

Universidad de las Ciencias Informáticas

Facultad 1

CESIM



Trabajo de Diploma para optar por el título de Ingeniero en Ciencias
Informáticas

**Desarrollo de las funcionalidades para la configuración de los
esquemas de vacunación en el sistema XAVIA-HIS.**

Autor: Daniela Rivera Feria

Tutor(es): Ing. Lourdes Escalona Peral

Ing. Luis M. Rosa Caballero

La Habana, septiembre 2020

"Año 62 de la Revolución"

DECLARACIÓN DE AUTORÍA

Declaro ser la única autora del trabajo de diploma "Desarrollo de las funcionalidades para la configuración de los esquemas de vacunación en el sistema XAVIA-HIS" y autorizo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 24 días del mes de septiembre del año 2020.



Daniela Rivera Feria

Autor



Ing. Lourdes Escalona Peral

Tutora



Ing. Luis M. Rosa Caballero

Tutor

Esta tesis está dedicada a:

- Mi mamá y mi papá porque son el origen de toda mi existencia.
- Eliani y Yoyi porque fueron mis amigos desde la infancia y la vida se los llevó demasiado rápido.
- Mi abuelo Laffita porque fue como un padre para mí.

AGRADECIMIENTOS

Agradezco en primer lugar a Dios por todo lo bueno que hay en esta vida, gracias por cada prueba en el camino, por cada obstáculo que logré superar, que me hizo crecer y seguir adelante.

A mí familia por los valores que me inculcaron, por su apoyo incondicional y por colaborar para que sea una buena profesional. Mamá, Papá, ustedes saben que son todo para mí, podría llenar hojas enteras de agradecimientos, solamente para ustedes, pero sería una tesis interminable, solo me queda decir " Matojo, sacamos sobresaliente".

A mí tutora Lourdes por su paciencia, sus consejos y acompañamiento que me formaron como investigadora, profesional y una mejor persona. A los trabajadores del centro les agradezco por ayudarme con aspectos importantes tanto del documento como de la aplicación.

A mis monis, Jessica y Yairys, mis mejores amigas y a Ro por estar siempre para mí y brindarme su ayuda en los momentos más difíciles.

Josué, eres y siempre serás importante en mi vida, gracias por estar 24/7 para mí y por ponerme siempre en primer lugar. Por favor gradúate ya.

A mis amigos porque a pesar de que no tuvieron incidencia directa en esta investigación tenerlos cerca me hizo bien, con ustedes los 5 años de la universidad se me fueron volando. Ustedes saben quiénes son, pero si les queda duda: Lisbeth, Eliani, Roxana, Amanda, Michel.

GRACIAS, GRACIAS, GRACIAS.

Los Sistemas de Información Hospitalaria (HIS) surgen con el propósito de apoyar la gestión de la información en las instituciones de salud. Cuba, se ha enfocado siempre en el desarrollo de todos los sectores de la sociedad dándole un seguimiento especial al sector de la salud, es por eso que ha incorporado las tendencias actuales para mejorar el proceso de atención médica.

El Centro de Informática Médica (CESIM) de la Universidad de las Ciencias Informáticas (UCI) desarrolló el sistema XAVIA-HIS el cual tiene como atributo fundamental una Historia Clínica Electrónica (HCE) única por paciente que interconecta las diferentes áreas de un hospital. En el sistema, el área de Epidemiología, permite gestionar: las encuestas de Infecciones Asociadas a Asistencias Sanitarias (IAAS), las fichas de Enfermedades de Declaración Obligatoria (EDO), la generación de reportes del módulo y las Incidencias de Inspección Sanitaria (IIS). Sin embargo, las funcionalidades que permiten la configuración de las vacunas y los esquemas de vacunación no están implementadas.

Por tanto, el objetivo de la investigación consiste en: Desarrollar las funcionalidades que permitan la configuración de los esquemas de vacunación en el sistema XAVIA-HIS para lograr que desde Emergencias, Hospitalización y Consulta Externa los médicos especialistas puedan visualizar las vacunas aplicadas a sus pacientes y las enfermeras puedan tener un control del carnet de vacunación y de la asignación de citas para próximas vacunas.

Palabras clave: epidemiología, esquemas de vacunación, sistemas de información hospitalaria, vacunas.

The Hospital Information Systems (HIS) arise with the purpose of supporting the management of information in health institutions. Cuba has always focused on the development of all sectors of society, giving special monitoring to the health sector, that is why it has incorporated current trends to improve the medical care process.

The Medical Informatics Center (CESIM) of the University of Informatics Sciences (UCI) developed the XAVIA-HIS system, which has as a fundamental attribute a single Electronic Medical Record (EHR) per patients that interconnects the different areas of a hospital. In the system, the Epidemiology areas allow to manage: The Infections Associated with Healthcare Surveys (IAAS), the Compulsory Declaration Diseases (EDO), the generation of module reports and the Health Inspection Incidences (IIS). However, the functionalities that allow the configuration of vaccines and vaccination schedules are not implemented.

Therefore, the objective of the research consists of: Developing the functionalities that allow the configuration of the vaccination schedules in the XAVIA-HIS system to achieve that from Emergencies, Hospitalization and Outpatient Consultation, specialist doctors can view the vaccines applied to their patients and the nurses can have a control of the vaccination card and the allocation of appointments for next vaccines.

Key words: epidemiology, vaccination schemes, hospital information systems, vaccines.

Índice

Introducción	1
Capítulo 1: Fundamentos teóricos y metodológicos de la investigación	7
1.1- Conceptos asociados a la investigación	7
1.2- Proceso de vacunación en las áreas de salud cubanas	12
1.3- Sistemas existentes vinculados al campo de acción	12
1.3.1- Análisis de los sistemas identificados	14
1.4- Metodología de desarrollo del software (AUP-UCI)	16
1.5- Entorno de Desarrollo	17
1.5.1- Lenguajes	17
1.5.2- Tecnologías a utilizar	18
1.5.3- Herramientas para el desarrollo	20
1.6- Conclusiones Parciales	22
Capítulo 2: Análisis y diseño de la propuesta de solución	23
2.1- Descripción de la propuesta de solución	23
2.2- Modelo conceptual	24
2.3- Modelo de negocio	24
2.4- Requisitos de software	25
2.4.1- Requisitos funcionales	25
2.4.2- Requisitos no funcionales	32
2.5- Modelo de diseño	35
2.5.1- Patrones arquitectónicos	35
2.5.2- Patrones de diseño	36
2.5.3- Diagrama de paquetes.....	38
2.5.4- Modelo de datos	39

2.5.5- Diagrama de clases de diseño.....	41
2.6- Conclusiones Parciales.....	42
Capítulo 3: Validación de la propuesta de solución.....	43
3.1- Modelo de implementación.....	43
3.2- Estrategias de codificación. Estándares y estilos usados.....	44
3.3- Seguridad Informática.....	46
3.4- Tratamiento de errores.....	46
3.5- Descripción de la propuesta de solución.....	47
3.6- Pruebas de software.....	50
3.6.1- Método de caja blanca. Técnica de camino básico.....	51
3.6.2- Pruebas de integración. Método de caja negra. Técnica de participación equivalente.	53
3.6.3- Pruebas de aceptación. Método de caja negra. Técnica de participación equivalente.	55
3.7- Conclusiones parciales.....	56
Conclusiones.....	57
Recomendaciones.....	58
Referencias bibliográficas.....	59
Anexos.....	66
Anexo 1: Descripción de requisitos por proceso.....	66
Anexo 2: Vista del RF2 "Ver detalles de la vacuna".....	70
Anexo 3: Vista del RF5 "Modificar vacuna".....	71
Anexo 4: Vista del RF6 " Eliminar vacuna".....	71
Anexo 5: Vista del RF3 " Buscar vacuna".....	72
Anexo 6: Viste del RF4 " Ver datos de la vacuna".....	72

Anexo 7: Vista del RF9 " Ver detalles del esquema de vacunación" 73

Anexo 8: Ver RF10 " Buscar esquema de vacunación" 73

Anexo 9: Ver RF11 " Ver datos del esquema de vacunación" 74

Anexo 10: Vista del RF12 " Modificar esquema de vacunación" 74

Anexo 11: Vista del RF13 " Eliminar esquema de vacunación" 75

Glosario de términos 76

Introducción

La salud es uno de los campos del saber en el cual las Tecnologías de la Información y las Comunicaciones (TIC) tienen un papel protagónico. Las TIC son un conjunto de herramientas para la transmisión, procesamiento y almacenamiento digitalizado de la información, que se emplean para mejorar la calidad de los servicios, logrando una gestión más eficiente y cómoda para los ciudadanos. Siendo su objetivo principal lograr que las personas tengan una mejor calidad de vida. (García Garcés et al., 2014)

El Big Data es un concepto utilizado dentro de las TIC para referirse a la gestión y análisis de enormes volúmenes de datos con el objetivo de recolectar información útil. Durante la atención médica se logra este proceso con los cuales después de ser analizados se puede conseguir: predecir epidemias, evitar muertes, mejorar la calidad de vida y desarrollar nuevos medicamentos y tratamientos. (Clinic Cloud, 2015)

La digitalización de los registros de salud es también uno de los grandes avances y beneficios que presentan las TIC, ingresar información en un sistema electrónico consume menos tiempo que los métodos basados en papel y reduce el riesgo de errores en los datos de los pacientes. (Clinic Cloud, 2015). El acceso a dichos registros es instantáneo y a través de dispositivos portátiles o de servicios en la nube de modo que los usuarios pueden acceder a la información mediante una conexión a Internet desde un dispositivo móvil o fijo en cualquier lugar, lo que aumenta la productividad en la atención sanitaria. (SERRANO PERELLÓ, 2016)

A pesar de sus limitaciones, Cuba tiene como propiedad la informatización de la sociedad y con la inclusión de las TIC en la esfera de la salud se busca el mejoramiento de la productividad en las instituciones hospitalarias y en el trato a los pacientes. El Ministerio de Salud Pública (MINSAP), órgano rector del Sistema Nacional de Salud (SNS) convocó para ello a un conjunto de instituciones que forman parte del Ministerio de Informática y las Comunicaciones, entre las que se encontraba la Universidad de las Ciencias Informáticas (UCI).

La UCI cuenta con un Centro productivo de Informática Médica (CESIM) que se ha enfocado en desarrollar productos para la optimización del trabajo y mejoramiento de la calidad de la atención médica. Dicho centro cuenta con varios sistemas de gestión siendo

el Sistema de Información Hospitalaria (HIS) el encargado de potenciar los procesos de dirección, incrementar la eficiencia administrativa y mejorar la calidad de los procesos mediante una mayor rapidez en el manejo de la información. (Pérez Labrador et al., 2018)

El HIS es una herramienta informática que posibilita la identificación y gestión de pacientes en un entorno hospitalario. Adicionalmente y dependiendo del tipo de solución debe permitir la gestión de los procesos y personas que intervienen en cualquier actividad. (Tich Consulting GRUPO ASISA, 2017)

El sistema XAVIA-HIS, producto creado en el CESIM presenta varias estructuras que le dan respuesta a la informatización de determinadas áreas de un hospital, entre las que se encuentran: Hospitalización, Emergencia, Epidemiología, Enfermería y Consulta Externa.

El departamento de Epidemiología de una institución de salud constituye el área del hospital que se encarga de procesar toda la información referente al comportamiento de las enfermedades y les da un seguimiento especial a aquellas con riesgo de contagio. Siguiendo esta filosofía, el módulo de Epidemiología del sistema XAVIA-HIS permite gestionar la información que se genera dentro de él y la proveniente de otros módulos del sistema que le son de interés para su departamento posibilitando:

- La gestión de las encuestas de Infecciones Asociadas a Asistencias Sanitarias (IAAS).
- El proceso de generación de los reportes del módulo.
- El proceso de gestión de las fichas de Enfermedades de Declaración Obligatoria (EDO).
- El proceso de generación de las Incidencias de Inspección Sanitaria (IIS). (Centro de Informática Médica, Departamento de Sistemas de Gestión Hospitalaria y Grupo de Mercadotecnia, 2018)

A partir de la experiencia de la autora y después de un análisis minucioso del sistema se pudo constatar que actualmente el HIS carece de las funcionalidades que permiten la configuración de las vacunas y los esquemas de vacunación existentes en el país lo que impide que médicos especialistas de áreas como Hospitalización, Emergencias y Consulta Externa puedan visualizar las vacunas aplicadas a sus pacientes y las que

tienen pendiente. La enfermera, actor encargado del proceso de vacunación no tiene un control riguroso sobre las vacunas que se le han aplicado a un paciente desde su nacimiento hasta la actualidad provocado por una falta de seguimiento del carnet de vacunación de una persona. Además, no se encuentra implementada la opción de notificación para las vacunas que deban ser reactivadas.

Después de realizar varias entrevistas a los especialistas de diferentes áreas de salud se definen como limitantes:

- La coexistencia de dos formas de trabajo, manual y automatizada hace engorroso el almacenamiento de la información generada en la especialidad de Epidemiología y trae consigo la pérdida y deterioro de los datos.
- La ausencia de las funcionalidades que permitan la configuración de las vacunas y los esquemas de vacunación impide el seguimiento oportuno del paciente acarreando el descontento de los mismos.
- La información escasa y desactualizada sobre la gestión de las vacunas en el módulo de Enfermería impide la asignación de citas de manera fácil y rápida para la aplicación de próximas vacunas.

Considerando la situación problemática se define el siguiente **problema a resolver**:
¿Cómo contribuir a la gestión de la información relacionada con los esquemas de vacunación del sistema XAVIA-HIS?

En correspondencia con el problema, el **objeto de estudio** está enfocado en el proceso de gestión de la información de los esquemas de vacunación en las instituciones hospitalarias.

Para solucionar el problema a resolver se define como **objetivo general** del trabajo de diploma: Desarrollar funcionalidades para la configuración de los esquemas de vacunación en el sistema XAVIA-HIS.

La investigación se enmarca en el **campo de acción** de la gestión de los esquemas de vacunación para el sistema XAVIA-HIS.

Una vez realizado el análisis general se plantean las siguientes **preguntas científicas**:

- ¿Cómo elaborar el marco teórico referente a la gestión de la información de los esquemas de vacunación en los sistemas de información hospitalaria de las instituciones de salud?
- ¿Cómo describir las herramientas, tecnologías y metodología utilizadas para la configuración de los esquemas de vacunación?
- ¿Cómo modelar el negocio y los requisitos del sistema a partir del análisis de las características y particularidades de los esquemas de vacunación?
- ¿Cómo implementar las funcionalidades para la configuración de los esquemas de vacunación propuestos?
- ¿Cómo integrar las funcionalidades al sistema XAVIA-HIS para comprobar que todos los elementos de la solución implementada funcionan de forma correcta en el sistema?

Para dar solución al objetivo general planteado se especifican las siguientes **tareas de la investigación:**

- Elaborar el marco teórico referente a la gestión de la información de los esquemas de vacunación en los sistemas de información hospitalaria de las instituciones de salud.
- Describir las herramientas, tecnologías y metodología utilizadas para la configuración de los esquemas de vacunación.
- Modelar el negocio y los requisitos del sistema a partir del análisis de las características y particularidades de los esquemas de vacunación.
- Implementar las funcionalidades para la configuración de los esquemas de vacunación propuestos.
- Integrar las funcionalidades al sistema XAVIA-HIS para comprobar que todos los elementos de la solución implementada funcionan de forma correcta en el sistema.

Los métodos científicos utilizados para desarrollar la investigación fueron:

Métodos teóricos

- **Analítico- Sintético:** se utilizó para el análisis de la teoría y la extracción de los principales conceptos a incluir en el marco teórico como son: (Sistemas de Información Hospitalaria y esquemas de vacunación). Luego de concluido este proceso se sintetizaron los resultados para dar una propuesta de solución.
- **Inductivo- Deductivo:** a través de este método se pudo arribar a conclusiones generales sobre la gestión de la información en los esquemas de vacunación a partir de las necesidades de los especialistas y de los sistemas informáticos estudiados.
- **Modelación:** se utilizó para representar por medio de diagramas el proceso de gestión de la información, teniendo como resultado un mejor entendimiento de la solución a implementar. Permitió la elaboración de los modelos ingenieriles establecidos por la metodología propuesta para guiar el desarrollo del software.

Métodos empíricos

- **Entrevista no estructurada:** se concretaron encuentros con el cliente mediante los cuales se pudo definir la necesidad de configurar los esquemas de vacunación, establecer los requisitos funcionales y observar el avance durante las fases de desarrollo y validación.
- **Observación:** se visitaron varias instituciones de salud, tales como, el hospital materno América Arias, el policlínico Antonio Pulido y varios consultorios de la UCI donde a través de la observación se pudo constatar cómo se lleva a cabo el proceso de vacunación en dichas áreas.

