

EHRGen: Generador de Sistemas Normalizados de Historia Clínica Electrónica Basados en openEHR

Pazos P.

openEHR en español, Montevideo, Uruguay

Resumen

El uso de sistemas informáticos en el área clínica permite que los profesionales de la salud manejen mayor cantidad y variedad de información de forma efectiva y eficiente, aumentando su capacidad de resolución de problemas de salud y la calidad asistencial de los pacientes¹. Estos son sistemas complejos, difíciles de mantener a largo plazo, y de costos elevados en su construcción y mantenimiento, tanto en tiempo como en recursos humanos y económicos. La falta de aplicación de estándares y buenas prácticas en su diseño y construcción repercute negativamente en la calidad global del producto final. Esta situación pone en riesgo la viabilidad de los proyectos de informatización del área clínica, a gran escala y largo plazo. En este trabajo planteamos un enfoque distinto en la construcción de sistemas de información en salud, que busca sortear los problemas mencionados, en pos de viabilizar los proyectos de informatización del área clínica en contextos de recursos limitados (económicos y humanos), apuntando a la creación de sistemas informáticos de calidad, basados en estándares, mantenibles a largo plazo y a costos controlados.

Se propone la creación de una herramienta de software para la automatización de tareas en el desarrollo informáticos y un proceso de desarrollo drásticamente diferente al proceso actual, donde el profesional de la salud pasa a jugar el rol de mayor importancia en el proceso, donde el foco está en la gestión del conocimiento clínico, dejando la tecnología en un segundo plano. Con el enfoque propuesto se podrían crear sistemas de registro clínico de gran calidad, en menos tiempo, y con un menor consumo de recursos. Estos sistemas estarían basados en estándares, creados para ser modificados y capaces de evolucionar a largo plazo sin necesidad de modificar el código fuente del software.

I. Introducción

El proceso habitual de desarrollo de software en salud implica que el conocimiento clínico sea implementado de forma “dura” en el software, creando una interdependencia que genera elevados costos y tiempos en la modificación del software a nuevos requerimientos de registro clínico. Esto se debe a que cambios en el conocimiento clínico repercuten en cambios al código fuente de las aplicaciones de software. Este enfoque se conoce como “modelado en un nivel”, donde existen representaciones de conceptos clínicos particulares en los componentes del software (estructuras de bases de datos, interfaz de usuario, lógica de negocio y servicios) [1]. La interdependencia entre el software y el conocimiento clínico es una consecuencia innecesaria que se puede evitar, pero es necesario repensar el proceso de desarrollo y la metodología de diseño de software.

El proyecto EHRGen propone una herramienta de software y un proceso de trabajo, orientado por la gestión del conocimiento clínico, para la creación de sistemas de Historia Clínica Electrónica (HCE). Este proceso busca solucionar algunos de los problemas y riesgos más comunes en la construcción y mantenimiento de software para el área clínica, como ser la comunicación entre profesionales sanitarios e informáticos, la agilidad en la implementación de cambios, la mantenibilidad de la HCE a largo plazo y costos acotados. EHRGen busca formalizar, automatizar y acelerar las tareas comunes en el desarrollo y mantenimiento de la HCE, con la flexibilidad suficiente para poder ser adaptada a nuevos requerimientos de registro clínico sin necesidad de modificar el código fuente de la aplicación.

¹ S. Indarte, P. Pazos, Estándares e Interoperabilidad en Salud Electrónica: Requisitos para una Gestión Sanitaria Efectiva y Eficiente. CEPAL, 2012.

El objetivo de EHRGen es permitir la creación de sistemas de Historia Clínica Electrónica (HCE) completos, de calidad, basados en estándares, capaces de ser implementados en entornos de recursos limitados, buscando servir de base a proyectos viables y sustentables a largo plazo y gran escala. Para lograr estos objetivos, EHRGen implementa del estándar openEHR [2] como piedra angular de todo el sistema de HCE. OpenEHR define una arquitectura de HCE que separa el conocimiento clínico de la aplicación de software [3]: el conocimiento es modelado formalmente y gestionado por fuera del software, transformando al mismo en una herramienta genérica donde los conceptos clínicos concretos pueden verse como parámetros configurables. Estos conceptos (ej. presión arterial, frecuencia cardíaca, diagnóstico, etc.) definen la estructura del registro clínico de la HCE.

Debido a que las modificaciones a la estructura del registro clínico son de los cambios solicitados con más frecuencia, con el enfoque de openEHR dichos cambios impactan solo en elementos de conocimiento gestionados por fuera del software, disminuyendo drásticamente las modificaciones al código fuente de la aplicación y los posteriores procesos de testeado y aseguramiento de la calidad. Esto repercute en ahorro de tiempo, baja de costos de mantenimiento, y en la calidad del software, ya que no se introducen errores por modificaciones al código fuente. OpenEHR también define un modelo de información [4] que permite conservar toda la información clínica generada durante la vida del paciente, en una estructura estándar, genérica e interoperable, capaz de sobrevivir a evoluciones y recambios tecnológicos.

Las principales contribuciones de este trabajo son:

- Proponer una forma novedosa para crear sistemas de información en salud, utilizando interfaces de usuario generadas automáticamente a partir de definiciones de conceptos clínicos.
- El diseño de un sistema que permite la modificación del registro clínico, en cuanto a estructura y datos, sin necesidad de modificar el código fuente de la aplicación o el esquema de la base de datos.
- La formalización del proceso de desarrollo de sistemas de información en salud, que se centra en la gestión del conocimiento clínico, y permite la evolución de la herramienta de software sin ser modificada internamente.

II. Marco de trabajo basado en openEHR

La herramienta de software del proyecto EHRGen es un software parcialmente implementado, que contiene las funcionalidades comunes a cualquier sistema de HCE, sin incluir las particularidades de cada uno, lo que se conoce como “marco de trabajo de desarrollo”. Es decir que EHRGen no es un sistema de HCE particular, sino una base genérica para crear cualquier HCE. Esta es una herramienta de código abierto liberada bajo la licencia Apache 2.0 [5], que cualquiera puede descargar, adaptar a sus necesidades y utilizar libremente. EHRGen implementa el estándar openEHR y su modelo en dos niveles (o modelo dual) que permite separar la gestión del conocimiento clínico de la gestión del software, donde los cambios en cada uno de estos elementos son procesos separados e independientes.

