

# **Editor de guías clínicas (GDL) basadas en arquetipos**

HISTORIAL DE REVISIONES			
NÚMERO	FECHA	MODIFICACIONES	NOMBRE
0.0.1	Julio 2017		JBC

# Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Contextualización . . . . .	1
1.2. Objetivos . . . . .	2
<b>2. Fundamentos tecnológicos</b>	<b>2</b>
2.1. Aplicaciones Web . . . . .	2
2.2. Patrones de diseño . . . . .	2
2.3. Patrones de arquitectura: AngularJS . . . . .	2
<b>3. Estado del arte</b>	<b>2</b>
<b>4. Métodos</b>	<b>2</b>
<b>5. Desarrollo</b>	<b>2</b>
5.1. Iteración 1 . . . . .	3
5.1.1. Análisis . . . . .	3
5.1.2. Diseño . . . . .	3
5.1.3. Implementación . . . . .	3
5.1.4. Pruebas . . . . .	3
5.2. Iteración 2 . . . . .	3
5.2.1. Análisis . . . . .	3
5.2.2. Diseño . . . . .	3
5.2.3. Implementación . . . . .	3
5.2.4. Pruebas . . . . .	3
<b>6. Testing</b>	<b>3</b>
6.1. Jasmine Framework . . . . .	3
6.2. Tests unitarios con Karma . . . . .	3
6.3. Tests end-to-end con Protractor . . . . .	4
<b>7. Material utilizado</b>	<b>4</b>
<b>8. Implantación y seguimiento (si corresponde)</b>	<b>4</b>
<b>9. Planificación y costes (planificación inicial vs. final) - (Opcional)</b>	<b>4</b>
<b>10. Resultados y discusión</b>	<b>4</b>
<b>11. Conclusiones</b>	<b>4</b>
<b>12. Futuros trabajos</b>	<b>4</b>

---

<b>13. Bibliografía</b>	<b>5</b>
13.1. Referencias . . . . .	5
<b>14. Glosario</b>	<b>5</b>
<b>A. Apéndices</b>	<b>5</b>
A.1. Manual de usuario . . . . .	5
A.2. Ejemplo de creación de una guía clínica . . . . .	5

## Resumen

Expresar y compartir contenido de soporte a la decisión clínica (CDS por sus siglas en inglés) de forma automatizada e independiente del lenguaje y de plataformas técnicas ha sido una tarea compleja durante mucho tiempo. Los dos principales motivos para compartir lógica de decisión entre diferentes sistemas son, por un lado, la carencia de modelos de información clínica compartidos y, por otro, el soporte flexible para los diferentes recursos terminológicos.

Para expresar la lógica que da soporte a la decisión clínica se necesita un modelo de representación de guías clínicas. Para ello la comunidad *openEHR*, comunidad virtual que trabaja en el desarrollo de especificaciones para la interoperabilidad de registros clínicos electrónicos entre sistemas sanitarios, ha desarrollado la especificación del lenguaje GDL (Guideline Definition Language). GDL un lenguaje formal diseñado para expresar lógica de soporte a la decisión médica, el cual, recientemente, ha pasado a formar parte del núcleo de la plataforma. Ha sido diseñado para ser independiente de lenguaje naturales y de terminologías clínicas haciendo uso, para ello de los diseños del Modelo de Referencia del Modelo de arquetipos de *openEHR*.

El principal objetivo de este proyecto ha sido la elaboración de una herramienta Web para la edición de guías clínicas expresadas en lenguaje GDL, la comunidad no disponía de una herramienta de gestión de dichas guías con un enfoque colaborativo como el que puede ofrecer una aplicación Web. Lo que se pretende es que sean los propios profesionales sanitarios, los que tienen el conocimiento necesario, los que elaboren las guías clínicas de una manera colaborativa y consensuada, con el fin de que puedan utilizarlas de manera exitosa en la práctica médica.

El resultado que se ha obtenido es un software capaz de crear y gestionar guías en formato GDL, las guías son validadas en el backend de la aplicación, denegándose la petición de envío cuando la guía posee algún error de elaboración/sintaxis. Se ha desarrollado una aplicación modular en AngularJS, framework Javascript de Google para el desarrollo de páginas web de una sola página basado en el patrón de arquitectura Modelo-Vista-Controlador, el cual nos ha permitido el desarrollo de una serie de controladores y servicios bien definidos como veremos en los siguientes capítulos. Dicho desarrollo modular permite una sencilla reutilización de sus componentes, así como el añadido y eliminación de otras funcionalidades utilizando la metodología de desarrollo ágil Scrum, que nos ha permitido reducir considerablemente los ciclos de desarrollo y ha asegurado que al final de cada ciclo se hubiesen desarrollado ciertas funcionalidades operativas de aplicación.