Con el desarrollo e integración de los esquemas de vacunación al módulo de Epidemiología del sistema XAVIA-HIS se esperan los siguientes beneficios:

- Gestión informatizada de la información referente a las vacunas y los esquemas de vacunación para proporcionar una mejora en la calidad de la atención médica.
- Garantizar la disponibilidad, completitud, seguridad y accesibilidad de la información de los pacientes.

- Facilitar el trabajo de los especialistas al tener toda la información correspondiente a los esquemas de vacunación en una HCE única.

El presente documento contiene tres capítulos a fin de lograr una adecuada organización de su información donde se exponen los resultados de la investigación realizada.

Capítulo 1: Fundamentación teórica y metodológica de la investigación:

En este capítulo se abordan los conceptos asociados a la investigación, se explica cómo se lleva a cabo el proceso de vacunación en las áreas de salud cubana. Además, se valoran los sistemas informáticos que se dedican a la implementación de los esquemas de vacunación y se describe la metodología, las herramientas y las tecnologías definidas para llevar a cabo la propuesta de solución.

Capítulo 2: Análisis y diseño de la propuesta de solución:

Describe la propuesta de solución a desarrollar basada en la metodología Proceso Unificado Ágil (AUP-UCI). Se exponen las etapas de desarrollo del software que permiten describir la propuesta de solución. Se presentan los artefactos que corresponden a esta etapa y además se presenta el patrón arquitectónico utilizado para el desarrollo de los esquemas de vacunación.

Capítulo 3: Diseño, implementación y prueba de la solución propuesta:

Especifica los diagramas referentes a las etapas de modelado de diseño. Modela las tablas de la base de datos generada y describe cada uno de sus atributos. Se realiza un estudio de los mecanismos para el tratamiento de errores y las pruebas pertinentes para la validación de la solución.

Capítulo 1: Fundamentos teóricos y metodológicos de la investigación

El presente capítulo tiene como objetivo abordar los conceptos básicos que brindan la base teórica y conceptual para el desarrollo de la solución propuesta. Se analizan un conjunto de sistemas informáticos similares, existentes a nivel internacional, que gestionan la información relacionada con las funcionalidades a desarrollar. Además, se hace referencia a la metodología de desarrollo, herramientas y tecnología a utilizar.

1.1- Conceptos asociados a la investigación

Para poder comprender de qué trata la investigación a la que se está haciendo referencia y facilitar la comprensión del funcionamiento de la especialidad de Epidemiología es necesario conocer los conceptos relacionados con las epidemias y los esquemas de vacunación.

Sistemas de Información de Salud

Los Sistemas de Información de Salud (SIS) tienen como objetivo normalizar, integrar y organizar toda la información de salud disponible en sus sistemas de información, en un repositorio accesible y seguro, así como la distribución de la información de la forma más conveniente para facilitar la toma de decisiones. (Canela-Soler et al., 2010)

Enfermedad

La Organización Mundial de la Salud (OMS) define enfermedad como: Alteración o desviación del estado fisiológico en una o varias partes del cuerpo, por causas en general conocidas, manifestada por síntomas y signos característicos, y cuya evolución es más o menos previsible. (Herrero Jaén, 2016)

Vacunas

Una vacuna es cualquier preparación destinada a generar inmunidad contra una enfermedad provocando la estimulación de anticuerpos. Las propiedades principales de una vacuna son seguridad y eficacia protectora. (Martínez, de Juanes Pardo y de Codes Ilario, 2015)

Esquemas de vacunación

El programa cubano de vacunación, creado en septiembre de 1962, tiene cuatro principios básicos: universalidad, integridad, participación y disponibilidad. Gracias a su implementación se ha logrado la erradicación de varias enfermedades y que otras no se consideren problemas de salud. (López Ambrón et al. 2018)

Los esquemas de vacunación recogen las vacunas que se le aplicarán a niños y adultos según la situación epidemiológica de la región o país en el que viven, especificando el nombre de la vacuna, número de dosis, las edades y vía de aplicación, así como la enfermedad que previene dicha vacuna. (Porras et al., 2006)

En la tabla1 se muestra el esquema nacional de vacunación vigente en Cuba para niños de 0-18 años de edad.

Tabla 1: Esquema nacional de vacunación cubano. (Elaboración propia)

Vacuna/ Protege	Edad de la dosis				Dosis (ml)	Vía	Región de la aplicación	Lugar de aplicación
	1 ^{ra}	2 ^{da}	3 ^{ra}	Reactivación				
BCG/ Tuberculosis	Al nacer	---	---	-----	0,05	ID	Deltoides Izquierdo	Maternidad
Heberbiovac HB/ Hepatitis B	Al nacer	---	---	-----	0,5	IM	1/3 medio de la cara antero- lateral del muslo	Maternidad
Heberpenta- L/Difteria, Tosferina,	2 m	4 m	6 m	18 m	0,5	IM	1/3 medio de la cara antero- lateral del	Policlínico

Tétanos, Hepatitis B y Haemophilus Influenzae Tipo B							muslo. A los 18 meses en el Deltoides	
VA- MENOGOC- BC/ Meningococo B y C	3 m	5 m	---	-----	0,5	IM	1/3 medio de la cara antero- lateral del muslo	Policlínico
IPV/ Poliomielitis	4m	8m	---	-----	0,1	ID	Deltoides Derecho	Policlínico
DTP-VAC*/ Difteria, Tétanos y Tosferina	-----	----	---	18 m	0,5	IM	Deltoides	Policlínico
Quimi-Hib*/ Hemofilus Influenzae Tipo B	-----	----	---	18 m	0,5	IM	Deltoides	Policlínico
PRS/ Parotiditis, Rubéola y Sarampión	12m	----	---	6 años	0,5	SC	Deltoides	Policlínico y Escuela

VA-DIFTET/ Difteria y Tétanos	-----	----	---	6 años	0,5	IM	Deltoides	Escuela
Vax-TYVI/ Fiebre Tifoidea	10 años	----	---	13 y 16 años	0,5	IM	Deltoides	Escuela
VAX- TET/Tétanos	-----	----	---	14 años c/10 años hasta los 55 Después c/5	0,5	IM	Deltoides	Escuela
OPV/ Poliomielitis	En campaña anual. Niños menores de 3 años y 9 años.				2 gotas	Oral	-----	Policlínicos y Consultorios

Epidemiología

Según Elkin (Alarcón, 2009) la Epidemiología es la ciencia que estudia las causas de la aparición, propagación, mantenimiento y descenso de los problemas de salud en poblaciones, con la finalidad de prevenirlos o controlarlos. La Epidemiología (López-Moreno, Garrido-Latorre y Hernández-Ávila, 2000) es la rama de la salud pública que tiene como propósito describir y explicar la dinámica de la salud poblacional; investiga la distribución, frecuencia y determinantes de las condiciones de salud en las poblaciones humanas.

Consulta Externa

La Consulta Externa son los actos médicos realizados de forma ambulatoria en un local adaptado para el diagnóstico, tratamiento o seguimiento de un paciente con base en la anamnesis (historia clínica) y la exploración física. Es un servicio muy importante dentro de cualquier hospital porque no sólo sirve de enlace entre los servicios de Emergencias y

Hospitalización, sino que también permite el seguimiento de pacientes hospitalizados («HOSPITAL JUAN CANALEJO» 2017).

El módulo de Consulta Externa del sistema XAVIA-HIS gestiona la información de los servicios que se brindan en el área del hospital con el mismo nombre, dichos servicios son: la planificación de horarios, los reportes diarios y principalmente los datos generados en las consultas a través de la hoja general de consulta. Es el área donde se brinda atención y posibilita el ingreso de grandes cantidades de pacientes. (Ing. Pedro Ernesto Salas Oliva, Dr.C . Arturo Orellana García, Ing. Lissette Soto Pelegrín, 2018)

Emergencia

Una emergencia es una atención de forma urgente y totalmente imprevista, ya sea por causa de accidente o suceso inesperado. En el sistema XAVIA-HIS, es el módulo que permite realizar la recepción del paciente en el área y su atención en observación, para lograr que haya un mayor control dentro del hospital y una atención eficiente al paciente. (Ing. Pedro Ernesto Salas Oliva, Dr.C . Arturo Orellana García, Ing. Lissette Soto Pelegrín, 2018)

Hospitalización

El principal objetivo de los profesionales que forman el equipo multidisciplinario de la unidad de Hospitalización es realizar un diagnóstico correcto y proporcionar el tratamiento y los cuidados asistenciales adecuados a la situación clínica del paciente ingresado hasta el momento del alta, buscando satisfacer tanto sus expectativas como las de sus familiares y acompañantes utilizando la mejor evidencia científica y proporcionando la mejor información para fomentar su auto cuidado una vez abandone el hospital. («Unidad de Hospitalización - CO.SA.GA.» 2012)

En el módulo de Hospitalización del sistema XAVIA-HIS se registran los datos asociados a: registro de la atención del paciente (Hojas de hospitalización), control y seguimiento al paciente hospitalizado mediante las evoluciones, solicitud de estudios y/o exámenes, indicación de órdenes médicas, transferencia de camas y servicios y egreso hospitalario (Ing. Pedro Ernesto Salas Oliva, Dr.C . Arturo Orellana García, Ing. Lissette Soto Pelegrín, 2018)

1.2- Proceso de vacunación en las áreas de salud cubanas.

En el proceso de vacunación: los preparativos, la propia vacunación y los cuidados posteriores que trae consigo son elementos claves para asegurar la eficacia y la eficiencia de las vacunas.

Dicho proceso inicia cuando el paciente acude al área de Enfermería con su tutor legal o solo en caso de ser mayor de edad, con una solicitud de vacunación y/o tarjeta de vacunación para ser vacunado. La enfermera clasifica al paciente para saber qué pasos debe seguir durante el proceso de vacunación. Si el paciente viene del Programa Materno Infantil (PAMI) consulta el esquema de vacunación para saber que vacuna le corresponde mientras que, si el paciente viene de áreas como Consulta Externa, Emergencia u Hospitalización, recibe la solicitud enviada por el médico especialista y verifica si la información que presenta el documento coincide con la vacuna que le corresponde por el esquema de vacunación.

En ambos casos la enfermera registra el nombre de la vacuna, la dosis que corresponde, fecha en que se aplica y el nombre del vacunador en el registro médico y en la tarjeta de vacunación de venir con una. Una vez registrada la información necesaria la enfermera aplica la vacuna y el paciente debe esperar una hora en la Enfermería para descartar una posible reacción adversa al medicamento, si esto ocurre la enfermera debe notificar de los síntomas que presenta el paciente para que este pueda ser atendido por el médico especialista de Emergencia, en caso de no existir reacción el proceso termina. En cualquiera de los dos escenarios el paciente debe recibir una nueva cita para vacunarse siempre y cuando la vacuna lleve una nueva dosis.

1.3- Sistemas existentes vinculados al campo de acción

Como parte de la investigación se llevó a cabo un estudio de los principales sistemas informáticos que gestionan la información y que, a su vez, brindan un conjunto de funcionalidades relacionadas con los esquemas de vacunación para luego poder incorporarlos a la propuesta de solución siempre y cuando se ajusten a las necesidades del sistema de salud nacional.

IMuni

Es un software panameño de asistencia médica que permite llevar el control de las citas de las vacunas de la población en los dispositivos móviles. Entre sus funcionalidades cabe destacar el registro de todas las vacunas del usuario y la notificación automática de las próximas vacunas e información sobre salud. Además, brinda geolocalización de los centros de salud cercanos a la casa del usuario y permite agregar las vacunas que se encuentran fuera del esquema de vacunación. Está disponible para sistemas IOS, en inglés, totalmente gratis y su última actualización es la v1.4.4 («Lanzan app IMuni para el control y seguimiento de vacunas - Secretaría Nacional de Ciencia, Tecnología e Innovación» 2017)

APP de la OMS/Europa

La OMS/Europa ha desarrollado una aplicación que permite avisar a los padres cuándo les corresponde una vacuna a sus hijos de manera que éstos lleven sus vacunas al día. La aplicación es válida para los sistemas operativos iOS (Mac) o para Android de los Smartphone. Es una herramienta barata que proporciona vínculos con los calendarios del país. («APP de la OMS para vacunas | Vacunas / Asociación Española de Vacunología» 2019)

NOMIVAC

El Registro Federal de Vacunación Nominalizado (NOMIVAC) gestiona desde el Sistema de Integrado de Información Sanitaria Argentino (SISA) las coberturas de vacunación de todo el país. La gestión del módulo permite registrar cada aplicación de una vacuna que recibe una persona, detallando la fecha, dosis, lote y establecimiento, entre otros datos relevantes. Este modelo de gestión informatizada a través de un registro nominalizado brinda a todos los niveles sanitarios la posibilidad de contar con datos actualizados, consistentes y confiables. Por otra parte, ofrece al ciudadano el acceso a su ficha, donde puede consultar su propio historial de vacunación al día. («Ayuda en línea SISA - Sistema Integrado de Información Sanitaria Argentino» 2013)

VACGAL

Es una nueva aplicación gratuita del Ministerio de Salud Gallego. Está diseñado para su uso en teléfonos inteligentes y tabletas con sistemas operativos Android y iOS. Permite a los padres consultar en cualquier momento el calendario de sus hijos o de sus mayores, si

son dependientes, e incluso fijar la cita para la próxima dosis. VACGAL ofrece recordatorios sobre vacunas pendientes y próximas a administrar. Además, la App ofrece la posibilidad de ver la correspondencia entre las vacunas administradas y el calendario de vacunas que por edad le corresponde a cada persona, y que puede cambiar de un año a otro. («VACGAL - A app do Programa Galego de Vacinación - Consellería de Sanidade - Servizo Galego de Saúde» 2018)

XAVIA-HIS 2.1

Sistema integral para la gestión hospitalaria del gobierno cubano. Tiene como atributo fundamental una HCE única por paciente y centralizada, que incluye toda la documentación, imágenes e información que se generan dentro del mismo. Aparece totalmente digitalizada, centralizada y almacenada con seguridad cumpliendo los principios de ética médica. Es un sistema que se comercializa, de código cerrado, que tiene como principales beneficios para el cliente: la gestión de la información de los procesos por los que transita dentro del hospital interrelacionando las diferentes áreas intrahospitalarias. Tiene en consideración elementos como la seguridad, la homogeneidad y estandarización de la información para un mayor control y una estadística más ágil y eficiente en la obtención de casos de estudios médicos (Universidad de las Ciencias Informáticas, 2019).

El módulo de Enfermería dentro del sistema XAVIA-HIS permite gestionar las vacunas, pero no de la forma más eficiente debido a que presenta solamente dos paneles; el primero muestra las vacunas aplicadas, especificando solamente la cantidad de dosis que presenta, dicha dosis se activa cuando se halla aplicado la anterior, permitiendo registrar la fecha en que el paciente fue vacunado. El segundo panel muestra un listado incompleto de vacunas existentes que han sido introducidas directamente en la base de datos, sin un adecuado sistema de gestión.

1.3.1- Análisis de los sistemas identificados

Después de analizar los sistemas homólogos vinculados al campo de acción se puede corroborar, que a pesar de existir hoy en día varias aplicaciones que visualicen esquemas de vacunación solo NOMIVAC gestiona el esquema y refleja los parámetros a tener en cuenta una vez aplicada la vacuna, detallando la fecha, dosis, lote y establecimiento, entre otros datos relevantes. En cuanto a los restantes criterios de comparación podemos

observar que solo la aplicación IMuni permite agregar las vacunas que se encuentran fuera del esquema de vacunación. La mayoría de las aplicaciones son de código cerrado, siendo 2 de las 5 aplicaciones web y 3 de pago. En la tabla 2, se puede ver un resumen del análisis realizado.

Tabla 2: Análisis de los sistemas que gestionan esquemas de vacunación. (Elaboración propia)

Sistemas	IMuni	APP de la OMS/Europa	NOMIVAC	VACGAL	XAVIA- HIS
Criterios de comparación					
Gestión de esquemas de vacunación			X		
Vacunas fuera del esquema de vacunación	X				
Aplicación web			X		X
Código cerrado	X		X	X	X
De pago		X	X		X

A pesar de existir un sistema que gestione las funcionalidades asociadas a los esquemas de vacunación no es posible su integración con el sistema XAVIA-HIS pues supone un gran impacto en la solución a desarrollar. Además, implicaría que una institución de salud tuviera más de un sistema de gestión hospitalaria lo que trae consigo un aumento de los

recursos humanos y de los costos de implementación. También es válido destacar que se necesitaría pagar una licencia para utilizar el software, ya que sería un proceso engorroso para el centro, sostener y evolucionar un producto de terceras personas.

