

# **Prototipo de Interoperabilidad con IRIS de InterSystems para el intercambio de alergias de pacientes**

Realizado por

**Armand Molina Trabal**

Tutor del proyecto

**Eugenio Fernández González**

**CURSO 2023 - 2024**

**Grado en Ingeniería Informática de Gestión y Sistemas de Información**



## **Agradecimientos**

Quiero agradecer en primera instancia a mi tutor, Eugenio Fernández González, por hacer posible la realización de este proyecto, su experiencia y conocimientos han sido fundamentales para el desarrollo de este trabajo.

Me gustaría hacer también una mención especial a InterSystems por su apoyo constante durante el proyecto, en particular a Pierre-Yves Duquesnoy, quien ha sido un profesional ejemplar. Gracias a su experiencia, he aprendido mucho y su colaboración ha sido vital para la culminación de este proyecto.



## **Abstract**

This project involves implementing a technical interoperability system between two hospitals using InterSystems' IRIS platform to enable secure and efficient data exchange, with a specific focus on patient allergies. By configuring IRIS and hospital environments, creating and populating databases, the system aims to facilitate seamless sharing of clinical data as if the hospitals were a single entity, thereby improving patient care quality, operational efficiency, and the overall experience for medical staff and patients by reducing errors and redundancies in data handling.

## **Resum**

Aquest projecte implica la implementació d'un sistema tècnic d'interoperabilitat entre dos hospitals utilitzant la plataforma IRIS d'InterSystems per permetre l'intercanvi segur i eficient de dades, amb un enfocament específic en les al·lèrgies dels pacients. Mitjançant la configuració d'IRIS i els entorns hospitalaris, la creació i l'ompliment de bases de dades, el sistema té com a objectiu facilitar l'intercanvi fluït de dades clíniques com si els hospitals fossin una sola entitat, millorant així la qualitat de l'atenció al pacient, l'eficiència operativa i l'experiència general del personal mèdic i els pacients al reduir errors i redundàncies en el maneig de dades.

## **Resumen**

Este proyecto implica la implementación de un sistema técnico de interoperabilidad entre dos hospitales utilizando la plataforma IRIS de InterSystems para permitir el intercambio seguro y eficiente de datos, con un enfoque específico en las alergias de los pacientes. Mediante la configuración de IRIS y los entornos hospitalarios, la creación y el llenado de bases de datos, el sistema tiene como objetivo facilitar el intercambio fluido de datos clínicos como si los hospitales fueran una sola entidad, mejorando así la calidad de la atención al paciente, la eficiencia operativa y la experiencia general del personal médico y los pacientes al reducir errores y redundancias en el manejo de datos.



# Índice

<b>Índice de figuras</b>	<b>v</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Objeto del proyecto . . . . .	1
1.2. Justificación del proyecto . . . . .	1
<b>2. Marco Teórico</b>	<b>3</b>
2.1. Sistemas de información sanitarios . . . . .	3
2.2. Interoperabilidad entre sistemas sanitarios . . . . .	3
2.3. Estándares y protocolos de interoperabilidad . . . . .	4
<b>3. Objetivos y alcance</b>	<b>5</b>
3.1. Propósito . . . . .	5
3.2. Objetivos . . . . .	5
3.2.1. Objetivos del proyecto . . . . .	5
3.2.2. Objetivos del producto . . . . .	6
3.2.3. Objetivos personales . . . . .	6
3.3. Alcance . . . . .	7
3.3.1. Límites . . . . .	7
<b>4. Análisis de referentes</b>	<b>9</b>
4.1. Investigación de referentes . . . . .	9
4.1.1. InterSystems . . . . .	10
4.1.2. European Health Data Space . . . . .	10
4.1.3. Ejemplos actuales . . . . .	11
<b>5. Metodología</b>	<b>13</b>
5.1. Jira Software . . . . .	13
5.2. MySQL & SQL Server . . . . .	13
5.2.1. MySQL WorkBench & SQL Server Management Tools . . . . .	14
5.3. Generación de datos . . . . .	14
5.3.1. Mockaroo . . . . .	14

5.4.	Microsoft Access . . . . .	15
5.5.	Lucidchart . . . . .	15
5.6.	IRIS de InterSystems: Edición Community . . . . .	15
<b>6.</b>	<b>Desarrollo</b>	<b>17</b>
6.1.	Prototipo de interoperabilidad . . . . .	17
6.1.1.	Esquema de extremo a extremo . . . . .	17
6.1.2.	Esquema conceptual . . . . .	19
6.2.	Hospital 1 . . . . .	20
6.3.	Hospital 2 . . . . .	24
6.4.	IRIS . . . . .	28
6.5.	Entrada de datos . . . . .	31
6.5.1.	Mockaroo fake data generator . . . . .	31
6.5.2.	Formulario Microsoft Access . . . . .	33
6.5.3.	Entrada directa a base de datos . . . . .	35
6.6.	Caso de uso . . . . .	37
6.7.	Flujo de funcionamiento . . . . .	39
6.7.1.	IRIS Data Platform . . . . .	42
6.8.	Envío de información . . . . .	44
6.9.	Testing . . . . .	44
6.9.1.	Creación de nueva alergia de paciente en Hospital 1 . . . . .	45
6.9.2.	Creación de nueva alergia de paciente en Hospital 2 . . . . .	49
6.9.3.	Creación de una alergia de un paciente con vigencia exenta . . . . .	51
6.9.4.	Creación de registros incorrectos . . . . .	52
<b>7.</b>	<b>Conclusiones</b>	<b>55</b>
7.1.	Resumen de los resultados obtenidos . . . . .	55
7.2.	Implicaciones teóricas y prácticas . . . . .	55
7.2.1.	Teóricas . . . . .	55
7.2.2.	Prácticas . . . . .	56
7.3.	Limitaciones y posibles mejoras . . . . .	56
7.3.1.	Escala del prototipo . . . . .	56
7.3.2.	Diversidad de sistemas . . . . .	56

7.3.3. Recursos y tiempo . . . . .	56
7.4. Recomendaciones para futuras investigaciones . . . . .	57
7.4.1. Ampliar el alcance del estudio . . . . .	57
7.4.2. Integración con tecnologías de IRIS . . . . .	57
7.4.3. Evaluación de impacto . . . . .	57
7.5. Conclusión final . . . . .	57
<b>8. Bibliografía</b>	<b>59</b>



## Índice de figuras

6.1. Esquema de extremo a extremo del prototipo . . . . .	17
6.2. Esquema del proyecto de interoperabilidad . . . . .	19
6.3. Esquema de Hospital 1 . . . . .	20
6.4. Diseño de base de datos de Hospital 1 . . . . .	21
6.5. Esquema de Hospital 2 . . . . .	24
6.6. Diseño de base de datos de Hospital 2 . . . . .	25
6.7. Esquema de IRIS . . . . .	28
6.8. Esquema de funcionamiento de IRIS . . . . .	29
6.9. Esquema de funcionamiento completo de IRIS . . . . .	30
6.10. Campos de datos con su tipo . . . . .	31
6.11. Tipos de campos . . . . .	32
6.12. Formulario alergia paciente de Hospital 1 . . . . .	33
6.13. Formulario alergia paciente de Hospital 1: paciente_NHC . . . . .	34
6.14. Formulario alergia paciente de Hospital 1: alergia_id . . . . .	34
6.15. Formulario alergia paciente de Hospital 1: fecha_diagnostico . . . . .	35
6.16. Consulta insert en Hospital 1 . . . . .	35
6.17. Consulta insert en Hospital 1 . . . . .	36
6.18. Proceso de introducción de una nueva alergia de un paciente . . . . .	37
6.19. Flujo de funcionamiento en origen . . . . .	39
6.20. Flujo de funcionamiento en IRIS . . . . .	40
6.21. Flujo de funcionamiento en destino . . . . .	41
6.22. Configuración de producción de IRIS . . . . .	42
6.23. Configuración de servicio de lectura de evento . . . . .	43
6.24. Configuración de respuesta en el proceso . . . . .	43
6.25. Inserción de la alergia del paciente en el formulario . . . . .	45
6.26. Consulta de la tabla evento en hospital de origen . . . . .	45
6.27. Consulta de la tabla evento en hospital de origen . . . . .	46
6.28. Traza del mensaje de registro de la alergia: lectura registro . . . . .	46
6.29. Traza del mensaje de registro de la alergia: procesamiento del mensaje hacía operación . . . . .	47
6.30. Traza del mensaje de registro de la alergia: respuesta de la operación . . . . .	47

6.31. Consulta de la tabla de evento en hospital de destino . . . . .	48
6.32. Consulta de la tabla de registro de alergias de paciente en hospital de destino . . . . .	48
6.33. Consulta de la tabla de registro de alergias de paciente en hospital de destino . . . . .	48
6.34. Consulta de inserción en base de datos de Hospital 2 . . . . .	49
6.35. Consulta de la tabla de evento de Hospital 2 . . . . .	50
6.36. Traza del mensaje en IRIS . . . . .	50
6.37. Consulta de la tabla evento en el Hospital 1 . . . . .	51
6.38. Inserción de alergia de paciente ya existente con vigencia exenta . . . . .	51
6.39. Consulta de alergia de paciente en el registro de alergias del Hospital 2 . . . . .	52
6.40. Consulta de inserción en Hospital 2 con alergia inexistente . . . . .	52
6.41. Consulta de inserción en Hospital 2 con paciente inexistente . . . . .	53
6.42. Inserción de alergia de paciente con paciente inexistente . . . . .	53
6.43. Inserción de alergia de paciente con alergia inexistente . . . . .	54

# 1. Introducción

## 1.1. Objeto del proyecto

El objeto de este proyecto es la creación de un prototipo de Interoperabilidad para el intercambio de alergias, el cual se enfoque en la problemática de la comunicación entre diversos sistemas heterogéneos.

Existe una gran diversidad de tecnologías implantadas para soportar una gran cantidad de sistemas de información, estas diferencias hacen que la comunicación entre sí sea más complicada, es por eso que surge la interoperabilidad, para hacer esas diferencias inexistentes.

La interoperabilidad es la base tecnológica para resolver este problema.

Con la finalización de este proyecto se espera crear el prototipo de una solución técnica, robusta y eficaz, que resuelva los posibles problemas en el intercambio de alergias de pacientes entre distintos sistemas de información sanitarios.

## 1.2. Justificación del proyecto

La necesidad de contar con un sistema eficiente y efectivo para el intercambio de alergias de pacientes entre diferentes sistemas de información sanitarios es crucial en el ámbito de la salud. La falta de interoperabilidad puede generar problemas graves, como la incorrecta administración de tratamientos médicos provocando reacciones adversas poniendo en riesgo la salud de los pacientes.

En 2001, el US Department of Health and Human Services estimó que alrededor de 770.000 personas al año son malheridas o incluso mueren a causa de errores de medicación, supone un coste de 5.6 millones de dólares cada año por hospital excluyendo costes de accesorios [11].

¿Qué ocurre en el caso de una emergencia clínica? La solución más fácil y efectiva es la realización de una prueba de alergias, pero ¿es la más veloz? ¿Es siempre útil? En un caso de urgencia en el que el tiempo suele ser escaso, la única solución es actuar y no tienes tiempo para realizar ninguna prueba, mientras que consultar los datos de ese paciente y recurrir a pasadas pruebas de alergias en otros hospitales puede tomar un tiempo mínimo y, por lo tanto, puede salvar muchas vidas.

Ante todo esto se plantea lo siguiente: ¿Es rentable implementar una solución de interoperabilidad? [10]

Teniendo en cuenta un centro clínico, la respuesta es más que afirmativa, poseer la información clínica de pacientes es algo más que importante ante este tipo de emergencias.

La justificación de este proyecto se basa en la importancia de superar las barreras tecnológicas en el ámbito sanitario que impiden una comunicación efectiva entre sistemas de información. La interoperabilidad no solo mejora la eficiencia y precisión en el intercambio de información, sino que también contribuye a la toma de decisiones por parte de los profesionales de la salud.

## 2. Marco Teórico

En la actualidad, el avance continuo de nuevas tecnologías y la diversificación de sistemas de información han creado un entorno en el que la interoperabilidad se vuelve crucial en cualquier sector, no solo en el sanitario. El problema se encuentra en la falta de un mismo formato de los datos, de tecnologías y de estándares de comunicación. Esta falta de estandarización dificulta la transferencia efectiva de información entre sistemas, lo que puede dar lugar a problemas como la pérdida de datos, exceso de esfuerzo o la falta de coherencia en la gestión de la información.

En el ámbito de la salud, donde la precisión de los datos es un factor vital, la falta de interoperabilidad puede tener consecuencias muy significativas. La diversidad de sistemas de información entre hospitales, cada uno con sus propias metodologías, tecnologías y estándares, crea multitud de obstáculos para lograr un intercambio eficiente de información médica.

### 2.1. Sistemas de información sanitarios

Los sistemas de información sanitarios en sí mismos son un conjunto muy complejo y completo de tecnologías, gestión de pacientes, gestión de citas de pacientes, gestión de pruebas de laboratorio, entre otros. Cada uno almacena y gestiona información relacionada sobre el centro clínico, de hecho la mayoría por no decir todos los sistemas dentro de un centro clínico almacenan datos de los pacientes, incluyendo así datos sobre alergias, por lo que esta información es tratada de diversas formas alrededor de un mismo sistema, pero al hablar de un único sistema de información toda la información dentro de él está unificada y sigue una cierta coherencia por lo que los problemas son menores.

Es en el momento de la conexión entre diversos sistemas donde empiezan las complicaciones, ya que los datos, las formas de gestionar los datos, los estándares propuestos, varían según el sistema y de aquí es donde sale el problema que la interoperabilidad aborda.

### 2.2. Interoperabilidad entre sistemas sanitarios

La interoperabilidad en el ámbito de la salud se basa en la capacidad de diferentes sistemas de información para intercambiar datos de manera efectiva, en el caso de este proyecto los datos son históricos de

alergias de pacientes. La diversidad de tecnologías entre hospitales clínicos supone diferentes tipos de comunicación entre sistemas de información, estos son desafíos comunes en el momento de realizar un sistema de interoperabilidad en un entorno sanitario. Comprender y solventar estos desafíos es esencial para diseñar una solución eficaz la cual aborde las complejidades de la comunicación entre sistemas de información.

### **2.3. Estándares y protocolos de interoperabilidad**

Para poder asegurar una comunicación efectiva entre sistemas de información, la adopción de estándares y protocolos de interoperabilidad es crucial para el intercambio de datos de manera comprensible por ambos sistemas. Un ejemplo de estándar es HL7 FHIR, el cual es uno de los más comunes en el ámbito de la salud, la aplicación de estándares facilita la integración, ya que reduce distinciones entre sistemas.

En el supuesto de este proyecto y debido a que sus dimensiones son pequeñas, no será necesaria la implementación de un estándar, el intercambio de información se llevará a cabo a través de el envío de paquetes con información relevante.

## 3. Objetivos y alcance

### 3.1. Propósito

El propósito principal de este proyecto es solventar los problemas de comunicación entre sistemas de información de centros clínicos con distintas características que los distinguen, facilitando así el flujo de datos entre estos y haciendo más eficaz muchos de los procesos que se realizan en un centro clínico, como es el caso de las pruebas para la detección de alergias.

Conseguir que al introducir información sobre una alergia en uno de los sistemas de información, esta alergia de este paciente aparezca en el otro sistema de información, funcionando de manera bidireccional y creando así un marco de interoperabilidad.

### 3.2. Objetivos

#### 3.2.1. Objetivos del proyecto

##### Objetivos principales

- Implementar un sistema de interconexión que permita la transferencia eficiente de datos entre diferentes sistemas de información en hospitales.
- Desarrollar mecanismos de validación y corrección de datos para asegurar la integridad y precisión de la información compartida entre sistemas.

##### Objetivos secundarios

- Establecer un entorno lo más similar a la realidad posible para la simulación de dos hospitales diferentes.
- Integrar un sistema de registro para rastrear cambios y asegurar la trazabilidad de la información.

### **3.2.2. Objetivos del producto**

#### **Objetivos principales**

- Mejorar la eficiencia de los procesos clínicos relacionados con la detección y gestión de alergias, reduciendo tiempos y posibles errores.
- Mantener un modelo lo más similar a la realidad para poder conseguir un prototipo lo más aplicable y escalable posible.

