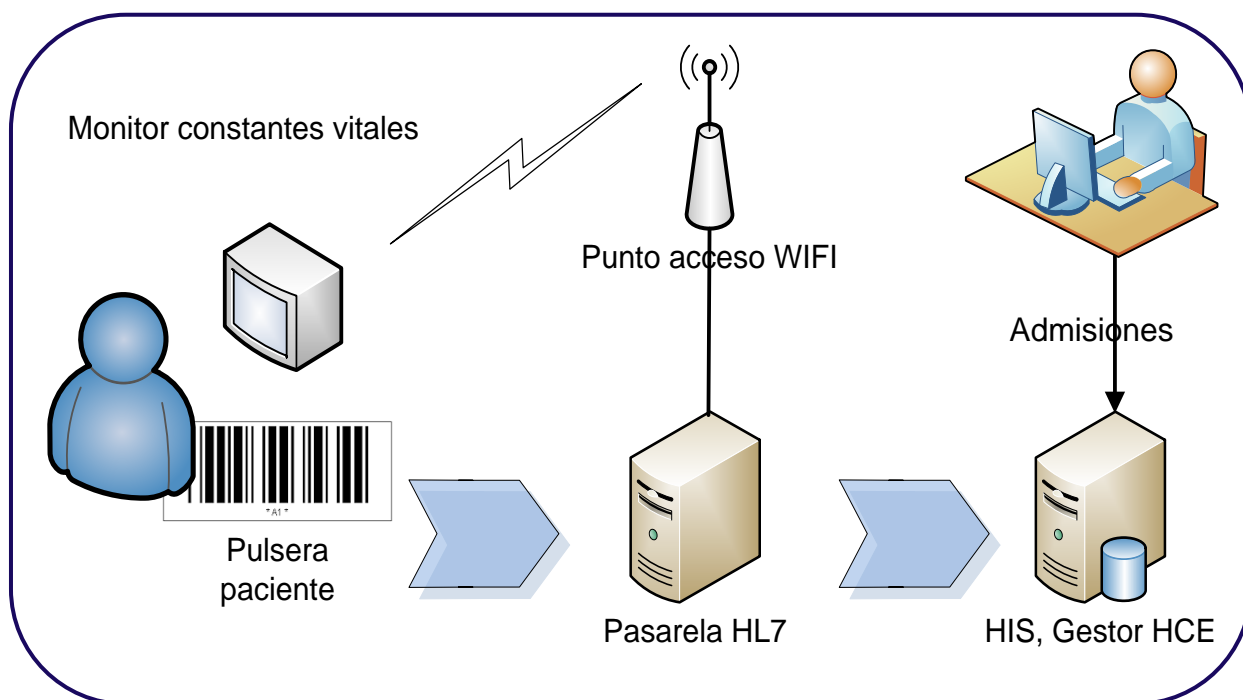




Dispositivos médicos e interoperabilidad “Open Source”



Alumno: Diego Lorca Moreno

Plan de Estudios: Grado de Ingeniería Informática

Área del TFG: Health IT

Consultor: Carlos Luis Sánchez Bocanegra

Profesor: José Antonio Morán Moreno

Fecha Entrega: Enero 2017



Esta obra está sujeta a una licencia de Reconocimiento-
NoComercial-SinObraDerivada [3.0 España de Creative
Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

Abstract

The EMR gathers patient's health data and integrates diverse information, included vital signs data.

Vital signs are captured constantly in a daily basis during patient's hospital stay. This data is usually documented manually and, therefore, prone to errors of transcription, omission and right patient's assignment.

Automatic sending of vital signs to the EMR improves efficiency, quality and patient's security. However, the limited connectivity and interoperability of medical equipment, due to a poor standards adoption by these devices, the wide spread of proprietary systems and a high implementation cost and integration, prevent its growth.

This Bachelor's Degree Final Project (TFG) develops a proposal based on ISO/IEEE11073, HL7 and IHE profiles with Rosetta Terminology standards, running on top of an "Open Source" software layer so that both cost and manufactures' proprietary solutions dependency can be highly reduced.

The design and implementation of a test environment with Oracle VirtualBox, Mirth Connect and OpenEMR reveals a working solution that fits the automatic vital signs' recordings. This contribution tries to increase the knowledge's base and promotes the adoption of similar solutions and aligned developments for other medical devices integrations with the EMR.

Key words: Medical device, interoperability, EMR, ISO/IEEE11073, HL7, IHE, Rosetta Terminology

Abstract

La HCE es un compendio de datos relacionados con la salud del paciente que integra información, incluida la relativa a sus constantes vitales.

La toma de constantes, realizada rutinariamente durante la estancia hospitalaria de los pacientes, suele documentarse manualmente y estar sometida a la aparición de errores de transcripción, omisión e incorrecta imputación a pacientes.

El envío automático de las constantes a la HCE mejora la eficiencia, la calidad y la seguridad. Sin embargo, la limitada conectividad e interoperabilidad de los equipos médicos debido a la poca adopción de estándares a nivel dispositivo, la proliferación de sistemas propietarios y su alto coste de implementación e integración dificultan su expansión.

Este TFG desarrolla una propuesta que usa los estándares ISO/IEEE11073, HL7 y perfiles IHE con Rosetta Terminology sobre software “Open Source” para reducir el coste y la dependencia asociadas a las soluciones propietarias.

El diseño e implementación en un entorno de pruebas con VirtualBox, Mirth Connect y OpenEMR revelan una solución operativa que da respuesta al registro automatizado. Esta aportación pretende ampliar la base del conocimiento e impulsar la adopción de soluciones que promuevan el desarrollo de propuestas similares, alineadas con la integración de otros dispositivos médicos con la HCE.

Palabras Clave: Dispositivo médico, interoperabilidad, HCE, ISO/IEEE11073, HL7, IHE, Rosetta Terminology

Resumen

La Historia Clínica Electrónica (HCE) es una realidad en todas las organizaciones que proveen Servicios de Salud a sus pacientes. El desarrollo y su implantación se están extendiendo para configurar la denominada HCE única o unificada a través de la integración sanitaria entre los diferentes proveedores, desde la atención ambulatoria y hospitalaria hasta la integración en un territorio, con vistas a un objetivo de unificación nacional, incluso europeo. Existen múltiples planes estratégicos e inversiones en tecnologías TIC enfocados a la consecución del objetivo.

En este entorno, la integración de las pruebas de diagnóstico, como las imágenes (Rx, TAC, ecografías...), las de laboratorio y los diferentes tipos de informes asociados al paciente son ya una realidad. Sin embargo, la integración automatizada de muchos dispositivos médicos, especialmente en el entorno de los hospitales donde hay instalados cientos de equipos, dista mucho y está pendiente de implementarse y alcanzar el nivel que las tecnologías anteriores.

El TFG se enmarca en este último escenario, el de la conectividad de los equipos médicos a la HCE y, concretamente, en la categoría de los monitores de constantes vitales discretas.

La metodología ágil usada, SCRUM, ha permitido avanzar en la realización y finalización del proyecto mediante la creación de productos y objetivos intermedios (*planificación, análisis del estado de situación, diseño y creación del modelo de pruebas en varias etapas*) en un proceso iterativo de ciclos de corta duración.

La revisión sistemática de la bibliografía disponible, así como de la situación actual en términos de estándares y recomendaciones de conectividad, de integración y de las propuestas existentes por parte de fabricantes, consultores y otros actores del mercado, ha definido el “estado del arte”, punto de partida para abordar el diseño de la solución adaptada a un caso de uso muy extendido en los servicios de urgencias y salas de hospitalización, la medición de las constantes vitales básicas de los

pacientes: frecuencia cardíaca (FC), presión arterial no invasiva (PANI), saturación de oxígeno (SpO2) y temperatura (Temp).

El diseño de la solución, que integra automáticamente en la HCE las constantes del paciente, ha permitido garantizar la construcción de un entorno de pruebas fácilmente reproducible con el uso de software “Open Source” que reduce al máximo el coste de licencias privativas. El uso de los estándares más actuales, los perfiles IHE, simplifican la interoperabilidad y la integración de los monitores de vitales y reducen, nuevamente, los costes derivados de las integraciones y la consultoría que siempre acompañan a las soluciones propietarias.

El diseño da paso a la creación del modelo-maqueta que integra los datos procedentes del monitor de constantes vitales y, a través del motor de integración Mirth Connect, configura el canal o interface que permite procesar las constantes del paciente y las integra en su HCE, gestionada por OpenEMR.

El desarrollo de la interface monitor paciente>> sistema OpenEMR se realiza con éxito y consigue alcanzar los objetivos marcados por el TFG.

En conclusión, la integración de los dispositivos médicos con la HCE mediante el uso de software “Open Source” y de los estándares ISO/IEEE11073, HL7 y perfiles IHE, hacen factible la creación de soluciones específicas, coste-eficientes, adaptadas a diferentes casos de uso o flujos de trabajo asistenciales dentro del hospital.

Nota: Este trabajo está destinado a cualquier persona y/o profesional cuyo marco laboral se encuentre en el ámbito del sector salud-hospital y, por tanto, tenga un conocimiento moderado y esté familiarizado con los conceptos y la terminología propia del entorno. Concretamente, y dado su enfoque técnico, los departamentos mejor posicionados se encuentran en los Servicios de Mantenimiento, Electromedicina e Informática de las organizaciones sanitarias.

Dedicatoria y agradecimientos

A mi mujer, Alicia, por su apoyo durante todo el tiempo
que he dedicado a la realización de este grado de
Ingeniería Informática.

Índice de contenidos

1.	INTRODUCCIÓN	1
1.1	Motivación, contexto y justificación del Trabajo	1
1.2	Objetivos del Trabajo	3
1.3	Enfoque y metodología	4
1.4	Planificación del Trabajo	5
1.5	Breve resumen de los productos obtenidos.....	5
1.6	Breve descripción de los capítulos de la memoria.....	6
2.	ESTADO DEL ARTE	7
2.1	La realidad de los dispositivos médicos	7
2.1.1	Conectividad y salida de datos	8
2.1.2	Protocolos de comunicación.....	9
2.2	La realidad en los hospitales	11
2.3	Las soluciones disponibles.....	12
2.3.1	Integración mediante sistema clínico departamental	13
2.3.2	Integración mediante proveedor de conectividad empresarial.....	14
2.3.3	Integración mediante pasarelas de dispositivos.....	14
2.4	La búsqueda de la solución.....	16
2.4.1	El uso de estándares y convenciones	16
2.4.2	HL7 – Health Level Seven	17
2.4.3	Estándar ISO/IEEE11073.....	19
2.4.4	IHE - Integrating the Healthcare Enterprise.....	22
3.	DISEÑO	26
3.1	Caso de uso del registro de constantes vitales.....	26
3.2	Esquema y estructura de la solución tecnológica	27
3.3	La elección del software	29
3.3.1	Mirth Connect	31
3.3.2	OpenEMR	32
3.3.3	Software de soporte.....	33
4.	IMPLEMENTACIÓN	35
4.1	Fases y actividades principales	35
4.2	El laboratorio, entorno de trabajo	36
4.2.1	Instalación máquinas virtuales y conectividad.....	36
4.2.2	Instalación de Mirth Connect	36
4.2.3	Instalación de OpenEMR.....	39
4.3	Mirth Connect, primer contacto	43
4.3.1	Configuración del conector de entrada de la interface.....	43
4.3.2	Configuración del conector de salida de la interface.....	46
4.3.3	Mediciones constantes vitales y envío a OpenEMR	48
4.4	OpenEMR y tablas BBDD implicadas en el registro	51
5.	CONCLUSIONES	56
6.	BIBLIOGRAFÍA	58
7.	ANEXOS	59

Índice de figuras

<i>Figura 1.</i> Metodología SCRUM y sprints del TFG.....	4
<i>Figura 2.</i> Planificación del TFG	5
<i>Figura 3.</i> Dispositivos médicos.....	7
<i>Figura 4.</i> Monitores multiparamétricos de paciente	8
<i>Figura 5.</i> Ventilador Maquet Servo-i.....	10
<i>Figura 6.</i> Esquema de la red de monitorización Datex-Ohmeda S/5.....	10
<i>Figura 7.</i> Gráfica de cuidados en UCI registro manual	12
<i>Figura 8.</i> Esquema integración mediante sistema información clínico	13
<i>Figura 9.</i> Gráfica de cuidados en UCI registro automático	13
<i>Figura 10.</i> Esquema integración mediante capa intermedia de conectividad	14
<i>Figura 11.</i> Esquema integración mediante pasarelas de dispositivos	14
<i>Figura 12.</i> Esquema básico transacción HL7	17
<i>Figura 13.</i> Ejemplo mensaje HL7	18
<i>Figura 14.</i> Capas modelo OSI y estándares X73.....	19
<i>Figura 15.</i> Esquema UML de clases del estándar X73.....	20
<i>Figura 16.</i> Ejemplo esquema UML de monitor constantes vitales.....	21
<i>Figura 17.</i> Proceso para la creación de un perfil IHE	22
<i>Figura 18.</i> Dominios IHE y perfiles.....	23
<i>Figura 19.</i> Perfil DEC y caso de uso	24
<i>Figura 20.</i> Esquema caso de uso registro constantes vitales.....	26
<i>Figura 21.</i> Componentes y virtualización del entorno de pruebas.....	27
<i>Figura 22.</i> Esquema de conectividad y red del entorno de pruebas.....	28
<i>Figura 23.</i> Canal-interface Mirth Connect y sus componentes	31
<i>Figura 24.</i> Esquema envío mensaje HL7 a BBDD, PDF y correo electrónico.....	31
<i>Figura 25.</i> Visualización resumen HCE paciente.....	32
<i>Figura 26.</i> Formulario de constantes vitales	33
<i>Figura 27.</i> Simulador de mensajes HL7 Inspector	33
<i>Figura 28.</i> Cronograma implantación.....	35
<i>Figura 29.</i> Máquinas virtuales en funcionamiento.....	36
<i>Figura 30.</i> Descarga e instalación de Mirth Connect para Linux	37
<i>Figura 31.</i> Administrador del servidor Mirth Connect.....	37
<i>Figura 32.</i> Puertos, base de datos, conexión y credenciales	37
<i>Figura 33.</i> Pantalla acceso local al administrador de Mirth Connect.....	38
<i>Figura 34.</i> Formulario registro administrador Mirth Connect	38
<i>Figura 35.</i> Administrador “Diego LM” conectado a Mirth Connect.....	38
<i>Figura 36.</i> Pantalla acceso remoto al administrador de Mirth Connect	39
<i>Figura 37.</i> Instalación de XAMPP	39
<i>Figura 38.</i> XAMPP en funcionamiento	40

<i>Figura 39.</i> Instalación de OpenEMR	40
<i>Figura 40.</i> Finalización instalación y acceso OpenEMR	40
<i>Figura 41.</i> Pantalla de OpenEMR	41
<i>Figura 42.</i> Pantalla de MySQL Workbench	41
<i>Figura 43.</i> Captura de la tabla BBDD del paciente en OpenEMR	41
<i>Figura 44.</i> Edición fichero “mysqld.cnf”	42
<i>Figura 45.</i> Conexión remota a MySQL.....	42
<i>Figura 46.</i> Usuario conectado a BBDD de OpenEMR	43
<i>Figura 47.</i> Asistente configuración canales de Mirth Connect	44
<i>Figura 48.</i> Estructura en árbol mensaje HL7 asistente Mirth Connect	44
<i>Figura 49.</i> Configuración filtro conector entrada canal Mirth Connect	45
<i>Figura 50.</i> Configuración transformador canal Mirth Connect	45
<i>Figura 51.</i> JavaScript transformador	46
<i>Figura 52.</i> Configuración conector salida canal Mirth Connect	46
<i>Figura 53.</i> Sentencia SQL conector salida canal Mirth Connect	47
<i>Figura 54.</i> Despliegue canal Mirth Connect	47
<i>Figura 55.</i> OpenEMR – alta de pacientes	48
<i>Figura 56.</i> OpenEMR - Registro manual constantes vitales paciente “Diego”	48
<i>Figura 57.</i> OpenEMR - Registro constantes en tendencias tabulares de “Diego”	48
<i>Figura 58.</i> Identificación paciente en monitor constantes vitales.....	49
<i>Figura 59.</i> Toma de constantes vitales en el monitor	49
<i>Figura 60.</i> Almacenamiento mediciones, validación y envío a la HCE	49
<i>Figura 61.</i> Captura mensaje HL7 en Mirth Connect.....	50
<i>Figura 62.</i> Segmentos y nomenclatura ISO/IEEE del mensaje HL7	50
<i>Figura 63.</i> Envío mensaje HL7 y escritura en OpenEMR	51
<i>Figura 64.</i> Script SQL conector de salida de Mirth Connect	52
<i>Figura 65.</i> Registro constantes en tablas OpenEMR paciente “Diego”	52
<i>Figura 66.</i> OpenEMR - Sin registro manual constantes vitales paciente “Laura”	53
<i>Figura 67.</i> OpenEMR - Registro constantes en tendencias tabulares de “Laura”	53
<i>Figura 68.</i> Pacientes admitidos	54
<i>Figura 69.</i> Registros de constantes vitales pacientes	54
<i>Figura 70.</i> Registros de formularios manuales y automáticos	55

Índice de tablas

<i>Tabla 1.</i> Estándares de conectividad y convenciones	16
<i>Tabla 2.</i> Ejemplo nomenclatura ISO/IEEE11073 de constantes vitales	21
<i>Tabla 3.</i> Ejemplo terminología Rosetta	24

Glosario

EMR: Electronic Medical Record; ver HCE.

HCE: La Historia Clínica Electrónica es un compendio de la información más importante y representativa relacionada con la salud del paciente, configurada por informes, resultados de pruebas, visitas realizadas, etc.

Dispositivo médico: Cualquier equipo usado en el entorno del paciente para el diagnóstico y/o el tratamiento de patologías con capacidad de recibir y/o enviar datos en formato electrónico a otros equipos o sistemas externos.

X73: Abreviación de la familia de estándares ISO/IEEE 11073 para la interoperabilidad de dispositivos médicos en diferentes contextos clínicos.

HL7: *Health Level Seven* engloba un conjunto de estándares que normalizan el intercambio de información entre diferentes actores y sistemas de información del sector salud.

IHE: *International Health Enterprise* es una iniciativa que promueve las integraciones entre dispositivos médicos y sistemas de información a través del desarrollo de propuestas concretas, denominadas “perfiles de integración”, que se adaptan a casos de uso y/o flujos de trabajo de diferentes situaciones clínicas.

Interoperabilidad semántica: Concepto que tiene sentido en el intercambio de mensajes entre dispositivos médicos y/o sistemas de información para poder interpretar el mismo significado de la nomenclatura y/o la terminología usada dentro de un contexto concreto.

Rosetta Terminology Mapping: Propuesta que define, armoniza y acota el contenido semántico de la nomenclatura definida en ISO/IEEE 11073, usada por diferentes fabricantes para los mensajes del perfil IHE PCD.

1. Introducción

1.1 Motivación, contexto y justificación del Trabajo

En la sociedad actual y, en el caso de los sistemas de atención sanitaria, la tendencia dibuja una situación en la que la incorporación de los TIC facilitan la gestión de todos los procesos asistenciales, su registro, almacenamiento, acceso y el compartir información de los datos asociados a la HCE de los pacientes, tanto entre proveedores (servicios de salud, hospitales) como cada día más, los propios pacientes.

El aumento de la esperanza de vida y el tratamiento de los pacientes con múltiples patologías, la falta y/o racionalización de los recursos y la demanda de una mayor transparencia, seguridad y mejora de los cuidados, empujan a la protocolización de los procesos y al registro de la actividad asociados a los mismos.

La inversión realizada en este entorno ha sido importante por parte de todos los actores implicados, tanto los de titularidad pública como de la privada. La inversión se realiza en tecnologías TIC para la gestión de la HCE de los pacientes, la integración de agendas, visitas, informes médicos, pruebas clínicas relacionada con analíticas, almacenamiento y gestión de imágenes generadas en las exploraciones de rayos-x, resonancia magnética, etc.

La consecución de la HCE única del paciente dentro de su comunidad queda lejos de conseguirse debido a múltiples factores. Por un lado, la integración a nivel de proveedores de servicios sanitarios está limitada por su diferente titularidad, la heterogeneidad de las plataformas tecnológicas desplegadas, la variedad de propuestas de estándares de información y la multitud de proveedores tecnológicos y de consultoría, entre otras causas. En el nivel organizacional existen políticas definidas y planes estratégicos para caminar hacia una HCE compartida a través de la integración de los diferentes sistemas de información dentro de este entorno complejo. No obstante, los dispositivos médicos que están directamente conectados a los pacientes y su integración automatizada con los diferentes procesos asistenciales siguen siendo un reto con múltiples aristas. La diversidad de dispositivos (monitores de paciente, bombas de fármacos, ventiladores, etc.), la falta de estandarización del acceso y de la salida de datos normalizados que faciliten una integración eficaz y eficiente en términos de esfuerzo económico, tecnológico, de implementación, interoperabilidad, mantenimiento y evolución durante el ciclo de vida de los dispositivos, son barreras que limitan su implantación.

Organizaciones y agencias de estandarización trabajan para definir, publicar y promover diferentes estándares que faciliten la interoperabilidad de los dispositivos médicos a nivel de mensajería, de nomenclatura, de la sintaxis y de la semántica

asociada para facilitar la integración entre dispositivos y sistemas de información con una relación coste-beneficio óptima.

Esta situación compleja, la dificultad de su dominio y, especialmente las barreras en la implantación de la conectividad e integración de los dispositivos médicos con la HCE, resumen las razones principales que motivan el análisis de posibles soluciones y la implementación de un caso de uso muy común en las organizaciones sanitarias, la toma de constantes vitales básicas. La idea es clarificar el panorama y promover la adopción dentro del sector, especialmente en hospitales y clínicas con recursos IT más reducidos [1], [2].

En este último escenario de la conectividad y la interoperabilidad a nivel de dispositivos médicos es donde se centra el TFG cuyo objetivo principal es la propuesta de un modelo replicable de un caso de uso asistencial, basado en las propuestas actuales de los estándares aplicables y la disponibilidad de software "Open Source". La integración de un monitor de presión arterial no invasiva a un sistema gestor de HCE mediante una interface normalizada y de código abierto desarrollan la propuesta del TFG.

1.2 Objetivos del Trabajo

El desarrollo del estudio persigue la consecución de unos objetivos principales y otros que se puede clasificar como secundarios.

Por un lado, se persigue obtener los siguientes objetivos principales:

- Facilitar la conexión e integración automática de las constantes vitales de los pacientes en el entorno clínico donde se aplica el caso de uso relacionado con la medición de signos vitales del paciente
- Mejorar la recogida de datos clínicos, su almacenamiento e integración con la HCE para aumentar la eficiencia documental, de los flujos de trabajo asistencial, de la mejora de la seguridad y los cuidados del paciente
- Conectar un dispositivo médico ampliamente extendido en el entorno hospitalario, el monitor de presión arterial no invasiva, a un sistema de HCE mediante una pasarela o middleware, ambos de código abierto, usando los estándares y recomendaciones actuales para la interoperabilidad de equipos médicos.

En segundo término, y como resultado de la consecución de los anteriores objetivos centrales, los secundarios:

- Analizar los diferentes estándares, las alternativas de interoperabilidad y las opciones de integración actuales para los dispositivos médicos en cabecera del paciente
- Desarrollar un caso de uso determinado susceptible de ser implementado para el mismo tipo de equipo médico de diferentes fabricantes que cumplan con los requisitos definidos en este trabajo

1.3 Enfoque y metodología

El modelo de desarrollo que se usará para el presente TFG está basado en una metodología ágil, un marco de trabajo basado en procesos de ciclos cortos e iterativos donde la consecución de los objetivos y la entrega del producto se estructuran con el trabajo realizado por un equipo multidisciplinar, no especializado, en el que los participantes trabajan de forma colaborativa. Concretamente la metodología será SCRUM, propuesta por Ken Schwaber y Jeff Sutherland en el año 1996.