Por lo anteriormente planteado se concluye que es necesario desarrollar las funcionalidades relacionadas con los esquemas de vacunación en el sistema XAVIA-HIS teniendo en cuenta aspectos como la inclusión de vacunas que se encuentren fuera de los esquemas de vacunación que puedan existir en el país.

1.4- Metodología de desarrollo del software (AUP-UCI)

CESIM, es un centro productivo de la Universidad de las Ciencias Informáticas (UCI), que se rige por el modelo de referencia de calidad CMMI-DEV versión 1.3, certificado en el nivel 2, que propone la aplicación de prácticas genéricas y específicas para la mejora continua de los procesos de una institución, y al no existir una metodología de software universal, ya que todas deben ser adaptadas a las características de cada proyecto, se decide hacer una variante de la metodología Proceso Unificado Ágil (AUP) de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la UCI. La misma tiene como objetivo aumentar la calidad del software que se produce.

Esta metodología consta de tres fases (Inicio, Ejecución y Cierre) y siete disciplinas (Modelado de negocio, Requisitos, Análisis y diseño, Implementación, Pruebas internas, Pruebas de liberación y Pruebas de aceptación). A partir de que la disciplina de Modelado de negocio propone tres variantes a utilizar en los proyectos: Casos de uso del negocio, Descripción del proceso de negocio y Modelado conceptual, existen tres formas de encapsular los requisitos: Casos de uso del sistema, Historias de usuario y Descripción de los requisitos de procesos. Además, propone 4 posibles escenarios en la etapa de desarrollo del software.

El escenario utilizado en el proyecto de desarrollo XAVIA-HIS es el 3 que plantea: proyectos que modelan el negocio con descripción de procesos de negocio (DPN) solo pueden modelar el sistema con descripción de requisitos de procesos (DRP). (Rodríguez, 2014)

1.5- Entorno de Desarrollo

El ambiente de desarrollo comprende todas las tecnologías, y herramientas utilizadas para la realización de la propuesta de solución. Son los programas que se utilizan para darle soporte al software haciendo en muchas ocasiones el trabajo más sencillo.

1.5.1- Lenguajes

Java

Es un lenguaje de programación robusto y multiplataforma. Una de sus principales características es la de ser un lenguaje interpretado y compilado. En el desarrollo de la propuesta de solución se utiliza la versión 1.6. (Fernández, 2005)

XHTML 1.0

El XHTML (eXtensible Hypertext Markup Language) o Lenguaje de Etiquetado Hipertextual Extensible es una reformulación del lenguaje HTML como aplicación XML que se recoge en la Recomendación del World Wide Web Consortium (W3C). Surgió ante los problemas de compatibilidad al usar un documento HTML en distintas plataformas. La especificación XHTML viene a ser una reformulación del HTML como aplicación XML. Su finalidad es que pueda ser usado como lenguaje de contenidos que sea a su vez conforme a XML. Los documentos XHTML son fácilmente visualizados, editados y validados con herramientas XML estándar.(Lamarca, 2018)

JavaScript

JavaScript un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. (Rugal Juan, 2013)

CSS 2

CSS (Cascading Style Sheets, por sus siglas en inglés) es un lenguaje de hojas de estilos en cascada creado para controlar la presentación de los documentos electrónicos definidos con XHTML. Es la mejor forma de separar los contenidos y su presentación y es imprescindible para la creación de páginas web complejas. Su presentación y la separación de los contenidos presentan numerosas ventajas, ya que obliga a crear documentos XHTML bien definidos y con significado completo. Además, mejora la accesibilidad del documento y reduce la complejidad de su mantenimiento. (Eguíluz Pérez, 2008)

UML 2.1

UML (Unified Modeling Language) o Lenguaje Unificado de Modelado en español, es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido concebido por los autores de los tres métodos más usados de orientación a objetos. Hay que tener en cuenta que el estándar UML no define un proceso de desarrollo específico, tan solo se trata de una notación. (Grau y Segura, 2008)

BPMN 2.0.2

Business Process Model and Notation (BPMN), en español Modelo y Notación de Procesos de Negocio, es una notación gráfica estandarizada que permite el modelado de procesos de negocio, en un formato de flujo de trabajo. BPMN tiene la finalidad de servir como lenguaje común para cerrar la brecha de comunicación que frecuentemente se presenta entre el diseño de los procesos de negocio y su implementación, esto facilitará una mejor comprensión de lo que se realiza. Está planeada para dar soporte únicamente a aquellos procesos que sean aplicables a procesos de negocios. (Briol, 2008)

1.5.2- Tecnologías a utilizar

Java Server Faces (JSF) 1.2

Java Server Faces (JSF) es el marco de trabajo de la interfaz de usuario (UI) orientada a componentes estándar para la plataforma Java EE. JSF incluye un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar

eventos, validar entradas, definir un esquema de navegación de las páginas y dar soporte para la internacionalización y accesibilidad. («JavaServer Faces.org» 2018).

Ajax4JSF

AJAX (Asynchronous JavaScript And XML, por sus siglas en inglés) no es en sí misma una tecnología, se trata de varias tecnologías independientes que se unen de formas sorprendentes. Las tecnologías que forman AJAX son: XHTML y CSS, para crear una aplicación basada en estándares; DOM (Dinamic Object Model), para la interacción y manipulación dinámica de la presentación; XML, para el intercambio y manipulación de información; XMLHttpRequest, para el intercambio asíncrono de información y JavaScript, para unir las demás tecnologías. La característica fundamental de AJAX es que permite actualizar parte de una página con información que se encuentra en el servidor sin tener que refrescar completamente la página. Entre las ventajas más significativas con las que cuenta están que es soportado por la mayoría de los navegadores modernos y presenta mayor velocidad debido a que no hay que retornar toda la página nuevamente. («Dynamic AJAX» 2005)

Jboss Server 4.2.2

JBoss Application Server es un servidor de aplicaciones Java de código abierto y multiplataforma desarrollado por JBoss, una división de Red Hat Inc. JBoss AS es una implementación de código abierto de Java 2 Enterprise Edition (J2EE) que se usa para desarrollar aplicaciones Java y otras aplicaciones y software basados en la Web. Corre sobre una máquina virtual de Java y es soportado por varios sistemas operativos. («What is Jboss Application Server?» 2018)

Jboss Seam 2.1.1 GA

Seam es una potente plataforma de desarrollo de código abierto para crear aplicaciones en Java. Permite a los desarrolladores ensamblar aplicaciones web complejas utilizando clases Java anotadas simples, un conjunto rico de componentes de UI y muy poco XML. (Seam, 2009)

Hibernate 3.0

Hibernate es una herramienta de mapeo objeto-relacional para el lenguaje de programación Java. Como un marco de trabajo de mapeo de objetos relacionales (ORM), Hibernate se encarga de la persistencia de los datos tal como se aplica a las bases de datos relacionales. Además, proporciona un potente lenguaje de consultas denominado Hibernate Query Language (HQL). (Hibernate, 2018)

RichFaces 3.3.0 GA

Richfaces, versión 3.3.0.GA, es un marco de trabajo de código abierto que añade a las aplicaciones capacidad Ajax en JSF sin recurrir a JavaScript. Richfaces aprovecha el marco de trabajo JSF, incluyendo su ciclo de vida, la validación, los medios de conversión y la gestión de los recursos estáticos y dinámicos. Los componentes de Richfaces con soporte Ajax y aspecto altamente personalizable puede ser fácilmente incorporado a aplicaciones JSF. Permite intensificar el conjunto de los beneficios de JSF al trabajar con Ajax, añadir la capacidad de Ajax a aplicaciones JSF, crear rápidamente una vista compleja basándose en sus componentes y escribir sus propios componentes con función de soporte Ajax. Permite además crear una moderna interfaz de usuario rica en vista y sensación basado en esta tecnología y probar y crear los componentes, las acciones, los escuchadores, y las páginas al mismo tiempo.(Filocamo, 2009)

Facelets 1.1.15. B1

Facelets es un poderoso, pero ligero lenguaje de declaración de páginas que es utilizado para construir vistas JSF usando plantillas de estilo HTML y para construir árboles de componentes. Hace uso de XHTML para la creación de páginas web, soporta librerías Facelets además de las que provee JSF. El uso de esta tecnología reduce el tiempo y esfuerzo que se necesita para el desarrollo. (Corporation, 2010)

1.5.3- Herramientas para el desarrollo

Jboss Developer Studio 8.1

JBoss Developer Studio, versión 8.1, es un conjunto de herramientas de desarrollo basadas en Eclipse. Permite desarrollar, probar e implementar aplicaciones web avanzadas, aplicaciones web móviles, aplicaciones empresariales transaccionales y aplicaciones y servicios de integración basados en la arquitectura orientada a servicios.

JBoss Developer Studio incluye un amplio conjunto de herramientas y soporte para varios modelos y marcos de programación. (Woguia, 2017)

PgAdminIII

PgAdmin 3 es una herramienta de código abierto para la administración de bases de datos PostgreSQL. Se diseña para responder a las necesidades de la mayoría de los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y hace simple la administración. Está disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo Microsoft Windows, Linux, FreeBSD, Mac OSX y Solaris. Soporta versiones de servidores 7.3 y superiores.(pgadmin.org, 2018)

PostgreSQL 9.4.1-1

PostgreSQL es un Sistema Gestor de Bases de Datos relacional orientado a objetos basado en la versión 4.2 de Postgres, desarrollado en la Universidad de California en el Departamento de Ciencias de la Computación de Berkeley. Es una herramienta de código abierto que soporta una gran parte del estándar SQL y ofrece nuevas características tales como consultas complejas, llaves foráneas, triggers y otras. Además, puede ser extendido por el usuario de varias maneras, por ejemplo, añadiéndole tipos de datos, funciones agregadas y métodos de indexación. Producto a su licencia liberada puede ser usado, modificado y distribuido por cualquiera y para cualquier propósito, ya sea privado, comercial o académico. (The PostgreSQL Global Development Group, 2018)

Visual Paradigm 8.0

Visual Paradigm es una herramienta poderosa que facilita el diseño de procesos. Provee a los desarrolladores de software de una plataforma para desarrollar aplicaciones de la calidad de manera rápida y fácil. Facilita una excelente interoperabilidad con otras herramientas CASE (Computer Aided Software Engineering por sus siglas en inglés). Permite el diseño de digramas de clases, casos de uso, secuencia, comunicación, actividad y demás artefactos de la Ingeniería de software, necesarios para la implementación de una solución correcta.(Visual Paradigm, 2019)

Axure RP Pro 7.0

Axure es uno de los estándares de la industria de software orientada a diseñar wireframes (esquema de páginas) y generar prototipos básicos o avanzados de forma fácil. Esta herramienta está dirigida tanto a la creación de aplicaciones web como de escritorio. Una de sus características más interesantes es que genera prototipos HTML de sitios web sin necesidad de ningún tipo de codificación, así como documentos de Microsoft Word para los procesos de documentación. Axure demuestra un importante grado de especialización, permite especificar el estado de cada elemento (Propuesto, Aceptado, Incorporado), el beneficio esperado (Crítico, Importante, Útil), el riesgo, la estabilidad, a quién va dirigido y a quién se le asignará la tarea. («Herramientas de diseño» AXURE, Diseño de wireframes y Prototipos» 2020)

1.6- Conclusiones Parciales

En el presente acápite luego de haber analizado minuciosamente los conceptos: Sistemas de Información de Salud, Epidemiología, enfermedad, vacunas y esquemas de vacunación; se logró comprender la interrelación que existen entre dichos conceptos y la manera en que fluye la información a través de ellos.

La investigación de los sistemas homólogos permitió la mejor comprensión de la especialidad y constituyó un punto de referencia para el desarrollo de las funcionalidades de los esquemas de vacunación para la especialidad de Epidemiología del sistema XAVIA-HIS.

El estudio de las herramientas, tecnologías, metodologías y lenguajes definidos en el Centro de Informática Médica proporcionó la posibilidad de desarrollar los esquemas de vacunación bajo los mismos estándares que posee.

Capítulo 2: Análisis y diseño de la propuesta de solución

Después de seleccionar el escenario 3 de la metodología AUP-UCI que plantea el modelado de negocio con descripción de procesos de negocios (DPN) y el modelado del sistema con descripción de requisitos de procesos (DRP) se pretende en el presente capítulo describir la propuesta de solución a partir de la especificación de los requisitos funcionales y la modelación de los artefactos ingenieriles correspondientes a las etapas de desarrollo de software de análisis y diseño. Se modelan los diagramas de clases del diseño y el diagrama de paquete basándose en el patrón arquitectónico Modelo- Vista- Controlador. Además, se describen los patrones de diseño aplicados en la implementación de las clases y se realiza el modelo de datos referente a la propuesta de solución especificando los atributos comunes y su correspondiente descripción.

2.1- Descripción de la propuesta de solución

Como resultado de la investigación realizada en el capítulo anterior y con el objetivo de adicionar los esquemas de vacunación al sistema XAVIA-HIS, se propone la implementación de funcionalidades que permitan la configuración de dichos esquemas en el módulo de Epidemiología lo que permitirá una mejor gestión de la información de las vacunas y los esquemas de vacunación, para un mayor control en el proceso de vacunación y asignación de nuevas citas para la aplicación de próximas vacunas.

Desde el módulo de Epidemiología el médico especialista podrá crear nuevas vacunas ajustando cada uno de sus parámetros, modificarlas o eliminarlas en caso de no estar asociadas a un esquema de vacunación. En cuanto a los esquemas podrá crearlos, modificarlos o eliminarlos, asociarles una o varias vacunas y ver sus datos para saber si quedó como se esperaba. Otorgará los permisos necesarios para que las enfermeras y los médicos especialistas de las áreas de Emergencia, Hospitalización y Consulta Externa puedan visualizar los esquemas de vacunación para poder asignar citas para la aplicación de vacunas, gestionar el proceso de vacunación y tener un control riguroso del carnet de vacunación.

2.2- Modelo conceptual

Un modelo conceptual es un artefacto útil que permite identificar de forma abstracta los conceptos y hechos relevantes del dominio del problema y transformarlos posteriormente en un esquema de una base de datos concreta. (Sevilla, 2012).

Con el objetivo de contribuir a un correcto levantamiento de requisitos se actualiza el modelo conceptual del módulo de Enfermería que se encuentra en el expediente del proyecto HIS_Producto en el documento CESIM_PRODUCTO_Modelo_Conceptual_ENF.

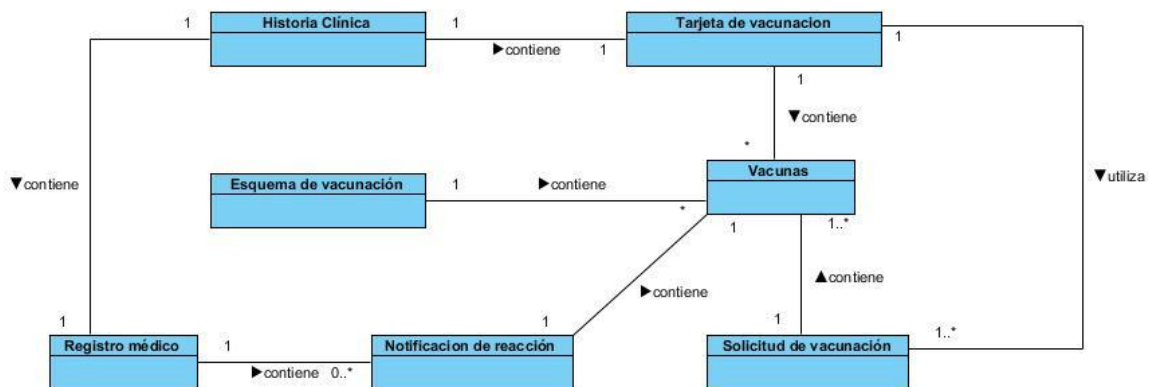


Ilustración 1: Modelo conceptual. (Elaboración propia)

2.3- Modelo de negocio

Un modelo de proceso del negocio describe cómo funciona el negocio, es decir, describe las actividades involucradas en el negocio y la manera en que estas se relacionan.

En la Ilustración 2 se muestra el modelo del proceso de negocio “Ejecutar programa de vacunación” definido para el área de Enfermería. Este proceso tiene su inicio en la creación del esquema de vacunación desde el Departamento Nacional de Epidemiología, institución que luego de crearlo, lo distribuye por todos los departamentos de Epidemiología del país y que a su vez estos deben hacérselo llegar a las enfermeras para que puedan ejecutar el proceso de vacunación (descrito en el epígrafe 1.2).

