

**PRÁCTICA EMPRESARIAL EN IBM DE COLOMBIA: DESARROLLO,
IMPLEMENTACIÓN Y SOPORTE DE SOLUCIONES PARA EL SECTOR SALUD
EN COLOMBIA POR MEDIO DE LAS HERRAMIENTAS SOFTWARE IBM CARE
MANAGEMENT Y CÚRAM, PARA LA GESTIÓN DEL CUIDADO PREVENTIVO
Y TRATAMIENTO A PACIENTES SANOS O IDENTIFICADOS CON
ENFERMEDADES CRÓNICAS.**

SANTIAGO NIÑO PARRA

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMECHANICAS
ESCUELA DE INGENIERIA DE SISTEMAS E INFORMATICA
BUCARAMANGA
2017**

**PRÁCTICA EMPRESARIAL EN IBM DE COLOMBIA: DESARROLLO,
IMPLEMENTACIÓN Y SOPORTE DE SOLUCIONES PARA EL SECTOR SALUD
EN COLOMBIA POR MEDIO DE LAS HERRAMIENTAS SOFTWARE IBM CARE
MANAGEMENT Y CÚRAM, PARA LA GESTIÓN DEL CUIDADO PREVENTIVO
Y TRATAMIENTO A PACIENTES SANOS O IDENTIFICADOS CON
ENFERMEDADES CRÓNICAS.**

SANTIAGO NIÑO PARRA

**Trabajo de grado presentado como requisito para optar al título de
Ingeniero de sistemas**

Director

CARLOS ALFONSO BELTRAN CASTRO

Ingeniero de Sistemas

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMECHANICAS
ESCUELA DE INGENIERIA DE SISTEMAS E INFORMATICA
BUCARAMANGA
2017**

DEDICATORIA

Niño y Parra, mi familia por igual, al haberme dedicado su apoyo incondicional y amor a través del tiempo en mi formación, dedico este esfuerzo a ustedes.

A mis amigos, los de siempre.

IBM Colombia por darme la oportunidad de dar el primer paso al mundo profesional.

AGRADECIMIENTOS

A mi hermana Dorian Viviana Niño Parra. A mis tías, Myriam Yolanda Parra, Xenia Niño y Johana Niño, mi gratitud infinita por parte fundamental por darme la oportunidad de crecer como profesional.

A mi tía Ligia Parra por toda su colaboración.

Mi director de proyecto Carlos Adolfo Beltrán Castro, por su apoyo, acompañamiento y acogida.

A Maurisabed Hernández y Mario Alonso Mateus Gómez, dos grandes mentores que me otorgaron enseñanzas muy valiosas para desenvolverme en el mundo profesional.

CONTENIDO

	Pág
INTRODUCCION	16
1 JUSTIFICACIÓN	17
2. OBJETIVOS	18
2.1 OBJETIVO GENERAL	18
2.1.1 Objetivos específicos	18
3. DESCRIPCION DE LA EMPRESA	19
3.1 NOMBRE	19
3.2 ANTECEDENTES	19
3.3 RESEÑA HISTORICA	19
3.4 MISIÓN DE LA EMPRESA	20
3.5 VISIÓN DE LA EMPRESA	20
3.6 EMPRESAS ASOCIADAS	21
3.6.1 Grupo Bolívar (Seguros Bolívar).	21
3.6.2 Compensar	21
4. MARCO TEÓRICO	22
4.1 IBM CARE MANAGEMENT	22
4.2 IBM CÚRAM	22
4.2.1 Tecnología propia del software IBM Cúram	22
• Interfaz de usuario	23
• Lógica de negocio/ Lógica de funcionalidad	23
• Almacenamiento en la base de datos.	23

4.2.2	Tecnologías asociadas IBM Cúram	23
	• Apache ANT	24
	• Tomcat.	24
	• Eclipse	24
	• IBM DB2	24
	• IBM Rational Software Architect (RSA)	24
	• WebSphere	25
4.2.3	Ciclo de desarrollos a partir del software Cúram.	25
	• Modelamiento	25
	- Entidades	25
	- Fachadas	26
	- Estructuras	26
	- Dominios	27
	• Construcción	27
	• Codificación	27
	- Desarrollo de funcionalidad	27
	- Desarrollo de interfaz gráfica de usuario	27
	• Pruebas	29
4.2.4	Procesos adicionales al ciclo de desarrollo Cúram	29
	- Despliegues	29
	• Tecnología para despliegues en EPS compensar	29
	• Tecnología para despliegues en Seguros Bolívar	30
	• Tecnologías comunes para la realización de despliegues	30
	• Proceso de despliegue	30
5.	METODOLOGIA	32

5.1	CARACTERÍSTICAS	32
5.2	FASES DE LA METODOLOGÍA	32
5.2.1	Planificar (Plan	32
5.2.2	Hacer (Do)	33
5.2.3	Controlar o Verificar (Check)	33
5.2.4	Actuar (Act)	33
6.	INFORME DE INSTALACIÓN LOCAL DEL ENTORNO DE DESARROLLO	34
6.1	PROCESOS DE INSTALACIÓN	35
6.1.1.	Instalación y configuración de apache ANT.	35
6.1.2	Instalación de la herramienta IBM db2 asociada a la base de datos de IBM Cúram	35
6.1.3	Instalación y configuración para los productos asociados al software Cúram	37
6.1.4	Instalación y configuración de java: Java development kit	38
6.1.5	Instalación y configuración de apache tomcat	38
6.1.6	Instalación y configuración de Rational Software Architect (RSA)	39
6.1.7	Instalación y configuración del entorno eclipse	40
6.1.8	Configuración de WebSphere	40
7.	DEMOSTRACIÓN DE LA FUNCIONALIDAD DE VISTA 360	41
7.1.	VISTA 360	41
7.2	PROCESOS PRELIMINARES	42

7.2.1	Adecuación de máquina virtual	42
7.3	PROCESOS ASOCIADOS A LA CREACIÓN DE VISTA 360	42
7.3.1	Creación de la fachada de servicios	42
7.3.2	Llamada de servicios	43
7.3.3	Integración de la vista 360 para interfaz gráfica	44
7.3.4	Configuración de interfaz gráfica	45
7.3.5	Configuración del proceso de carga	45
7.4	PROCESO DE DEMOSTRACIÓN	45
8.	SEGUROS BOLÍVAR	47
8.1	REPORTE DE MANTENIMIENTO	47
8.2.	MANTENIMIENTOS FORMALES EN SOFTWARE CÚRAM PARA SEGUROS BOLÍVAR	47
8.2.1	Criterios fecha inicio – fecha fin	47
	- Desarrollo del mantenimiento	48
8.2.2	Validación de fecha en la creación de caso	49
	- Desarrollo del mantenimiento	50
8.2.3	Orden de fecha de registro de caso	51
	- Desarrollo del mantenimiento	51
8.2.4	Campos Obligatorios	52
	- Desarrollo del mantenimiento	52
8.2.5	Búsqueda por documento	53
	- Desarrollo del mantenimiento	53
8.3	MANTENIMIENTOS NO FORMALES EN SOFTWARE	

	CÚRAM PARA SEGUROS BOLÍVAR	54
8.3.1	Desarrollo del mantenimiento	54
8.3.2	Detalles del caso	55
	- Desarrollo del mantenimiento	56
9.	EPS COMPENSAR	57
9.1	FLUJO DE ACTIVIDADES	57
9.1.1	Análisis de caso de uso	57
9.1.2	Actualización de proyectos en entorno remoto	58
9.1.3	Creación del modelo	58
9.1.4	Configuración de la lógica de funcionalidades	58
9.1.5	Configuración de la capa de presentación	59
9.1.6	Pruebas	59
9.1.7	Construcción de scripts para la actualización de la base de datos en el ambiente remoto y establecimiento de seguridad	59
9.1.8	Despliegue	60
9.2	REPORTE DE INGRESO DE USUARIOS	60
9.2.1	Creación del Modelo	61
9.2.2	Configuración de la lógica de funcionalidades	61
9.2.3	Configuración de la interfaz de usuario	62
9.3	REPORTE DE NOTIFICACIONES	62
9.3.1	Análisis de caso de uso	63
9.3.2	Creación del modelo	63
9.3.3	Configuración de la lógica de funcionalidades	64

9.3.4	Configuración de la interfaz de usuario	65
9.3.5	Resultados	65
9.4	INVESTIGACIÓN DE LA FUNCIONALIDAD DE NOTIFICACIONES	66
9.4.1	Objetivos de la investigación	66
9.4.2	Implicaciones en el modelo	67
9.4.3	Implicaciones con respecto a la interfaz de usuario y lógica de la funcionalidad.	67
9.5	REAPERTURA DE CASO CERRADO MANUALMENTE	68
9.5.1	Análisis de casos de uso	69
9.5.2	Creación del modelo	70
9.5.3	Configuración de la lógica de funcionalidades	70
9.5.4	Configuración de la capa de presentación	71
9.5.5	Resultados	71
9.6	AMPLIACIÓN DE LA FUNCIONALIDAD “ASIGNAR CASO”	72
9.6.1	Análisis de caso de uso	73
9.6.2	Configuración del modelo	73
9.6.3	Configuración de la lógica de la funcionalidad	74
9.6.4	Configuración de interfaz de usuario	75
9.7	BÚSQUEDA DE TAREAS	75
9.7.1	Análisis de casos de uso	75
9.7.2	Modificación del modelo	76

9.7.3	Modificación en la lógica de la funcionalidad.	76
9.7.4	Modificación de la interfaz de usuario	76
9.7.5	Resultados	77
9.8	SOLUCIÓN A LA PROBLEMÁTICA ANTE LA CARGA DE PACIENTES EN UNIDAD BÁSICA CON RIESGO ASOCIADOS	77
9.8.1	Desarrollo de la investigación y solución	77
10.	CONCLUSIONES	79
	BIBLIOGRAFIA	80

LISTA DE FIGURAS

		Pág.
Figura 1	Logo IBM Colombia	19
Figura 2	Tecnología del software IBM Cúram	22
Figura 3	Ciclo Básico de desarrollo con el software Cúram	25
Figura 4	Estructura de una entidad en el modelo	26
Figura 5	Estructura de una fachada en el modelo	26
Figura 6	Estructura en el modelo	27
Figura 7	Entorno del Trabajo en IBM Cúram	28
Figura 8	Metodología PDCA	32
Figura 9	Formato de datos provenientes de lector de pulso	42
Figura 10	Vista 360 para el paciente registrado	46
Figura 11	Comportamiento de validación con expedición de alerta	49
Figura 12	Validación en la creación del caso	51
Figura 13	Campos obligatorios presentes en el módulo de “creación de casos”	53
Figura 14	“Búsqueda de casos” con hipervínculo incorporado hacia “detalles de caso” a partir del número de caso	56
Figura 15	Módulo de generación de reporte de notificaciones	66

Figura 16	Estructura de la entidad asociada a las notificaciones.	67
Figura 17	La re-apertura de casos	71
Figura 18	Lista producida por la búsqueda de casos	72
Figura 19	Módulo de búsqueda de tareas luego de la intervención	77

RESUMEN

TITULO: Práctica empresarial en IBM de Colombia: desarrollo, implementación y soporte de soluciones para el sector salud en Colombia por medio de las herramientas software IBM Care Management y Cúram, para la gestión del cuidado preventivo y tratamiento a pacientes sanos o identificados con enfermedades crónicas.*

AUTOR: Santiago Niño Parra**

PALABRAS CLAVE: IBM, IBM Cúram, solución

DESCRIPCIÓN

En años recientes, Business International Machine en Colombia fijó su atención en el sistema de salud actual colombiano y todos los procesos referentes a dicho sistema. Con el ánimo de poder realizar un cambio adecuado de enfoque, en lo respecta a un cuidado preventivo, y la optimización de procesos inherentes a los entes en salud, IBM realizó la apertura del proyecto del Centro de Innovación y de Excelencia en el departamento de Santander, en el municipio de Floridablanca, como evidencia de la apuesta hacia dicha industria, contando con software especializado en el área social y de la salud como IBM Care Management y Cúram.