#### **Objetivos secundarios**

- Desarrollar una interfaz de usuario intuitiva y fácil de usar que permita a los profesionales de la salud ingresar y acceder a la información sobre alergias de los pacientes.
- Proporcionar herramientas de análisis de datos que ayuden a los profesionales de la salud a obtener información relevante para la toma de decisiones.

### **3.2.3. Objetivos personales**

#### **Objetivos principales**

- Adquirir experiencia en el diseño e implementación de soluciones para la interoperabilidad de sistemas de información.
- Fomentar la mejora continua mediante la retroalimentación y la adaptación a cambios, contribuyendo al crecimiento personal y profesional.

#### **Objetivos secundarios**

- Profundizar mis conocimientos en estándares de interoperabilidad y tecnologías de seguridad de la información en el ámbito de la salud.
- Desarrollar habilidades de resolución de problemas y toma de decisiones en un contexto de desarrollo de software para el sector sanitario.

### 3.3. Alcance

El alcance de este proyecto abarca la creación de un sistema de interoperabilidad para la transferencia de datos sobre alergias entre sistemas de información hospitalarios. Se centra en garantizar que la información relevante sobre cada paciente sea accesible en diferentes hospitales. El proyecto incluirá:

- Desarrollo de infraestructura entre los sistemas de información de los diferentes hospitales.
- Implementación de mecanismos de validación y corrección de datos para asegurar la calidad de la información transferida.
- Integración de un sistema de registro de transacciones para rastrear cambios y facilitar la revisión de nuevos registros.
- Realización de pruebas exhaustivas para garantizar el correcto funcionamiento del sistema de interoperabilidad.
- Implementar distintas interfaces de usuario para cada hospital con el fin de diferenciarlos.

#### 3.3.1. Límites

Es importante definir claramente los límites del proyecto para gestionar las expectativas y garantizar un enfoque realista. Algunos de los límites del proyecto incluyen:

- No abordar la integración con sistemas externos no relacionados con la gestión de alergias, centrándose exclusivamente en la interoperabilidad en este ámbito.
- No incluir la gestión completa de historias clínicas, centrándose específicamente en la información relacionada con alergias de pacientes.
- No abordar problemas legales y éticos específicos relacionados con la gestión de datos de salud, dejando esos aspectos legales para los centros hospitalarios.

Establecer estos límites contribuirá a una mayor efectividad del proyecto, permitiendo minimizar el foco hacia los objetivos establecidos en primera instancia.



## 4. Análisis de referentes

### 4.1. Investigación de referentes

El Observatorio del Marco Nacional de Interoperabilidad (NIFO) plantea un marco de interoperabilidad (EIF) [1] basado en cuatro capas de interoperabilidad:

- Interoperabilidad Legal: Es la capacidad de los sistemas para cumplir con las leyes y regulaciones aplicables.
- Interoperabilidad Organizacional: Es la capacidad de las organizaciones de trabajar juntas de manera eficaz.
- Interoperabilidad Semántica: Es la capacidad de los sistemas para entender el significado de los datos que intercambian.
- Interoperabilidad Técnica: Es la capacidad de los sistemas para comunicarse entre sí utilizando el mismo lenguaje y protocolos.

Además, plantea un componente transversal, el cual pasa por los cuatro niveles, la “gobernanza integrada de los servicios públicos”. Finalmente, estas capas están dentro de la capa de gobernanza de interoperabilidad.

Una gran variedad de empresas, tanto grandes como pequeñas, operan en el ámbito de la interoperabilidad. Algunas de ellas son:

- IBM
- Microsoft
- Oracle
- SAP
- InterSystems
- Cisco

Entre muchas otras involucradas en todo tipo de sectores.

#### **4.1.1. InterSystems**

InterSystems es una de las empresas más grandes del mundo en cuanto a la interoperabilidad se refiere, destacándose especialmente en el ámbito de la tecnología de la información y la salud. Fundada en 1978 y con sede en Cambridge, Massachusetts, se ha consolidado como un líder en el desarrollo de soluciones que facilitan la conexión y el intercambio de datos entre sistemas heterogéneos [2].

Encargados de resolver algunos de los problemas más críticos relacionados con la integración de datos y la interoperabilidad. La empresa cuenta con una enorme presencia global, con oficinas y socios en más de 25 países, y sus soluciones son utilizadas por miles de organizaciones en todo el mundo [3].

Una de las herramientas más conocidas de InterSystems es IRIS, una plataforma de datos que provee una tecnología de integración sumamente potente y flexible. InterSystems IRIS permite unir datos aislados, conectando procesos, aplicaciones y personas de una manera eficiente y segura. La plataforma soporta múltiples plataformas de gestión de datos, incluyendo SQL, NoSQL y la gestión de datos en memoria, lo que la hace altamente adaptable a diferentes necesidades empresariales. Esta última funcionalidad se podrá ver directamente implementada en el proyecto.

InterSystems IRIS incluye una gran variedad de diversas tecnologías, las cuales abarcan desde interfaces para el intercambio de mensajes y datos en tiempo real, hasta herramientas para la analítica avanzada y el desarrollo de aplicaciones. La plataforma también está diseñada para operar en entornos de nube híbrida, lo que brinda una mayor flexibilidad y escalabilidad a las organizaciones que la utilizan.

Este proyecto en cuestión no utilizará al completo la herramienta debido a que se trata de un proyecto relativamente pequeño, pese a eso, se maximizará el uso de las herramientas todo lo posible.

El sector en el que este proyecto se basará será el de la salud. Más concretamente, en lo que se basará la interoperabilidad que se llevará a cabo será en el campo de las alergias de pacientes. Es por eso que se ha decidido tomar como ejemplo InterSystems, ya que ninguna empresa conoce los datos sanitarios como InterSystems [3]. Más de mil millones de historias clínicas en todo el mundo se gestionan con la tecnología de InterSystems. Las instituciones incluidas en la lista de honor de los mejores hospitales de U.S. News & World Report 2020 utilizan productos de InterSystems.

#### **4.1.2. European Health Data Space**

Se debe mencionar la importancia del European Health Data Space (EHDS) [4]. Es un pilar esencial en el sector de la salud y es el encargado de la comunicación eficiente de datos. Es un ecosistema comprendido

---

por reglas, estándares, prácticas e infraestructuras. Establece un estándar internacional para la gestión de la información sanitaria. El EHDS se posiciona como un líder indiscutible en la mejora de la atención médica a nivel mundial.

### **4.1.3. Ejemplos actuales**

La interoperabilidad está presente en multitud de sectores y campos. En el sector de la salud existen muchos usos de la interoperabilidad. Un ejemplo claro es México, quien ha implantado un expediente clínico unificado en la red de 30 hospitales de la capital. Estos hospitales atienden a una población de 4 millones de habitantes.

Con la ayuda de soluciones como la Historia Clínica Electrónica, y las soluciones para el intercambio de datos de salud, ehCOS Clinical Share y ehCOS EMPI, se ha conseguido en 2 años que todos los sistemas de atención al paciente en los 30 hospitales sean interoperables entre sí, ayudando a mejorar la seguridad y continuidad asistencial del paciente desde cualquier punto de atención de la capital. Con ello, también se ha aumentado la eficiencia del sistema de salud y disminuido costes de repetición de pruebas [6].



## 5. Metodología

### 5.1. Jira Software

Con las tareas confeccionadas se procede a utilizar Jira Software [5] para el seguimiento de las tareas debido a la cantidad de características interesantes que ofrece. Permite escalar sin problemas, sin importar el tamaño del proyecto y se adapta a cualquier entorno. Además de su fácil usabilidad, esta herramienta permitirá llevar un seguimiento claro sobre el avance del proyecto.

Además, Jira Software ofrece una amplia gama de herramientas de colaboración que permiten a los equipos trabajar de manera conjunta y eficiente, con funciones como la asignación de tareas, la visualización de dependencias y la generación de informes personalizados.

En este proyecto Jira ha sido utilizado para realizar un seguimiento de las tareas. Dentro del proyecto creado se ha confeccionado un tablero kanvan, en el cual se han especificado varias columnas sobre las que se irán desplazando las tareas: Por hacer, En curso, TO-check y Listo.

### 5.2. MySQL & SQL Server

Estas herramientas serán algunas de las principales para el desarrollo del proyecto, mas concretamente para la creación de las bases de datos.

Primeramente para el hospital 2 o EHR2, se utilizará MySQL. Este ofrece multitud de funcionalidades [7]:

- Escalabilidad y flexibilidad
- Muy fácil de usar
- Posee un alto rendimiento
- Seguridad en tus datos
- Aplicación de desarrollo integral

Entre muchos otros puntos que la posicionan como uno de los mejores gestores de bases de datos del mercado.

Por otro lado, para la creación de la base de datos respectiva al hospital 1 o EHR1, se utilizará SQL Server, el cual ofrece características similares a MySQL. Algunas de las diferencias que tienen se encuentran en la compatibilidad con plataformas, donde SQL Server está estrechamente integrado con el ecosistema de Microsoft mientras que MySQL tiene compatibilidad con una variedad más amplia de plataformas. Lo mismo ocurre con lenguajes de programación y frameworks.

SQL Server ofrece una gama más amplia de características avanzadas como el análisis de datos que lleva integrado, entre otras, mientras que MySQL ofrece un conjunto de características más básico.

### **5.2.1. MySQL WorkBench & SQL Server Management Tools**

Para facilitar el acceso a estas bases de datos y tratar de asemejar lo máximo posible el prototipo a la realidad, se manejarán ambas bases de datos desde una misma máquina. Por lo tanto, la máquina Windows en la cual está alojada el IRIS y a su vez está alojada la base de datos del hospital 1, podrá acceder directamente a la gestión de MySQL gracias a la herramienta de MySQL WorkBench. Por otro lado, la base de datos del hospital 1 en SQL Server se manejará a través de SQL Server Management Tools.

## **5.3. Generación de datos**

El prototipo necesita de datos para poder asemejarse lo máximo a la realidad. Es por eso que, previamente a la inserción de datos manualmente a través de formularios, las bases de datos contendrán sets de datos que se obtienen de páginas de generación de datos. Teniendo en cuenta que las BBDD contienen todo tipo de variables, se necesitará de la generación de datos más completa posible.

### **5.3.1. Mockaroo**

Mockaroo es una herramienta en línea que permite la generación de grandes volúmenes de datos simulados de manera rápida y sencilla. Además, incluye una gran variedad de tipos de datos, lo cual es una característica imprescindible en el prototipo [12].

Se generarán datos de prueba para la realización correcta de las pruebas del proyecto, utilizando datos de todo tipo posteriormente explicados en el desarrollo.

## 5.4. Microsoft Access

Para poder escribir sobre las BBDD de una manera más sencilla, se utilizará Microsoft Access [8]. Esta herramienta facilitará la edición y la inserción de alergias de pacientes a través de la creación de formularios que generarán consultas a la base de datos. Por eso, Microsoft Access es la herramienta perfecta debido a su sencillez de uso, su rapidez y su eficiencia.

Se procederá por tanto a la creación de un formulario de entrada de datos el cual tenga distintos campos de entrada, los necesarios para añadir un registro, este formulario funcionará únicamente con el Hospital 1.

Además se creará otro formulario para la consulta de datos, este funcionará únicamente con el Hospital 2, y permitirá comprobar las alergias de un paciente en concreto.

## 5.5. Lucidchart

Lucidchart es una herramienta de edición y diagramación que permite a los usuarios crear diagramas, esquemas y gráficos de manera cómoda y con muy buenos resultados. Para esquematizar el funcionamiento y la estructura, entre muchas otras cosas en este proyecto, se utilizará la herramienta Lucidchart ya que es de gran ayuda para el entendimiento del proyecto.

Todos los esquemas del proyecto han sido creados con esta herramienta, la cual a través de la gran cantidad de figuras y facilidades que ofrece ha permitido la creación de todos y cada uno de los esquemas.

## 5.6. IRIS de InterSystems: Edición Community

Una vez decididas las herramientas para configurar el entorno de bases de datos, lo siguiente a conocer es la herramienta de interoperabilidad, IRIS de InterSystems [9].

IRIS de InterSystems es una plataforma de datos multimodelo que combina las capacidades de una base de datos relacional, una base de datos NoSQL, una plataforma de análisis y una plataforma de desarrollo de aplicaciones.

Esta herramienta se encuentra disponible en varias versiones según la necesidad del cliente. En el caso del prototipo del proyecto, la versión más conveniente es la de Community.

Posteriormente en el desarrollo se detalla el uso de esta herramienta más profundamente, además de la

explicación de la configuración en el Anexo 1.

### **Edición Community**

La edición Community de IRIS es una versión gratuita y de código abierto. Esta aplicación está mayoritariamente dirigida a desarrolladores, estudiantes y organizaciones que buscan experimentar con las últimas tecnologías de datos sin costo alguno, por lo que es la herramienta perfecta en lo que a los objetivos de este proyecto respecta. **Ventajas de la edición Community**

- Gratuita y de código abierto
- Completa: Incluye todas las características básicas de la plataforma IRIS
- Escalable a grandes cargas de trabajo
- Ofrece una comunidad de soporte de usuarios

### **Desventajas de la edición Community**

- Soporte limitado: Pese a ofrecer un soporte respaldado por una comunidad, no ofrece ningún tipo de soporte oficial
- Uso comercial: La edición Community no puede ser usada para fines comerciales

Finalmente, se opta por la edición Community de IRIS de InterSystems ya que es una excelente opción para el desarrollo de prototipos experimentales. Es una plataforma completa, escalable y con una gran comunidad de usuarios.

## 6. Desarrollo

### 6.1. Prototipo de interoperabilidad

La explicación del prototipo debe ser lo más clara y visible posible para facilitar el entendimiento del mismo, es por eso que para explicar el prototipo en profundidad se utilizarán varios esquemas y diseños: Esquema extremo a extremo del proyecto, Esquema conceptual del proyecto, diseños de BBDD y finalmente un esquema del funcionamiento de IRIS en el proyecto.

#### 6.1.1. Esquema de extremo a extremo

Es esencial evidenciar las capas que componen cada pilar del proyecto para así dejar en claro las capas en las que este proyecto se llevará a cabo, a continuación se muestra la estructura de capas que conforman cada uno de las partes del proyecto, las cuales son: Hospital 1, IRIS y Hospital 2.

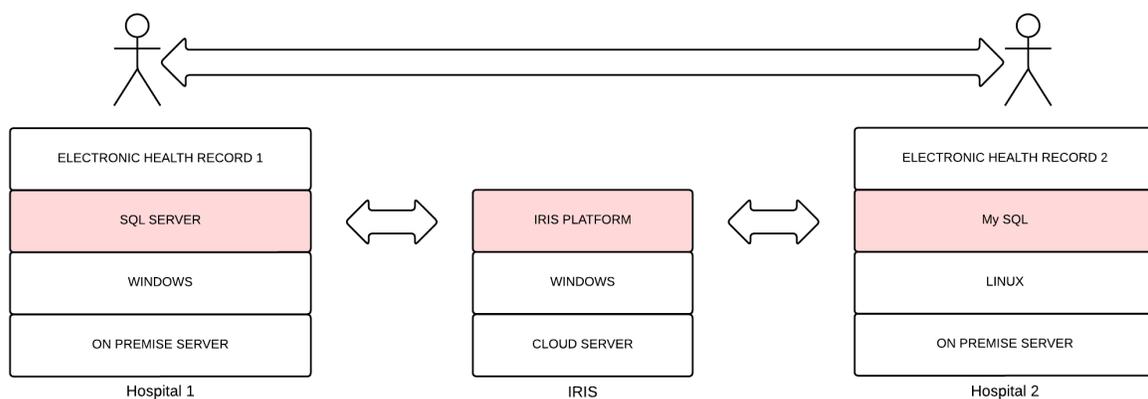


Figura 6.1: Esquema de extremo a extremo del prototipo

En el gráfico se observa primeramente dos muñecos que representan el mismo paciente, dando a entender que este puede registrar una alergia nueva en cualquiera de los dos hospitales, ese registro aparecerá posteriormente y casi de manera inmediata en el otro hospital.

Por otro lado se pueden observar las distintas capas dentro de cada pilar, primeramente y más arriba se encuentra las BBDD, en este caso EHR1 y EHR2, en el caso de IRIS y en este prototipo en específico, no se incluye ninguna base de datos, posteriormente, la capa en la que este prototipo trabaja, My SQL, IRIS Platform y SQL Server, los hospitales, utilizarán My SQL y SQL Server para gestionar sus bases de datos y será a donde IRIS se conectará para poder interactuar con las bases de datos, IRIS además será la encargada de reenviar los mensajes, leer nuevos registros, modificar registros de respuesta y guardar información en cada base de datos.

