/sect_A.3.html

IE	Module	Reference	Usage	Covered by this Transformation Guideline
Patient	Patient	C.7.1.1	M	Yes
	Clinical Trial Subject	C.7.1.3	U	No
Study	General Study	C.7.2.1	M	Yes
	Patient Study	C.7.2.2	U	No
	Clinical Trial Study	C.7.2.3	U	No
Series	SR Document Series	C.17.1	M	Yes
	Clinical Trial Series	C.7.3.2	U	No
Equipment	General Equipment	C.7.5.1	M	Yes
Document	SR Document General	C.17.2	M	Yes
	SR Document Content	C.17.3	M	Yes
	SOP Common	C.12.1	M	Yes

Figura 5.3: Conjunto de módulos de cabecera

A continuación se presentan tres tablas relativas a los tres módulos donde se indican los atributos empleados, la etiqueta identificadora del campo que a posteriori veremos cuando transformemos el objeto DICOM-SR a texto y una breve descripción del campo.

Nombre atributo	Etiqueta	Descripción
Patient's Name	(0010,0010)	Nombre del paciente
Patient ID	(0010,0020)	Identificador del paciente.
Patient's Sex	(0010,0040)	Sexo: M(ale), F(emale) u O(ther).

Tabla 5.1: Atributos del módulo paciente

Respecto a la tabla 5.1 se debe destacar que a pesar de no haber ningún campo relativo al nombre del paciente en el objeto JSON, por razones de protección de datos ya que es información que está saliendo de un centro hospitalario, se ha decidido incluir con un guión bajo para que pueda ser utilizado internamente en los centros sanitarios y puedan identificar a pacientes por nombre. Estos atributos se han insertado de la siguiente forma haciendo uso de la librería DCMTK:

```
std::string string_sex = j["patient_data"]["sex"]["value"];
std::string string_patient_id = j["patient_data"]["patient_id"]["value"];

if (string_sex == "Male"){
string_sex = "M";
}
else if(string_sex == "Female"){
string_sex = "F";
}
else {
string_sex = "0";
```

```

}

const char *patient_id = string_patient_id.c_str();
const char *sex = string_sex.c_str();

DSRDocument *doc = new DSRDocument();

doc->createNewDocument(DSRTypes::DT_BasicTextSR);
doc->setPatientName("");
doc->setPatientSex(sex);
doc->setPatientID(patient_id);

```

El fragmento anterior de código presenta la forma en la que se accede a la información del objeto JSON (`j[clave_objetivo]`), así como las instrucciones para crear un documento DICOM-SR y adjuntar valor a los atributos de cabecera. El uso de la librería `nlohmann` nos permite acceder de una forma muy intuitiva a los valores de cada objeto. A pesar de ser valores de tipo *string*, deben convertirse a *char* para poder asignarlos a cada uno de los atributos usados en la cabecera. Más adelante presentaremos la singularidad que presenta `nlohmann` a la hora de recoger variables de tipo entero, tipo fecha o booleanos y cómo realizar un correcto tratamiento del valor para que pueda adjuntarse a el informe estructurado.

Nombre atributo	Etiqueta	Descripción
Study Instance UID	(0020,000D)	Identificador único del estudio.
Study Date	(0008,0020)	Fecha de creación.

Tabla 5.2: Atributos del módulo estudio

Los valores asignados tanto al identificador único, como a la fecha del estudio son asignados de forma automática cuando se genera el informe estructurado.

Nombre atributo	Etiqueta	Descripción
Completion Flag	(0040,A491)	Grado de completitud del documento
Verification Flag	(0040,A493)	Indica si el documento está verificado.

Tabla 5.3: Atributos del módulo documento

Los valores de los atributos de la tabla 5.3 también son generados de forma automática cuando se construye el informe estructurado. Estos valores se asignan de acuerdo a las características del documento. El grado de completitud indica si el contenido del documento contiene todas las observaciones significativas dentro del tema del nodo raíz, el grado de completitud puede ser parcial (*PARTIAL*) o completo (*COMPLETE*). Por último, los valores del indicador de verificación pueden variar entre verificado (*VERIFIED*) o no verificado (*UNVERIFIED*). Para que un documento pueda ser verificado debe haber una persona responsable cuyo nombre aparezca en el atributo *Observer Name* (0040,A075), indicando así la persona responsable de la verificación del documento.

Como resultado se obtiene una cabecera común personalizada para todos los estudios de neuroblastoma. Esta cabecera aparece en los visores DICOM que se

usan para visualizar este tipo de objetos. Podemos extraer también los datos de la cabecera empleado el método *dcmdump* de la librería DCMTK para volcar el objeto DICOM en un fichero de texto. En la figura 5.4 observamos como en la esquina superior derecha aparecen la información de la cabecera relativa al nombre, al id del paciente y al género del mismo. En la esquina inferior derecha aparece la fecha de generación del informe estructurado. La visualización del informe se ha realizado con el visor DICOM RadiAnt⁸.

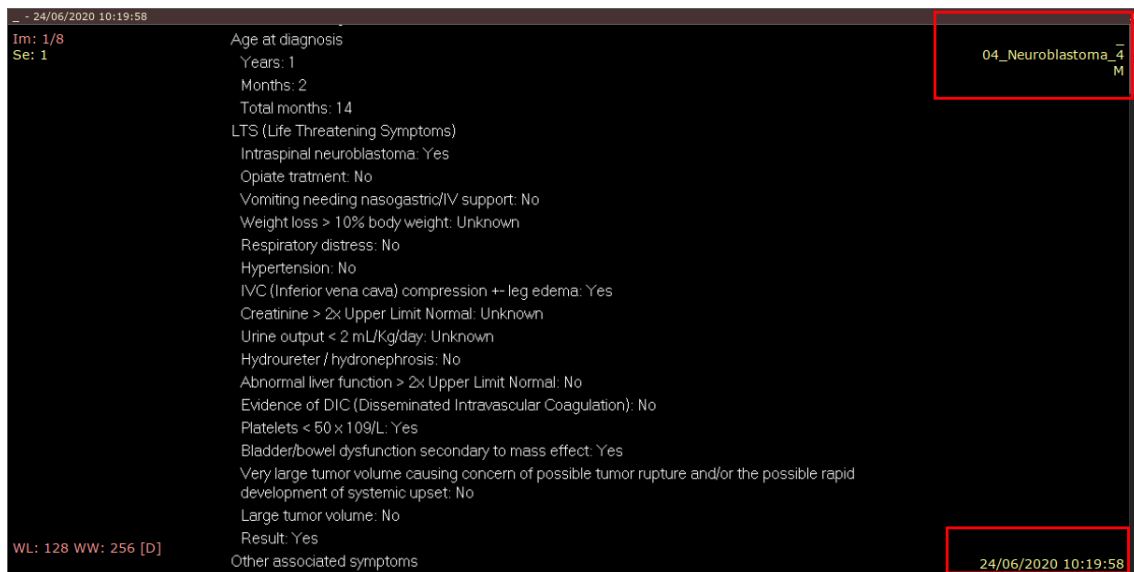


Figura 5.4: Informe estructurado de NB en el visor DICOM

Como hemos comentado, haciendo uso de la instrucción *dcmdump* sobre el objeto DICOM, podemos obtener en formato texto el contenido del mismo. En la figura 5.5 se muestra la misma información que en el visor pero esta vez bajo la estructura en texto del documento. Queda resaltado en rojo el nombre del paciente, el id, el género y la fecha. Hasta aquí queda explicado cómo se ha desarrollado la cabecera del informe estructurado que genera el módulo, cuáles han sido los atributos que se han seleccionado para ella y cómo podemos obtener la información de la cabecera usando un visor DICOM o haciendo un volcado del objeto a un fichero de texto.

⁸<https://www.radiantviewer.com>

```

# Dicom-Data-Set
# Used TransferSyntax: Little Endian Explicit
(0008,0012) DA [20200624] # 8, 1 InstanceCreationDate
(0008,0013) TM [101958] # 6, 1 InstanceCreationTime
(0008,0014) UI [1.2.276.0.7230010.3.0.3.6.5] # 28, 1 InstanceCreatorUID
(0008,0016) UI =BasicTextSRStorage # 30, 1 SOPClassUID
(0008,0018) UI [1.2.276.0.7230010.3.1.4.8323329.2319.1592986798.719033] # 54, 1 SOPInstanceUID
(0008,0020) DA (no value available) # 0, 0 StudyDate
(0008,0023) DA [20200624] # 8, 1 ContentDate
(0008,0030) TM (no value available) # 0, 0 StudyTime
(0008,0033) TM [101958] # 6, 1 ContentTime
(0008,0050) SH (no value available) # 0, 0 AccessionNumber
(0008,0060) CS [SR] # 2, 1 Modality
(0008,0070) LO (no value available) # 0, 0 Manufacturer
(0008,0090) PN (no value available) # 0, 0 ReferringPhysicianName
(0008,1111) SQ (Sequence with undefined length #=0) # u/l, 1 ReferencedPerformedProcedureStepSequence
(fffe,e0dd) na (SequenceDelimitationItem) # 0, 0 SequenceDelimitationItem
(0010,0010) PN [_] # 2, 1 PatientName
(0010,0020) LO [04_Neuroblastoma_4] # 18, 1 PatientID
(0010,0030) DA (no value available) # 0, 0 PatientBirthDate
(0010,0040) CS [M] # 2, 1 PatientSex
(0020,000d) UI [1.2.276.0.7230010.3.1.2.8323329.2319.1592986798.719034] # 54, 1 StudyInstanceUID
(0020,000e) UI [1.2.276.0.7230010.3.1.3.8323329.2319.1592986798.719035] # 54, 1 SeriesInstanceUID
(0020,0010) SH (no value available) # 0, 0 StudyID
(0020,0011) IS [1] # 2, 1 SeriesNumber
(0020,0013) IS [1] # 2, 1 InstanceNumber
(0040,a040) CS [CONTAINER] # 10, 1 ValueType
(0040,a043) SQ (Sequence with undefined length #=1) # u/l, 1 ConceptNameCodeSequence
(fffe,e000) na (Item with undefined length #=3) # u/l, 1 Item
(0008,0100) SH [1] # 2, 1 CodeValue

```

Figura 5.5: Fragmento de cabecera del objeto DICOM-SR

5.4 Nodo raíz y título del informe

Como se comentó en el punto 2.1.1, el informe estructurado que sigue el estándar DICOM se compone de tres capas (cabecera, nodo raíz y contenido). Tras la explicación de cómo se ha implementado la primera de ellas, en esta sección nos introducimos en el desarrollo de la capa intermedia, el nodo raíz que incluye el título del informe. Esta es la capa más simple, es la que menos información contiene pero es la base de la tercera capa, el contenido del árbol. De este nodo raíz, nacerán todos los nodos hijos del árbol necesarios para estructurar la información. En DCMTK, la generación del nodo raíz se realiza de la siguiente forma:

```

doc->getTree().addContentItem(DSRTypes::RT_isRoot,
DSRTypes::VT_Container);
doc->getTree().getCurrentContentItem().setConceptName
(DSRCodedEntryValue("1", "PrimageNB", "Neuroblastoma structured report"));

```

Como se puede observar, al árbol del documento previamente creado (doc) se le van a empezar a añadir nodos denominados *ContentItem* haciendo uso del método *addContentItem* que nos provee la librería DCMTK. Debido a que el árbol está vacío, el primer nodo debe ser de tipo raíz (*RT_isRoot*) y de tipo CONTAINER ya que este será el contenedor principal del documento. Esta información es pasada como argumento al método para añadir los nodos. Todos los elementos irán encapsulados bajo este nodo raíz. Una vez hemos añadido el nodo podemos obtenerlo haciendo uso del método *getCurrentContentItem*. Este método nos permite rescatar el ítem en el que nos encontramos actualmente y podemos editar atributos como el *ConceptName*. De esta forma podemos editar el nodo para anejarle una entrada codificada cuyos parámetros se indican como argumentos en el método *DSRCodedEntryValue*, donde le pasamos la codificación del nodo: código

del término, terminología empleada y valor. En este caso, el valor de esta variable corresponde al título del documento al ser un nodo raíz. En la figura 5.6, se muestra en el contenido el texto del objeto DICOM unos rectángulos rojos enmarcando el tipo del nodo, así como los atributos y valores del mismo.

```
(0020,0011) IS [1] # 2, 1 SeriesNumber
(0020,0013) IS [1] # 2, 1 InstanceNumber
(0040,a040) CS [CONTAINER] # 10, 1 ValueType
(0040,a043) SQ (Sequence with undefined length #-1) # u/l, 1 ConceptNameCodeSequence
(fffe,e000) na (Item with undefined length #-3) # u/l, 1 Item
(0008,0100) SH [1] # 2, 1 CodeValue
(0008,0102) SH [PrimageNB] # 10, 1 CodingSchemeDesignator
(0008,0104) LO [Neuroblastoma structured report] # 32, 1 CodeMeaning
(fffe,e00d) na (ItemDelimitationItem) # 0, 0 ItemDelimitationItem
(fffe,e0dd) na (SequenceDelimitationItem) # 0, 0 SequenceDelimitationItem
```

Figura 5.6: Detalle del nodo raíz y sus atributos en el objeto DICOM

Figura 5.7: Detalle del nodo raíz en HTML

El visor de imágenes RadiAnt permite visualizar el informe estructurado en formato HTML. Se ha hecho uso de esta utilidad en la figura 5.7 para mostrar la estructura de árbol que se genera tras el desarrollo del nodo raíz. Vemos como el título corresponde a este nodo del cual cuelgan otras variables que correspondrían a la capa última, la capa del contenido. Hasta aquí, queda explicado cómo se ha desarrollado el nodo raíz y el título del informe estructurado que genera el módulo, y cuáles han sido los atributos que se han seleccionado para realizar la codificación de este nodo base y la apariencia que este toma tanto en texto como en formato HTML.

5.5 Contenido y verificaciones del DICOM-SR

Una vez explicada la implementación tanto de la cabecera del informe, como del nodo raíz del árbol generador, es momento de explicar cómo se lleva a cabo la implementación de los distintos nodos que componen el árbol. En primer lugar, uno de los factores más importantes es el desplazamiento a lo largo de esta estructura de datos, es imprescindible conocer los métodos que nos permiten desplazarnos entre nodos, métodos que nos permitan indicar el nivel de colocación del nodo y el retroceso a nodos padre para construir nuevas ramas.