A. Implementación de openEHR y el modelo dual

OpenEHR define un modelo de información y un modelo de conocimiento. El modelo de información de openEHR [4] es capaz de representar cualquier estructura de información clínica dentro de una aplicación

de software. Es un modelo genérico, pequeño y estable, características que buscamos en el software. El modelo de conocimiento de openEHR, llamado “modelo de arquetipos”, permite representar conceptos clínicos particulares que definen la estructura del registro clínico (ej. presión arterial, diagnóstico, órdenes de estudios, etc.) en distintos contextos asistenciales (emergencia, hospitalización, etc.). Los conceptos clínicos particulares son gestionados por fuera del sistema de HCE, lo que permite evolucionar el registro clínico sin necesidad de modificar el código fuente de la aplicación.

B. openEHR: Arquetipos y Bases de Conocimiento

El conocimiento representado por los arquetipos es en esencia un gran conjunto de definiciones de conceptos clínicos como presión arterial, frecuencia cardíaca, evaluación de vía aérea, diagnóstico, prescripción de medicamento, plan de cuidados, procedimiento quirúrgico, etc. Estas definiciones de conceptos se almacenan en una Base de Conocimiento, que puede ser utilizada y compartida entre distintos sistemas de información. La comunidad de openEHR mantiene una Base de Conocimiento internacional llamada Clinical Knowledge Manager (CKM) [6], que permite la colaboración entre distintos profesionales para modelar, traducir y utilizar los conceptos clínicos ahí definidos.

Los arquetipos pueden ser traducidos a diversos idiomas y utilizados en otros países, pueden ser compuestos para formar conceptos clínicos complejos, y pueden ser especializados desde arquetipos más generales. Esto brinda una gran flexibilidad en cuanto al modelado del conocimiento, y muestra la necesidad de contar con un proceso bien definido de gestión del conocimiento, que garantice la consistencia y coherencia entre los arquetipos existentes en la base de conocimiento.

La Base de Conocimiento que será utilizada por un determinado sistema de HCE debe ser creada o adaptada por expertos del dominio clínico, como médicos, personal de enfermería, personal técnico, entre otros. Esto supone un cambio radical al clásico proceso de desarrollo de software, por ejemplo, la etapa de análisis de requerimientos se reduciría al mínimo ya que los requerimientos sobre los registros clínicos son modelados formalmente en la Base de Conocimiento. Y no solo modelados, sino que en una forma computable, haciendo que el pasaje entre el requerimiento y su implementación pueda hacerse de forma automática o semi-automática. Esto se contrasta al enfoque actual donde los conceptos clínicos son modelados representados en documentos narrativos que no permiten ser procesados de forma automática.

Si bien esta automatización supone una mejora al proceso de desarrollo en la disminución de tiempos y costos, el gran cambio al proceso es que el conocimiento clínico ya no debe ser extraído, procesado, interpretado y gestionado por analistas informáticos, sino que el conocimiento es directamente modelado por personal clínico. Cabe señalar que los profesionales informáticos no están capacitados para la gestión de conocimiento clínico y esto trae problemas como el relevamiento incorrecto o incompleto de requerimientos, incluyendo el costo del retrabajo necesario para realizar correcciones en dichos requerimientos. Esto también genera problemas de comunicación entre el equipo clínico y el equipo informático, perjudicando al proyecto.

Con el nuevo proceso, el requerimiento de crear un registro clínico electrónico para la atención en el departamento de emergencia de un determinado hospital, se transforma en la selección de un conjunto de arquetipos, definidos por los expertos del dominio clínico, que definirán la estructura interna de dicho registro. Ahora el equipo informático debe proveer un software capaz de procesar esos arquetipos de modo que el usuario final obtenga el resultado esperado. EHRGen es un ejemplo de una herramienta de este tipo.

En el proceso propuesto, el modelado de conceptos se realiza mediante herramientas especialmente diseñadas para dicho fin. Los expertos del dominio deben ser capacitados en el uso de estas herramientas, y en conceptos generales de modelado y calidad de la información. La herramienta central en este proceso de modelado del conocimiento es el editor de arquetipos [7], que permite crear conceptos clínicos de distintos tipos mediante una interfaz de usuario amigable.

Una base de conocimiento, lista para ser utilizada, estaría compuesta por arquetipos existentes obtenidos del CKM, posiblemente traducidos para el idioma local y/o adaptados o extendidos según las necesidades de cada caso, y por arquetipos totalmente nuevos creados por el equipo clínico.

A medida que aumenta la cantidad y variedad de conceptos en la Base de Conocimiento, aumenta la capacidad de reutilización de conceptos en nuevos contextos asistenciales (emergencia, ambulatorio, hospitalización, etc.), lo que disminuye los tiempos de los nuevos desarrollos y ahorra recursos. Además como distintos sistemas de HCE utilizan y reutilizan los mismos conceptos, los registros clínicos realizados en distintos contextos asistenciales mantienen una consistencia estructural y coherencia semántica, lo que permite la comunicación e integración de la información residente en distintos sistemas, permitiendo el uso de la información clínica para fines asistenciales y para fines secundarios como gestión, investigación, control de enfermedades, etc. Esta es la gran ventaja de la utilización de estándares en los sistemas de información en salud.

La reutilización de los conceptos y la capacidad de procesamiento de la información tiene un gran potencial para repercutir positivamente en los tiempos, costos totales y retorno de inversión del proyecto.