SCRUM reposa sobre los pilares de la transparencia, la inspección continua del trabajo que funciona y la adaptación a los cambios. El resultado es la generación de productos intermedios operativos, incrementales en cada ciclo, que construyen con el desarrollo ágil el producto principal, final.

Si bien el desarrollo del TFG tiene una duración corta para aplicar completamente la metodología SCRUM, se pueden aplicar sus componentes más importantes, definiendo en el marco de desarrollo los siguientes:

La **pila del producto** representa los requisitos priorizados del TFC, es una lista en evolución, compuesta por las actividades de las diferentes PACs

La **planificación de cada ciclo** de desarrollo o **sprint** representa cada una de las PACs. Al inicio de cada PAC se planifica el esfuerzo, los requisitos para elaborarla y sus prioridades, se confecciona **la pila del sprint** como camino de ruta y compromiso en la ejecución.

Durante el transcurso del **sprint** se revisa de forma diaria la evolución del trabajo y se proponen mejoras o cambios que puedan ser necesarios para la consecución correcta del objetivo marcado mediante la actualización de la **pila del sprint**.

En el contexto del TFG, el **sprint** representa el desarrollo de cada PAC y la monitorización y/o modificación del camino, las actividades, recursos, etc. para su correcta realización.

La finalización de cada **sprint** entrega una parte del producto o **incremento iterativo**, desarrollado y funcional.

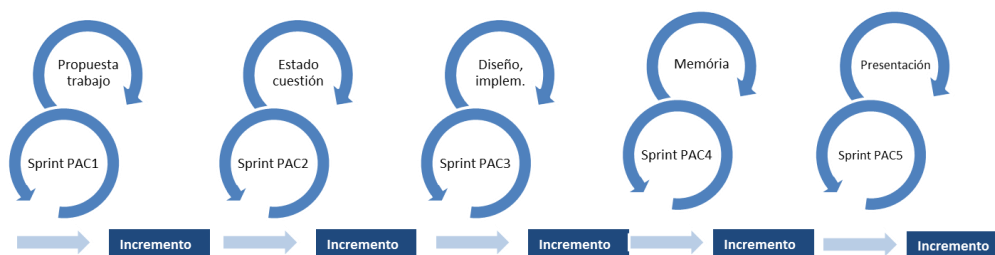


Figura 1. Metodología SCRUM y sprints del TFG

1.4 Planificación del Trabajo

El diagrama de Gantt muestra las diferentes actividades y los hitos más importantes del desarrollo del TFG, dentro del marco temporal establecido para su finalización.

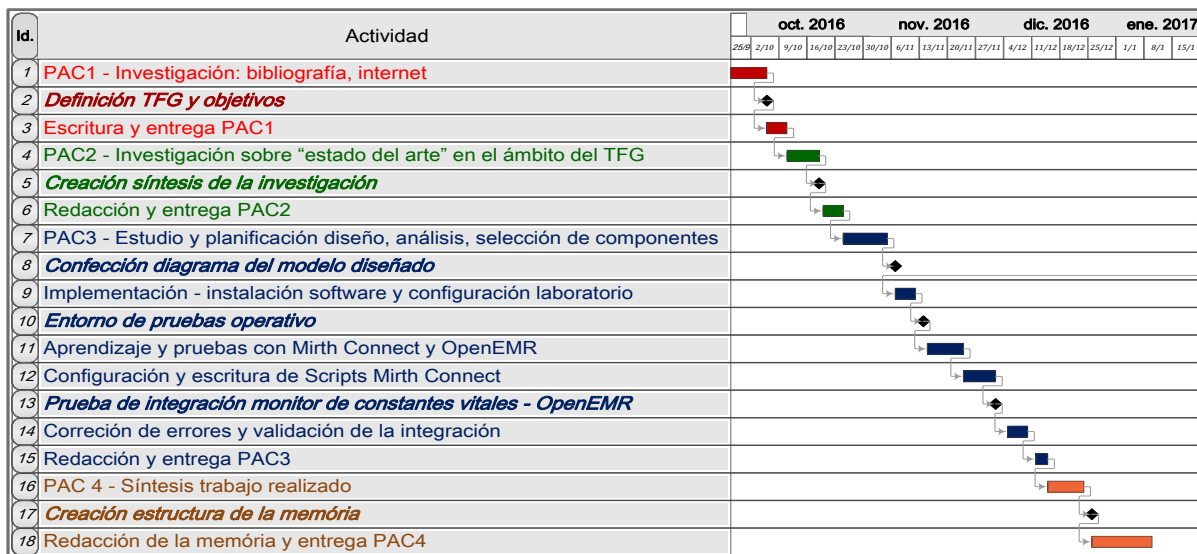


Figura 2. Planificación del TFG

Los principales recursos necesarios para la realización del proyecto son:

- Ordenador con suficientes recursos para la virtualización del laboratorio
- Elementos para la creación de una red informática: conmutador, punto de acceso WIFI y cables Ethernet cat5/6
- Software "Open Source" para la creación de la interface de conectividad y del sistema de gestión de Historia Clínica Electrónica del paciente; un sistema gestor y bases de datos
- Software de soporte: Entorno IDE, simulador mensajería HL7
- Acceso a Internet para descargas, consulta de bibliografía, foros, etc.
- Monitor de constantes vitales

1.5 Breve resumen de los productos obtenidos

Finalmente, a través de los hitos marcados se obtienen productos incrementales que conllevan el desarrollo y consecución de los objetivos definidos en el TFC. A modo de resumen su relación es la siguiente:

- Informe de propuesta del TFC y una definición clara de objetivos y su alcance
- Informe del análisis y las conclusiones de la situación actual o estado de la cuestión, referente a los estándares y las recomendaciones, los proveedores y las estrategias relacionadas con la conectividad e interoperabilidad de los dispositivos médicos más usados en el entorno hospitalario

- Diseño, propuesta de la solución y de los recursos necesarios que dan respuesta el caso de uso descrito
- Realización del modelo-prototipo para la implementación de la solución, pruebas de validación y documentación mediante soluciones “*Open Source*”
- Memoria del trabajo, informe de conclusiones y propuestas futuras

1.6 Breve descripción de los capítulos de la memoria

El trabajo se estructura y desarrolla fundamentalmente en cinco capítulos.

El primer capítulo se centra en la definición del trabajo, su extensión, motivación y la definición y descripción de los objetivos y los entregables.

Un segundo capítulo se dedica al “estado de situación” de la materia relacionada con el TFG para fijar la base, el punto de partida y el posterior desarrollo del proyecto.

El capítulo tres se centra íntegramente en el diseño y en la selección de los diferentes componentes que configurarán el caso de uso definido para la toma de constantes con un monitor y su integración en la HCE.

El cuarto se dedica a la implementación del proyecto y la validación de la integración del monitor de contantes vitales con la HCE.

Finalmente, y en el quinto, se resumen las conclusiones obtenidas durante el TFG.

2. Estado del arte

2.1 La realidad de los dispositivos médicos

Los equipos o dispositivos médicos abarcan categorías muy variadas. Desde la perspectiva de la integración y la interoperabilidad los que tienen una conexión directa con el paciente, tanto para fines diagnósticos como terapéuticos, son los que se encuentran más aislados con respecto al intercambio de datos e información con los sistemas de información clínicos de los hospitales, especialmente, los dispositivos que integran más de un parámetro vital.

A lo largo de todo el hospital y en las unidades de urgencias, de cuidados intensivos, los quirófanos y otras unidades, la presencia de bombas de infusión de fármacos, los ventiladores, los monitores de paciente, las máquinas de hemofiltración, etc. constituyen una de las herramientas fundamentales que los profesionales sanitarios tienen a su alcance para atender a los pacientes.

Los parámetros ajustados y los medidos o monitorizados por todos estos dispositivos médicos son parte importante de la documentación que acompaña al paciente durante su estancia hospitalaria. Sin embargo, en la mayoría de los casos los datos son registrados manualmente por el profesional sanitario debido a múltiples factores que impiden una captura fácil y automatizada.



Bomba infusión Monitor UCI Monitor constantes Ventilador¹

Figura 3. Dispositivos médicos

El primer factor que dificulta la conectividad es precisamente la especialización de cada dispositivo médico que marca la finalidad de su uso y, por tanto, condiciona su diseño, fabricación y comercialización, sometidos a rigurosos procesos de calidad y cumplimiento de normas y legislación de obligado cumplimiento para proteger a los pacientes delante de eventuales errores en su funcionamiento, generación de falsas alarmas etc.

¹ Fotos equipos médicos de <http://medicalequipmentdynamics.com/en/> [en línea, 15 de octubre, 2016]

La especialización y la seguridad del paciente han sido históricamente argumentos que han marcado un diseño propietario de los fabricantes para garantizar el correcto funcionamiento del producto en el entorno clínico y promover la diferenciación entre los competidores.

Son pocas las compañías, siempre multinacionales, las que tienen capacidad de fabricar dispositivos médicos, mantenerlos y evolucionarlos en el tiempo durante un ciclo de vida que se encuentra en el entorno de los 10-15 años.

Todos estos equipos médicos tienen un sistema operativo y protocolos propietarios, desarrollados y diseñados por cada fabricante para optimizar el funcionamiento en tiempo real, determinista en la ejecución de acciones, generación de alarmas y presentación de la información a los usuarios.

La heterogeneidad de equipos y fabricantes conlleva una falta de compatibilidad a todos los niveles que en la práctica impide el intercambio de datos entre equipos de diferentes categorías, dentro de la misma categoría entre diferentes fabricantes, incluso para generaciones diferentes del mismo dispositivo médico para el mismo fabricante.

2.1.1 Conectividad y salida de datos

No obstante, estos “silos de datos e información” que son los dispositivos médicos, han incorporado en los últimos años alguna salida de datos a través de conectores, no homogenizados y diferentes en cada caso, para la captura por parte de un ordenador o sistema externo que permita su procesamiento.



Extraído del catálogo de especificaciones técnicas monitor Philips M8003A² *“Data Out can be configured up to two times for each monitor, either on two ports of one MIB/RS232 board or on one port each on two MIB/RS232 boards”*



Extraído del catálogo de especificaciones técnicas monitor GE Carescape B450³ *I/O Connectors, Serial Port Available via USB converter*

Figura 4. Monitores multiparamétricos de paciente

² http://www.medtechnica.co.il/files/Monitors%20and%20Defibrillators/IntelliVue_MP40_50_Patient_Monitor.pdf, [en línea, 15 de octubre, 2016]

³ http://www3.gehealthcare.com/en/products/categories/patient_monitoring/patient_monitors/carescape_monitor_b450#tabs/tab_E5E8D68EF50B42A8B42FF9DE4973E2DA, [en línea, 15 de octubre, 2016]

En ambos casos ninguno de los fabricantes especifica qué tipo de protocolo utiliza, incluso los conectores de salida de datos son distintos; todavía es la situación más extendida, protocolos propietarios y conexiones dispares.

2.1.2 Protocolos de comunicación

La mayoría de los fabricantes de los dispositivos médicos pueden suministrar la documentación relacionada con el protocolo de comunicación de sus equipos. El protocolo detalla qué datos están disponibles, la codificación usada, el formato y estructura de los mensajes, y cómo capturarlos para facilitar a los ingenieros programadores el desarrollo de drivers específicos que los integren en los sistemas de información.

Un ejemplo de la documentación que detalla las especificaciones de este tipo de interface se muestra en el manual de referencia del ventilador de UCI “*SERVO-i / SERVO-s, Computer Interface Emulator*” del fabricante Maquet Critical Care AB. El documento detalla cómo se codifican, interpretan y se obtienen los valores de los diferentes parámetros numéricos, curvas y alarmas⁴.

La interface serie envía los datos en formato ASCII o binario:

4.4 The RS-232 communication settings

Baud rate: 9600
 Data length: 7 or 8 bits
 Stop bits: 1 or 2 bits
 Parity: Even
 Data format: ASCII or Binary (Binary format requires 8 data bits)
 Handshake: XON/XOFF
 Both the Data length (7 or 8 bits) and Stop bits (1 or 2 bits) are auto-detected.

La estructura de alguno de los múltiples mensajes y sus datos enviados:

Parameters:	<k>	= 00 – 33, trend channel number
	<year>	= 00 – 99
	<month>	= 01 – 12
	<day>	= 01 – 31
	<hour>	= 00 – 23
	<minute>	= 00 – 59
	<sec>	= 00 – 59
	<start_time>	= <year><month><day><hour><min><sec>
	<length>	= <hour><min><sec>
	<t _{01 <td>= <year><month><day><hour><min><sec></td>}	= <year><month><day><hour><min><sec>
	<value _{(0)(-n + 1) <td>= 0000 – 9999</td>}	= 0000 – 9999

⁴ http://www.medteh.info/fr/16/SERVOi_SERVREFM.pdf [en línea, 17 de octubre, 2016]



Figura 5. Ventilador Maquet Servo-i⁵

Los dispositivos de monitorización suelen conectarse en una estructura de red para centralizar la vigilancia en un control de monitorización. Aunque Ethernet es la tecnología más extendida, el protocolo de red sigue siendo propietario e incompatible entre fabricantes. En este caso la alternativa es similar a la conexión por el puerto serie, requiriéndose un driver específico para capturar y traducir los datos capturados de la red hacia el sistema externo.

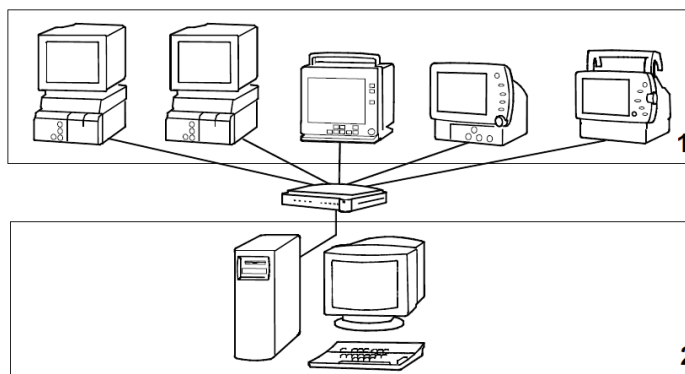


Figura 6. Esquema de la red de monitorización Datex-Ohmeda S/5⁶

1. Monitores de paciente conectados a través de un switch
2. Central de monitorización

Los monitores son del tipo de vigilancia continua en áreas de UCI, urgencias, reanimación, etc. Su instalación se realiza sobre la infraestructura cableada Ethernet del hospital, un ancho de banda estándar de 100 Mbps es suficiente para su correcto funcionamiento ya que los datos e información que generan en forma de curvas, valores numéricos y alarmas son mensajes de poco peso.

⁵ Foto extraída de <http://www.maquet.com/es/productos/servo-i/?ccid=32>, [en línea, 18 de octubre, 2016]

⁶ Datex-Ohmeda S/5 Central, ViewStation and Network Technical Reference Manual, http://www.frankshospitalworkshop.com/equipment/documents/pulse_oximeter/service_manuals/Datex_Ohmeda_S-5_-_Service_manual.pdf, [en línea, 18 de octubre, 2016]

2.2 La realidad en los hospitales

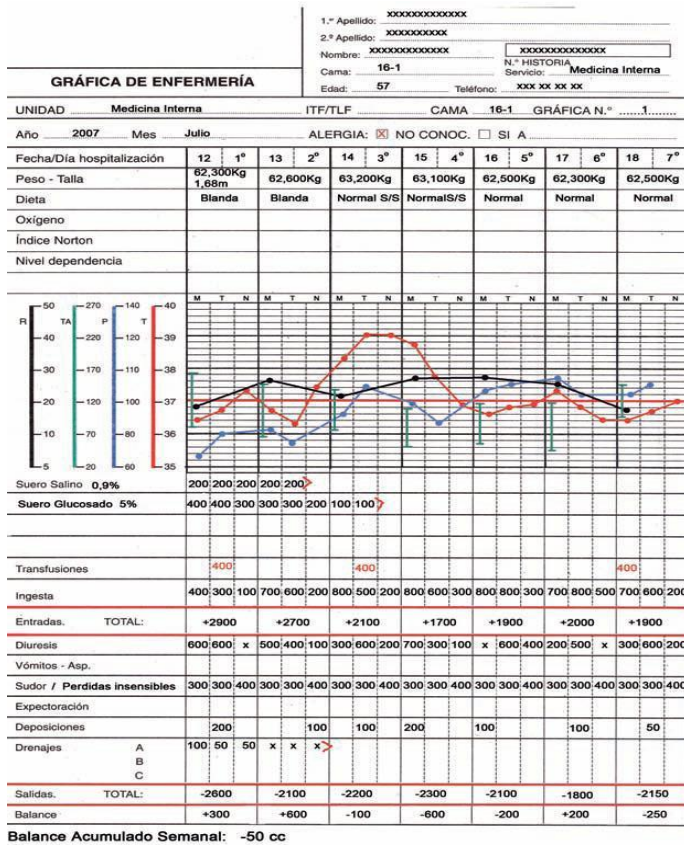
El modelo y la implantación de la HCE están bien desarrollados y desplegados a través de proyectos concretos que dan respuesta a las necesidades de los ciudadanos, de los profesionales de la salud y de herramientas que permiten una gestión eficaz y eficientemente del sistema. Algunos ejemplos son el proyecto IANUS de la Consejería de Sanidad de la Xunta de Galicia, ARGOS del Institut Català de la Salut [3], DIRAYA del Servicio Andaluz de Salud, etc.

En todos ellos el paciente es el centro del sistema y la estación clínica la principal herramienta de acceso a la HCE, que integra, fundamentalmente, la gestión administrativa (consultas externas, listas de espera, programaciones...), la gestión clínica (órdenes médicas, tratamientos, informes...) y la integración de sistemas de información departamentales que soportan los procesos asistenciales (laboratorio, farmacia, diagnóstico por la imagen...).

No obstante, el reto de los proyectos es la integración de los datos provenientes de los dispositivos médicos, la conectividad directa con ellos y su integración en los procesos asistenciales para el registro y la documentación de los parámetros ajustados, pautados y monitorizados por los equipos.

Algunos escenarios reales en la actualidad, entre otros:

- Servicios de urgencias que toman constantes vitales del paciente y se documentan manualmente en papel o estación clínica
- Procesos quirúrgicos donde se genera la “hoja de anestesia” manualmente sobre plantilla de papel.
- Las unidades de cuidados críticos que registran todas las constantes del paciente y las infusiones de fármacos sobre la gráfica diaria.



Gráfica cuidados en UCI de registro manual.

Las tendencias coloreadas corresponden a las principales constantes vitales del paciente: frecuencia cardíaca, respiratoria, presión arterial y temperatura.⁷

Figura 7. Gráfica de cuidados en UCI registro manual

Muchos hospitales están realizando inversiones alineadas con esta necesidad de registro electrónico y automatizado de la captación de datos, especialmente en las unidades de críticos y en las salas de hospitalización.

2.3 Las soluciones disponibles

La heterogeneidad de los dispositivos médicos, la falta de estandarización y de acuerdos marcos de amplia aceptación en el sector sanitario dan lugar a la aparición de múltiples opciones y soluciones técnicas que llevan asociadas la venta de productos y servicios, hasta cierto punto, propietarios del proveedor y, por tanto, que pueden dificultar o hipotecar la evolución temporal de la solución adoptada.

Los escenarios o las estrategias de integración que actualmente pueden plantearse requieren de un análisis y consenso entre los diferentes actores dentro de la organización sanitaria (profesionales sanitarios, ingenieros de electromedicina, informática y sistemas de información...) antes de tomar una decisión que defina el camino a seguir y la estrategia de implantación en el tiempo.

⁷ <http://enfermeriazenaira.blogspot.com.es/2013/03/hojas-de-registro.html>, [en línea, 19 de octubre, 2016]

2.3.1 Integración mediante sistema clínico departamental

El escenario integra los dispositivos médicos a través de una solución comercial o desarrollada a medida. Las conexiones se realizan a través de los puertos serie, el proveedor del SI clínico desarrolla los drivers específicos de cada equipo, los integra en su sistema y los pone a disposición de su aplicación clínica para su tratamiento.

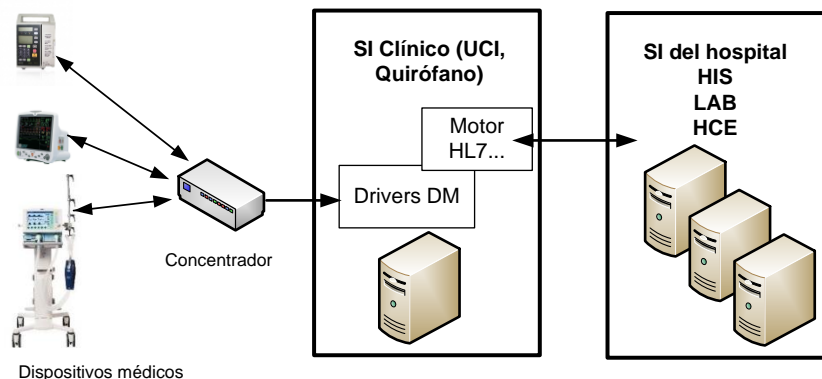
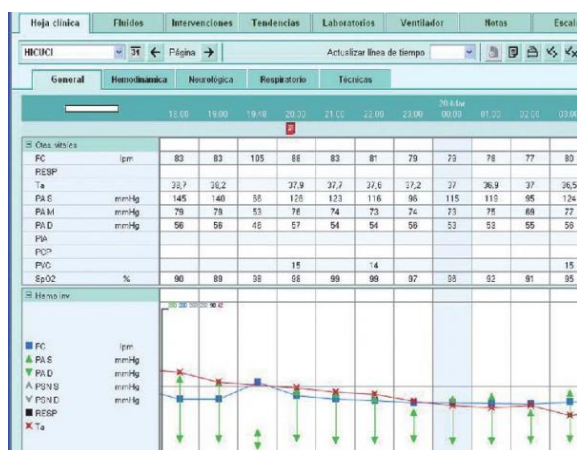


Figura 8. Esquema integración mediante sistema información clínico⁸

La integración con la HCE se realiza mediante un motor de integración que traduce y convierte los datos y protocolos propietarios a mensajes estandarizados para el intercambio de información mediante HL7, XML, etc.

Dentro de este tipo de esquema se enmarcan soluciones comerciales como iMDSoft Metavisión⁹, GE Centricity Critical Care¹⁰, Philips IntelliSpace Critical Care¹¹.



Gráfica cuidados en UCI con captura de datos automatizada de los dispositivos médicos integrados¹².

Registro de constantes vitales del paciente en formatos numérico y de tendencias gráficas.

Figura 9. Gráfica de cuidados en UCI registro automático

⁸ Fuente: Diseñado y elaboración propios

⁹ <http://www.imd-soft.com/products/intensive-care>, [en línea, 18 de octubre, 2016]

¹⁰ http://www3.gehealthcare.es/es-es/productos/categorias/healthcare_it/high_acuity_care_solution/centricity_critical_care#tabs/tabF430B545557A41D6BDB45343367809E2, [en línea, 18 de octubre, 2016]

¹¹ <http://www.philips.es/healthcare/product/HCNOCTN332/intellispace-critical-care-and-anesthesia>, [en línea, 18 octubre 2016]

¹² <http://periodicos.saludextremadura.com/m/noticias/noticia.php?id=698>, [en línea, 19 de octubre, 2016]

2.3.2 Integración mediante proveedor de conectividad empresarial

La estrategia desplaza la integración de todos los dispositivos médicos de los diferentes departamentos del hospital a una capa intermedia que asume la conectividad y la gestión de los drivers, normaliza los datos y los pasa a los diferentes sistemas de información disponibles en el hospital mediante protocolos ampliamente aceptados, HL7, XML, etc.