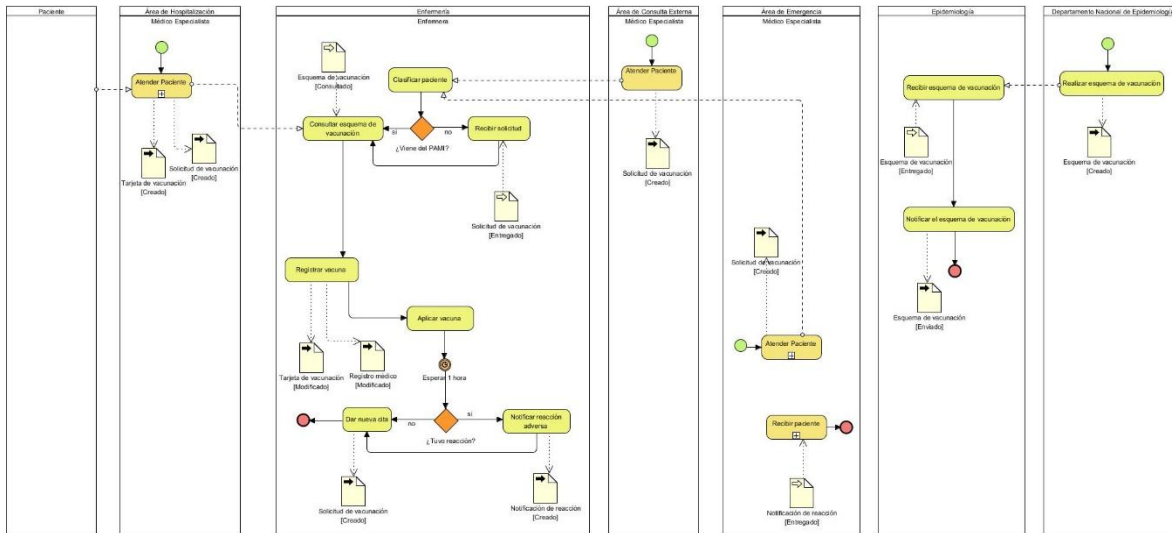


Ilustración 2: Diagrama de proceso de negocio "Ejecutar programa de vacunación". (Elaboración propia)

2.4- Requisitos de software

Los requisitos del software definen las funciones, capacidades o atributos de cualquier sistema de software. Estos se pueden dividir en dos categorías: funcionales y no funcionales. Los requisitos funcionales son los que describen las funciones que el sistema será capaz de realizar mientras que los requisitos no funcionales tienen que ver con las características que de una u otra forma pueden limitar el sistema. (Ian Sommerville, 2012)

2.4.1- Requisitos funcionales

En la tabla 3 se muestran los requisitos funcionales que poseerá el sistema acorde a la propuesta de solución que se desarrollará. Para la descripción y la especificación de los requisitos del software de la propuesta de solución se actualizaron los documentos CESIM_PRODUCTO_Descripción_de_requisitos_por_proceso_Epidemiología y CESIM_PRODUCTO_Especificación_de_requisitos_de_software_Epidemiología.

Posteriormente se describirá en la tabla 4 el requisito funcional "Crear vacuna". Para obtener la descripción del requisito "Crear esquema de vacunación" (Ver Anexo 1). Mientras que el resto de los requisitos se encuentran en la documentación antes mencionada.

Tabla 3: Requisitos funcionales. (Elaboración propia)

Nº	Nombre
RF1	Crear vacuna
RF2	Ver detalles de la vacuna
RF3	Buscar vacuna
RF4	Ver datos de la vacuna
RF5	Eliminar vacuna
RF6	Modificar vacuna
RF7	Crear período de reactivación
RF8	Crear esquema de vacunación
RF9	Ver detalles del esquema de vacunación
RF10	Buscar esquema de vacunación
RF11	Ver datos del esquema de vacunación
RF12	Modificar esquema de vacunación
RF13	Eliminar esquema de vacunación

Tabla 4: Especificación del RF1 "Crear vacuna". (Elaboración propia)

Descripción textual	El requisito inicia cuando el actor accede a la opción Crear vacuna y el sistema brinda la posibilidad de introducir los datos, el actor introduce los datos de la vacuna, el sistema adiciona la vacuna y el requisito termina.
Actores	Epidemiólogo
Precondiciones	La región de aplicación debe estar configurada
Flujo de eventos	
Flujo básico "Crear vacuna"	
1.	El requisito inicia cuando el actor accede a la opción Crear vacuna.
2.	<p>El sistema brinda la posibilidad de introducir los datos:</p> <ul style="list-style-type: none"> • Nombre • Cantidad de dosis <p>Brinda la posibilidad de seleccionar los datos:</p> <ul style="list-style-type: none"> • Vía • Reactivación. Ver Flujo alternativo 1: " Crear período de reactivación" • Enfermedad que previene. Se ejecuta el requisito, ver requisito Elementos Comunes: Seleccionar enfermedad

	<ul style="list-style-type: none"> Región de aplicación. Ver Flujo alternativo 2: “Seleccionar región de aplicación” <p>y permite:</p> <ul style="list-style-type: none"> Aceptar. Cancelar operación. Ver Flujo alternativo 3: “Cancelar operación.”
3.	El actor introduce los datos deseados.
4.	El actor selecciona la opción Aceptar.
5.	<p>El sistema valida los datos.</p> <p>Si hay datos incompletos. Ver Flujo Alternativo 4: “Existen datos incompletos.”</p> <p>Si hay datos incorrectos. Ver Flujo Alternativo 5: “Existen datos incorrectos.”</p>
6.	<p>El sistema adiciona los datos de la vacuna y muestra los datos de la misma.</p> <p>Se ejecuta el requisito, ver requisito Ver detalles de la vacuna.</p>
7.	El requisito termina.
Secciones	
N/A	
Flujos alternativos	
Flujo alternativo 1 " Crear período de reactivación"	

1.	El actor selecciona la opción Adicionar período de reactivación. Se ejecuta el requisito, ver requisito Crear período de reactivación.
2.	El sistema regresa al paso 2 del Flujo básico .
Flujo alternativo 2 “Seleccionar región de aplicación”	
1.	El actor selecciona la opción Seleccionar región de aplicación
2.	Regresa al paso 2 del Flujo básico .
Flujo alternativo 3 “Cancelar operación”	
1.	El actor selecciona la opción Cancelar operación.
2.	El sistema regresa a la interfaz anterior.
3.	Regresa al paso 7 del Flujo básico .
Flujo alternativo 4 “Existen datos incompletos”	
1.	El sistema muestra un indicador sobre los campos nombre, vía, región de aplicación y código.
2.	Regresa al paso 3 del Flujo básico .
Flujo alternativo 5 “Existen datos incorrectos”	
1.	El sistema muestra un indicador sobre los campos incorrectos cantidad de dosis y código.

2.	Regresa al paso 3 del Flujo básico .	
Pos-condiciones		
1.	Se creó la vacuna.	
Validaciones		
1.	Vacunas	
Conceptos	Vacunas	Modelo conceptual Enfermería
	N/A	Modelo conceptual Elementos Comunes
Restricciones del sistema	N/A	
Dependencias	Obligatoria	Ver detalles de la vacuna Elementos Comunes:: Seleccionar enfermedad.
	Opcional	Crear período de reactivación
Requisitos especiales	N/A	
Asuntos pendientes	N/A	
Prototipo elemental de interfaz gráfica de usuario		

Q Buscar...

Crear vacuna

Nombre: Via:

Reactivación + Adicionar período de reactivación

Listado de reactivación

	Tiempo	Edad de inicio	Edad de fin
<input type="checkbox"/>	10 años	15 años	55 años
<input type="checkbox"/>	5 años	55 años	120 años
<input type="checkbox"/>	3 años	10 años	16 años

⏪ ⏩ ⏴ ⏵

Enfermedad que previene

Código: Descripción: Estructura:

Buscar

🔍 Búsqueda simple

Clasificación Internacional de Enfermedades (CIE)

	Código	Descripción	Estructura
<input checked="" type="checkbox"/>	(B15- B19)	Hepatitis viral	Grupo
<input type="checkbox"/>	B15	Hepatitis aguda tipo A	Categoría

⏪ ⏩ ⏴ ⏵

Diagnóstico

Código	Descripción	
(B15- B19)	Hepatitis viral	🗑

Región de aplicación

	Nombre
<input checked="" type="checkbox"/>	Deltoide izquierdo
<input type="checkbox"/>	Deltoide derecho

⏪ ⏩ ⏴ ⏵

Región de aplicación seleccionada

Descripción	
Deltoide izquierdo	🗑

Aceptar Cancelar

Q Buscar...

Crear período de reactivación

Período de reactivación

Tiempo: Años Edad de inicio: Años Edad de fin: Años

Aceptar Cancelar

Formatos de entrada/salida

N/A
Entradas
N/A
Salidas
N/A

2.4.2- Requisitos no funcionales

Los requisitos no funcionales se adoptan a partir de los definidos para el sistema XAVIA-HIS en el documento CESIM_PRODUCTO_Especificación_de_requisitos_de_software_EC.doc. Algunos de ellos son:

RNF1- Requisito de Seguridad:

El sistema permite el acceso del usuario al sistema, mostrando las entidades a las cuales tiene permiso. De cada entidad se mostrarán bloqueados los módulos a los cuales no tiene acceso. Dentro de cada módulo a que se tiene acceso sólo se mostrarán aquellas funcionalidades configuradas como permitidas para el usuario. Dentro de las funcionalidades permitidas, se limitará a los permisos que tenga asignado el usuario en cuanto a departamentos, servicios o especialidades. El sistema notifica cuando no se tienen permisos para el recurso solicitado.

El sistema registra los datos de la acción realizada en la bitácora (usuario que realizó la transacción, tipo de operación que realizó, fecha y hora e información contenida en el registro modificado). La bitácora muestra todas las trazas de las acciones de los usuarios en el sistema.

RNF2- Requisito de Usabilidad:

El sistema notifica al usuario la existencia de datos incorrectos o incompletos cuando

accede a funcionalidades crear o modificar. Para las funcionalidades eliminar, el sistema muestra mensajes de confirmación.

RNF3- Requisito de Eficiencia:

Toda funcionalidad del sistema y transacción de negocio debe responder al usuario en menos de 5 segundos.

El sistema se encuentra disponible y se puede navegar por él. En caso que no se cumplan los siguientes requerimientos el sistema no funcionará correctamente.

Características de Software:

- **PC Cliente:** Computadora con cualquier sistema operativo, Mozilla Firefox 3.x o superior y que cuente con el plugin de Flash Player 10.x o superior.
- **Servidor de Aplicaciones:** Servidor con cualquier sistema operativo que soporte el Java Runtime Environment (JRE) 1.6.0_₋₂₄ o superior y al JBoss AS 4.2.2.GA
- **Servidor de Base de Datos:** Servidor con cualquier sistema operativo que soporte a PostgreSQL Server 8.4 o superior en los servidores de base de datos de cada hospital, y Oracle 10g o superior para los servidores de base de datos del Centro de Datos.

Características del Hardware:

- **PC Cliente**

CPU: Intel Celeron 1.8 GHz

RAM: 512 Mb

Servidor de Aplicaciones

CPU: 8 núcleos 2.6 GHz

RAM: 16 Gb

HDD: 10 Gb

SAN: 40 Gb

- **Servidor de base de datos**

CPU: 4 núcleos 2.6 GHz

RAM: 8 Gb

HDD: 10 Gb

SAN: 25 Gb

RNF4- Requisito de Mantenibilidad:

El sistema funciona correctamente y los componentes y/o plantillas reutilizadas se encuentran acorde a las pautas de diseño definidas.

RNF5- Requisito de Portabilidad:

El sistema adapta sus componentes a la resolución de pantalla seleccionada por el usuario. El sistema funciona correctamente luego de ser instalado y configurado.

RNF6- Requisito de Apariencia o interfaz:

Interfaz de usuario: El sistema adapta sus interfaces según lo establecido en el documento: Pautas de diseño de interfaz de usuario definido por el CESIM.

RNF7- Requisito de Restricciones del diseño y la implementación:

- El análisis y diseño de la aplicación se hace bajo los principios del paradigma Orientado a Objetos.
- El lenguaje de programación a utilizar, en coherencia con el paradigma, es JAVA.
- La herramienta CASE utilizada para el análisis y diseño es Visual Paradigm.
- Se emplea además el siguiente estándar internacional:

CIE10: para diagnósticos médicos

2.5- Modelo de diseño

Los modelos de diseño muestran los objetos o clases de un sistema y los diferentes tipos de relaciones existentes entre estas entidades. Los modelos de diseño son el diseño mismo, son el puente entre los requisitos y la implementación del sistema. Tienen que ser abstractos con el fin de que el detalle innecesario no oculte las relaciones entre ellos y los requisitos del sistema. Pero también debe tener los detalles suficientes para que los programadores tomen las decisiones de implementación. (Sommerville, 2005)

2.5.1- Patrones arquitectónicos

Los patrones arquitectónicos se utilizan para expresar una estructura de organización base o esquema para un software. Proporcionando un conjunto de subsistemas predefinidos, especificando sus responsabilidades, reglas, directrices que determinan la organización, comunicación, interacción y relaciones entre ellos.

El patrón Modelo-Vista-Controlador (MVC), es uno de los patrones arquitectónicos que se utilizan en el sistema XAVIA-HIS. Surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos, a partir de estandarizar el diseño de las aplicaciones. El patrón MVC es un paradigma que divide en tres módulos bien definidos las partes que conforman una aplicación el Modelo, las Vistas y los Controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo(ver ilustración 3). A partir del uso de framework basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores. (González y Romero, 2012).

Debido a lo mencionado con anterioridad se propone utilizar este paradigma para el desarrollo de la propuesta de solución.

Modelo

El modelo desconoce la existencia de las vistas y el controlador.(Pantoja, 2004). Es un componente que se encarga de manipular, gestionar y actualizar los datos. Si se utiliza

una base de datos es aquí donde se realiza la búsqueda, consultas, filtros y actualizaciones. (Garcia, 2017)

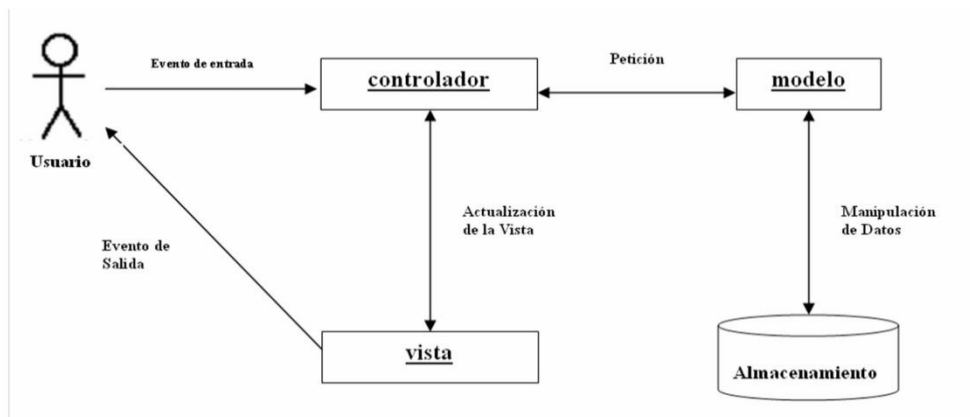
Vista

Las vistas son el conjunto de clases que se encargan de mostrar al usuario final la información contenida en el modelo. (Pantoja, 2004). Desde el punto de vista del programador este es el componente que se encarga del frontend; la programación de la interfaz de usuario si se trata de una aplicación de escritorio. (Garcia, 2017)

Controlador

El controlador es un objeto que se encarga de dirigir el flujo del control de la aplicación debido a mensajes externos, como datos introducidos por el usuario u opciones del menú seleccionadas por él. A partir de estos mensajes, el controlador se encarga de modificar el modelo o de abrir y cerrar vistas. El controlador tiene acceso al modelo y a las vistas, pero las vistas y el modelo no conocen de la existencia del controlador.(Pantoja, 2004)

Ilustración 3: Patrón arquitectónico Modelo- Vista- Controlador. (Pantoja, 2004)



2.5.2- Patrones de diseño

Los patrones de diseño surgen de la necesidad de analizar, compartir y documentar el conocimiento sobre ciertos tipos de problemas durante el diseño orientado a objeto. Dichos patrones también se aplican en otras áreas del desarrollo y se recomienda utilizarlos en cualquier momento para no partir de cero.