Una consecuencia muy positiva del nuevo enfoque es el empoderamiento del profesional sanitario, ya que le permite una participación más activa en el desarrollo de los sistemas de información clínicos, y un mayor control sobre los requerimientos, ya que son definidos directamente por ellos mismos en forma de arquetipos. Esto repercute en potenciar la adopción de la herramienta entre otros profesionales de la salud, porque ahora el interlocutor es un par médico no un informático. Otro punto interesante es que este enfoque puede ayudar a mejorar la comunicación entre el equipo clínico y el equipo informático (factor más que problemático en cualquier proyecto de sistemas de información en salud), ya que ahora se cuenta con un nuevo elemento, conocido por ambos perfiles, que actúa de interfaz entre ellos: el arquetipo.

C. Arquitectura de EHRGen

La especificación de openEHR plantea que un sistema mínimo de Historia Clínica Electrónica basado en tres componentes fundamentales: el repositorio clínico (RC), el repositorio demográfico (RD) y la base de conocimiento (BC) (fig. 1). El RC mantiene toda la información clínica de los pacientes en algún formato que debe cumplir con el modelo de información del estándar. Esto permite que la información clínica pueda ser compartida fácilmente entre diversos sistemas y repositorios. Este es el principio básico para la implementación de una verdadera historia clínica única de cada paciente (con un enfoque distribuido), longitudinal a la vida del paciente (desde que nace hasta que muere) y transversal a diversas instituciones (toda la información clínica, independientemente de dónde se haya generado, capaz de compartirse usando cualquier protocolo de comunicación y formato de intercambio).



Fig. 1: arquitectura de HCE mínima según openEHR

El RD contiene la información de las personas y sus roles (paciente, médico, etc.), junto a su identificación, identidad, medios de contacto y ubicación. También puede contener información de organizaciones a las cuales están vinculadas las personas y roles.

La BC mantiene los arquetipos y terminologías que serán utilizados por los sistemas de HCE. Los arquetipos modelan la estructura interna de los conceptos clínicos y contienen referencias a distintas terminologías médicas, permitiendo la reutilización de la información clínica, tanto para fines asistenciales, como para la educación, gestión, investigación y análisis epidemiológico, entre otros usos. El diseño de EHRGen respeta la arquitectura mínima propuesta por openEHR, y define otros componentes que permiten la implementación de sistemas de HCE modificables y adaptables a distintos contextos (fig. 2).



Fig. 2 arquitectura de componentes de EHRGen

El componente Knowledge Access (KA) permite que la aplicación acceda a los elementos contenidos en la base de conocimiento (plantillas de interfaz de usuario, arquetipos y terminologías), mediante servicios de búsqueda y carga. Por ejemplo, los arquetipos se encuentran en formato ADL (Archetype Definition Language) [8], en un repositorio físico (sistema de archivos, base de datos, etc.), y KA se encarga de leer y procesar los archivos ADL, cargarlos en memoria, dejándolos accesibles a los demás componentes de EHRGen.

El componente Information Model (IM) implementa el modelo de información de openEHR como un conjunto de clases persistentes, que mediante el componente de Object-Relational Mapping [9] de Grails

Framework [10], permite el almacenamiento y carga de objetos desde bases de datos relacionales u orientadas a documentos (JSON o XML). Debido al modelo dual de openEHR, no es necesario modificar el IM ante la necesidad de registrar nueva información clínica, ya que las modificaciones al registro clínico se realizan sobre los arquetipos, por fuera del software.

El componente Data Binder implementa gran parte de la lógica que le brinda flexibilidad a EHRGen. Su objetivo es el de crear estructuras del IM a partir de los datos ingresados por un usuario en una pantalla de registro clínico. Asimismo valida los datos ingresados en base a las restricciones especificadas en los arquetipos que se utilizaron para generar la pantalla. Que se utilicen las mismas estructuras y restricciones de los arquetipos para generar la interfaz de usuario y para validar los datos ingresados en estas permite la consistencia entre lo que un médico ve en dicha interfaz y la información que se almacenará en la base de datos.

GuiGen es el componente encargado de generar la interfaz de usuario a partir de la estructura definida dentro de los arquetipos y las plantillas de interfaz de usuario. EHRGen es un sistema basado en Web, por lo que GuiGen genera código HTML. Actualmente se está trabajando en ampliar la capacidad de GuiGen para poder generar interfaces de usuario en otras tecnologías (aplicaciones móviles y de escritorio).

Las plantillas son agregadores de arquetipos que contienen directivas para la generación de la interfaz de usuario (ubicación de cada componente, tipo de componente a mostrar, etc.). Éstas permiten seleccionar qué partes de los arquetipos serán utilizadas para generar las pantallas de registro clínico, o sea que permiten personalizar el uso de los arquetipos para distintos contextos. Por ejemplo, el arquetipo de presión arterial define estructuras para la presión sistólica y diastólica, el dispositivo que se utilizó para tomar la medida y para la posición del paciente en el momento de la medida. En una consulta de medicina general o medicina familiar, solo es necesaria la estructura con la presión sistólica y diastólica, pero en el contexto de un examen diagnóstico puede interesar el registro del dispositivo utilizado para la medida.

Demographic Access permite acceder a repositorios demográficos externos a través de una interfaz de servicios, que permite conectarse a distintos repositorios demográficos y realizar búsquedas, seleccionar pacientes para los cuales se crearán registros clínicos, etc. Esto es particularmente útil porque en general los hospitales cuentan con un repositorio de pacientes, y en lugar de duplicar el registro y agregar la complejidad de mantenerlo sincronizado, EHRGen busca adaptarse al existente. Igualmente EHRGen permite trabajar con una base de datos local de pacientes, considerando que en algunos contextos de uso no se podrá contar con repositorios demográficos externos.

Workflow Manager se encarga de manejar la interacción con el usuario, permitiéndole el acceso a distintas pantallas para la visualización e ingreso de información clínica en registros de salud de los pacientes. También permite acceder a información demográfica, por ejemplo realizando búsquedas en el repositorio demográfico, para luego seleccionar un paciente y crearle un nuevo registro clínico.

III. Metodología de trabajo

La metodología de trabajo de EHRGen es la que permite explotar todas las ventajas de openEHR y su modelo dual. Esta metodología se compone de cuatro procesos principales: Modelado base, Modelado de

registros clínicos, Generación de interfaz de usuario y la Utilización del sistema. El nombre que proponemos para esta metodología es el de “desarrollo de software orientado a la gestión del conocimiento”, en este caso conocimiento clínico.