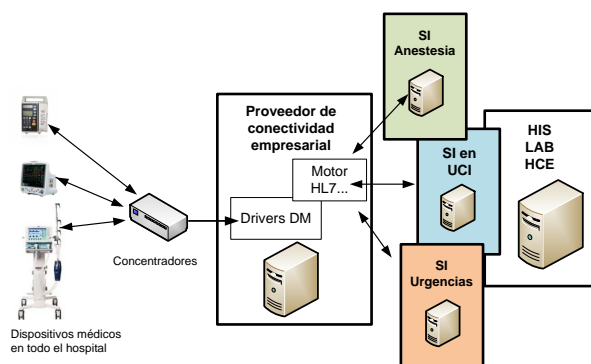


Figura 10. Esquema integración mediante capa intermedia de conectividad¹³

Proveedores que comercializan este tipo de estructura e integración: Capsule Tech Inc.¹⁴, Cerner CareAware iBus¹⁵.

2.3.3 Integración mediante pasarelas de dispositivos

Existen categorías de dispositivos médicos ampliamente extendidos e instalados en los hospitales en múltiples servicios como son los monitores de paciente y las bombas de infusión de fármacos. Conscientes de la necesidad de integración, los fabricantes de estos equipos proponen pasarelas específicas que concentran y agrupan sus dispositivos médicos, traducen su lenguaje propietario y normalizan los parámetros y sus valores hacia interfaces de salida mediante mensajería HL7.

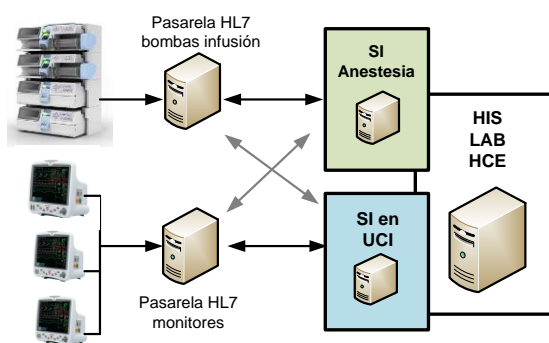


Figura 11. Esquema integración mediante pasarelas de dispositivos¹⁶

¹³ Fuente: Diseñado y elaboración propios

¹⁴ <http://www.capsuletech.com/smartlinx>, [en línea, 19 de octubre, 2016]

¹⁵ http://www.cerner.com/solutions/medical_devices/device_connectivity/, [en línea, 19 de octubre, 2016]

¹⁶ Fuente: Diseñado y elaboración propios

Alguna de estas propuestas se materializa en productos como el Hospira MedNet Gateway¹⁷ para el caso de bombas y el GE Carescape Gateway Server para los monitores¹⁸.

Cualquiera de las soluciones anteriores da respuesta a la integración y la interoperabilidad de los dispositivos médicos con los sistemas de información clínicos departamentales y/o la HCE. Sin embargo, todos ellos requieren de un determinado nivel de estandarización de los dispositivos con un fabricante concreto, de una inversión importante en términos económicos, de proyectos de integración complejos y de contratos de mantenimiento correctivos y evolutivos durante el ciclo de vida de los productos.

La flexibilidad en términos de reemplazo de alguno de los componentes de la solución es limitada por la alta dependencia e interrelación de dispositivos, drivers y pasarelas que continuamente están sometidos a actualizaciones de software, comprometiendo las interfaces implementadas y requiriendo en muchos casos reconfiguraciones para mantenerlas activas; todo ello conlleva un alto coste.

Los concentradores¹⁹ de las figuras 8, 10 y 11, como los del fabricante Lantronix, son *switches* especiales que aíslan eléctricamente a los dispositivos médicos conectados a los servidores y que, si bien recogen muchos valores, éstos son parámetros numéricos y de muy poco peso; en entornos de anestesia y UCI se suele instalar un concentrador por quirófano-box, es un tema de ergonomía más que de saturación de la red Ethernet de datos.

No obstante, dependiendo del número de pacientes y de los equipos conectados a cada uno de ellos, los servidores requieren unas especificaciones y recursos concretos para poder gestionar todos los datos que reciben, presentarlos a los usuarios, etc. Dependiendo de la complejidad y el tamaño de la instalación, los fabricantes de los sistemas de información departamental y de la solución de conectividad empresarial marcan los requerimientos mínimos de la infraestructura en términos de CPU, RAM, capacidad de discos, redundancia, etc.

En el caso de las pasarelas dedicadas a los dispositivos, los requerimientos y las dimensiones de la implementación viene delimitada en el mismo producto, los dispositivos y la pasarela son suministrados por el mismo fabricante y, por tanto, no depende de terceras empresas o de la conexión de otros dispositivos.

¹⁷ https://www.hospira.com/en/products_and_services/clinical_software/mednet, [en línea, 19 de octubre, 2016]

¹⁸ http://www3.gehealthcare.com/en/products/categories/patient_monitoring/networking/carescape_gateway#tabs/tabB5B0BC147071498B8C744537C8AD9B65, [en línea, 19 de octubre, 2016]

¹⁹ <https://www.lantronix.com/products/eds-md/>, [en línea, 19 de octubre, 2016]

2.4 La búsqueda de la solución

El escenario óptimo de la interoperabilidad es aquel en que únicamente se tuviera que conectar el dispositivo médico a otro equipo o sistema de información y entre ellos se entendieran sin necesidad de configurarlos. El objetivo perseguido es la “Interoperabilidad Plug&Play”

2.4.1 El uso de estándares y convenciones

El uso de estándares en los dispositivos médicos proporciona soluciones más efectivas y eficientes durante su diseño y fabricación que facilitan el retorno de la inversión durante la fase de implementación a través de un mayor número de instalaciones.

La adopción de los estándares, armonizaciones y consensos sobre los dispositivos médicos representa la base para mejorar la conectividad y la interoperabilidad entre equipos y con los sistemas de información clínica. Su uso se extiende a las diferentes capas del modelo OSI para definir diferentes conceptos que abarcan la integración y la interoperabilidad:

Concepto	Estándar, propuesta	Referencia
Interacción, casos de uso	HL7, IHE, Continua, DICOM	http://www.hl7spain.org/ http://www.ihe-e.org/ http://www.continuaalliance.org/ http://dicom.nema.org/
Mensajes, contexto	Vocabulario, semántica: ISO/IEEE 11073; SNOMED-CT, CIE-10, LOINC... Formato-sintaxis: HL7, ISO/IEEE 11073, DICOM, Web Services... Codificación: HL7, XML, binario...	http://standards.ieee.org/index.html http://www.snomed.org/snomed-ct https://www.w3.org/XML/
Transporte, capa física	Ethernet, USB, RS-232C, TCP/IP, HL7-MLLP...	

Tabla 1. Estándares de conectividad y convenciones

Estos estándares y marcos de referencia están en continuo desarrollo en el sector de la sanidad a través de sus comités y asociaciones. Algunos están más relacionados y enfocados a los dispositivos médicos [4], [5].

2.4.2 HL7 – Health Level Seven

La organización HL7 se formó en el año 1987 como una asociación sin ánimo de lucro con el objetivo de desarrollar estándares para el intercambio de información, integración e interoperabilidad de los sistemas de información clínicos que soportan la actividad asistencia en el entorno sanitario. Actualmente la organización está formada por más de 500 socios que representan a los diferentes interesados en el sector salud (fabricantes, agencias gubernamentales, proveedores...).

El protocolo HL7 actúa en la capa siete, aplicación, de la arquitectura de comunicaciones del modelo OSI.

El modelo básico de transacciones se inicia con un evento disparador que hace que un sistema A envíe un mensaje concreto a otro sistema B receptor y, como respuesta, B devuelva un mensaje de respuesta al sistema A:

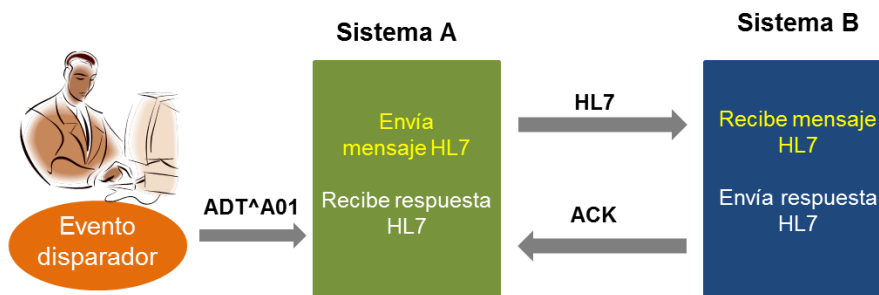


Figura 12. Esquema básico transacción HL7²⁰

Ejemplos de eventos que inician los mensajes pueden ser la admisión de un paciente, la petición de una prueba por parte de un facultativo, la disponibilidad de un resultado de laboratorio o de un informe, entre otros muchos.

Los **mensajes HL7** tienen una estructura bien definida formada por un conjunto de **segmentos** requeridos u opcionales; cada segmento identifica el tipo de información que lleva mediante un código de tres caracteres (p. ej. datos pacientes, resultado analítica, petición de prueba, etc.); cada segmento está formado con **campos** que almacenan **componentes** o datos.

Los **delimitadores** son caracteres que se utilizan para construir el mensaje y diferenciar los diferentes componentes de su estructura:

- Final de Segmento <CR>
- Separador de Campo |
- Separador de Componente ^
- Separador de Subcomponente &
- Carácter de Repetición ~
- Carácter de Escape \

²⁰ Fuente: Diseñado y elaboración propia

Ejemplo de mensaje HL7 v2²¹

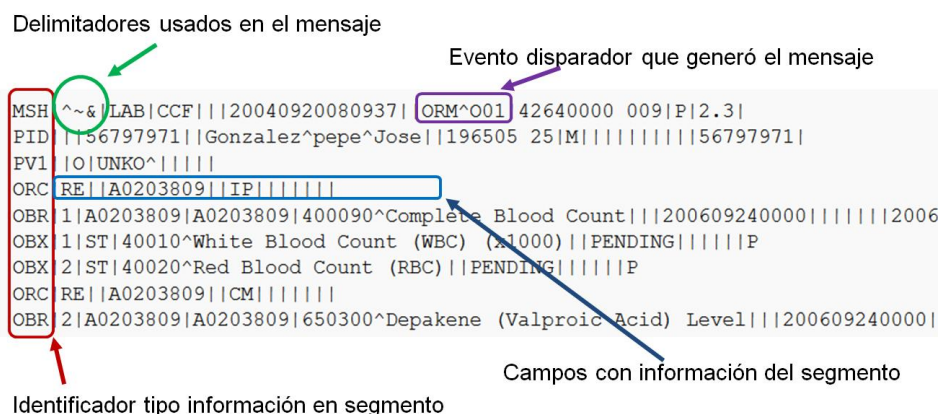


Figura 13. Ejemplo mensaje HL7

El mensaje contiene los siguientes segmentos:

- MSH** cabecera mensaje
- PID** identificación del paciente
- PV1** información de la visita del paciente
- ORC** orden de petición al laboratorio
- OBR** prueba solicitada
- OBX** información sobre la prueba

En el primer segmento se aprecia el mensaje ORM^O01 que generó la petición.

El desarrollo actual de los mensajes HL7 evoluciona y cubre prácticamente cualquier dominio y transacción en las áreas del entorno sanitario: gestión administrativa del paciente, peticiones de cualquier tipo, resultados y observaciones, planificación de agendas, planificación de cuidados del paciente, logística, finanzas, etc.

Existen dos versiones, la familia v2, la más usada e implementada en la práctica a través de sockets y protocolo MLLP, inició su andadura en el 1997 y ha sido actualizada a lo largo de los años, generando las versiones 2.1, 2.2, 2.3, 2.3.1, 2.4, 2.5, 2.5.1 y 2.6. Todas las versiones son compatibles hacia atrás.

En 2005 se publicó la v3 con un cambio significativo, la sintaxis usada en todos los mensajes se codifica en XML y los protocolos recomendados son SOAP. La v3 no es compatible con la familia v2 y, por tanto, las interfaces v2 no pueden comunicarse directamente con las implementadas en v3 sin realizar una modificación considerable.

²¹ <https://es.wikipedia.org/wiki/HL7>, [en línea, 21 de octubre, 2016]

El nivel de adopción de la v3 es muy bajo, incluso se habla de fracaso, debido a la incompatibilidad entre familias HL7, a la complejidad de su comprensión e implementación y a otros múltiples factores²².

El estándar HL7 tiene una extensión muy amplia y su estructura permite una gran variabilidad, no es una solución “Plug&Play”²³.

2.4.3 Estándar ISO/IEEE11073

La familia de normas ISO/IEEE11073, conocidas como X73, surgen de la armonización de estándares anteriores y fueron creadas para abordar el problema de la falta de transparencia, la facilidad de configuración, la integración y la interoperabilidad de los equipos médicos en los hospitales y en el punto de cuidado del paciente (X73PoC). Su andadura empieza con una primera versión (año 2004) centrada en el escenario de las unidades de cuidados intensivos.

En Europa las impulsa el Comité Europeo de Estandarización (CEN) a través de su comité técnico 251 y en España AENOR las promueve mediante su comité técnico normalizador 139.

Las ISO/IEEE11073 se encuentran en continua evolución generando nuevas versiones y normas que actualmente abordan la ubicuidad y los dispositivos de la salud personales a través de la familia X73PhD.

El estándar X73 cubre los siete niveles de la pila OSI para facilitar el intercambio de información entre los dispositivos médicos y un sistema centralizado de captura de datos [6].

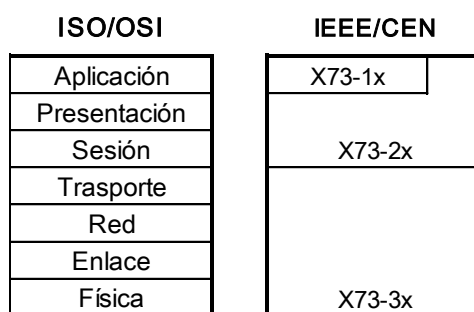


Figura 14. Capas modelo OSI y estándares X73

A nivel de mensajes e interoperabilidad semántica cada dispositivo debe usar la misma terminología y estructura de datos. El estándar define una estructura jerarquizada para la información donde cada dato asociado a un dispositivo especializado se especifica dentro del contexto del árbol a través de sus parámetros formalizados mediante atributos de su clase o como instancias de clase de datos

²² <http://hl7es.blogspot.com.es/2013/12/el-ocaso-de-hl7-v3.html>, [en línea, 22 de octubre, 2016]

²³ <http://hl7es.blogspot.com.es/2014/07/hl7-en-10-pasos.html>, [en línea, 22 de octubre, 2016]

dentro de otras clases. La estructura de clases básica se representa en el siguiente diagrama UML [7]:

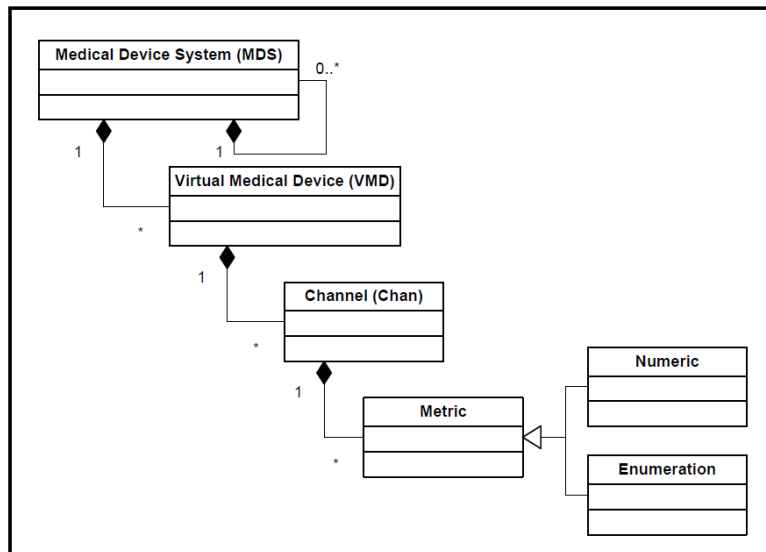


Figura 15. Esquema UML de clases del estándar X73

Las clases abstractas representan objetos reales, un ejemplo ilustrativo:

- **El sistema dispositivo médico (MDS)** identifica un contexto general del equipo y sus atributos: Monitor multiparamétrico, fabricante GE, modelo Dash 5000, nº serie XA23568Z. El MDS puede llevar asociados otros MDS, p. ej. la conexión a un ventilador.
- **El dispositivo médico virtual (VMD)** representa la especialización del equipo: Monitor hemodinámico, monitor de gases, etc.
- **La clase canal (Chan)** refleja los subcomponentes o bloques del VMD: Canal de ECG, canal de presión invasiva...
- **La clase métrica** es la única que hereda para proporcionar atributos especializados de clase: Presión invasiva activa
- **La clase numérica** refleja los valores representados numéricamente y sus unidades: 125, mmHg
- **La clase enumerada** representa los tipos personalizados del dispositivo: Presión arterial, Presión Venosa Central, etc.

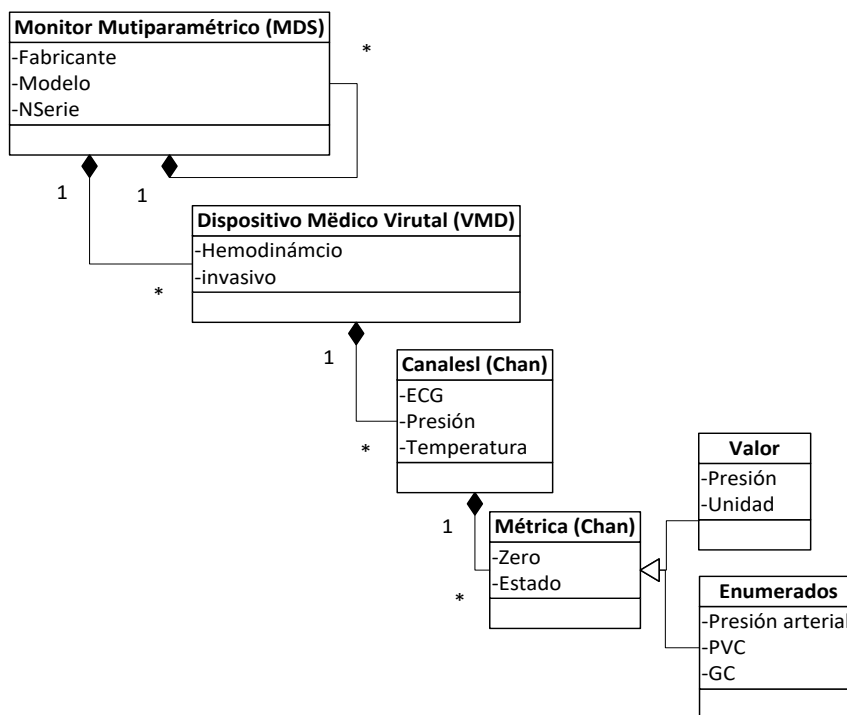


Figura 16. Ejemplo esquema UML de monitor constantes vitales²⁴

La nomenclatura ISO/IEEE11073 está optimizada para la semántica de los dispositivos médicos. Cubre una codificación muy amplia que soporta el modelo de información, los parámetros del dispositivo, sus unidades, eventos, alarmas, etc. Cada uno de los términos se representa mediante un identificador de referencia basado en texto y un código numérico de 16 o 32 bits.

Ejemplos del formato que identifica los términos:

Descripción	Codificación
Frecuencia respiratoria	MDC_RESP_RATE (2::20490)
Asistolia	MDC_EVT_ECG_ASYSTOLE (3::3076)
Taquicardia	MDC_EVT_ECG_TACHY (3::3120)

Tabla 2. Ejemplo nomenclatura ISO/IEEE11073 de constantes vitales

²⁴ Fuente: Diseñado y elaboración propia a través de la interpretación de las clases UML del estándar X73

Los estándares X73 definen muchos modelos de información y sus objetos, cubriendo curvas dinámicas, alarmas de los dispositivos, etc. Representan un verdadero impulso en la categoría de los dispositivos médicos para la homogenización, facilitan la integración e interoperabilidad y su adopción está cada día más extendida en el sector sanitario

2.4.4 IHE - Integrating the Healthcare Enterprise

IHE se constituyó en el año 1998 como una asociación de usuarios de radiología y expertos en TIC para colaborar en el desarrollo de soluciones que permitieran la interoperabilidad entre sistemas de información y aplicaciones. Actualmente es una organización con delegaciones en muchos países, compuesta por muchos profesionales de la sanidad y la industria.

La filosofía es identificar problemas o casos de uso reales y, analizando sus requerimientos y flujo de trabajo, proponer una solución basada en los estándares existentes (HL7, DICOM...), desarrollar el marco técnico y proponer a los fabricantes su adopción en los productos, son los denominados **perfiles IHE**.

Los dominios clínicos cubiertos actualmente por IHE abarcan, entre otros, la Radiología, Cardiología, Farmacia, Laboratorio, Anatomía patológica, Endoscopia y los **Dispositivos de Cuidado del Paciente**.

En cada uno de estos dominios existen perfiles que han sido creados siguiendo una metodología de trabajo, activada por el comité implicado que acuerda la necesidad de una solución al caso planteado y finalizada con la publicación del marco técnico de un nuevo perfil IHE²⁵.

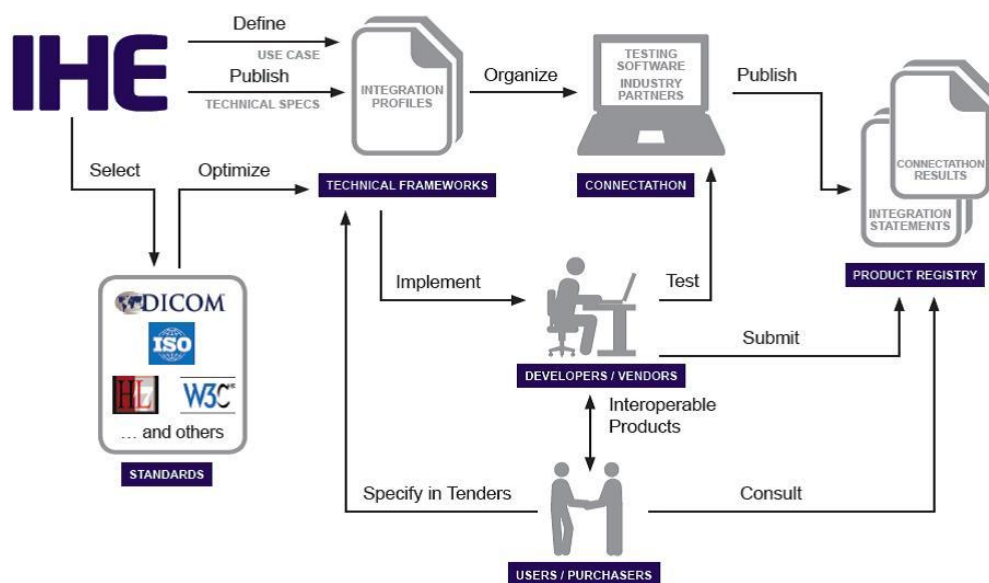


Figura 17. Proceso para la creación de un perfil IHE

²⁵ https://www.ihe.net/IHE_Process/, [en línea, 22 de octubre, 206]

Parte importante de este proceso son los fabricantes que participan en la fase de implementación del perfil y su comprobación en eventos anuales denominados “**Connectathon**”, reuniones donde la industria muestra cómo sus productos adoptan el perfil para facilitar la interoperabilidad. Su próxima reunión 2017 se realiza en Venecia.²⁶

El **perfil IHE** a través de su marco técnico, guía detallada que identifica los **actores** o sistemas que se pretenden integrar, define sus **transacciones**, las reglas y los estándares que deben usar para compartir e intercambiar la información.