2.5.2.1- Patrones GRASP

Los Patrones de Software de Asignación de Responsabilidad General (por sus siglas en inglés GRASP) son una pareja de problema-solución que codifican buenos principios y sugerencias relacionadas frecuentemente con la asignación de responsabilidades. (Visconti y Astudillo, 2012). A continuación, se muestran los patrones utilizados en la propuesta de solución:

Bajo acoplamiento: Se basa en tener las clases lo menos ligadas entre sí, de tal forma que, en caso de producirse una modificación entre alguna de ellas, se tenga la mínima repercusión posible en el resto de las clases. Potencia la reutilización del código y disminuye la dependencia y se pone de manifiesto en todas las clases del sistema.

Alta cohesión: Se utiliza cuando una clase tiene una responsabilidad moderada en un área funcional y colabora con otras clases para llevar a cabo las tareas. La alta cohesión también se pone de manifiesto en todas las clases del sistema.

Experto: Determina cuál es la clase que debe asumir una responsabilidad a partir de la información que posee. O sea, la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De este modo se obtiene un diseño con mayor cohesión y así la información se mantiene encapsulada (bajo acoplamiento). Una buena práctica es que este patrón se aplique a todas las clases que conforman el sistema y un ejemplo de esto es la clase `CrearEsquemaVacunacionControlador_consulta`.

Creador: Ayuda a identificar quien debe ser el responsable de la creación de nuevos objetos o clases. Se pone de manifiesto en la clase `CrearVacunaControlador_consulta` al ser esta la clase que crea todas las instancias de los objetos que se manejan a la hora de crear la vacuna, para luego realizar operaciones con dichas instancias.

Controlador: Sirve como intermediario entre una determinada interfaz y el algoritmo que lo implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado. Sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto permite aumentar la reutilización del código y a la vez tener un mayor control. Este patrón se evidencia en la clase

CrearVacunaControlador_consulta pues es la clase encargada de instanciar todos los objetos de otras clases y además es la encargada de salvar los datos de la vacuna.

2.5.3- Diagrama de paquetes

El diagrama de paquetes es uno de los diagramas estructurales comprendidos en UML 2.5, representa de forma estática los componentes del sistema de información que está siendo modelado. Es utilizado para definir los distintos paquetes a nivel lógico que forman parte de la aplicación y la dependencia entre ellos. Este diagrama es utilizado en los sistemas de información con programación orientada a objeto por desarrolladores y analistas. (Teniente López E, Costal Costa D, 2013)

En la Ilustración 4 se puede apreciar que el diagrama está compuesto por dos grandes paquetes (Repositorio de clases y Ejecutar programa de vacunación), ambos tienen una relación de dependencia lo que indica que el paquete " Ejecutar programa de vacunación " utiliza la información contenida dentro del paquete " Repositorio de clases " .

Dentro del paquete Repositorio de clases se encuentran 3 subpaquetes donde se agrupan las entidades, las secciones y las vistas. En el subpaquete secciones se encuentran las clases controladoras agrupadas en un paquete que contiene las clases autogeneradas, otro con las clases personalizadas que se crean a partir de las clases autogeneradas y las clases controladoras propias del proceso.

El subpaquete de las entidades contiene las entidades autogeneradas desde la base de datos utilizando el Mapeo objeto relacional de Hibernate y las personalizadas que son las autogeneradas y modificadas para la gestión de la información.

Por último, está el paquete de Vistas que agrupa todas las vistas, las cuales se crean a partir de las peticiones realizadas a las clases controladoras, las que a su vez acceden a las entidades para persistir y obtener los datos necesarios que se tomen o carguen de las vistas.

Dentro del paquete Ejecutar programa de vacunación se encuentran 13 paquetes que hacen referencia a cada uno de los requisitos que serán implementados en el sistema.

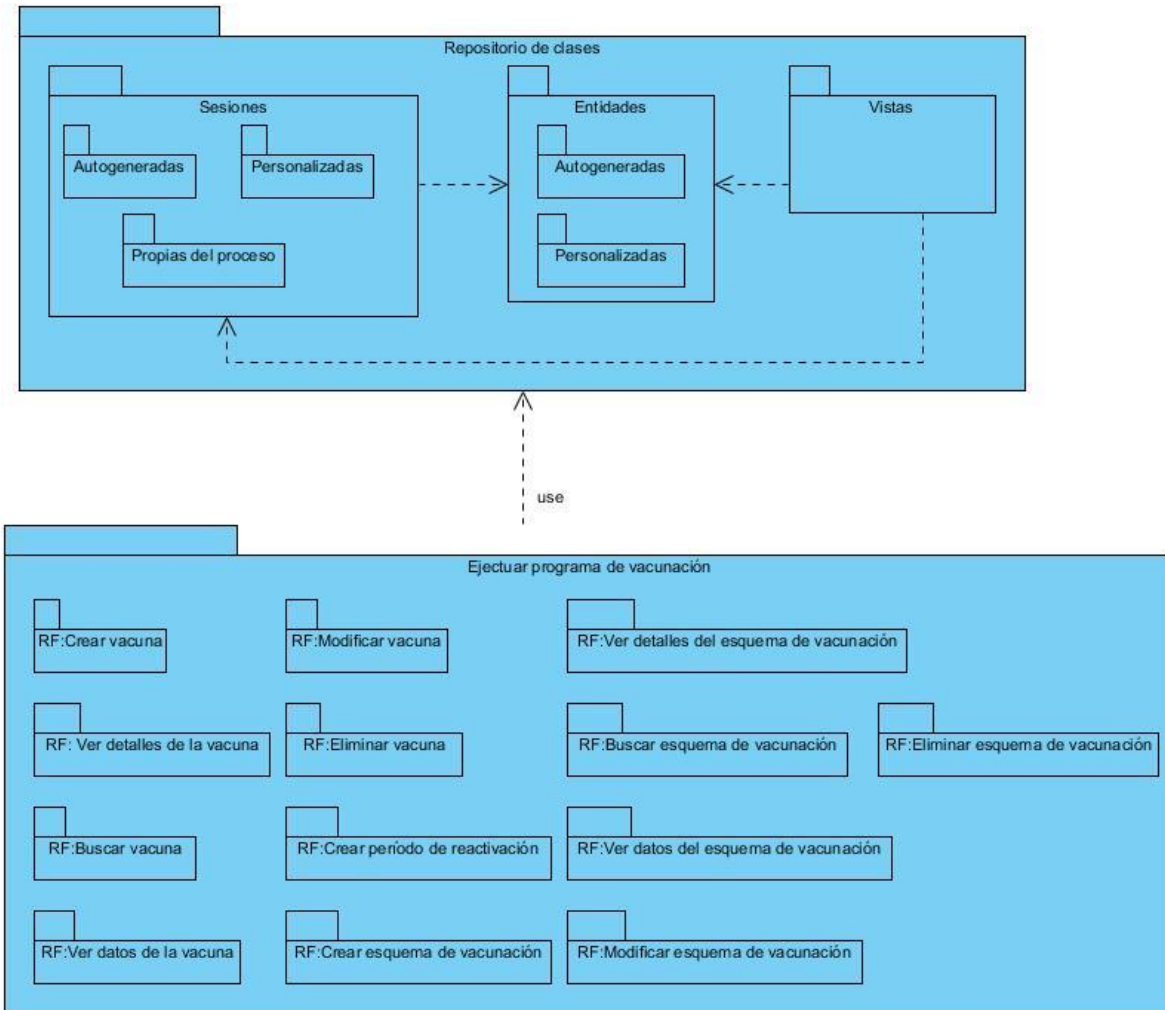


Ilustración 4: Diagramas de paquetes. (Elaboración propia)

2.5.4- Modelo de datos

Un modelo de datos es la vía para documentar un diseño de sistema de software complejo como un diagrama de fácil comprensión, utilizando texto y símbolos para representar la forma en que los datos necesitan fluir. El diagrama se puede utilizar como un mapa para la construcción de un nuevo software o para la reingeniería de una aplicación antigua. (Rouse, 2016)

A continuación se presenta el modelo de datos referente a los esquemas de vacunación a partir de las entidades implementadas donde se refleja la relación que existe entre las tablas de la base de datos y los atributos que la componen. (Ver Ilustración 5)

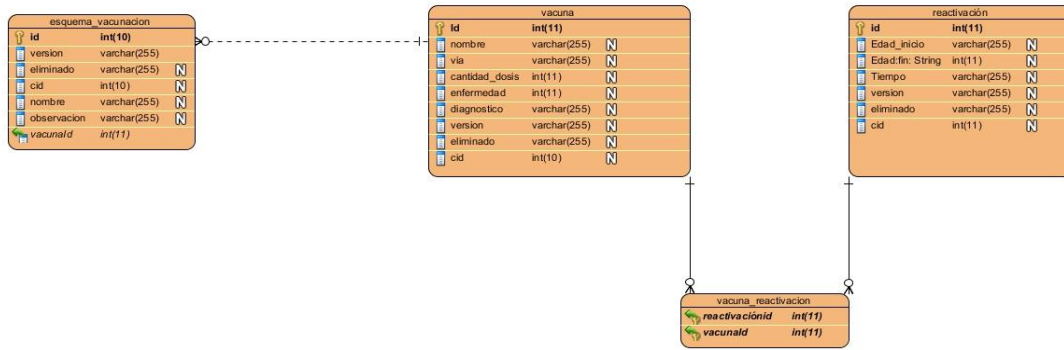


Ilustración 5: Modelo de datos " Esquema de vacunación". (Elaboración propia)

Para facilitar la implementación de las funcionalidades del sistema en la tabla 5, se describen los atributos que son comunes en todas las entidades.

Tabla 5: Descripción de los atributos comunes en las tablas de modelo de datos. (Elaboración propia)

Atributos	Tipo	Descripción
id	Integer	Identificador necesario en cada entidad para las referencias en las relaciones entre tablas.
Versión	Integer	Indica con que versión de la entidad se está trabajando. Es usado para garantizar que se está trabajando con la versión de la entidad más actualizada que existe en la base de datos.
código	Varchar	Permite identificar la entidad.
eliminado	Varchar	Permite la identificación lógica con que cuenta el sistema, cuando está en verdadero indica que la entidad está eliminada.
cid	Integer	Permite identificar quién realiza alguna acción sobre

		la entidad.
--	--	-------------

2.5.5- Diagrama de clases de diseño

Un diagrama de clases de diseño especifica las clases que contienen la propuesta de solución y las relaciones que existen entre ellas. Dicha relación se basa en las peticiones que se hacen y en el flujo de datos que se maneja entre ellas. Para modelar las clases de diseño se utiliza el patrón arquitectónico MVC.

La estructura general de los diagramas de clases de diseño del Esquema de vacunación, están compuestos por páginas clientes que son construidas por páginas servidoras y que a su vez contienen formularios que muestran y capturan información. Las páginas servidoras invocan métodos o responsabilidades en las clases controladoras que según la acción solicitada pueden modificar las entidades.

En la Ilustración 6 se muestra el diagrama de clases de diseño del requisito “Crear esquema de vacunación” que constituye la base para la implementación de la propuesta de solución.

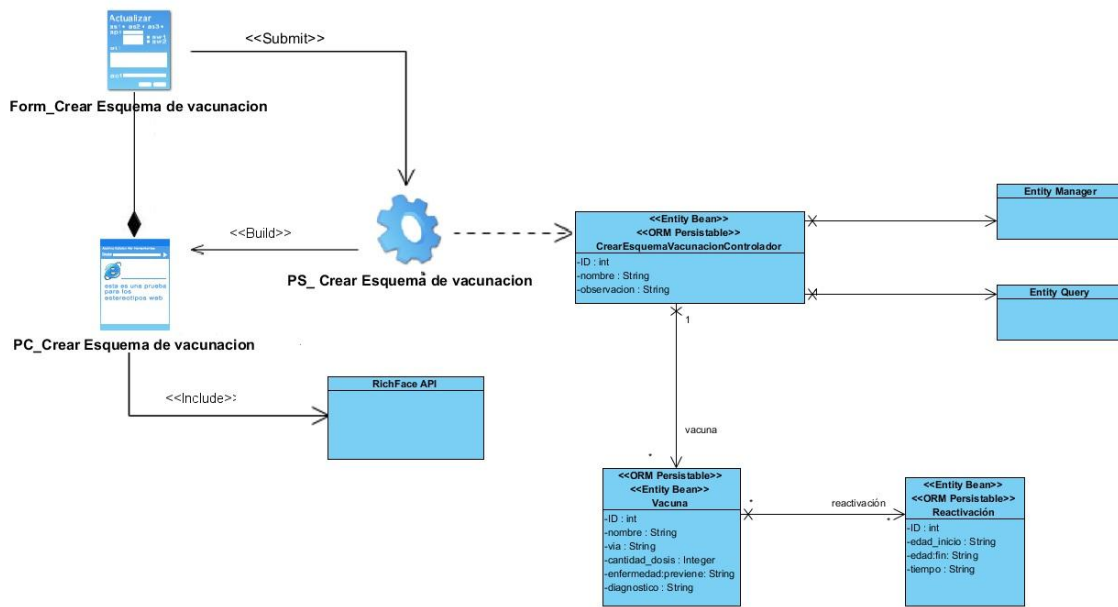


Ilustración 6: Diagrama de clases de diseño "Crear esquema de vacunación". (Elaboración propia)

2.6- Conclusiones Parciales

A través del estudio de la documentación sobre la metodología AUP-UCI se logró un mejor entendimiento de la gestión de la información lo que permitió identificar los requisitos funcionales para el desarrollo de los esquemas de vacunación.

El modelo de negocio permitió comprender el funcionamiento del proceso "Crear vacuna" para integrarlo a las funcionalidades de los esquemas de vacunación.

Se realizó la descripción de requisitos por procesos, permitiendo recoger la información para un correcto desarrollo de las funcionalidades.

Basado en el patrón arquitectónico MVC y los patrones GRASP, se modelaron los artefactos ingenieriles utilizados como punto de partida para la implementación de la propuesta de solución.

Capítulo 3: Validación de la propuesta de solución

En el capítulo se abordan los elementos asociados a la implementación de las clases y subsistemas de la propuesta de solución. Se incluyen los resultados de las pruebas realizadas para validar la correcta implementación de los requisitos y se realiza un estudio de los mecanismos para el tratamiento de errores. Además, la seguridad informática es abordada con la finalidad de prevenir acciones que afecten la disponibilidad, integridad y confidencialidad de la información contenida.

3.1- Modelo de implementación

La etapa de implementación del software es el proceso de convertir una especificación del sistema en un sistema ejecutable. Esta fase comprende la materialización, en forma de código, de todos los artefactos, descripciones y arquitectura propuestos en la etapa de análisis y diseño; con el objetivo de conformar el producto final requerido por el cliente.

Diagrama de componentes

Los diagramas de componentes son utilizados para mostrar los elementos del software y la relación que existe entre ellos en un sistema. Se utilizan para mostrar la dependencia de compilación de los ficheros de código, relaciones de derivación entre ficheros de código fuente y ficheros que son resultados de la compilación.

A continuación, se presenta el diagrama de componentes del RF8 "Crear esquema de vacunación" correspondiente al modelado de la implementación de la propuesta de solución. En él se especifican las librerías que se utilizan en cada una de las clases que se implementan en la propuesta de solución.