El modo de colocar un nodo en el árbol es sencillo, la complejidad radica en la gran cantidad de variables que se manejan y las condiciones que deben cumplirse para que entre distintos pacientes se generen unos campos u otros del informe. Por ejemplo, en el caso de que dos pacientes tengan distintos estadios, tal como se indica en el anexo C, deben hacerse unas pruebas u otras, viéndose esto reflejado en el resultado final del árbol. A continuación se muestra un extracto del código fuente utilizado donde se añade uno de estos nodos. Vamos a repasar su constructor y los parámetros que se usan, junto con las distintas opciones que DCMTK nos ofrece. Además aprovecharemos para repasar la depuración necesaria para conseguir unos datos acordes a la codificación de fecha usada por DCMTK.

```
doc->getTree().addContentItem(DSRTypes::RT_hasObsContext, DSRTypes::VT_DateTime, DSRTypes::AM_belowCurrent);
doc->getTree().getCurrentContentItem().setConceptName(DSRCodedEntryValue("432213005", "SRT", "Diagnosis date"));

bool checker = j["diagnosis"]["associated_symptoms"]["diagnosis_date"]["value"].is_null();
if (checker == false) {

    std::string string_diagnosis_date = j["diagnosis"]["associated_symptoms"]["diagnosis_date"]["value"];
    std::string chars = "-T.Z";
    for (char c : chars) {
        string_diagnosis_date.erase(std::remove(string_diagnosis_date.begin(),
            string_diagnosis_date.end(), c), string_diagnosis_date.end());
    }
    std::string sub_string_diagnosis_date = string_diagnosis_date.substr(0, 12);
    const char *diagnosis_date = sub_string_diagnosis_date.c_str();
    doc->getTree().getCurrentContentItem().setObservationDateTime(diagnosis_date);
    doc->getTree().getCurrentContentItem().setStringValue(diagnosis_date);
}
```

Figura 5.8: Inserción de un ítem

Pasamos a comentar el contenido de la figura 5.8. En primer lugar es necesario indicar la inserción del nodo haciendo uso del método *addContentItem* del mismo modo que con el nodo raíz. Si seguimos inspeccionando la línea, encontramos ya una diferencia respecto a los nodos de tipo CONTAINER, y es el uso de la palabra reservada *RT_hasObsContext* para indicar el tipo de relación que tiene ese nodo en el árbol. Así pasamos de indicar que este nodo es un contenedor que contiene otros nodos a indicar que este nodo tiene un *Observation Context*, es decir, que la información del nodo describe quién o qué está realizando la interpretación, en este caso se trata de la fecha de diagnóstico, que es lo que se está interpretando. Este *Observation Context* será común a todos los nodos que no sean de tipo CONTAINER. Los tipos de relaciones entre nodos que existen en DCMTK son los que se muestran a continuación. Además pueden consultarse el significado y el modo de uso de los atributos en la documentación de la librería⁹.

```
enum E_RelationshipType {
    RT_invalid, RT_unknown, RT_isRoot, RT_contains,
    RT_hasObsContext, RT_hasAcqContext, RT_hasConceptMod, RT_hasProperties,
    RT_inferredFrom, RT_selectedFrom, RT_last = RT_selectedFrom
}
```

Avanzando en la construcción del nodo tenemos el tipo. Ya hemos hablado con anterioridad de los tipos existentes en DICOM, así que la implementación en DCMTK es directa y son los siguientes.

⁹<https://support.dcmk.org/docs/classDSRTypes.html>

```
enum E_ValueType {
    VT_invalid, VT_Text, VT_Code, VT_Num,
    VT_DateTime, VT_Date, VT_Time, VT_UIDRef,
    VT_PName, VT_SCoord, VT_SCoord3D, VT_TCoord,
    VT_Composite, VT_Image, VT_Waveform, VT_Container,
    VT_byReference, VT_includedTemplate, VT_last = VT_includedTemplate
}
```

Por último, pero no menos importante debe indicarse el modo del nodo. Este atributo nos va a marcar en qué posición se generará el nodo, ya que no es lo mismo que un nodo sea hijo de su predecesor y esté a un nivel de profundidad mayor, que deba estar a la misma profundidad siendo así hermano. Los modos existentes se muestran en el siguiente fragmento.

```
enum E_AddMode { AM_afterCurrent, AM_beforeCurrent,
    AM_belowCurrent, AM_belowCurrentBeforeFirstChild }
}
```

La segunda línea de de la figura 5.8 consiste en la tríada necesaria para la codificación de la cual ya se ha hablado con anterioridad en la sección 5.4. A continuación se procede a comprobar si ese campo es nulo (*isNull()*) por si se diera el caso en el que no hubiera fecha disponible y el objeto JSON tuviese el valor nulo. Si no hiciésemos esta comprobación cuando accediéramos a esa variable nuestro módulo fallaría por intentar insertar en el informe estructurado un valor nulo. Una vez nos hemos asegurado de que el valor existe lo recogemos en una variable de tipo cadena como se establece en la línea cinco. Si estuviéramos ante una variable que hubiésemos especificado como tipo TEXT, directamente saltaríamos a la línea 11 donde se realiza la conversión a *char* y asignaríamos el valor al nodo como aparece en la última línea. Sin embargo nos encontramos ante una variable de tipo fecha que se presenta en el objeto JSON de esta forma:

```
"diagnosis_date": {
    "value": "2020-05-10T22:00:00.000Z",
    "type": "date"
}
```

Como se observa, el formato con el que se guarda el valor de la variable no es el adecuado ya que según las especificaciones DCMTK emplea la notación YYYYMMDDHHMM¹⁰ como una serie de 12 caracteres consecutivos. Por ello hacemos un filtrado de los símbolos (-T:.Z) para eliminarlos de la variable y quedarnos con los 12 primeros caracteres. Una vez realizado esto se procede a insertar el valor de la variable en el nodo. En la figura 5.9, se muestra como queda insertado este primer nodo y la arquitectura de una pequeña parte del informe estructurado. En verde se han indicado las hojas del árbol y en otros colores los distintos contenedores, hijos intermedios que no son nodos finales, pero que constituyen nodos creadores de distintas ramas.

¹⁰(Y = año, M = mes, D = día, H = hora, M = minutos)

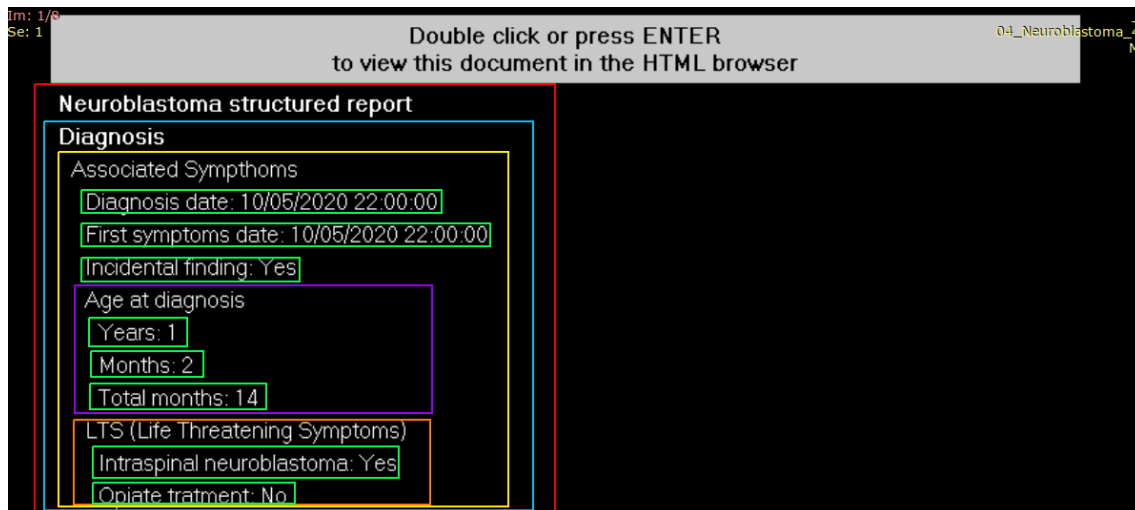


Figura 5.9: Extracto del informe estructurado con los componentes encuadrados

Hasta aquí se ha explicado el funcionamiento de lo que es la creación del informe estructurado, la mecánica consiste en realizar una lectura del objeto JSON para ir extrayendo valores de variables e ir asignándolos al esquema del informe estructurado que hemos extraído a partir de las tablas del anexo C. A partir de aquí, se va a entrar en detalle en algunas de las peculiaridades de la implementación. Ya hemos visto como trabajar con las variables de tipo fecha. Entre las restantes veremos el método necesario para saltar al nodo padre, la extracción de datos de tipo numérico, de tipo booleano y la recuperación de pruebas que constituyan un *array* de variables de tamaño indeterminado.

Ya hemos visto que la adición de nodos resulta directa. Indicando el modo del nodo podremos colocarlo por debajo del nodo actual, tras el nodo actual, antes del actual o como primer hijo del nodo actual. Una vez la rama del árbol que constituye el contenedor se haya completado es necesario subir tantos niveles como niveles de profundidad contenga el contenedor. Para ello DCMTK nos ofrece distintos métodos para desplazarnos de manera vertical en dirección a la raíz. El método *gotoRoot()* nos permite establecer el puntero interno al nodo raíz, añadiendo así el próximo nodo inmediatamente por debajo del nodo raíz. También tenemos el método *gotoNode(const size_t searchID, const OFBool startFromRoot=OFTrue)* que nos permite indicar al nodo que queremos saltar directamente haciendo una búsqueda en profundidad desde el nodo raíz al nodo objetivo. Por último también, tenemos el método *goUp()* que nos permite saltar al nodo padre inmediatamente superior al nodo al que actualmente está apuntando el cursor.

Para la extracción de valores de tipo booleano o entero del objeto JSON con el que trabajamos, es necesario hacer uso del método *get<tipo>()* sobre el objeto JSON. En la figura 5.11 se muestra cómo se realiza el tratamiento para los objetos de tipo booleano. El origen de este tipo de datos proviene del uso de *check boxes* donde únicamente es posible elegir una opción como se muestra en la figura 5.10.

Patient data	Diagnosis	Treatment and evaluation	Follow-up
Associated symptoms	Laboratory	Imaging	Staging
Diagnosis date	16/06/2020	Age at diagnosis (months)	1
Incidental finding	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No <input type="checkbox"/> Unknown	First symptoms date	09/06/2020

Figura 5.10: Extracto del informe estructurado con los componentes encuadrados

Esta implementación genera el siguiente resultado en el JSON:

```
incidental_finding": {
  "type": "boolean",
  "yes": true,
  "no": false,
  "unknown": false
}
```

Para lidiar con este objeto deberemos extraer el valor de cada una de las claves de *incidental_finding* y dependiendo de cuál de las tres claves contenga el valor *true* haremos la conversión a tipo *char* para poder anexarlo al nodo tal como se muestra en la figura 5.11.

```
doc->getTree().addContentItem(DSRTypes::RT_hasObsContext, DSRTypes::VT_Code, DSRTypes::AM_afterCurrent);
doc->getTree().getCurrentContentItem().setConceptName(DSRCodedEntryValue("C48619", "NCIt", "Incidental finding"));
bool bool_yes = j["diagnosis"]["associated_symptoms"]["incidental_finding"]["yes"].get<bool>();
bool bool_no = j["diagnosis"]["associated_symptoms"]["incidental_finding"]["no"].get<bool>();
bool bool_unknown = j["diagnosis"]["associated_symptoms"]["incidental_finding"]["unknown"].get<bool>();
if (bool_yes == true) {
  doc->getTree().getCurrentContentItem().setCodeValue(DSRCodedEntryValue("373066001", "SRT", "Yes"));
}
else if (bool_no == true) {
  doc->getTree().getCurrentContentItem().setCodeValue(DSRCodedEntryValue("373067005", "SRT", "No"));
}
else if (bool_unknown == true) {
  doc->getTree().getCurrentContentItem().setCodeValue(DSRCodedEntryValue("261665006", "SRT", "Unknown"));
}
```

Figura 5.11: Inserción de un nodo de tipo CODE a raíz de un booleano

Una última particularidad con la que nos hemos encontrado durante la implementación del informe estructurado es la recuperación de los valores relativos a variables que provienen de pruebas que constituyen un número indeterminado de resultados. Un ejemplo lo encontramos en la sección *Diagnosis/Laboratory/Molecular studies/NCA/Number of alterations/Alteration i*. En este caso si la prueba NCA se evalúa a *True* como resultado, podemos tener cualquier valor numérico para el número de alteraciones y deberemos extraer la información necesaria de cada una de ellas independientemente del número. En la figura 5.12 vemos cómo se ha implementado, donde dependiendo de ese número de alteraciones, deberemos rellenar la información de cada una de ellas. Esta implementación da como resultado el siguiente fragmento del objeto JSON.

```
"number": {
```

```

    "type": "number",
    "value": 2
  },
  "alterations": [
    {
      "chromosome": {
        "type": "string",
        "value": "2"
      },
      "grain_loss": {
        "type": "string",
        "value": "Gain (+)"
      }
    },
    {
      "chromosome": {
        "value": "3",
        "type": "string"
      },
      "grain_loss": {
        "value": "Loss (-)",
        "type": "string"
      }
    }
  ]

```

Figura 5.12: Número de alteraciones en la plataforma PRIMAGE

Para resolver esta situación desde la implementación del código, lo que haremos será guardar el valor del número de alteraciones que existen. Tras esto crearemos un bucle donde se crearán tantos nodos de tipo CONTAINER como número de alteraciones cromosómicas. De estos contenedores colgarán los dos nodos hoja resultado que indicarán el número del cromosoma donde se ha producido la alteración y el resultado de la misma. En la figura 5.13 vemos como se ha desarrollado la solución a esta particularidad y en la figura 5.14 el resultado final donde se marcan los nodos hoja en color verde y los distintos contenedores que se han creado en color azul y rojo.

```

doc->getTree().addContentItem(DSRTypes::RT_hasObsContext, DSRTypes::VT_Text, DSRTypes::AM_afterCurrent);
doc->getTree().getCurrentContentItem().setConceptName(DSRCodedEntryValue("27", "PrimageNB", "NCA Number of alterations"));
int num_alterations = j["diagnosis"]["laboratory"]["molecular_studies"]["nca"]["number"]["value"].get<int>();
std::string string_alterations = to_string(num_alterations);
const char *alterations = string_alterations.c_str();
doc->getTree().getCurrentContentItem().setStringValue(alterations);

for (int i = 0; i < num_alterations; i++) {
doc->getTree().addContentItem(DSRTypes::RT_contains, DSRTypes::VT_Container, DSRTypes::AM_afterCurrent);
doc->getTree().getCurrentContentItem().setConceptName(DSRCodedEntryValue("C28344", "NCIT", "NCA Alteration"));
doc->getTree().addContentItem(DSRTypes::RT_hasObsContext, DSRTypes::VT_Text, DSRTypes::AM_belowCurrent);
doc->getTree().getCurrentContentItem().setConceptName(DSRCodedEntryValue("91272006", "SRT", "Chromosome"));

std::string str_chr = j["diagnosis"]["laboratory"]["molecular_studies"]["nca"]["alterations"][i]["chromosome"]["value"];
const char *chr = str_chr.c_str();
doc->getTree().getCurrentContentItem().setStringValue(chr);

doc->getTree().addContentItem(DSRTypes::RT_hasObsContext, DSRTypes::VT_Text, DSRTypes::AM_afterCurrent);
doc->getTree().getCurrentContentItem().setConceptName(DSRCodedEntryValue("28", "PrimageNB", "Gain/loss"));

std::string str_gain_loss = j["diagnosis"]["laboratory"]["molecular_studies"]["nca"]["alterations"][i]["gain_loss"]["value"];
const char *gain_loss = str_gain_loss.c_str();
doc->getTree().getCurrentContentItem().setStringValue(gain_loss);
}

```

Figura 5.13: Implementación número de alteraciones

Im: 3/8
 Se: 1
 Status ALK mutation R1275Q: No
 Ploidy: Evaluated
 Method Karyotyping: Evaluated
 Ploidy Status: Pentaploid
 NCA: Evaluated
 Method FISH: Evaluated
 NCA Status: Yes
 NCA Number of alterations: 2
 NCA Alteration
 Chromosome: 2
 Gain/loss: Gain (+)
 NCA Alteration
 Chromosome: 3
 Gain/loss: Loss (-)
 SCA: Evaluated
 Method aCGH: Evaluated
 SCA Status: Yes
 SCA Typical 1p: Yes
 SCA Typical +1q: Yes
 SCA Typical +2p: Yes
 X: 82 Y: 114 Val: 0
 WL: 158 WW: 274
 04_Neuroblastoma_4
 M
 24/06/2020 10:19:58

Figura 5.14: Resultado de la implementación del número de alteraciones

Para finalizar el capítulo, a modo de resumen hemos visto como se ha implementado el módulo generador de informes estructurados siguiendo el estándar DICOM haciendo uso de la librería DCMTK. Hemos seguido el desarrollo desde la cabecera del objeto, pasando por el nodo raíz y el título del documento hasta llegar a comentar los aspectos más relevantes y que han causado alguna dificultad a la hora de desarrollar el código. Hemos mostrado algunos fragmentos de código para clarificar estos puntos críticos y también se han mostrado fragmentos del informe estructurado resultante.