El dominio **IHE-PCD** (Patient Care Device) es el específico de los dispositivos médicos. Surgió en 2005 para dar respuesta a la integración de datos entre los equipos y la HCE. Su objetivo es reducir el coste y la complejidad de las integraciones mediante soluciones interoperables concretas. Define una serie de perfiles que describen escenarios clínicos reales y que fueron diseñados para intercambiar datos o alarmas de diferentes tipos de dispositivos médicos.

Los perfiles simplifican el desarrollo de las integraciones, desde los estándares aceptados se desarrollan marcos técnicos concretos para casos de uso. La situación actual de los diferentes perfiles del dominio se presenta en el siguiente diagrama [8]:

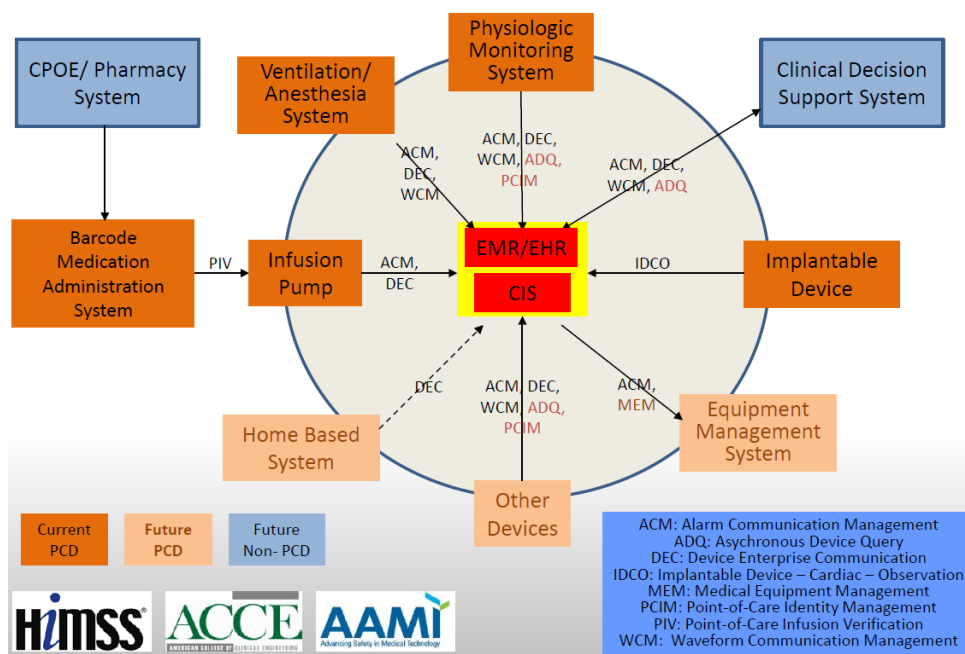


Figura 18. Dominios IHE y perfiles

El perfil DEC (Device Enterprise Communication) define el caso de uso de un dispositivo médico, DOR o Device Observation Reporter, que envía información clínica (signos vitales, localización y ajustes) al sistema consumidor, DOC o Device Observation Consumer, que los recibe. [9]

²⁶ <http://connectathon.ihe-europe.net/>, [en línea, 22 de octubre, 2016]

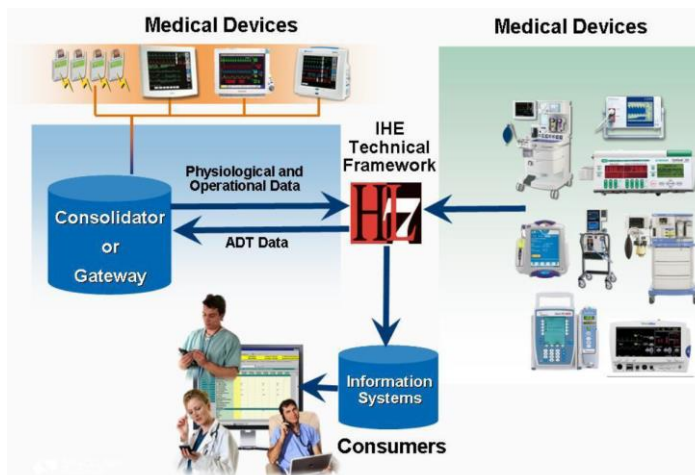


Figura 19. Perfil DEC y caso de uso

La “**interoperabilidad semántica**” de los perfiles, y por tanto la verdadera integración entre dispositivos y/o sistemas de información, requiere que todos los participantes usen y compartan una misma representación semántica, uniforme en las mediciones y unidades de las mismas. El proyecto **Rosetta Terminology Management** (RTM) surgió dentro del grupo IHE con el objetivo de armonizar los términos que usaban diferentes dispositivos y fabricantes. Se analizaron diferentes estándares como SNOMED, LOINC e ISO/IEEE 11073-10101 y se observó que el ISO/IEEE reflejaba mejor los términos de los dispositivos médicos.

El proyecto ha conseguido, a través de un proceso abierto de consenso, armonizar la terminología (hRTM) mapeando los términos y las unidades usados por los principales fabricantes a unos identificadores únicos de términos, basados en la nomenclatura ISO/IEEE 11073-10101²⁷.

Actualmente están armonizados más de 900 términos. La tabla armonizada hRTM está disponible públicamente, un ejemplo de términos armonizados²⁸:

Grupo	REFID	UOM_MDC	UOM_U CUM
CVS_ECG_HR	MDC_ECG_HEART_RATE	MDC_DIM_BEAT_PER_MIN	/min 1/min {beat}/min
TEMP	MDC_TEMP_AXIL	MDC_DIM_DEGC MDC_DIM_FAHR	Cel [degF]
RESP_APNEA	MDC_TIME_PD_APNEA	MDC_DIM_SEC	s

Tabla 3. Ejemplo terminología Rosetta

²⁷ http://wiki.ihe.net/index.php/PCD_Profile_Rosetta_Terminology_Mapping, [en línea, 23 de octubre, 206]

²⁸ <https://rtmms.nist.gov/rtmms/index.htm#!hrosetta>, [en línea, 23 de octubre, 206]

Resumiendo, **IHE es la mejora propuesta actual** que se aproxima a la solución ideal mediante el uso de estándares y consensos expresados en perfiles que dan respuesta a casos de uso concretos:

- A nivel mensajes usa HL7 V2.6 restringido por IHE DEC PCD-01 y RTM
- A nivel semántico la nomenclatura base ISO/IEEE 11073-10101 y hRTM
- A nivel de transporte HL7 V2 con protocolo MLLP sobre TCP/IP

Algunos fabricantes ya han incorporado en los últimos dispositivos médicos que comercializan la propuesta IHE. La paulatina adopción por parte de todos los actores implicados y su implantación en casos reales son los mejores promotores e impulsores del cambio. Un ejemplo claro es la reciente decisión tomada por la **Comisión Europea** en julio de 2015 de seleccionar **27 perfiles IHE** para promover su uso en el **marco de la contratación pública**²⁹.

Por su lado, IHE y sus comités continúan trabajando, ampliando el desarrollo de nuevos perfiles e impulsando su uso a través de la publicación de los marcos técnicos.

²⁹ <http://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX:32015D1302>, [en línea, 23 de octubre, 2016]

3. Diseño

El objetivo perseguido en esta fase se centra en la definición de la maqueta que será implementada posteriormente y el caso de uso definido para el registro e integración de las constantes vitales discretas de pacientes en la HCE de un hospital.

3.1 Caso de uso del registro de constantes vitales

El registro de constantes en el entorno hospitalario suele producirse en entornos que requieren de la movilidad del dispositivo médico hacia la ubicación del paciente. Los actores y/o roles que intervienen en este escenario son varios, entre ellos configuran un flujo de trabajo. En el caso planteado en el presente proyecto su interrelación es la siguiente:

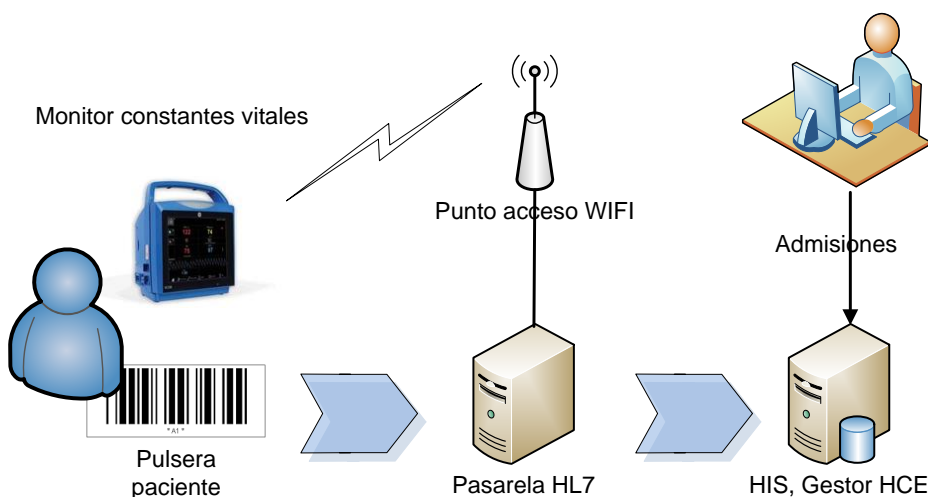


Figura 20. Esquema caso de uso registro constantes vitales³⁰

El flujo de trabajo asociado al registro de las constantes del paciente se realiza a través de un sencillo protocolo de cinco pasos, diseñado para el caso de uso:

1. En primer lugar, el paciente se registra y admite en el sistema de gestión del hospital (HIS), la visita se asocia a su número de Historia Clínica
2. Se imprime la pulsera que identifica al paciente con un código de barras. Entre los datos identificativos, la codificación del número de la HCE es el identificador inequívoco y el más importante del paciente
3. El profesional sanitario realiza una identificación positiva del paciente y lo registra en el monitor con la ayuda del lector de código de barras
4. Toma las constantes vitales de la presión arterial, la temperatura y la pulsioximetría, añade comentarios y valida las determinaciones
5. El monitor de constantes³¹, mediante la infraestructura WIFI y a través del motor de integración HL7, envía las constantes vitales del paciente a la HCE

³⁰ Fuente: Diseñado y elaboración propios

La pasarela HL7 debe dimensionarse con una capacidad suficiente que permita gestionar el volumen de datos máximo que pudiera darse en un momento determinado. No obstante, y dado que los monitores de constantes vitales sólo envían mensajes con poco peso (valores y texto plano) de manera puntual o discreta cuando el personal médico toma las constantes del paciente y las valida, los requerimientos técnicos de la pasarela HL7 son básicos y muy genéricos.

A modo de ejemplo, un hospital de 1000 camas puede tener en el entorno de los 150-250 dispositivos para la toma de constantes discretas; difícilmente, puede coincidir que más de un 25% se encuentre enviando datos a la HCE a través de la pasarela y, por tanto, el volumen de datos enviado a través de la infraestructura WIFI a la pasarela HL7 es mínimo.

3.2 Esquema y estructura de la solución tecnológica

El modelo a nivel de arquitectura, los componentes de software y de hardware del laboratorio de pruebas diseñado para este proyecto, pretende reflejar un escenario habitual en la estructura tecnológica de los hospitales.

Uno de los objetivos iniciales estaba enfocado en proponer una alternativa “Open Source” del máximo número de componentes de la plataforma para minimizar el coste asociado a las licencias y su mantenimiento, huyendo de las alternativas basadas en soluciones propietarias. La siguiente figura refleja muestra el modelo.

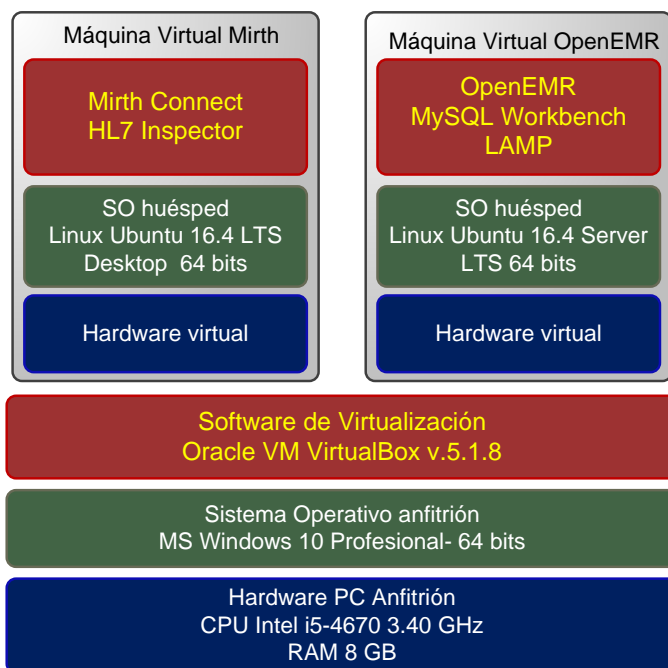


Figura 21. Componentes y virtualización del entorno de pruebas³²

³¹ Monitor marca GE Healthcare modelo CARESCAPE VC150; http://www3.gehealthcare.com/en/products/categories/patient_monitoring/patient_monitors/carescape_vc150 [en línea, 29 de noviembre, 2016]

³² Fuente: Diseñado y elaboración propios

La plataforma se instalará sobre un PC Windows 10 Pro con suficientes recursos sobre el que se crearán, mediante VirtualBox de Oracle, dos máquinas virtuales, la correspondiente a un ordenador de sobremesa Linux versión 16.4 LTS y la que despliega un servidor, también, versión Linux 16.4. Las dos máquinas virtuales representan dos servidores independientes en el entorno real, una destinada a la gestión de las integraciones del hospital y la otra a las aplicaciones HIS e HCE.

En cada máquina virtual se instalará el SO Linux en su variante Ubuntu mediante la obtención de las correspondientes imágenes, sobremesa y servidor, y se creará la conectividad para que se puedan intercambiar mensajes con las aplicaciones hospedadas y el exterior. Sus especificaciones:

- **Linux-Mirth:** SO Ubuntu 16.4 Desktop con entorno gráfico Gnome, 1 microprocesador, 1.5 MB RAM, 20 GB SATA HDD, tarjeta de red configurada en modo “puente”
- **Linux-OpenEMR:** SO Ubuntu 16.4 Server, 1 microprocesador, 1 MB RAM, 20 GB SATA HDD, red configurada en modo “puente”.

En la máquina servidor se instalará un entorno gráfico ligero, Lubuntu, para facilitar el trabajo en combinación con la consola.

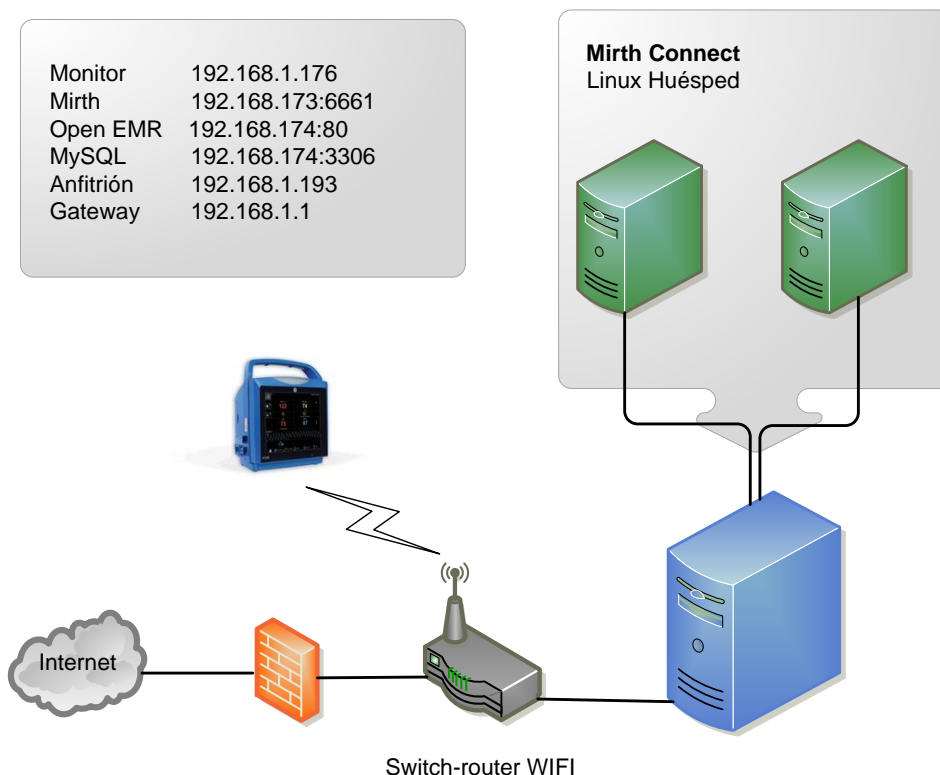


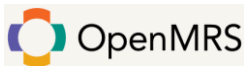
Figura 22. Esquema de conectividad y red del entorno de pruebas³³

³³ Fuente: Diseñado y elaboración propia

3.3 La elección del software

Las soluciones de software “Open Source” disponibles son limitadas. Analizando su implantación en el mercado sanitario en términos de prestaciones, comunidades y foros activos que las promueven y que ayudan a mantener un desarrollo continuo, evolutivo, las alternativas encontradas más interesantes que pueden considerarse son las siguientes:

- Sistema de gestión de la Historia Clínica de Pacientes: OpenMRS³⁴, GnuHealth³⁵ y OpenEMR³⁶ [10]
- Motores de integración: Open eHealth Integration Platform (IPF)³⁷ y Mirth Connect³⁸ [11]



La solución que nació en Kenia para dar respuesta a las necesidades en entornos con falta de recursos. Programado con Java funciona sobre PostgreSQL y MySQL, da respuesta a las necesidades principales de una organización sanitaria a través de la integración con estándares abiertos (HL7, DICOM, etc.). Actualmente está implementado en muchos países de África, Latinoamérica y zonas del sudeste asiático.



Otra alternativa totalmente gratuita y libre que ofrece respuesta a cualquier tipo de organización de la salud, desde clínicas pequeñas hasta sistemas de salud regionales. Desarrollado con Python sobre la base de datos PostgreSQL empezó su desarrollo en Las Palmas de Gran Canaria en el 2008 para dar respuesta al ámbito rural y actualmente se extiende por bastantes países a través de una red de colaboradores que la adaptan al cliente final.



La solución con más años de desarrollo, la más implementada, traducida a 30 idiomas y con una comunidad muy activa. La solución se apoya en una arquitectura LAMP, funcionando sobre el servidor web Apache y la base de datos MySQL.

Su interoperabilidad está basada en la adopción de todos los estándares del sector sanitario y dispone de módulos que cubren prácticamente cualquier departamento del hospital.

³⁴ <http://openmrs.org/>, [en línea, 29 de noviembre, 2016]

³⁵ <http://health.gnu.org/es/index.html>, [en línea, 29 de noviembre, 2016]

³⁶ <http://www.open-emr.org/>, [en línea, 29 de noviembre, 2016]

³⁷ <http://oehf.github.io/ipf/>, [en línea, 29 de noviembre, 2016]

³⁸ <https://www.mirth.com/>, [en línea, 29 de noviembre, 2016]

El proyecto está cerca de conseguir la certificación ONC de la agencia de certificación del gobierno de EEUU para las tecnologías de la información en el ámbito de la salud.

Actualmente está soportada por OEMR³⁹, organización sin ánimo de lucro cuyo objetivo es garantizar la disponibilidad gratuita de un sistema de registro de salud electrónico y los servicios asociados.



Open eHealth

La propuesta se posiciona como un software que puede ser integrado en cualquier aplicación Java para crear interfaces entre aplicaciones del entorno de la salud. Construido sobre el motor mediador y de enrutamiento Apache Camel, su dominio abarca los perfiles de integración IHE, Continua, HL7v2 y HL7v3, librerías para la gestión de documentos clínicos (CDA) basados en XML y el futuro marco de estándares FHIR.



Mirth Connect es el motor de integración HL7 multiplataforma del sector de la salud más extendido. De código abierto, garantiza la interoperabilidad a través de la creación de canales de conexión entre aplicaciones, el filtrado y la conversión de los datos para los estándares soportados (HL7v2, HL7v3, DICOM, EDI, XML...). Basado en Java y con conectores disponibles para MySQL, PostgreSQL, Oracle y SQLserver.

El estudio de las prestaciones de las alternativas anteriores y su adaptación a los requisitos de este proyecto define como **mejor alternativa** a la pasarela **Mirth Connect** como intermediario, concentrador o motor de integración entre los monitores de constantes vitales y el sistema de HCE.

OpenEMR es la elección **para la gestión de los pacientes**, especialmente porque su base de datos integra una tabla dedicada a las constantes vitales y por ser la solución más implementada.

³⁹ <http://www.oemr.org/>, [en línea, 29 de noviembre, 2016]

3.3.1 Mirth Connect

El motor de integración permite crear múltiples canales o interfaces para el intercambio de mensajes entre diferentes sistemas y aplicaciones entre los extremos. El diagrama siguiente muestra la arquitectura, el principio de funcionamiento y los componentes que configuran un canal de comunicación [12]:

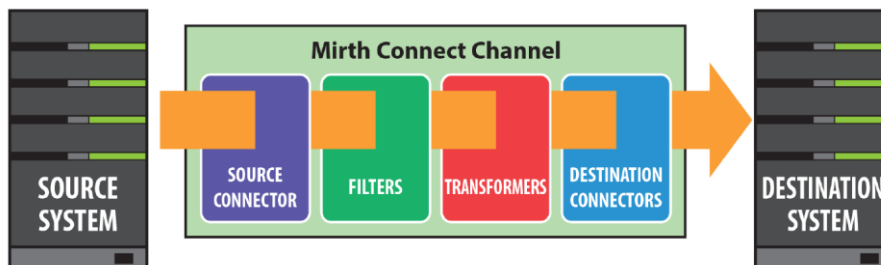


Figura 23. Canal-interface Mirth Connect y sus componentes

Los datos procedentes del sistema fuente se recogen a través de un conector de entrada para procesarlos y entregarlos a un conector de salida que finalmente los enviará al sistema o aplicación de destino.

Los filtros y transformadores son módulos que permiten manipular los mensajes que atraviesan el canal. La manipulación de los mensajes se realiza mediante el mapeo de valores, la creación de variables, el uso de funciones y la potencia que ofrece la creación de guiones utilizando el lenguaje **Java** y **JavaScript**.

Un canal siempre tiene un conector fuente, pero puede enviar mensajes a varios conectores de salida simultáneamente.

En la figura se aprecia cómo un mensaje HL7 enviado por una aplicación es capturado por el canal de entrada mediante el protocolo MLLP y, tras su procesamiento, se entrega a través de tres conectores de salida a diferentes aplicaciones: inserción en una base de datos, generación de un fichero *pdf* y envío de un mensaje mediante correo electrónico [12].



Figura 24. Esquema envío mensaje HL7 a BBDD, PDF y correo electrónico

Los protocolos y métodos de transporte soportados por los canales son múltiples e incluyen MLLP, lectura/escritura de archivos, TCP/IP, FTP, HTTP, JDBC, JMS, PDF, RTF, Web Services, etc.

Nexgen Healthcare, la actual empresa propietaria de Mirth Connect, comercializa extensiones que dotan a la aplicación de más prestaciones: Modo de transmisión ASTM E1381, estándar de datos ASTM E1394, lector de correo electrónico POP3 o IMAP, autenticación LDAP, gestor SSL y conector para puerto serie.

El servidor Mirth Connect necesita una base de datos para almacenar su configuración y los mensajes que transforma y procesa. Los conectores de bases de datos disponibles son Apache Derby (base de datos por defecto integrada en Mirth), PostgreSQL, MySQL, Oracle y SQLServer.

En el caso de que el entorno de producción maneje muchas transacciones, se recomienda instalar una de las bases de datos soportadas por Mirth Connect y dejar Apache Deby para el entorno de pruebas.