### Palabras clave

Guías clínicas, arquetipos, sistema de soporte a la decisión, interoperabilidad.

## 1. Introducción

La toma de decisiones es el proceso mediante el que se realiza la elección de una alternativa para resolver un problema. La toma de decisiones está presente cotidianamente tanto en la actividad humana como en la de una organización. La clave del éxito está en tomar buenas decisiones. El concepto de apoyo a las decisiones es muy antiguo [keen] y ha evolucionado principalmente desde dos áreas de investigación: los estudios teóricos de organización de la toma de decisiones, desarrollados en el Carnegie Institute of Technology a principios de los años 60 y el trabajo técnico sobre sistemas informáticos interactivos, llevados a cabo principalmente en el Instituto Tecnológico de Massachusetts en la década de los 60. Posteriormente, en la década de los 70, los sistemas de apoyo a las decisiones pasan a convertirse en un campo de investigación como tal.

Es necesario que se cumplan ciertos requisitos para tener éxito tomando decisiones, como son:

- Conocimiento en profundidad del dominio de la organización
- Experiencia
- Información disponible:
  - abarcable por un conjunto de personas
  - no siempre orientable al proceso analítico
- Uso de metodologías de análisis sencillas e intuitivas

### 1.1. Contextualización

En el ámbito de la salud las decisiones clínicas se toman mediante razonamiento deductivo utilizando el conocimiento que tienen los profesionales sanitarios sobre la fisiopatología humana teniendo en cuenta la información disponible para cada caso concreto.

Parece bastante obvio que la experiencia de otros médicos y profesionales de la salud puede ayudar a la toma de decisiones. A principios de los años 60 ya se empezaron a dedicar esfuerzos para crear sistemas de soporte a la decisión clínica [bonis]. Incluso una década antes, se dedicaron muchos esfuerzos a realizar diagnósticos a partir de la información que los médicos habían almacenado en sus computadoras, se basaban en reglas lógicas básicas. Dos ejemplos clásicos son el sistema de diagnóstico de alteraciones electrolíticas de Bleich [bleich] y el realizado por Warner et al. [warner] basado en la realización de razonamientos probabilísticos mediante el uso intensivo del teorema de Bayes. Hasta este momento se fundaron lo que fueron las bases teóricas del desarrollo de los SSDC. Durante las dos décadas siguientes se desarrollaron los primeros SSDC basados en grandes ordenadores. Sin embargo no fue hasta los 80 y 90, con la llegada de los ordenadores personales y el uso generalizado de los mismos, cuando se aprovecharon potencialmente este tipo de sistemas. Hoy en día el soporte a la decisión clínica forma parte de muchos sistemas sanitarios. <Comentar la situación de los SSD clínicos actuales>

Los sistemas de salud actuales requieren para sí una característica fundamental para garantizar la continuidad asistencial<sup>1</sup>. Para conseguir esto es inevitable que la información clínica sea representada de una manera interoperable, de modo que diferentes organizaciones puedan hacer uso de dicha información independientemente de quién, cómo y cuándo se haya generado. La continuidad asistencial del paciente es indispensable en multitud de situaciones como cambios de residencia de los pacientes, cambios organizativos, cambios de niveles asistenciales, etc. La manera de conseguir la deseada interoperabilidad es aplicando la normalización. Las normas proporcionan la base sobre la que poder crear acuerdos con un objetivo en común, en este caso la continuidad asistencial.

Dificultad de conseguir la deseada interoperabilidad en entornos sanitarios: en el ámbito médico las organizaciones actúan, en cierto modo, como silos de información aislados, es decir, son autónomos a la hora de tomar sus propias decisiones, de implantar los sistemas de información que cada una de ellas quiere. Esto se repite incluso dentro de las propias organizaciones, un ejemplo claro podrían ser los propios servicios dentro de un hospital que, en numerosas ocasiones, utilizan sistemas distintos. Surge, de esta manera, la necesidad de crear una metodología que nos permita representar el conocimiento médico y, de este modo, poder separarlo de la información clínica existente en los sistemas. .

---

#### nota

An admonition paragraph draws the reader's attention to auxiliary information. Its purpose is determined by the label at the beginning of the paragraph.

---

<sup>1</sup> **Continuidad asistencial:** concepto que define el proceso, centrado en el paciente, en el cual intervienen diferentes profesionales sanitarios en diferentes periodos de tiempo y lugares con el objetivo compartido de mejorar la calidad de la asistencia prestada.

---

## 1.2. Objetivos

El objetivo principal de este proyecto ha consistido en el desarrollo de una aplicación web para la gestión de clínicas basadas en arquetipos y expresadas en lenguaje GDL. Una herramienta de este tipo permite a los profesionales sanitarios generar conocimiento clínico interoperable al hacer uso de las tecnologías web de una manera colaborativa y consensuada.