Siendo los orígenes de este centro, los proyectos emprendidos por el equipo de practicantes que surgieron esencialmente de requerimientos y mantenimientos que migraron desde las divisiones especializadas en salud para EPS Compensar y Seguros Bolívar: centros de salud, prevención y seguros respectivamente, que contrataron los servicios de IBM Colombia en Bogotá.

En este orden de ideas, durante el desarrollo de la práctica, se realizó la instalación particularmente de la solución IBM Cúram; el desarrollo de una demostración sobre el funcionamiento de IBM Cúram y posteriormente el análisis de problemáticas y/o mantenimientos concernientes al software presente al interior de las empresas aliadas.

* Trabajo de grado

** Facultad de Ingenierías Físico-mecánicas, Escuela de Ingeniería de Sistemas. Director: Carlos Adolfo Beltrán Castro.

ABSTRACT

TITLE: Business Practice at IBM Colombia: development, implementation and support solutions for the health sector in Colombia through IBM Care Management and Cúram of preventive care and treatment to healthy patients or identified with chronic diseases.*

AUTHOR: Santiago Niño Parra**

KEYWORDS: IBM, IBM CÚRAM, Solution

DESCRIPTION

In the last recent years, International Business Machine IBM has focused its attention in the Colombian Healthcare System and all current processes related to that particular system. In the interest of changing the approach in the field of Preventive Care, and also to optimize the processes within the health agencies, IBM created the Excellence and Innovation Center, in the Department of Santander, in the municipality of Floridablanca. This project is a bet in the industry which involves the deployment of specialized software for social and healthcare applications, such as IBM Care Management and Cúram.

At the early stage of the Center formation, the research worked within the staff of interns, were derived essentially from requirements and maintenance needs that migrated from the specialized health divisions of EPS Compensar and Seguros Bolivar: health, prevention and insurance centers respectively, which hired IBM services in Bogota, Colombia.

During the development of the internship, the IBM Software Solution, Cúram, was installed and later demonstrated. Subsequently, the analysis of derivate incidents and / or software maintenance within each company involved was performed

* Degree work

** Faculty of Physic mechanical Engineering, School of Engineering Systems. Director: Carlos Adolfo Beltrán Castro.

INTRODUCCIÓN

IBM trazó el objetivo de posesionarse como fuente de tecnología software para el sistema de salud colombiano. La adaptación de la ciudadanía a estos cambios, el proceso de creación de nuevas infraestructuras físicas y tecnológicas, así como las regulaciones legales y administrativas del sistema de salud, han conllevado a múltiples retos en esta iniciativa.

El ánimo de la multinacional en la innovación dentro de dicho mercado, ha derivado en el desarrollo de IBM Care Management, conjuntamente con la adquisición de Cúram y diversas otras tecnologías de la misma corriente, originando el portafolio de productos para el sector salud a nivel nacional, contando con una gama de módulos y soluciones adaptables al mercado y al cliente del sector salud.

Para cada una de las empresas afiliadas actualmente a la tecnología de IBM, como es EPS Compensar y Seguros Bolívar en la ciudad de Bogotá, la necesidad de la adaptación de los productos contratados se ha constituido en un asunto imperioso que ha llevado a la generación de los desarrollos adecuados y pertinentes a las actividades de prevención de la enfermedad y cuidado de la salud de los usuarios de estas entidades.

La importancia de esta práctica se sustenta en la responsabilidad y el compromiso adquiridos con el continuo crecimiento de los productos, a partir de la interpretación y traducción en lenguaje tecnológico de los requerimientos de los clientes de IBM. De esta manera, se ve reflejada en la implementación de múltiples módulos funcionales, y el compromiso con su mantenimiento diligente, eficiente y continuo.

1. JUSTIFICACIÓN

Actualmente, el sistema de salud ha estado en medio de cuestionamientos en torno a la prestación de sus servicios. El Sistema General de Seguridad Social en Salud (SGSSS), correspondiente a la ley 100 de 1993, atraviesa por dificultades que ha costado al país la confianza de su gente, la vida de decenas por déficit de prestación del servicio; además, pérdidas millonarias ante el mal manejo de las políticas de salud. Todos estos escenarios se han venido en el marco de las siguientes problemáticas:

- Flujo de información (de datos no estructurados, en su mayoría) erróneamente manejados.
- Desconocimiento del estado y de la misma población sobre el estado actual del sistema de salud colombiano, dificultando transiciones en entidades prestadoras de salud (EPS), sectorización para ser atendidos en un nueva IPS, y cambios sobre su estatus dentro del sistema (Cotizante a beneficiario y viceversa)
- Fragmentaciones de funciones en los actores de salud.
- Sistema de salud aún con falencias en el alcance y eficacia del enfoque preventivo.
- Escasez de sistemas adecuados sin manejo especializado de la información que deriva en problemas de inter operatividad entre entidades de salud y de reacción inmediata.

De esta manera, se hace necesario implementar soluciones; por eso IBM ha tratado de contribuir ante las dificultades plenamente identificadas con el fin de cambiar el paradigma que se ha venido estableciendo en el territorio de salud colombiano.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

Realización de soporte hacia la empresa Seguros Bolívar y Compensar y desarrollar e implementar prototipos enfocados a programas de salud preventiva, con el propósito hacer parte de la producción de productos, reconocimiento y promoción de tecnologías presentes en la empresa International Business Machine Colombia

2.1.1 Objetivos específicos

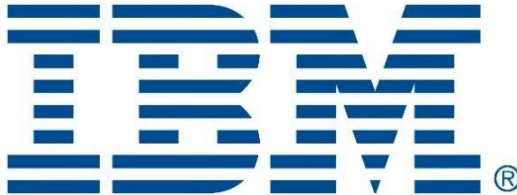
- Realizar un informe en cual se evidencie el proceso de instalación local del entorno IBM Care Management, con énfasis en la solución IBM Cúram, en IBM Rational Software Architect (RSA) y IBM WebSphere y herramientas externas inherentes, para la construcción de prototipos y/o actualización de componentes, modelos a partir de variables clínicas y realización de pruebas.
- Realizar prototipo denominado “Vista 360” a partir del software Rational Software Architect, de lenguajes asociados a la programación orientada a objetos y programación orientada a la web, para el Modelamiento y programación de funcionalidad de variables clínicas provenientes de sensores de pulso y temperatura ojalados en un microcontrolador arduino. Adicional:
 - a. Programación de servicio web REST inbound e implementación.
 - b. Acoplamiento con la plataforma IBM Cúram.
 - c. Modelamiento e implementación de servicio batch (cargue de datos de los pacientes) para validación.Con el objetivo de la visualización en tiempo real y almacenamiento de las variables clínicas de los pacientes para su respectivo análisis por el personal especializado.
- Realizar soporte de mantenimiento ante errores reportados en la plataforma web Mantis bug tracker: Care Managment para la empresa Seguros Bolívar y para la EPS Compensar que implica la corrección, actualización de modelos clínicos, servicios web y consultas a bases de datos y/o creación de componentes en el producto IBM Cúram.

3. DESCRIPCIÓN DE LA EMPRESA

3.1 NOMBRE

IBM COLOMBIA

Figura N° 1 Logo IBM Colombia



3.2 ANTECEDENTES

IBM, originaria de los Estados Unidos y como empresa multinacional, abrió sus puertas en el territorio colombiano en el año de 1973. Actualmente en el país, esta empresa se ha encargado de tomar las riendas en diversos sectores, abordando el mercado con amplia gama de productos en el área de producción, distribución, comercio y banca a partir de la integración tecnología de desarrollo.

El reconocimiento que posee IBM se debe a su gestión y competitividad a través del país. Ofreciendo transformación de los sistemas presentes en el país, generando como resultado procesos más inteligentes, seguros y eficientes.

3.3 RESEÑA HISTÓRICA

Tabulating Machine, fue el nombre al cual fue otorgado a la pequeña empresa originada por Herman Hollerith, creador de un máquina perforadora en los Estados Unidos que derivó, no solamente en un creciente éxito, sino, en consecuencia, a la fusión de distintas compañías de la misma corriente en el mercado con su empresa.

Fue entonces en 1924 donde las riendas de la compañía fueron asumidas por Thomas J. Watson. International Business Machine (IBM) fue la denominación asumida en esa época en la compañía que, básicamente, se encargaba de la venta de máquinas industriales de diversa índole.

Dicha herramienta, pionera de grandes inventos que comenzaron la revolución del mercado hasta la década de 1960, en el cual la producción de ordenadores, inicialmente, para fines empresariales mutó los objetivos hasta un trabajo con alta relevancia en IBM hasta 1981 donde fue presentado el primer ordenador personal.

A partir del éxito de estos ordenadores, fue inevitable la expansión de la oferta por parte de IBM para llegar a originarse el desarrollo de software que terminó siendo la opción insignia que marcó el derrotero del futuro de la empresa. Actualmente, IBM se ha convertido en la multinacional referente en cuanto a fabricación de tecnología cognitiva.

3.4 MISIÓN DE LA EMPRESA

La misión de IBM gira en torno a objetivo de poder proveer a los clientes servicios y soluciones con valor agregado: innovación. La característica fundamental de todo producto es precisamente ser un ícono de la diferencia entre los servicios actuales con miras a que los clientes puedan cumplir sus metas dentro del mundo corporativo.

3.5 VISIÓN DE LA EMPRESA

IBM quiere caracterizarse a nivel mundial por la innovación de los productos y servicios con los que se consiga un impacto en el mejoramiento del estilo de vida de la población mundial, a partir de servicios cognitivos que puedan trazar los lineamientos de procesos inteligentes que estimulen y promuevan la evolución de la tecnología mundial al servicio del hombre.

3.6 EMPRESAS ASOCIADAS

En este apartado se describen las empresas dedicadas al sector salud al cual, a partir del presente convenio con el empresa International Business Machine IBM, se construyeron los desarrollos en la práctica empresarial.

3.6.1 Grupo Bolívar (Seguros Bolívar). Grupo Bolívar es una empresa multidisciplinaria que cuenta con representaciones en los sectores financiero, de aseguramiento y de construcción, entre otros. El sector de aseguramiento viene representado por Seguros Bolívar, entidad que se constituye en la segunda mayor aseguradora de dicho mercado en Colombia.

3.6.1 Compensar. Compensar es una caja de compensación que presta sus servicios en la ciudad de Bogotá, Cundinamarca, donde es reconocida por su amplio campo de servicio en la atención de sus afiliados en materia de salud, educación, crédito y recreación. El primero de ellos se ve materializado en la actual EPS Compensar.

4. MARCO TEÓRICO

4.1 IBM CARE MANAGEMENT

IBM Care Management es un conjunto de tecnologías gestionadas por IBM especializadas para escenarios sociales desfavorables con el objeto de proporcionar gestión de atención integral. Dentro IBM Care Management se encuentra la solución Cúram que a continuación se comentará.

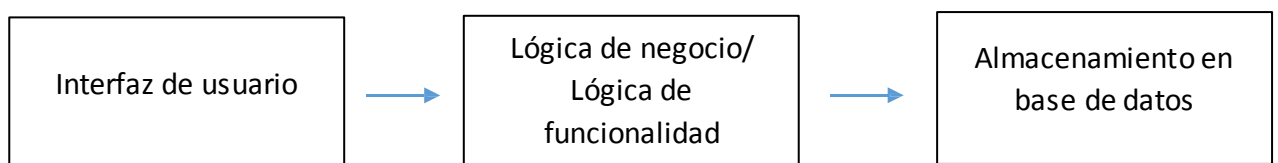
4.2 IBM CÚRAM

Cúram se traduce en el idioma irlandés “cuidado”. Provee soluciones relativas al sector salud, estructurada por medio de una base de funcionalidades pre configuradas y con componentes que brindan servicios específicos según sea el escenario relativo a las problemáticas sociales y de salud. Cúram fue el producto central dentro de la práctica empresarial.

Cúram se caracteriza por una arquitectura escalable que provee flexibilidad en la implementación de aplicaciones, con rendimiento capaz de soportar diversos lenguajes que actúan en comunión con el fin de implementar y desplegar procesos que se plantean.