Finalmente, las dos últimas capas de cada pilar, las cuales representan el sistema operativo y el servidor o máquina en el que se encuentran. Tanto el Hospital 1 como el Hospital 2 se encuentran en un ON PREMISE SERVER el cual representa el servidor propio de cada hospital, por otro lado IRIS se encuentra en un supuesto CLOUD SERVER y será al que accedan ambos hospitales, por último tanto Hospital 1 como IRIS utilizarán el Sistema operativo de Windows y Hospital 2 utilizará Linux/MacOS.

### 6.1.2. Esquema conceptual

A continuación se muestra la estructura conceptual del prototipo del proyecto, está basada en diferentes componentes los cuales se interconectan entre sí.

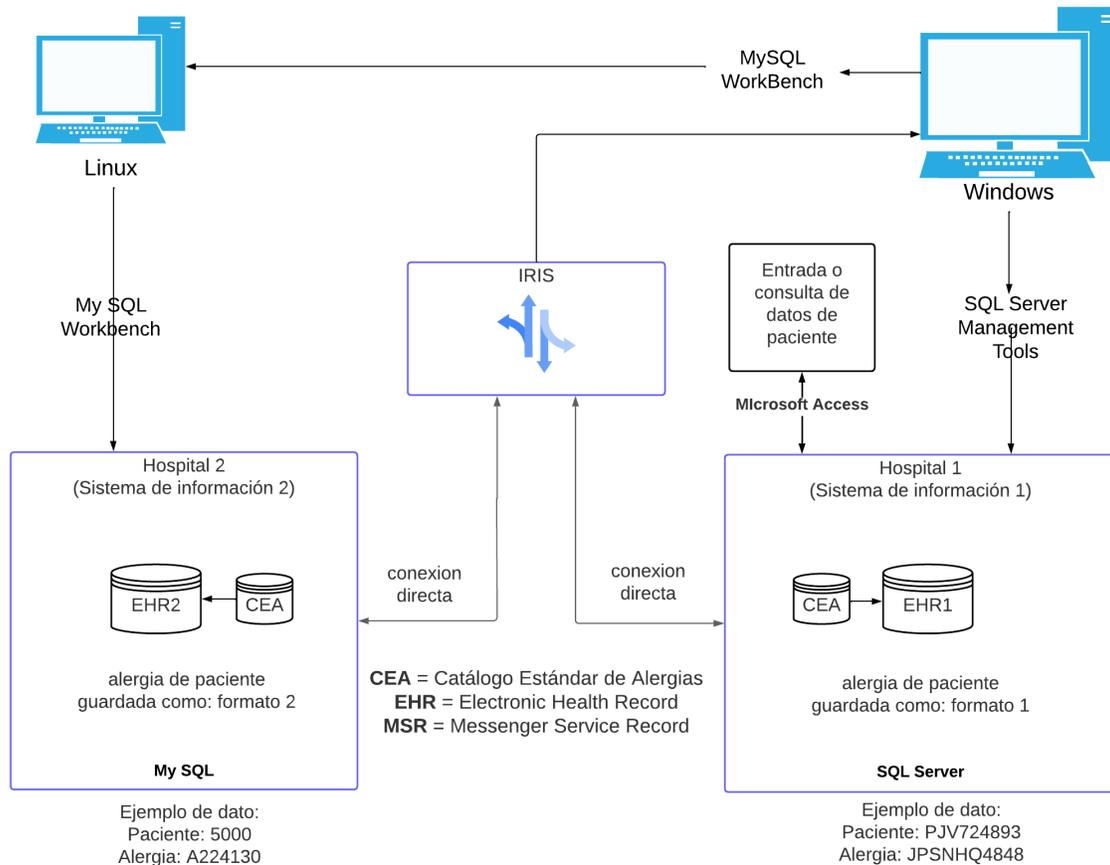


Figura 6.2: Esquema del proyecto de interoperabilidad

En primera instancia, en el esquema se observan tres divisiones (los contenedores), dos de ellas son dos sistemas de información sanitarios (los dos recuadros inferiores), a la izquierda el Hospital 2 y a la derecha el Hospital 1, ambos tratan con datos de pacientes y alergias, la diferencia entre ellos reside en la diferencia en los formatos de almacenamiento de datos, resolver esta diferencia entre sistemas será el problema a resolver y lograr que la comunicación sea fluida y eficaz será la finalidad del proyecto.

Además en el esquema se puede observar como están representadas las dos máquinas que simularán los

hospitales (Linux y Windows), el servicio IRIS, los formularios de entrada y consulta de datos, ejemplos de datos y finalmente las relaciones entre los diferentes componentes.

Para facilitar el funcionamiento y el manejo del proyecto, ambos hospitales, incluso IRIS se manejarán desde la máquina o servidor windows el cual tiene una conexión directa con MySQL WorkBench a Linux, hospeda al servicio de IRIS y además gestiona su propia base de datos con el administrador de SQL Server Management Tools.

## 6.2. Hospital 1

En el esquema se encuentra toda la información referente al Hospital 1, todo lo que lo compone, desde donde está alojado, hasta información sobre su base de datos.

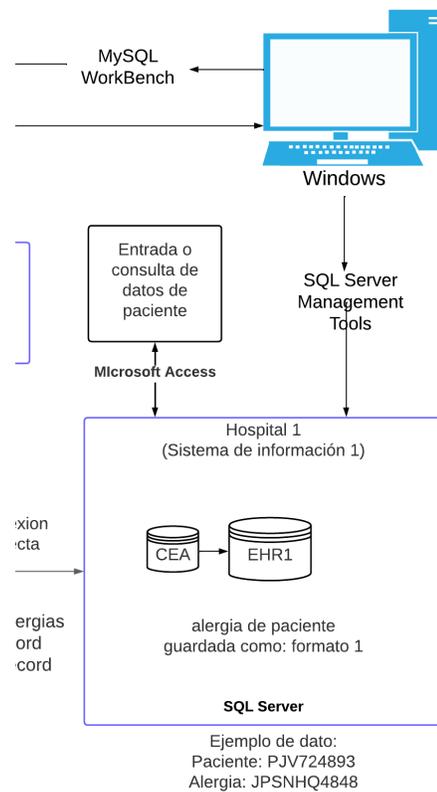


Figura 6.3: Esquema de Hospital 1

Este esquema representa el sistema de información del Hospital 1, el cual como se ha podido comprobar anteriormente es el encargado de gestionar todo el proyecto.

Este hospital se encuentra hospedado en la máquina de Windows, dentro de esta se encuentra su base de datos propia EHR1 (Electronic Health Record 1) y CEA (Catalogo Estándar de Alergias) el cual se implementa en EHR1 para facilitar la uniformidad de los datos.

Esta base de datos será manipulada a través de una herramienta administradora de BBDD, en este caso, SQL Server Management Tools.

A diferencia del anterior hospital y para simular una diferenciación mayor entre hospitales, este hospital cuenta con un formulario de entrada de datos realizado con la herramienta de Microsoft Access, el cual permite registrar una alergia para un paciente en una fecha en concreto.

Además este sistema de información guardará los datos en un formato específico, este formato está descrito en el diseño de base de datos del Hospital 1 (EHR1).

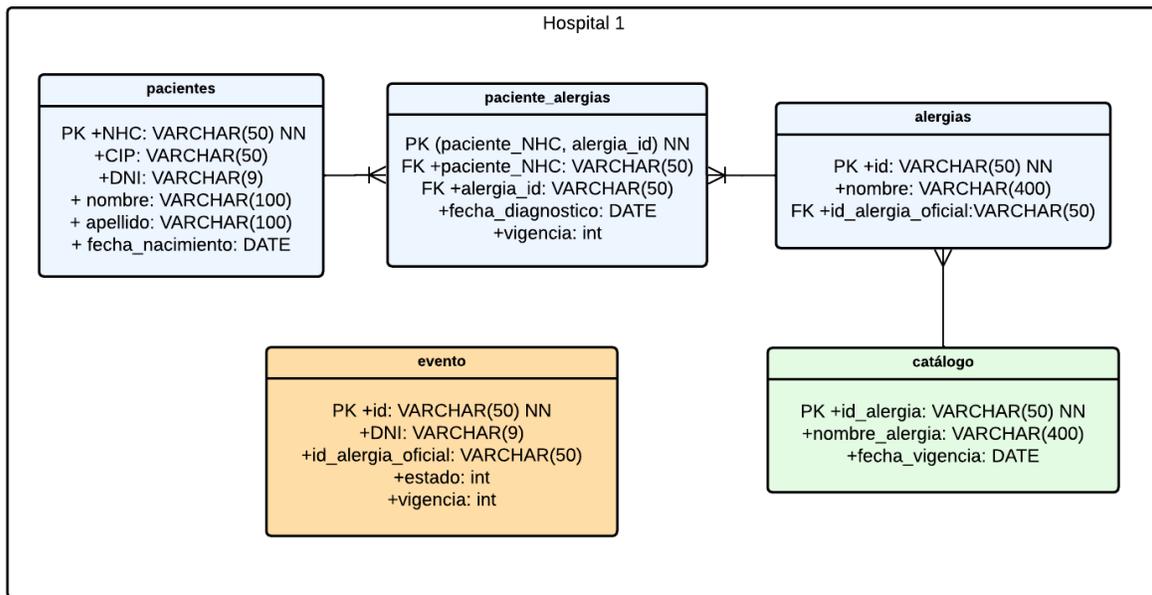


Figura 6.4: Diseño de base de datos de Hospital 1

En este diseño de base de datos se puede observar de color celeste las tablas pertenecientes a la base de datos EHR1 (pacientes, paciente\_alergias, alergias), de un color verdoso, la tabla perteneciente a la base de datos del estándar de alergias (catálogo) y finalmente la tabla de evento que se utilizará para la comunicación con IRIS y que se explicará posteriormente.

La primera tabla es el registro de pacientes (pacientes) la cual contiene toda la información relevante de cada paciente, contiene distintos campos de información entre los cuales se encuentran:

- Número de Historia Clínica (NHC): Se trata de la Primary Key de la tabla y es alfanumérica
- Código de Identificación del Paciente (CIP): Se trata de una Unique Key y es alfanumérica
- Documento Nacional de Identidad (DNI): Se trata de una Unique Key y es alfanumérica
- Nombre de paciente: Una variable alfanumérica
- Apellido: Una variable alfanumérica
- Fecha de nacimiento: Una variable de tipo Fecha

Por otro lado, la tabla de alergias, la cual contiene toda la información referente a las alergias de ese hospital.

- ID de la alergia: Se trata de una variable alfanumérica
- Nombre de la alergia: Se trata de una variable alfanumérica
- ID de la alergia oficial: Se trata de una Foreign Key que viene de la tabla Catálogo y es una variable alfanumérica

Entre las tablas anteriormente mencionadas, (pacientes y alergias) se encuentra paciente\_alergias, esta tabla es la más importante en el proyecto, representa las alergias registradas de cada paciente por lo que contendrá toda la información con la que se transaccionará y sobre la que la tabla de evento interactuara. Esta tabla es la que almacena el diagnóstico realizado de la detección de una alergia a un paciente y contiene:

- El NHC de paciente a través del cual IRIS accederá al DNI del paciente, ya que es el identificador más útil en este caso
- ID de la Alergia a través del cual IRIS accederá al ID de Alergia Oficial
- Fecha diagnóstico: Se trata de una variable de tipo Fecha se trata de la fecha en la que se hizo el diagnóstico
- vigencia: Es un valor de tipo integer el cual indica la validez del registro
- La primary key que será autogenerada mediante paciente\_NHC y alergia\_id

Además de esta información la tabla paciente\_alergias contiene un trigger para que cada vez que se añada un valor, este valor sea automáticamente añadido a la tabla evento, únicamente en el caso de que se añada un valor a esta tabla que previamente haya sido invocado por otro trigger (el caso de que venga desde evento) no se procesará por el trigger.

La tabla Catálogo contiene la información estandarizada de las alergias, esta contiene los siguientes campos:

- ID oficial de la alergia: Se trata de la Primary Key de la tabla y es la misma que se utiliza en Alergias como Foreign Key, se trata de una variable alfanumérica
- Nombre de la alergia: Una variable alfanumérica
- Fecha de vigencia: Una variable de tipo Fecha

Finalmente, la tabla evento la cual moverá los registros de alergias de pacientes de hospital en hospital, es la tabla de la cual IRIS leerá información y se la pasará al otro hospital, esta contiene los siguientes campos:

- id de evento: Una variable alfanumérica que guardará el identificador único de cada evento
- DNI: Documento Nacional de Identidad de cada paciente, el cual se recupera a través del NHC
- id oficial de alergia: id que servirá en el otro hospital, ya que es el oficial y único para ambos hospitales, se recupera a través del id de alergia
- estado: Variable int la cual puede contener 5 valores diferentes
  - 0: El valor todavía no ha sido leído por IRIS
  - 1: El valor ha sido leído por IRIS
  - 2: El valor ha sido procesado en el otro hospital con éxito
  - 3: Ha habido un error en la transacción
  - 4: El valor debe ser añadido en el hospital actual
- vigencia: Es un valor de tipo integer el cual indica la validez del registro

Además de esta información, la tabla evento contiene un Trigger que funciona para añadir valores en el mismo hospital en caso de que el estado sea 4. Por otro lado, IRIS cada 5 segundos leerá la tabla y en el caso de que el estado de algún registro sea 0 IRIS procesará este registro y posteriormente lo marcará con el estado 1.

### 6.3. Hospital 2

En el esquema se encuentra toda la información referente al Hospital 2, todo lo que lo compone, desde donde está alojado, hasta información sobre su base de datos.

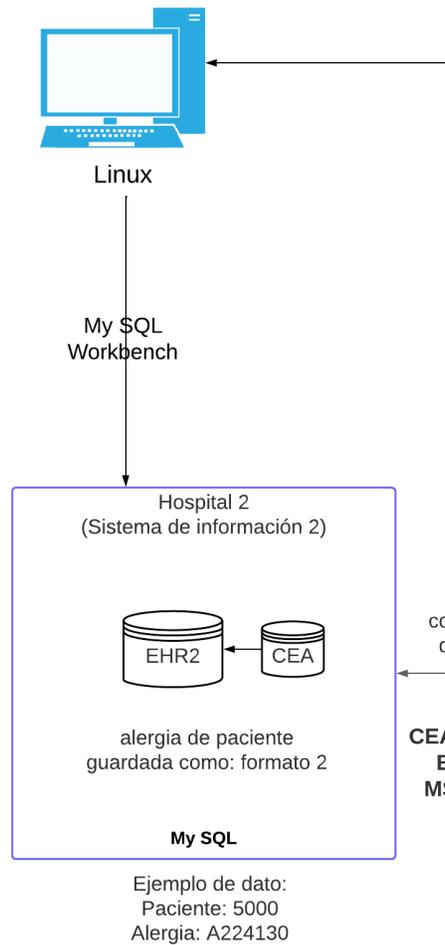


Figura 6.5: Esquema de Hospital 2

Este hospital se encuentra hospedado en una máquina o servidor Linux, dentro de este se encuentra su base de datos propia EHR2 (Electronic Health Record 2) y CEA (Catalogo Estándar de Alergias) el cual se implementa en EHR2 para facilitar la uniformidad de los datos.

Esta base de datos será manipulada a través de una herramienta administradora de BBDD, en este caso, MySQL Workbench, para la consulta de alergias de pacientes este hospital contará también con un formulario.

Además este sistema de información guardará los datos en un formato específico, este formato está descrito en el diseño de base de datos del Hospital 2 (EHR2).