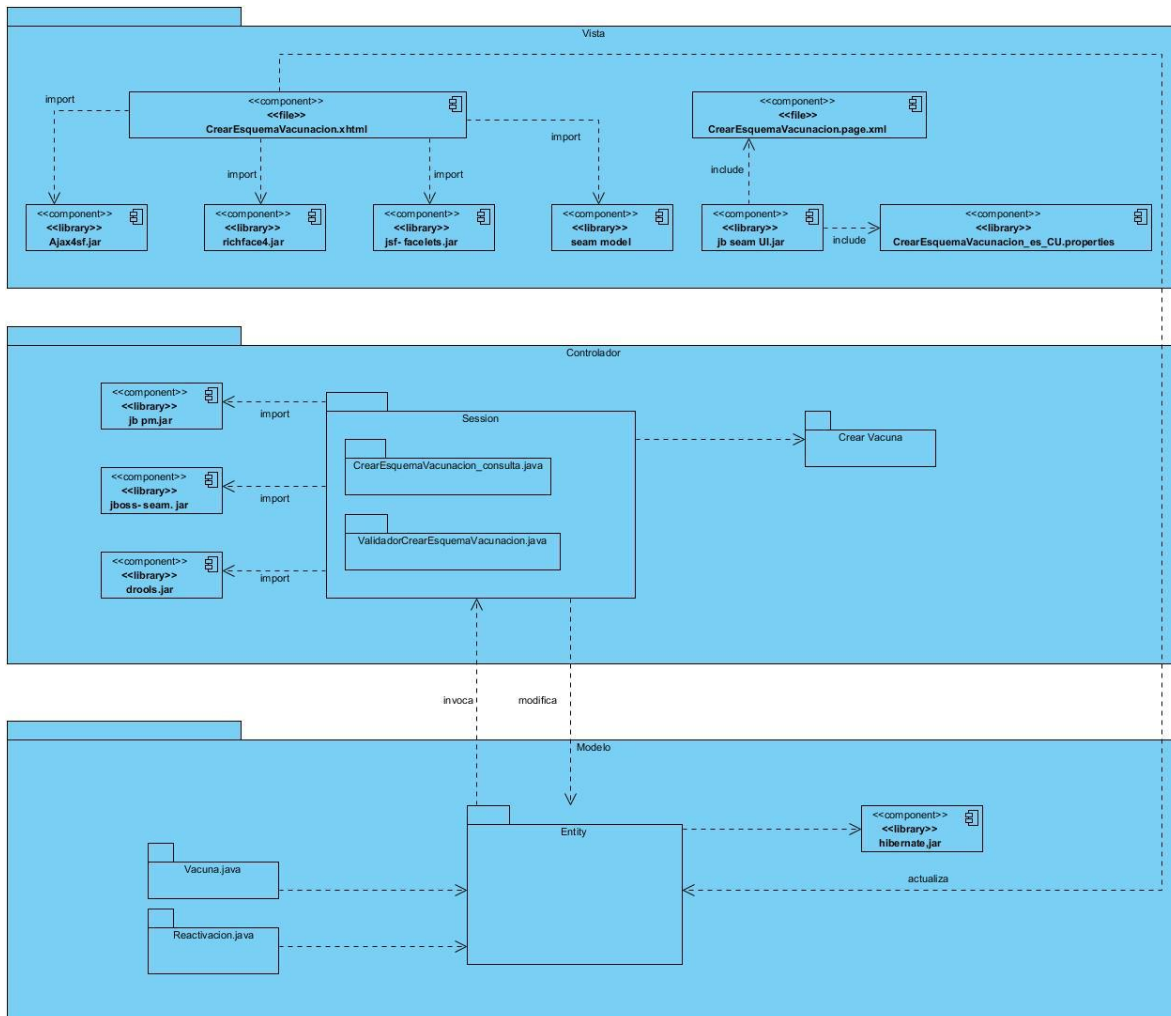


Ilustración 7: Diagrama de componentes. (Elaboración propia)

3.2- Estrategias de codificación. Estándares y estilos usados

Los estándares de programación son convenciones que determinan la forma en que se codificará el programa según el lenguaje de programación que se utilizará. Permiten que los programas sean legibles y facilita el proceso de mantención al momento de realizar una modificación al código. (Pezo A., 2018)

Un estándar de programación es una forma de normalizar la programación de forma tal que, al trabajar en un proyecto, cualquier persona involucrada en el mismo tenga acceso y comprenda el código. Permite definir la escritura y organización del código fuente de un programa. Facilita a un programador la modificación del código fuente, aunque no esté

trabajando en el equipo y definir la forma en la que deben ser declaradas las variables, clases y comentarios. (Arias Calleja, 2014).

A continuación, se presentan los estándares de codificación definidos en el sistema XAVIA-HIS para darle solución a la problemática planteada:

- Se debe utilizar como idioma el español y las palabras no se acentuarán.
- Se deben usar siempre dos líneas en blanco en las siguientes circunstancias:
 - Entre las secciones de un fichero fuente.
 - Entre las definiciones de clases e interfaces.
- Se debe usar una línea en blanco en las siguientes circunstancias:
 - Entre métodos.
 - Entre las variables locales de un método y su primera sentencia.
 - Antes de un comentario de bloque o un comentario de línea.
 - Entre las distintas secciones lógicas de un método para facilitar la lectura.
- Se debe dejar un espacio en blanco en la siguiente situación:
 - Entre una palabra clave del lenguaje y un paréntesis.

En cuanto a las normas de inicialización, declaración y colocación de variables, constantes, clases y métodos:

- Todas las instancias y variables de clases o métodos empezarán con minúscula. Las palabras internas que lo forman, si son compuestas, empiezan con su primera letra en mayúsculas.
- Los nombres de variables no deben empezar con los caracteres subguión "_" o signo de peso "\$", aunque ambos están permitidos por el lenguaje.
- Los nombres de las clases deben ser sustantivos, cuando son compuestos tendrán la primera letra de cada palabra que lo forma en mayúscula. Mantener los

nombres de las clases simples y descriptivas. Usar palabras completas, evitar acrónimos y abreviaturas.

3.3- Seguridad Informática

La Seguridad Informática es la disciplina que se ocupa de diseñar las normas, procedimientos, métodos y técnicas destinadas a conseguir un sistema de información seguro y fiable. Es el área de la Informática que se enfoca en la protección de la infraestructura computacional y todo lo relacionado con esta y especialmente la información contenida o circulante. (Gómez Vieites, 2017).

El sistema XAVIA-HIS tiene a su cargo un conjunto de información sensible sobre la atención a los pacientes, es por ello que el tema de la seguridad informática es de gran impacto en el centro.

Las funcionalidades del sistema encargado de la seguridad son: iniciar y cerrar sesión de trabajo, registrar trazas y administrar seguridad, todas estas operaciones son reutilizadas de otros componentes del sistema.

Para iniciar la sesión de trabajo un usuario debe acceder al sistema e insertar su nombre de usuario y contraseña. El sistema verifica que los datos introducidos sean válidos y de acuerdo a los permisos de este usuario, tiene o no acceso al módulo al que desea entrar. A la configuración de las vacunas y los esquemas de vacunación en el módulo de Epidemiología solo tendrá acceso el epidemiólogo.

El registro de trazas es vital en el sistema ya que archiva las acciones que realiza el usuario, que pueden ser: inicio o cierre de sesión, acceso a un módulo, modificación de un atributo de una entidad o cualquier otra operación. Se hace necesario administrar los permisos que se asignan a los usuarios para la navegación en el sistema y ello se logra a través de la funcionalidad: administrar seguridad. El sistema posibilita asignar o denegar permiso a roles y usuarios en las funcionalidades de los módulos.

3.4- Tratamiento de errores

Las excepciones son el medio que ofrecen algunos lenguajes de programación para tratar situaciones anómalas que puedan suceder mientras se ejecuta un programa. La forma en la que el programador gestione la misma es lo que se

conoce como manejo o tratamiento de la excepción. (UNIVERSIDAD DE LA RIOJA, 2020)

En el sistema XAVIA-HIS el control de las excepciones se realiza en donde se insertan o modifican los datos y se desempeña el proceso de validación. Si ocurre una excepción en la navegación que incumple una redirección, se maneja mediante los `pages.xml` que engloban las excepciones y la página a la que el sistema redirecciona. Para los errores en las clases controladoras se utiliza el componente `FacesMessages` del framework Seam, el cual mediante el bloque `try` detecta cuándo ocurre un fallo y a través del bloque `catch` se manejan dichas excepciones. También se controlan los errores en la validación de datos provenientes de la interfaz de usuario, puesto que encierran una lógica compleja en cierta medida. Existe un archivo denominado `page.xml`, que engloba la configuración de todos los mensajes que se deben mostrar por cada tipo de excepción, así como la página a la que el sistema redirecciona en caso de la aparición de un error sorpresivo.

3.5- Descripción de la propuesta de solución

Después de culminada la etapa de implementación se decide mostrar la solución desarrollada con el objetivo de dar respuesta a la problemática existente. Para ello se detalla la información que se recolecta en los esquemas de vacunación y la manera en que se deberá trabajar explicando el flujo a seguir por el usuario para lograr acceder a cada uno de los requisitos implementados.

El RF1 " Crear vacuna", como se puede visualizar en la Ilustración 8, permite introducir el nombre de la vacuna y la cantidad de dosis a aplicar de dicha vacuna, seleccionar la vía de aplicación, que puede ser: intradérmica, intramuscular u oral y seleccionar si tiene reactivación o no; esta opción despliega el panel listado de reactivación que brinda la posibilidad de seleccionar uno o varios períodos de reactivación existentes o crear uno nuevo en caso de ser necesario.

Crear vacuna

Nombre:* Via:* Cantidad de dosis

Reactivación + Adicionar periodo de reactivación

Listado de reactivación

	Tiempo	Edad de inicio	Edad de fin
<input type="checkbox"/>	10 años	15 años	55 años
<input type="checkbox"/>	5 años	55 años	120 años
<input type="checkbox"/>	3 años	10 años	16 años

⏪ ⏩ ⏴ ⏵

Ilustración 8: Vista del RF1 "Crear vacuna". Panel "Listado de reactivación". (Elaboración propia)

Para Adicionar el período de reactivación (RF7) se debe introducir el tiempo de cada período, la edad de inicio y la edad de fin, además, te permite seleccionar si los atributos anteriores se muestran en días, meses o años (ver ilustración 9).

Crear período de reactivación Buscar...

Período de reactivación

Tiempo:* Edad de inicio:* Edad de fin:*

Aceptar ² Cancelar ¹

Ilustración 9: Vista del RF7 "Crear período de reactivación"

En la implementación se utiliza el componente CIE10 (Clasificador Internacional de Enfermedades v10) para seleccionar la enfermedad que previene la vacuna. Como se puede ver en la Ilustración 10, a partir de los criterios de búsqueda: código, descripción y estructura, se muestra un listado de la o las enfermedades que correspondan con las opciones introducidas o seleccionadas. Un panel de diagnóstico se activa una vez haya sido seleccionada la enfermedad y que se puede desmarcar en caso de error.

CAPÍTULO 3: VALIDACIÓN DE LA PROPUESTA DE SOLUCIÓN

Enfermedad que previene

Código: *	Descripción:	Estructura:	
<input style="width: 90%;" type="text"/>	<input style="width: 90%;" type="text"/>	<Seleccione>	<input type="button" value="Buscar"/>

Búsqueda simple

Clasificación Internacional de Enfermedades (CIE)

	Código	Descripción	Estructura
<input checked="" type="checkbox"/>	(B15- B19)	Hepatitis viral	Grupo
<input type="checkbox"/>	B15	Hepatitis aguda tipo A	Categoría

◀ ▶ ⏪ ⏩

Diagnóstico

Código	Descripción
(B15- B19)	Hepatitis viral

Ilustración 10: Vista del RF1 "Crear vacuna". Componente CIE10. (Elaboración propia)

En el panel región de aplicación, tal como se puede ver en la Ilustración 11, se brinda la posibilidad de seleccionar una o dos regiones de aplicación, en dependencia de la dosis que le corresponde al paciente y que se refleja en un segundo panel llamado diagnóstico, que se puede desmarcar en caso de error.

Región de aplicación

	Nombre
<input checked="" type="checkbox"/>	Deltoide izquierdo
<input type="checkbox"/>	Deltoide derecho

◀ ▶ ⏪ ⏩

Región de aplicación seleccionada

Descripción
Deltoide izquierdo

Ilustración 11: Vista del RF1 "Crear vacuna". Panel "Región de aplicación". (Elaboración propia)

Una vez terminado el proceso, el sistema permite Ver detalles de la vacuna (Ver Anexo 2) permitiendo al usuario modificar la información con la que no se encuentre satisfecho (Ver Anexo 3) o eliminarlo en caso de no haberlo agregado a un esquema de vacunación (Ver Anexo 4). A esta opción se puede llegar también a través de la interfaz Buscar vacuna (Ver Anexo 5) donde introduciendo ciertos criterios de búsqueda el sistema lista la vacuna deseada o muestra la información de que no se encontró la vacuna solicitada permitiendo que esta sea creada. El requisito Ver datos de la vacuna (Ver Anexo 6) se muestra a partir de la búsqueda de la vacuna.

El RF8 " Crear esquema de vacunación" (Ver Ilustración 12), permite introducir el nombre que llevará el esquema de vacunación y una pequeña descripción en el campo observaciones especificando quiénes recibirán las vacunas del esquema y el rango de edades que comprende.

Ilustración 12: Vista del RF8 "Crear esquema de vacunación"

La creación del esquema de vacunación se muestra en la solución a través de un Ver detalles del esquema de vacunación (Ver Anexo 7) luego de haberse configurado las vacunas para ser agregadas. El usuario encargado de la creación del esquema verifica su existencia a través del requisito Buscar esquema de vacunación (Ver Anexo 8) y se listan los esquemas generados dando la posibilidad de que estos puedan ser visualizados (Ver Anexo 9). Al acceder a esta funcionalidad se muestra el esquema de vacunación con todos los datos que se recogieron durante la consulta, con la particularidad de que ninguno de los campos puede ser editado. Para editar los campos se debe acceder a la opción Modificar esquema de vacunación (Ver Anexo 10) y para eliminarlo a la opción Eliminar esquema de vacunación (Ver Anexo 11).

3.6- Pruebas de software

Las pruebas de software (Sanz, 2005) se definen como una actividad en la cual un sistema o uno de sus componentes se ejecutan en circunstancias previamente especificadas, los resultados se observan y luego se registran. Las pruebas de software (Navarro, Ruiz y Perez, 2009) comprenden una fase del proceso de desarrollo que se centra en asegurar la calidad, fiabilidad y robustez de un software, dentro de un contexto

o escenario donde está previsto que este sea utilizado. Son investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto.

Las estrategias de pruebas se describen en muchas literaturas como un proceso en espiral que comienza por la ingeniería del sistema, levantamiento de requisitos, el diseño de los requerimientos y la implementación de los mismos. Al mismo nivel de cada una de estas etapas de desarrollo de software se realizan las pruebas del sistema, validación, integración y unidad. (Sommerville, 2005)

Para comprobar el correcto funcionamiento de la solución obtenida se realizan los siguientes tipos de prueba:

- **Pruebas funcionales:** se basan en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas.
- **Pruebas estructurales:** se basan en medir la totalidad de las pruebas mediante la evaluación del tipo de estructura.

A partir de la utilización de las técnicas de pruebas dinámicas se pudieron utilizar los métodos de caja negra para las pruebas funcionales y los de caja blanca para las estructurales.

3.6.1- Método de caja blanca. Técnica de camino básico.

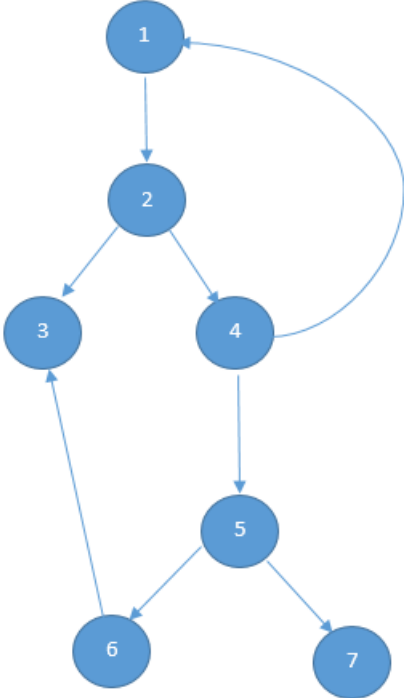
El método de caja blanca se centra en los detalles procedimentales del software, por lo que su diseño está ligado al código fuente. (Sánchez Peño, 2015).

La técnica empleada en este tipo de prueba fue la de camino básico, que consiste en diseñar un caso de prueba por cada camino independiente del programa, lo que garantiza que se prueben todos los caminos de ejecución del programa al menos una vez. (Sánchez Peño, 2015). Además, permite obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. (Sommerville, 2011)

Complejidad ciclomática

La complejidad ciclomática se realiza a todos los métodos de un sistema informático. En la Tabla 6, se presenta la prueba realizada al método listarVacunas (), que permite listar las vacunas a añadir al esquema de vacunación.

Tabla 6: Prueba de unidad utilizando complejidad ciclomática. Método listarVacuna (). (Elaboración propia)

Fórmula 1	Fórmula 2	Fórmula 3	Grafo resultante
$V(G) = (A - N) + 2$ $V(G) = (8 - 7) + 2$ $V(G) = 3$	$V(G) = P + 1$ $V(G) = 2 + 1$ $V(G) = 3$	$V(G) = R + 1$ $V(G) = 2 + 1$ $V(G) = 3$	
<p>A: cantidad de aristas (8)</p> <p>N: cantidad de nodos (7)</p> <p>P: número de nodos predicado contenidos en el grafo de flujo G.</p> <p>Nodo predicado: cada nodo que contiene una condición.</p> <p>R: cantidad de regiones en el grafo.</p>			
<p>Prueba de complejidad ciclomática</p>			
<p>Después de aplicadas las tres fórmulas podemos observar que el método tiene una complejidad ciclomática igual a 3 demostrando la efectividad de las fórmulas.</p>			
<p>Prueba del camino básico</p>			
<p>Una vez calculada la complejidad se define como límite superior lo que indica hay que realizarle al código 3 pruebas para garantizar que se ejecute completamente al menos</p>			

una vez. La funcionalidad listarVacuna () posee poco riesgo debido a que el resultado se encuentra en el intervalo de 1-9.