4.2.1 Tecnología propia del software IBM Cúram. Se comprende de tres capas:

Figura N°2. Tecnología del software IBM Cúram



- **Interfaz de usuario.** El usuario interactúa con un navegador web en este sentido. Por lo cual la comunicación se resume en solicitudes HTTP hacia el servidor web, presente en la tecnología Cúram, que responde con páginas intuitivas a base del lenguaje HTML.

La interfaz de usuario es una aplicación web de java. Esto involucra el uso de Java Server Pages (JSP), servlets, y patrones de diseños como lo es modelo-vista-controlador.

- **Lógica de negocio/ Lógica de funcionalidad.** IBM Cúram utiliza la estructura software configurable JavaBeans, la base de codificación para consultas manuales Java Database Connectivity y lógica de procesos estándar Java. A partir de esta tecnología se encuentra el soporte de toda transacción.

A partir de uso de dicha tecnología java, permite al desarrollador de crear la lógica necesaria en lo que respecta a los asuntos empresariales del cliente. En lo que respecta a la aplicación lógica de negocio, se despliega a partir de un servidor de aplicaciones apropiado Java Enterprise Edition, como lo es IBM WebSphere.

Fundamentalmente existen tres objetos en esta capa:

- Fachada: Corresponde a los aspectos lógicos visibles para los usuarios. Contiene la responsabilidad de realizar transacciones.

- Objetos de proceso: Se encarga de la comunicación individual de componentes o tablas de la base de datos con respecto a aspectos lógicos.

- Objetos de entidad: Como bien lo indica, entidades, que se encargan de representar lógicamente un aspecto del modelo de datos o particularmente una tabla de la base de datos.

- **Almacenamiento en la base de datos.** Corresponde a la capa que permite la persistencia de ciertos objetos en el modelo de datos.

4.2.2 Tecnologías asociadas IBM Cúram. IBM Cúram propiamente no contiene entornos de desarrollo integrado para la creación de nuevos componentes o edición de ellos. Esta solución utiliza ciertas herramientas para el funcionamiento de dicha tecnología

- **Apache ANT.** “Apache ANT es una biblioteca y líneas de comandos proveniente del lenguaje Java, cuya misión es conducir procesos: el principal uso conocido de ANT es la acumulación de aplicaciones Java. ANT suministra una serie de tareas integradas que permiten compilar, ensamblar, probar y ejecutar aplicaciones Java”¹ Apache ANT una herramienta fundamental al permitir la construcción de desarrollos con respecto a aspecto pre-configurados del producto Cúram.
- **Tomcat.** Es un software de implementación de código abierto ligado a tecnología Java Servlets. La presencia de tomcat ayuda a la ejecución de procesos dentro de servidores prestos para la realización diversas solicitudes ejecutadas por el usuario. Tomcat mantiene la seguridad y diferencia entre funciones de usuarios finales para el acceso y ejecución de solicitudes.
- **Eclipse.** Eclipse es una plataforma de código abierto basada en Java que adicionalmente permite la extensión de sus funcionalidades propias como entorno de desarrollo. De esta manera, Eclipse acopla tanto a Apache ANT, como tomcat para el funcionamiento correcto de la tecnología Cúram. Por lo cual, resulta un entorno de desarrollo propio de las necesidades de este producto para la creación o edición de desarrollos.
- **IBM DB2.** IBM DB2 es una familia de productos de sistema de gestión que funge como herramienta fundamental para IBM Cúram en el aspecto del modelo de datos.
- **IBM Rational Software Architect (RSA).** “Es una completa herramienta de diseño, modelado y desarrollo para la entrega de software de extremo a extremo. Utiliza lenguaje de modelado unificado (UML) para diseñar aplicaciones Java de empresa y servicios web. Rational Software Architect se basa en la infraestructura de software de código abierto Eclipse y se puede ampliar con plug-ins.”²

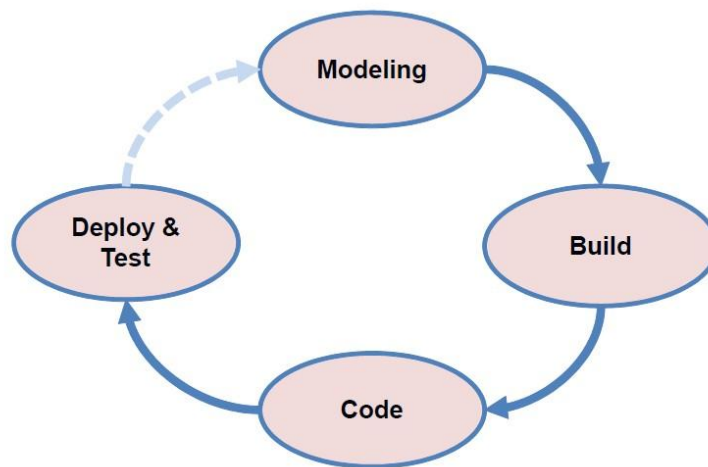
¹ Another Neat Tool: Apache ANT [en línea]. <<http://ant.apache.org/>> [citado en 1 de abril 2017].

² IBM Rational Software Architect [en línea]. <<http://www-03.ibm.com/software/products/es/ratsadesigner>> [citado en 1 de abril 2017].

- **WebSphere.** Es un conjunto de herramientas que funcionan de manera transparente para el usuario y que permite a los clientes crear y gestionar sitios web empresariales sofisticados. Tiene un papel fundamental al ser el gestor de la aplicación IBM Cúram, lo que permite el acceso de la plataforma utilizando su estructura al ser “sincronizada” con esta herramienta, utilizando las estructuras de clases, servicios web programados, acceso a consultas a bases de datos, además de otorgarle seguridad a cada actividad que se realiza.

4.2.3. Ciclo de desarrollos a partir del software Cúram.

Figura N° 3. Ciclo Básico de desarrollo con el software Cúram

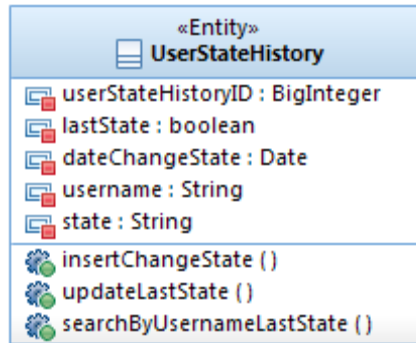


- **Modelamiento.** El modelamiento se entiende como la definición de artefactos para la obtención de la base del código para la creación de desarrollos. Los artefactos que se mencionan, se refieren a herramientas propias de IBM Rational Software Architect, en las que pueden ser de los siguientes tipos:

- **Entidades.** Son objetos exclusivos para la creación de campos persistentes como parte de la estructura de datos de IBM Cúram donde es posible configurar operaciones básicas de lectura, escritura, eliminación y sobre-escritura a partir de las llaves primarias de las tablas o sobre atributos particulares.

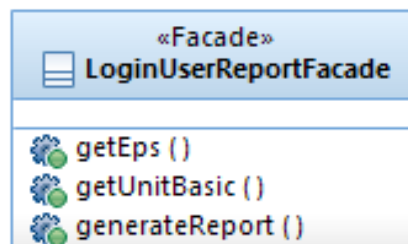
El uso de las operaciones configuradas en la fase de modelamiento, consisten en la instanciación de dicha entidad y la invocación de las operaciones que maneja utilizando parámetros adecuados.

Figura N° 4. Estructura de una entidad en el modelo



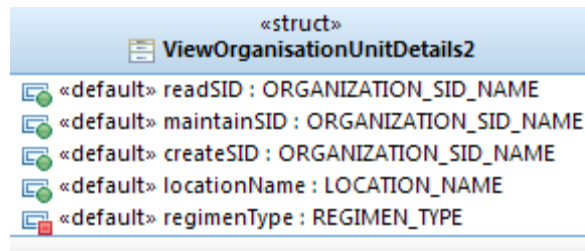
- **Fachadas.** El modelamiento de este artefacto, permite la creación automática de la estructura propia del patrón de diseño asociada a este mismo nombre. Por su puesto, se puede asociar en este punto las operaciones o lo que en equivalencia se interpreta como los métodos que harán parte de la clase. Dentro de estas operaciones se permite la asociación de datos o conjunto de datos a partir del siguiente artefacto que se describirá.

Figura N° 5. Estructura de una fachada en el modelo



- **Estructuras.** Corresponde a una unidad que puede contener un conjunto de datos para que puedan utilizarse como parámetros de entrada o salida en operaciones declaradas. La construcción de las estructuras se fundamenta en parámetros que son declarados para estas estos artefactos. Los parámetros son datos individuales. El tipo de dato es declarado a partir de su dominio.

Figura Nº 6. Estructura en el modelo



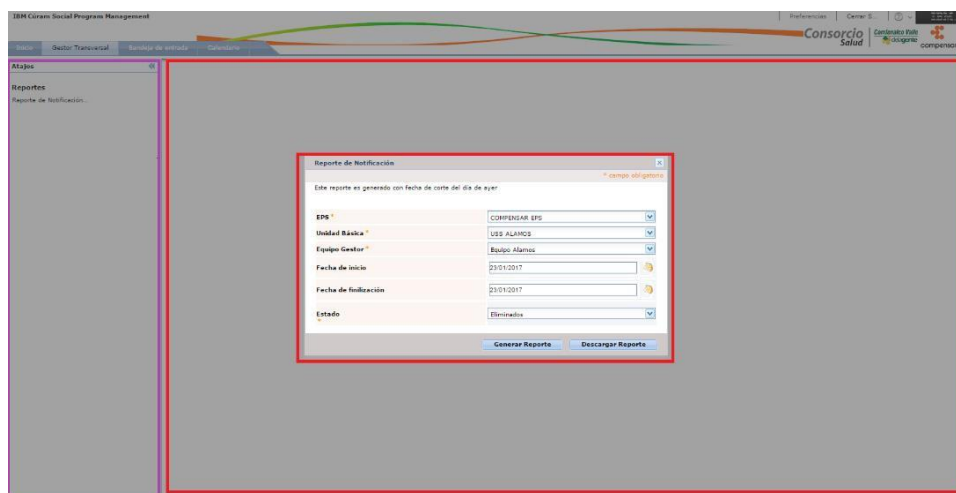
- **Dominios.** Se especifica un tipo de dato, con un nombre particular que posteriormente se asocia a un parámetro que haya sido vinculado a una estructura.
- **Construcción.** La tecnología Cúram cuenta con líneas de comandos específicos ejecutados en el entorno de la consola del sistema operativo Windows que construye automáticamente las base propiamente de su arquitectura: Crea archivos que corresponden a las clases especificadas en el modelo, edición e inclusión de nuevos campos y operaciones en el modelo de datos, como también el establecimiento de la interfaz de usuario con estilos pre-configurados.
- **Codificación.** Esta fase corresponde al proceso de programación de los desarrollos, orientando al funcionamiento esperado del desarrollo propuesto a partir de modelo construido. El proceso de construcción, en síntesis, permite que el programador este dedicado exclusivamente a los asuntos del desarrollo de funcionalidad y la programación del diseño de la interfaz gráfica. En este orden de ideas, la codificación está dividida en dos partes:
 - **Desarrollo de funcionalidad.** Se trataría de la codificación de la lógica que se realiza a partir del lenguaje java por medio de entorno de desarrollo eclipse a partir la configuración preliminar que se estableció a partir del modelo.
 - **Desarrollo de interfaz gráfica de usuario.** El desarrollador se encarga de esta faceta partir del lenguaje XML que puede ser manejado partir de cualquier

editor de texto mientras se especifica la extensión propia de estos archivos UIM, User Interface Metadata, que corresponde a un solución de diseño web con estilos característicos para este escenario.

Adicionalmente, el desarrollador debe crear las “propiedades” para cada componente de la interfaz gráfica; las propiedades en el software que se debió modificar. Es un concepto abierto para lenguaje XML que se refiere a un conjunto de archivos especializados en la declaración de variables en los cuales, se le asigna un texto que se verá reflejado en la vista de usuario.

Cabe destacar que el software Cúram maneja un orden estricto para cada componente de la capa de presentación del software: Paneles de opciones (Recuadro púrpura) y áreas de trabajo (Recuadros rojos) manejan archivos de propiedades por aparte. Esto se debe a que los paneles de opciones varían dependiendo de los roles que el software maneja.