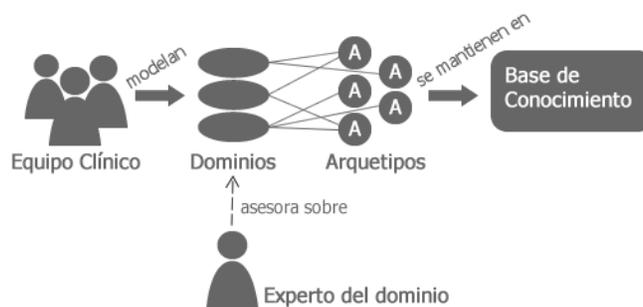


Fig. 3: Proceso de modelado base.

El objetivo del modelado base es generar los elementos semánticos mínimos sobre los cuales se basará todo el proceso de gestión del conocimiento, es decir que se crean los arquetipos que representan conceptos clínicos simples, de forma amplia y auto-contenida. Los arquetipos son piedra angular de todo el sistema de registro clínico, permitiendo la mantenibilidad del sistema e interoperabilidad de la información.

Como se muestra en la fig. 3, un equipo de profesionales de la salud (médicos, enfermeras, técnicos, etc.), capacitados en el uso de herramientas de modelado del conocimiento, analiza los procesos y los registros clínicos de distintos dominios asistenciales (ambulatorio, emergencia, hospitalización, etc.), detectando los conceptos clínicos que definen cada registro clínico. Algunos conceptos pueden ser muy generales, como la “presión arterial”, o muy complejos y específicos para cada dominio, donde se podría contar con la colaboración de un experto en ese dominio particular, para lograr un modelo correcto y de calidad de cada concepto.

Luego los arquetipos son cargados en la Base de Conocimiento, donde quedan disponibles para su uso posterior. Un concepto clínico puede reutilizarse en más de un dominio, incluso puede ocurrir que en la Base de Conocimiento ya existan los arquetipos necesarios para los dominios que se están analizando. A medida que la Base de Conocimiento crece en contenido, el nivel de reutilización aumenta. Esto reduce los tiempos de modelado, permite la consistencia estructural y coherencia semántica entre los registros clínicos de distintos dominios.

El modelado de registros clínicos para un dominio, consiste en la selección y composición de un conjunto de arquetipos en una plantilla (fig. 4). En la plantilla se especifican los arquetipos y las partes de éstos que se utilizarán para modelar el registro clínico. Esto es sumamente útil ya que los arquetipos son definiciones amplias de conceptos clínicos, y en general no se utiliza toda su estructura interna. Entonces las plantillas brindan la flexibilidad necesaria para representar distintos registros, manteniendo la semántica original del concepto clínico. Las plantillas también estarán contenidas en la base de conocimiento, donde quedan disponibles para su utilización por distintas aplicaciones de registro clínico electrónico.

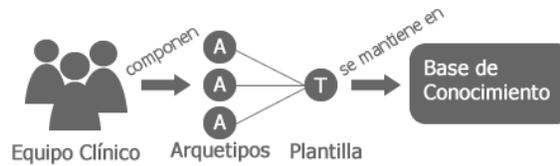


Fig. 4: Proceso de modelado de registros clínicos.

La etapa de generación de interfaz de usuario (GUI) puede ser opcional, dependiendo de las características del software que utiliza las plantillas y arquetipos. En el caso de EHRGen, que soporta la generación automática de interfaz de usuario, el equipo de profesionales informáticos configuran el componente GuiGen con las plantillas presentes en la Base de conocimiento, y a partir de estas GuiGen genera un conjunto de archivos de interfaz de usuario en formato HTML (fig. 5).

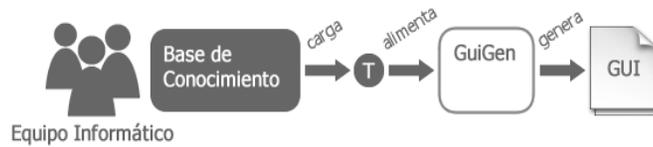


Fig. 5: Proceso de generación de interfaz de usuario.

GuiGen genera un objeto de GUI para cada clase de uso (*create*, *show* y *edit*), que se utilizan para crear nuevos registros, visualizarlos o editarlos, respectivamente. GuiGen soporta múltiples idiomas, por lo que genera un objeto de GUI para cada idioma configurado en EHRGen. Las figs. 6 y 7 muestran ejemplos de interfaces de registro clínico generadas automáticamente desde los arquetipos y plantillas por GuiGen.

La imagen muestra una interfaz de usuario con los siguientes elementos:

- Evaluación de triage:** Una lista de opciones con botones de radio:
 - estable
 - estable con lesiones a tratar
 - inestable, grave
 - moribundo, agónico
 - muerto
- Fecha y Hora de evaluación:** Campos de selección para día (20), mes (May), año (2012), hora (12) y minutos (57).
- Justificación de evaluación:** Un campo de texto con el contenido "herida en miembro inferior".
- Guardar:** Un botón ubicado en la parte inferior derecha.

Fig. 6: GUI generada a partir del arquetipo de "Triage"

(A) Vía aérea (B) Columna (B) Ventilación (C) Estado circulatorio (D) Disfunción neurológica

Evaluación de vía aérea:		Acciones sobre vía aérea:	
Presenta alteraciones:	<input type="text"/>	Máscara de oxígeno:	<input type="text"/>
Trauma sobre vía aérea:		Apoyo ventilatorio (BVM): <input type="text"/>	
Trauma directo sobre vía aérea:	<input type="text"/>	Maniobra de desobstrucción:	
Tipo de trauma:	<input type="text"/>	<input type="text"/>	
Dificultad respiratoria:	<input type="text"/>	Intubación orotraqueal:	
Obstrucción de vía aérea:		Protocolo de intubación:	
Vía aérea obstruida:	<input type="text"/>	<input type="text"/>	
Causa de obstrucción:	<input type="text"/>	Máscara de fastrach: <input type="text"/>	
Otras causas de obstrucción:		Máscara laríngea: <input type="text"/>	
<input type="text"/>		Combitubo: <input type="text"/>	
		Abordaje quirúrgico:	
		Tipo: <input type="text"/>	
		Otras acciones:	
		<input type="text"/>	
Resumen de actuación:			
Diagnóstico / Actuación:			
<input type="text"/>			
Fecha y Hora de evaluación: 20 May 2012 12:57			
Guardar			