Los requisitos para su instalación no son muy exigentes⁴⁰:

- Oracle Java Runtime Environment 1.6 o superior
- Al menos 100 MB de espacio libre en disco
- Sistema operativo: Windows (XP, 7, 10 o Server; 32 o 64 bits); Linux con Kernel 2.4 o superior; Mac OS X (10X)

3.3.2 OpenEMR

EL gestor de la HCE más extendido y traducido a múltiples idiomas integra los módulos fundamentales para la gestión de los pacientes y sus datos clínicos⁴¹:

- Datos demográficos del paciente
- Programación, citas y agenda de pacientes
- Registros médicos: Visitas, datos laboratorio, notas clínicas, informes, vacunas, medicamentos y **registro de constantes vitales**, entre otros.
- Gestión de recetas electrónicas
- Asistente para la ayuda en la toma de decisiones clínicas
- Portal para acceso del paciente a su información médica, etc.

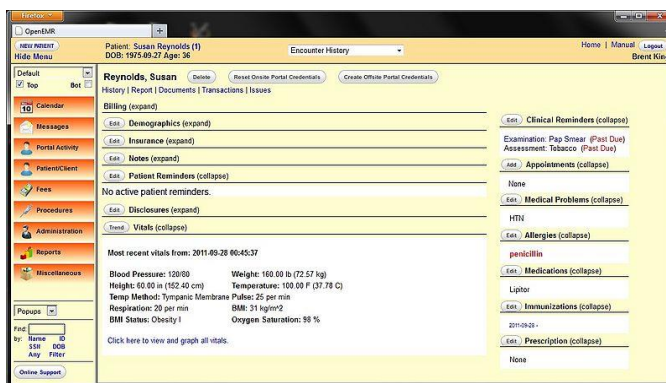


Figura 25. Visualización resumen HCE paciente

⁴⁰ <http://www.mirthcorp.com/community/wiki/display/mirth/System+Requirements>, [en línea, 8 de diciembre, 2016]

⁴¹ http://www.open-emr.org/wiki/index.php/OpenEMR_Features, [en línea, 8 de diciembre, 2016]

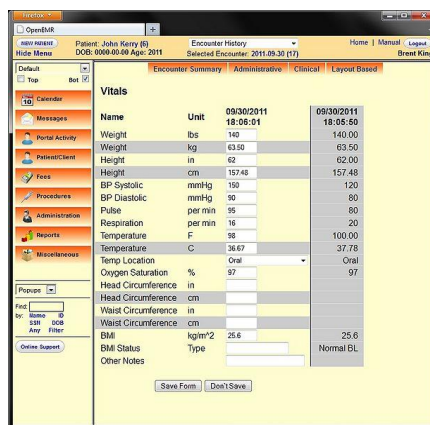


Figura 26. Formulario de constantes vitales

La aplicación de código abierto puede personalizarse para adaptarla a los profesionales y al tipo de hospital.

Los requisitos para la instalación de OpenEMR son⁴²:

- Base de datos MySQL
- Servidor Web o compatible con PHP5
- Sistema operativo Windows o Linux 32 o 64 bits

3.3.3 Software de soporte

HL7 Inspector

Esta herramienta multiplataforma ayuda a agilizar el proceso reiterativo del desarrollo de la integración. El pequeño programa actúa como simulador y emisor de mensajes HL7 y permite reducir el tiempo de parametrización, validación y pruebas de la comunicación relacionadas con los mensajes HL7 entre aplicaciones.⁴³

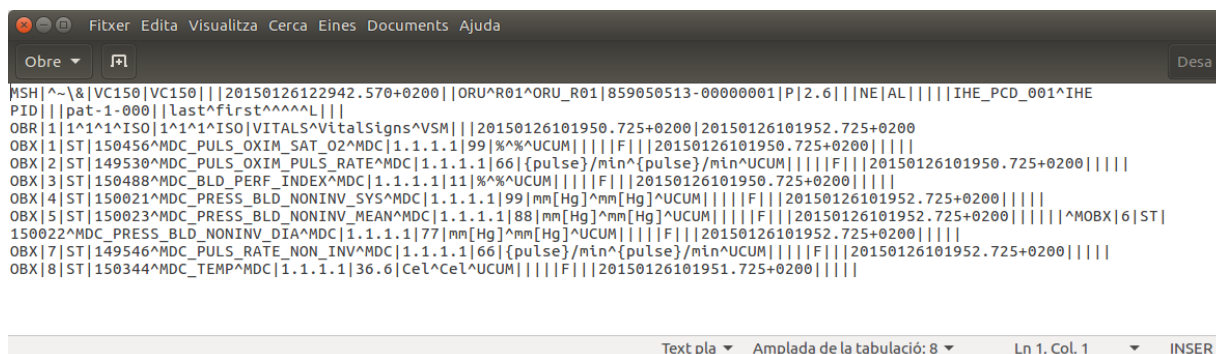


Figura 27. Simulador de mensajes HL7 Inspector

⁴² http://www.open-emr.org/wiki/index.php/OpenEMR_Wiki_Home_Page#Installation_Manuals, [en línea, 8 diciembre, 2016]

⁴³ <https://sourceforge.net/projects/hl7inspector/>, [en línea, 29 de noviembre, 2016]

MySQL Workbench⁴⁴

Esta herramienta multiplataforma desarrollada por Oracle configura un entorno de desarrollo muy potente para la base de datos MySQL. Administra conexiones a la BBDD e integra un potente editor SQL que permite crear, visualizar, interrogar y manipular las BBDD, las tablas que las componen, etc.

MySQL Workbench es de utilidad para analizar y comprender la estructura de la base de datos OpenEMR y cómo insertar las constantes vitales en la HCE.

⁴⁴ <http://mysqlworkbench.org/>. [en línea, 8 de diciembre, 2016]

4.2 El laboratorio, entorno de trabajo

4.2.1 Instalación máquinas virtuales y conectividad

En base al esquema de virtualización mostrado y descrito en el punto 3.2, en primer lugar, se procede a descargar e instalar el software de Oracle VM VirtualBox 5.1⁴⁵ para Windows, creándose las dos máquinas virtuales para sistema operativo Linux, y, posteriormente, descargar las imágenes Linux de Ubuntu Desktop y Ubuntu Server versión 16.4 LTS (Xenial Xerus)⁴⁶ que se instalan siguiendo los asistentes de instalación en las correspondientes máquinas virtuales.

Mediante la consola de ambas máquinas y con el comando *ping* se comprueba que la conectividad entre ambas máquinas y su anfitrión Windows 10 es correcta y funciona, para que tengan visibilidad entre ellas, las tarjetas de red se configuran en modo puente. El esquema de direccionamiento IP es el mostrado en la figura del punto 3.2.

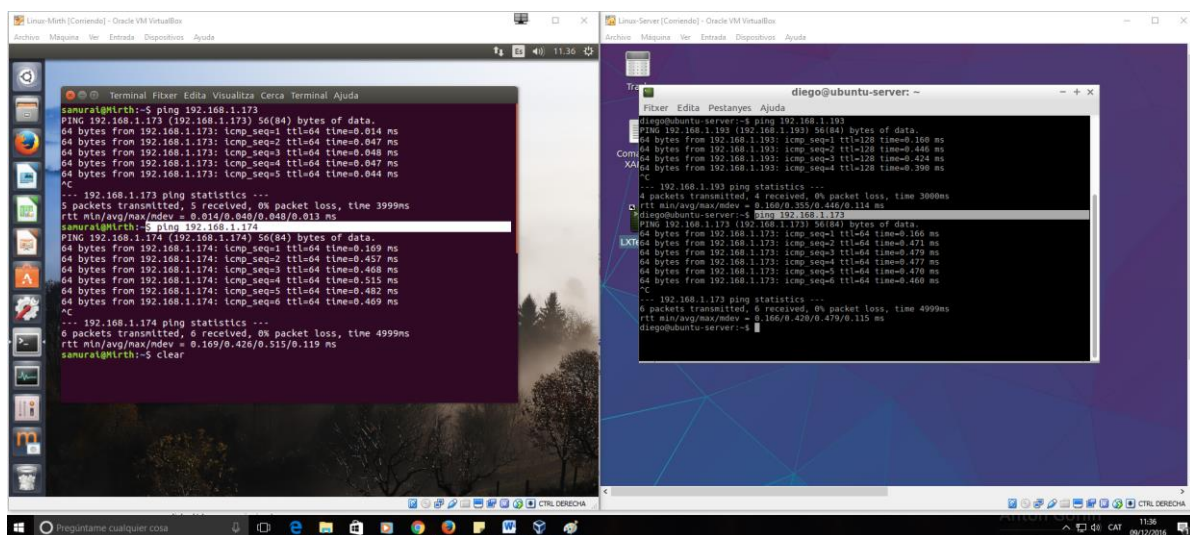


Figura 29. Máquinas virtuales en funcionamiento

4.2.2 Instalación de Mirth Connect

Antes de instalar el motor de integración es necesario instalar el paquete Oracle de Java ya que Mirth Connect se ejecuta mediante Java Web Start, software incluido en Java Runtime Environment (JRE) a partir de la versión Java 5.0. La instalación se realiza fácilmente desde la consola con el comando *sudo apt-get install oracle-java8-installer*

⁴⁵ <https://www.virtualbox.org/wiki/Downloads>, [en línea, 2 de noviembre, 2016]

⁴⁶ <http://releases.ubuntu.com/16.04/>, [en línea, 2 de noviembre, 2016]

A continuación, se descarga Mirth Connect desde la página web oficial, hay que cumplimentar un pequeño cuestionario, y se instala desde la consola. ⁴⁷

El paquete Mirth Connect instala cuatro aplicaciones:

- Un servidor que integra la base de datos Apache Derby
- La aplicación para el Administrador del servidor Mirth Connect
- Una consola para la gestión mediante la línea de comandos
- La aplicación para el Administrador de Mirth Connect

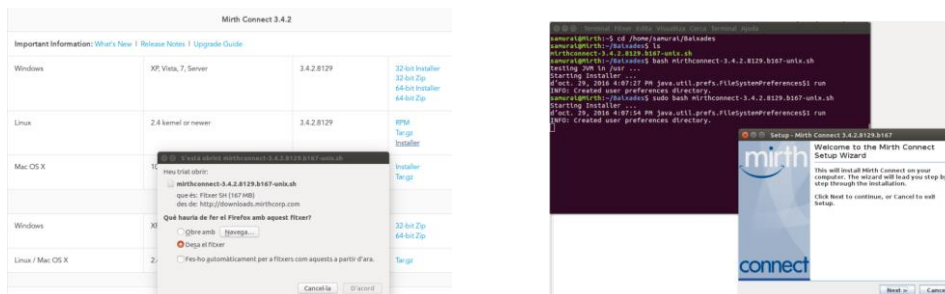


Figura 30. Descarga e instalación de Mirth Connect para Linux

Siguiendo los pasos del asistente se llega al final de la instalación y se ejecuta el administrador del servidor de Mirth.

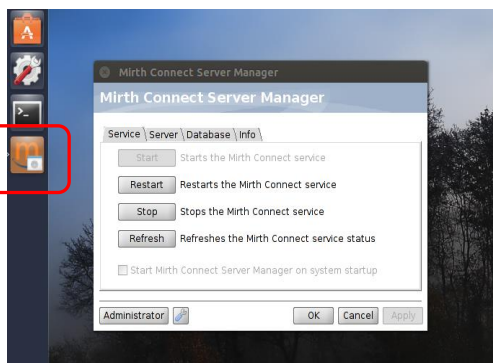


Figura 31. Administrador del servidor Mirth Connect

El administrador del servidor Mirth gestiona la puesta en marcha/parada del servicio asociado al servidor y la configuración general de puertos y base de datos que usa.

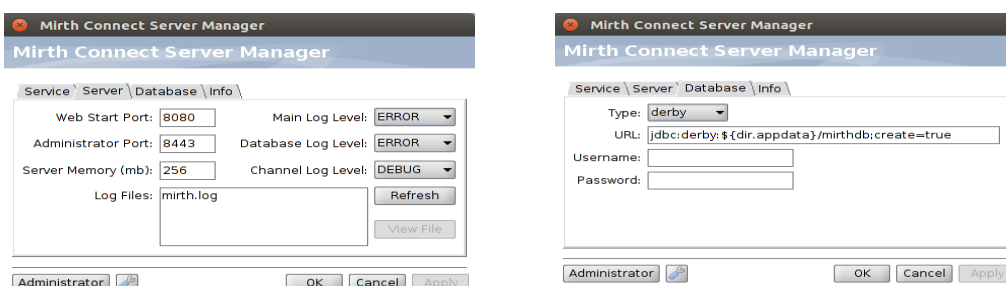


Figura 32. Puertos, base de datos, conexión y credenciales

⁴⁷ <https://www.mirth.com/>, [en línea, 2 de noviembre, 2016]

Desde el botón “Administrador” se lanza y ejecuta Java Web Start, luego “Run” y aparece el acceso al administrador de Mirth Connect con credenciales por defecto: **admin, admin**.

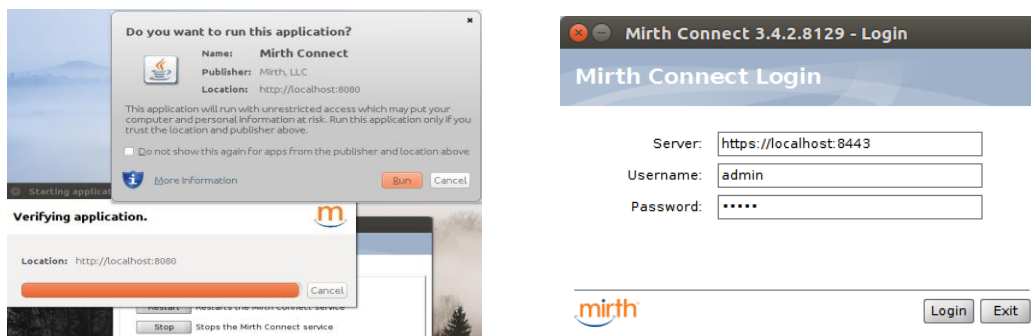


Figura 33. Pantalla acceso local al administrador de Mirth Connect

La primera vez que se accede al administrador de Mirth Connect aparece un formulario que cada usuario administrador tiene que cumplimentar.

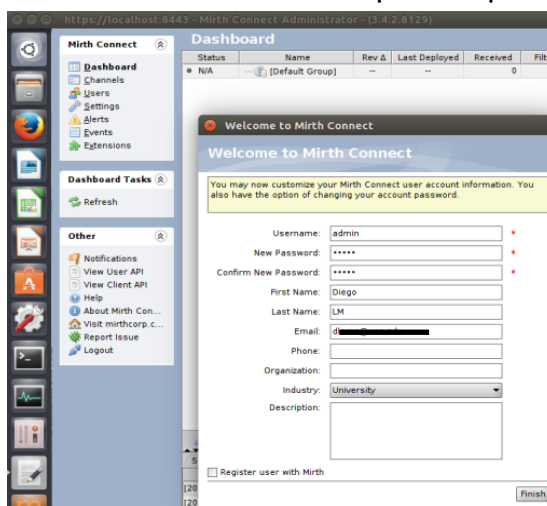


Figura 34. Formulario registro administrador Mirth Connect

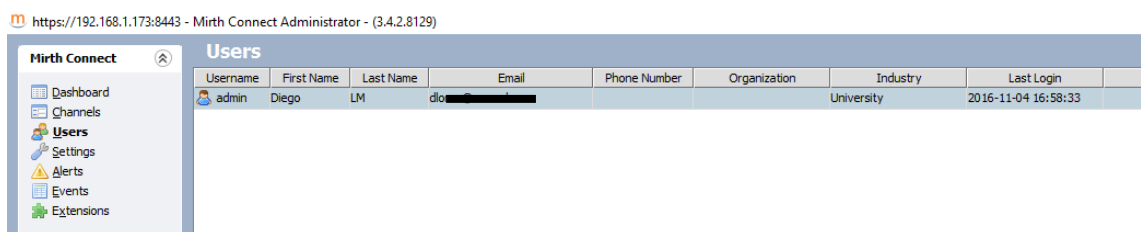


Figura 35. Administrador “Diego LM” conectado a Mirth Connect

Alternativamente, se puede accederse al administrador desde cualquier navegador de internet apuntando a la dirección IP del servidor y puerto 8080 que lanzará la aplicación Java Web Start:

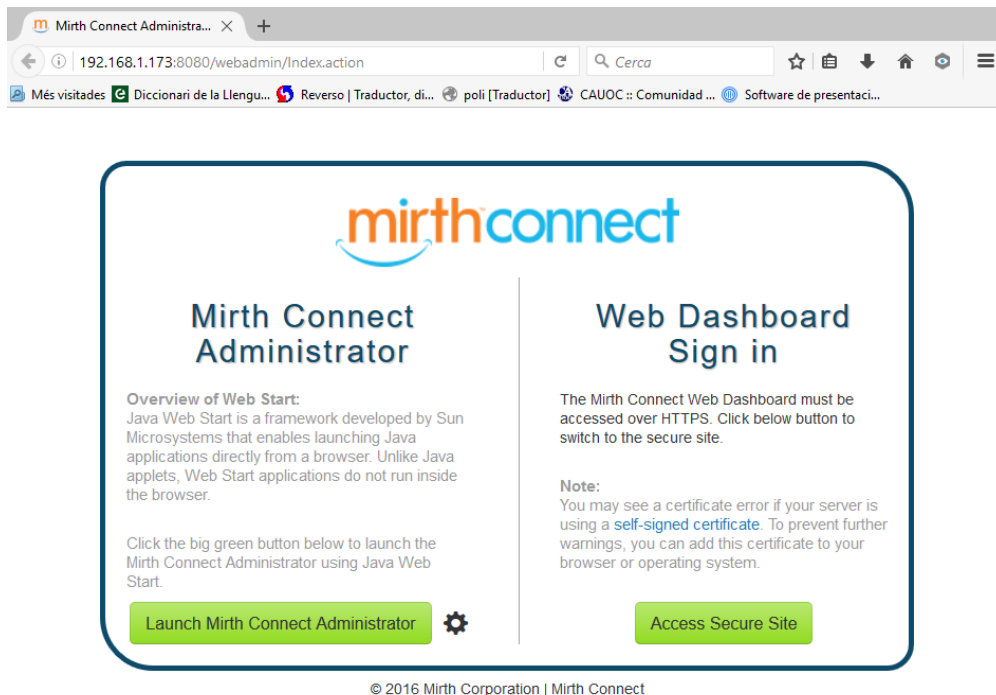


Figura 36. Pantalla acceso remoto al administrador de Mirth Connect

4.2.3 Instalación de OpenEMR

OpenEMR funciona sobre un servidor web compatible con PHP5 y la base de datos MySQL. El instalador XAMPP de Bitnami agiliza y simplifica la instalación de todos los componentes necesarios para una correcta instalación. La versión Linux de 64 bits es la que se adapta a los requisitos del escenario de este TFG⁴⁸.

XAMPP se instala en la máquina Linux-Server mediante un sencillo proceso:

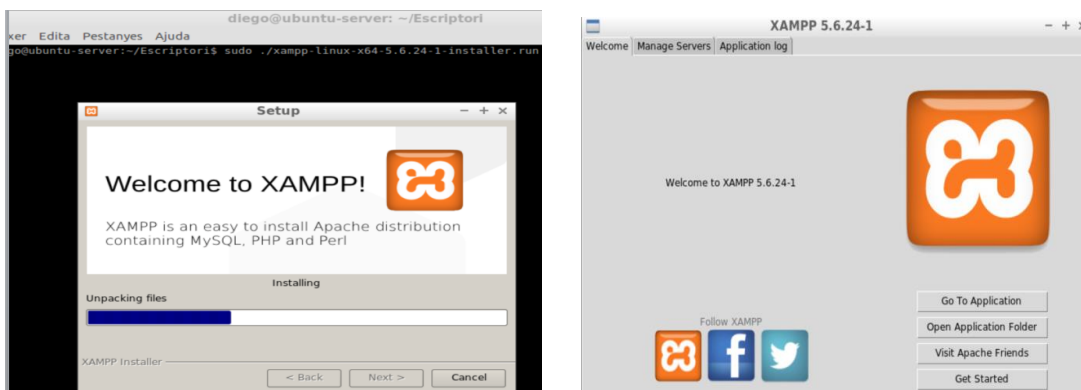


Figura 37. Instalación de XAMPP

⁴⁸ <https://www.apachefriends.org/download.html>, [en línea, 2 de noviembre, 2016]

Se comprueba que XAMPP funciona - **localhost** en el navegador, el servidor web **Apache** está encendido y funcionando.

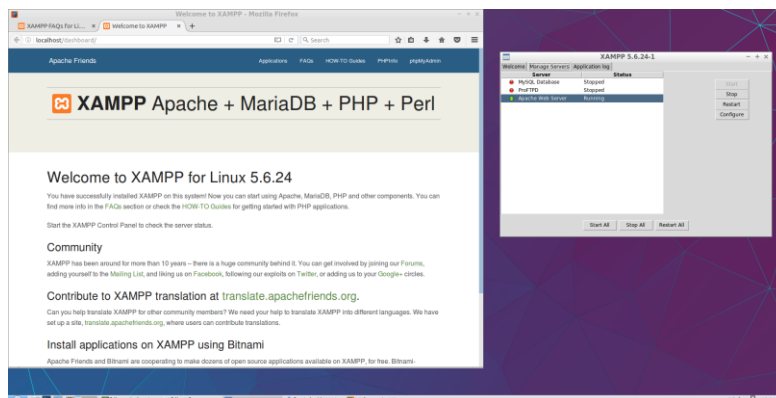


Figura 38. XAMPP en funcionamiento

Desde la consola de Linux también puede gestionarse LAMP mediante los siguientes comandos:

- Iniciar los servicios XAMPP `sudo /opt/lampp/lampp start`
- Detener los servicios XAMPP `sudo /opt/lampp/lampp stop`

Una vez instalado LAMP, se descarga OpenEMR 4.2.2 en su versión Linux de la página oficial; se instala siguiendo los pasos del instalador⁴⁹:

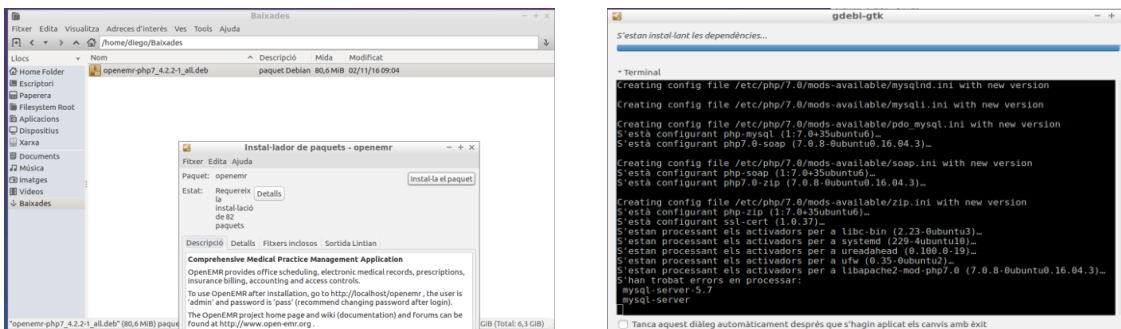


Figura 39. Instalación de OpenEMR

La contraseña elegida, tanto para el servidor MySQL como para el usuario de OpenEMR, es 1005.