Figura N° 7. Entorno de trabajo Cúram para EPS Compensar



Por otro lado, la tecnología Cúram maneja archivos especializados basados en el lenguaje XML para la comunicación entre la capa de presentación de usuarios con la lógica programa que influye, por su puesto, en la acción de varios artefactos, generalmente botones, que puede pertenecer al módulo. No obstante, esta clase de archivos, al ser especializados en el sentido estricto de la palabra, influyen en acciones específicas.

- **Pruebas.** Las pruebas, básicamente, parten de la creación de usuarios, casos y actividades, por medio del entorno local de Cúram, suministrando datos ficticios al sistema que serán alojados en la base de datos para probar el comportamiento del sistema, generalmente, haciendo uso los depuradores del entorno eclipse.

Además, en términos de aspectos sobre consultas SQL, se tomaban en cuenta los entornos especializados para la realización de pruebas para edición de consultas analizadas en el período. Para que las pruebas pudieran ser confiables, se necesitaba el diseño de estas mismas orientadas a la aproximación de las operaciones que se realiza en la EPS Compensar y Seguros Bolívar.

4.2.4 Procesos adicionales al ciclo de desarrollo Cúram

- **Despliegues.** Hace referencia a un conjunto de actividades orientados a los cambios realizados que pretenden actualizarse dentro del servidor que aloja el ambiente de pruebas en proyectos específicos contenidos en software Cúram donde aglomeran los desarrollos propios para la evolución del producto cuyo objetivo es poder modelarse a partir de las reglas de negocio del cliente. Para la realización de dichas actividades, fue necesario contar con diversas tecnologías acordadas por IBM para cada ente asociada en producto Cúram.

- **Tecnología para despliegues en EPS compensar**

- **Git hub**

Se trata de un software que asume la administración de versiones en proyectos alojados dentro de su dominio, particularmente asociado a la plataforma IBM Bluemix, presto para el manejo remoto de los mismos, por lo que se puede realizar la fundación de un “repositorio” si distintos usuarios realizan cambios a los proyectos en cuestión, o bien alojado de manera local si se trata de un proyecto.

Dado que Git Hub otorga facultades a los usuarios, dependiendo de su papel dentro de administración de los proyecto, es posible la actualización de proyectos de manera local a partir de los presentes en el repositorio y viceversa, la eliminación y fusión de versiones.

En términos generales, IBM Bluemix es una plataforma que provee servicios de diversa índole. En este caso, se hace aprovechamiento del convenio de Git hub y dicha plataforma para el manejo de versiones en la nube.

- **Tecnología para despliegues en Seguros Bolívar**

- **Apache Subversión.** Es un sistema de control de versiones en el cual ofrece la posibilidad del manejo de cambios en los proyectos contenidos en el repositorio.
- **TortoiseSVN.** TortoiseSVN corresponde a un software de código abierto, asociado a Apache Subversión, en el cual puede realizar fusiones y verificadores de versiones: También, el manejo de actualizaciones, descargas de archivos a partir de los presentes en los servidores remotos que alojan los proyectos.

- **Tecnologías comunes para la realización de despliegues**

- **PuTTY .** Es un cliente SSH y telnet como herramienta de comunicación entre servidores que alojan los proyectos. Permite la ejecución de comandos apropiados para este proceso.
- **WinSCP.** Es una aplicación de software libre en el cual se utiliza el protocolo SFTP. De esta manera, WinSCP, ofrece comunicación con servidores remotos; adicionalmente, ofreciendo la ventaja de ser un medio gráfico, facilita procesos de carga y descarga de archivos; creación, eliminación y cambio de directorios en los ambientes de manera intuitiva.

- **Proceso de despliegue.** Luego que se realizó la actualización de los proyectos con respecto al repositorio, el procedimiento siguiente consiste en la compresión de dichos archivos para la actualización del ambiente. Esto obedece a la prevención de errores humanos que pudieran alterar de manera crítica la configuración actual de los servidores, dado que posteriormente se realiza la sobre- escritura de proyectos, trasladando dichos archivos desde la máquina local al servidor en el directorio apropiado: particularmente donde se realizaron nuevos desarrollos con respecto a las antiguas versiones del producto Cúram.

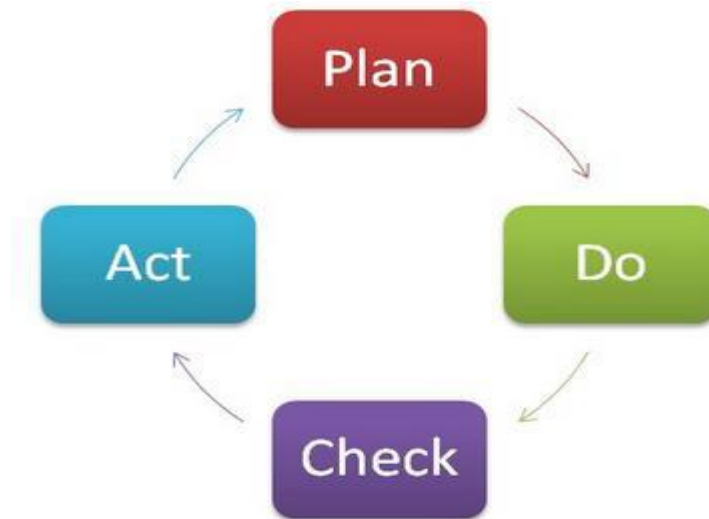
Aunque, por definición, el despliegue básicamente otorga la posibilidad de la actualización de producto Cúram, específicamente de los nuevos desarrollos y/o

mantenimientos de este mismo software en los proyectos intervenidos, el simple proceso de sobre-escritura de los proyectos en el ambiente de pruebas, no supone cambios tangibles dentro del producto.

Por tanto, los pasos fundamentales siguientes al proceso se encuentran en la ejecución de comandos contenidos dentro del servidor en el cual se construye toda la configuración presente, tanto la lógica del producto como los aspectos propios de la capa de presentación al usuario, tomando todas las clases pre configuradas, recientemente modificadas, y de la misma manera en lo que se refiere a archivos propios de la configuración web.

5. METODOLOGÍA

Figura N° 8. Metodología PDCA



5.1 CARACTERÍSTICAS

Esta metodología describe los cuatro pasos esenciales que se deben llevar a cabo de forma sistemática para lograr la mejora continua, entendiendo como tal al mejoramiento continuado del procesos de desarrollos de funcionalidades. La metodología PDCA implementada y adaptada basada en el trabajo conjunto con la administración de los proyectos asignados en el centro de innovación y excelencia.

5.2 FASES DE LA METODOLOGÍA

5.2.1 Planificar (Plan): Se realiza la asignación de la labor. Se abordan los conceptos inherentes al desarrollo que se requiere construir, la realización de soporte o continuidad del desarrollo llegado el caso en que aún no se haya terminado su elaboración para la siguiente revisión de avances de los proyectos, con el propósito de confeccionar el cronograma de procesos. Esto incluye correcciones del proceso para la celeridad del desarrollo, inclusión o eliminación de aspectos que hace parte de la asignación

5.2.2 Hacer (Do): Se aborda la construcción de la funcionalidad y/o mantenimiento, implementando cambios sugeridos o nuevos aspectos presentes en la revisión de avances.

5.2.3 Controlar o Verificar (Check): Una vez se implementa las correcciones sobre el proceso, nuevos aspectos o la eliminación de ellos, se realiza las pruebas pertinentes en el sistema por un periodo de tiempo asignado. Si bien el comportamiento del desarrollo necesita nuevos asuntos por contemplar se procede a la siguiente fase. De lo contrario se finaliza.

5.2.4. Actuar (Act): Por último, es necesario realizar análisis de asuntos de desarrollo no contemplados y necesarios por implementar para el cumplimiento del desarrollo.

6. INFORME DE INSTALACIÓN Y CONFIGURACIÓN LOCAL DEL ENTORNO DE DESARROLLO

En el inicio de la práctica empresarial, se realizaron asignaciones a cada estudiante en algunos campos de acción que abarcaban las tecnologías presentes en IBM: Internet de las cosas (IoT), almacenamiento en la nube y Software Cúram.

En lo que respecta a las labores asignadas (que se prologaron hasta el final de la práctica) se enfocaron hacia la solución Cúram. Dentro del material de capacitación, se encontraba un documento en que se evidenciaban los pasos para la instalación local, incluyendo las tecnologías asociadas. Sin embargo, se originaron inconveniente que originaron retrasos en el desempeño de esta labor, por no contar con conocimientos documentados suficientes.

En base a esta situación, era imperioso realizar la documentación a base de pre-saberes con énfasis en escenarios problema en la instalación y principalmente en la configuración; diagnósticos y solución para evitar inconvenientes a practicantes y desarrolladores que posteriormente realizaran la instalación.

La característica principal que comprende dicho proceso es no intuitiva. Es decir, a diferencia de los programas de software que particularmente se conocen, en los cuales el progreso de instalación no requiere de vastos conocimientos técnicos, aún en ciertos elementos con la ventaja de tener módulos para una instalación guiada, todo lo que respecta a la instalación conjunta de la herramienta implicaba procesos adicionales o imprescindible conocimiento técnico.

Más allá de una solicitud del jefe del equipo técnico, se realizaba dicho proceso de documentación en pro del conocimiento en este proceso para poder liderar la instalación más delante de este software y sus herramientas para 6 equipos más en sincronía (la cual resulta exitosa debido a los conocimientos adquiridos) preferiblemente para el comienzo de la capacitación para todos los estudiantes en Centro de Investigación y Excelencia de IBM.

El informe viene en conjunto con pre-saberes documentados por ingenieros de IBM, aspectos no incluidos previamente y empíricos de mi parte para el tratamiento de errores.

6.1 PROCESOS DE INSTALACIÓN

Este apartado está dedicado a comentar los pasos de instalación que involucran todas las tecnologías presentes para el desarrollo por medio del software Cúram e inconvenientes en cada paso del proceso.

6.1.1. Instalación y configuración de apache ANT. Apache ANT se encuentra vinculado con la máquina personal a partir de la configuración de variables locales a través de la configuración avanzada de propiedades del sistema operativo Windows. En total, involucra la declaración de dos variables en las cuales se ve especificada la ruta a directorios específicos.

Los conflictos que se originan a raíz de esta herramienta, son a partir de la sintaxis de la variable. Un error en este sentido implica la existencia de comunicación entre el sistema operativo y apache ANT.

La relevancia de la problemática subyace en inconvenientes imperceptibles de este tipo hasta el momento que se intenta ejecutar el software Cúram. Por medio de la experiencia en lo que respecta al comportamiento del software, se hace posible enfocarse en la revisión de las variables declaradas.

6.1.2 Instalación de la herramienta IBM DB2 asociada a la base de datos de IBM Cúram. La documentación previa, sugería la instalación del producto más sencillo: DB2 Express.

Constituía la herramienta externa donde más inconvenientes se habían presentado. Lo orígenes de las problemáticas se resumían en dos campos: La licencia del producto y la configuración:

- La configuración involucraba la creación de un usuario en el sistema operativo, asignando nombre y contraseña, permitiendo que la máquina local se convirtiera en servidor local. Aunque se trataba de un error de usuario, la manera en la cual se identificaba este error no era completamente intuitivo hasta la hora de la elaboración de la base de datos.

La configuración del usuario como servidor local se encontraba encriptada. De esta manera, si el usuario realizaba algún error en la configuración de la contraseña, era necesario la desinstalación del producto.

No obstante, la desinstalación del producto no eliminaba todos los archivos implicados en este producto. Además, se mantenía cierto conjunto de servicios configurados en el sistema operativo. Así, pues, era necesario el manejo comando para la desinstalación de servicios y la eliminación manual de los aspectos mencionado.

- Luego de la configuración de este producto y del software Cúram se realizaba la creación del modelo de datos. Esta configuración parte de la ejecución de comandos específicos en el entorno de Windows, que a partir de ello se realiza automáticamente por medio de la tecnología Cúram. Sin embargo, muy comúnmente el desarrollo de esta configuración se vio interrumpida casi instantáneamente; la creación del modelo de datos fallaba por razones que la misma consola de Windows no emitía particularmente en términos del inconveniente.