Fig. 7: GUI generada a partir de varios arquetipos

La etapa de utilización del sistema por los usuarios finales, se puede conceptualizar en tres grandes interacciones: la solicitud para crear un nuevo registro clínico, el ingreso de datos y la visualización de datos. Estas interacciones son manejadas por el Workflow Manager, que se encarga de recibir pedidos de los usuarios, comunicarse con los componentes internos, y devolver una respuesta acorde. Esta respuesta incluye una pantalla generada por GuiGen y puede incluir un conjunto de datos para mostrarle al usuario. Primeramente el usuario debe ingresar al sistema y seleccionar el dominio clínico dentro del que va a trabajar (fig. 8).



Fig. 8: Listado de dominios clínicos

Dentro del dominio seleccionado, el usuario abre un nuevo documento clínico para el paciente, el cual incluye una serie de secciones que se corresponden a las plantillas que modelan el documento clínico.

Luego el usuario secciona una de esas secciones solicitando crear un nuevo registro (fig. 9), esto es análogo a seleccionar una plantilla con identificador *tid* y clase *create*. WM procesa la solicitud, obtiene la interfaz correspondiente, considerando el idioma que está utilizando el usuario, y devuelve la GUI solicitada que se muestra como un formulario en el navegador web del usuario.

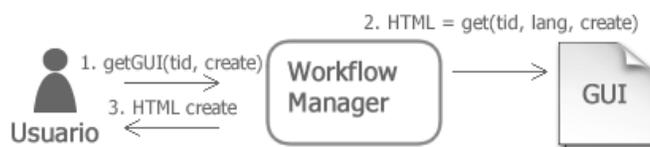


Fig. 9: Solicitud para creación de nuevo registro

En el ingreso de datos (fig. 10), se envían los datos y el identificador de la plantilla que se utilizó (*tid*). WorkFlow Manager se encarga de enviar los datos a Data Binder para construir una estructura válida del IM (*rmi*). Data Binder también valida los datos contra las restricciones presentes en la plantilla y en los arquetipos referenciados. En caso de no haber errores, la estructura se almacena en el repositorio de datos clínicos y los datos ingresados le son mostrados al usuario. En caso de haber algún error de validación, se le muestra la interfaz de edición con los datos ingresados, indicando los datos con errores. Luego el usuario deberá corregir y repetir el proceso hasta que no hayan errores.

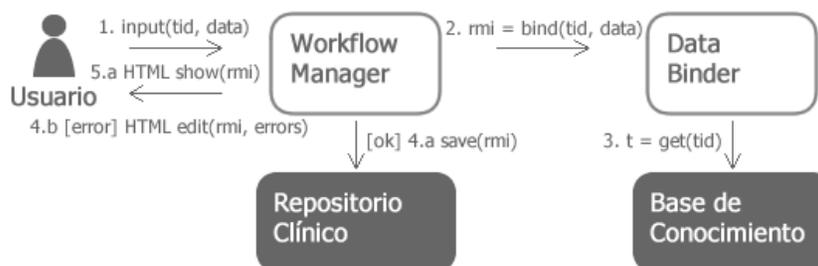


Fig. 10: Ingreso de datos

En la interacción de visualización de datos (fig. 11), el usuario solicita ver los datos ingresados y almacenados en una estructura del modelo de información que tiene el identificador *rmi_id*, utilizando la plantilla *tid*, y como es visualización se utiliza la clase “show”. El proceso es similar al caso de solicitud para crear nuevo registro, con la diferencia de que los datos deben cargarse del repositorio clínico y ser mostrados en la interfaz de usuario.

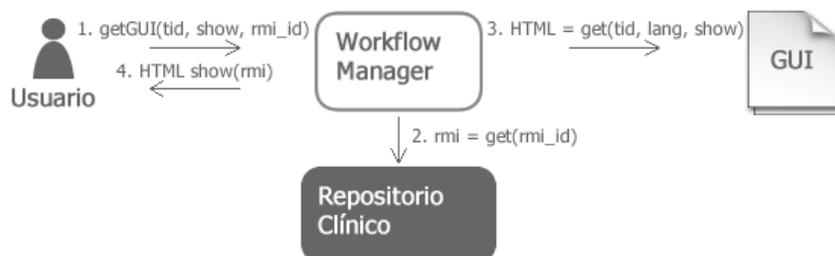


Fig. 11: Visualización de datos

Existe una cuarta interacción para editar los registros existentes, cuyos pasos están incluidos en los ya mencionados. La edición presupone que los registros no han sido cerrados y firmados por el médico

responsable, sino que son registros de trabajo cuyo contenido podrá ser corregido según sea necesario hasta el cierre del registro. Luego del cierre del registro, éste no podrá modificarse.

IV. Discusión

A priori el proceso de trabajo propuesto podría verse como un agregado de complejidad innecesaria al proceso habitual de desarrollo de software en salud, pero en esencia lo que se propone es una formalización de algunos aspectos que quedan a la deriva en el proceso habitual, como la extracción y formalización del conocimiento clínico de los profesionales. Como fue mencionado, con el enfoque propuesto los ahorros de tiempo y costos comienzan a notarse con la reutilización del conocimiento, y es natural que en etapas tempranas del proyecto no se vean estos ahorros. El principal argumento que sustenta el nuevo enfoque es el ahorro potencial en la etapa de mantenimiento y evolución del software de HCE, debido a que esta etapa corresponde a un 66% o más del ciclo de vida total del software, por lo que el mantenimiento no solo consume más tiempo que la construcción del software, sino que además puede ser más costoso [11]. Se debe tener en cuenta que los costos de mantenimiento pueden hacer que el proyecto sea insostenible a largo plazo, este punto debería ser de especial interés para los proyectos a gran escala, por ejemplo a nivel país.