CAPÍTULO 6

Despliegue del módulo

En este capítulo se presentan los métodos existentes para llevar a cabo el despliegue del módulo desarrollado. En primer lugar se hablará de las herramientas existentes para confeccionar el despliegue de aplicaciones, se analizarán algunas de las tecnologías más recientes y utilizadas en la actualidad y se propondrán argumentos a favor y en contra para realizar un análisis comparativo entre las distintas opciones que tenemos. En segundo lugar se expondrá el modo de utilización de Docker y analizaremos los comandos y la configuración necesaria para la creación de imágenes y contenedores que se puedan integrar en la plataforma PRIMAGE. Al final de este capítulo se habrá cumplido el objetivo último relativo al despliegue del módulo de creación de informes para su integración en la plataforma PRIMAGE.

6.1 Herramientas de despliegue

El proceso de despliegue se conforma tras el desarrollo y las pruebas que se han ido comentando a lo largo de la etapa de desarrollo con la que hemos conseguido generar un informe estructurado en base a un objeto JSON que hemos obtenido de la plataforma web del proyecto PRIMAGE, ya que hasta el momento las ejecuciones del módulo se han llevado a cabo en la máquina local de desarrollo y es momento de pasar al despliegue en el entorno real. Debido a que el módulo tiene una funcionalidad muy específica, no es necesario hacer ninguna guía de planificación donde se explique el ritmo al que va a integrarse el módulo en el sistema y cómo afectará al resto de los usuarios de la plataforma. La correcta integración del módulo supondrá la adición de un botón que al hacer clic ejecute nuestro programa, generando y descargando el objeto DICOM del paciente seleccionado. En esta sección vamos a repasar las principales herramientas que nos permiten automatizar el despliegue de nuestras aplicaciones para tener una visión general de lo que podemos y no podemos hacer en esta etapa.

En los últimos años la gran variedad de *software*, *hardware* y el auge de las aplicaciones web ha hecho complicada la integración de un entorno único para desarrollo que luego pudiera trasladarse de forma efectiva al entorno real. En muchas ocasiones la máquina de desarrollo no contiene el mismo sistema o versión que el servidor donde irá alojada la aplicación, de aquí nace el uso de contenedores. Los contenedores se basan en la virtualización de entornos que nos permiten ejecutar

nuestros desarrollos independientemente del sistema operativo en el que se vaya a implantar la solución.

Docker es una de las tecnologías más conocidas para este propósito. Docker permite la creación de contenedores en cualquier máquina y transportarlos a otro entorno para que funcionen independientemente del equipo físico donde operen. Podríamos pensar que para esto ya existen las máquinas virtuales (MVs), dado que uno puede montar una MV en su propio equipo para simular un sistema y trabajar independientemente del entorno, pero aquí asoma una ventaja de los contenedores Docker, y es la ligereza. Mientras en una MV debe empaquetarse el sistema operativo anfitrión, el hipervisor y el sistema operativo de la máquina virtual, en Docker la creación de un contenedor simplemente supone la adición de un proceso más a la lista de los ejecutados por el equipo. Los contenedores Docker se autogestionan en términos de CPU, elementos de entrada y salida o elementos de comunicaciones en red. Por esto Docker nos ofrece un ahorro de recursos y una facilidad en la gestión a la hora de desplegar aplicaciones.

Otro aspecto clave que favorece a Docker es la implementación por capas que utiliza. Cada vez que se ejecuta una acción se guarda un registro por lo que la restauración a versiones anteriores de la imagen del contenedor se puede realizar de forma rápida y eficiente.

El aspecto negativo de Docker es que cuando el grado de anidamiento de contenedores es elevado, la gestión se vuelve muy costosa ya que cada contenedor es descompuesto en múltiples capas y finalmente es mejor optar por un gran contenedor que aúne todas las aplicaciones. De esta desventaja nace la herramienta para Docker Kubernetes de la cual hablaremos más adelante. Una última desventaja desde el punto de vista de la seguridad es que el *daemon* de ejecución requiere permisos de administrador para acceder a la raíz del sistema. Un acceso no requerido en una máquina local puede tener menos impacto que un proceso de Docker que se ejecute en un servidor web.

Como siempre, es interesante conocer y analizar otras alternativas ya que no todas ofrecen los mismos servicios y cada una puede adaptarse de una mejor o peor forma al proyecto.

RKT consiste también en una herramienta para la gestión de contenedores. La principal diferencia con Docker es que al ejecutarse exclusivamente sobre plataformas linux permite gestionar los permisos de grupo para que la ejecución sea llevada a cabo por usuarios con unos privilegios, pero sin llegar a ser super usuarios. Es un aspecto que mejora la seguridad en la ejecución de contenedores. Sin embargo, la integración de RKT resulta menos intuitiva ya que debe realizarse una instalación y una configuración con más pasos manuales. Lo mismo ocurre con PODMAN y SINGULARITY. Estas herramientas comparten la ejecución de contenedores sin necesidad de accesos *root*, pero suponen una mayor gestión en la configuración y despliegue de contenedores, por lo que resulta menos ágil.

Recordando el punto débil de Docker en cuanto a la gestión de múltiples contenedores, Kubernetes nos ofrece la posibilidad de gestionar los contenedores que tenemos. Kubernetes actúa como un director donde en base a unos requerimientos de escalabilidad, balanceo de carga, almacenamiento, etc. especificados al nodo *master*, el cual se encarga de que cada uno de los nodos hijo, que contienen contenedores Docker, reciba los recursos necesarios para el buen funcionamiento.

to. En la figura 6.1 se muestra la arquitectura de esta herramienta que permite la gestión de contenedores.

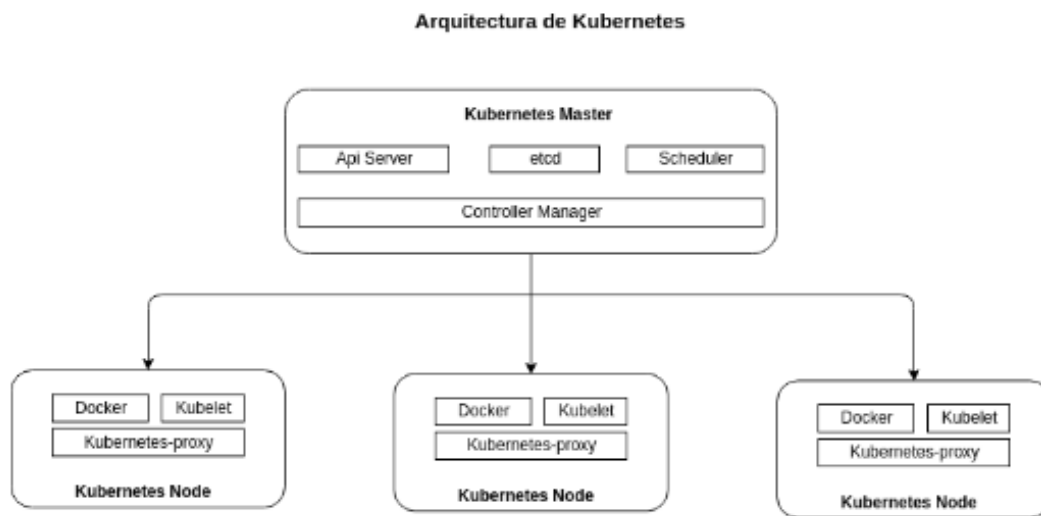


Figura 6.1: Arquitectura de Kubernetes

Tras el pequeño análisis realizado de las distintas herramientas para la gestión de contenedores, en este TFM se ha decidido usar Docker, ya que actualmente las aplicaciones de QUIBIM están alojadas en contenedores Docker debido a la facilidad que ofrecen en la instalación, configuración y creación de los mismos, por lo que por coherencia se decide usar esta tecnología.

6.2 Uso de Docker

Con el uso de Docker se pretende crear una imagen de un contenedor idéntica al sistema donde se ha desarrollado la aplicación lista para la ejecución. La idea es automatizar todos los pasos que se han seguido en la sección de creación del proyecto en Ubuntu para que se genere una imagen con sistema operativo Ubuntu que contenga las herramientas y las librerías necesarias para compilar el proyecto y generar el ejecutable del módulo desarrollado. Esta automatización viene recogida en un fichero denominado Dockerfile. Este fichero que se encuentra en el anexo B contiene todas las instrucciones necesarias para llevar a cabo el proceso de despliegue. Pasamos a comentar algunas de las funciones principales para tener un conocimiento sobre el funcionamiento de dicho archivo.

En primer lugar es necesario obtener una imagen base sobre la que montaremos nuestra infraestructura. Esta imagen se encuentra en los repositorios de Docker y haciendo uso de la primitiva *FROM* obtenemos la última de las versiones de Ubuntu.

```
FROM ubuntu:latest
```

Tras esta instrucción, ya tendríamos una imagen ligera de Ubuntu sobre la que podemos trabajar. A continuación se han indicado parámetros sobre el creador (*MANTAINER*) y se ha inhabilitado la interacción por consola para que el proceso sea automático y salte los pasos en los que es necesaria la acción del usuario. Estos pasos son los relativos a la configuración de la zona horaria de trabajo.

La siguiente primitiva de interés es *RUN*. Esta primitiva nos permite ejecutar cualquier comando que nosotros ejecutaríamos en un terminal. Si observamos el *dockerfile* vemos como se ha usado el comando *RUN* en dos ocasiones. La primera de ellas ha sido para instalar las utilidades básicas que serán necesarias más adelante (compiladores, herramientas de compresión, gestión de ficheros, etc.) y la segunda ocasión en la que se ha usado la primitiva *RUN* ha sido para ejecutar todas las instrucciones relativas a la obtención, instalación y compilación de las librerías *DCMTK* y *nloann/JSON*. Se ha hecho uso de una única primitiva a la que se le han concatenado todos estos comandos haciendo uso de la combinación de símbolos *&&* debido a que es más eficiente y menos problemático el uso de una única primitiva *RUN*. Esto se debe a que por cada primitiva se crea una instancia de la imagen y a modo de símil esto consistiría en ir construyendo la imagen por capas. Cuantas más capas tengamos menos eficiente es la creación de la imagen, más tamaño ocupará y más problemas tendremos a la hora de movernos por directorios, por ello se ha hecho uso de una única capa donde se han ejecutado todos los comandos.

También debemos destacar una primitiva usada, *COPY*. Esta es una primitiva muy útil debido a que nos permite copiar desde la máquina anfitrión ficheros que se encuentren en el mismo directorio que el *dockerfile* a la imagen que se está creando. En nuestro caso se ha hecho uso de *COPY* para copiar las instrucciones de compilación, las instrucciones de creación del proyecto *CMake* y el código fuente del módulo desarrollado.

Por último, a modo de final se hace uso de la primitiva *ENTRYPOINT*. A esta primitiva se le indica entre corchetes el comando que se ejecutaría desde consola para ejecutar el módulo desarrollado. Como se puede observar, simplemente se trata del lanzamiento del archivo ejecutable (*NBStructuredReport*) obtenido en pasos previos de compilación en el directorio *project*.

Una vez tenemos el *dockerfile* listo para el despliegue, se debe usar *docker* para lanzarlo y crear la imagen de la siguiente forma. En este caso el parámetro *-t* nos sirve para indicar el nombre de la imagen.

```
docker build -t nbofficial ./
```

Una vez ejecutada esta instrucción y tras todo el proceso de instalación y configuración se añadiría a nuestra lista de imágenes la imagen creada.


```
[david@HP-Pavillon-dv6-Notebook-PC ~]$ docker image ls -a
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nbofficial	latest	925fc3444751	46 hours ago	1.42GB
<none>	<none>	e34f893d2755	46 hours ago	403MB
<none>	<none>	5f3be761ff48	46 hours ago	403MB
<none>	<none>	76a7c864c423	46 hours ago	403MB
<none>	<none>	90254096314d	46 hours ago	402MB
<none>	<none>	1db09954be17	46 hours ago	402MB
nbtest	6ready	703ba5f6e03d	46 hours ago	1.43GB
nbtest	5codenotrunning	0711ee0cbb0e	47 hours ago	1.42GB

Figura 6.2: Imagen creada tras la ejecución del dockerfile

Una vez creada la imagen podemos lanzar contenedores de esa imagen que nos permiten ejecutar la aplicación o acceder a ellos como si de una máquina independiente se tratase. En la figura 6.3 vemos como se ha creado un contenedor Docker sobre la imagen creada. El ID de la imagen que aparece en la columna *IMAGE* pertenece a la imagen lanzada anteriormente. La creación de contenedores se realiza con la ejecución del comando que se muestra a continuación. Donde se le indica el nombre del contenedor con la opción `-name` y el ID de la imagen sobre la que se quiere montar con la opción `-it`. La opción `bash` se usa cuando queremos acceder al contenedor como si de un equipo independiente se tratase. Se nos abrirá una sesión de esa imagen que estará estructurada tal como se ha especificado en el dockerfile.

```
docker run --name containerNB -it 925fc3444751 bash
```

```
[david@HP-Pavillon-dv6-Notebook-PC ~]$ docker container ls -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b1a39b82cb29	925fc3444751	"bash"	46 hours ago	Exited (0) 46 hours ago		containerNB
6a614e464c5d	76a7c864c423	"/bin/sh -c 'mkdir /_'"	46 hours ago	Exited (1) 46 hours ago		exciting_lehmann
8bfa53974017	703ba5f6e03d	"bash"	46 hours ago	Exited (0) 46 hours ago		containerTestNB

Figura 6.3: Contenedor de la imagen creada

Esta es la forma de trabajar con Docker. La creación de imágenes que simulan entornos que puedan funcionar independientemente del lugar es lo que hace de Docker una potente herramienta que nos permite de una forma fácil y rápida hacer un despliegue óptimo de nuestras aplicaciones. Se ha conseguido implementar una solución que automatiza la creación de un entorno de ejecución del módulo que se ha desarrollado.

CAPÍTULO 7

Conclusiones

En este capítulo en primer lugar se va a presentar el resultado del proyecto y la relación de objetivos y subobjetivos alcanzados a lo largo del desarrollo del proyecto. En segundo lugar se comentarán las dificultades encontradas y los aprendizajes obtenidos del mismo. Finalmente se analizará la relación del trabajo desarrollado con los estudios cursados donde se propondrá una relación de competencias que han sido trabajadas a lo largo del trabajo de fin de máster.

En el presente TFM, se ha desarrollado un módulo para la creación de informes estructurados siguiendo el estándar DICOM a raíz de la información introducida en la plataforma desarrollada por QUIBIM para el proyecto PRIMAGE.

Repasando los objetivos y subobjetivos plasmados en la sección 1.2, se confirma que se han alcanzado positivamente. A lo largo del desarrollo del TFM se han ido alcanzando los subobjetivos indicados para llevar a cabo el desarrollo del proyecto.

El subobjetivo que se ha alcanzado en primer lugar es el de la selección de la herramienta para la implementación de informes DICOM-SR en la plataforma PRIMAGE. Para alcanzarlo se ha realizado un análisis de los distintos estándares y la librería necesaria para implementar informes estructurados DICOM-SR. Para el subobjetivo relacionado con el diseño de las plantillas DICOM SR a partir de los formularios existentes, se han creado las tablas anexadas con todas las variables del paciente codificadas siguiendo terminologías estándar o propias. El subobjetivo relacionado con la codificación y desarrollo de los informes existentes en DICOM SR se ha conseguido al desarrollar el módulo en C++ con las librerías necesarias para ello.

Finalmente, para despliegue del módulo de creación de informes en la plataforma del proyecto PRIMAGE se ha hecho un estudio de las herramientas confeccionadas para este fin existentes y se ha procedido a hacer uso de contenedores Docker para conseguirlo.

La envergadura del proyecto PRIMAGE, la gran cantidad de variables recogidas para cada paciente junto con el uso de múltiples tecnologías para desarrollo y despliegue han hecho de este trabajo uno de los más completos que haya podido realizar hasta el día de hoy.