## 2. Fundamentos tecnológicos

Fundamentos tecnológicos

### 2.1. Aplicaciones Web

Aplicaciones Web y Single Page Applications

### 2.2. Patrones de diseño

Patrones de diseño utilizados en el desarrollo.

- Servicios
- Factorías
- Inyección de dependencias

### 2.3. Patrones de arquitectura: AngularJS

Patrón de arquitectura MVC aprovechando las características de AngularJS.

- Módulos en AngularJS
- Templates
- Controllers
- Objeto `$scope` de AngularJS
- Two-way databinding

## 3. Estado del arte

Estado de la cuestión relacionado con la interoperabilidad clínica. Normas UNE EN ISO-13606 y *openEHR*: Modelos de referencia y arquetipos, modelo de guías clínicas GDL. Arquetipos clínicos. Terminologías clínicas: SNOMED-CT, ICD10, LOINC, ICPC...

## 4. Métodos

Metodologías ágiles. Scrum

## 5. Desarrollo

Número de iteraciones, hablar de cada una de ellas

---

## **5.1. Iteración 1**

Text iteración 1

### **5.1.1. Análisis**

Texto análisis iteración 1

### **5.1.2. Diseño**

Texto diseño iteración 1

### **5.1.3. Implementación**

Texto implementación iteración 1

### **5.1.4. Pruebas**

Texto pruebas iteración 1

## **5.2. Iteración 2**

Texto análisis iteración 2

### **5.2.1. Análisis**

Texto análisis iteración 2

### **5.2.2. Diseño**

Texto diseño iteración 2

### **5.2.3. Implementación**

Texto implementación iteración 1

### **5.2.4. Pruebas**

Texto pruebas iteración 2

## **6. Testing**

### **6.1. Jasmine Framework**

hablar del framework Jasmine (utilizado en los dos siguientes testrunners)

### **6.2. Tests unitarios con Karma**

Utilización de Karma test runner para probar nuestros servicios y controladores.

---



### 6.3. Tests end-to-end con Protractor

Hablar del framework de tests Protractor y mencionar cómo lo hemos aplicado para realizar pruebas end-to-end en nuestro proyecto.

## 7. Material utilizado

- Sistema operativo Ubuntu 14.04, GNOME 3.
- IDE JetBrains WebStorm.
- NodeJS como entorno de ejecución de la aplicación cliente.
- SGBD PostgreSQL.
- Servidor de aplicaciones JBoss Server EAP 6.2.
- Astah Community Edition como herramienta de modelado UML.
- AsciiDoc como herramienta de generación de documentos.
- Git/Github como sistema de control de versiones y repositorio remoto de código.
- Open Project como software de gestión de proyectos.
- Travis CI como herramienta de integración continua.

## 8. Implantación y seguimiento (si corresponde)

Texto implantación y seguimiento

## 9. Planificación y costes (planificación inicial vs. final) - (Opcional)

Texto planificación y costes

## 10. Resultados y discusión

Mencionar y discutir los resultados obtenidos, tanto los que se pudieron abordar como los que no.

## 11. Conclusiones

Conclusiones finales sobre el proyecto.

## 12. Futuros trabajos

La herramienta supone el punto de partida hacia una serie de nuevas funcionalidades de gran utilidad para la comunidad. coo la edición colaborativa en tiempo real, la resolución de conflictos, la gestión del historial y de versiones de las guías clínicas, etc.

---

## 13. Bibliografía

### 13.1. Referencias

- [1] [keen] Keen, P. G. W. Decision support systems: an organizational perspective. Addison-Wesley. ISBN 0-201-03667-3.
- [2] [bonis] Bonis Julio, Sancho Juan J., Sanz Ferrán. Computer-assisted Clinical Decision Support Systems. Clinical Medicine, Elsevier Ed. 2004;122 Supl 1:39-44.
- [3] [bleich] Bleich HL. Computer evaluation of acid-base disorders. J Clin Invest, 48 (1969), pp. 1689-96 <http://dx.doi.org/10.1172/JCI106134>
- [4] [warner] Warner HR, Toronto AF, Veasey LG, Stephenson RA. Mathematical approach to medical diagnosis. JAMA, 177 (1961), pp. 75-81.

## 14. Glosario

Glosario utilizado en el documento, con respectivos significados

### A glossary term

The corresponding (indented) definition.

### A second glossary term

The corresponding (indented) definition.

## A. Apéndices

One or more optional appendixes go here at section level 1.

### A.1. Manual de usuario

Manual de usuario de la herramienta con capturas de pantalla.

### A.2. Ejemplo de creación de una guía clínica

Desarrollo, paso a paso, de una guía clínica con algo de complejidad, con capturas de pantalla.

---