A este factor se le agrega que conceptualmente son varios los sistemas de HCE que se construirán, uno por cada dominio clínico: ambulatorio, prehospitalario, emergencias, hospitalización, entre otros; la reutilización de los conceptos que definen el registro clínico se manifiesta tanto en la evolución de cada sistema, como en la multiplicidad de sistemas. Utilizando EHRGen en este contexto, u otra herramienta similar, lo que se tendría es una sola herramienta de software configurable, y una amplia base de conocimiento gestionada por profesionales sanitarios. Y los cambios al registro clínico son realizados por los mismos profesionales, por fuera del software.

Es claro que el enfoque propuesto para modelado de sistemas de HCE en dos niveles cambia radicalmente el proceso de desarrollo de software, lo que puede verse como un riesgo en algunos casos y como una oportunidad en otros. Estos argumentos deberían ser considerados en el momento de decidir que enfoque de arquitectura y diseño se utilizará en un proyecto de HCE, y dependiendo del proyecto se deberá optar por seguir el enfoque de diseño en un nivel o en dos niveles (modelo dual).

V. Estado actual y trabajo futuro

EHRGen tiene varios usos y aplicaciones potenciales. El primero, para el cual fue diseñado, es para la creación de sistemas de HCE completos para distintos contextos asistenciales. Un uso similar es el de integrarse como aplicación de registro clínico en un entorno donde ya existen otras aplicaciones de HCE, por ejemplo en proyectos a gran escala que involucran múltiples instituciones. En este tipo de proyectos la interoperabilidad entre los sistemas es de vital importancia, y contar con un modelo de información estándar facilita las tareas de integración. Además EHRGen fue diseñado para facilitar la integración con otros sistemas, por medio de distintos canales de comunicación como por ejemplo Servicios Web.

Otra aplicación posible de EHRGen es en el prototipado ágil de sistemas de HCE. Esto sirve tanto a las instituciones sanitarias que buscar hacer sus propios desarrollos informáticos, como a proveedores de software, permitiendo crear prototipos rápidamente mediante la selección y configuración de un conjunto de arquetipos. Logrando una herramienta de registro clínico funcional, sin necesidad de programación, que

puede servir para validar requerimientos con los profesionales de la salud, ajustando el prototipo a necesidades particulares, para luego ser tomado como referencia o como base de desarrollo del sistema final.

Existen algunos proyectos desarrollados utilizando EHRGen, como Traumagen [12] y HME SOS Telemedicina [13].

Algunas características y funcionalidades de EHRGen a destacar son:

- Internacionalizable y localizable: los términos en la interfaz de usuario son traducibles a cualquier idioma o cultura local.
- Integración de la clasificación internacional de enfermedades (CIE-10) para la codificación de diagnósticos. Capacidad de integrar otras terminologías, vocabularios controlados y clasificaciones.
- Búsqueda semántica: capacidad de buscar datos en los registros clínicos orientada al uso directo por profesionales de la salud, basada completamente en los conceptos clínicos que ellos entienden y manejan a diario, de forma independiente de la tecnología subyacente (SQL, tablas, etc.).

A. Búsqueda semántica

A modo de ejemplificar la funcionalidad de búsqueda semántica, se incluyeron algunas capturas de pantalla. En la fig. 12 se muestra el panel inicial de búsqueda, donde se listan los conceptos clínicos disponibles en la base de conocimiento a la que tiene acceso la aplicación. En esta pantalla, el usuario debe seleccionar un concepto para el cual desea visualizar los datos almacenados en la base de datos. Por ejemplo puede seleccionar “evaluación de triage de trauma”.

Luego el sistema le muestra la estructura interna del arquetipo que modela el concepto (ver fig. 13), donde el usuario puede seleccionar las partes de la estructura para las que desea ver datos, por ejemplo podría seleccionar “evaluación de triage”, “justificación” y “fecha de la evaluación”. Luego el sistema presenta los datos correspondientes y permite realizar agregaciones simples. Por ejemplo si se desea agregar por “evaluación de triage”, el sistema mostrará la cantidad de registros para cada una de las evaluaciones posibles: estable, estable con lesiones, inestable grave, agónico, muerto.

B. Tecnologías utilizadas

EHRGen es fue desarrollado sobre tecnologías Java. Se utilizó Grails [14] como framework base, Java [15] y Groovy [16] como lenguajes de programación, MySQL [17] o PostgreSQL [18] como gestor de bases de datos.

Concepto	Arquetipo	Acciones
Frecuencia cardíaca	openEHR-EHR-OBSERVATION.frecuencia_cardiaca.v1	[seleccionar concepto]
frecuencia respiratoria	openEHR-EHR-OBSERVATION.frecuencia_respiratoria.v1	[seleccionar concepto]
Presión arterial	openEHR-EHR-OBSERVATION.presion_arterial.v1	[seleccionar concepto]
Escala de coma de Glasgow	openEHR-EHR-OBSERVATION.glasgow_coma.v1draft	[seleccionar concepto]
Vías venosas prehospitalario	openEHR-EHR-ACTION.vias_venosas_prehospitalario.v1	[seleccionar concepto]
Reposición prehospitalaria	openEHR-EHR-ACTION.reposicion_prehospitalaria.v1	[seleccionar concepto]
ubicacion geografica	openEHR-EHR-ADMIN_ENTRY.ubicacion_geografica.v1	[seleccionar concepto]
administracion de sustancias	openEHR-EHR-ACTION.administracion_de_sustancias.v1	[seleccionar concepto]
Diagnosticos de trauma	openEHR-EHR-OBSERVATION.diagnosticos.v1	[seleccionar concepto]
Triage de trauma	openEHR-EHR-EVALUATION.triage_trauma.v2	[seleccionar concepto]
Prehospitalario	openEHR-EHR-ACTION.prehospitalario.v2	[seleccionar concepto]