Siendo una falla muy difícil de diagnosticar, se necesitó un amplio período de tiempo para conocer su causa. Se encontró como el producto particularmente DB2 Express se encontraba sin licencia, contrariamente a los demás productos. Por esta razón, fue necesaria la desinstalación del producto, los procesos de eliminación inherentes, y la instalación de un producto más robusto. Se documentó dicha novedad.

- No muy distante al anterior escenario, la licencia en ciertos casos expiraba cuando ya se había hecho la correcta generación del modelo de datos. El entorno de desarrollo eclipse reportaba la imposibilidad de realizar conexión con el servidor local. Si bien era una falla fácil de diagnosticar, este escenario generaba más problemas que el anterior mencionado.

Comúnmente, ya se había trabajado con el software Cúram. Por lo cual, esto implicaba la existencia de registros y datos de prueba que eran inaccesibles al momento de la expirar la licencia. Aunque se realizaba el tratamiento adecuado, de igual manera contenía consecuencias para los desarrollos, particularmente para la generación de pruebas.

- La adecuación del modelo de datos en ciertos momentos se veía interrumpido al reportar colisiones de llaves primarias entre tablas. Aunque este escenario no era precisamente un asunto muy común, existió en la experiencia de la práctica laboral. El diagnóstico que surgió a partir de este escenario, era la pre-existencia de archivos relacionados con productos anteriormente instalados de DB2. Esto pudo haber originado naturalmente por desinstalación incompleta del producto.
- La pluralidad de usuarios en el sistema operativo Windows, era una situación que podría acarrear ciertos inconvenientes. Durante el proceso de configuración de la base de datos, generalmente se pedía la aprobación de algún usuario con privilegios administrativos. Si bien el usuario creado por el producto DB2 y el usuario principal, usuario propio que se configuró en la máquina local, compartían el mismo estatus en sistema, podría originar la confusión en la elección del usuario adecuado en ejecución de la configuración de comandos.

En la experiencia de la práctica laboral se corroboraron los efectos en los cuales se realiza la elaboración automática de la base de datos con permisos de usuario no-principal: El software Cúram puede ser ejecutado, sin embargo su entorno no muestra más que una interfaz de usuario en blanco. El diagnóstico, que naturalmente fue documentado, se realizó a partir la última idea comentada: el usuario principal no podía observar el contenido del entorno Cúram. Por lo cual, se hizo la eliminación de la base de datos y reconstrucción de esta misma con el usuario indicado.

6.1.3 Instalación y configuración para los productos asociados al software cúram. IBM Cúram, en sí, se estructura con productos que son solicitados por el cliente, como vale recordar en este apartado. Como una buena práctica que se tomó en cuenta en la construcción de la nueva documentación, fue el orden en que se instalaban los números productos. IBM Cúram Plataforma, es un aspecto

base para el software Cúram ya que contiene aspectos fundamentales pre configurados. En este orden de ideas, los productos restantes deben ser instalados posteriormente a esta base y conservadoramente como sugiere la documentación.

Estas observaciones sobre la configuración ordenada de factores para la instalación de la solución Cúram se realizaron dado que, en la experiencia de instalación, particularmente liderada, se encontraron proyectos que debían contener el producto Cúram ya instalado en pro de desarrollos; sin embargo, existía la ausencia de algunos directorios.

6.1.4 Instalación y configuración de java: Java development kit. El tratamiento de instalación y de configuración sobre java no dista con respecto a la configuración de Apache ANT. La configuración de las variables de sistema operativo es realizada orientando hacia el directorio particular de java development kit. Cabe aclarar en este punto, como se realiza también la configuración de variables en el entorno de desarrollo eclipse. Este tratamiento lo comparte Apache ANT.

En lo que respecta a esta configuración, no se encontraron observaciones para sumar a la documentación.

6.1.5 Instalación y configuración de apache tomcat. El entorno eclipse habilitado para los desarrollos en IBM, se encontraba con una asociación pre-configurada con la herramienta apache Tomcat. Este vínculo, básicamente, permitía el funcionamiento conjunto entre el entorno de desarrollo con los proyectos de esta última herramienta.

De esta manera no era necesario la realización de procesos preliminares y permitía únicamente el acople de los proyectos, aclarando en el entorno de desarrollo la versión de tomcat y configurando la ruta en la cual se encontraba el directorio de esta herramienta.

Durante la configuración del tomcat, dentro de los proyecto de esta herramienta, se hace la asociación al software Cúram, declarando las rutas apropiadas en las cuales se hace referencia a los proyectos especificados para la creación de la interfaz gráfica. Este paso fue un aspecto que en un principio se omitió, dado que en la documentación inicial no se encontraba un apartado sobre esta fase del proceso.

Si bien se encontraba ignorado, las consecuencias de realizar este paso no permitían que se desplegara el ambiente Cúram, dado que Tomcat resulta fundamental en el despliegue de la interfaz gráfica. Por esta razón, fue necesaria la inclusión de la observación en el informe de instalación.

6.1.6 Instalación y configuración de Rational Software Architect (RSA). Este software que permite el modelamiento en términos computacionales de sistemas sociales, de escenarios específicos como contextos clínicos en el caso que se ha tratado por todo el periodo de práctica empresarial, permite la configuración de paquetes individuales basados en las necesidades del desarrollo para emular dichos sistemas en modelos que posteriormente serán la base de la lógica de cada una de las representaciones.

A partir de una instalación guiada, es posible la inclusión de dichos paquetes para una configuración personalizada del producto. No obstante, la dependencia de ciertos elementos con otros que se encuentran en dicho directorio de paquetes, resultan en un inconveniente para el usuario.

En algunos casos se evidenció que la instalación no culminaba me manera satisfactoria. De esta manera, era necesario realizar la observación estricta de los paquetes adecuados para el desarrollo en el software Cúram, evitando problemas futuros en este sentido.

Por otro lado, naturalmente RSA como herramienta externa de la solución Cúram requería que se realizara el tratamiento apropiado de la conexión con los proyectos presentes en la configuración del software Cúram. Para estos propósitos, se creaba un archivo que serviría como conector entre ambas herramientas.

Aunque solamente se trata de una línea que constituía el archivo en cuestión, en algunos casos RSA resultaba con configuración evidentemente inconexa de los proyectos de Cúram. El diagnóstico que se planteó en el tratamiento de este escenario se trataba de la estricta sintaxis de los archivos en algunos casos, un espacio y la orientación de slashes (/) para la declaración del directorio, que pueden impedir el trabajo conjunto en dichas herramientas.

6.1.7 Instalación y configuración del entorno eclipse. Eclipse en sí no resulta una herramienta que necesite un tratamiento de instalación local: Se trata de simplemente de un conjunto de directorios que permiten la ejecución del entorno en la máquina.

Los aspectos importantes que giraban en torno a esta herramienta se comprendían en la situación imperiosa de la conexión de los programas externos para el desarrollo en el entorno del software Cúram en configuración del entorno de eclipse, como bien se había mencionado con respecto a Apache ANT, Tomcat y Java Development kit. La importación de los proyectos propios del software Cúram, permitía de esta manera el desarrollo a partir de dicho entorno, con el trabajo conjunto entre las herramientas interconectadas.

6.1.8 configuración de WebSphere. Representó un caso particular, dado que fue finalmente gestionado únicamente con fines de la demostración de la vista 360. Aunque la configuración que se realizó, basado en estas circunstancias, no hizo parte en el informe donde aglomeraba las tecnologías anteriormente comentadas, se realizó una consideración fundamental con respecto este software en la documentación asociada a la vista.

7. DEMOSTRACIÓN DE LA FUNCIONALIDAD DE VISTA 360

Esta demostración fue el producto de un conjunto de procesos con el fin de ser una evidencia para el público de la industria de salud sobre el funcionamiento del software Cúram, para ser identificado como un producto robusto, configurable y flexible para el ajuste basado en el funcionamiento del sistema que maneja el cliente.

Para la implementación de esta funcionalidad, se creó un lector de pulso y temperatura, a través de un procesador Arduino, en el cual registraban en tiempo real estas dos medidas. Aunque se realizaron los procesos adecuados para el tratamiento de dichas variables, se tomó la decisión de evidenciar únicamente el pulso como conjunto de datos que alimentaría a la funcionalidad.

7.1 VISTA 360

Corresponde a una funcionalidad presente en software Cúram que otorga una interfaz de usuario que permite tener acceso a diferentes tipos de información del paciente, facilitando el seguimiento o la gestión para este mismo.

Es una de las herramientas principales de usuarios profesionales en salud para tener la información completa del paciente. Esta funcionalidad se caracteriza por su dinamismo, su ágil tiempo de respuesta; además, es altamente configurable.

Por su puesto, el lector de pulso debía alimentar un registro en el tiempo en que se colocaba en marcha el dispositivo. En lo que respecta a los datos obtenidos, estos fueron alojados dentro de la plataforma IBM Bluemix en formato de texto JSON.

Figura N° 9. Formato de datos provenientes de lector de pulso



```
{
  "total_rows": 26,
  "offset": 0,
  "rows": [
    {
      "id": "002a6d74cf94dd0a2a52edbc910e702b",
      "key": "dato",
      "value": 118
    },
    {
      "id": "00d1c63fef24d90cb4b65a815c56158e",
      "key": "dato",
      "value": 118
    },
    {
      "id": "10ae7df652d8dd168c2366cf335c479c",
      "key": "dato",
      "value": 118
    },
    {
      "id": "424ac070aeec90054ad5a83f07e29727",
      "key": "dato",
      "value": 109
    },
    {
      "id": "4cff60df89826d17dd64edadca4da032",
      "key": "dato",
      "value": 105
    },
    {
      "id": "8be95756bdcadd7660960928493d6aba",
      "key": "dato",
      "value": 109
    },
    {
      "id": "8be95756bdcadd7660960928494f1ffd",
      "key": "dato",
      "value": 112
    },
    {
      "id": "a11ed2ce9f3c6e8191226c127721c7ad",
      "key": "dato",
      "value": 118
    },
    {
      "id": "a11ed2ce9f3c6e8191226c1277e1b419",
      "key": "dato",
      "value": 105
    },
    {
      "id": "a11ed2ce9f3c6e8191226c1277f2564f",
      "key": "dato",
      "value": 103
    },
    {
      "id": "a4431bfa406da576b1b5f8dbb0255dbc",
      "key": "dato",
      "value": 118
    },
    {
      "id": "c6895b195315aeed1f257fbbb74d006a",
      "key": "dato",
      "value": 109
    },
    {
      "id": "c6895b195315aeed1f257fbbb7bcdcf9",
      "key": "dato",
      "value": 118
    },
    {
      "id": "cd74a0ca92958924b9e4894b8dd103a5",
      "key": "dato",
      "value": 109
    },
    {
      "id": "e7b9d4c5b964a4892ddd35bb4cb01c1e",
      "key": "dato",
      "value": 109
    },
    {
      "id": "ea4524ec275ff934ce8a6cbe1c500641",
      "key": "dato",
      "value": 109
    },
    {
      "id": "ea4524ec275ff934ce8a6cbe1cd81039",
      "key": "dato",
      "value": 118
    },
    {
      "id": "ea4524ec275ff934ce8a6cbe1cff8186",
      "key": "dato",
      "value": 109
    },
    {
      "id": "efc05f409ba8e33384ba2c233997e9eeb",
      "key": "dato",
      "value": 109
    },
    {
      "id": "efc05f409ba8e33384ba2c2339974fff",
      "key": "dato",
      "value": 109
    }
  ]
}
```

7.2 PROCESOS PRELIMINARES

7.2.1 Adecuación de máquina virtual. El motivo por el cual se realizó una réplica del sistema operativo Windows, fue por el propósito de ofrecer al cliente la oportunidad de interactuar con el software Cúram.

Por otro lado, para realización del desarrollo, naturalmente, se tomó en cuenta la instalación de las herramientas externas propias del funcionamiento del software Cúram en el sentido mencionado. Adicionalmente, se realizó la instalación del software IBM WebSphere.