Entre las dificultades que se han encontrado, la primera de ellas ha sido tener que aprender sobre terminologías y estándares sanitarios para la codificación de

todas las variables en la que se ha tenido que realizar una búsqueda exhaustiva de términos, siglas y nomenclaturas de diversos campos del ámbito sanitario. La siguiente dificultad a resaltar ha sido la dificultad para integrar la librería DCMTK en el proyecto. La poca información al respecto sobre esta librería unida a la poca soltura en la creación y desarrollo de programas en C++ han hecho que se haya invertido mucho tiempo al arranque de la programación. A medida que la curva de aprendizaje se superaba todo ha sido mucho más ligero. Se ha decidido especificar en este TFM cómo crear proyectos integrando esta librería tanto en Windows como en Linux ya que resulta muy útil para proyectos del ámbito sanitario y servirá de ayuda a futuros desarrolladores. La tercera dificultad a la que nos hemos tenido que enfrentar ha sido al aprendizaje de la herramienta de despliegue Docker, ya que nunca antes me había encargado de la parte de despliegue de una aplicación. Esta etapa conforma un valor añadido al proyecto ya que se culmina el trabajo con esta integración en una plataforma de un proyecto de gran envergadura a nivel europeo.

Para concluir, con este TFM se ha aprendido en que consiste el estándar DICOM y más específicamente el servicio de informes estructurados. Se ha aprendido a integrar librerías en proyectos CMake y se ha trabajado con el lenguaje C++. Hemos podido realizar un análisis de los datos y de distintos formatos empleados (JSON, XML, DCM). Se ha podido desarrollar un proyecto de principio a fin, llegando incluso a la parte de despliegue, parte que ni en el grado ni en el máster se llegan a ver en profundidad. Por último, se ha encontrado un área de aplicación de la ingeniería informática que suele ser menos conocida, pero que tiene un gran potencial tanto a nivel de estandarización de documentos, pasando por el análisis por computador de imágenes médicas y llegando hasta la explotación de datos de pacientes para poder dar soporte a la toma de decisiones de los profesionales sanitarios.

7.1 Relación del trabajo desarrollado con los estudios cursados

Para la correcta consecución del proyecto ha sido necesario realizar una combinación de los conocimientos que se han ido adquiriendo con el paso por el Grado en Ingeniería Informática y conocimientos extraídos del Máster en Ingeniería Informática. Se han necesitado conocimientos de programación, de informática médica, de integración de aplicaciones así como conocimiento sobre los distintos estándares de codificación.

En conclusión, se puede decir que se trata de un proyecto ligado a los estudios cursados donde se ha dado solución a un problema haciendo uso de las tecnologías más avanzadas disponibles para ello. Para finalizar se presentan las competencias transversales que se evalúan a lo largo del proyecto y que han ido aflorando a medida que el TFM iba siendo desarrollado.

- CT_1. Comprensión e integración: la integración de distintas tecnologías ha permitido el desarrollo del TFM.

- CT_2. Aplicación y pensamiento práctico: Se ha realizado un estudio del área de aplicación así como de las opciones ya disponibles en el mercado.
- CT_3. Análisis y resolución de problemas: se ha hecho uso del conocimiento para la resolución de los problemas que han ido surgiendo.
- CT_5. Diseño y proyecto. Para llevar a cabo el TFM se ha seguido una metodología expresada en la memoria.
- CT_6. Trabajo en equipo y liderazgo: el TFM ha sido llevado tanto por el tutor UPV como de la tutora externa, la integración con el miembro de la empresa era clave en el proyecto debido a que la empresa QUIBIM es la desarrolladora de la plataforma web del proyecto PRIMAGE.
- CT_10. Problemas contemporáneos: el TFM da solución a problemas actuales en el ámbito de la informática médica.
- CT_11. Aprendizaje permanente: ha sido necesario adquirir conocimientos sobre conceptos y herramientas para el desarrollo del TFM.
- CT_12. Planificación y gestión del tiempo: ha sido necesaria una gestión del tiempo para poder acabar el TFM a tiempo.

CAPÍTULO 8

Trabajos futuros

Para finalizar, en esta sección se van a presentar los avances que se podrían implementar desde el proyecto que se ha realizado para lograr mejoras en el módulo.

En primer lugar se podría realizar una implementación genérica para cualquier tipo de formulario web que generase un objeto JSON. De cara a la expansión de la estandarización de informes estructurados a otras enfermedades distintas a las que se exponen en este proyecto se propone el desarrollo de un módulo que haciendo uso de una base de datos de términos estandarizados recorra el JSON extrayendo las variables por sí mismo y codificando automáticamente tanto las variables como los valores que estas puedan adquirir. Con el desarrollo de este módulo podríamos unificar cada uno de los módulos que serían necesarios para cada formulario web distinto en uno. También se mejoraría el mantenimiento del módulo ante cambios en el formulario web, ya que actualmente se deben revisar los cambios uno por uno y ver en qué medida afectan al módulo, y debido la gran cantidad de líneas de código por las que está compuesto resulta un trabajo que requiere de demasiado tiempo, que con el módulo genérico el módulo se seguiría trabajando independientemente de los cambios en la estructura del formulario.

Otro aspecto, que por falta de tiempo no se ha podido abordar ha sido la mejora de la eficiencia de ejecución. Actualmente el módulo se ejecuta de forma secuencial de principio a fin, un análisis de los objetos JSON de forma concurrente por parte de distintos hilos de ejecución mejoraría los tiempos.

Finalmente, se propone que en la línea del manejo de datos de informes se podría desarrollar una herramienta de indexación entre informes donde se pudiesen realizar búsquedas de informes de pacientes que tuviesen parámetros comunes, esto permitiría realizar una rápida consulta de pacientes similares que se han tratado con anterioridad para poder consultar el seguimiento que se les hizo y poder tomar unas decisiones u otras en base a los informes de otros pacientes similares con un mismo diagnóstico y un pronóstico similar al actual. Esta información conformaría una gran base de datos sobre la enfermedad que permitiría mejorar con el tiempo diagnósticos y tratamientos.

Bibliografía

- [1] "A.3 structure of dicom sr documents." http://dicom.nema.org/dicom/2013/output/chtml/part20/sect_A.3.html, (accedido 4 junio, 2020).
- [2] D. A. Clunie, *DICOM structured reporting*. PixelMed publishing, 2000.
- [3] N. BioPortal, "Snomed ct: Ncbo bioportal." <https://bioportal.bioontology.org/ontologies/SNOMEDCT>, (accedido 6 junio, 2020).
- [4] N. BioPortal, "Radlex: Ncbo bioportal." <https://bioportal.bioontology.org/ontologies/RADLEX>, (accedido 6 junio, 2020).
- [5] P. Project, "Primage project." <https://www.primageproject.eu/>, 2018 (accedido 11 mayo, 2020).
- [6] L. Martí-Bonmatí, Á. Alberich-Bayarri, R. Ladenstein, I. Blanquer, J. D. Segrelles, L. Cerdá-Alberich, P. Gkontra, B. Hero, J. García-Aznar, D. Keim, *et al.*, "Primage project: predictive in silico multiscale analytics to support childhood cancer personalised evaluation empowered by imaging biomarkers," *European Radiology Experimental*, vol. 4, pp. 1–11, 2020.
- [7] J. M. Maris and K. K. Matthay, "Molecular biology of neuroblastoma," *Journal of clinical oncology*, vol. 17, no. 7, pp. 2264–2264, 1999.
- [8] O. T. Gómez, P. P. R. López, and J. S. Bacalla, "Criterios de selección de metodologías de desarrollo de software," *Industrial data*, vol. 13, no. 2, pp. 70–74, 2010.
- [9] L. M. Fernández Cacho, M. Á. Gordo Vega, and S. Laso Cavadas, "Enfermería y salud 2.0: recursos tics en el ámbito sanitario," *Index de Enfermería*, vol. 25, no. 1-2, pp. 51–55, 2016.
- [10] A. C. Garcia and A. M. Alvarez, "Informe radiológico estructurado en el adenocarcinoma de páncreas.." <http://www.sedia.es/congreso2017/comunicaciones/posters-e/ver-poster/54929687cd5531c914d320f1b777f03f>, 2017 (accedido 2 junio, 2020).
- [11] F. Ballesteros, "Desarrollo de aplicaciones dicom para la gestión de imágenes biomédicas," *Recuperado de http://www.elai.upm.es/webantigua/spain/Investiga/GCII/personal/fballesteros/Dicom.p df*, 2003.
- [12] R. Noumeir, "Benefits of the dicom structured report." <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3045165/>, (accedido 16 junio, 2020).

- [13] G. Bora, S. Bora, S. Singh, and S. M. Arsalan, "Osi reference model: An overview," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 7, no. 4, pp. 214–218, 2014.
- [14] R. H. Dolin, L. Alschuler, C. Beebe, P. V. Biron, S. L. Boyer, D. Essin, E. Kimber, T. Lincoln, and J. E. Mattison, "The hl7 clinical document architecture," *Journal of the American Medical Informatics Association*, vol. 8, no. 6, pp. 552–569, 2001.
- [15] C. y. B. S. Ministerio de Sanidad, "¿qué es snomed ct?." <https://www.msccbs.gob.es/profesionales/hcdsns/areaRecursosSem/snomed-ct/quees.htm>, (accedido 6 junio, 2020).
- [16] RSNA, "Radlex radiology lexicon." <https://www.rsna.org/en/practice-tools/data-tools-and-standards/radlex-radiology-lexicon>, (accedido 8 junio, 2020).
- [17] C. J. McDonald, S. M. Huff, J. G. Suico, G. Hill, D. Leavelle, R. Aller, A. Forrey, K. Mercer, G. DeMoor, J. Hook, *et al.*, "Loinc, a universal standard for identifying laboratory observations: a 5-year update," *Clinical chemistry*, vol. 49, no. 4, pp. 624–633, 2003.
- [18] E. G. Brown, L. Wood, and S. Wood, "The medical dictionary for regulatory activities (meddra)," *Drug safety*, vol. 20, no. 2, pp. 109–117, 1999.
- [19] NCIT, "Ncithesaurus." <https://ncithesaurus.nci.nih.gov/ncitbrowser/>, (accedido 8 junio, 2020).
- [20] I. Blanquer, V. Hernandez, and D. Segrelles, "Trencadis-a wsrf grid middleware for managing dicom structured reporting objects," *Studies in health technology and informatics*, vol. 120, p. 381, 2006.
- [21] J. Salavert Torres, "Desarrollo e integración en trencadis de servicios grid para el almacenamiento, indexación y búsqueda de objetos dicom-sr asociados a estudios de imágenes dicom de distintos centros hospitalarios empleando componentes de glite.," 2011.
- [22] A. von Wangenheim, C. L. Barcellos Jr., R. Andrade, I. de Carlos Back Giuliano, A. F. Borgatto, and D. F. de Andrade, "Implementing dicom structured reporting in a large-scale telemedicine network," *Telemedicine journal and e-health : the official journal of the American Telemedicine Association*, vol. 19, pp. 535–541, Jul 2013.
- [23] DICOM@OFFIS, "Dcmtk - dicom toolkit." <https://dicom.offis.de/dcmtk.php.en>, (accedido 9 junio, 2020).
- [24] K. J. Dreyer, "Why ihe?," 2000.
- [25] M. Yip, "Native json benchmark." <https://github.com/miloyip/nativejson-benchmark>, (accedido 29 junio, 2020).

APÉNDICE A

Terminología propia del proyecto

"1", "PrimageNB", "Neuroblastoma structured report"
"2", "PrimageNB", "Intraspinal neuroblastoma"
"3", "PrimageNB", "IVC (Inferior vena cava) compression +/- leg edema"
"4", "PrimageNB", "Creatinine > 2x Upper Limit Normal"
"5", "PrimageNB", "Urine output < 2 mL/Kg/day"
"6", "PrimageNB", "Abnormal liver function > 2x Upper Limit Normal"
"7", "PrimageNB", "Platelets < 50 x 10⁹/L"
"8", "PrimageNB", "Bladder/bowel dysfunction secondary to mass effect"
"9", "PrimageNB", "Very large tumor volume causing concern of possible tumor rupture and/or the possible rapid development of systemic upset"
"10", "PrimageNB", "mol/sl"
"11", "PrimageNB", "Biopsy date"
"12", "PrimageNB", "Biopsy type"
"13", "PrimageNB", "Bone marrow studies"
"14", "PrimageNB", "TH above SIOPEN cut point"
"15", "PrimageNB", "PHOX2B above SIOPEN cut point"
"16", "PrimageNB", "Immunocytology result"
"17", "PrimageNB", "Immunocytology"
"18", "PrimageNB", "qRT-PCR and Immunocytology"
"19", "PrimageNB", "Other metastasis"
"21", "PrimageNB", "Method IcWGS"
"22", "PrimageNB", "Method deepWGS"
"23", "PrimageNB", "Status MYCN"
"24", "PrimageNB", "Status ALK mutation F114L"
"25", "PrimageNB", "Status ALK mutation F1245Q"
"26", "PrimageNB", "Status ALK mutation R1275Q"
"27", "PrimageNB", "NCA Number of alterations"
"28", "PrimageNB", "Gain/loss"
"29", "PrimageNB", "SCA Typical 1p-"
"30", "PrimageNB", "SCA Typical +1q"
"31", "PrimageNB", "SCA Typical +2p"
"32", "PrimageNB", "SCA Typical 3p-"
"33", "PrimageNB", "SCA Typical 4p-"
"34", "PrimageNB", "SCA Typical +11q"
"35", "PrimageNB", "SCA Typical 11q-"
"36", "PrimageNB", "SCA Typical +17q"

"37", "PrimageNB", "SCA Atypical"
"38", "PrimageNB", "SCA Number of alterations"
"39", "PrimageNB", "SCA Atypical Alteration"
"40", "PrimageNB", "Method WGS"
"41", "PrimageNB", "Status ALT"
"42", "PrimageNB", "Method PNA tel FISH"
"43", "PrimageNB", "Method PNA tel FISH + PML-body"
"41", "PrimageNB", "Status ALT"
"44", "PrimageNB", "Number of datasets available"
"45", "PrimageNB", "Type of analysis"
"46", "PrimageNB", "Score SIOOPEN"
"47", "PrimageNB", "Ipsilateral tumor extension"
"48", "PrimageNB", "Tumor encasing carotid and/or vertebral artery and/or internal jugular vein"
"49", "PrimageNB", "Tumor extending to base of skull"
"50", "PrimageNB", "Tumor compressing the trachea"
"51", "PrimageNB", "Cervico thoracic junction"
"52", "PrimageNB", "Tumor encasing brachial plexus roots"
"53", "PrimageNB", "Tumor encasing subclavial vessels and/or vertebral and/or carotid artery"
"50", "PrimageNB", "Tumor compressing the trachea"
"54", "PrimageNB", "Tumor encasing the aorta and/or major branches"
"55", "PrimageNB", "Tumor compressing the trachea and/or principal bronchi"
"56", "PrimageNB", "Lower mediastinal tumor, infiltrating the costovertebral junction T9 and T12"
"57", "PrimageNB", "Tumor infiltrating the porta hepatis and/or the hepatoduodenal ligament"
"58", "PrimageNB", "Tumor encasing branches of the superior mesenteric artery at the mesenteric root"
"59", "PrimageNB", "Tumor encasing the origin of the celiac axis and/or of the superior mesenteric artery"
"60", "PrimageNB", "Tumor invading one or both renal pedicles"
"61", "PrimageNB", "Tumor encasing the aorta and/or vena cava"
"62", "PrimageNB", "Tumor encasing the iliac vessels"
"63", "PrimageNB", "Pelvic tumor crossing the sciatic notch"
"64", "PrimageNB", "Intraspinal tumor extension whatever the location provided that more than 1/3 of the spinal canal in the axial plane is invaded and/or the perimedullary leptomenigeal spaces are not visible and/or the spinal cord signal is abnormal "
"65", "PrimageNB", "Infiltration adjacent organs"
"66", "PrimageNB", "Infiltration of adjacent organs/structures - Pericardium, ?diaphragm, kidney, liver, duodenopancreatic block and mesentery"
"67", "PrimageNB", "Conditions to be recorded but not considered IDRFs"
"68", "PrimageNB", "Multifocal primary tumours"
"69", "PrimageNB", "Orbita"
"70", "PrimageNB", "LINES group"
"71", "PrimageNB", "End date of chemotherapy"
"72", "PrimageNB", "Number of cycles"
"73", "PrimageNB", "CADO"