Fig. 12: búsqueda semántica: selección del concepto²

Concepto	Arquetipo	Acciones
Triage de trauma	openEHR-EHR-EVALUATION.triage_trauma.v2	[seleccionar concepto]

desde:

Nombre	Tipo RM	Tipo AM	Path	Seleccionar
Triage de trauma	EVALUATION	CComplexObject	/	<input type="checkbox"/>
Arbol	ITEM_TREE	CComplexObject	/data[at0001]	<input type="checkbox"/>
Evaluación de triage	ELEMENT	CComplexObject	/data[at0001]/items[at0002]	<input checked="" type="checkbox"/>
-	DvOrdinal	CDvOrdinal [[local] 1]at0003, [local] 2]at0004, [local] 3]at0005, [local] 4]at0006, [local] 5]at0007	/data[at0001]/items[at0002]/value	<input type="checkbox"/>
Justificación de evaluación	ELEMENT	CComplexObject	/data[at0001]/items[at0008]	<input checked="" type="checkbox"/>
-	DV_TEXT	CComplexObject	/data[at0001]/items[at0008]/value	<input type="checkbox"/>
Fecha y Hora de evaluación	ELEMENT	CComplexObject	/data[at0001]/items[at0009]	<input checked="" type="checkbox"/>
-	DV_DATE_TIME	CComplexObject	/data[at0001]/items[at0009]/value	<input type="checkbox"/>
-	DvDateTime	CPrimitiveObject	/data[at0001]/items[at0009]/value/value	<input type="checkbox"/>

Seleccionar paths				
		<input type="checkbox"/>		
66	31-12-2011 00:10:00	Evaluación de triage estable	Justificación de evaluación qwertyu	Fecha y Hora de evaluación 2011-12-31 12:09:00
62	31-12-2011 00:10:00	Evaluación de triage estable con lesiones a tratar	Justificación de evaluación khgfd	Fecha y Hora de evaluación 2011-12-31 12:09:00
58	31-12-2011 00:10:00	Evaluación de triage inestable, grave	Justificación de evaluación oigfjhg	Fecha y Hora de evaluación 2011-12-31 12:09:00
54	31-12-2011 00:10:00	Evaluación de triage estable	Justificación de evaluación kjhgfkjhg	Fecha y Hora de evaluación 2011-12-31 12:09:00
20	31-12-2011 00:08:28	Evaluación de triage estable con lesiones a tratar	Justificación de evaluación ddfgdfgdg	Fecha y Hora de evaluación 2011-12-31 12:09:00

Ver agregacion

Fig. 13: búsqueda semántica: selección de campos

² EHRGen v0.7: búsqueda semántica
<http://informatica-medica.blogspot.com/2012/01/nuevo-open-ehrgen-v07.html>

C. Trabajo futuro

EHRGen fue diseñado para la creación de sistemas de HCE basados en web, pero existe la necesidad de soportar otras plataformas como dispositivos móviles y aplicaciones de escritorio. Actualmente el componente GuiGen se encuentra en un proceso de rediseño para soportar la generación de interfaz de usuario para distintas tecnologías y dispositivos, entre estos se busca generar aplicaciones para dispositivos Android y aplicaciones de escritorio con tecnologías Java como Swing y SWT. De esta forma se agregaría aún más flexibilidad a la herramienta pudiendo generar distintos tipos de aplicaciones finales, reutilizando todos los componentes ya desarrollados en EHRGen: Data Binder, Information Model, Workflow Manager, Knowledge Access y Demographic Access.

También se está planificando el poder agregar definiciones de procesos como elementos en la base de conocimiento. El objetivo es que EHRGen considere estas definiciones para poder generar menús y recorridos entre las distintas pantallas de registro, tal que se adapten al flujo de trabajo del usuario, mejorando la experiencia del mismo al utilizar el sistema. De la misma forma que con los arquetipos, se buscará que los procesos clínicos sean modelados por expertos del dominio clínico, utilizando una herramienta de edición de procesos.

Por último, se están realizando pruebas con mensajería HL7 con el objetivo de facilitar la integración de EHRGen en entornos que ya soportan HL7. Existen tres áreas de especial interés para la integración mediante mensajería: ingresos y egresos de pacientes (ADT), órdenes de estudios y resultados (laboratorio e imagenología), y la prescripción y dispensación de medicamentos. Es probable que en la próxima versión de EHRGen permita un primer nivel de integración utilizando mensajería HL7.

VI. Conclusión

En este trabajo se investigaron los problemas y necesidades más frecuentes en los proyectos de informatización en el área clínica, detectando que algunos de estos problemas son introducidos o fomentados por la forma en la que los profesionales informáticos desarrollamos los sistemas de información. Esto se debe a que los procesos de desarrollo de software más comunes no se adaptan a la dinámica de las instituciones sanitarias ni a la forma de trabajo y cultura de los profesionales de la salud. Y se buscó hacer foco en los proyectos a largo plazo y gran escala, donde los tiempos y costos en la construcción, y sobre todo en la evolución de los sistemas, puede atentar contra la sostenibilidad de este tipo de proyecto, antes de lograr un impacto positivo en la salud de los pacientes, o incluso de lograr algún retorno de la inversión realizada, por ejemplo en la mejora de la gestión de los pacientes (nivel individual) o de las enfermedades (nivel de la población). Estos problemas fueron abordados desde un enfoque que modifica radicalmente el proceso de desarrollo habitual, centrándose en la aplicación de estándares en el diseño de los sistemas, y en la gestión del conocimiento clínico por parte de expertos en el dominio, ya no más por profesionales informáticos. Este último punto es posible gracias a la aplicación del estándar openEHR, que formaliza los elementos de conocimiento y la forma de gestionarlos, proponiendo un verdadero cambio de paradigma en la construcción de sistemas de información en salud, centrándose en la

mantenibilidad de los sistemas a largo plazo y en el uso eficiente de los recursos (humanos, económicos y temporales).