7.3 PROCESOS ASOCIADOS A LA CREACIÓN DE LA VISTA 360

7.3.1 Creación de la fachada de servicios. El punto de partida de este proceso fue la creación de dominios y estructuras apropiadas, además de la configuración y la fachada, los cuales tuvieron el papel de toma de los datos del lector de datos.

Los dominios fueron creados bajo la observación de la naturaleza de los datos provenientes de IBM Bluemix. En la imagen anterior se pudo notar como en el formato JSON existían tres datos: número único de registro de dato, el nombre y su valor. Por lo cual, dentro del modelo, se configuraron tres dominios respectivamente: uno, el cual pudiera alojar una variable tipo long, un segundo que pudiera alojar un dato tipo string y un tercero que alojara un dato tipo int. Cada uno, por su puesto, con su nombre distintivo.

El segundo paso en la configuración del modelo, fue la creación de una estructura que contuviera el número único del dato. Esta estructura fue creada con el propósito de ser los parámetros de entrada dado que, a partir de dicho dato, era posible a través de la lógica, obtener el número de identificación y a partir de ellos, la consulta a la plataforma Bluemix para que finalmente se pudiera obtener el valor del pulso.

De esta manera, también se hizo la construcción de una estructura de salida vacía. Cabe aclarar que la tecnología de modelamiento permite la creación de “lista”, tal cual como la estructura que se comentó en el cual se encontraba vacía con el propósito de, a través de la consulta, recibir los datos provenientes. La configuración de la facha “vista 360” que se modeló estuvo compuesta por un método en cual se realizaba la consulta.

7.3.2 Llamada de servicios. Esto hace referencia al conjunto de proceso transparente para el usuario en el cual se hace consumo de datos alojados en la plataforma IBM Bluemix. Dichos proceso parten de la lógica que es plasmada a partir de los métodos presentes en la facha de la vista 360 en cual contiene una función específica en la provisión de datos particulares.

Para la construcción de la lógica del método, se contó con los siguientes aspectos:

- En primer lugar se declaraba de creación de un objeto de la lista vacía que se había modelado.
- A partir del uso de parámetro de entrada asociado al método, se declara el uso de este mismo para obtener el valor de la identificación. La manera de obtener este valor es a partir de la invocación de un método creado automáticamente, dado que sólo se trata de una operación implícita “get” y que es una parte del funcionamiento de la tecnología de JavaBeans en Cúram, en el momento del modelamiento.
- Luego de este paso, se construyó una estructura try-catch dado que se haría una consulta a través de una dirección URL. Por su puesto, es apropiado realizar dicho tratamiento si bien la dirección suministrada no

contiene dato alguno o realmente no corresponde a una página apropiada. En contenido es el siguiente

- i. En lo que respecta a la comunicación entre la lógica de la funcionalidad y el entorno web, se realizaba en primer lugar el registro de dicha URL en una tabla de propiedades del software Cúram que hace parte del modelo de datos. A partir de ello, existe una clase particular en lógica del software en la cual es posible tomar dicha propiedad presente el modelo de datos, utilizar una variable en la cual representa dicha dirección y puede ser usada en cualquier clase a partir de este punto. De esta manera, se declara consulta a la URL (Bajo la sintaxis apropiada en el contexto de consulta a partir de cliente REST) como parámetro.
- ii. Como se puede recordar en el formato JSON, tal como la imagen se nota en la adecuación de los datos, era necesario realizar un tratamiento para la separación de estos mismos a partir del hecho en que, al consumir el servicio de consulta, los datos se encontraban en yuxtaposición. El motivo fundamentalmente radicaba en la necesidad de separar, obtener los datos de consulta y utilizar únicamente el valor del pulso.
- iii. Luego de la obtención de los datos del pulso, se declaró un bucle “while” para permitir que cada dato individual pudiera alojarse en la lista vacía que se declaró inicialmente.

En lo respecta a la estructura lógica de una eventual falla dentro de la consulta, se manejó una declaración de alerta al cliente cuando no se encontrara un registro o bien el URL no correspondiera a una página existente.

7.3.3 Integración de la vista 360 para interfaz gráfica. Fue necesario la creación de un archivo a partir del lenguaje XML en cual se especifica un nombre del servicio de consumo de datos por el cual se obtiene la información. Además, se declara la propiedad del software Cúram de la vista 360 propiamente creada para los datos del pulso. Por otro lado, dentro de la generación del modelo, se realiza automáticamente la creación de un “text helper” el cual, básicamente siendo una clase, permite la comunicación dentro de la lógica del software, propiamente para la toma datos de pulso, y con la plataforma IBM BlueMix.

A partir de la tecnología Cúram, la programación de los archivos apropiados de .jsp que fueron creados automáticamente permiten que la importación de los datos muestre la gráfica de acuerdo a la información.

7.3.4 Configuración de interfaz gráfica. La interfaz gráfica para el usuario debe permitir la visualización de los datos del pulso de manera gráfica. Naturalmente, para ello, era necesario realizar la programación en lenguaje XML de los parámetros de entrada. Este conjunto lo compone el número de identificación del paciente para el sistema Cúram y el valor de pulso. Por su puesto, se trata del valor que provee el consumo del servicio.

7.3.5 Configuración del proceso de carga. La elaboración del proceso de carga se estableció bajo dos aspectos:

- Modelamiento de fachada que originó una clase ejecutable que se encarga de tomar un archivo de texto que contiene un formato estándar con los datos de registro de pacientes

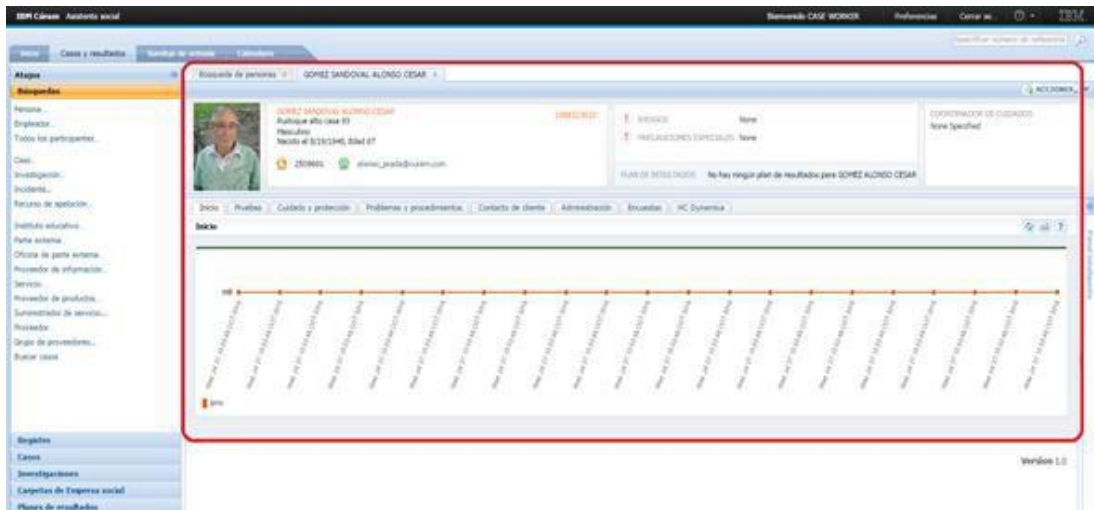
- Inherentemente a la elaboración de esta fachada de creación, dos clases más: La primera se encarga de validar la estructura del formato de registro y la segunda clase, se encarga de validar si los datos ingresados son coherentes con las reglas de negocio del sistema.

Luego de proveer al software Cúram de estas tres clases, era posible elaborar un formato de registro con el cual se creó un paciente con fines demostrativos.

7.4 PROCESO DE DEMOSTRACIÓN

El inicio de sesión se realizaba bajo un usuario de asistente social con el cual básicamente se realiza la gestión de los casos de los pacientes teniendo en sus privilegios la posibilidad de buscar la información de los mismos.

Figura N° 10. Vista 360 para el paciente registrado



La imagen que respecta al usuario registrado, fue un atributo editado para la presentación de la demostración.

8. SEGUROS BOLÍVAR

Seguros Bolívar se encuentra en los procesos de inclusión de desarrollos a partir de los aspectos acordados en base a casos de uso aprobados para la fase I y fase II del producto Cúram, con el fin de adaptarlo a las necesidades de negocio.

8.1 REPORTE DE MANTENIMIENTO

IBM tiene a disposición la plataforma Mantis bug tracker en la cual se cuenta con documentación estándar de reportes que permiten la comunicación y las asignaciones de tareas, entre usuarios activos en la empresa e involucrados a las actividades propias de Seguros Bolívar para posteriormente abordar las problemáticas, realizar los cambios apropiados dentro del software, reportar dichas modificaciones y posteriormente, desplegar los cambios realizados.

8.2 MANTENIMIENTOS FORMALES EN SOFTWARE CÚRAM PARA SEGUROS BOLÍVAR

Las labores que se realizaron para este producto, en su mayoría, fueron dedicadas a los componentes del producto Cúram “Búsqueda de casos”, “Búsqueda de personas”. Estos componentes, como sus nombres lo indican, comprenden la función del manejo de la información de las personas aseguradas y los asuntos particulares de estos clientes.

8.2.1 Criterios fecha inicio – fecha fin. En la búsqueda de casos, existen parámetros de búsqueda en la cual dicho módulo puede filtrar información en la base de datos del servidor de Seguros Bolívar a partir de una consulta estipulada en la lógica de este módulo. Entre las posibilidades de una búsqueda parametrizada, se encuentra un rango de fechas que puede utilizarse como filtro, en conjunto con otros parámetros o solamente con dichos datos.

La problemática que se presentaba recaía en el hecho de que el rango de fechas presentaba ciertas restricciones a partir de las reglas de negocio estipuladas por el

cliente. En este escenario, imperaba el mantenimiento de validaciones y/o mensajes de alerta en las búsquedas, particularmente, en dos casos:

- La fecha inicial del rango de búsqueda no podía ser posterior a la fecha final: el mensaje que se mostraba hacía el usuario era un mensaje erróneo con respecto a lo especificado por el cliente.
 - La fecha final no podía ser una posterior a la fecha actual: Era permitida la búsqueda de casos con una fecha final futura con respecto a la fecha actual.
- **Desarrollo de mantenimiento** IBM Cúram ofrece la posibilidad de la edición o bien la creación de mensajes personalizados para el cliente con el fin de lanzar alertas con respecto a acciones en la plataforma en las cuales no se puede generar alguna acción del sistema dada las restricciones de las reglas de negocio.

En términos generales, las validaciones correspondientes fueron realizadas a partir de la siguiente estructura:

```
if(...)
{
throw new ApplicationException (...);
}
```

AppException es el conjunto de excepciones pre configuradas para Cúram. No obstante. Los parámetros enviados a partir de dicha línea, otorgaban la posibilidad de buscar los mensajes disponibles de alerta para después ser desplegados por el sistema.

En primer lugar, con respecto a la validación de fecha final sobre la fecha inicial, fue necesario la identificación de la validación que se encontraba presente. En este caso, la validación (correspondiente a la condición que se presentaba comparando dichas fechas) era correcta.