"75", "PrimageNB", "First treatment completely finished"
"76", "PrimageNB", "Low/Intermediate risk patients"
"77", "PrimageNB", "Pleura lung"
"78", "PrimageNB", "Periphereal blood"
"79", "PrimageNB", "Date of last visit"
"80", "PrimageNB", "Status TERT Breakpoint/Rearrangement"
"81", "PrimageNB", "N4"
"82", "PrimageNB", "N5"
"83", "PrimageNB", "N6"
"84", "PrimageNB", "N7"
"85", "PrimageNB", "Modified N7"
"86", "PrimageNB", "OMS (Opsocionus-Myocionus Syndrome)"
"87", "PrimageNB", "Other associated diseases"
"88", "PrimageNB", "Nodular glanglioneuroblastoma"
"89", "PrimageNB", "Intermixed glanglioneuroblastoma"
"90", "PrimageNB", "Maturing ganglioneuroma"
"91", "PrimageNB", "No/insufficient material"
"92", "PrimageNB", "Not otherwise specified (NOS)"
"93", "PrimageNB", "Differentiating"
"94", "PrimageNB", "Poorly differentiated"
"95", "PrimageNB", "Undifferentiated"
"96", "PrimageNB", "Cannot be determined"
"97", "PrimageNB", "Metastasis excluding bone marrow and lymph nodes (distant)"
"98", "PrimageNB", "Tru-cut/core needle biopsy"
"99", "PrimageNB", "Biopsy, not specified"
"100", "PrimageNB", "Not amplified"
"101", "PrimageNB", "Amplification heterogeneous"
"102", "PrimageNB", "Sanger seq"
"103", "PrimageNB", "NGS panel sequencing"
"104", "PrimageNB", "WES"
"105", "PrimageNB", "Diploid/Tetraploid"
"106", "PrimageNB", "Triploid/Hexaploid"
"107", "PrimageNB", "mRNA-array"
"108", "PrimageNB", "SNParray"
"109", "PrimageNB", "NEOplus assay"
"110", "PrimageNB", "New Oncology"
"111", "PrimageNB", "Sanger"
"112", "PrimageNB", "Oncoscan"
"113", "PrimageNB", "Infinum Human Methylation 450 K BeadChips"
"114", "PrimageNB", "GSE49711"
"115", "PrimageNB", "GSE94035"
"116", "PrimageNB", "GSE120572"
"117", "PrimageNB", "GSE49710"
"118", "PrimageNB", "E-MTAB-8248"
"119", "PrimageNB", "GSE45547"
"120", "PrimageNB", "EGAS00001001308"
"121", "PrimageNB", "EGAS00001003244"
"122", "PrimageNB", "GSE84291"
"123", "PrimageNB", "GSE79187"

"124", "PrimageNB", "GSE73515"
"125", "PrimageNB", "SPECT and planar"
"126", "PrimageNB", "1"
"127", "PrimageNB", "2"
"128", "PrimageNB", "2A"
"129", "PrimageNB", "2B"
"130", "PrimageNB", "3"
"131", "PrimageNB", "4"
"132", "PrimageNB", "4S"
"133", "PrimageNB", "L1"
"134", "PrimageNB", "L2"
"135", "PrimageNB", "M"
"136", "PrimageNB", "MS"
"137", "PrimageNB", "LINES"
"138", "PrimageNB", "LINESG 1"
"139", "PrimageNB", "LINESG 2"
"140", "PrimageNB", "EUNS"
"141", "PrimageNB", "INES"
"142", "PrimageNB", "HR-NBL1"
"143", "PrimageNB", "NB97"
"144", "PrimageNB", "NB2004"
"145", "PrimageNB", "Chemotherapy and surgery"
"146", "PrimageNB", "Conclusion metastatic response"
"147", "PrimageNB", "mCR"
"148", "PrimageNB", "mPR"
"149", "PrimageNB", "mSD"
"150", "PrimageNB", "mPD"
"151", "PrimageNB", "SIOOPEN"
"152", "PrimageNB", " Other metastatic sites"

APÉNDICE B

Dockerfile

```
FROM ubuntu:latest
MAINTAINER DAVID d.chichell@gmail.com
ARG DEBIAN_FRONTEND=noninteractive
RUN apt update

# Install
RUN apt -y install build-essential libssl-dev wget cmake unzip

COPY build.sh /project/build.sh
COPY CMakeLists.txt /project/CMakeLists.txt
COPY Main.cc /project/Main.cc

# Install DCMTK
RUN mkdir /home/DCMTK &&\
cd /home/DCMTK &&\
wget https://dicom.offis.de/download/dcmtd/dcmtd365/dcmtd-3.6.5.tar.gz -P
/home/DCMTK &&\
tar -zxvf /home/DCMTK/dcmtd-3.6.5.tar.gz -C /home/DCMTK &&\
mkdir /home/DCMTK/dcmtd-3.6.5-build &&\
cd /home/DCMTK/dcmtd-3.6.5-build &&\
cmake /home/DCMTK/dcmtd-3.6.5 &&\
make -j$(nproc) &&\
make install &&\
cd / &&\
# v3.0

# Install JSON parser
mkdir /home/JSON &&\
cd /home/JSON &&\
wget https://github.com/nlohmann/json/archive/develop.zip -P /home/JSON &&\
unzip /home/JSON/develop.zip -d /home/JSON &&\
mkdir /home/JSON/json-develop-build &&\
cd /home/JSON/json-develop-build &&\
cmake /home/JSON/json-develop &&\
make -j$(nproc) &&\
make install &&\
```

```
cd / &&\
# v4.0

# Compile code
cd /project &&\
cmake ./ &&\
make &&\
cd /

ENTRYPOINT ["/project/./NBStrucuredReport"]
```

APÉNDICE C
Tabla de relación de términos NB

1. Patient data [HEADER DATA]

Variable	Labels PRIMAGE platform
Patient ID	-
Sex	Male Female

2. Diagnosis [SNOMED_CT, 439401001, VT_Container]

2.1. Associated symptoms [LOINC, MTHU021540, VT_Container]

Section	Variable	Labels PRIMAGE platform
	Diagnosis date (Biopsy date) [SNOMED_CT, 432213005, VT_DateTime]	-
	Age at diagnosis (months) [SNOMED_CT,423493009,VT_Text]	-
	Incidental finding [NCIt,C48619,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	First symptoms date [COVIDCRFRAPID,Symptom_onset,VT_DateTime]	-
LTS (Life Threatening Symptoms) [ONTOPARON,RisqueVital,VT_Container]	Intraspinal neuroblastoma [PrimageNB,2,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]

PRIMAGE: Clinical variables

		Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Pain requiring opiate treatment [SNOMED_CT, 726582005, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Vomiting needing nasogastric/IV support [SNOMED_CT,419907002,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Weight loss > 10% body weight Weight loss [AI-RHEUM,WTLOS,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Respiratory distress without evidence of infection [SNOMED_CT,271825005,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Hypertension [RADLEX,RID39097,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	IVC (Inferior vena cava) compression ± leg edema [PrimageNB,3,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Creatinine > 2x Upper Limit Normal [PrimageNB,4,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]

PRIMAGE: Clinical variables

		Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Urine output < 2 mL/Kg/day [PrimageNB,5,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Hydroureter / hydronephrosis [SNOMED_CT,69758005,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Abnormal liver function > 2x Upper Limit Normal [PrimageNB,6,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Evidence of DIC (Disseminated Intravascular Coagulation) [SNOMED_CT,67406007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Platelets < 50 x 10 ⁹ /L [PrimageNB,7,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Bladder/bowel dysfunction secondary to mass effect [PrimageNB,8,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Very large tumor volume causing concern of possible tumor rupture and/or the	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]

PRIMAGE: Clinical variables

	possible rapid development of systemic upset [PrimageNB,9,VT_Code]	Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
Other associated symptoms [LOINC,LP133892-2,VT_Container]	Spinal cord syndrome [MEDR,10077133,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Symptomatic spinal cord compression ¹ [SNOMED_CT,81669005,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	OMS (Opsoclonus-Myoclonus Syndrome) [PrimageNB,86,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Watery diarrhea [MEDRA,10055422,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Horner's syndrome [SNOMED_CT,271730003,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Others [SNOMED_CT,74964007,VT_Container]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Others – Value ³ [SNOMED_CT,394617004,VT_Text]	-

PRIMAGE: Clinical variables

Other associated diseases [PrimageNB,87,VT_Code]	NF (Neurofibromatosis) [SNOMED_CT,81669005,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Genetic syndrome [Read Codes, Clinical Terms Version 3,Xa4hE,VT_Container]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Genetic syndrome – Value ² [SNOMED_CT,394617004,VT_Text]	-
	Others [SNOMED_CT,74964007,VT_Container]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown [SNOMED_CT,261665006,DSRCodedEntryValue]
	Others – Value ³ [SNOMED_CT,394617004,VT_Text]	-

¹To specify only if Spinal cord syndrome is Yes

²The name of the genetic syndrome should be filled when Genetic syndrome is Yes.

³The name of other associated symptoms only have to be entered if Others is Yes

2.2. Laboratory [SNOMED_CT,261904005,VT_Container]

2.2.1. LDH

Variable	Labels PRIMAGE platform
LDH [MEDRA,10024048,VT_Container]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]

PRIMAGE: Clinical variables

LDH unit ¹ [SNOMED_CT,767525000,VT_Code]	IU/L [SNOMED_CT,767525000,DSRCodedEntryValue] µmol/sl [PrimageNB,10,DSRCodedEntryValue]
Upper normal value ¹ [NCIt,C25706,VT_Text]	-
LDH result ¹ [SNOMED_CT,250644007,VT_Text]	-
Elevated [SNOMED_CT,396697000,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]

¹These variables appear and should be filled when LDH is Evaluated

2.2.2. Pathology tumor

Variable	Labels PRIMAGE platform
Pathology tumor [NCIt,C19097,VT_Container]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Biopsy date ¹ [PrimageNB,11,VT_DateTime]	-
Histology type ¹ [LOINC,MTHU015568,VT_Text]	Neuroblastoma [SNOMED_CT,87364003,StringValue] Ganglioneuroblastoma [SNOMED_CT,69515008,StringValue] Nodular ganglioneuroblastoma [PrimageNB,88,StringValue] Intermixed ganglioneuroblastoma [PrimageNB,89,StringValue] Ganglioneuroma [SNOMED_CT,116371000119107,StringValue] Mature ganglioneuroma

PRIMAGE: Clinical variables

	<p>Maturing ganglioneuroma [PrimageNB,90,StringValue] No/insufficient material [PrimageNB,91,StringValue] Not available [LOINC, LA7338-2, StringValue] Not otherwise specified (NOS) [PrimageNB,92,StringValue]</p>
<p>Grade of differentiation (Neuroblastic component)¹ [SNOMED_CT,263522009,VT_Text]</p>	<p>Differentiating [PrimageNB,93,StringValue] Poorly differentiated [PrimageNB,94,StringValue] Undifferentiated [PrimageNB,95,StringValue] Cannot be determined [PrimageNB,96,StringValue] Not otherwise specified (NOS) [PrimageNB,92,StringValue]</p>
<p>Mitosis-Karyorrhexis Index (MKI)¹ [NCIt,C106317,VT_Code]</p>	<p>High [SNOMED_CT,75540009,DSRCodedEntryValue] Intermediate [SNOMED_CT,11896004,DSRCodedEntryValue] Low [SNOMED_CT,62482003,DSRCodedEntryValue] Not evaluable [SNOMED_CT,75540009,DSRCodedEntryValue] Not done [SNOMED_CT, 385432009, DSRCodedEntryValue] Not applicable</p>

PRIMAGE: Clinical variables

	[SNOMED_CT,75540009,DSRCodedEntryValue]
Type of sample ¹ [RCTV2,4I2..00,VT_Text]	Primary tumor site [SNOMED_CT,399687005,StringValue] Lymph nodes (distant) [MESH,D008198,StringValue] Bone marrow [RADLEX,RID38594,StringValue] Metastasis excluding bone marrow and lymph nodes (distant) [PrimageNB,97,StringValue] Other [SNOMED_CT,74964007,StringValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Biopsy type ^{1 2} [PrimageNB,12,VT_Text]	Fine needle aspiration (FNA) [RCD,X79u1,StringValue] Tru-cut/core needle biopsy [PrimageNB,98,StringValue] Excisional biopsy [SNOMED_CT,8889005,StringValue] Surgical resection [LOINC,LA17736-2,StringValue] Biopsy, not specified [PrimageNB,99,StringValue] Unknown[SNOMED_CT,261665006,StringValue]

¹These variables appear and should be filled when Pathology tumor is Evaluated

²Biopsy type should be specified when Type of Sample is Primary tumor site, Lymph nodes (distant) or Metastasis excluding bone marrow and lymph nodes (distant).

2.2.3. Bone marrow studies

Variable	Labels PRIMAGE platform
Bone marrow studies [PrimageNB,13,VT_Container]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Bone marrow study date ¹	-

PRIMAGE: Clinical variables

[SNOMED_CT,410672004,VT_DateTime]	
Aspirate (cytology) ¹ [MEDdra,10005982,VT_Container]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Aspirate (cytology) – Result ^{1 2} [SNOMED_CT,394617004,VT_Code]	Positive [SNOMED_CT,10828004,DSRCodedEntryValue] Negative [SNOMED_CT,373067005,DSRCodedEntryValue] Inconclusive [SNOMED_CT,419984006,DSRCodedEntryValue]
Trephine biopsy ¹ [SNOMED_CT,50590007,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Trephine biopsy – Result ^{1 3} [SNOMED_CT,394617004,VT_Code]	Positive [SNOMED_CT,10828004,DSRCodedEntryValue] Negative [SNOMED_CT,373067005,DSRCodedEntryValue] Inconclusive [SNOMED_CT,419984006,DSRCodedEntryValue]
Minimal Residual Disease (MRD) ¹ [MEDdra,10079987,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Minimal Residual Disease (MRD) -Method ^{1 4} [SNOMED_CT,260686004,VT_Code]	qRT-PCR [IOBC,201606016242659673,DSRCodedEntryValue] Immunocytology [PrimageNB,17,DSRCodedEntryValue] qRT-PCR and Immunocytology [PrimageNB,18,DSRCodedEntryValue]
Minimal Residual Disease (MRD) – TH above SIOPEN cut point ^{1 4 5} [PrimageNB,14,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Minimal Residual Disease (MRD) – PHOX2B above SIOPEN cut point ^{1 4 5} [PrimageNB,15,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Minimal Residual Disease (MRD) – Immunocytology result ^{1 4 6} [SNOMED_CT,394617004,VT_Code]	Positive [SNOMED_CT,10828004,DSRCodedEntryValue] Negative [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,,LA21136-9,DSRCodedEntryValue]

¹These variables appear and should be filled when Bone marrow studies is Evaluated

PRIMAGE: Clinical variables

²The result of Aspirate should be included if Aspirate (cytology) is Evaluated

³The result of Trephine biopsy should be included if Trephine biopsy is Evaluated

⁴These variables appear and should be filled when Minimal Residual Disease (MRD) is Evaluated

⁵Variables to fill when Minimal Residual Disease (MRD) – Method is qRT-PCR

⁶Variables to fill when Minimal Residual Disease (MRD) – Method is Immunocytology