El total de caminos independientes establecidos fue de 4 y se muestran a continuación:

Camino 1: 1, 2, 3.

Camino 2: 1, 2, 4, 1, 2, 3.

Camino 3: 1, 2, 4, 5, 7.

Camino 4: 1 2, 4, 5, 6, 3.

Al comprobar los caminos posibles se verificó que los diseños de casos de pruebas elaborados cubrían los 4 caminos posibles, de forma tal que siempre fue posible corroborar el correcto funcionamiento de la aplicación para este método.

3.6.2- Pruebas de integración. Método de caja negra. Técnica de participación equivalente.

Son una técnica sistemática para construir la arquitectura del software mientras se llevan a cabo pruebas para descubrir errores asociados con la interfaz. El objetivo es tomar los componentes probados de manera individual y construir una estructura de programa que se haya dictado por diseño. Dichas pruebas se pueden realizar de manera ascendente si los módulos son de menor complejidad mientras que si la complejidad es alta la integración se realiza de forma descendente. (Pressman, 2010). Se realizan después de haber realizado cada una de las pruebas unitarias con la intención de probar los métodos aplicados en el desarrollo.

Para la integración de las funcionalidades que permitieron implementar los esquemas de vacunación en el sistema XAVIA-HIS se utilizó la integración ascendente, ya que primero se integraron las clases pertenecientes al requisito Crear vacuna. Al aplicar estas pruebas se pudo constatar que los flujos de información con el resto de los módulos del sistema funcionan correctamente.

Método de caja negra. Técnica de participación equivalente.

Las pruebas de caja negra o pruebas de comportamiento se enfocan en los requerimientos funcionales del software. Esta prueba intenta encontrar errores en las siguientes categorías: funciones incorrectas o faltantes, errores de interfaz, errores en la estructura de datos o en el acceso a bases de datos externas, errores de comportamiento o rendimiento y errores de iniciación y terminación. (Pressman, 2010)

De las técnicas de prueba de caja negra se utiliza la técnica de partición equivalente la cual divide el campo de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba. Un caso de prueba ideal descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar. (Pressman, 2010)

Tabla 7: Resultados del método de caja negra mediante la técnica de participación equivalente en las pruebas de integración. (Elaboración propia)

Iteraciones	NC detectadas	Correspondencia	Interfaz	Funcionalidad	NC resueltas
1ra	20	6	8	6	12
2da	7	3	2	2	5
3ra	0	0	0	0	0
Total	27	9	10	8	17

Fueron realizadas 3 iteraciones de los casos de pruebas para erradicar las no conformidades (NC) encontradas, clasificándose en correspondencia, interfaz y funcionalidad. A la par de las iteraciones se realizaron pruebas de regresión para comprobar si las no conformidades de la iteración anterior fueron resueltas.

Correspondencia: se encontró que las descripciones de varios escenarios no están en correspondencia con sus nombres.

Interfaz: en algunos casos se encontraron paneles que no cumplían con las pautas de diseño establecidas.

Funcionalidad: la respuesta del sistema no coincidía con el comportamiento que tenía el sistema al probar ciertos escenarios.

Mediante el diseño de casos de prueba se comprobó el correcto funcionamiento de las funcionalidades desarrolladas para la configuración de los esquemas de vacunación, solucionando las no conformidades en la 3ra iteración.

3.6.3- Pruebas de aceptación. Método de caja negra. Técnica de participación equivalente.

Una prueba de aceptación puede variar desde una "prueba de conducción" informal hasta una serie de pruebas planificadas y ejecutadas sistemáticamente. La mayoría de los constructores de software realizan un proceso llamado pruebas alfa y pruebas betas. La diferencia entre dichas pruebas consiste en que por lo general en la prueba beta el desarrollador no está presente y por tanto no puede controlar el proceso. (Sommerville, 2005)

A la propuesta de solución le fueron aplicadas pruebas de aceptación de tipo alfa, pues el cliente, en este caso el proyecto, junto al desarrollador realizaron pruebas guiadas por casos de prueba mediante el método de caja negra, utilizando la técnica de partición equivalente.

Se tomó como base los 13 requisitos funcionales, generándose la misma cantidad de Diseños de Casos de Prueba (DCP). A continuación, se muestran los resultados obtenidos:

Iteraciones	NC detectadas	Correspondencia	Interfaz	NC resueltas
1ra	6	3	3	6
2da	4	2	2	4
3ra	1	0	1	0

Total	11	5	6	10
--------------	----	---	---	----

Las causas de las no conformidades detectadas fueron:

- **Errores de interfaz:** algunos paneles no cumplían con las pautas de diseño establecidas.
- **Errores de correspondencia:** las descripciones de algunos escenarios no estaban en correspondencia con su nombre.

3.7- Conclusiones parciales

- La utilización del MVC permitió el modelado de los diferentes artefactos ingenieriles utilizados como punto de partida para la implementación de la propuesta de solución.
- El uso de los estándares de codificación y el tratamiento de errores fortalecen la construcción del código proporcionándole legibilidad, independencia y mantenibilidad.
- Las pruebas de software permitieron constatar la calidad en la implementación realizada.

Conclusiones

Al finalizar la investigación quedó demostrado el cumplimiento del objetivo de la investigación: desarrollar funcionalidades para la configuración de los esquemas de vacunación en el sistema XAVIA-HIS. Por lo que se llegó a las siguientes conclusiones:

- El análisis de los sistemas permitió definir las características que debe poseer la propuesta de solución, haciendo posible estructurar la información a recoger en los esquemas de vacunación.
- Se asumieron las herramientas y las tecnologías establecidas en el CESIM posibilitando la integración de la solución propuesta con el sistema XAVIA-HIS.
- El uso de la metodología AUP-UCI hizo posible contar con los artefactos ingenieriles acorde a los documentados en el expediente del proyecto del sistema XAVIA-HIS.
- La integración de la propuesta de solución al sistema XAVIA-HIS brindó la posibilidad de contar, en el sistema, con los esquemas de vacunación existentes en el país, con datos específicos, logrando obtener una historia clínica del paciente más completa y un mejor seguimiento del paciente en las consultas.
- La ejecución de las pruebas, permitió detectar y corregir deficiencias presentes en la aplicación.

Recomendaciones

Después de analizar los resultados del presente trabajo de diploma se recomienda:

- Implementar las funcionalidades en el módulo de Enfermería que permitan Gestionar las solicitudes de vacunación para una mayor eficiencia en la atención médica.
- Modificar las funcionalidades de Gestionar vacunas en el módulo de Enfermería agregándole una sección que muestre las vacunas previamente aplicadas y, además, permita buscar la solicitud de vacunación asociada al paciente para poder registrar la nueva vacuna o dosis aplicada.

Referencias bibliográficas

- 1- ALARCÓN, J., 2009. Epidemiología: concepto, usos y perspectivas. *Revista Peruana de Epidemiología*, vol. 13, no. 1, pp. 1-3. ISSN 1609-7211.
- 2- APP de la OMS para vacunas | Vacunas / Asociación Española de Vacunología. [en línea], 2019. [Consulta: 13 febrero 2020]. Disponible en: <https://www.vacunas.org/app-de-la-oms-para-vacunas-2/>.
- 3- Arias Calleja, Manuel. 2014. *Carmen. Estándares de codificación*. 2014. Ayuda en línea SISA - Sistema Integrado de Información Sanitaria Argentino. [en línea], 2013. [Consulta: 13 febrero 2020]. Disponible en: https://sisa.msal.gov.ar/sisadoc/docs/050203/nomivac_intro.jsp.
- 4- BRIOL, P., 2008. *BPMN, the Business Process Modeling Notation Pocket Handbook*. S.I.: LuLu. com. ISBN 1409202992.
- 5- CANELA-SOLER, J., ELVIRA-MARTÍNEZ, D., LABORDENA-BARCELÓ, M.J. y LOYOLA-ELIZONDO, E., 2010. Sistemas de Información en Salud e indicadores de salud: una perspectiva integradora. *Medicina Clinica*, vol. 134, no. SUPPL. 1, pp. 3-9. ISSN 00257753. DOI 10.1016/S0025-7753(10)70002-6.
- 6- CENTRO DE INFORMÁTICA MÉDICA, C., DEPARTAMENTO DE SISTEMAS DE GESTIÓN HOSPITALARIA, C. y GRUPO DE MERCADOTECNIA, C., 2018. *CESIM_Salud_Manual_de_usuario_HIS_Epidemiologia*. 2018. S.I.: s.n.
- 7- Clinic Cloud- Principales beneficios de las TIC en la salud. [en línea], 2015. [Consulta: 10 de febrero 2020]. Disponible en: <https://clinic-cloud.com/blog/beneficios-tics-en-salud>.
- 8- CORPORATION, O., 2010. The Java EE 6 Tutorial. [en línea], no. November. [Consulta: 20 febrero 2020]. Disponible en: <https://docs.oracle.com/javaee/6/tutorial/doc/index.html>.
- 9- Dynamic AJAX - Interactive Website Programming - AJAX. [en línea], 2005. [Consulta: 20 febrero 2020]. Disponible en: http://www.dynamicajax.com/fr/What_Is_AJAX-271.html.

- 10- EGUÍLUZ PÉREZ, J., 2008. Introducción a CSS. *Introducción a CSS*, FERNÁNDEZ, O.B., 2005. Introducción al lenguaje de programación Java. *Una guía básica*, vol. 9.
- 11- FILOCAMO, D., 2009. *JBoss RichFaces 3.3* [en línea]. S.l.: s.n. [Consulta: 20 febrero 2020]. ISBN 9781847196880. Disponible en: <https://books.google.com/cu/books?id=vTQNAgAAQBAJ&printsec=frontcover&dq=filocamo+2009&hl=es&sa=X&ved=0ahUKEwj5ez73-HnAhXRMd8KHYY-AkwQ6AEIMDAB#v=onepage&q=filocamo 2009&f=false>.
- 12- García Garcés, H., Navarro Aguirre, L., López Pérez, M., Rodríguez Orizondo, M.F., 2014. Tecnologías de la Información y la Comunicación en salud y educación médica. *Revista de Ciencias Médicas de Santa Clara*, vol. 6, no. 1, ISSN 2077-2874.
- 13- GARCIA, M., 2017. MVC (Modelo-Vista-Controlador): ¿qué es y para qué sirve? *octubre 05* [en línea]. [Consulta: 25 febrero 2020]. Disponible en: <https://codingornot.com/mvc-modelo-vista-controlador-que-es-y-para-que-sirve>.
- 14- Gómez Vieites, Álvaro. 2017. *Enciclopedia de la Seguridad Informática 2da Edición*. s.l. : RA-MA, 2017.
- 15- GONZÁLEZ, Y.D. y ROMERO, Y.F., 2012. Patrón Modelo-Vista-Controlador. *Revista Telemática*, vol. 11, no. 1, pp. 47-57. ISSN 1729-3804.
- 16- GRAU, X.F. y SEGURA, M.I.S., 2008. Desarrollo orientado a objetos con UML. *Recuperado el*, vol. 1.
- 17- Herramientas de diseño » AXURE, Diseño de wireframes y Prototipos. [en línea], 2020. [Consulta: 20 febrero 2020]. Disponible en: <http://multimedia.uoc.edu/blogs/labeines/es/prototipat/axure-disseny-de-wireframes-i-prototips/>.
- 18- HERRERO JAÉN, S., 2016. Formalización del concepto de salud a través de la lógica: impacto del lenguaje formal en las ciencias de la salud. *Ene*, vol. 10, no. 2, pp. 0. ISSN 1988-348X.

- 19- HIBERNATE, 2018. Your relational data. Objectively. - Hibernate ORM. [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <http://hibernate.org/orm/>.
- 20- HOSPITAL JUAN CANALEJO [en línea], 2017. 2017. S.l.: s.n. [Consulta: 20 febrero 2020]. Disponible en: <https://www.hospitaljuancardona.es/>.
- 21- IAN SOMMERVILLE, 2012. Requerimientos Del Software De. *Universidad Veracruzana* [en línea]. S.l.: [Consulta: 19 febrero 2020]. Disponible en: <https://es.slideshare.net/RosaOrtega6/requerimientos-del-software-77144553>.
- 22- ING. PEDRO ERNESTO SALAS OLIVA, DR.C . ARTURO ORELLANA GARCÍA, ING. LISSETTE SOTO PELEGRIN, D.C.J.F.R.P., 2018. Instrucciones para la preparación de Ponencias para Informática 2009 - cts143c.pdf. [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <moz-extension://cdd1bedb-8ba9-42f7-a471-660c2afb645f/enhanced-reader.html?openApp&pdf=http%3A%2F%2Fwww.informaticahabana.cu%2Fsites%2Fdefault%2Ffiles%2Fponencias2018%2FSLD12.pdf>.
- 23- JavaServer Faces.org. [en línea], 2018. [Consulta: 20 febrero 2020]. Disponible en: <http://www.java-serverfaces.org/>.
- 24- LAMARCA, M., 2018. Hipermedia/Multimedia. *Hipertexto: El nuevo concepto de documento en la cultura de la imagen. Recuperado de <http://www.hipertexto.info/documentos/hipermedia.htm>,*
- 25- Lanzan app iMuni para el control y seguimiento de vacunas - Secretaría Nacional de Ciencia, Tecnología e Innovación. [en línea], 2017. [Consulta: 13 febrero 2020]. Disponible en: <https://www.senacyt.gob.pa/lanzan-app-imuni-para-el-control-y-seguimiento-de-vacunas/>.
- 26- López Ambrón L., Egües Torres Ll., Pérez Carreras A., Galindo Santana BM., Galindo Sardiña MA., Resik Aguirre S., et al. Experiencia cubana en inmunización, 1962-2016. *Rev. PanamSaludPublica*. 2018; 42:e34.
- 27- LÓPEZ-MORENO, S., GARRIDO-LATORRE, F. y HERNÁNDEZ-AVILA, M., 2000. Desarrollo histórico de la epidemiología: su formación como disciplina científica. *Salud pública de México*, vol. 42, pp. 133-143. ISSN 0036-3634.

- 28- MARTÍNEZ, M.P.A., DE JUANES PARDO, J.R. y DE CODES ILARIO, A.G., 2015. Conceptos generales. Calendarios de vacunación sistemática del niño y del adulto en España. Impacto de los programas de vacunación. *Enfermedades Infecciosas y Microbiología Clínica*, vol. 33, no. 1, pp. 58-65. ISSN 0213-005X.
- 29- NAVARRO, P.L.M., RUIZ, D.S. y PEREZ, G.M., 2009. Open HMI Tester: un Framework Open-source para herramientas de pruebas de software. *Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos*, vol. 3, no. 4.
- 30- PANTOJA, E.B., 2004. El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing. *Acta Nova*, vol. 2, no. 4, pp. 493.
- 31- PÉREZ LABRADOR, J.H., ARENCIBIA PRIETO, M., JIMÉNEZ DÍAZ, D.E. y TELLERÍA PRIETO, M. del C., 2018. Sistema de Información Clínico Hospitalaria. Arquitectura y Mapa de camas. *Revista de Ciencias Médicas de Pinar del Río*, vol. 22, no. 1, pp. 92-100. ISSN 1561-3194.
- 32- Pezo A. Felipe., 2018. La importancia de los estándares de programación. [en línea]. [Consulta: 10 septiembre 2020]. Disponible en: <https://fepa.cl/la-importancia-de-los-estandares-de-programacion/>.
- 33- PGADMIN.ORG, 2018. pgAdmin - PostgreSQL Tools. [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <https://www.pgadmin.org/>.
- 34- PORRAS, O., ABDELNOUR, A., ARTAVIA, E., BALDODANO, A., BARRANTES, M., FERNANDEZ, J. y RIVERA, R., 2006. Esquema de vacunación para el grupo de edad de 0-18 años. *Acta Médica Costarricense*, vol. 48, no. 4, pp. 204-207.
- 35- Pressman, Roger.S., 2010, *Ingeniería de software. Un enfoque práctico. Séptima edición*. ISBN: 978-607-15-0314-5.
- 36- RODRÍGUEZ, T., 2014. Metodología de desarrollo para la Actividad productiva de la UCI. , pp. 1-16.