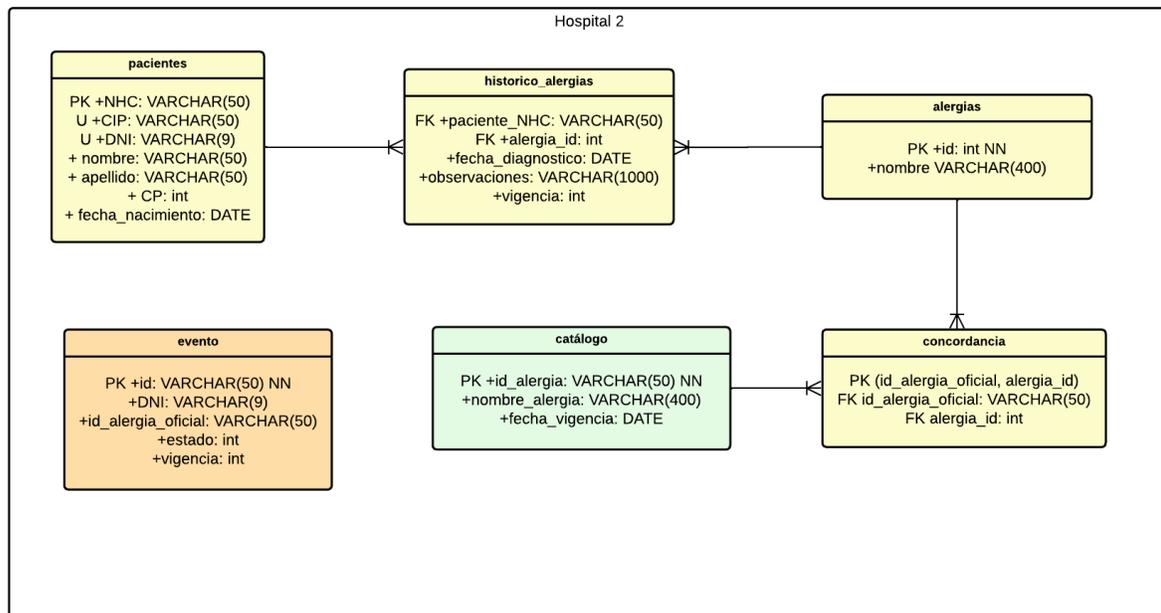


Figura 6.6: Diseño de base de datos de Hospital 2

En este sistema de información a diferencia del anterior se puede observar una arquitectura distinta, contiene 4 tablas de color amarillo pertenecientes a la base de datos de EHR2 (pacientes, historico\_alergias, alergias y concordancia), una de color verde (catálogo) y finalmente la tabla de evento que se utilizará para la comunicación con IRIS. Esta vez la concordancia entre EHR2 y CEA no es tan directa como anteriormente, en este caso se utiliza una tabla intermedia la cual hace de puente y permite no tener que tocar nada de la base de datos original del Hospital 2 (EHR2).

La primera tabla es el registro de pacientes la cual contiene la información relevante del paciente, esta contiene distintos campos de información entre los cuales se encuentran:

- Número de Historia Clínica (NHC): Se trata de la Primary Key de la tabla y es alfanumérica
- Código de Identificación del Paciente (CIP): Se trata de una Unique Key y es alfanumérica
- Documento Nacional de Identidad (DNI): Se trata de una Unique Key es alfanumérica
- Nombre de paciente: Se trata de una variable alfanumérica
- Apellido: Se trata de una variable alfanumérica
- Código postal (CP): Se trata de una variable numérica
- Fecha de nacimiento: Se trata de una variable de tipo Fecha

Por otro lado, la tabla de alergias, la cual contiene toda la información referente a las alergias de ese hospital.

- ID de la alergia: Se trata de una variable numérica
- Nombre de la alergia: Se trata de una variable alfanumérica

Entre las tablas anteriormente mencionadas, pacientes y alergias, se encuentra historico\_alergias, esta tabla es la más importante en el proyecto, representa las alergias del paciente por lo que contendrá toda la información con la que se transacciona. Esta tabla es la que simboliza el diagnóstico de la alergia y contiene:

- El NHC de paciente a través del cual IRIS accederá al DNI del paciente, ya que es el identificador más útil en este caso
- ID de la Alergia a través del cual IRIS accederá al ID de Alergia Oficial
- Fecha diagnóstico: Se trata de una variable de tipo Fecha concretamente es la fecha en la que se hizo el diagnóstico
- vigencia: Es un valor de tipo integer el cual indica la validez del registro

Además de esta información la tabla historico\_alergias contiene un trigger para que cada vez que se añada un valor, este valor sea automáticamente añadido a la tabla evento, únicamente en el caso de que se añada un valor a esta tabla que previamente haya sido invocado por otro trigger (el caso de que venga desde evento) no se procesará por el trigger.

Entre el catálogo y la alergia se encuentra la tabla concordancia, la cual se utiliza para poder relacionar estas dos sin tener que disponer de una foreign key en la tabla alergia. Esta tabla contiene lo siguiente:

- El ID de Alergia oficial proveniente del catálogo
- El ID de Alergia proveniente del hospital
- La primary key que se generará automáticamente teniendo en cuenta ambas foreign keys

La tabla catálogo contiene la información estandarizada de las alergias, esta contiene los siguientes campos:

- ID oficial de la alergia: Se trata de la Primary Key de la tabla y es la que misma que se utiliza en Alergia como Foreign Key, se trata de una variable alfanumérica
- Nombre de la alergia: Se trata de una variable alfanumérica
- Fecha de vigencia: Se trata de una variable de tipo Fecha

Finalmente, la tabla evento la cual moverá los registros de hospital en hospital, es la tabla de la cual IRIS leerá información, funcionará de la misma forma que la anterior tabla evento previamente explicada, contiene los siguientes campos:

- id de evento: Una variable alfanumérica que guardará el identificador único de cada evento
- DNI: Documento Nacional de Identidad, se recuperará a través del NHC
- id oficial de alergia: id que servirá en el otro hospital, ya que es el oficial y único para ambos hospitales, se recuperará a través del id de alergia
- estado: Variable int la cual puede contener 5 valores diferentes
  - 0: El valor todavía no ha sido leído por IRIS
  - 1: El valor ha sido leído por IRIS
  - 2: El valor ha sido procesado en el otro hospital con éxito
  - 3: Ha habido un error en la transacción
  - 4: El valor debe ser añadido en el hospital actual
- vigencia: Es un valor de tipo integer el cual indica la validez del registro

Además de esta información, la tabla evento contiene un Trigger que funciona para añadir valores en el mismo hospital en caso de que el estado sea 4. Por otro lado, IRIS cada 5 segundos leerá la tabla y en el caso de que el estado de algún registro sea 0 IRIS procesará este registro y posteriormente lo marcará con el estado 1.

## 6.4. IRIS

En el esquema se encuentra toda la información externa que afecta a IRIS, donde está alojado y la conexión directa hacia ambas BBDD.

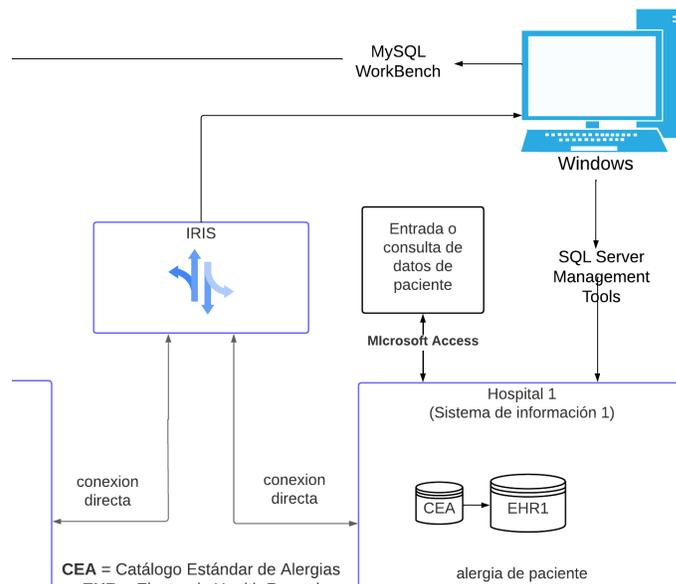


Figura 6.7: Esquema de IRIS

En el esquema se observa a IRIS alojada en la máquina Windows que será donde será gestionada pese a que en este proyecto está alojada en una máquina local, la solución más óptima si se está trabajando con varios hospitales en ubicaciones distintas es en una nube externa a estos hospitales.

A continuación se procederá a la explicación del funcionamiento de IRIS en el proyecto, utilizando esquemas realizados con LucidChart, se han realizado dos esquemas, un primer esquema que representa el funcionamiento de un hospital a otro, este funcionamiento no es del todo real, ya que en este proyecto IRIS está funcionando en ambas direcciones por lo que en el segundo esquema se aprecia el bidireccionalidad.

Para un mejor entendimiento primeramente se procederá a explicar el primer esquema de una única dirección de funcionamiento.

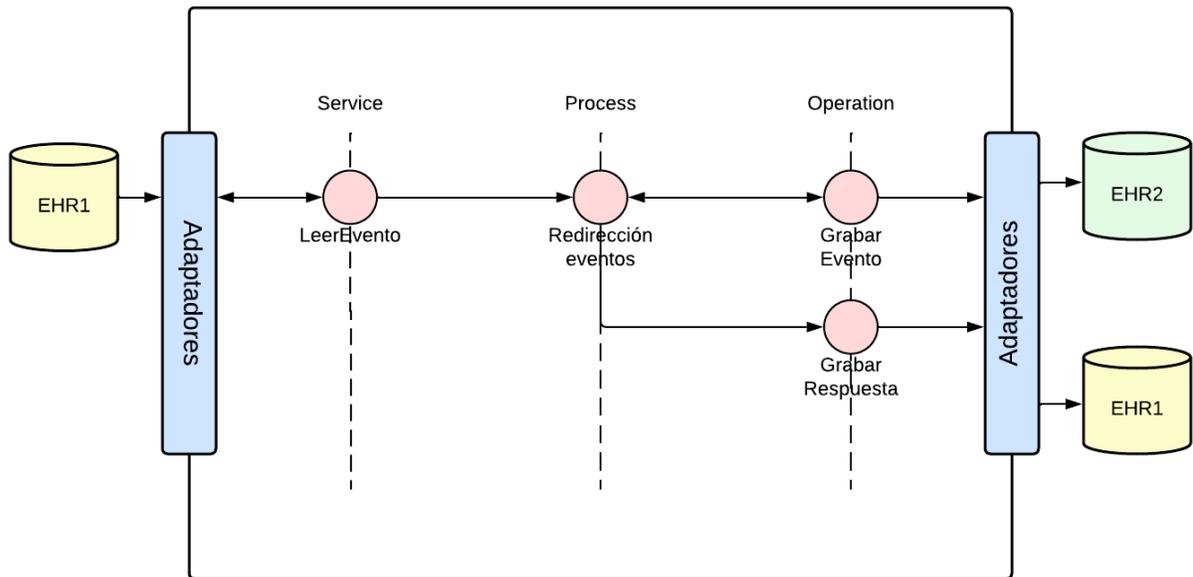


Figura 6.8: Esquema de funcionamiento de IRIS

En este esquema se pueden distinguir varias secciones, primeramente las BBDD las cuales están representadas al inicio y al final del proceso, los adaptadores para la conexión de IRIS a las bases de datos y dentro de IRIS se pueden distinguir tres partes: Service Process y Operation.

Primeramente IRIS mediante el **business service** LeerEvento, lee cada cinco segundos de la base de datos EHR1, a través de un adaptador, este lee directamente de la tabla de evento previamente mencionada y una vez leído reescribe el estado del registro ha leído, posteriormente, el registro se reenvía al **business process** el cual se encarga de redirigir el mensaje, en este caso lo redirige al **business operation** el cual recibe el registro y a través de los adaptadores registra el nuevo evento en el otro hospital, en este caso EHR2.

Una vez registrado el evento queda grabar la respuesta en la base de datos de origen EHR1, por lo que el business operation de grabar evento devolverá un mensaje de respuesta el cual recibirá el business process de redirección de eventos y reenviará al business operation el cual se encarga de grabar respuestas en la base de datos de origen EHR1.

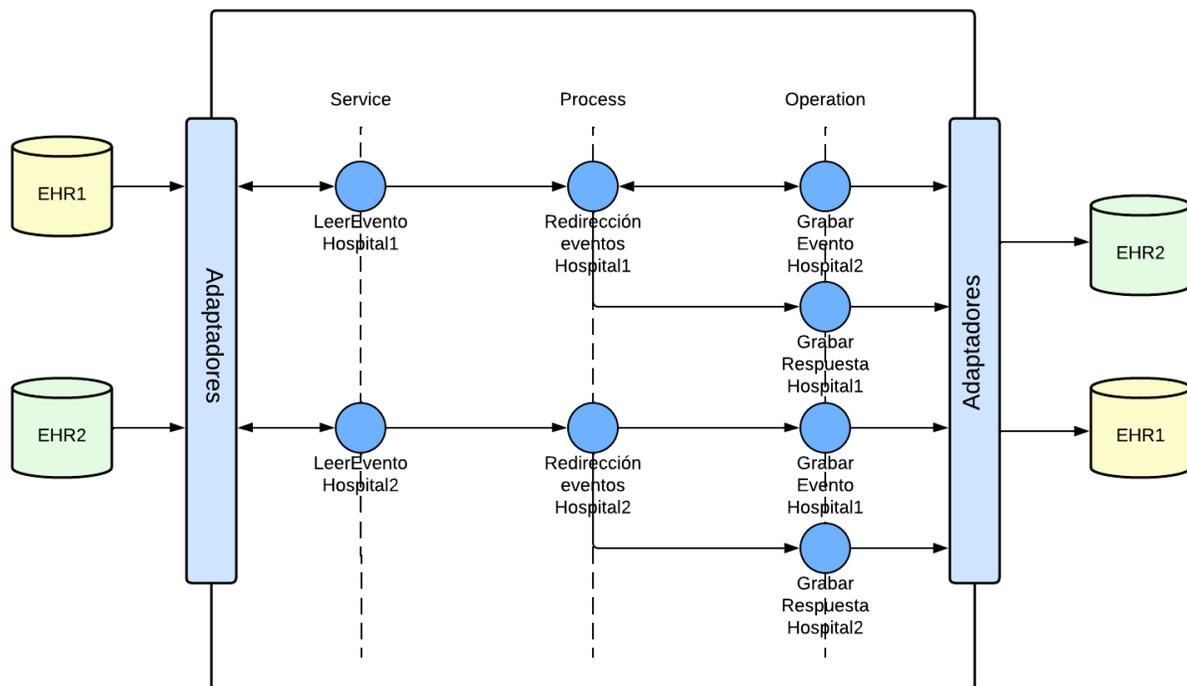


Figura 6.9: Esquema de funcionamiento completo de IRIS

En este otro esquema se puede observar la realidad del sistema, este funciona en ambas direcciones, tanto de EHR1 a EHR2 como de EHR2 a EHR1, posteriormente se explica el funcionamiento en detalle.

A diferencia del anterior esquema, este cuenta con dos business service lectores de eventos que efectúan la misma función de lectura pero en diferentes BBDD.

Por otro lado, se añaden dos business process que al igual que el anterior esquema cumplen la misma función pero con diferentes destinos.

En cuanto a los business operation, ambos graban eventos y respuestas en ambas BBDD, por lo que tienen acceso a las dos.

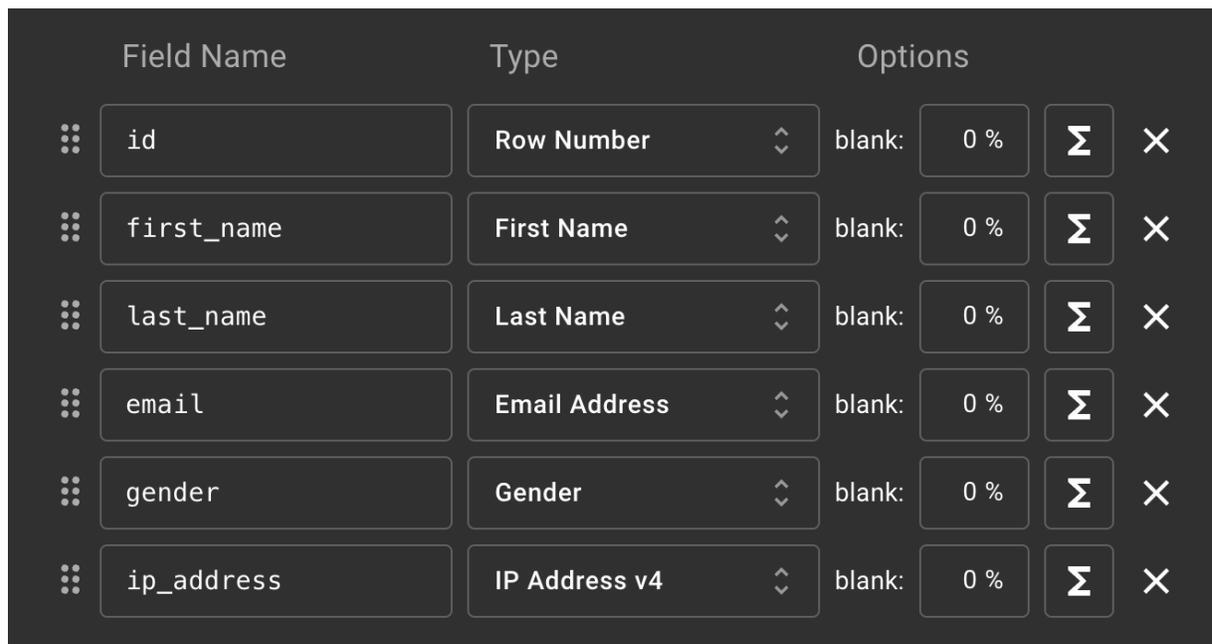
Finalmente se puede observar como en este caso desde el inicio ya están presentes ambas BBDD y en el momento de grabar también, la diferencia es que ambos caminos llevan a ambas BBDD de destino pero no de origen.