2.2.4. Liquid biopsy (peripheral blood)

Variable	Labels PRIMAGE platform
Liquid biopsy (peripheral blood) [NCIt,C135727,VT_Container]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Minimal Residual Disease (MRD) ¹ [MEDdra,10079987,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Minimal Residual Disease (MRD) -Method ^{1 4} [SNOMED_CT,260686004,VT_Code]	qRT-PCR [IOBC,201606016242659673,DSRCodedEntryValue]
Minimal Residual Disease (MRD) – TH above SIOPEM cut point ^{1 4 5} [PrimageNB,14,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result[LOINC,LA21136-9,DSRCodedEntryValue]
Minimal Residual Disease (MRD) – PHOX2B above SIOPEM cut point ^{1 4 5} [PrimageNB,15,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result[LOINC,LA21136-9,DSRCodedEntryValue]

¹Variables to fill when Liquid biopsy (peripheral blood) is Evaluated

²Variables to fill when Minimal Residual Disease (MRD) is Evaluated

³Variables to fill when Minimal Residual Disease (MRD) – Method is qRT-PCR

2.2.5. *Molecular studies*

Variable	Labels PRIMAGE platform
Molecular studies [LOINC,MTHU012766,VT_Container]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Sample type ¹ [RCTV2,4I2..00,VT_Code]	Primary tumor [OCHV,42697,DSRCodedEntryValue] Bone marrow [RADLEX,RID38594,DSRCodedEntryValue] Lymph nodes [MESH,D008198,DSRCodedEntryValue] Other metastasis [PrimageNB,19,DSRCodedEntryValue]
Status MYCN ¹ [PrimageNB,23,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Status MYCN – Method/s ¹² [SNOMED_CT,260686004,VT_Code]	FISH [SNOMED_CT,426329006,DSRCodedEntryValue] SNPa [ProteinOntology,_DDB_G0285111,DSRCodedEntryValue] PCR [LOINC,LA26418-6,DSRCodedEntryValue] Southern Blot [MolecularInteraction,MI_0103,DSRCodedEntryValue] MLPA [IOBC,200906058748194996,DSRCodedEntryValue] aCGH [ERO,ERO_0001818,DSRCodedEntryValue] IcWGS [PrimageNB,21,DSRCodedEntryValue] deepWGS [PrimageNB,22,DSRCodedEntryValue]
Status MYCN - Status ¹² [PrimageNB,23,VT_Code]	Not amplified [PrimageNB,100,DSRCodedEntryValue] Gain [NCIt,C37956,DSRCodedEntryValue] Amplification [NCIt,C25418,DSRCodedEntryValue] Amplification heterogeneous [PrimageNB,101,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Status ALK amplification ¹	Evaluated [NCIt,C25214,DSRCodedEntryValue]

PRIMAGE: Clinical variables

[NCIt,C131136,VT_Code]	Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
<p>Status ALK amplification – Method/s¹³ [SNOMED_CT,260686004,VT_Code]</p>	<p>FISH [SNOMED_CT,426329006,DSRCodedEntryValue] SNPa [ProteinOntology,_DDB_G0285111,DSRCodedEntryValue] PCR [LOINC,LA26418-6,DSRCodedEntryValue] Southern Blot [MolecularInteraction,MI_0103,DSRCodedEntryValue] MLPA [IOBC,200906058748194996,DSRCodedEntryValue] aCGH [ERO,ERO_0001818,DSRCodedEntryValue] IcWGS [PrimageNB,21,DSRCodedEntryValue] deepWGS [PrimageNB,22,DSRCodedEntryValue]</p>
<p>Status ALK amplification – Status¹³ [NCIt,C131136,VT_Code]</p>	<p>Not amplified [PrimageNB,100,DSRCodedEntryValue] Gain [NCIt,C37956,DSRCodedEntryValue] Amplification [NCIt,C25418,DSRCodedEntryValue] Amplification heterogeneous [PrimageNB,101,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]</p>
<p>Status ALK mutation¹ [MEDdra,10071999,VT_Code]</p>	<p>Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]</p>
<p>Status ALK mutation – Method/s¹⁴ [SNOMED_CT,260686004,VT_Code]</p>	<p>Sanger seq [PrimageNB,102,DSRCodedEntryValue] NGS panel sequencing [PrimageNB,103,DSRCodedEntryValue] WES [PrimageNB,104,DSRCodedEntryValue] WGS [PrimageNB,21,DSRCodedEntryValue] ddPCR</p>

PRIMAGE: Clinical variables

	[NCIt,C166064,DSRCodedEntryValue]
Status ALK mutation – F114L ¹⁴ [PrimageNB,24,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Status ALK mutation – F1245Q ¹⁴ [PrimageNB,25,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Status ALK mutation – R1275Q ¹⁴ [PrimageNB,26,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Status ALK mutation – Other ¹⁴ [SNOMED_CT,74964007,VT_Text]	-
Ploidy ¹ [SNOMED_CT,88183001,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Ploidy – Method/s ¹⁵ [SNOMED_CT,260686004,VT_Code]	Karyotyping deepWGS [MEDdra,10023306,DSRCodedEntryValue] FACS [IOBC,200906032506360760,DSRCodedEntryValue] SNPa [ProteinOntology,_DDB_G0285111,DSRCodedEntryValue] aCGH [ERO,ERO_0001818,DSRCodedEntryValue] IcWGS [PrimageNB,21,DSRCodedEntryValue] deepWGS [PrimageNB,22,DSRCodedEntryValue]
Ploidy – Status ¹⁵ [SNOMED_CT,88183001,VT_Code]	Diploid/Tetraploid [PrimageNB,105,DSRCodedEntryValue] Triploid/Hexaploid [PrimageNB,106,DSRCodedEntryValue] Pentaploid [LOINC,PATO_0001383,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]

PRIMAGE: Clinical variables

<p>NCA¹ [IOBC,200906021998510604,VT_Code]</p>	<p>Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]</p>
<p>NCA – Method/s¹⁶ [SNOMED_CT,260686004,VT_Code]</p>	<p>FISH [SNOMED_CT,426329006,DSRCodedEntryValue] SNPa [ProteinOntology,_DDB_G0285111,DSRCodedEntryValue] aCGH [ERO,ERO_0001818,DSRCodedEntryValue] IcWGS [PrimageNB,21,DSRCodedEntryValue] deepWGS [PrimageNB,22,DSRCodedEntryValue]</p>
<p>NCA – Status¹⁶ [IOBC,200906021998510604,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]</p>
<p>NCA – Number of alterations¹⁶⁷ [PrimageNB,27,VT_Text]</p>	<p>-</p>
<p>NCA – Alteration i – Chromosome¹⁶⁷⁸ [SNOMED_CT,91272006,VT_Text]</p>	<p>1-22, X,Y</p>
<p>NCA – Alteration i – Gain/Loss¹⁶⁷⁸ [PrimageNB,28,VT_Text]</p>	<p>Gain (+) Loss (-)</p>
<p>SCA¹ [MESH,D020754,VT_Text]</p>	<p>Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]</p>
<p>SCA – Method/s¹⁹ [SNOMED_CT,260686004,VT_Code]</p>	<p>FISH [SNOMED_CT,426329006,DSRCodedEntryValue] SNPa [ProteinOntology,_DDB_G0285111,DSRCodedEntryValue] aCGH [ERO,ERO_0001818,DSRCodedEntryValue] IcWGS [PrimageNB,21,DSRCodedEntryValue] deepWGS [PrimageNB,22,DSRCodedEntryValue]</p>
<p>SCA – Status¹⁹ [MESH,D020754,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]</p>

PRIMAGE: Clinical variables

SCA – Typical – 1p ^{-19 10} [PrimageNB,29,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Typical – +1q ^{19 10} [PrimageNB,30,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Typical – +2p ^{19 10} [PrimageNB,31,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Typical – 3p ^{-19 10} [PrimageNB,32,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Typical – 4p ^{-19 10} [PrimageNB,33,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Typical – +11q ^{19 10} [PrimageNB,34,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Typical – 11q ^{-19 10} [PrimageNB,35,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Typical – +17q ^{19 10} [PrimageNB,36,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Atypical ^{19 10} [PrimageNB,37,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
SCA – Atypical – Number of alterations ^{19 10 11} [PrimageNB,38,VT_Text]	-

PRIMAGE: Clinical variables

SCA – Atypical i - Chromosome ^{1 9 10 11 12} [SNOMED_CT,91272006,VT_Text]	1-22, X,Y
SCA – Atypical i – Arm ^{1 9 10 11 12} [NCIt,C13355,VT_Text]	p q
SCA – Atypical i – Gain/Loss ^{1 9 10 11 12} [PrimageNB,28,VT_Text]	Gain (+) [NCIt,C37956,StringValue] Loss (-) [NCIt,C25559,StringValue]
ATRX deletion ¹ [OGG,OGG_3000000546,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
ATRX deletion – Method/s ^{1 13} [SNOMED_CT,260686004,VT_Code]	SNPa [ProteinOntology,_DDB_G0285111,DSRCodedEntryValue] WGS [PrimageNB,40,VT_Code]
ATRX deletion – Status ^{1 13} [OGG,OGG_3000000546,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Status TERT ¹ [OGG,OGG_3000007015,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Status TERT – Method/s ^{1 14} [SNOMED_CT,260686004,VT_Code]	FISH [SNOMED_CT,426329006,DSRCodedEntryValue] SNPa [ProteinOntology,_DDB_G0285111,DSRCodedEntryValue] aCGH [ERO,ERO_0001818,DSRCodedEntryValue] IcWGS [PrimageNB,21,DSRCodedEntryValue] deepWGS [PrimageNB,22,DSRCodedEntryValue]
Status TERT – Breakpoint/Rearrangement ^{1 14} [PrimageNB,80,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]
Status ALT ¹ [PrimageNB,41,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]

PRIMAGE: Clinical variables

Status ALT – Method/s ^{1 15} [SNOMED_CT,260686004,VT_Code]	PNA tel FISH [PrimageNB,42,DSRCodedEntryValue PNA tel FISH + PML-body [PrimageNB,43,DSRCodedEntryValue c-circle assay [NCIt,C105650,DSRCodedEntryValue
Status ALT – Status ^{1 15} [PrimageNB,41,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] No result [LOINC,LA21136-9,DSRCodedEntryValue]

¹To be filled if Molecular studies is Evaluated

²To be filled if Status MYCN is Evaluated

³To be filled if Status ALK amplification is Evaluated

⁴To be filled if Status ALK mutation is Evaluated

⁵To be filled if Ploidy is Evaluated

⁶To be filled if NCA is Evaluated

⁷To be filled if NCA-Status is Yes

⁸Fill per number of alterations i=[1, n_alterations]

⁹To be fill if SCA is Evaluated

¹⁰To be filled if SCA-Status is Yes

¹¹To be filled if SCA-Atypical is Yes

¹²Fill per number of Atypical alterations i=[1, n_alterations]

¹³To be filled if ATRX deletion is Evaluated

¹⁴To be filled if Status TERT is Evaluated

¹⁵To be filled if Status ALT is Evaluated

2.2.6. *GPOH gene expression classifier*[ACGT - MO,GPOH-PID,VT_Container]

Variable	Labels PRIMAGE platform
GPOH gene expression classifier [ACGT - MO,GPOH-PID,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Result [SNOMED_CT,394617004,VT_Code]	Favorable [NCIt,C102560,DSRCodedEntryValue] Unfavorable [NCIt,C102561,DSRCodedEntryValue]

2.2.7. *OMICS dataset* [SWO,SWO_0000640,VT_Container]

Variable	Labels PRIMAGE platform
OMICS dataset available [SWO,SWO_0000640,VT_Container]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Number of datasets available ¹ [PrimageNB,44,VT_Text]	-
Type of analysis i ¹² [PrimageNB,45,VT_Text]	mRNA-sequencing [NCIt,C129432,StringValue] mRNA-array [PrimageNB,107,StringValue] Targeted sequencing [HHEAR,HHEAR_00251,StringValue] IcWGS [PrimageNB,21,StringValue] deepWGS [PrimageNB,22,StringValue] WES [PrimageNB,104,StringValue] SNParray [PrimageNB,108,StringValue] WES [PrimageNB,104,StringValue] Methylation array[NCIt,C165222,StringValue] Other [SNOMED_CT,74964007,StringValue]
Type of analysis – Other value ¹²³ [SNOMED_CT,74964007,VT_Text]	-
Method ¹² [SNOMED_CT,260686004,VT_Text]	Illumina HiSeq 2000 [EFO,EFO_0004203,StringValue] Oligonucleotide microarray [IOBC,200906026011769611,StringValue] Agilent [SYN,Agilent,StringValue] NEOplus assay [PrimageNB,109,StringValue] New Oncology [PrimageNB,110,StringValue] Sanger [PrimageNB,111,StringValue] ALK amplicon seq [EFO,EFO_0003747,StringValue]

PRIMAGE: Clinical variables

	<p>Illumina HiSep 2000/4000 [EFO,EFO_0008563,StringValue] Illumina HiSeq 2500 [EFO,EFO_0008565,StringValue] Affymetrix CytoScan HD Array [ERO,ERO_0001265,StringValue] Oncoscan [PrimageNB,112,StringValue] Infinum Human Methylation 450 K BeadChips [PrimageNB,113,StringValue] Illumina [SYN,Illumina,StringValue] Custom [OCHV,C0162343,StringValue] Other [SNOMED_CT,74964007,StringValue]</p>
<p>Method – Other value^{1 2 4} [SNOMED_CT,74964007,VT_Text]</p>	-
<p>Accession number or link^{1 2} [NCIt,C25402,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]</p>
<p>Number/Link^{1 2 5} [NCIt,C25402,VT_Text]</p>	<p>GSE49711 [PrimageNB,114,StringValue] GSE94035 [PrimageNB,115,StringValue] GSE120572 [PrimageNB,116,StringValue] GSE49710 [PrimageNB,117,StringValue] E-MTAB-8248 [PrimageNB,118,StringValue] GSE45547 [PrimageNB,119,StringValue] EGAS00001001308 [PrimageNB,120,StringValue] EGAS00001003244 [PrimageNB,121,StringValue] GSE84291 [PrimageNB,122,StringValue] GSE79187 [PrimageNB,123,StringValue] GSE73515 [PrimageNB,124,StringValue] Other [SNOMED_CT,74964007,StringValue]</p>
<p>Number/Link – Other value^{1 2 5 6} [SNOMED_CT,74964007,VT_Text]</p>	-

¹To be filled OMICS dataset available is Yes

PRIMAGE: Clinical variables

²One element per number of datasets available $i=[1-n_datasets]$

³To be filled if Type of analysis is Other

⁴To be filled if Method is Other

⁵To be filled if Accession number or link is Yes

⁶To be filled if Number/Link is Other

2.3. Imaging [SNOMED_CT, 363679005, VT_Container]

Note: *Purple color refers to variables that are automatically filled ones the imaging studies are uploaded to the platform*

2.3.1. MRI[RADLEX, RID10813, VT_Container]

Variable	Labels PRIMAGE platform
MRI [RADLEX, RID10813, VT_Code]	Done [SNOMED_CT, 385658003, DSR-codedEntryValue] Not done [SNOMED_CT, 385660001, DSR-codedEntryValue] Not available [LOINC, LA7338-2, DSR-codedEntryValue] Missing [LOINC, LA14698-7, DSR-codedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹ [SNOMED_CT, 410672004, VT_DateTime]	-

¹To be filled if MRI is evaluated

2.3.2. CT[RADLEX, RID10321, VT_Container]

PRIMAGE: Clinical variables

Variable	Labels PRIMAGE platform
CT [RADLEX, RID10321, VT_Code]	Done [SNOMED_CT, 385658003, DSRcodedEntryValue] Not done [SNOMED_CT, 385660001, DSRcodedEntryValue] Not available [LOINC, LA7338-2, DSRcodedEntryValue] Missing [LOINC, LA14698-7, DSRcodedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹ [SNOMED_CT, 410672004, VT_DateTime]	-

¹To be filled if CT is evaluated

2.3.3. MIBG entire body [RADLEX, RID45807, VT_Container]

Variable	Labels PRIMAGE platform
MIBG entire body [RADLEX, RID45807, VT_Container]	Done [SNOMED_CT, 385658003, DSRcodedEntryValue] Not done [SNOMED_CT, 385660001, DSRcodedEntryValue] Not available [LOINC, LA7338-2, DSRcodedEntryValue] Missing [LOINC, LA14698-7, DSRcodedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹ [SNOMED_CT, 410672004, VT_DateTime]	-