- 37- ROUSE, M., 2016. ¿Qué es Modelado de datos? - Definición en WhatIs.com. *TechTarget* [en línea]. [Consulta: 21 febrero 2020]. Disponible en: <https://searchdatacenter.techtarget.com/es/definicion/Modelado-de-datos>.
- 38- RUGAL JUAN, 2013. ¿Qué es JavaScript (JS) y para qué se utiliza? [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <https://www.tusclases.mx/blog/que-javascript>.
- 39- SánchezPeño, José Manuel. 2015. Pruebas de software. Fundamentos y Técnicas. Madrid: Universidad Politécnica de Madrid, 2015.
- 40- SANZ, L.F., 2005. Un sondeo sobre la práctica actual de pruebas de software en España. *REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software*, vol. 1, no. 2, pp. 43-54. ISSN 1885-4486.
- 41- SEAM, 2009. Seam Framework - JBoss Seam. [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <http://seamframework.org/>.
- 42- SERRANO PERELLÓ J., 2016. Análisis de soluciones cloud para almacenamiento de archivos y trabajo colaborativo. Valencia. Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universidad Politécnica de Valencia.
- 43- SEVILLA, G. de I. del S. y B. de D. de L. y S.I. de, 2012. Introducción al Modelado Conceptual | Enhanced Reader. [en línea]. [Consulta: 19 febrero 2020]. Disponible en: <moz-extension://cdd1bedb-8ba9-42f7-a471-660c2afb645f/enhanced-reader.html?openApp&pdf=http%3A%2F%2Fwww.lsi.us.es%2Fdocencia%2Fget.php%3Fid%3D6889>.
- 44- SOMMERVILLE, I., 2005. *INGENIERIA DEL SOFTWARE. Séptima edición* [en línea]. S.l.: s.n. ISBN 8478290745. Disponible en: http://danielr.obolog.es/ingenieria-software-355416%5Cnhttp://fondoeditorial.uneg.edu.ve/citeg/numeros/c02/c02_art10.pdf.
- 45- TENIENTE LÓPEZ E, COSTAL COSTA D, S.S.M., 2013. *Diagrama de paquetes* [en línea]. S.l.: s.n. [Consulta: 21 febrero 2020]. Disponible en: <https://diagramasuml.com/paquetes/>.

- 46- Tich Consulting GRUPO ASISA, 2017. Sistemas de Información Hospitalaria. [en línea]. [Consulta: 11 septiembre 2020]. Disponible: <http://download.microsoft.com/documents/es-es/digital-heroes/GreenCube-Tich-Consulting.pdf>
- 47- THE POSTGRESQL GLOBAL DEVELOPMENT GROUP, 2018. PostgreSQL: Documentation: 9.4: What is PostgreSQL? [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <https://www.postgresql.org/docs/9.4/intro-what-is.html>.
- 48- Unidad de Hospitalización - CO.SA.GA. [en línea], 2012. [Consulta: 20 febrero 2020]. Disponible en: <https://cosaga.com/servicios/unidad-de-hospitalizacion/>.
- 49- UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS, 2019. Productos de software. [en línea]. [Consulta: 17 de febrero 2020]. Disponible en: <http://www.uci.cu/investigacion-y-desarrollo/productos>.
- 50- UNIVERSIDAD DE LA RIOJA, 2020. Tratamiento de errores. [en línea]. [Consulta: 10 septiembre 2020]. Disponible en https://www.unirioja.es/cu/jearansa/0910/archivos/EIPR_Tema05.pdf.
- 51- VACGAL - A app do Programa Galego de Vacinación - Consellería de Sanidade - Servizo Galego de Saúde. [en línea], 2018. [Consulta: 20 febrero 2020]. Disponible en: <https://www.sergas.gal/Saude-publica/VACGAL-A-app-do-PGV>.
- 52- VIEITES, Á.G., 2011. *Enciclopedia de la seguridad informática*. S.I.: Grupo Editorial RA-MA. ISBN 8499643949.
- 53- VISCONTI, M. y ASTUDILLO, H., 2012. Fundamentos de Ingeniería de Software. ,
- 54- VISUAL PARADIGM, 2019. Visual Paradigm Product Overview. [en línea]. [Consulta: 20 febrero 2020]. Disponible en: https://www.visual-paradigm.com/support/documents/vpuserguide/12/13/5963_visualparadi.html.
- 55- What is Jboss Application Server? [en línea], 2018. [Consulta: 20 febrero 2020]. Disponible en: <https://www.techopedia.com/definition/3525/jboss-application-server-jboss-as>.

56-WOGUIA, E.N., 2017. *JBoss: Developers Guide*. S.I.: Packt Publishing Ltd. ISBN 1788292928.

Anexos

Anexo 1: Descripción de requisitos por proceso.

Tabla 8: Especificación del RF8 "Crear esquema de vacunación. (Elaboración propia).

Descripción textual	El requisito inicia cuando el actor accede a la opción Crear esquema de vacunación y el sistema brinda la posibilidad de introducir los datos, el actor introduce los datos del esquema, el sistema crea el esquema y el requisito termina.
Actores	Epidemiólogo
Precondiciones	Deben haberse registrado los diferentes tipos de vacunas que se asocian a los esquemas de vacunación
Flujo de eventos	
Flujo básico Crear esquema de vacunación	
1.	El requisito inicia cuando el actor accede a la opción Crear esquema de vacunación.
2.	<p>El sistema brinda la posibilidad de introducir los datos:</p> <ul style="list-style-type: none"> • Nombre • Observaciones <p>Brinda la posibilidad de seleccionar las:</p> <ul style="list-style-type: none"> • Vacunas <p>y permite:</p>

	<p>- Aceptar.</p> <p>- Cancelar operación. Ver Flujo alternativo 1: “Cancelar operación.”</p>
3.	El actor introduce los datos deseados.
4.	El actor selecciona la opción Aceptar.
5.	<p>El sistema valida los datos.</p> <p>Si hay datos incompletos. Ver Flujo Alternativo 2: “Existen datos incompletos.”</p> <p>Si hay datos incorrectos. Ver Flujo Alternativo 3: “Existen datos incorrectos.”</p>
6.	El sistema crea el esquema de vacunación y muestra los datos de la misma. Se ejecuta el requisito, ver requisito Ver detalles del esquema de vacunación.
7.	El requisito termina.
Sección 1	
N/A	
Flujos alternativos	
Flujo alternativo 1 “Cancelar operación”	
1.	El actor selecciona la opción Cancelar operación.
2.	El sistema regresa a la interfaz Crear esquema de vacunación.

3.	Regresa al paso 7 del Flujo básico	
Flujo alternativo 2 “Existen datos incompletos”		
1.	El sistema muestra un indicador rojo sobre los campos nombre y vacunas seleccionadas.	
2.	Regresa al paso 3 del Flujo básico .	
Flujo alternativo 3 “Existen datos incorrectos”		
1.	El sistema muestra un indicador rojo sobre el campo nombre.	
2.	Regresa al paso 3 del Flujo básico .	
Pos-condiciones		
1.	Se creó el esquema de vacunación	
Validaciones		
1.	Modelo conceptual de Enfermería	
Conceptos	Esquema de vacunación	Modelo conceptual Enfermería
	N/A	Modelo conceptual Elementos Comunes
Restricciones	N/A	

del sistema		
Dependencias	Obligatoria	Ver detalles del esquema de vacunación
	Opcional	N/A
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Prototipo elemental de interfaz gráfica de usuario

Crear esquema de vacunación Q Buscar...

Datos del esquema de vacunación

Nombre:*

Vacunas

- BCG
- Heberbiovac HB
- Heberpenta-L
- VA-MENGOC-BC
- IPV
- DTP- VAC*
- Quimi- Hib*
- PRS
- VA-DIFTET
- Vax-TYVI
- Vax- TET
- OPV

Vacunas seleccionadas

Observaciones:

Aceptar ⁶ Cancelar ⁵

Formatos de entrada/salida

N/A

Entradas

N/A

Salidas

N/A

Anexo 2: Vista del RF2 "Ver detalles de la vacuna"

Ver detalles de la vacuna Q Buscar...

Datos de la vacuna

Nombre: BCG Vía: ID Cantidad de dosis: 1 1 Reactivación

Diagnóstico

Código	Descripción
(A15- A19)	Tuberculosis 🗑️

Región de aplicación seleccionada

Descripción
Deltoide izquierdo 🗑️

Modificar 6
Eliminar 7
Salir 2

© Universidad de las Ciencias Informáticas, 2020

Ilustración 13: Vista del RF2 "Ver detalles de la vacuna. (Elaboración propia)

Anexo 3: Vista del RF5 "Modificar vacuna"

Modificar vacuna Buscar...

Datos de la vacuna

Nombre:* Via:* Cantidad de dosis

Reactivación

Listado de reactivación

Tiempo	Edad de inicio	Edad de fin
<input type="checkbox"/> 10 años	15 años	55 años
<input type="checkbox"/> 5 años	55 años	120 años
<input type="checkbox"/> 3 años	10 años	16 años

<< >>

Enfermedad que previene

Código:* Descripción: Estructura:

Clasificación Internacional de Enfermedades (CIE)

Código	Descripción	Estructura
<input type="checkbox"/> (A15- A19)	Tuberculosis	Grupo

<< >>

Diagnóstico

Código	Descripción
<input type="checkbox"/> (A15-A19)	Tuberculosis

Región de aplicación

Nombre
<input type="checkbox"/> Deltoide izquierdo
<input type="checkbox"/> Deltoide derecho

<< >>

Región de aplicación seleccionada

Descripción
<input type="checkbox"/> Deltoide izquierdo

Ilustración 14: Vista del RF5 "Modificar vacuna". (Elaboración propia)

Anexo 4: Vista del RF6 " Eliminar vacuna "

Eliminar vacuna
✕

Se eliminará la vacuna seleccionada. Si selecciona Sí se perderán todos los datos. ¿Desea continuar?

Ilustración 15: Vista del RF6 "Eliminar vacuna". (Elaboración propia)

Anexo 5: Vista del RF3 " Buscar vacuna "

Buscar vacuna Q Buscar...

Criterios de búsqueda << 1

Nombre:*

Buscar Cancelar 2

Búsqueda simple 3

+ Adicionar vacuna 4

Listado de vacunas

Nombre	Vía	Cantidad de dosis	Reactivación	
BCG	ID	1	No	
Heberbiovac HB	IM	1	No	

⏪ < > ⏩

Salir

Ilustración 16: Vista del RF3 "Buscar vacuna". (Elaboración propia)

Anexo 6: Vista del RF4 " Ver datos de la vacuna "

Ver datos de la vacuna Q Buscar...

Datos de la vacuna

Nombre: BCG Vía: ID Cantidad de dosis: 1 1 Reactivación

Diagnóstico

Código	Descripción	
(A15- A19)	Tuberculosis	

Región de aplicación seleccionada

Descripción

Deltoide izquierdo

© Universidad de las Ciencias Informáticas, 2020

Salir 2

Ilustración 17: Vista del RF4 "Ver datos de la vacuna". (Elaboración propia)

Anexo 7: Vista del RF9 " Ver detalles del esquema de vacunación"

Ver detalles del esquema de vacunación Q Buscar...

Datos del esquema de vacunación

Nombre: Esquema de vacunación1

Listado de vacunas

Nombre
BCG
Heberbiovac HB

Observaciones:

Este esquema es para el grupo de 0-18 años

Modificar Eliminar Salir

10

Ilustración 18: Vista del RF9: " Ver detalles del esquema de vacunación". (Elaboración propia)

Anexo 8: Ver RF10 " Buscar esquema de vacunación"

Buscar esquema de vacunación Q Buscar...

Criterios de búsqueda

Nombre: *

Buscar Cancelar

+ Adicionar esquema de vacunación

Listado de esquemas

Nombre	
Esquema de vacunación1	
Esquema de vacunación2	

Salir

© Universidad de las Ciencias Informáticas, 2020

1

Ilustración 19: Vista del RF10 "Buscar esquema de vacunación". (Elaboración propia)

Anexo 9: Ver RF11 " Ver datos del esquema de vacunación"

Ver datos del esquema de vacunación

Q Buscar...

Datos del esquema de vacunación

Nombre: Esquema de vacunación1

Listado de vacunas

Nombre
BCG
Heberbiovac HB

Observaciones:

Este esquema es para el grupo de 0-18 años

Salir

Ilustración 20: Ver RF11 " Ver datos del esquema de vacunación". (Elaboración propia)

Anexo 10: Vista del RF12 " Modificar esquema de vacunación"

Modificar esquema de vacunación

Q Buscar...

Datos del esquema de vacunación

Nombre:*
Esquema de vacunación1

Vacunas

- BCG
- Heberbiovac HB
- Heberpenta- L
- VA-MENGOC- BC
- IPV
- DTP- VAC*
- Quimi- Hib*
- PRS
- VA-DIFTET
- Vax-TYVI
- Vax- TET
- OPV

Vacunas seleccionadas

- BCG
- Heberbiovac HB
- Heberpenta- L
- VA-MENGOC- BC
- IPV
- DTP- VAC*
- Quimi- Hib*
- PRS
- VA-DIFTET
- Vax-TYVI
- Vax- TET
- OPV

Observaciones:

Este esquema es para el grupo de 0-18 años

Aceptar Cancelar

Ilustración 21: Ver RF12 " Modificar esquema de vacunación". (Elaboración propia)

Anexo 11: Vista del RF13 " Eliminar esquema de vacunación "

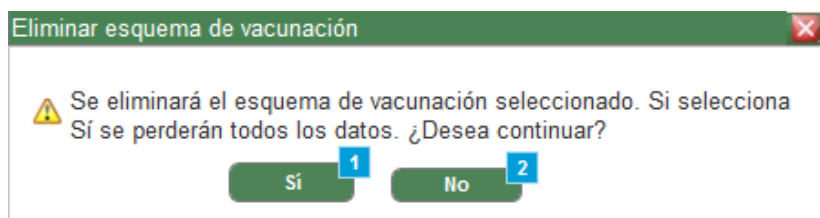


Ilustración 22: Ver RF13 " Eliminar esquema de vacunación". (Elaboración propia)

Glosario de términos

Antígeno de superficie (HBsAg): un resultado positivo o reactivo de la prueba de HBsAg significa que la persona está infectada con el virus de la hepatitis tipo B y puede ser una infección aguda o crónica. Las personas infectadas pueden transmitir el virus a través de su sangre.

BCG (bacilo de Calmette- Guérin): es una vacuna contra la enfermedad de tuberculosis. También se emplea como tratamiento contra el cáncer ya que actúa como modificante de la respuesta biológica.

Deltoides: músculo del hombro que debe su nombre por su similitud a la letra grieta delta.

DTP-VAC*: protege contra la difteria, tosferina y tétanos.

Heberbiovac HB (vacuna recombinante contra la hepatitis tipo B): protege contra el virus de la hepatitis tipo B y sus secuelas, y resulta un buen aliado en el control de la hepatitis de tipo D.

Heberpenta L (vacuna pentavalente líquida): es una vacuna que protege contra la difteria, tétanos, tosferina, hepatitis tipo B y haemophilus Influenzae tipo B.

Haemophilus Influenzae: son cocobacilos Gram-negativo no móviles responsables de una serie de enfermedades como la meningitis y la neumonía.

ID: forma de administración rápida donde el medicamento se inyecta en la piel.

IM: forma de administración rápida donde el medicamento se inyecta dentro del músculo.

IVP: vacuna que puede proteger a las personas contra la polio, enfermedad causada por un virus. Se contagia principalmente de persona a persona o al consumir comida o bebida infectada por la materia fecal de una persona infectada.

OPV: vacuna antipoliomelítica oral.

PRS: vacuna contra el sarampión, la papera y la rubéola.

Quimi-Hib*: es una inmunización activa contra las enfermedades invasivas causadas por el haemophilus Influenzae tipo b para niños entre los 2 meses y los 5 años de edad.

SC: forma de inyección subcutánea, que se aplica en el tejido adiposo, justo debajo de la piel.

VA-DIFTET: está indicada para la prevención del tétano y la difteria en niños menores de 7 años. También se recomienda en niños con afectaciones neurológicas comprobadas en los cuales debe evitarse la inmunización con la vacuna DTP-VAC* o para completar el ciclo primario de vacunación si las dosis previas de DTP-VAC* le ha causado reacción adversa.

Vax- TET: vacuna antitetánica.

Vax-TYVI: vacuna antitifoídica de polisacáridos.