## 6.5. Entrada de datos

En cuanto a los datos de las BBDD, para poder hacer inserciones y modificaciones posteriores de alergias de pacientes se necesita una base sobre la cual poder interactuar, esto se obtiene con la generación de datos, se generarán gracias a la página web de Mockaroo. Una vez los datos están generados e insertados en el modelo, se procede a la inserción a través de formulario o inserciones directas con consulta a la base de datos.

### 6.5.1. Mockaroo fake data generator

Esta herramienta ofrece la posibilidad de generar múltiples filas de datos con diferentes columnas y tipos.



Field Name	Type	Options
id	Row Number	blank: 0 % $\Sigma$ X
first_name	First Name	blank: 0 % $\Sigma$ X
last_name	Last Name	blank: 0 % $\Sigma$ X
email	Email Address	blank: 0 % $\Sigma$ X
gender	Gender	blank: 0 % $\Sigma$ X
ip_address	IP Address v4	blank: 0 % $\Sigma$ X

Figura 6.10: Campos de datos con su tipo

En la imagen se observa como dentro de la web, mockaroo ofrece varios campos con distintos tipos, dentro de los tipos de columnas hay varias opciones.

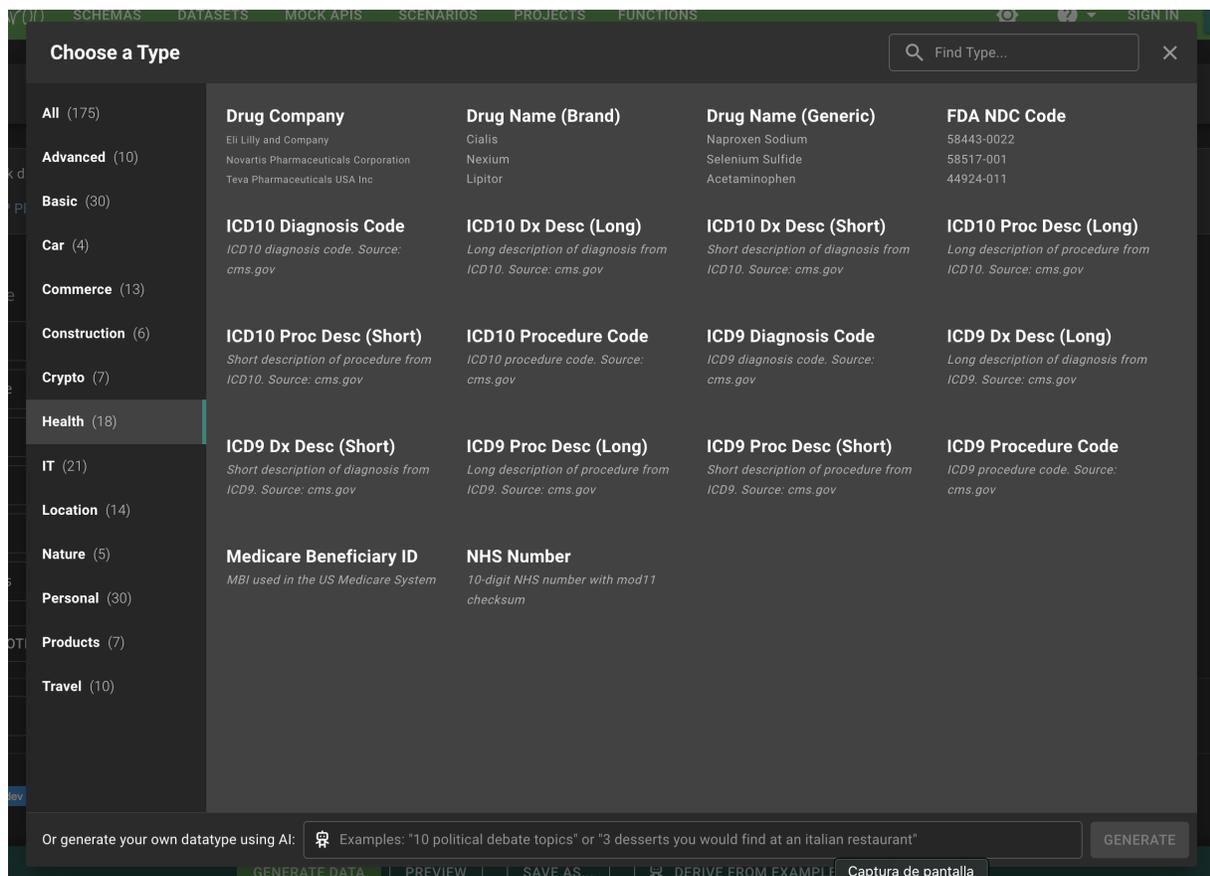


Figura 6.11: Tipos de campos

Hay una gran cantidad de tipos de campos que ofrece Mockaroo, en este caso los que más se utilizarán serán:

- Alfanumérico aleatorio para generar campos como: ID, DNI, NHC, CP, CIP entre otros
- Nombres y apellidos
- Fechas aleatorias entre límites establecidos
- Nombres de medicamentos: estos simularán los nombres de alergias
- Numéricos aleatorios para decidir el estado de vigencia

### 6.5.2. Formulario Microsoft Access

Con todos los datos necesarios introducidos el servicio de entrada de datos vía formulario se puede utilizar sin encontrar ningún error.

Como se ha comentado previamente este formulario está conectado dentro del sistema de información del Hospital 1, incide directamente en la base de datos EHR1.

dbo paciente alergias

paciente\_NHC 7557062671

alergia\_id 4VT2KQ4ND34 +!

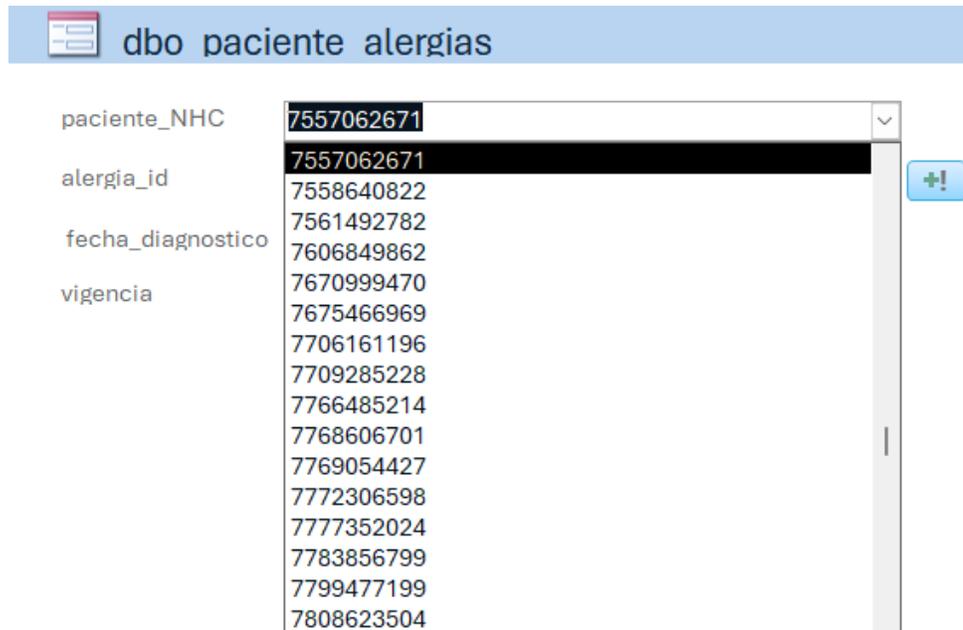
fecha\_diagnostico 18/05/2024

vigencia

Figura 6.12: Formulario alergia paciente de Hospital 1

Una vez iniciado el formulario se observan cuatro opciones para la inserción, el objetivo es poder añadir una nueva alergia de un paciente por lo que el formulario es idóneo, además ofrece la opción de añadir una fecha de cuando se realizó esta prueba y la vigencia de la misma, marcando o desmarcando en función de si la alergia que se quiere añadir será una nueva alergia de paciente o por lo contrario sea una inserción para deshabilitar una alergia ya existente de un paciente.

En las siguientes imágenes se puede observar las distintas opciones que ofrece este formulario, no solo es posible escribir el paciente o la alergia manual, sino que cuenta con un selector de paciente y alergia.



The screenshot shows a web form titled "dbo paciente alergias". It contains four fields: "paciente\_NHC", "alergia\_id", "fecha\_diagnostico", and "vigencia". The "paciente\_NHC" field is a dropdown menu with the value "7557062671". The "alergia\_id" field is a dropdown menu with a list of patient IDs. The "fecha\_diagnostico" and "vigencia" fields are empty. A blue button with a plus sign and an exclamation mark is located to the right of the "alergia\_id" dropdown.

Field	Value
paciente_NHC	7557062671
alergia_id	7557062671
fecha_diagnostico	
vigencia	

- 7558640822
- 7561492782
- 7606849862
- 7670999470
- 7675466969
- 7706161196
- 7709285228
- 7766485214
- 7768606701
- 7769054427
- 7772306598
- 7777352024
- 7783856799
- 7799477199
- 7808623504

Figura 6.13: Formulario alergia paciente de Hospital 1: paciente\_NHC



The screenshot shows a web form titled "dbo paciente alergias". It contains four fields: "paciente\_NHC", "alergia\_id", "fecha\_diagnostico", and "vigencia". The "paciente\_NHC" field is a dropdown menu with the value "7557062671". The "alergia\_id" field is a dropdown menu with a list of allergy IDs. The "fecha\_diagnostico" and "vigencia" fields are empty. A blue button with a plus sign and an exclamation mark is located to the right of the "alergia\_id" dropdown.

Field	Value
paciente_NHC	7557062671
alergia_id	4VT2KQ4ND34
fecha_diagnostico	4VT2KQ4ND34
vigencia	

- 4X19N21MX06
- 4X63ND9AH13
- 4X63YC8QW97
- 4XD6EV9HK91
- 5AC9UV4WJ67
- 5AD3D88GV57
- 5AT1KD7JF27
- 5AY9MM4RW04
- 5C46KW2GY68
- 5CC7KD7CC69
- 5CF6CA0VW69
- 5CU8CW0GK49
- 5DF0N77PF57
- 5DH0NX4RC79
- 5DQ8TA9JT63

Figura 6.14: Formulario alergia paciente de Hospital 1: alergia\_id

En cuanto a la fecha existen varias opciones, de igual manera se puede escribir manual y por otro lado también es posible seleccionar una fecha o incluso seleccionar el día de hoy.

Figura 6.15: Formulario alergia paciente de Hospital 1: fecha.diagnostico

### 6.5.3. Entrada directa a base de datos

Finalmente otra opción para la entrada de registros a las BBDD, es la opción de la entrada directa de datos a través de una consulta de inserción, esta opción es válida para ambas BBDD y se puede realizar desde los administradores de las mismas, MySQL WorkBench para EHR2 y SQL Server Management Tools para EHR1.

```

1 • INSERT INTO historico_alergias (paciente_NHC, alergia_id, fecha_diagnostico, observaciones, vigencia)
2 VALUES ("4049946745", 136420, '2022-05-04', "N/A", 1)
    
```

Figura 6.16: Consulta insert en Hospital 1

Para EHR2 se puede observar como la inserción de una nueva alergia se realiza directamente en la tabla historico\_alergias y se añaden manualmente los valores necesarios.

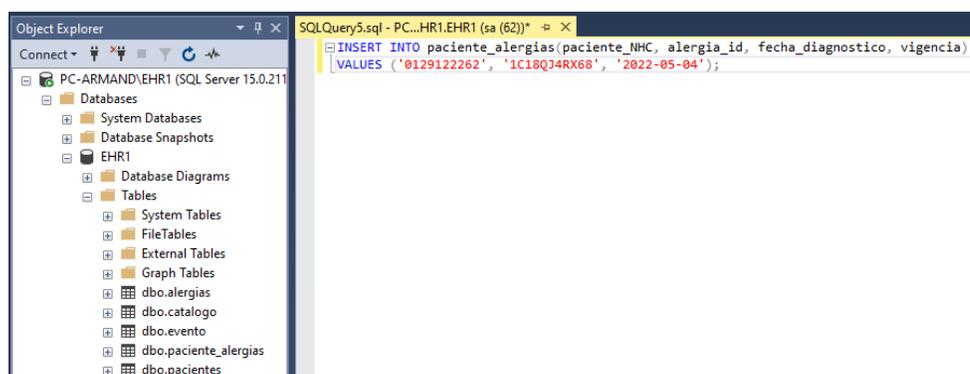


Figura 6.17: Consulta insert en Hospital 1

Por otro lado en EHR1 los valores se añaden a la tabla paciente\_alergias y de igual manera se añaden manualmente, se puede observar como en este caso la tabla tiene una estructura diferente.

### 6.6. Caso de uso

Para poder explicar el flujo del proyecto se recurre a la creación de un caso de uso el cual describe el flujo que sigue la introducción de una nueva alergia de un paciente en un hospital, desde el Hospital 1 al Hospital 2.

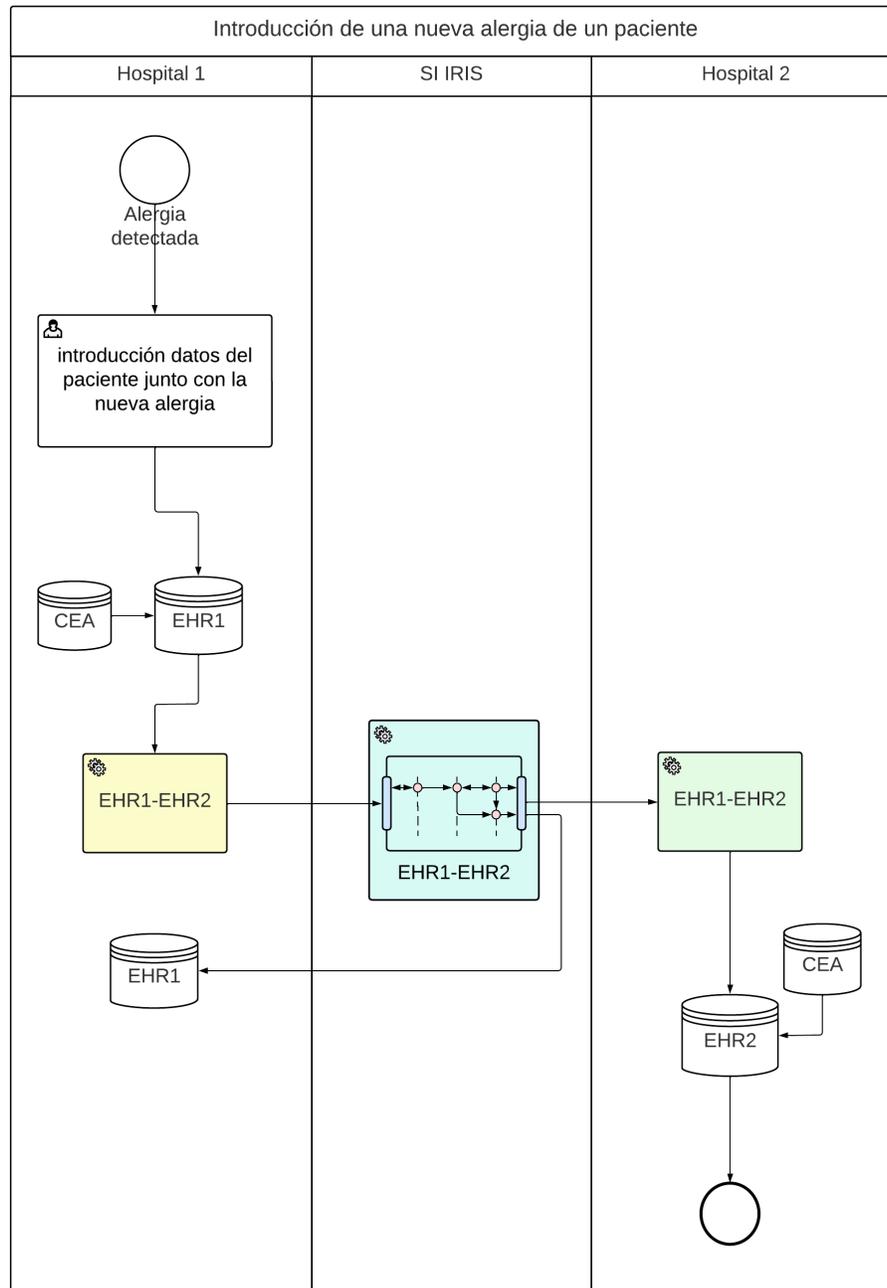


Figura 6.18: Proceso de introducción de una nueva alergia de un paciente

Primeramente en el esquema se distinguen tres claros carriles, los cuales representan: El Hospital 1 a la izquierda, el sistema de interoperabilidad (SI) IRIS y finalmente el Hospital 2 a la derecha.