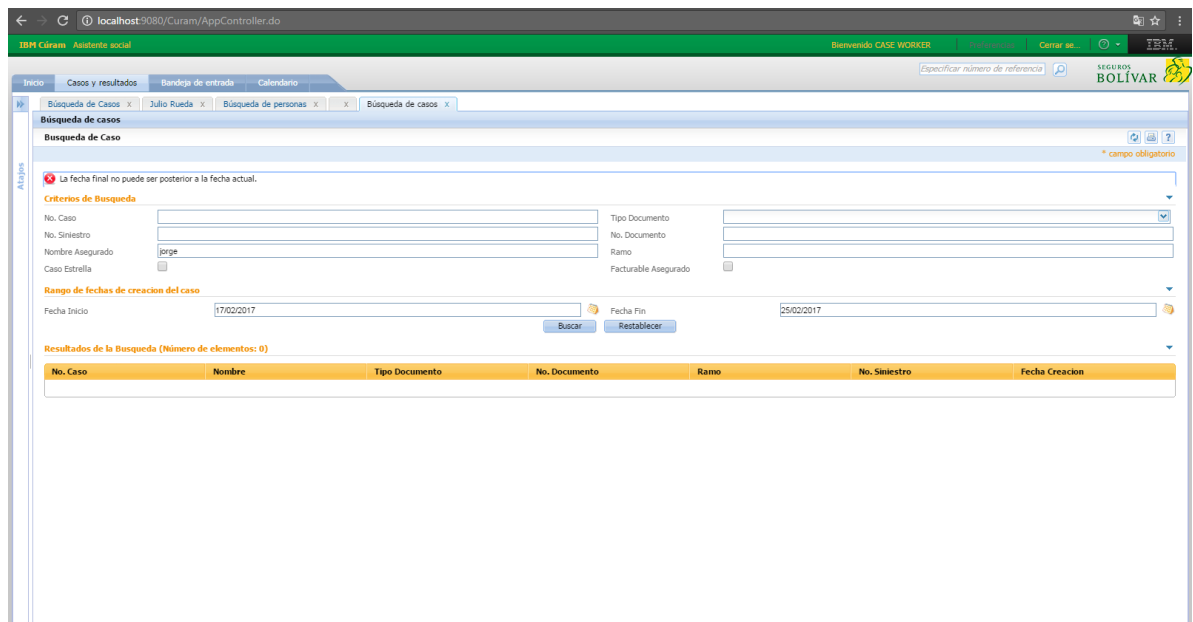
Sin embargo, el parámetro que enviaba a partir de AppException no correspondía a un mensaje alerta adecuado para dicho caso. Por lo cual, era necesario identificar dentro de la biblioteca de mensajes configurados y realizar la edición del

presente mensaje. Por otro lado, para la siguiente validación, era necesario la declaración de una variable que pudiera alojar el valor de la fecha actual, obtenida a partir del recurso de la clase java.util.Date, currentDate.

La condición que fue construida para la validación entre la fecha actual y la fecha fue gracias a la comparación entre ellas gracias al método “after” de java.util.Date, que otorga la posibilidad de conocer si una fecha es posterior a otra.

Con respecto a la alerta, fue necesaria la creación de un nuevo mensaje para que pudiera ser visible ante el usuario. Como resultado, se pudieron obtener las validaciones correspondientes a cada caso.

Figura N° 11. Comportamiento de validación con expedición de alerta.



8.2.2 Validación de fechas en la creación de caso. La creación de caso es un módulo desplegado a partir de “Búsqueda de casos” donde, al encontrar resultados, se permite redirigir al usuario a este módulo para la recolección de datos sobre la persona asegurada, información de pólizas e información sobre eventos que ocurren sobre este individuo. A partir de ello, el sistema registra la creación de un nuevo caso.

La problemática radicaba en que el sistema permitía la creación con sucesos teniendo en cuenta fechas futuras con respecto a la actual y sin tener en cuenta el límite de hora con la presente.

- **Desarrollo del mantenimiento.** No muy distante del proceso en que se desarrolló el mantenimiento sobre la validación a partir de fecha final y la fecha actual, se realizó la creación de un mensaje nuevo de alerta sobre este hecho en la creación de la caso.

Por su puesto, inherente al proceso de validación, era necesaria la creación de la condición cuyo parámetro estaba estructurado fundamentalmente por la comparación entre fechas, a partir un método de la clase Date, after, para corroborar si la fecha donde se iba a realizar el reporte de los hechos era posterior a la fecha actual.

Por otro lado, sobre la validación de la hora con respecto a la actual, fue necesario realizar los siguientes puntos en la lógica de la programación.

- La hora que el sistema registra estaba basada en una variable de la clase DateTime que comprende hora y fecha. Sin embargo, el sistema Cúram registra la hora actual pero con la fecha de 1/1/1970 por defecto. Por lo cual, en primer lugar era precisa la obtención de la fecha actual y, posteriormente, la conversión de Date a DateTime para comparar la fecha que el sistema registra sobre la creación de casos.
- No obstante, paralelamente, se creó un objeto Date a partir del dato DateTime del sistema Cúram. Este proceso era necesario antes de la comparación entre fechas DateTime como se había comentado en el primer punto del proceso, dado que era pertinente conocer cuantos días era necesarios para la actualización de la fecha y sólo se podría saber a partir a partir del resultado de la comparación con el método “subtract” de la clase Date. Guardado el número de días en una variable, se calculaba cuantas horas son equivalentes a dichos días calculados anteriormente, ya que esta operación ha de ser utilizada en el dato inicial del sistema DateTime, empleando el método addHours.
- Sumando las horas apropiadas, la fecha del sistema Cúram se actualiza para que sea posible la comparación de la horas entre la actual y la del sistema. Una vez hecho eso, se creó la validación y posteriormente el mensaje de alerta

para dar aviso al usuario que el reporte del caso no se puede realizar en esos casos.

Como resultado de estos procesos, se obtuvieron las validaciones correspondientes:

Figura N° 12. Validación en la creación del caso

The screenshot shows a web application interface for creating a case. The main window is titled 'Detalles del asegurado' and is divided into three steps: 'Datos del asegurado', 'Información de pólizas', and 'Detalle del evento'. The current step is 'Detalle del evento'. A red error message at the top indicates 'La hora ingresada no es válida.' The form contains several fields for event details, including 'Caso Estrella', 'Facturable asegurado', 'Descripción', 'Fecha del evento' (23/02/2017), 'Hora del evento' (19:00), 'País de Ocurrencia' (Colombia), 'Ciudad de Ocurrencia' (MEDELLIN (ANTIOQUIA)), 'Dirección de ocurrencia', and 'Causa' (Explosion). There is a section for 'Riesgos Afectado' with a 'prueba' option. Other fields include 'Valor estimado' (0), 'Reporta directamente el asegurado', and a 'Reportar:' section with fields for 'Nombre', 'Tipo documento' (Cédula de Ciudadanía), 'Numero documento', 'Telefono fijo', 'Celular', 'Correo Electronico', and 'Vinculo con el asegurado' (Conyuge). The form has 'Cancelar', 'Anterior', and 'Enviar' buttons at the bottom.

8.2.3 Orden de fecha de registro de caso. Este caso fue relativo a la pantalla de búsqueda de caso en el cual, el despliegue de la lista de personas con casos asociados a estos mismos clientes, se mostraban en desorden con respecto a la antigüedad del caso.

- **Desarrollo del mantenimiento.** El despliegue de la lista, a partir de los parámetros de búsqueda, está asociado con una consulta a la base de datos de Cúram que se centraba en el método de la clase responsable de la búsqueda en este módulo del producto. Dado que se había identificado cómo los resultados no obedecían a un orden específico, era preciso la comprensión de la consulta SQL para la inclusión de un "ORDER BY" asociado por supuesto con el nombre del parámetro de la fecha de apertura de caso.

8.2.4 Campos obligatorios. Los campos obligatorios corresponden a los datos elementales para módulos de registros o de búsqueda. Si bien no se ingresan, Cúram tiene la facultad de tomar dichos campos vacíos que son obligatorios como una excepción que puede producirse y en consecuencia para traducirse en una alerta al usuario en la misma pantalla en la cual se intenta realizar la toma de la información.

El reporte de este hecho fue realizado dado a causa la obligatoriedad de los datos de “Descripción”, “Dirección de ocurrencia” y “Causa”.

- **Desarrollo del mantenimiento.** La definición de los campos obligatorios va de la mano con la construcción del modelo. No obstante, dado que el modelo cuenta con múltiples fachadas, alojadas en múltiples paquetes, intuitivamente no sería posible conocer de esta manera bajo qué métodos y qué clases parte el funcionamiento de este módulo.

Por lo cual, fue necesaria la observación inicial del código fuente del módulo, posteriormente el archivo de formato web en el cual se encuentra estructurada la vista de este módulo para conocer la clase por la que parte la estructura la lógica del módulo.

A partir del conocimiento de las clases que intervienen, es posible la intervención necesaria en este caso del modelo. La asignación de estos campos se encuentra en propiedades de la operación o bien el método de la fachada en el cual se está utilizando. RSA proporciona un espacio en el que se hace la declaración de lo que se va a considerar como campo obligatorio.

Dado que se trata de un cambio que va a repercutir en la vista del módulo, era necesaria una actualización del cliente a partir de los comandos propios del sistema Cúram.

Figura N° 13 Campos obligatorios presentes en el módulo de “creación de casos”

8.2.5 Búsqueda por documento. A partir de la observación del comportamiento de la búsqueda de casos, se pudo determinar cómo es posible realizar dicha acción bajo el uso de parámetros como el número de documento o tipo de documento, en conjunto con otros datos de filtro de búsqueda o simplemente por el uso exclusivo de uno de dichos parámetros.

La problemática que subyace en este caso, se encuentra en el hecho que no existe una relación entre el tipo de documento y número de documento, cuando es imperiosa la especificación de ambos datos siempre y cuando se intente realizar la búsqueda con alguno de los campos mencionados.

- **Desarrollo de mantenimiento.** En el método correspondiente existían condiciones sobre dichos datos para corroborar si el usuario había ingresado alguno de los datos para ser incluido en la consulta a la base de datos que proveía la búsqueda de casos.

Por ende, era permitido que algún campo, entre el tipo y número de documento, fuera ingresado finalmente a la consulta. Sin embargo, la construcción de la lógica anterior otorgaba independencia entre los datos en cuestión.

De esta manera se abordaron las condiciones, expresando en términos de código la interrelación de estos datos, uniendo las condiciones a través del operador lógico “and” , construyendo dos mensajes específicos y sus validaciones en el caso si un dato, entre tipo y número, si bien alguno de ellos hizo falta con el fin de alertar al usuario.

8.3 MANTENIMIENTOS NO FORMALES EN SOFTWARE CÚRAM PARA SEGUROS BOLÍVAR

Los mantenimientos no formalizados hacían parte de tareas asignadas que no fueron reportadas en la plataforma de Mantis bug tracker.

8.3.1 Desarrollo de mantenimiento. Dado que el servidor de seguros bolívar puede albergar una cantidad considerable de casos para una persona asegurada por esta entidad, el diseño del módulo sobre esta clase de búsqueda tuvo en cuenta el número máximo de 500 casos que se encuentran a partir de dicha funcionalidad.

Mientras exista la posibilidad que el ente asegurador decida el cambio del número máximo, era necesario el cambio de dicha validación, dado que existiría siempre una modificación sencilla dentro de la lógica del módulo de búsqueda. No obstante, esta situación implicaría actividades inherentes al cambio, como lo es un despliegue a los ambientes en que se maneja, por lo cual sería un tiempo importante que se perdería a partir de dichos cambios.

La validación presente, antes de los cambios hechos a partir del mantenimiento, presentaban la siguiente estructura:

```
If (NumeroDeCasos > 500)
{
throw new ApplicationException (...);
}
```

Lo que se definió para la mitigación de las posibles consecuencias ante el cambio de dicho número, fue la definición de una propiedad perteneciente a una tabla

especifica que aloja dichos valores en modelo de datos de Cúram, en el cual, mantendrá un valor constante del número máximo de casos. La razón de dicho cambio se encuentra en que, si se decide un cambio en ese número, solamente por medio de un UPDATE en la propiedad anteriormente mencionada, se hace dicha actualización sin mayores implicaciones.

Las tareas alrededor de este mantenimiento, en síntesis, fueron:

- Creación de la propiedad de Cúram que albergue el número de casos para la búsqueda de estos mismos.
- Creación de línea SQL “INSERT” para la incorporación de dicho campo a los ambientes de IBM para el producto de Seguros Bolívar.
- Creación de consulta por medio del lenguaje java a la tabla respectiva de las propiedades de Cúram para la obtención del valor del número máximo de casos.
- Modificación de condición que proveía la validación del máximo de casos.