El modelado de conceptos clínicos como arquetipos openEHR, realizado directamente por personal de la salud, presenta numerosas mejoras al proceso de desarrollo de software, entre ellas:

- Mayor participación de los profesionales sanitarios en el desarrollo de la herramienta que luego utilizarán.
- Los conceptos clínicos son gestionados por personal clínico, ya no por informáticos que no están preparados para dicho fin.
- Disminución de errores de interpretación o relevamiento incompleto en los requerimientos sobre el registro clínico.
- Formalización del registro clínico, con capacidad de reutilización de los conceptos entre distintos sistemas, distintas instituciones, incluso en distintos países.
- Mejora de la comunicación entre el equipo clínico y el informático, ahora se cuenta con un nuevo elemento que actúa de interfaz: el arquetipo. Los clínicos conocen el contenido, y los informáticos pueden procesar las estructuras por software.

Como riesgos del enfoque propuesto, se plantean algunas necesidades para que éste sea viable:

- Capacitar profesionales sanitarios en el modelado de conceptos clínicos, en el uso de herramientas de modelado y en el modelo dual del estándar openEHR.
- Crear equipos de profesionales clínicos para gestionar el conocimiento clínico del dominio de trabajo que se defina (una institución, una federación de instituciones o a nivel país).
- Capacitar profesionales informáticos debido a que el proceso de desarrollo cambia radicalmente, además deben crear herramientas para gestionar los elementos en la base de conocimiento.

Todas estas necesidades son subsanables con la debida capacitación y organización interna de los equipos de trabajo.

El proyecto EHRGen plasma el enfoque de openEHR en una herramienta de software concreta, permitiendo generar sistemas de registro de información clínica, basados en openEHR, heredando las características antes mencionadas. La herramienta de software desarrollada es acompañada por la propuesta de un proceso formal de creación y mantenimiento de sistemas de registro clínico centrados a la gestión del conocimiento.

Este proceso de desarrollo orientado a la gestión del conocimiento clínico permite formalizar tareas y especializar el trabajo de los distintos roles que participan en el proyecto, con responsabilidades bien definidas y separadas. Estos elementos no están del todo claros en los procesos usuales de desarrollo de sistemas de información en salud, llevando a que algunos roles desarrollen tareas para las cuales no están preparados, como por ejemplo los analistas informáticos que deben extraer, interpretar y formalizar el conocimiento de los médicos en forma de requerimientos, afectando negativamente a la calidad de los

mismos por errores (muchas veces conceptuales) debido a no manejar la terminología médica y a no conocer los procesos asistenciales.

El proceso de trabajo planteado en este artículo permite evitar estos problemas, además de automatizar tareas, acelerando los tiempos de desarrollo y disminuyendo los costos por las modificaciones más frecuentes (registro clínico) en el mantenimiento del sistema, permitiendo crear y mantener actualizados los sistemas a largo plazo. Actualmente EHRGen permite generar la interfaz de usuario a partir de los arquetipos y plantillas en la base de conocimiento, y en el futuro se incluirá la generación de flujos de trabajo a partir de definiciones de procesos asistenciales que también estarán especificados en la base de conocimiento. Lo que se busca es evolucionar la aplicación con la mínima modificación manual al código fuente del software, con la mayor automatización y reutilización posible de objetos de conocimiento.

La clave del proceso está en la (re)utilizando los conceptos clínicos ya modelados en la base de conocimiento, los que definen la estructura del registro clínico. Ahora la gestión del conocimiento pasa a ser responsabilidad de profesionales sanitarios, ya no más de los profesionales informáticos, los cuales pasan a enfocarse en mejorar el software.

EHRGen es una herramienta de código abierto, puede ser descargada libremente, su código puede ser modificado según necesidades particulares y utilizar libremente respetando la licencia Apache 2.0 [5].

Referencias

- [1] T. Beale, *Archetypes: Constraint-based Domain Models for Future-proof Information Systems*, 2002.
http://www.openehr.org/publications/archetypes/archetypes_beale_oopsla_2002.pdf
- [2] Estándar openEHR v1.0.2
<http://www.openehr.org/releases/1.0.2/roadmap.html>
- [3] Thomas Beale and Sam Heard, *openEHR Architecture – Architecture Overview*, 2008.
<http://www.openehr.org/releases/1.0.2/architecture/overview.pdf>
- [4] T. Beale, S. Heard, D. Kalra, D. Lloyd, *openEHR EHR Information Model*, 2008.
http://www.openehr.org/releases/1.0.2/architecture/rm/ehr_im.pdf
- [5] Apache License v2.0 <http://www.apache.org/licenses/LICENSE-2.0.html>
- [6] OpenEHR Clinical Knowledge Manager (CKM)
<http://openehr.org/knowledge>
- [7] Ocean Archetype Editor
<https://wiki.oceaninformatics.com/confluence/display/TTL/Archetype+Editor+Releases>
- [8] Archetype Definition Language
<http://www.openehr.org/releases/1.0.2/architecture/am/adl.pdf>
- [9] Fussell, M.L., *Foundations of Object Relational Mapping. White Paper*, ChiMu Corporation (1997).
- [10] Grails Object-Relational Mapping
<http://grails.org/doc/latest/guide/GORM.html>
- [11] Pressman, Roger S. (2003). «El proceso». *Ingeniería del Software, un enfoque Práctico, Quinta edición edición..* Mexico: Mc Graw Hill.
- [12] P. Pazos, L. Carrasco, *Traumagen: historia clínica electrónica con acceso a estudios radiológicos digitales, especializada en la atención de pacientes gravemente traumatizados*, 2010.
- [13] A. Prieto Padrón, Á. Rodríguez León, *Sistema de Historia Médica Electrónica para SOS Telemedicina*, Venezuela, 2012.
- [14] Grails Framework
<http://grails.org>
- [15] Java Programming Language
<http://www.oracle.com/us/technologies/java/overview/index.html>

[16] Groovy, An agile dynamic language for the Java Platform

<http://groovy.codehaus.org>

[17] MySQL Database

<http://www.mysql.com>

[18] PostgreSQL Database

<http://www.postgresql.org>

Datos de Contacto:

Pablo Pazos. openEHR en español Montevideo, Uruguay. pablo@openehr.org.es