Cada carril contiene lo que ocurre en ese mismo lugar ordenado temporalmente de arriba hacia abajo, en este proceso se pueden encontrar varios tipos de componentes como: Evento inicial y final, las bases de datos y las automatizaciones que simbolizan la interoperabilidad.

### **Primer carril**

Primeramente analizando la parte superior, se encuentra en el primer carril el evento que corresponde a la nueva alergia detectada en el paciente la cual se debe registrar. Este suceso lleva al siguiente paso, la introducción de los datos del paciente junto con la nueva alergia, esto será introducido a través de un formulario, mostrado previamente, en caso de que esta inserción hubiese sido realizada desde el otro hospital, la inserción habría sido a través de una consulta de inserción.

Una vez introducidos los datos, estos son guardados dentro de la base de datos propia del hospital de manera que quedan visibles para ese hospital pero no para el Hospital 2.

IRIS que está constantemente leyendo valores introducidos en la base de datos lee este nuevo dato y se envía al segundo carril, el servicio de IRIS.

Finalmente en este carril se observa como se posteriormente se guarda el valor de respuesta en el Hospital 1, tanto el de lectura como el de posterior respuesta de introducción errónea o correcta

### **Segundo carril**

En este segundo carril, el sistema de información de IRIS, recibe la información perteneciente del hospital emisor, esta información la procesa a través de la aplicación de IRIS Community y la reenvía al hospital receptor.

En el recurador se ve representado el esquema previamente explicado, el cual recibe un valor de el Hospital 1 y lo reenvía al Hospital 2, además de el guardado de la respuesta.

### **Tercer carril**

En este último sistema de información del segundo hospital, se recibe la información referente al paciente con la alergia nueva a introducir y se hace una traducción al formato de este mismo segundo hospital para

poder almacenar correctamente la información. Esta información llega como una tabla evento la cual contiene la información y se utiliza para rellenar las tablas necesarias.

En este punto ambos hospitales contienen la misma información.

### 6.7. Flujo de funcionamiento

Para una comprensión total del flujo de funcionamiento se han realizado tres esquemas de cada parte del proceso, en el Hospital 1, en IRIS y finalmente en el Hospital 2. Con la introducción de una alergia en el Hospital 1 se da el siguiente funcionamiento:

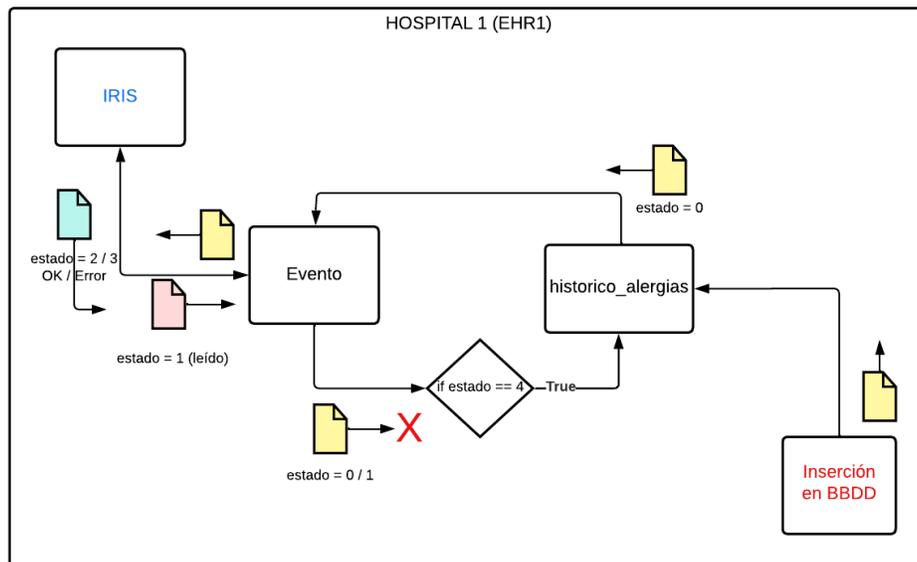


Figura 6.19: Flujo de funcionamiento en origen

Primeramente se inserta una registro en la base de datos EHR1, a la izquierda del esquema, la nota de color amarillo representa el registro.

El registro es introducido en paciente\_alergias como un nuevo registro, el trigger de paciente\_alergias es lanzado con esta inserción. Este trigger realiza una consulta con los datos del registro previo para crear un nuevo registro en la tabla evento, para realizar esto recoge los datos de otras tablas externas y finalmente pone valor 0 a la variable estado.

Una vez el registro se encuentra en la tabla evento, se lanza el trigger instalado en evento, este necesita que el nuevo valor introducido sea valor 4 por lo que no realizará nada. Por otro lado IRIS constantemente le valores de evento y en caso de que el valor del nuevo evento sea 0 este es recogido, por lo que al leer el nuevo valor IRIS procede a recoger el valor.

Una vez el valor es recogido por IRIS, hay varias posibilidades, en color rojo, un registro dice que el valor ha sido leído por IRIS por lo que el registro previo dentro de evento pasa a tener estado 1 y ya no puede ser leído por IRIS. En el momento final del proceso, IRIS reenvía el resultado, la nota de color azul, este cambiará el valor estado del registro previo, en caso de que la inserción en el otro hospital haya sido satisfactoria el valor será 2, 3 en caso de que haya habido un error.

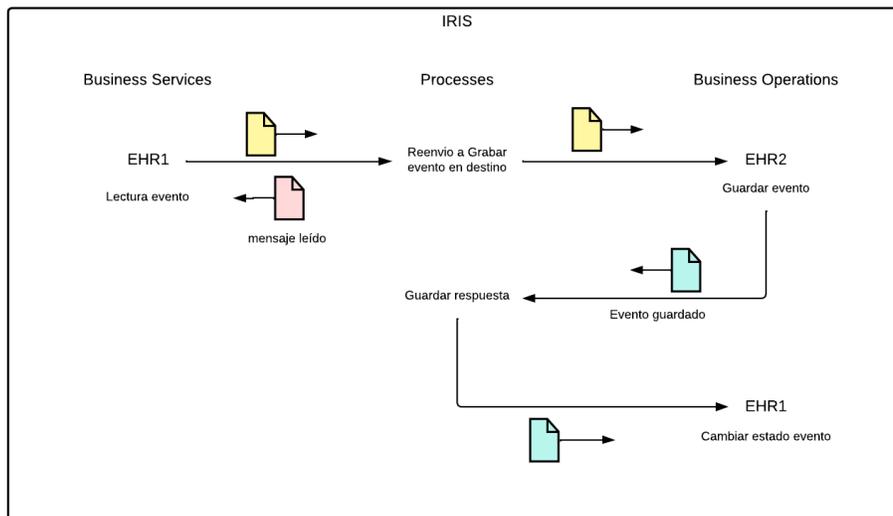


Figura 6.20: Flujo de funcionamiento en IRIS

En este esquema se pueden ver las tres fases previamente explicadas de IRIS, primeramente el business service lee el registro del Hospital 1 y la reenvía al proceso, la nota de color amarillo, una vez leído el mensaje este se envía a la base de datos de origen en modo de respuesta, la nota de color rojo.

Una vez el proceso lo recibe, este lo reenvía al business operation encargado de guardar eventos, este guarda el evento en el Hospital 2 y recibe la respuesta de la base de datos, el registro contendrá toda la información necesaria, además incluirá el valor de estado a 4 para que así se añada en la misma base de datos.



### 6.7.1. IRIS Data Platform

IRIS es una plataforma la cual es de mucha utilidad en este proyecto, resuelve la problemática de la interoperabilidad técnica, en este apartado se presentan distintas configuraciones que se consideran esenciales, para más información sobre la configuración se debe consultar el anexo 1: Configuración IRIS Data Platform, este muestra en profundidad la configuración de IRIS en el proyecto.

Una de las pantallas principales que muestran toda la configuración de la interoperabilidad, es la configuración de la producción la cual muestra todos los servicios, procesos y operaciones del proyecto.

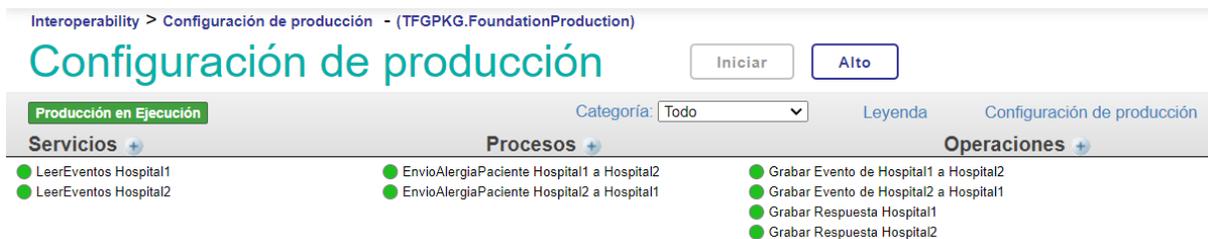


Figura 6.22: Configuración de producción de IRIS

En la imagen se puede observar todos los componentes previamente explicados en el esquema de IRIS, además se puede observar como todos los componentes permanecen encendidos y activos siendo la luz verde en cada uno indicativo de esto.

Ambos servicios de lectura de eventos tienen configurados un data source name adecuado para su base de datos para poder acceder de manera correcta, además de ciertas configuraciones para el correcto funcionamiento, dentro del anexo se entra en profundidad.

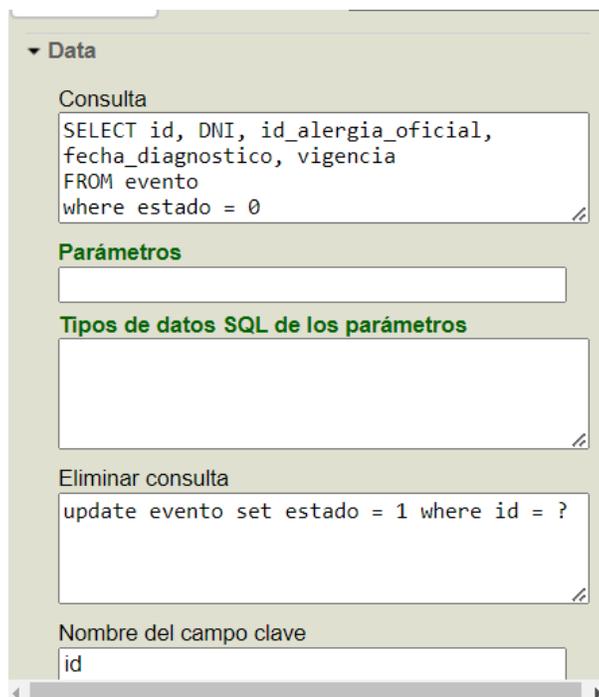


Figura 6.23: Configuración de servicio de lectura de evento

En la misma configuración de ambos servicios se encuentra la consulta que se realiza hacia la base de datos cada cinco segundos, se observa como recoge todos los datos de la tabla de evento y además en el apartado de eliminar consulta realiza una actualización que marca el estado uno que simboliza que se ha realizado la lectura del registro.

Además, IRIS permite la programación de ciertos servicios y funciones, en el caso de ambos servicios se ha realizado una función la cual se ejecuta en cuanto se procesa un valor, esta función recoge todos los valores de la consulta y los envía al proceso.

El proceso tiene una regla de negocio configurada, la cual realiza un reenvío de los datos siempre hacía la operación de grabado de registro, a excepción del mensaje de respuesta.

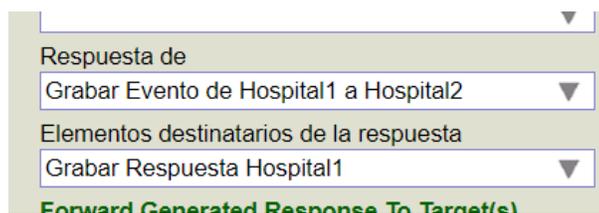


Figura 6.24: Configuración de respuesta en el proceso

El mensaje de respuesta, ambos procesos tienen una configuración asignada que reenvía los mensajes de estas operaciones hacía la operación de grabado de respuesta.

Ambas operaciones tienen configurada de igual manera que los servicios de lectura unos data source name de su base de datos en cuestión para poder acceder a ella.

Finalmente, las operaciones al igual que los servicios también admiten la programación de funciones, por lo que tienen programada la función de grabado de registro, el cual recoge los valores del registro y realiza un insert en la tabla de evento de destino y por otro lado la operación de respuesta hace un update del estado del evento iniciado en el hospital de origen

## **6.8. Envío de información**

En cuanto las normativas y regulaciones establecidas finalmente se ha optado por el modelo más conveniente para el prototipo, teniendo en cuenta que las dimensiones no son muy elevadas se ha decidido que lo más adecuado puede ser una mensajería sencilla para la comunicación entre ambos sistemas, un sistema de envío de paquetes que pasará a través de los diferentes sistemas, un registro con los valores necesarios para poder incluir los nuevos valores donde sea conveniente.

## **6.9. Testing**

Para realizar las pruebas del correcto funcionamiento del proyecto es necesario realizar todo lo necesario para asegurar que se tiene en cuenta cada posibilidad que se pueda dar dentro del proyecto. Se realizarán las siguientes pruebas:

- Creación de una nueva alergia de paciente en un hospital y posterior comprobación de que aparece en el otro hospital y lo mismo al revés
- Creación de una nueva alergia de paciente con un paciente inexistente y comprobar que el estado de la transacción es pendiente
- Creación de una nueva alergia de paciente con una alergia inexistente y comprobar que el estado de la transacción es pendiente
- Creación de una alergia de paciente ya existente con una vigencia exenta para marcar esa alergia de paciente como exenta

### 6.9.1. Creación de nueva alergia de paciente en Hospital 1

Primeramente se procede a la creación de una alergia en el Hospital 1 por lo que se hace uso del formulario configurado.

The screenshot shows a form with the following fields:

- paciente\_NHC:** 6201478477
- alergia\_id:** 9VD0PG1NK68
- fecha\_diagnostico:** 18/05/2024
- vigencia:**

Figura 6.25: Inserción de la alergia del paciente en el formulario

Se utilizará al paciente con número de historia clínica 6201478477 y se le registrará la nueva alergia con código 9VD0PG1NK68 a día 18 de mayo de 2024 y como la alergia es nueva y se quiere registrar y no finalizar la vigencia previa, se marca la casilla de vigencia.

Tras la inserción se puede observar como el registro aparece en la tabla de evento del hospital de origen.

The screenshot shows a SQL query window with the following query and results:

```

SELECT TOP (1000) [id]
      ,[DNI]
      ,[id_alergia_oficial]
      ,[estado]
      ,[fecha_diagnostico]
      ,[vigencia]
FROM [EHR1].[dbo].[evento]
  
```

	id	DNI	id_alergia_oficial	estado	fecha_diagnostico	vigencia
1	200043	76549589X	66253100fc13ae5974a241ba	0	2024-05-18	1

Figura 6.26: Consulta de la tabla evento en hospital de origen

El estado de este registro es cero, ya que todavía no ha sido iniciada la producción de IRIS, una vez activada, el registro cambia a uno en cuanto es leído.

```

SELECT TOP (1000) [id]
, [DNI]
, [id_alergia_oficial]
, [estado]
, [fecha_diagnostico]
, [vigencia]
FROM [EHR1].[dbo].[evento]

```

	id	DNI	id_alergia_oficial	estado	fecha_diagnostico	vigencia
1	200043	76549589X	66253100fc13ae5974a241ba	2	2024-05-18	1

Figura 6.27: Consulta de la tabla evento en hospital de origen

Inmediatamente se puede observar como el estado cambia a dos, debido a la rapidez de todo el proceso, el estado dos indica que la inserción del registro en el otro hospital ha sido satisfactoria, por lo que en este momento se podría decir que la prueba ha funcionado correctamente.