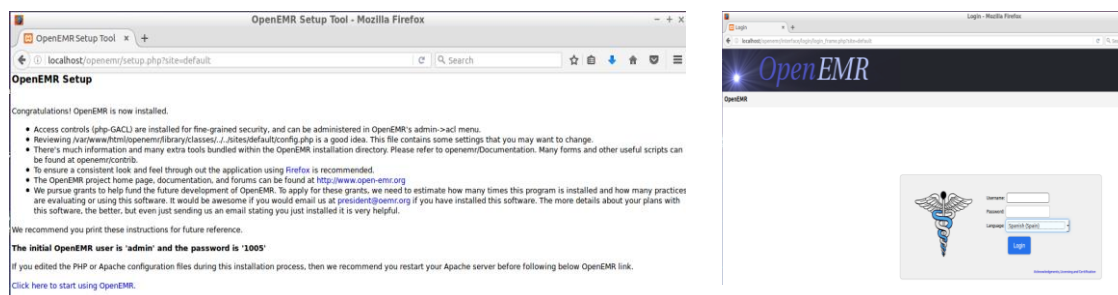


Figura 40. Finalización instalación y acceso OpenEMR

⁴⁹ http://www.open-emr.org/wiki/index.php/OpenEMR_4.2.2_Linux_Installation, [en línea, 2 de noviembre, 2016]

Tras acceder desde la barra del navegador con **localhost/openemr**, se introducen los credenciales definidos durante el proceso de instalación para entrar en OpenEMR: usuario: **admin**, password **1005**.

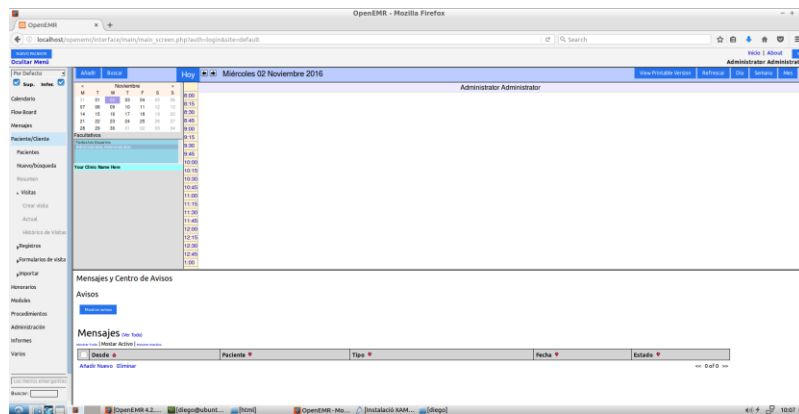


Figura 41. Pantalla de OpenEMR

Para analizar y gestionara mejor la BBDD de OpenEMR, se **instala Oracle MySQL Workbench**, un entorno de gestión gráfico muy potente para MySQL.

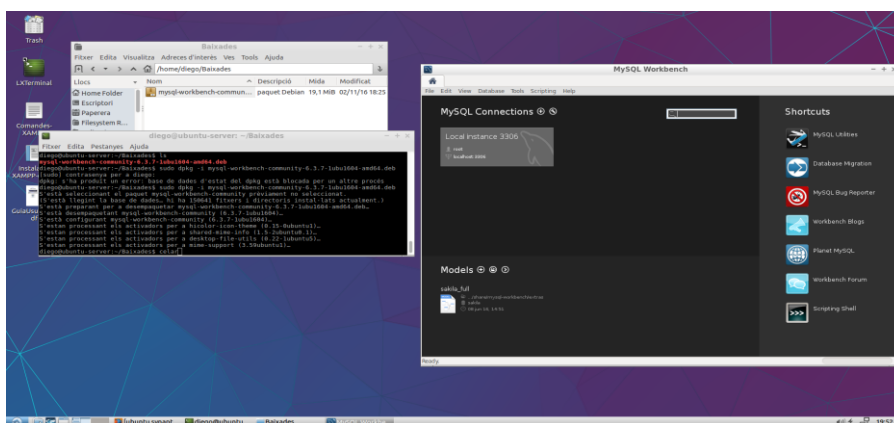


Figura 42. Pantalla de MySQL Workbench

Se establece una conexión local a la BBDD “openemr” para ver sus tablas.

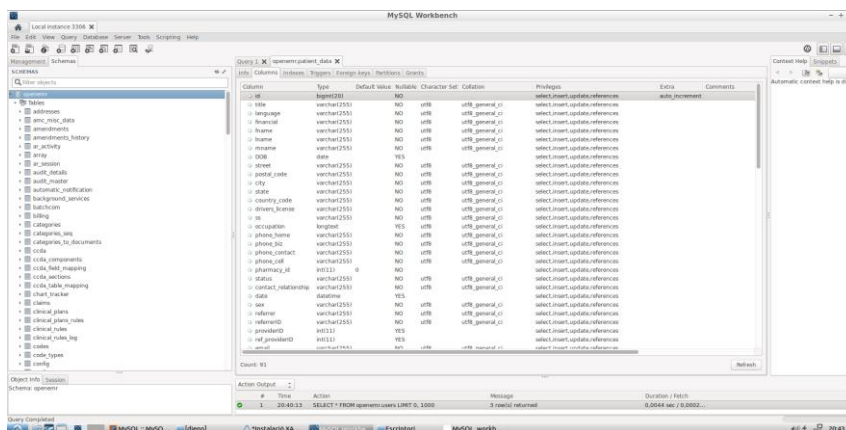


Figura 43. Captura de la tabla BBDD del paciente en OpenEMR

Por defecto MySQL únicamente permite el acceso desde el equipo local donde la BBDD ha sido instalada.

El motor de integración Mirth se encuentra instalado en una máquina virtual externa y, por tanto, es necesario habilitar la conexión a MySQL para equipo remotos mediante la edición del fichero de configuración *mysqld.cnf*, modificando la variable *bind-address= 127.0.0.1* por la de la máquina remota; en nuestro caso se permite la conexión desde cualquier dirección IP (0.0.0.0). En la consola se edita el fichero con **sudo nano /etc/mysql/ mysqld.cnf**

```
*<mysqld.cnf>
Fixber Editar Cercar Opcions Ajut
# It has been reported that passwords should be enclosed with ticks/quotes
# especially if they contain '#' chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

[mysqld_safe]
socket = /var/run/mysqld/mysqld.sock
nice = 0

[mysqld]
#
# * Basic Settings
#
user = mysql
pid-file = /var/run/mysqld/mysqld.pid
socket = /var/run/mysqld/mysqld.sock
port = 3306
basedir = /usr
datadir = /var/lib/mysql
tmpdir = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address = 0.0.0.0
#
# * Fine Tuning
#
key_buffer_size = 16M
```

Figura 44. Edición fichero “mysqld.cnf”

Se crea el usuario “mirth” y se le dan privilegios de acceso a MySQL:

```
diego@ubuntu-server:/$ mysql -u root -p
```

```
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 5.7.16-0ubuntu0.16.04.1 (Ubuntu)
```

```
mysql> CREATE USER mirth IDENTIFIED BY '1005';
Query OK, 0 rows affected (0,04 sec)
```

```
mysql> GRANT ALL PRIVILEGES ON *.* TO mirth@%' IDENTIFIED BY '1005';
Query OK, 0 rows affected, 1 warning (0,01 sec)
```

Se reinicia el servicio con *sudo service mysql restart*

En la máquina virtual remota Linux-mirth se instala un cliente de MySQL y se comprueba que la conexión remota a MySQL puede establecerse.

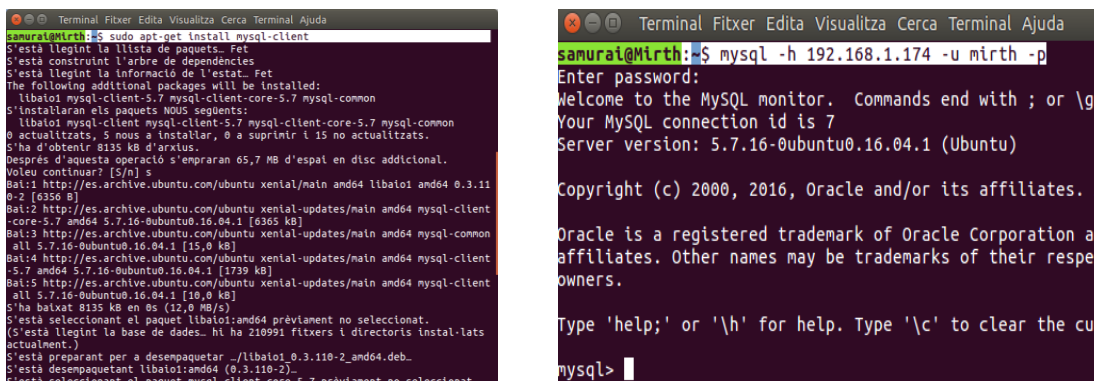


Figura 45. Conexión remota a MySQL

Desde la máquina local donde está instalado OpenEMR/MySQL puede comprobarse que la conexión del usuario *mirth* se ha establecido correctamente.

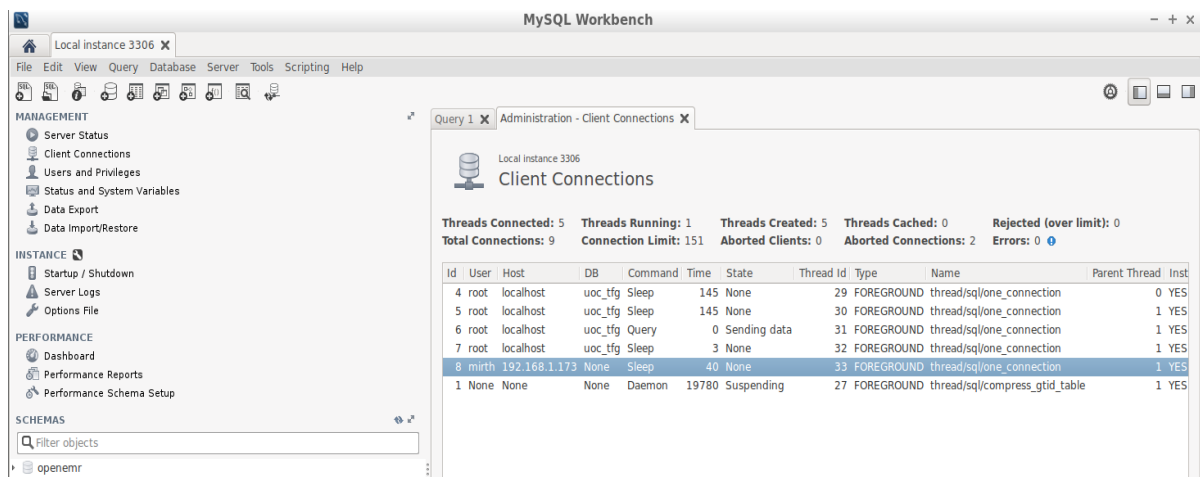


Figura 46. Usuario conectado a BBDD de OpenEMR

OpenEMR se ha instalado correctamente y su base de datos es accesible desde equipos externos, en el caso de este laboratorio, desde Mirth Connect.

4.3 Mirth Connect, primer contacto

La comprensión y el manejo de las diferentes funcionalidades de Mirth Connect se adquieren mediante la configuración y el desarrollo de un sencillo caso en el que el motor de integración recibirá un mensaje HL7 y lo escribirá en un fichero. Existe una buena descripción en el blog del grupo HL7 español [13].

4.3.1 Configuración del conector de entrada de la interface

Para el caso práctico y real de conexión con el monitor de signos vitales se crea un canal/interface específica que permitirá leer el mensaje procedente del monitor a través de la conexión WIFI y escribir los valores de las constantes vitales del paciente en la base de datos de Open EMR.

El canal se ha denominado “*Monitor_Constantes_Vitales_a_OpenEMR*”

El conector del canal de entrada es del tipo **escucha TCP/IP**. No obstante, y dado que el mensaje HL7 perfil IHE-PCD-01 que envía el monitor VC150 lleva mucha información, es necesario activar y configurar otros módulos de la interface, concretamente algún filtro y transformadores para el conector de entrada y el conector de salida del canal-interface.

Asistente de configuración

Mirth Connect integra herramientas que asisten en el tratamiento y la manipulación de los mensajes para agilizar su configuración. Las siguientes capturas reflejan su uso y cómo asisten al programador en su tarea de configuración.

En la pestaña “Message Templates” se pueden copiar mensajes para facilitar la extracción de datos y el mapeo de variables.

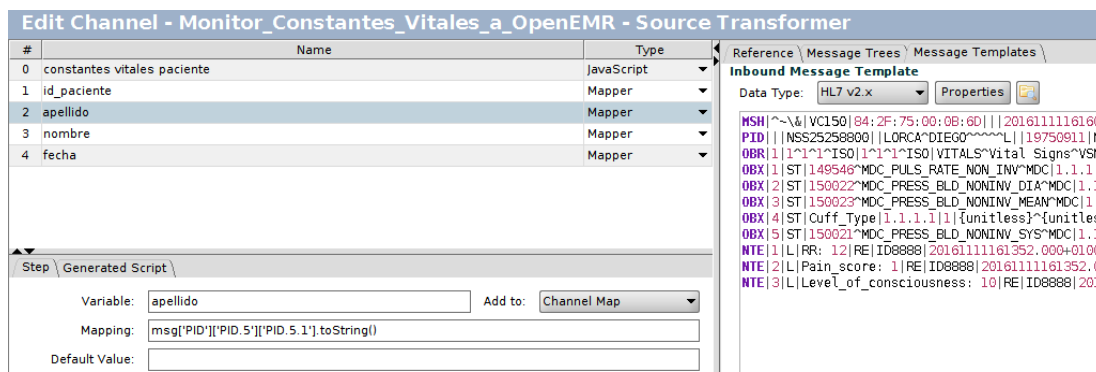


Figura 47. Asistente configuración canales de Mirth Connect

La pestaña “Message Trees” muestra la estructura jerárquica en forma de árbol del mensaje. Simplemente arrastrando el campo de interés se genera el código que se mapea/asocia a la variable que se define, en este caso el *apellido*.

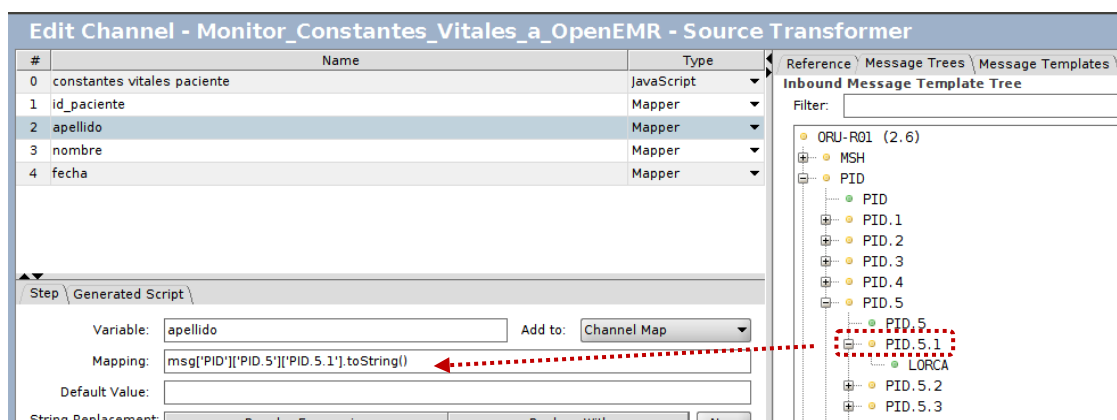


Figura 48. Estructura en árbol mensaje HL7 asistente Mirth Connect

Configuración filtro del mensaje

Se crea un filtro sobre el canal de entrada/fuente de datos para asegurar que el mensaje recibido es el adecuado y el paciente está identificado:

El mensaje HL7 recibido debe cumplir dos condiciones para ser procesado:

- En el componente MSH.21.1 debe aparecer el término “IHE_PCD_001”
- El campo/componente PID.3.1 debe llevar información y no estar vacío

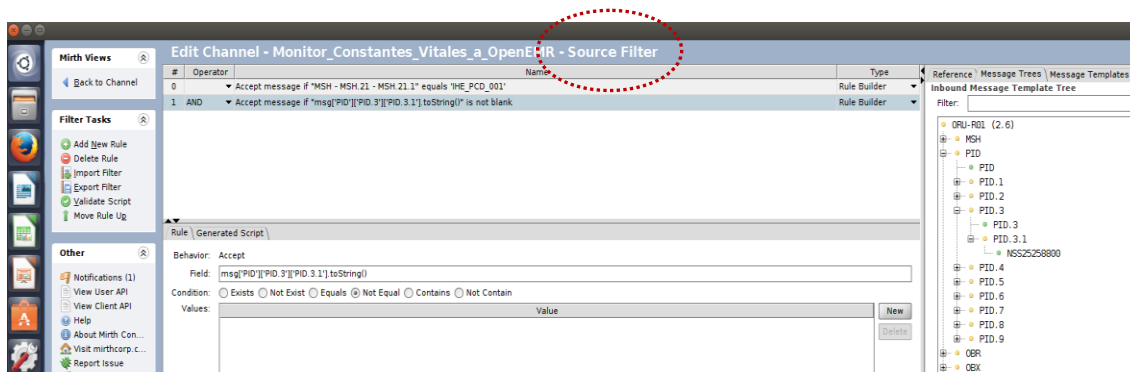


Figura 49. Configuración filtro conector entrada canal Mirth Connect

Estas condiciones garantizan que el mensaje se ha generado por un monitor de constantes vitales que integra el perfil IHE PCD-01 y que en el mensaje recibido el paciente está identificado.

Configuración de los transformadores del mensaje

El transformador es una herramienta que permite extraer campos concretos del mensaje de entrada para mapearlo, manipularlos y convertirlos a otros formatos.

En el escenario de este TFG se configuran varios transformadores en el conector de fuente del canal para extraer algunos componentes del mensaje HL7 de entrada y generar unas variables asociadas al canal que incluirán los valores mapeados. Las variables y sus valores pueden ser usadas para posteriores procesamientos, en nuestro caso para configurar el mensaje de salida hacia OpenEMR.

A continuación se muestra la creación de cuatro (4) transformadores. Tres de ellos, “id_paciente”, “apellido” y “nombre” son del tipo “Mapper” o de conversión directa, el quinto, “constantes_vitales_paciente” es del tipo “JavaScript”, un pequeño programa que dota a Mirth de una herramienta muy potente para tratamientos avanzados de los mensajes que circulan por la interface.

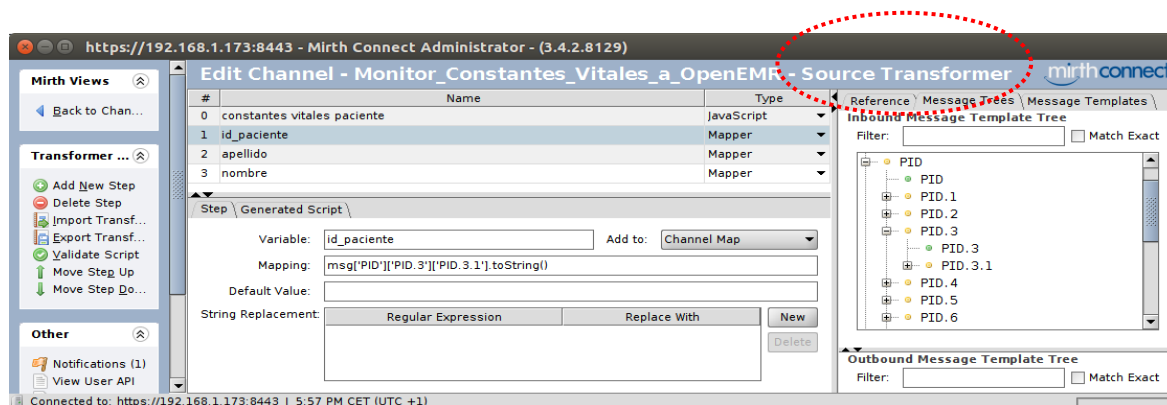


Figura 50. Configuración transformador canal Mirth Connect

El JavaScript “*constantes_vitales_paciente*” se ha creado para iterar sobre todos los segmentos del mensaje HL7 y extraer de los segmentos OBX los campos de interés correspondientes a las constantes vitales del paciente. Los campos extraídos pasan a ser las siguientes variables del canal: “*PNI-sistolica*”, “*PNI-diastolica*”, “*PNI-media*”, “*Fpulso*”, “*SpO2*” y “*Temperatura*”. JavaScript completo en el Anexo I.

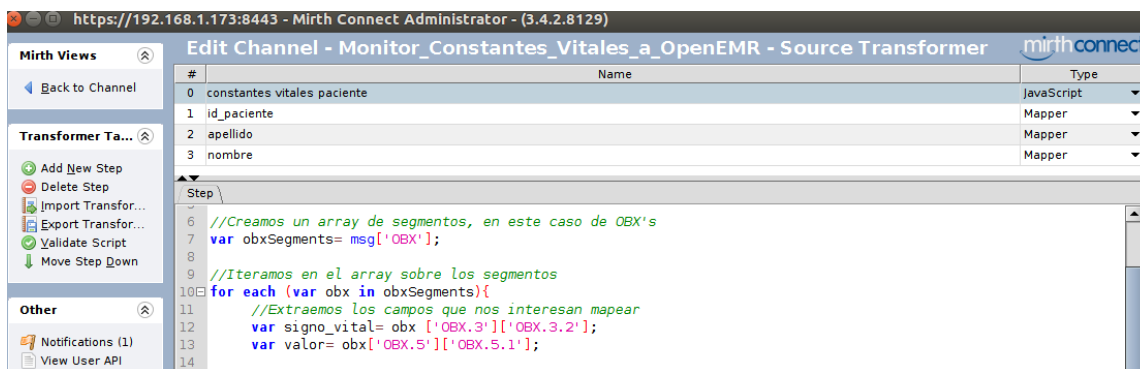


Figura 51. JavaScript transformador

4.3.2 Configuración del conector de salida de la interface

El conector de salida será el encargado de insertar en la BBDD de OpenEMR los valores de las constantes recibas desde el monitor por el conector fuente.

La salida se configura como “**escritor base de datos**” mediante la selección del driver **mysql**, con los parámetros que apuntan al servidor de OpenSource y sus credenciales de acceso a la base de datos **mysql**, destacados con el círculo rojo.

En la parte inferior, marcadas en círculo verde, aparecen las variables de la interface que se crearon durante la configuración del conector de entrada.

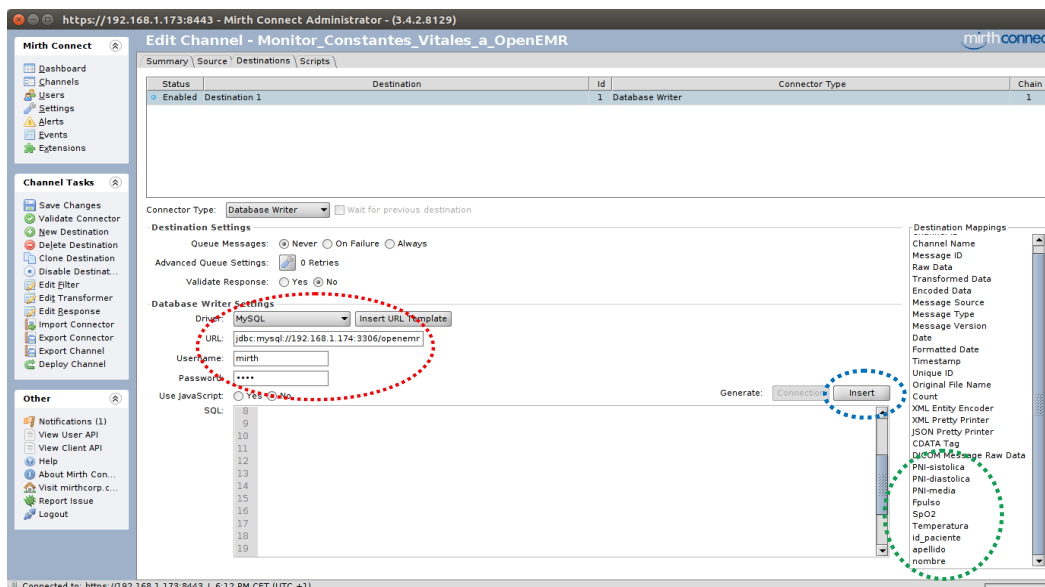


Figura 52. Configuración conector salida canal Mirth Connect

La sentencia SQL de inserción en mySQL se realiza con la ayuda que proporciona Mirth. Se pulsa “Insert” y en la ventana emergente seleccionamos “Get Tables”. Aparecen todas las tablas de la base de datos de OpenEMR, prueba de que la conexión se ha realizado correctamente; se localiza la tabla “form_vitals”, la destinada a guardar las constantes vitales del paciente.