PRIMAGE: Clinical variables

Technique ¹ [SNOMED_CT, 272394005, VT_Text]	SPECT [MEDDRA,10041458,StringValue] Planar [EPO,Planar,StringValue] SPECT and planar [PrimageNB,125,StringValue] Unknown[SNOMED_CT,261665006,StringValue]
Dose unit ¹ [SNOMED_CT, 408102007, VT_Text]	mCi [SNOMED_CT,261665006,StringValue] MBq [SNOMED_CT,229034000,StringValue] Unknown[SNOMED_CT,261665006,StringValue]
Dose value ^{1,2} [OBI, OBI_0002491, VT_Text]	-
Image acquisition ¹ [NCIt, C43385, VT_Text]	4h 24h Other [SNOMED_CT,74964007,StringValue] Unknown[SNOMED_CT,261665006,StringValue]
Time after injection in h ^{1,3} [DCM, 130194, VT_Text]	
Uptake [NCIt, C71688, VT_Container] primary site ¹ [LOINC, MTHU010322, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake skeleton ¹ [FMA, fma303630, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake soft tissue ¹ [NCIt, C12471, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake liver ¹ [FMA, fma7197, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]

PRIMAGE: Clinical variables

Uptake skin ¹ [RADLEX, RID34290, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Score SIOPEN ¹ [PrimageNB, 46, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Score SIOPEN - Value ^{1 4} [SNOMED_CT, 394617004, VT_Code]	-

¹To be filled if MIBG entire body is evaluated

²Dose value has to be filled if Dose unit is not Unknown

³Time after injection in h has to be filled if Image acquisition includes Other

⁴Score SIOPEN value has to be filled if Score SIOPEN is Yes

2.3.4. PET/CT [RADLEX,RID10341,VT_Container]

Variable	Labels PRIMAGE platform
PET/CT [RADLEX,RID10341,VT_Code]	Done [SNOMED_CT, 385658003, DSRcodedEntryValue] Not done [SNOMED_CT, 385660001, DSRcodedEntryValue] Not available [LOINC, LA7338-2, DSRcodedEntryValue] Missing [LOINC, LA14698-7, DSRcodedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹ [SNOMED_CT, 410672004, VT_DateTime]	-
FDG-18F (administered dose) ¹ [SNOMED_CT, 35321007, VT_Text]	-
Uptake [NCIt, C71688, VT_Container] primary site ¹	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]

PRIMAGE: Clinical variables

[LOINC, MTHU010322, VT_Code]	
Uptake primary site - Level ^{1 2} [SNOMED_CT, 276625007, VT_Code]	Mild (< mediastinal blood pool) [SNOMED_CT,255604002,DSRCodedEntryValue] High (> liver)[SNOMED_CT,75540009,DSRCodedEntryValue] Moderate[SNOMED_CT,6736007,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,StringValue]
Uptake skeleton metastasis ¹ [RADLEX, RID5231, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake skeleton metastasis – Level ^{1 3}	Mild (< mediastinal blood pool) [SNOMED_CT,255604002,DSRCodedEntryValue] High (> liver)[SNOMED_CT,75540009,DSRCodedEntryValue] Moderate[SNOMED_CT,6736007,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,StringValue]
Uptake soft tissue metastasis ¹ [RADLEX, RID5231, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake soft tissue metastasis - Level ^{1 4}	Mild (< mediastinal blood pool) [SNOMED_CT,255604002,DSRCodedEntryValue] High (> liver)[SNOMED_CT,75540009,DSRCodedEntryValue] Moderate[SNOMED_CT,6736007,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,StringValue]
Uptake liver metastasis ¹ [RADLEX, RID5231, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]

¹To be filled if PET/CT is evaluated

PRIMAGE: Clinical variables

²To specify if Uptake primary site is Yes

³To specify if Uptake skeleton metastasis is Yes

⁴To specify if Uptake soft tissue metastasis is Yes

2.3.5. Imaging defined risk factors [IOBC,201006093642984336,VT_Container]

Variable	Labels PRIMAGE platform
Ipsilateral tumor extension within [PrimageNB,47,VT_Container] neck-chest [LOINC,MTHU059898,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]
Ipsilateral tumor extension within chest-abdomen [LOINC,LP264363-5,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]
Ipsilateral tumor extension within abdomen-pelvis [LOINC,LP264569-7,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]
Neck – [FMA,fma7155,VT_Container] Tumor encasing carotid and/or vertebral artery and/or internal jugular vein [PrimageNB,48,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]
Neck – Tumor extending to base of skull [PrimageNB,49,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]

PRIMAGE: Clinical variables

<p>Neck – Tumor compressing the trachea [PrimageNB,50,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Cervico-thoracic junction [PrimageNB,51,VT_Container] – Tumor encasing brachial plexus roots [PrimageNB,52,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Cervico-thoracic junction – Tumor encasing subclavian vessels and/or vertebral and/or carotid artery [PrimageNB,53,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Cervico-thoracic junction – Tumor compressing the trachea [PrimageNB,50,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Thorax [FMA,fma9576,VT_Container] – Tumor encasing the aorta and/or major branches [PrimageNB,54,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Thorax – Tumor compressing the trachea and/or principal bronchi [PrimageNB,55,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Thorax – Lower mediastinal tumor, infiltrating the costo-vertebral junction T9 and T12 [PrimageNB,56,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>

PRIMAGE: Clinical variables

<p>Thoraco-abdominal – Tumor encasing the aorta and/or vena cava [PrimageNB,56,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Abdomen/Pelvis [LOINC,LP264569-7,VT_Code] – Tumor infiltrating the porta hepatis and/or the hepatoduodenal ligament [PrimageNB,57,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Abdomen/Pelvis – Tumor encasing branches of the superior mesenteric artery at the mesenteric root [PrimageNB,58,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Abdomen/Pelvis – Tumor encasing the origin of the celiac axis and/or of the superior mesenteric artery [PrimageNB,59,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Abdomen/Pelvis – Tumor invading one or both renal pedicles [PrimageNB,60,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Abdomen/Pelvis – Tumor encasing the aorta and/or vena cava [PrimageNB,61,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Abdomen/Pelvis – Tumor encasing the iliac vessels [PrimageNB,62,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>

PRIMAGE: Clinical variables

<p>Abdomen/Pelvis – Pelvic tumor crossing the sciatic notch [PrimageNB,63,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Intraspinal [LOINC,360179007,VT_Container] tumor extension whatever the location provided that more than 1/3 of the spinal canal in the axial plane is invaded and/or the perimedullary leptomenigeal spaces are not visible and/or the spinal cord signal is abnormal [PrimageNB,64,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Infiltration of adjacent organs/structures [PrimageNB,65,VT_Container] – Pericardium, diaphragm, kidney, liver, duodenopancreatic block and mesentery [PrimageNB,66,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Conditions to be recorded but not considered IDRFs [PrimageNB,67,VT_Container] – Multifocal primary tumours [PrimageNB,68,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Conditions to be recorded but not considered IDRFs – Pleural effusion [SNOMED_CT,60046008,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Conditions to be recorded but not considered IDRFs – Ascites [SNOMED_CT,389026000,VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>

2.4. Staging [NCIt,C15608,VT_Container]

2.4.1. Primary tumor site[SNOMED_CT,399687005,VT_Container]

Variable	Labels PRIMAGE platform
Abdominal adrenal [SNOMED_CT,277112006,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Abdominal other [SNOMED_CT,74964007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Cervical [SNOMED_CT,261064006,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Pelvic [FMA,fma9578,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Thoracic [SNOMED_CT,261179002,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Unknown [SNOMED_CT,74964007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Others [SNOMED_CT,74964007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Others – Value ¹ [NCIt,C25712,VT_Text]	-

¹Other value should only be specified if Others is Yes

2.4.2. Metastasis[RADLEX,RID5231,VT_Container]

Variable	Labels PRIMAGE platform
----------	-------------------------

PRIMAGE: Clinical variables

Metastasis [RADLEX,RID5231,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Bone ¹ [FMA,fma30317,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Bone marrow ¹ [RADLEX,RID38594,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Liver ¹ [FMA,fma7197,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant lymph nodes ¹ [MESH,D008198,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Skin ¹ [RADLEX,RID34290,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Orbita ¹ [PrimageNB,69,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
CNS ¹ [SNOMED_CT,261731003,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Lungs/Pleura ¹ [RADLEX,RID13437,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Other ¹ [SNOMED_CT,74964007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]

PRIMAGE: Clinical variables

Other - Value ^{1,2} [NCIt,C25712,VT_Text]	-
---	---

¹These variables should only be filled if Metastasis is Yes

²Specify Other value only when Other is Yes

2.4.3. Staging[NCI,C15608,VT_Container]

Variable	Labels PRIMAGE platform
Risk group – Consensus INRG [CCO,CCONT_0000062,VT_Code]	Very low [SNOMED_CT,260362008,DSRCodedEntryValue] Low [SNOMED_CT,62482003,DSRCodedEntryValue] Intermediate [SNOMED_CT,11896004,DSRCodedEntryValue] High [SNOMED_CT,75540009,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Risk group – Database [SNOMED_CT,409720004,VT_Code]	Very low [SNOMED_CT,260362008,DSRCodedEntryValue] Low [SNOMED_CT,62482003,DSRCodedEntryValue] Intermediate [SNOMED_CT,11896004,DSRCodedEntryValue] High [SNOMED_CT,75540009,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
INSS [SNOMED_CT,409720004,VT_Text]	1[PrimageNB,126,StringValue] 2[PrimageNB,127,StringValue] 2A[PrimageNB,128,StringValue] 2B[PrimageNB,129,StringValue] 3[PrimageNB,130,StringValue] 4[PrimageNB,131,StringValue] 4S[PrimageNB,132,StringValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]

PRIMAGE: Clinical variables

INRGSS [NCIt,C133427,VT_Code]	L1[PrimageNB,133,StringValue] L2[PrimageNB,134,StringValue] M[PrimageNB,135,StringValue] MS[PrimageNB,136,StringValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
----------------------------------	--

3. Treatment and evaluation [NCIt,C15839,VT_Container]

3.1. First treatment[SNOMED_CT,708255002,VT_Container]

*Note: In **Blue** are represented variables/values only applicable to Low/Intermediate risk patients and in **Green** are represented variables/values only applicable to High risk patients.*

Variable	Labels PRIMAGE platform
Clinical trial [SNOMED_CT,110465008,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Clinical trial – Name ¹ [NCIt,C42614,VT_Text]	LINES[PrimageNB,137,StringValue] LINESG 1[PrimageNB,138,StringValue] LINESG 2[PrimageNB,139,StringValue] EUNS[PrimageNB,140,StringValue] INES[PrimageNB,141,StringValue] HR-NBL1[PrimageNB,142,StringValue] NB97[PrimageNB,143,StringValue] NB2004[PrimageNB,144,StringValue]
LINES group ^{1,2} [PrimageNB,70,VT_Text]	-

PRIMAGE: Clinical variables

Type of treatment [RADLEX,RID8,VT_Text]	No treatment (observation) [LOINC,LA21213-6,DSRCodedEntryValue] Chemotherapy [NCIt,C15632,DSRCodedEntryValue] Surgery [SNOMED_CT,1003143,DSRCodedEntryValue] Chemotherapy and surgery [PrimageNB,145,StringValue]
Date of surgery ³ [SDO,OperationDate,VT_DateTime]	-
Date of chemotherapy initiation ⁴ [LOINC,LP249274-4,VT_DateTime]	-
End date of chemotherapy ⁴ [PrimageNB,71,VT_DateTime]	-
VP/Carbo ⁴ [IOBC,200906063214672861,VT_Container] VP/Carbo ⁴ [IOBC,200906063214672861,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
VP/Carbo – Number of cycles ⁴⁵ [PrimageNB,72,VT_Text]	-
CADO ⁴ [PrimageNB,73,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
CADO – Number of cycles ⁴⁵ [PrimageNB,72,VT_Text]	-
CO ⁴ [PrimageNB,74,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
CO – Number of cycles ⁴⁵ [PrimageNB,72,VT_Text]	-
N4 ⁴ [PrimageNB,81,VT_Container] N4 ⁴ [PrimageNB,81,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
N4 – Number of cycles ⁴⁵	-

PRIMAGE: Clinical variables

[PrimageNB,72,VT_Text]	
N5 ⁴ [PrimageNB,82,VT_Container] N5 ⁴ [PrimageNB,82,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
N5 – Number of cycles ^{4,5} [PrimageNB,72,VT_Text]	-
N7 ⁴ [PrimageNB,83,VT_Container] N7 ⁴ [PrimageNB,83,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
N6 – Number of cycles ^{4,5} [PrimageNB,72,VT_Text]	-
N7 ⁴ [PrimageNB,84,VT_Container] N7 ⁴ [PrimageNB,84,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
N7 – Number of cycles ^{4,5} [PrimageNB,72,VT_Text]	-
Rapid COJEC ⁴	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Rapid COJEC – Number of cycles ^{4,5} [PrimageNB,72,VT_Text]	-
Modified N7 ⁴ [PrimageNB,85,VT_Container] Modified N7 ⁴ [PrimageNB,85,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Modified N7 – Number of cycles ^{4,5} [PrimageNB,72,VT_Text]	-

PRIMAGE: Clinical variables

Others ⁴ [SNOMED_CT,74964007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Others – Number of cycles ^{4,5} [PrimageNB,72,VT_Text]	-
Other - Name ^{4,6} [SNOMED_CT,703503000,VT_Text]	-

¹Variables to fill only if Clinical trial is Yes

²Specify LINES group only if clinical trial name is LINES

³Specify Date of surgery only if Type of treatment is Surgery or Chemotherapy and surgery

⁴Variables to fill only if Type of treatment is Chemotherapy or Chemotherapy and surgery

⁵Specify the number of cycles only if the value if the specific treatment is Yes

⁶Specify the name of other treatment/s only if Others is Yes

3.2. Treatment evaluation[SWEET,Evaluation,VT_Container]

Variable	Labels PRIMAGE platform
First treatment completely finished ¹ [PrimageNB,75,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Reason for no completion ^{1,2} [NCIt,C25638,VT_Code]	Toxicity [NCIt,C27990,DSRCodedEntryValue] Progression [SNOMED_CT,246450006,DSRCodedEntryValue] [SNOMED_CT,74964007,DSRCodedEntryValue]
Reason for no completion – Other value ^{1,3} [NCIt,C25712,VT_Text]	-
Date of evaluation [NCIt,C164340,VT_Text]	-

¹Variables to fill only if Treatment received is Chemotherapy or Chemotherapy and surgery

²Specify only if First treatment completely finished is No

³Specify only if the Reason for no completion is Other

3.2.1. Imaging[SNOMED_CT,363679005,VT_Container]

PRIMAGE: Clinical variables

Note: *Purple color refers to variables that are automatically filled ones the imaging studies are uploaded to the platform*

3.2.1.1. MRI[RADLEX, RID10813, VT_Container]

Variable	Labels PRIMAGE platform
MRI [RADLEX, RID10813, VT_Code]	Done [SNOMED_CT, 385658003, DSRcodedEntryValue] Not done [SNOMED_CT, 385660001, DSRcodedEntryValue] Not available [LOINC, LA7338-2, DSRcodedEntryValue] Missing [LOINC, LA14698-7, DSRcodedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹ [SNOMED_CT, 410672004, VT_DateTime]	-

¹To be filled if MRI is evaluated

3.2.1.2. CT[RADLEX, RID10321, VT_Container]

Variable	Labels PRIMAGE platform
CT [RADLEX, RID10321, VT_Code]	Done [SNOMED_CT, 385658003, DSRcodedEntryValue] Not done [SNOMED_CT, 385660001, DSRcodedEntryValue] Not available [LOINC, LA7338-2, DSRcodedEntryValue] Missing [LOINC, LA14698-7, DSRcodedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹	-