Para más respaldo, IRIS ofrece un seguimiento de los mensajes y registros que fluyen por el proceso.

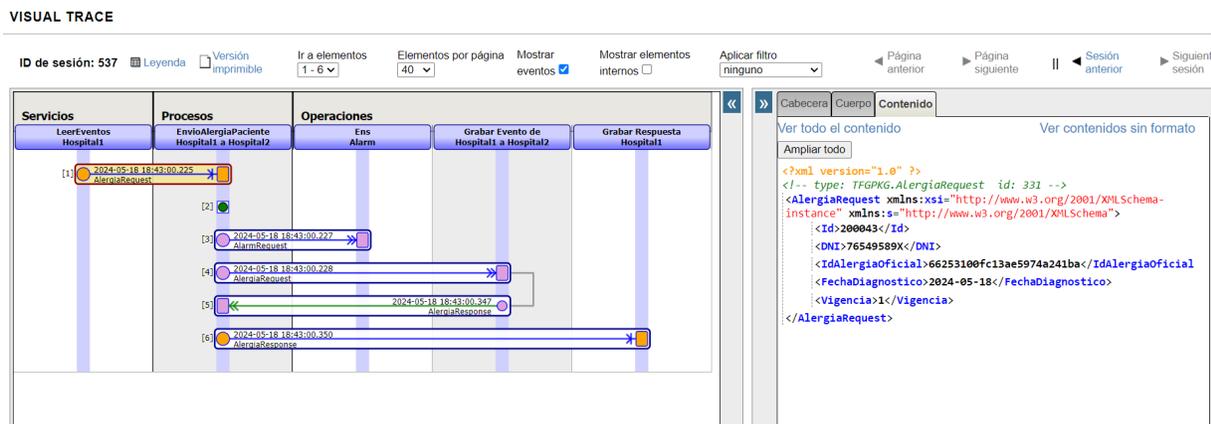


Figura 6.28: Traza del mensaje de registro de la alergia: lectura registro

En la imagen se pueden observar varias columnas que simbolizan los puntos por los que ha pasado el mensaje, de arriba a abajo fluye el recorrido del proceso. Se puede observar como el servicio de lectura transporta el mensaje hasta el proceso, también se puede observar su contenido, el cual contiene la información de la alergia del paciente previamente introducido.

IRIS permite navegar entre los diferentes puntos temporales del mensaje.

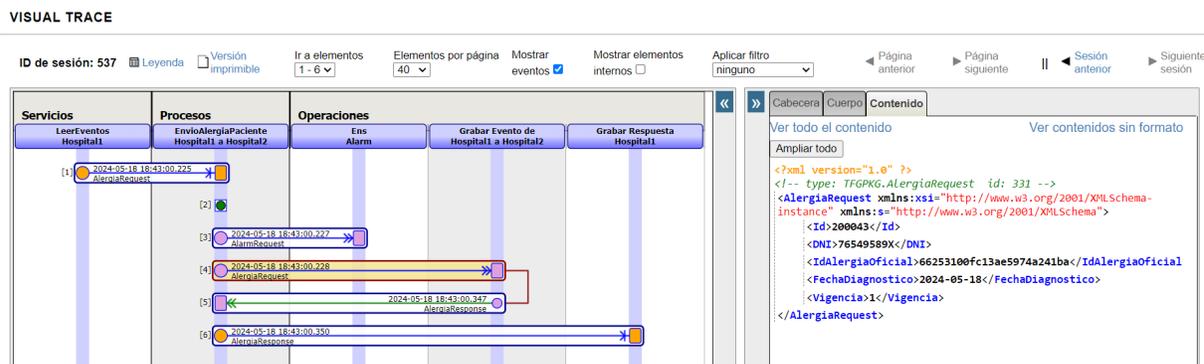


Figura 6.29: Traza del mensaje de registro de la alergia: procesamiento del mensaje hacía operación

Al hacer clic en cualquiera de los puntos del mensaje, se viaja hacía él, por ejemplo el mensaje que lleva de proceso a operación muestra como el contenido del mensaje sigue siendo el mismo que al inicio, por el momento parece funcionar correctamente.

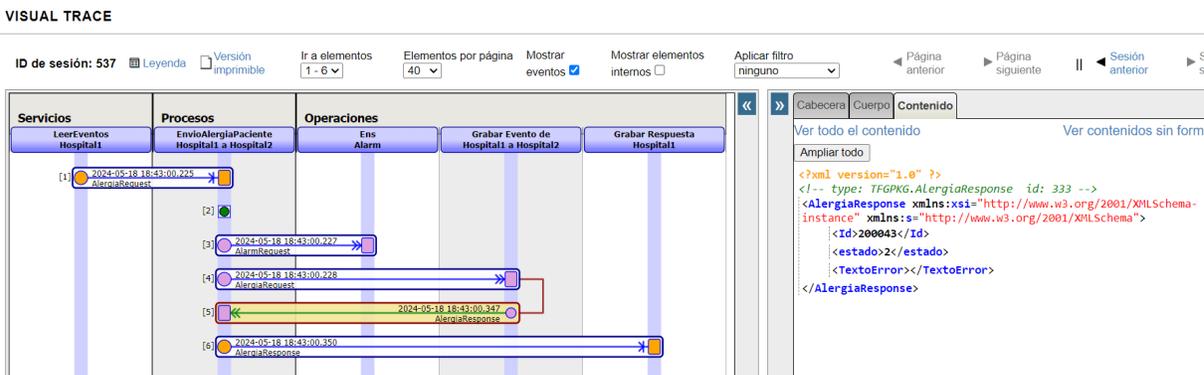


Figura 6.30: Traza del mensaje de registro de la alergia: respuesta de la operación

En el mensaje de respuesta se observa como ha sido satisfactoria la inserción, ya que el mensaje que devuelve contiene un dos, este dos es el mismo que el previamente mostrado en el hospital de origen.

El siguiente paso consiste en la comprobación de la inserción directamente el hospital de destino.

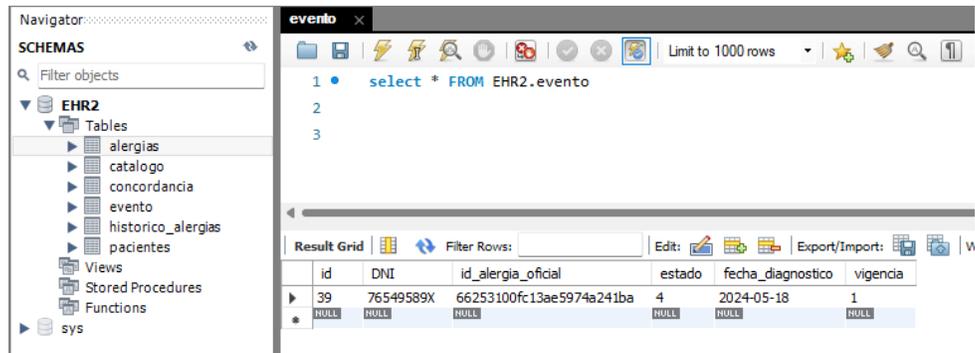


Figura 6.31: Consulta de la tabla de evento en hospital de destino

Al hacer una consulta en el hospital de destino en la tabla evento, se puede observar como el registro aparece, este contiene el estado 4 por lo que debería introducirse directamente en el registro de alergias de pacientes de esta base de datos.

Para consultar si este registro ha sido introducido procedemos a utilizar el formulario correspondiente de consulta.

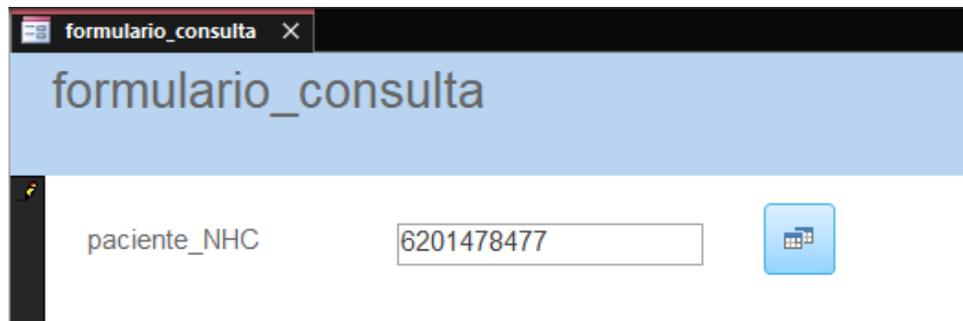


Figura 6.32: Consulta de la tabla de registro de alergias de paciente en hospital de destino

Al introducir el NHC del paciente en cuestión, este nos retorna todos sus registros.

id	paciente_NI	alergia_id	fecha_diagn	observacion	vigencia
96	6201478477	1471	18/05/2024	from EHR1	1
97	6201478477	631398	15/03/2023	lacinia aenean	0

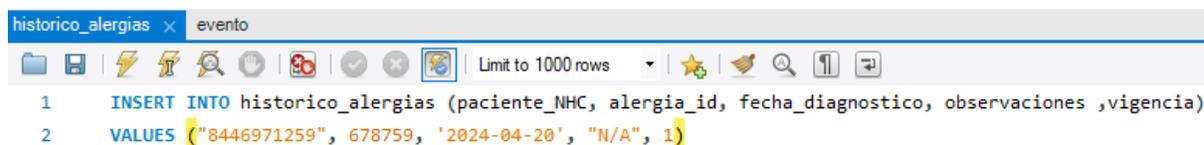
Figura 6.33: Consulta de la tabla de registro de alergias de paciente en hospital de destino

Finalmente al hacer una consulta de las alergias de este paciente, se puede observar como el paciente en cuestión ya tenía una alergia registrada, la cual no está vigente y además también aparece la alergia introducida en el hospital de origen, esta aparece con el mismo número de historia clínica que al inicio, pero con diferente id de alergia, esto lo único que sugiere es que este hospital registra las alergias en un formato distinto, con distinto nombre y código, pero se refiere a la misma alergia.

Ante este resultado se puede confirmar que la inserción ha sido un éxito y la alergia de paciente ha aparecido en ambos hospitales, esto significa que el paciente podría ir al hospital destino y aparecer en él como si hubiera sido allí dónde se realizó la prueba.

### 6.9.2. Creación de nueva alergia de paciente en Hospital 2

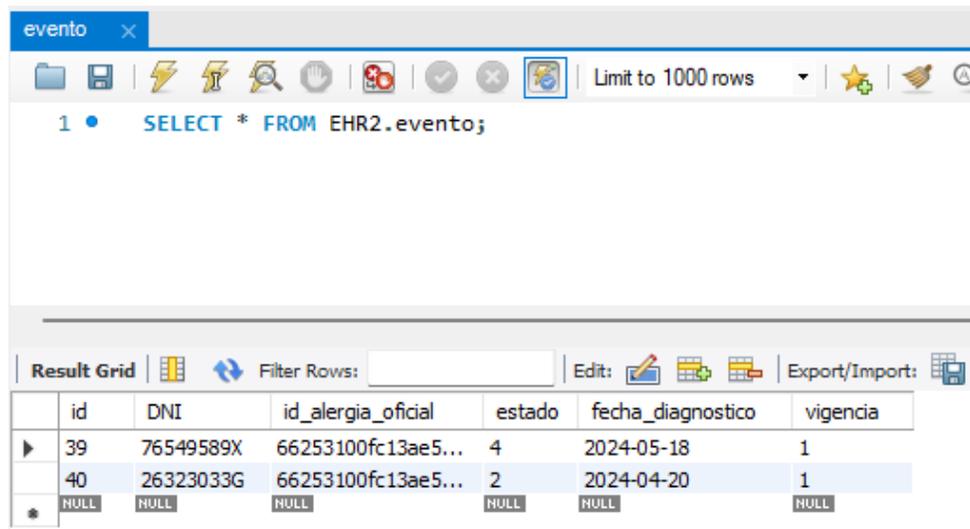
Para el test de esta prueba no se hará un seguimiento tan exhaustivo, se realizarán las comprobaciones oportunas.



```
historico_alergias x evento
Limit to 1000 rows
1  INSERT INTO historico_alergias (paciente_NHC, alergia_id, fecha_diagnostico, observaciones ,vigencia)
2  VALUES ("8446971259", 678759, '2024-04-20', "N/A", 1)
```

Figura 6.34: Consulta de inserción en base de datos de Hospital 2

Para la creación de esta alergia a diferencia de la inserción anterior, se realiza una consulta de inserción directa en la base de datos de origen.



The screenshot shows a database query tool window titled 'evento'. The query executed is `SELECT * FROM EHR2.evento;`. The results are displayed in a grid with the following columns: id, DNI, id\_alergia\_oficial, estado, fecha\_diagnostico, and vigencia.

	id	DNI	id_alergia_oficial	estado	fecha_diagnostico	vigencia
▶	39	76549589X	66253100fc13ae5...	4	2024-05-18	1
	40	26323033G	66253100fc13ae5...	2	2024-04-20	1
*	NULL	NULL	NULL	NULL	NULL	NULL

Figura 6.35: Consulta de la tabla de evento de Hospital 2

Se puede observar como el nuevo evento ha sido registrado (id = 40), el siguiente paso será comprobar como este mensaje ha pasado hacia el otro Hospital, el registro de IRIS es el siguiente.

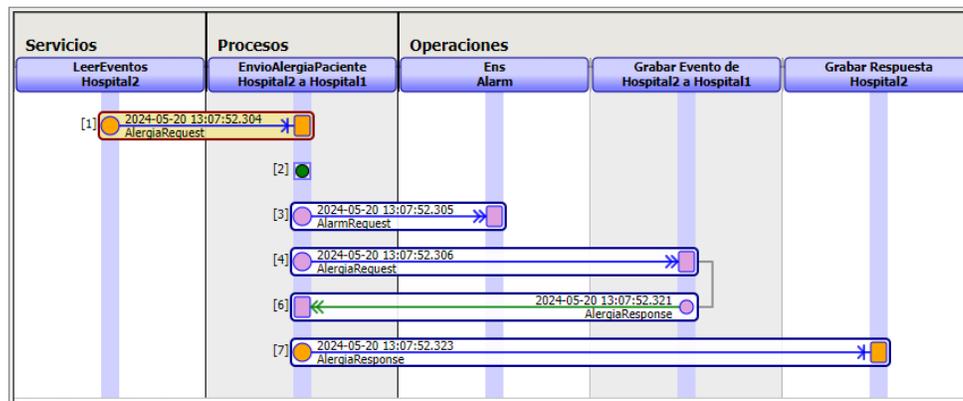


Figura 6.36: Traza del mensaje en IRIS

El mensaje se ha transmitido sin ningún problema y siguiendo todos los pasos requeridos según IRIS, se procede a comprobar en el hospital de destino.

	id	DNI	id_alergia_oficial	estado	fecha_diagnostico	vigencia
1	200043	76549589X	66253100fc13ae5974a241ba	2	2024-05-18	1
2	200044	26323033G	66253100fc13ae5974a2419a	4	2024-04-20	1

Figura 6.37: Consulta de la tabla evento en el Hospital 1

En la figura se puede observar como se ha registrado con éxito el evento.

### 6.9.3. Creación de una alergia de un paciente con vigencia exenta

Por otro lado una funcionalidad no tan importante, pero esencial para el proyecto, es la posibilidad de inhabilitar alergias de pacientes en caso de que esta haya cambiado.