Pulsando “Generate” se crea la sentencia INSERT INTO; los valores serán los que proporcionan las variables creadas anteriormente.

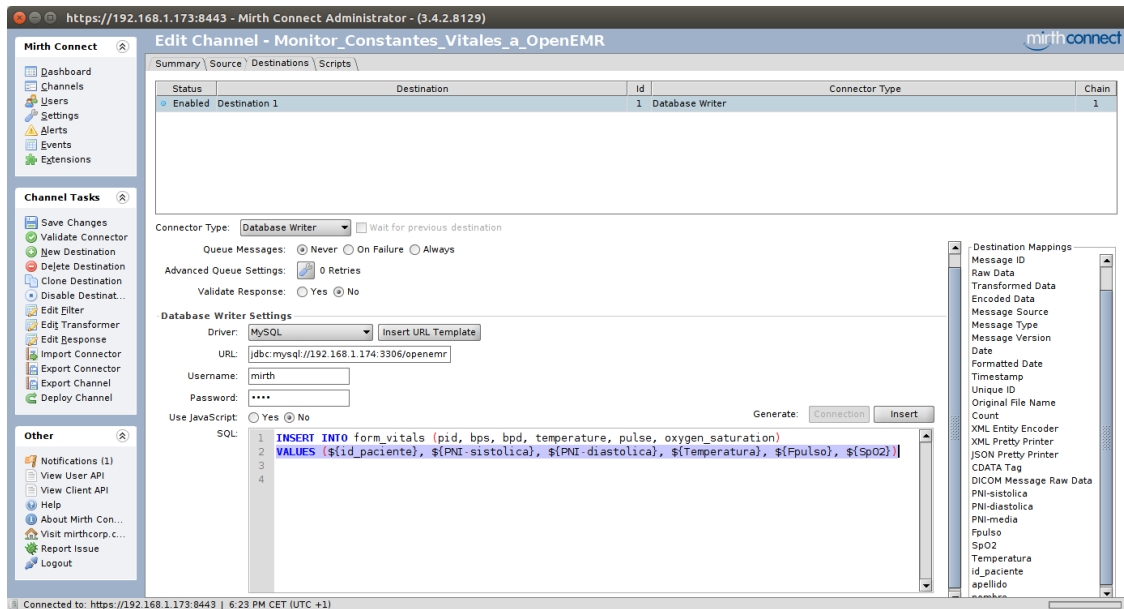


Figura 53. Sentencia SQL conector salida canal Mirth Connect

A continuación, se activa el canal “Monitor_Constantes_Vitales_a_OpenEMR” y se despliega en el servidor Mirth:

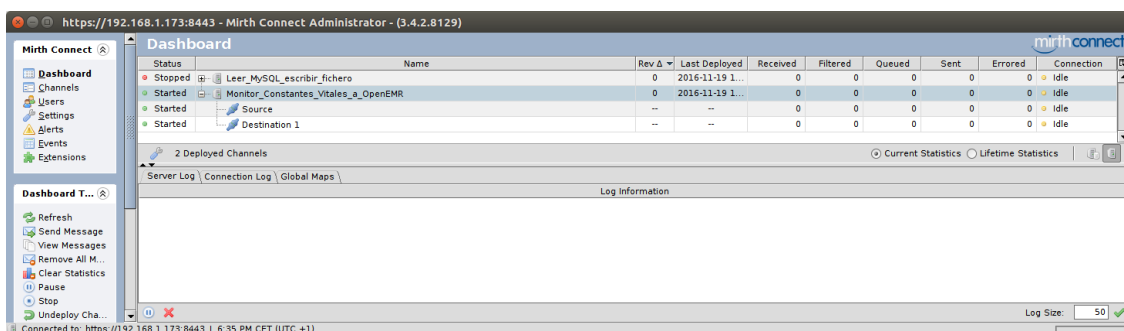


Figura 54. Despliegue canal Mirth Connect

En este momento todo el entorno de pruebas está configurado y listo para tomar las constantes vitales con el monitor VC150 y enviarlas a OpemEMR.

4.3.3 Mediciones constantes vitales y envío a OpenEMR

Para validar el correcto funcionamiento, se replica el caso de uso descrito en el punto 3.1 siguiendo los pasos protocolizados:

1. El paciente se admite en OpenEMR con su número de Historia Clínica o N° de S.S. En nuestro caso se dan de alta dos pacientes que permitirán realizar las pruebas.

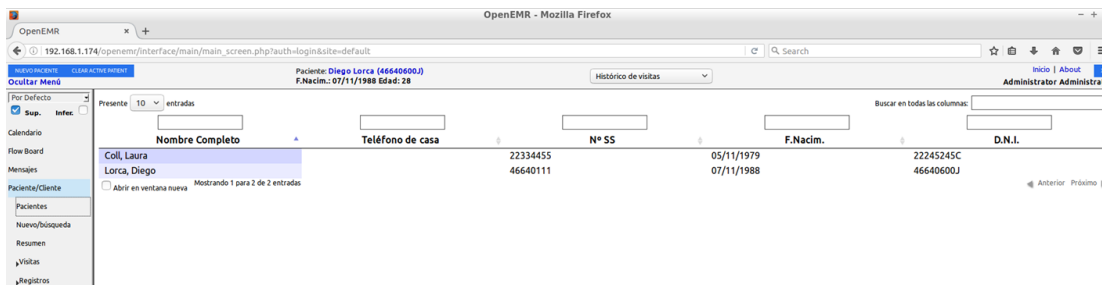


Figura 55. OpenEMR – alta de pacientes

Abrimos una visita y registramos manualmente información, entre ella las constantes vitales a través del formulario de gestión de pacientes de OpenEMR:

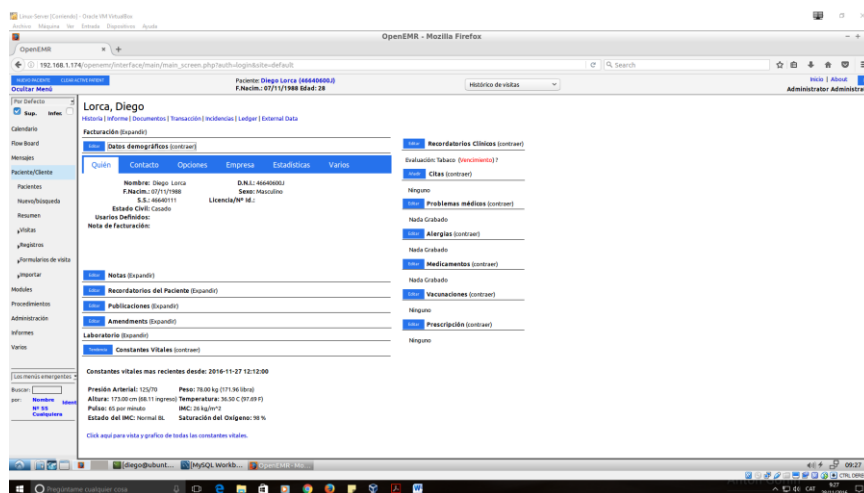


Figura 56. OpenEMR - Registro manual constantes vitales paciente “Diego”

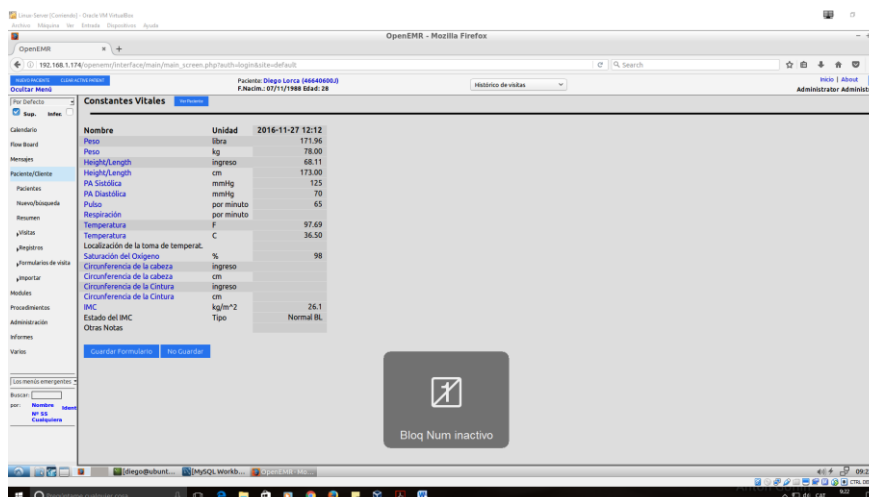


Figura 57. OpenEMR - Registro constantes en tendencias tabulares de “Diego”

- Se imprime el código de barras de la pulsera que identifica al paciente con su número de HCE, el identificador inequívoco y más importante del paciente



- Se identifica al paciente en el monitor⁵⁰ con la ayuda del lector de código de barras

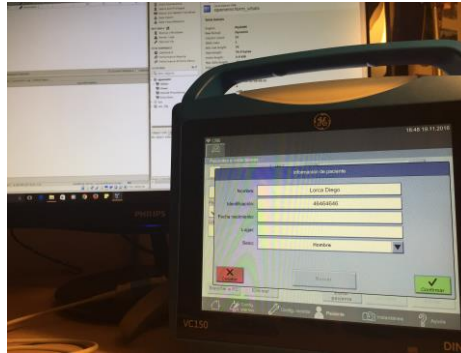


Figura 58. Identificación paciente en monitor constantes vitales

- Se procede a la toma las constantes vitales



Figura 59. Toma de constantes vitales en el monitor

- El monitor a través de Mirth envía las mediciones a OpenEMR

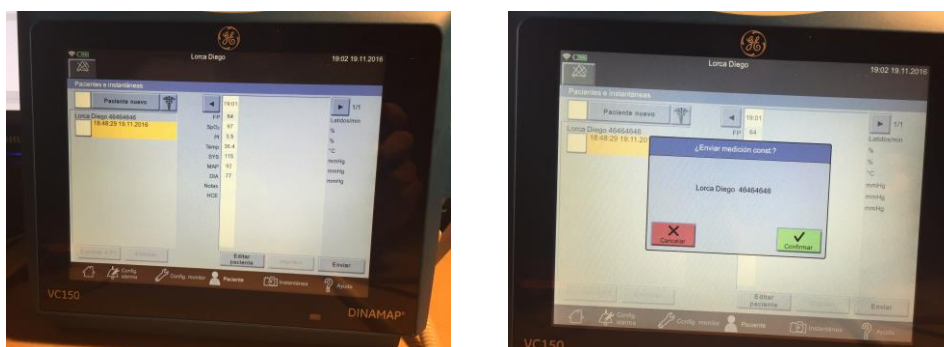


Figura 60. Almacenamiento mediciones, validación y envío a la HCE

⁵⁰ Monitor marca GE Healthcare modelo CARESCAPE VC150; http://www3.gehealthcare.com/en/products/categories/patient_monitoring/patient_monitors/carescape_vc150 [en línea, 29 de noviembre, 2016]

Recepción por el conector de entrada de la interface

La captura siguiente muestra el mensaje recibido:

Figura 61. Captura mensaje HL7 en Mirth Connect

El mensaje HL7 enviado se almacena en Mirth Connect y su análisis permite comentar algunos aspectos de la integración.

```
MSH|^|^&|VC150|84:2F:75:00:0B:6D|||20161128160430.057+0100||ORU^R01^ORU_R01|1182112391-0000001|P|2.6|||NE|AL|||IHE_PCD_001^IHE...
PID||4664011||Lorca^Diego^^^^L||M|...
OBR|1|1^1^1^ISO|1^1^1^ISO|VITALS^Vital Signs^VSM||20161119155941.000+0100|20161119155941.000+0100|...
OBX|1|ST|149546^MDC_PULS_RATE_NON_INV^MDC|1.1.1.1|87|{pulse}/min^{pulse}/min^UCUM|...
OBX|2|ST|150456^MDC_PULS_OXIM_SAT_O2^MDC|1.1.1.1|96|%%^UCUM|||F||20161119155941.000+0100|...
OBX|3|ST|150022^MDC_PRESS_BLD_NONINV_DIA^MDC|1.1.1.1|70|mm[Hg]^mm[Hg]^UCUM|||F||...
OBX|4|ST|150023^MDC_PRESS_BLD_NONINV_MEAN^MDC|1.1.1.1|89|mm[Hg]^mm[Hg]^UCUM|||F||...
OBX|5|ST|150344^MDC_TEMP^MDC|1.1.1.1|36.2|Cel^Cel^UCUM|||F||20161119155941.000+0100|...
OBX|6|ST|Cuff_Type|1.1.1.1|1|{unitless}^{unitless}^UCUM|||F||20161119155941.000+0100|...
OBX|7|ST|150488^MDC_BLD_PERF_INDEX^MDC|1.1.1.1|6|%%^UCUM|||F||20161119155941.000+0100|...
OBX|8|ST|149530^MDC_PULS_OXIM_PULS_RATE^MDC|1.1.1.1|88|{pulse}/min^{pulse}/min^UCUM|||F||...
OBX|9|ST|150021^MDC_PRESS_BLD_NONINV_SYS^MDC|1.1.1.1|122|mm[Hg]^mm[Hg]^UCUM|||F||...
NTE|1||RR: 12|RE|555666|20161119155941.000+0100...
NTE|2||Pain_score: 1|RE|555666|20161119155941.000+0100...
NTE|3||Free_Text_1: Estudiante concentrado por TFG - Informatica UOC|RE|555666|...
```

Figura 62. Segmentos y nomenclatura ISO/IEEE del mensaje HL7

Sobre fondo verde se puede apreciar que los resultados corresponden a un mensaje HL7 adaptado al perfil IHE PCD-01.

El fondo azul destaca el uso semántico de la nomenclatura ISO/IEEE 11073-10101 y la terminología armonizada hRTM.

Destacan sobre el fondo amarillo los valores de las constantes enviadas a OpenEMR.

Envío por el conector de salida de la interface

La captura muestra el envío y la correcta escritura en la base de datos de OpenEMR:

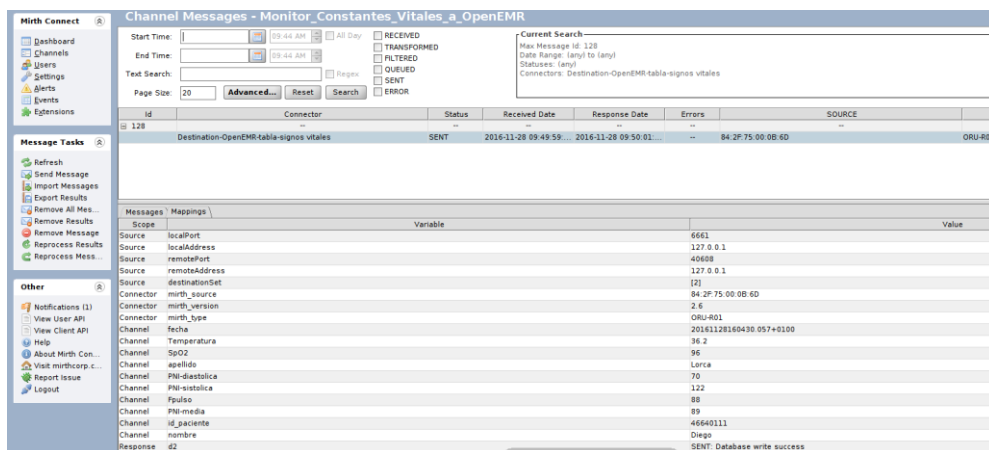


Figura 63. Envío mensaje HL7 y escritura en OpenEMR

Se realiza una consulta a MySQL y se observa que **el registro se ha grabado correctamente en la tabla “form_vitals”**, sin embargo, **los valores de las nuevas constantes enviadas por el monitor de paciente no aparecen en la interface del usuario de OpenEMR**, en la tabla de tendencias de constantes vitales mostrada anteriormente.

4.4 OpenEMR y tablas BBDD implicadas en el registro

Un análisis más detallado de las tablas MySQL de OpenEMR revela que hay tres tablas que están relacionadas estrechamente para mostrar la información de las constantes vitales del paciente. Las tablas “form_vitals”, “form_encounter” y “forms” tienen que actualizarse para registrar correctamente y visualizar los datos en el interface de OpenEMR[15]:

1. La nueva fila que se crea al insertar las constantes en “form_vitals” tiene la columna “id” con valor autoincrementado
2. La tabla “form_encounter” registra las visitas de los pacientes
3. La tabla “forms” mantiene un registro de todos los formularios asociados a los pacientes y a las visitas; por tanto, tiene que actualizarse con los valores anteriores:
 - ▶ La “id” de la fila correspondiente a las nuevas constantes registradas en “form_vitals”
 - ▶ El número de visita “encounter” almacenado en en la tabla “form_encounter”
 - ▶ El tipo de formulario usado, en nuestro caso se denomina “Vitals”

Las tres tablas tienen en común la identificación del paciente, basada en su “pid”, el número de la HCE.

En base a las conclusiones anteriores es necesario rediseñar la sentencia SQL inicial del conector de salida de Mirth Connect para que dé respuesta a las nuevas necesidades. Estos requerimientos imponen un cambio de estrategia ya que el asistente incluido en Mirth no permite generar varias secuencias SQL en una misma conexión a la BBDD. La solución pasa por programar un guión **JavaScript** que realice todas las operaciones, captura siguiente y detalle en Anexo II.

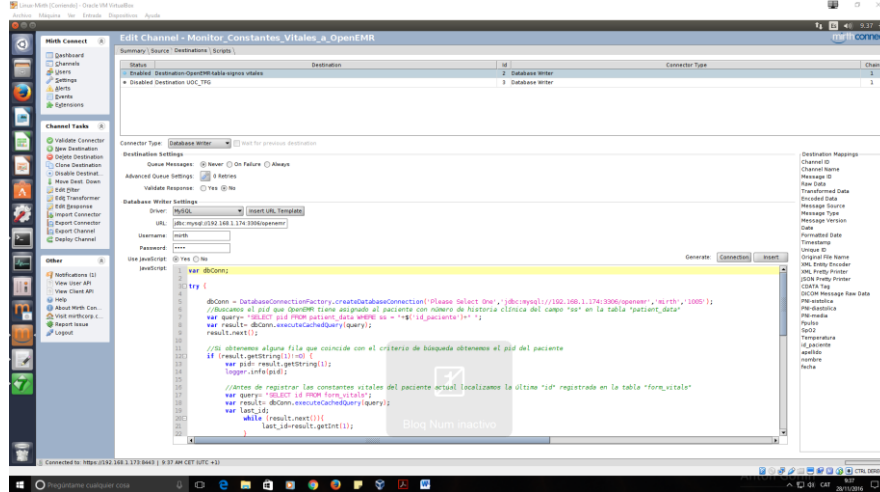


Figura 64. Script SQL conector de salida de Mirth Connect

Nuevamente se registran las contantes con el monitor de paciente siguiendo el procedimiento del epígrafe 4.3.3.

En esta ocasión el registro se realiza en todas las tablas siguiendo las instrucciones del JavaScript, la pantalla de OpenEMR ya muestra correctamente la actualización de la HCE y en la página de tendencias de las constantes se aprecia el nuevo registro, añadido al manual que se realizó durante la primera visita del paciente:

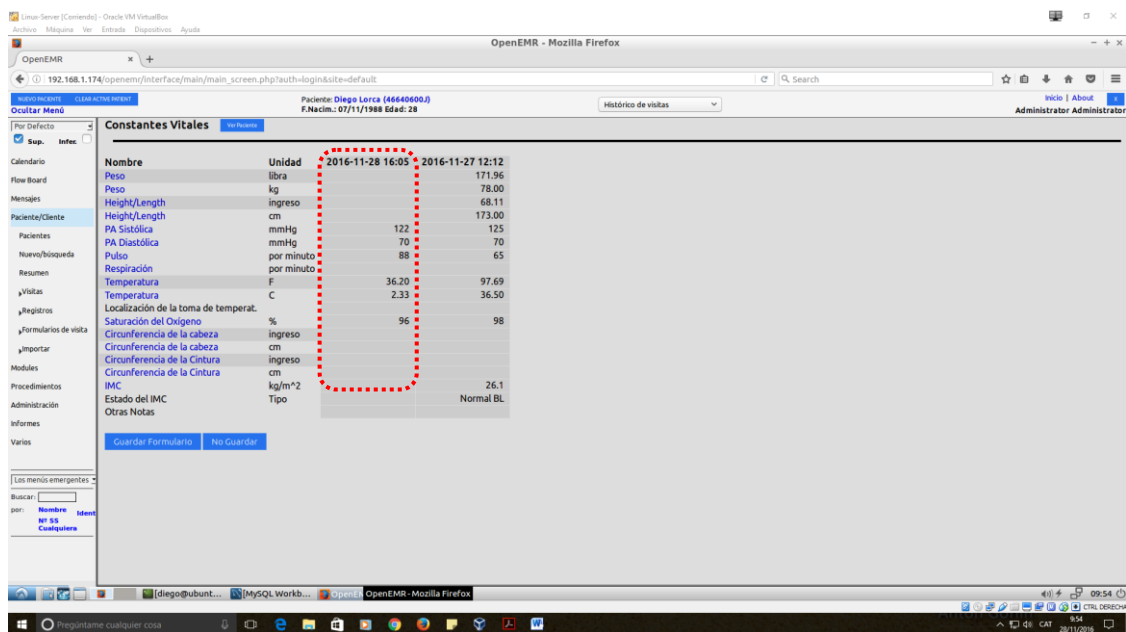


Figura 65. Registro constantes en tablas OpenEMR paciente “Diego”

En el caso del otro paciente, Laura, no se introdujeron sus constantes vitales manualmente durante su visita.

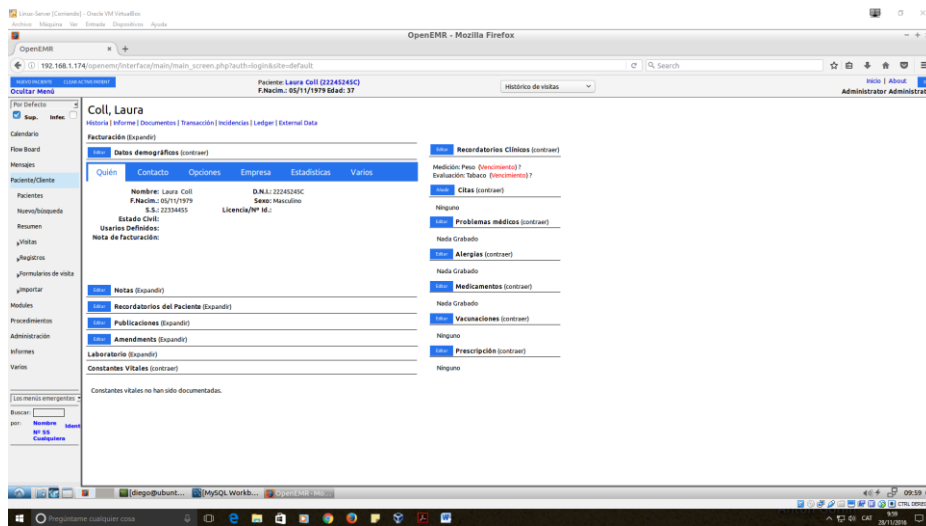
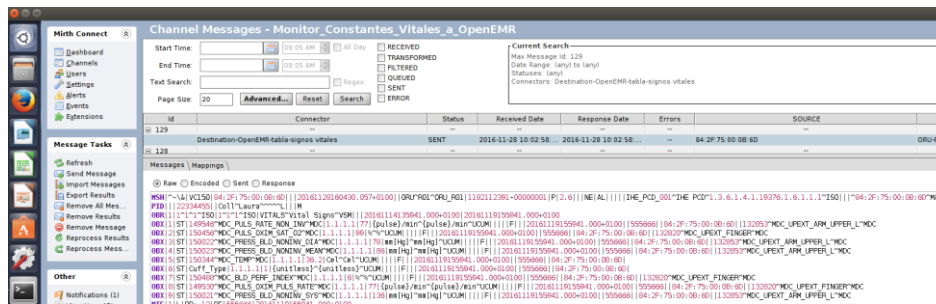


Figura 66. OpenEMR - Sin registro manual constantes vitales paciente “Laura”

Al igual que en el paciente anterior, las mediciones se realizan con el monitor y su registro se envían directamente a OpenEMR una vez se valida por el usuario.