PRIMAGE: Clinical variables

[SNOMED_CT, 410672004, VT_DateTime]

¹To be filled if CT is evaluated

3.2.1.3. MIBG entire body [RADLEX, RID45807, VT_Container]

Variable	Labels PRIMAGE platform
MIBG entire body [RADLEX, RID45807, VT_Container]	Done [SNOMED_CT, 385658003, DSRcodedEntryValue] Not done [SNOMED_CT, 385660001, DSRcodedEntryValue] Not available [LOINC, LA7338-2, DSRcodedEntryValue] Missing [LOINC, LA14698-7, DSRcodedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹ [SNOMED_CT, 410672004, VT_DateTime]	-
Technique ¹ [SNOMED_CT, 272394005, VT_Text]	SPECT [MEDDRA,10041458,StringValue] Planar [EPO,Planar,StringValue] SPECT and planar [PrimageNB,125,StringValue] Unknown[SNOMED_CT,261665006,StringValue]
Dose unit ¹ [SNOMED_CT, 408102007, VT_Text]	mCi [SNOMED_CT,261665006,StringValue] MBq [SNOMED_CT,229034000,StringValue] Unknown[SNOMED_CT,261665006,StringValue]
Dose value ^{1,2} [OBI, OBI_0002491, VT_Text]	-
Image acquisition ¹ [NCIt, C43385, VT_Text]	4h 24h Other

PRIMAGE: Clinical variables

	[SNOMED_CT,74964007,VT_Text] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Time after injection in h ¹³ [DCM, 130194, VT_Text]	
Uptake [NCIt, C71688, VT_Container] primary site ¹ [LOINC, MTHU010322, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake skeleton ¹ [FMA, fma303630, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake soft tissue ¹ [NCIt, C12471, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake liver ¹ [FMA, fma7197, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake skin ¹ [RADLEX, RID34290, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Score SIOPEN ¹ [PrimageNB, 46, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Score SIOPEN - Value ¹⁴ [SNOMED_CT, 394617004, VT_Code]	-

¹To be filled if MIBG entire body is evaluated

²Dose value has to be filled if Dose unit is not Unknown

³Time after injection in h has to be filled if Image acquisition includes Other

⁴Score SIOPEN value has to be filled if Score SIOPEN is Yes

3.2.1.4. PET/CT[RADLEX,RID10341,VT_Container]

Variable	Labels PRIMAGE platform
PET/CT [RADLEX,RID10341,VT_Code]	Done [SNOMED_CT, 385658003, DSRCodedEntryValue] Not done [SNOMED_CT, 385660001, DSRCodedEntryValue] Not available [LOINC, LA7338-2, DSRCodedEntryValue] Missing [LOINC, LA14698-7, DSRCodedEntryValue]
Manufacturer and model ¹ [SNOMED_CT, 706821004, VT_Text]	-
Date ¹ [SNOMED_CT, 410672004, VT_DateTime]	-
FDG-18F (administered dose) ¹ [SNOMED_CT, 35321007, VT_Text]	-
Uptake [NCIt, C71688, VT_Container] primary site ¹ [LOINC, MTHU010322, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]
Uptake primary site - Level ^{1 2} [SNOMED_CT, 276625007, VT_Code]	Mild (< mediastinal blood pool) [SNOMED_CT,255604002,DSRCodedEntryValue] High (> liver)[SNOMED_CT,75540009,DSRCodedEntryValue] Moderate[SNOMED_CT,6736007,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,StringValue]
Uptake skeleton metastasis ¹ [RADLEX, RID5231, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]

<p>Uptake skeleton metastasis – Level^{1 3}</p>	<p>Mild (< mediastinal blood pool) [SNOMED_CT,255604002,DSRCodedEntryValue] High (> liver)[SNOMED_CT,75540009,DSRCodedEntryValue] Moderate[SNOMED_CT,6736007,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,StringValue]</p>
<p>Uptake soft tissue metastasis¹ [RADLEX, RID5231, VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]</p>
<p>Uptake soft tissue metastasis - Level^{1 4}</p>	<p>Mild (< mediastinal blood pool) [SNOMED_CT,255604002,DSRCodedEntryValue] High (> liver)[SNOMED_CT,75540009,DSRCodedEntryValue] Moderate[SNOMED_CT,6736007,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,StringValue]</p>
<p>Uptake liver metastasis¹ [RADLEX, RID5231, VT_Code]</p>	<p>Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Not evaluable [RADLEX,RID39225,DSRCodedEntryValue]</p>

¹To be filled if PET/CT is evaluated

²To specify if Uptake primary site is Yes

³To specify if Uptake skeleton metastasis is Yes

⁴To specify if Uptake soft tissue metastasis is Yes

3.2.2. INRC (International Neuroblastoma Response Criteria) [NCIt,C102869,VT_Container]

Note: Variables included in this section depend on the patient risk group

3.2.2.1. Low/Intermediate risk patients [PrimageNB,76,VT_Container]

PRIMAGE: Clinical variables

Variable	Labels PRIMAGE platform
Primary tumor site [SNOMED_CT,399687005,VT_Container] Primary tumor site [SNOMED_CT,399687005,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Primary tumor site – Response ¹ [NCIt,C25755,VT_Text]	Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Metastasis [RADLEX,RID5231,VT_Container] response [NCIt,C25755,VT_Text]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]
Metastasis response – Bone ² [FMA,C25755,VT_Text]	Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]

PRIMAGE: Clinical variables

<p>Metastasis response – Lymph nodes² [MESH,C25755,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>
<p>Metastasis response – Bone marrow² [RADLEX,RID38594,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>
<p>Metastasis response – Liver² [FMA,fma7197,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>

PRIMAGE: Clinical variables

<p>Metastasis response – Skin² [RADLEX,fma7197,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>
<p>Metastasis response – CNS² [IOBC,200906047768068685,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>
<p>Metastasis response – Pleura/Lung² [PrimageNB,77,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>

PRIMAGE: Clinical variables

<p style="text-align: center;">Other² [SNOMED_CT,74964007,VT_Text]</p>	<p style="text-align: center;">Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>
<p style="text-align: center;">Conclusion metastatic response² [PrimageNB,146,VT_Text]</p>	<p style="text-align: center;">mCR [PrimageNB,147,StringValue] mPR [PrimageNB,148,StringValue] mSD [PrimageNB,149,StringValue] mPD [PrimageNB,150,StringValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]</p>
<p style="text-align: center;">Overall response [PrimageNB,141,VT_Text]</p>	<p style="text-align: center;">Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue]</p>

¹Specify Primary tumor site response only if Primary tumor site is evaluated

²Specify all these variables only if Metastasis response is Evaluated

3.2.2.2. High risk patients

Variable	Labels PRIMAGE platform
<p style="text-align: center;">Primary tumor site [SNOMED_CT,399687005,VT_Container] Primary tumor site</p>	<p style="text-align: center;">Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]</p>

PRIMAGE: Clinical variables

[SNOMED_CT,399687005,VT_Code]	
<p>Primary tumor site – Response¹ [NCIt,C25755,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]</p>
<p>Metastasis [RADLEX,RID5231,VT_Container] response [NCIt,C25755,VT_Text]</p>	<p>Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Metastasis response – Bone² [FMA,C25755,VT_Text]</p>	<p>Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]</p>
<p>Metastasis response – Bone – SIOPEN^{2 3} [PrimageNB,151,VT_Code]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue] Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]</p>
<p>Metastasis response – Bone – PET/CT^{2 3} [RADLEX,RID10341,VT_Code]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]</p>

PRIMAGE: Clinical variables

	Not applicable [SNOMED_CT,385432009,DSRCodedEntryValue]
Metastasis response – Bone marrow ² [RADLEX,RID38594,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Metastasis response – Bone marrow – Apirate ^{2,4} [MEDdra,10005982,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Metastasis response – Bone marrow – Apirate – Result ^{2,4,5} [SNOMED_CT,394617004,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Metastasis response – Bone marrow – Trepine biopsy ^{2,4} [SNOMED_CT,50590007,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Metastasis response – Bone marrow – Trepine biopsy – Result ^{2,4,6} [SNOMED_CT,394617004,VT_Code]	Positive [SNOMED_CT,10828004,DSRCodedEntryValue] Negative [SNOMED_CT,373067005,DSRCodedEntryValue]
Metastasis response – Liver ² [FMA,fma7197,VT_Text]	Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue] Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]
Metastasis response – Other metastatic sites ² [PrimageNB,152,VT_Text]	Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue] New involvement[SNOMED_CT,278112009,StringValue]

PRIMAGE: Clinical variables

	<p>Evaluated [NCIt,C25214,StringValue] Not evaluated [RADLEX,RID39225,StringValue] Not applicable [SNOMED_CT,385432009,StringValue]</p>
<p>Conclusion metastatic response² [PrmageNB,146,VT_Text]</p>	<p>mCR [PrmageNB,147,StringValue] mPR [PrmageNB,148,StringValue] mSD [PrmageNB,149,StringValue] mPD [PrmageNB,150,StringValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]</p>
<p>Overall response [PrmageNB,146,VT_Text]</p>	<p>Complete response [LOINC,LA28366,StringValue] Partial response[LOINC,LA28369-9,StringValue] Stable disease[NCIt,C18213,StringValue] Progression disease[MEDDRA,10061818,StringValue]</p>

¹Specify Primary tumor site response only if Primary tumor site is evaluated

²Specify all these variables only if Metastasis response is Evaluated

³Specify only if Metastasis response – Bone is Evaluated

⁴Specify only if Metastasis response – Bone marrow is Evaluated

⁵Specify only if Metastasis response – Bone marrow – Aspirate is Evaluated

⁶ Specify only if Metastasis response – Bone marrow – Trepine biopsy is Evaluated

3.2.3. Minimal residual disease[MEDdra,10079987,VT_Container]

Variable	Labels PRIMAGE platform
<p>Bone marrow [RADLEX,RID38594,VT_Code]</p>	<p>Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]</p>
<p>Bone marrow – Method¹ [SNOMED_CT,260686004,VT_Code]</p>	<p>qRT-PCR [IOBC,201606016242659673,DSRCodedEntryValue] Immunocytology [PrmageNB,17,DSRCodedEntryValue] qRT-PCR and Immunocytology [PrmageNB,18,DSRCodedEntryValue]</p>

PRIMAGE: Clinical variables

Bone marrow – Response ¹ [RADLEX,RID38594,VT_Code]	Complete[SNOMED_CT,255594003,DSRCodedEntryValue] Not complete[LOINC,LA4699-0,DSRCodedEntryValue] Inconclusive [SNOMED_CT,419984006,DSRCodedEntryValue]
Peripheral blood [PrimageNB,78,VT_Code]	Evaluated [NCIt,C25214,DSRCodedEntryValue] Not evaluated [RADLEX,RID39225,DSRCodedEntryValue]
Peripheral blood – Method ² [SNOMED_CT,260686004,VT_Code]	qRT-PCR [IOBC,201606016242659673,DSRCodedEntryValue]
Peripheral blood – Response ² [RADLEX,RID38594,VT_Code]	Complete[SNOMED_CT,255594003,DSRCodedEntryValue] Not complete[LOINC,LA4699-0,DSRCodedEntryValue] Inconclusive [SNOMED_CT,419984006,DSRCodedEntryValue]

¹Specify Bone marrow response only when Bone marrow is Evaluated

²Specify Peripheral blood response only when peripheral blood is Evaluated

4. Follow-up[NCIt,C16033,VT_Container]

Note: In **Green** are represented variables that only apply to High risk patients. In **Orange** are represented variables that only apply to Very low risk patients

Variable	Labels PRIMAGE platform
TVD ⁰ [PrimageNB,81,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Poor responder ⁰ [PrimageNB,80,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Date of end of treatment (Total) [LOINC,MTHU060967,VT_DateTime]	-
Progression [SNOMED_CT,246450006,VT_Container]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]

PRIMAGE: Clinical variables

Progression [SNOMED_CT,246450006,VT_Code]	
Date of first progression ¹ [SNOMED_CT,410672004,VT_DateTime]	-
Type of progression ¹ [SNOMED_CT,410656007,VT_Code]	Local [SNOMED_CT,255470001,DSRCodedEntryValue] Distal [SNOMED_CT,46053002,DSRCodedEntryValue] Combined [SNOMED_CT,89780004,DSRCodedEntryValue]
Distant site - Skeleton ^{1,2} [SNOMED_CT,410656007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Bone marrow ^{1,2} [NCIt,C12431,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Lymph nodes ^{1,2} [MESH,Lymph nodes,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – CNS metastasis ^{1,2} [NCIt,C4015,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Lung metastasis ^{1,2} [OCHV,C0153676,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site - Unknown ^{1,2} [NCIt,C17998,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site - Others ^{1,2} [SNOMED_CT,74964007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Others - Value ^{1,2,3}	-

PRIMAGE: Clinical variables

[OGMS, OGMS_0000105, VT_Code]	
Relapse [OGMS,OGMS_0000105,VT_Container] Relapse [OGMS,OGMS_0000105,VT_Text]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Date of first relapse ⁴ [SNOMED_CT,410672004,VT_DateTime]	-
Type of relapse ⁴ [SNOMED_CT,410656007,VT_DateTime]	Local [SNOMED_CT,255470001,DSRCodedEntryValue] Distal [SNOMED_CT,46053002,DSRCodedEntryValue] Combined [SNOMED_CT,89780004,DSRCodedEntryValue]
Distant site - Skeleton ^{4,5} [SNOMED_CT,410656007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Bone marrow ^{4,5} [NCIt,C12431,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Lymph nodes ^{4,5} [MESH,Lymph nodes,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – CNS metastasis ^{4,5} [NCIt,C4015,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Lung metastasis ^{4,5} [OCHV,C0153676,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site - Unknown ^{4,5} [NCIt,C17998,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]

PRIMAGE: Clinical variables

Distant site - Others ^{4,5} [SNOMED_CT,74964007,VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue] Unknown[SNOMED_CT,261665006,DSRCodedEntryValue]
Distant site – Others - Value ^{4,5,6} [OGMS, OGMS_0000105, VT_Code]	-
Second malignancy ⁷ [MEDdra, 10039801, VT_Container] Second malignancy ⁷ [MEDdra, 10039801, VT_Code]	Yes [SNOMED_CT,373066001,DSRCodedEntryValue] No [SNOMED_CT,373067005,DSRCodedEntryValue]
Second malignancy – Diagnosis date ^{7,8} [SNOMED_CT, 373066001, VT_DateTime]	-
Second malignancy – Type of tumor ^{7,8} [LOINC, LP207876-6, VT_Text]	-
Regression ⁹ [NCIT, C18243, VT_Text]	
Current status [NCIt, C115767, VT_Container] State [SNOMED_CT, 398070004, VT_Text]	Alive [NCIt,C37987,StringValue] Dead [NCIt,C28554,StringValue]
Date of last visit ¹⁰ [PrimageNB, 79, VT_DateTime]	-
Date of death ¹¹ [SNOMED_CT, 399753006, VT_DateTime]	-
Cause of death ¹¹ [SNOMED_CT, 184305005, VT_Text]	Diseases [PCAO,Diseases,StringValue] Toxicity [NCIt,C27990,StringValue] Others[SNOMED_CT,74964007,VT_Code]

⁰Do not apply if risk group is Very low

¹Specify only if Progression is Yes

²Specify if Type of progression is Distal or Combined

PRIMAGE: Clinical variables

³*Specify if Distant – Others is Yes*

⁴*Specify only if Relapse is Yes*

⁵*Specify if Type of relapse is Distal or Combined*

⁶*Specify if Distant – Others is Yes*

⁷*Only if High risk patient*

⁸*Specify if Second malignancy is Yes*

⁹*Only if Type of treatment is No treatment (observation)*

¹⁰*Only if Current status is Alive*

¹¹*Only if Current status is Dead*