Para eso se empleará la alergia previamente introducida con el número de historia clínica 6201478477 y el identificador de alergia 9VD0PG1NK68, a diferencia de la anterior, este registro tendrá una fecha posterior además de desmarcar la casilla de vigencia.

dbo\_paciente\_alergias

paciente\_NHC: 6201478477

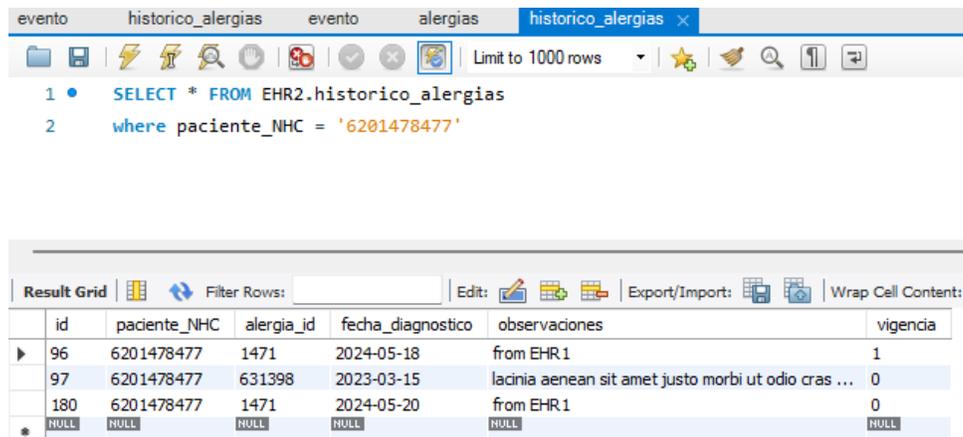
alergia\_id: 9VD0PG1NK68

fecha\_diagnostico: 20/05/2024

vigencia:

Figura 6.38: Inserción de alergia de paciente ya existente con vigencia exenta

Se puede observar en la imagen como la alergia del paciente exenta ha sido registrada en el formulario del Hospital 1. El siguiente paso será comprobar de que manera aparece en el hospital de destino, en este caso el Hospital 2.



The screenshot shows a database query tool interface. At the top, there are tabs for 'evento', 'historico\_alergias', 'evento', 'alergias', and 'historico\_alergias'. Below the tabs is a toolbar with various icons and a 'Limit to 1000 rows' dropdown. The SQL query is displayed as follows:

```
1 • SELECT * FROM EHR2.historico_alergias
2 • where paciente_NHC = '6201478477'
```

Below the query, a 'Result Grid' is shown with the following data:

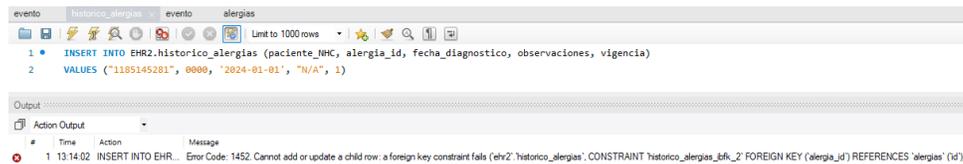
	id	paciente_NHC	alergia_id	fecha_diagnostico	observaciones	vigencia
▶	96	6201478477	1471	2024-05-18	from EHR 1	1
	97	6201478477	631398	2023-03-15	lacinia aenean sit amet justo morbi ut odio cras ...	0
	180	6201478477	1471	2024-05-20	from EHR 1	0
*	NULL	NULL	NULL	NULL	NULL	NULL

Figura 6.39: Consulta de alergia de paciente en el registro de alergias del Hospital 2

Se puede observar como aparecen ambos registros, la diferencia está en que el nuevo registro tiene una fecha más reciente por lo que se entenderá como el válido.

#### 6.9.4. Creación de registros incorrectos

Finalmente, la comprobación de la inserción de registros incorrectos. Para realizar esta prueba se procede a la directa inserción en ambos hospitales.



The screenshot shows a database query tool interface. At the top, there are tabs for 'evento', 'historico\_alergias', 'evento', 'alergias'. Below the tabs is a toolbar with various icons and a 'Limit to 1000 rows' dropdown. The SQL query is displayed as follows:

```
1 • INSERT INTO EHR2.historico_alergias (paciente_NHC, alergia_id, fecha_diagnostico, observaciones, vigencia)
2 • VALUES ('1185145281', 0000, '2024-01-01', 'N/A', 1)
```

Below the query, an 'Output' section shows the execution results:

#	Tina	Action	Message
1	13:14:02	INSERT INTO EHR...	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('ehr2.'historico_alergias', CONSTRAINT 'historico_alergias_bfk_2' FOREIGN KEY ('alergia_id') REFERENCES 'alergias' ('id'))

Figura 6.40: Consulta de inserción en Hospital 2 con alergia inexistente

En la imagen se observa como al intentar introducir un registro con una alergia incorrecta, salta un error y el modelo no permite la inserción, ya que detecta esa alergia como inexistente.

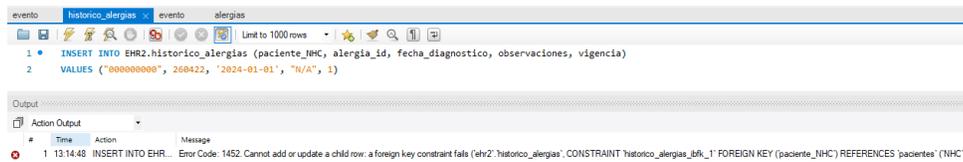


Figura 6.41: Consulta de inserción en Hospital 2 con paciente inexistente

En la imagen se observa como al intentar introducir un registro con una paciente incorrecto, salta un error y el modelo no permite la inserción, ya que detecta ese paciente como inexistente.

Por otro lado para probar en el Hospital 1 se realiza la inserción desde el formulario correspondiente.

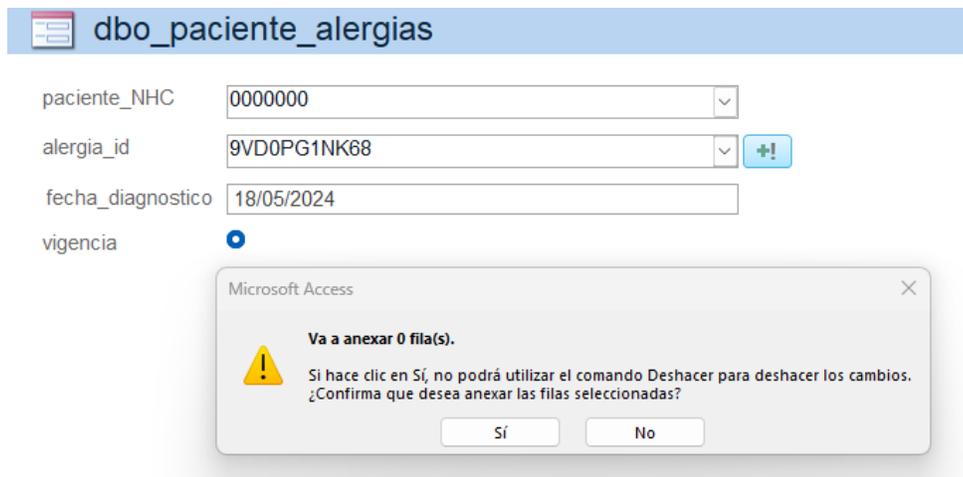
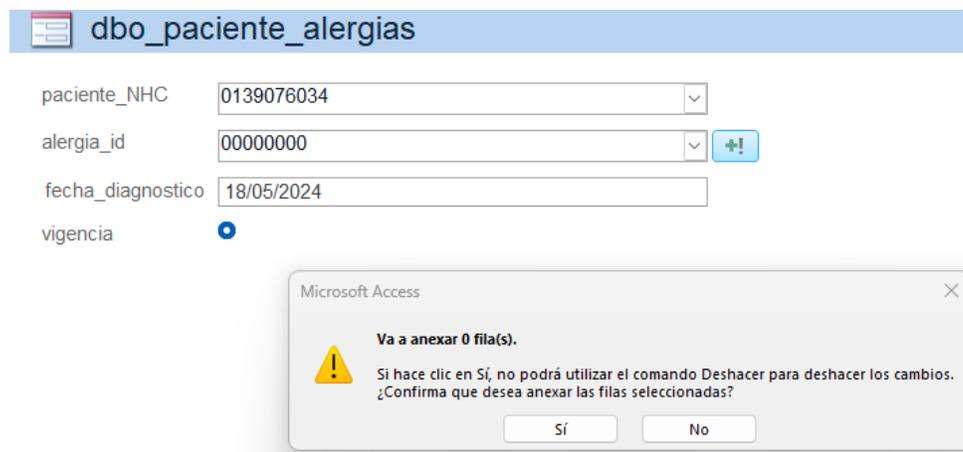


Figura 6.42: Inserción de alergia de paciente con paciente inexistente

En la figura se muestra como Microsoft Access muestra un mensaje el cual informa de las filas que añadirá esa consulta, en este caso 0 por lo que al introducir un paciente incorrecto el modelo no permite la inserción.



The screenshot shows a Microsoft Access form titled "dbo\_paciente\_alergias". The form has four fields: "paciente\_NHC" with the value "0139076034", "alergia\_id" with the value "00000000", "fecha\_diagnostico" with the value "18/05/2024", and "vigencia" which is currently empty. A blue "+!" button is located to the right of the "alergia\_id" field. A warning dialog box is overlaid on the form, titled "Microsoft Access". The dialog box contains a yellow warning triangle icon and the text: "Va a anexar 0 fila(s). Si hace clic en Sí, no podrá utilizar el comando Deshacer para deshacer los cambios. ¿Confirma que desea anexar las filas seleccionadas?". At the bottom of the dialog box are two buttons: "Sí" and "No".

Figura 6.43: Inserción de alergia de paciente con alergia inexistente

Por otro lado de igual manera que con el paciente inexistente, al introducir una alergia de paciente con un identificador de alergia inexistente el modelo no añade nada.

## **7. Conclusiones**

### **7.1. Resumen de los resultados obtenidos**

El propósito principal del proyecto era la creación y confección de un prototipo de interoperabilidad utilizando IRIS de InterSystems.

Por un lado, la creación de un entorno adecuado para integrar el prototipo en cuestión, un entorno lo más alterado posible y con la mayor diferenciación entre hospitales.

Por otro lado la interoperabilidad del entorno, no únicamente con IRIS sino también incidiendo en los hospitales, añadiéndoles diferentes controles de eventos entre otros ajustes.

El objetivo principal se basaba en demostrar cómo se puede lograr la interoperabilidad entre ambos hospitales utilizando la plataforma IRIS.

Se configuraron las dos bases de datos requeridas por el proyecto, cada una representando un hospital con sus propios sistemas de información. A través de IRIS, se establecieron servicios, procesos y operaciones que permitieron la comunicación y el intercambio de datos entre estos sistemas heterogéneos. Los resultados obtenidos demostraron que es posible integrar y compartir información de manera eficiente y segura, mejorando la coordinación y la toma de decisiones clínicas entre los hospitales.

### **7.2. Implicaciones teóricas y prácticas**

Los resultados obtenidos tienen varias implicaciones teóricas y prácticas:

#### **7.2.1. Teóricas**

Este trabajo contribuye al conocimiento sobre interoperabilidad en sistemas de información de salud, específicamente en los registros de alergias de pacientes, demostrando que es posible interoperar sistemas heterogéneos mediante la construcción de un prototipo de interoperabilidad.

### **7.2.2. Prácticas**

En un contexto real, la implementación de sistemas interoperables puede llevar a una mejor gestión de la información de hospitales, en concreto de la información de alergias de pacientes, mejorando y optimizando los procesos clínicos y administrativos. Los resultados sugieren de forma positiva la adopción de sistemas de interoperabilidad en hospitales para mejorar sus servicios y la calidad de atención al cliente.

## **7.3. Limitaciones y posibles mejoras**

A pesar de los resultados positivos, el proyecto presenta ciertas limitaciones a tener en cuenta:

### **7.3.1. Escala del prototipo**

El prototipo fue desarrollado en un entorno ciertamente controlado con un número limitado de hospitales y de flujos de datos. En un entorno real, con un volumen mucho mayor de datos y usuarios, podrían surgir desafíos adicionales, como por ejemplo:

- Un mayor volumen de hospitales
- Multitud de flujos de entrada de datos, no solo desde hospitales locales sino datos externos de otros países
- Un mayor volumen de datos

### **7.3.2. Diversidad de sistemas**

La interoperabilidad realizada en este proyecto esta basada en únicamente dos hospitales con entradas de datos únicas en cada hospital, en un entorno real, la diversidad de distintos sistemas y distintos tipos de datos y tecnologías suponen multitud de nuevos retos a tener en cuenta.

### **7.3.3. Recursos y tiempo**

Las limitaciones de tiempo y recursos influyeron directamente en el alcance del proyecto. Con más tiempo y recursos, se podrían haber desarrollado más y mejores funcionalidades, con nuevas y diversas tecnologías.

## **7.4. Recomendaciones para futuras investigaciones**

Varias cosas a tener en cuenta para futuras investigaciones pasan por:

### **7.4.1. Ampliar el alcance del estudio**

Incluir más sistemas de información diferentes para probar la escalabilidad y flexibilidad del prototipo de interoperabilidad.

### **7.4.2. Integración con tecnologías de IRIS**

Como se ha mencionado durante el proyecto, una de las limitaciones ha sido la falta de uso de las distintas tecnologías que IRIS ofrece, por lo que una recomendación sería exprimir algunas de las tecnologías más interesantes que ofrece IRIS para la manipulación de los datos.

### **7.4.3. Evaluación de impacto**

Evaluar el impacto de la interoperabilidad en sistemas hospitalarios, teniendo en cuenta puntos como: La calidad de la atención al paciente, la eficiencia operativa o la satisfacción del personal médico y administrativo.

## **7.5. Conclusión final**

El desarrollo del prototipo de interoperabilidad con IRIS de InterSystems ha demostrado que tan viable es la integración de sistemas de información hospitalaria heterogéneos. Los resultados obtenidos indican que, con la tecnología adecuada, es posible mejorar significativamente el intercambio de información y la coordinación entre diferentes hospitales.

A pesar de las limitaciones previamente expuestas, este trabajo sienta las bases para futuras investigaciones y desarrollos en el campo de la interoperabilidad, con el potencial de transformar la gestión de la información en el sector sanitario y mejorar la calidad de atención al paciente.



## 8. Bibliografía

- [1] National Interoperability Framework Observatory (NIFO). “Interoperability layer”. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://joinup.ec.europa.eu/collection/nifo-national-interoperability-framework-observatory/3-interoperability-layers>
- [2] InterSystems. “Interoperability”. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://www.intersystems.com/data-platform/interoperability/>
- [3] InterSystems. “Ciencias de la salud y de la vida”. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://www.intersystems.com/es/industrias/tecnologia-sanitaria/>
- [4] European Health Data Space (EHDS). “The European Health Data Space (EHDS)”. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://www.european-health-data-space.com/>
- [5] Atlassian. “Jira — Software de seguimiento de proyectos e incidencias — Atlassian”. Collaboration software for software, IT and business teams. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://www.atlassian.com/es/software/jira>
- [6] José Manuel Gómez Montón. “Interoperabilidad de los sistemas de salud: ventajas y retos”. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://www.ehcos.com/interoperabilidad-los-sistemas-salud/#:~:text=En%20el%20sector%20de%20la,y%20utilizar%20la%20informaci%20intercambiada.>
- [7] Ricardo Ollarves. “7 razones para elegir MySQL como gestor de base de datos”. Blog Interfell. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://blog.interfell.com/7-razones-para-elegir-mysql>
- [8] Marc Bolufer Gil. “Análisis de las Ventajas y Desventajas de Access: ¿Qué beneficios y limitaciones ofrece esta herramienta? - Ventajas y desventajas top”. Ventajas y desventajas top. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://ventajasydesventajastop.com/access-ventajas-y-desventajas/>
- [9] InterSystems. “IRIS InterSystems”. Accedido el 10 de enero de 2024. [En línea]. Disponible: <https://www.intersystems.com/es/sistema-de-gestion-de-bases-de-datos/gestion-de-bases-de-datos/>

[10] Pierre-Yves Duquesnoy, "Tackling Complex Interoperability Challenges with a Data-First Approach - Sales Engineer at InterSystems," LinkedIn, <https://www.linkedin.com/in/pierre-yves-duquesnoy/>, consultado en enero 12, 2024.

[11] D. Ray, S., & P. Gray, J. (2019). ADRs, ADEs and SEDs: A bird's eye view. ScienceDirect. <https://www.sciencedirect.com/science/article/abs/pii/S0378608019300583?ref=letsgetchecked-blog.ghost.io>, consultado el 15 de marzo de 2024.

[12] Mockaroo - Random Data Generator and API Mocking Tool — JSON / CSV / SQL / Excel. (s.f.). Mockaroo - Random Data Generator and API Mocking Tool — JSON / CSV / SQL / Excel. <https://www.mockaroo.com/>