La página de tendencias de las constantes refleja el registro:

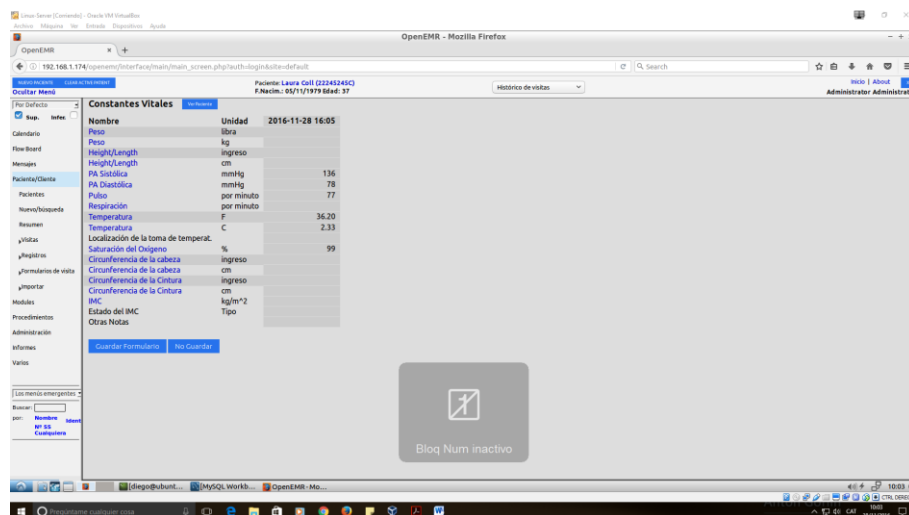


Figura 67. OpenEMR - Registro constantes en tendencias tabulares de “Laura”

La misma información de los dos pacientes, vista desde la perspectiva de la base de datos de OpenEMR, extraída con sencillas sentencias SQL con MySQLWorkbench:

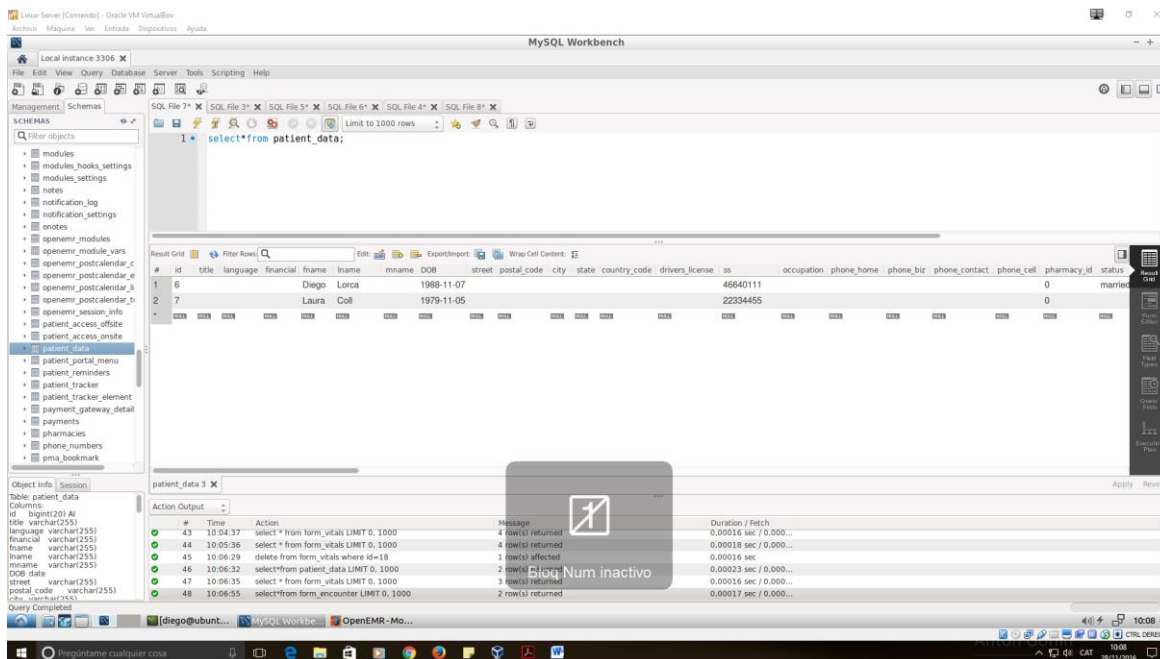


Figura 68. Pacientes admitidos

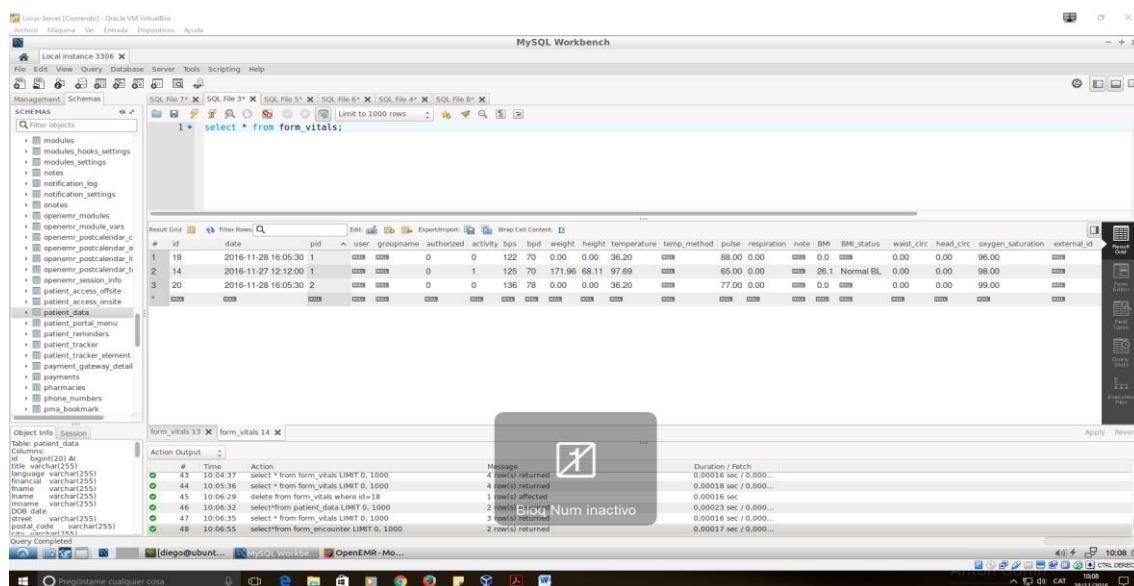


Figura 69. Registros de constantes vitales pacientes

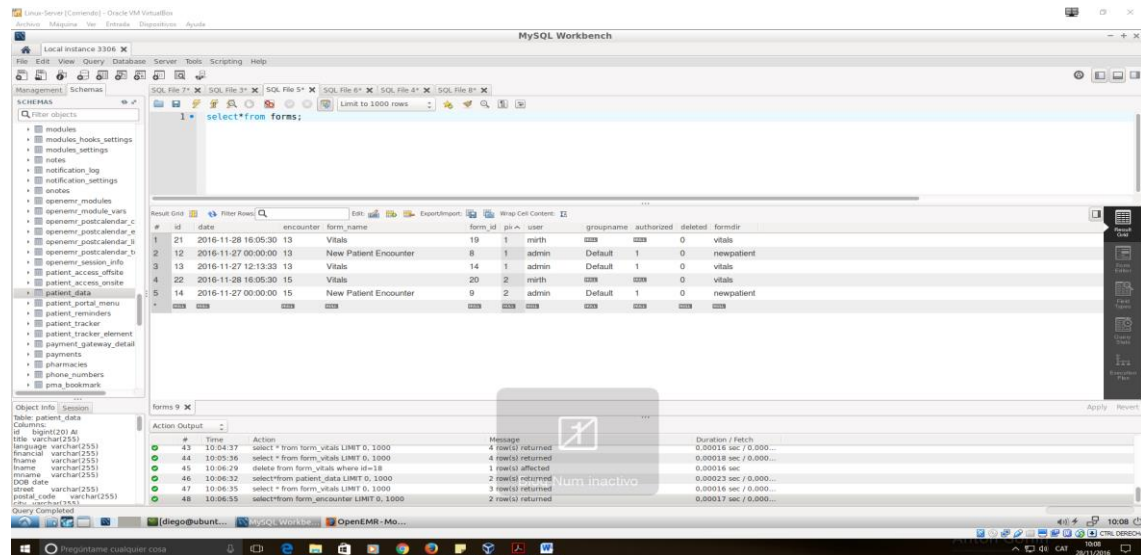


Figura 70. Registros de formularios manuales y automáticos

El caso de uso para el registro de constantes vitales planteado en el escenario del epígrafe 3.1 se ha realizado satisfactoriamente.

El JavaScript del conector de entrada extrae los datos del mensaje HL7, perfil IHE PCD-01, nomenclatura ISO/IEEE 11073-10101 con terminología armonizada hRTM y, por tanto, puede reutilizarse con cualquier monitor de constantes que la integre.

El JavaScript del conector de salida se ha programado específicamente para insertar las constantes en OpenEMR y deberá modificarse para otros gestores de HCE.

5. Conclusiones

La realización del proyecto TFG ha permitido obtener algunas conclusiones finales, las más importantes a citar:

1. La situación actual de los dispositivos médicos existentes en los hospitales dista mucho de aproximarse al ideal que permita la integración e interoperabilidad automatizada con la HCE. La **falta de salidas de datos estandarizadas** y los **sistemas operativos “propietarios”** de los fabricantes de equipos médicos, junto a un consenso bastante reciente en cuanto a qué estándares aplicar a lo largo de toda la cadena de diseño-fabricación-adopción-estándar, son las **principales barreras** y la razón de ser de las **dificultades de integración**, de la existencia de múltiples estrategias de conectividad y de la proliferación de compañías consultoras y su oferta de servicios profesionales.
2. La armonización y el consenso en torno a los estándares **ISO/IEEE11073, HL7** con la propuesta de **perfiles IHE** han sentado las **bases para avanzar en el camino correcto** de su adopción, hacia la simplificación de las integraciones y la reducción en el consumo de recursos asociados: económicos y de tiempo.
3. La integración de los dispositivos médicos con la HCE mediante el uso de **software “Open Source”** y de los estándares **ISO/IEEE11073, HL7** con **perfiles IHE** hacen **factible la creación de soluciones** específicas, coste-eficientes y adaptadas a diferentes casos de uso o flujos de trabajo asistenciales.
4. El objetivo principal del TFG, la integración anterior, ha podido realizarse dentro de la planificación temporal marcada para el proyecto usando la metodología ágil SCRUM. No obstante, y a pesar de que el monitor de constantes vitales usado en el laboratorio integra el **perfil PCD-01**, la **falta de compatibilidad directa con el perfil** y con la nomenclatura ISO/IEEE11073, tanto del motor de integración Mirth Connect como del gestor OpenEMR, **ha dificultado la integración**.
5. Las dificultades descritas en el punto anterior han requerido el **mapeo de las variables** de interés y la **programación de scripts** para su extracción e inserción en la base de datos OpenEMR. Esta fase ha sido la **mayor consumidora del recurso tiempo** ya que se ha tenido que estudiar y aprender las bases de la programación de *JavaScript* para escribir los guiones.
6. La realización del proyecto ha permitido poner en práctica muchos aspectos y temas tratados durante el grado de Informática. La creación y configuración de redes, la instalación de servidores y de diferentes componentes de software, la programación de scripts, el estudio de gestores de bases de datos y la manipulación de sus tablas, etc.

7. En la situación actual, es necesario **avanzar y promover la adopción de los estándares por** parte de todos los actores implicados en el entorno de las integraciones e interoperabilidad de los equipos médicos, **los fabricantes** de tecnología médica en las diferentes categorías de dispositivos (ventiladores, bombas de infusión, incubadoras...), **los proveedores de sistemas de gestión para la HCE y los diseñadores de soluciones de conectividad middleware** o motores de integración; todos ellos deben aunar esfuerzos para hacer de la conectividad e interoperabilidad de los dispositivos médicos una realidad fácil de implantar.

8. Las **líneas de trabajo futuras** podrían centrarse en **propuestas concretas, adaptadas a diferentes casos de uso y con otros dispositivos médicos**, para promover la reutilización del conocimiento e impulsar la generación de artículos y trabajos basados en casos reales. Sin duda este movimiento, **con el tiempo**, ayudará a conseguir la tan preciada **conectividad e interoperabilidad semántica *plug&play***.

6. Bibliografía

- [1] GEE, TIM (2012). *EMR Integration for Medical Devices: The Basics*, [en línea], <http://medicalconnectivity.com/2011/04/03/emr-integration-for-medical-devices-the-basics> [fecha de consulta: 2 de octubre 2016].
- [2] IATRIC SYSTEM INC. (2015). *Barriers to Medical Device Integration*, [en línea], <http://docs.iatric.com/hs-fs/hub/395219/file-2755268782-pdf/> [fecha de consulta: 29 septiembre 2016].
- [3] GARCIA, R., JOANA, J.M., LUCIA, A., BOLART, J. (2011). *Gestió amb èxit de projectes de transformació: El cas ICS*. Barcelona: Profit Editorial. 2011.
- [4] MOORMAN, BRIDGET (2010). *Medical Device Interoperability: Standards Overview*, [en línea], http://www.continuaalliance.org/sites/default/files/132-138_IT_WorldMA2010.pdf [fecha de consulta: 30 de septiembre 2016]
- [5] ZALESKI, JOHN (2012). *Medical Device Interoperability and Data Integration to Clinical Information Systems. Medical Device Data Alignment*, [en línea], http://s3.amazonaws.com/rdcms-aami/files/production/public/FileDownloads/HT_Interoperability/2012_Fall_Horizons_Medical_Device_Data_Alignment.pdf [fecha de consulta: 9 de octubre 2016]
- [6] TicSalut, *Normes ISO 11073/IEEE 1073. Plantejament general i pautes d'aplicació*, [en línea], http://www.ticsalut.cat/media/upload/pdf/integracio_dispositiu_medics_normes_iso_11073-ieee-1073_ticsalut_editora_21_17_1.pdf [fecha de consulta: 20 de octubre 2016]
- [7] IHE (2015). *IHE Patient Care Device (PCD) Technical Framework. Volume 3 IHE PCD TF Semantic Content*, [en línea], http://www.ihe.net/uploadedFiles/Documents/PCD/IHE_PCD_TF_Vol3.pdf, [fecha de consulta: 20 de octubre 2016]
- [8] MERRITT, STEVE. (2010). *Presentation for Electronic Health Record Association, Device Connectivity & IHE PCD*, [en línea], http://wiki.ihe.net/index.php/PCD_Presentations, [fecha de consulta: 23 de octubre 2016]
- [9] IHE(2011). *IHE Patient Care Device User Handbook*, [en línea], http://www.ihe.net/resources/upload/ihe_pcd_user_handbook_2011_edition.pdf, [fecha de consulta: 22 de octubre 2016]
- [10] RAMONRAMON (2013). *Gestión hospitalaria (HIS) con software libre*. [en línea], <https://www.ramonramon.org/blog/2013/06/12/gestion-hospitalaria-his-con-software-libre/>, [fecha de consulta: 28 de noviembre 2016]
- [11] PATESCO (2013). *Introduction To Mirth*. [en línea], <http://wiki.patesco.ca/doku.php?id=hl7:mirth:intro>, [fecha de consulta: 29 de noviembre 2016]
- [12] Mirth® Connect 3.4 User Guide (2016), [en línea], <https://www.mirth.com/>, [fecha de consulta: 8 de diciembre 2016]
- [13] HL7 en Español (2013). *Estrenando la pequeña navaja suiza*, [en línea], <http://hl7es.blogspot.com.es/2013/03/estrenando-la-pequena-navaja-suiza.html>, [fecha de consulta: 8 de diciembre 2016]
- [14] GE Healthcare-Innokas Medical (2015). *CARESCAPE™ VC150 Vital Signs Monitor HL7 Reference Manual*.
- [15] Sourceforge OpenEMR (2013). *DB insert to form_vitals?*, [en línea], <https://sourceforge.net/p/openemr/discussion/202506/thread/702fddcd/>, [fecha de consulta: 9 de diciembre 2016]

7. Anexos

Los anexos I y II muestran los guiones JavaScripts que se han creado en la plataforma Mirth Connect para poder realizar la integración del monitor de constantes vitales con la historia clínica electrónica de OpenEMR. No obstante, esta memoria de TFG se complementa con dos ficheros:

- *Conector_Fuente_Java_Script_Transformador_extracción_constanes_vitales_paciente*
- *Conector_Salida_Java_Script_Inserción_a_tabla_signos_vitales_OpenEMR*

Ambos ficheros corresponden a la exportación en formato XML de la configuración de los conectores de entrada y salida de Mirth Connect, incluyendo los JavaScripts. La importación de los mismos en Mirth Connect permite reconstruir y usar los interfaces creados para propósitos de restauración, copia y/o expansión a otras instalaciones que pudieran plantearse la misma configuración y caso de uso descrito en este TFG.

El anexo III corresponde a la grabación de un vídeo de corta duración que muestra el funcionamiento de la propuesta sobre el entorno de pruebas.

Anexo I – JavaScript “constantes_vitales_paciente”

```

/* Script que localiza la existencia de los segmentos OBX en el mensaje HL7*/
/* del canal de entrada, independientemente del número de repeticiones y su*/
/* posición en el mensaje, y busca el parámetro o signo vital clínico para**/
/* convertirlo en una variable que posteriormente pueda ser usada para***/
/* procesarla*/

//Se crea un array de segmentos, en este caso de OBX's
var obxSegments= msg['OBX'];

//Iteramos en el array sobre los segmentos
for each (var obx in obxSegments){
    //Extraemos los campos que nos interesan mapear
    var signo_vital= obx ['OBX.3']['OBX.3.2'];
    var valor= obx['OBX.5']['OBX.5.1'];

    /**Función utilizada para ver por consola de Mirth la información**/
    /**de puntos de control o saltos dentro del script *****/

    logger.info(signo_vital);

    // Se genera una lista de todos los segmentos del mensaje HL7 y
    // se localiza la posición del segmento OBX actual en la listas
    var segments= obx.parent().children();
    var obxIndex= obx.childIndex();
  
```

```

// Se inicia la iteración sobre los segmentos OBX para localizar
// el campo que nos interesa y convertirlo en variable
var nextIndex= obxIndex;
while (nextIndex<segments.length()){

    var nextSegment= segments[nextIndex];
    var nextSegmentType= nextSegment.localName();

        if (nextSegmentType=='OBX'){

            var obx= nextSegment;

            if (signo_vital=='MDC_PRESS_BLD_NONINV_SYS'){
                channelMap.put('PNI-sistolica',valor);
            }

            if (signo_vital=='MDC_PRESS_BLD_NONINV_DIA'){
                channelMap.put('PNI-diastolica',valor);
            }

            if (signo_vital=='MDC_PRESS_BLD_NONINV_MEAN'){
                channelMap.put('PNI-media',valor);
            }

            if (signo_vital=='MDC_PULS_OXIM_PULS_RATE'){
                channelMap.put('Fpulso',valor);
            }

            if (signo_vital=='MDC_PULS_OXIM_SAT_O2'){
                channelMap.put('SpO2',valor);
            }

            if (signo_vital=='MDC_TEMP'){
                channelMap.put('Temperatura',valor);
            }
            else{
                break;
            }
        }
    //Finalizada la iteración se salta a otros segmentos
    ++nextIndex;
}

//Final script

```


Anexo II – JavaScript “Destination-OpenEMR-tabla-signos-vitales”

```

var dbConn;

try {

dbConn = DatabaseConnectionFactory.createDatabaseConnection('Please Select
One', 'jdbc:mysql://192.168.1.174:3306/openemr', 'mirth', '1005');

//Se Busca el pid que OpenEMR tiene asignado al paciente con número de HCE igual
//al campo "ss" en la tabla "patient_data"
var query="SELECT pid FROM patient_data WHERE ss="+$('id_paciente)+" ";
var result= dbConn.executeCachedQuery(query);
result.next();
//Si se hay fila que coincide con el criterio de búsqueda, se obtiene el pid
  if (result.getString(1)!=0) {
    var pid= result.getString(1);
    //Antes de registrar las constantes vitales del paciente actual
    //localizamos la última "id" registrada en la tabla //"form_vitals"
    var query= "SELECT id FROM form_vitals";
    var result= dbConn.executeCachedQuery(query);
    var last_id;
    while (result.next()){
      last_id=result.getInt(1);
    }
    //Próxima fila que se insertará en la tabla "form_vitals" será:
    var next_id= last_id+1;
    logger.info(next_id);
    //Se insertan los valores procedentes del monitor de constantes
    var sql1= "INSERT INTO form_vitals (id, date, pid, bps, bpd,
    temperature, pulse, oxygen_saturation) VALUES ("+next_id+",
    "+$('fecha')+", "+pid+" , "+$('PNI-sistolica')+", "+$('PNI-
    diastolica')+", "+$('Temperatura')+", "+$('Fpulso')+", "+$('SpO2')+");";
    var result = dbConn.executeUpdate(sql1);
    //Se actualiza la tabla "forms" con el registro del formulario
    //correspondiente a la inserción realizada
    var query="SELECT encounter FROM form_encounter WHERE pid="+pid+"";
    var result= dbConn.executeCachedQuery(query);
    result.next();
    if (result.getString(1)!=0) {
      var encounter= result.getInt(1);
      logger.info(encounter);
    }
    //Se inserta el registro del formulario correspondiente a las
    //constantes vitales en la tabla "forms"
    var sql1= "INSERT INTO forms (date, encounter, form_name,
    form_id, pid, user, formdir) VALUES ("+$('fecha')+",
    "+encounter+", 'Vitals', "+next_id+", "+pid+", 'mirth',
    'vitals')";
    var result = dbConn.executeUpdate(sql1);
  }
} finally {

  if (dbConn) {
    dbConn.close();
  }
} // Final script

```


Anexo III – Video del funcionamiento de la solución propuesta

Los videos “*MirthConnect_video1*” y “*OpenEMR_video2*” que se adjuntan con este TFG, complementan la memoria del proyecto:

- Ofrecen una visión rápida de la configuración y la conectividad de las dos máquinas virtuales que configuran el entorno de pruebas
- Muestran el funcionamiento y la configuración básica de Mirth Connect
- Enseñan cómo enviar mensajes HL7-IHE a Mirth Connect desde un dispositivo externo, simulado mediante el programa HL7 Inspector
- Visualizan la interface de OpenEMR y el resultado de la integración automática de las constantes vitales del paciente en su HCE

Nota: Ha sido necesario realizar dos videos debido a que las instalaciones y el funcionamiento de Mirth Connect y OpenEMR se encontraban en equipos/máquinas virtuales diferentes.