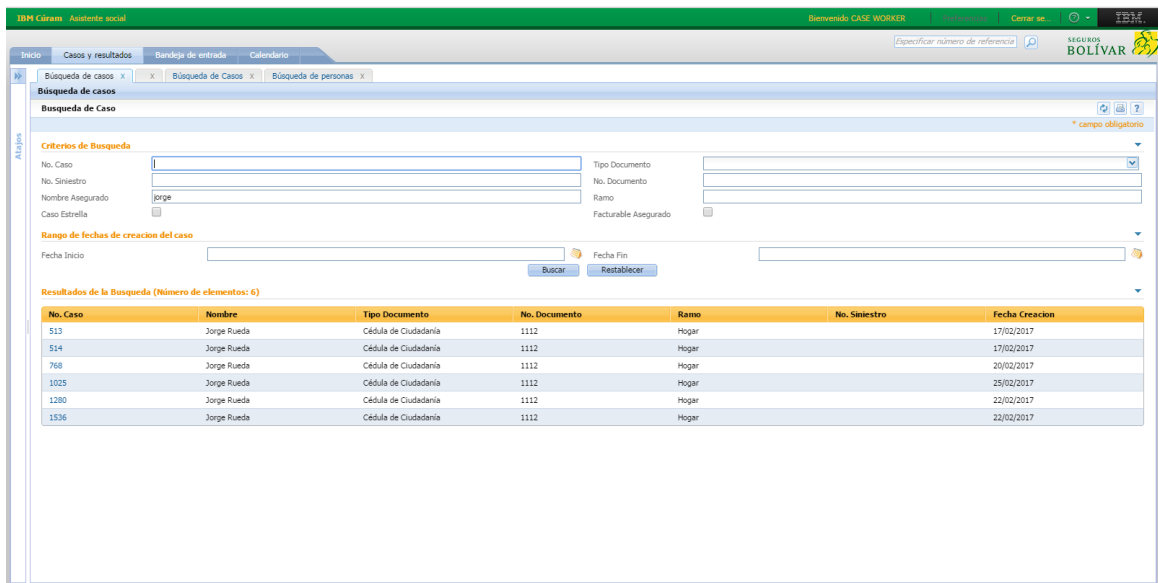
8.3.2 Detalles del caso. Detalles de caso es un módulo anteriormente configurado en el cual se encarga de alojar datos personales de clientes, que se caracteriza por el amplio espacio en la toma en cuenta sobre aspectos propios de las actividades del ente asegurador en sus clientes.

Las actividades en este contexto, se refiere a todos los programas que la empresa propone a sus clientes para la prevención de riesgo que este mismo ente considera bajo el perfil de cada cliente.

Inicialmente, este módulo se encontraba separado de la vista de usuario “búsqueda de caso”, módulo que inherentemente debe contar con tales detalles que el primero no cuenta. Para evitar congestión de datos “búsqueda de caso”, era necesario realizar un mantenimiento en el cual, básicamente, a partir de la lista de casos encontrados por medio de la búsqueda y específicamente a partir del número del caso (primera columna), se puede re-direccionar hacia el módulo “Detalles del caso”.

- **Desarrollo del mantenimiento.** Cúram cuenta con herramientas en el lenguaje XML que permiten, a partir de un dato configurado dentro de la vista al usuario, pueda convertirse en un hipervínculo. Por lo cual, fue necesario el uso del conocimiento técnico para estos propósitos, proporcionado la sintaxis adecuada de acuerdo en la programación XML de un hipervínculo en un campo sencillo.

Figura N° 14. “Búsqueda de casos” con hipervínculo incorporado hacia “detalles de caso” a partir del número de caso



9. EPS COMPENSAR

Durante la práctica, fueron asignados un conjunto de desarrollos a partir de casos de usos aprobados, como mutuo acuerdo entre el cliente, EPS Compensar, e IBM, como parte de las fases de adecuación del software Cúram para las actividades propias de éste primer ente en la ciudad de Bogotá.

Las labores comprendidas en este enfoque de la práctica se encontraban divididas en varias facetas con respecto a los casos de uso: Algunas asignaciones giraban en torno al desarrollo completo de lo expresado en el caso de uso en cuestión, como en otros casos se trataba de desarrollos como extensión de funcionalidad de módulos anteriormente desarrollados a base de la documentación mencionada.

Por otra parte, ciertas labores fueron encomendadas a partir de la necesidad de realizar investigaciones adecuadas sobre el estado de arte del software Cúram para ciertas funcionalidades. Esto viene por cuenta de desarrollos futuros en los cuales se necesita la comprensión del funcionamiento de ciertos aspectos del producto con el fin de realizarlos sin afectar otras funcionalidades.

Las asignaciones partieron del líder técnico de los asuntos de EPS Compensar. Indistintamente de las labores comentadas, cada una de ellas comprendía un flujo ordenado de actividades para su realización.

9.1 FLUJO DE ACTIVIDADES

9.1.1 Análisis del caso de uso. Para cada reporte de la asignación relacionada con la EPS Compensar, se especificaban sobre el caso de uso que comprendía la tarea a realizar. Por lo cual, era necesario la observación del caso de uso para la comprensión del contexto del escenario a tratar.

En un proceso de análisis de caso de uso, inherentemente se toman en cuenta las entradas, precondiciones, flujos de eventos, flujos alternativos flujos de excepción, puntos de extensión, pos condición de ejecución exitosa y fallida, y reglas de negocios relacionados.

Particularmente, el análisis de los casos de uso se vio orientada dependiendo de la labor encomendada.

9.1.2 Actualización de proyectos en entorno remoto. Por medio de la herramienta Git Hub se realiza dicho proceso. Es una práctica adecuada para el inicio de los desarrollos al estar acorde de la última versión del software Cúram.

9.1.3 Creación del modelo. Si bien a través del análisis de los casos de uso se pueden identificar los cambios sobre funcionalidades pre – configuradas o bien las características sobre el desarrollo a realizar, en ambos escenarios, era imperiosa la confección apropiada del modelo: la creación de nuevas entidades dentro de modelo de datos de Cúram, nuevas funcionalidades y campos asociados a estos últimos, o bien el cambio de configuración para dichos aspectos que son reflejados dentro las variables expresadas en la documentación y el flujo de eventos.

9.1.4 Configuración de la lógica de funcionalidades. Al tratarse de cambios expresados en el caso de uso, particularmente en flujo de eventos, eventos alternos y excepciones, era necesaria la identificación de, no solo las clases en las cuales se realizaron cambios dentro del modelo, sino también de las clases directamente relacionadas.

Esto, por su puesto, se realizó a raíz del hecho que ciertas invocaciones a otros métodos con clases relacionadas respecto a las cuales se realizaron determinados cambios, presentaban modificaciones necesarias a partir de las primeras acciones: en el modelo y la lógica.

A partir de dicha identificación se construyó cada lógica de manera sistémica, tomando en cuenta cada proceso que el sistema debe realizar en dicho modulo, excepciones y flujos alternos, y por supuesto las validaciones adecuadas, y llegado el caso, identificadas a partir de las reglas de negocio expresadas en los casos de uso.

9.1.5 Configuración de la capa de presentación. O bien configuración de la interfaz de usuario. Los casos de uso generalmente presentaban un prototipo del módulo a desarrollar. Por el cual, resultaba apropiado siendo una guía en la confección de la vista al usuario que EPS Compensar requería para el software Cúram. Por medio de este prototipo, era posible identificar de una manera más tangible la funcionalidad del módulo y el diseño del código para la capa de presentación a partir lenguaje XML, especificado para los desarrollos web en Cúram.

9.1.6 Pruebas. Las pruebas se veían orientadas a escenario particulares. El diseño de las pruebas venían siendo orientadas por los flujos normales, alternos y de excepciones con respecto al funcionamiento del sistema, con el propósito de conocer si bien actuaba a partir de lo planteado en caso de uso.

9.1.7 Construcción de scripts para la actualización de la base de datos en el ambiente remoto y establecimiento de seguridad. El motivo por el cual se agrupan estas dos fases dentro del flujo de actividades tiene que ver con la naturaleza de estos dos procesos, dado que, su común denominador, se encuentra en que son líneas de comandos SQL pertenecientes a la familia de dos de sus operaciones básicas: INSERT y UPDATE, comúnmente.

La necesidad de la construcción de estos “scripts” viene dado a que, si bien se han desarrollado nuevas funcionalidades y se pretende más adelante subir dichos nuevos desarrollos o cambios al repositorio donde aloja el proyecto común entre ingenieros encargados de las operaciones referentes a EPS Compensar, en lo que se refiere a nuevos campos en una tabla en la base de datos o una nueva entidad en el modelo de datos, necesitan un proceso adicional para que surta efecto para los demás usuarios con respecto a los cambios en los archivos donde se realizaron los desarrollos.

En este caso fue necesario expresarlo en forma de un archivo SQL y construir las líneas adecuadas, con el propósito de que todos los demás, personas encargadas en este proyecto, actualizaran sus bases de datos y de esta manera se hiciera uso de nuevas funcionalidades.

De esta misma manera, luego que se haya probado el comportamiento de los módulos realizados, generalmente cuando se trata de desarrollos nuevos, se procede también al establecimiento de seguridad con respecto a las funcionalidades creadas en los mismos.

Básicamente esto se debe a que el software Cúram comprende cierta cantidad de roles que cuentan a su vez con privilegios. Estos privilegios vienen siendo el conjunto de funcionalidades que pueden ser usados por cada uno de ellos.

Los “scripts” SQL creados para estos propósitos se tratan básicamente de la inserción de métodos creados, contenidos en clases particulares, para que el sistema Cúram pueda diferenciar entre dichos roles el uso de las funcionalidades creadas.

9.1.8 Despliegue. La totalidad del proceso de despliegue para EPS Compensar difiere únicamente, en lo que respecta al proceso para Seguros Bolívar, en la tecnología que proporciona la administración de los proyectos del software Cúram.

9.2 REPORTE DE INGRESO DE USUARIOS

Este desarrollo se propuso dada la necesidad de realizar la intervención de la funcionalidad actual dentro de un usuario administrativo que tenía privilegios para eliminar usuarios dentro del sistema.

El trasfondo del escenario que llevó a estudio y posteriormente el cambio de la funcionalidad, se encontraba en la inexistencia un historial en el cual se pudiera evidenciar qué usuarios fueron eliminados, la fecha de su eliminación y el usuario que ejecutó la acción.

Por otro lado, la funcionalidad solo contaba con la opción de eliminar, cuando, en algunos escenarios se requería en un momento dado deshabilitar un usuario y en un futuro, llegado al caso, re-incorporarlo al sistema.

Correspondiendo a primer caso al cual se hizo el aborde técnico para EPS Compensar, el líder de este sector realizó un guía de aspectos por realizar a partir del caso mencionado. Por tanto, no se hizo ningún análisis de documentación.

9.2.1 Creación del Modelo. Fue necesaria la creación de una nueva entidad en el modelo de datos que contuviera una llave primaria, número único de historial además de sus otros atributos como usuario, último estado, fecha de cambio y estado actual.

Por otra parte, en esta entidad se modelaron también operaciones de inserción para permitir la entrada de nueva información en lo que se refiere a las actividades en este contexto. De otro modo, en lo que se concierne a la actualización de información que estaría ya presente en el historial de la base de datos, se incluyó una operación de sobre escritura para la modificación, particularmente, del estado del usuario.

Una tercera operación se incluyó para la entidad, dado que permite, a través del dato de último estado, obtener y utilizar la información presta en la base de datos y específicamente en dicha entidad. Luego de configuración de la entidad, se realizó la construcción de una fachada en la cual se contemplaban básicamente dos operaciones o métodos: Habilitar usuario y Deshabilitar usuario.

9.2.2 Configuración de la lógica de funcionalidades. Dentro la guía elaborada por el líder de EPS Compensar, se propuso el modelamiento de una fachada en la cual se pudiera generar un reporte basado en la información presente en la entidad recientemente descrita. Sin embargo, en el trabajo que se realizó se hizo una excepción que fue tomada en cuenta en esta fase de desarrollo.

Dado que se hizo un análisis en el código actual en el software Cúram, se encontraron métodos desarrollados anteriormente que cumplían con propósitos de generación de reportes distintos, pero con aspectos comunes entre ellos.

Si pretendía inicialmente incluir una fachada en la cual tendría el papel de la generación de reportes, que consistía en la configuración de tres operaciones: dos de las cuales se encargaban de la obtención de información fundamental para la generación del reporte mencionado: obtención de EPS y obtención de Unidad básica (sub-unidad de la EPS). En lo que respecta al método restante, se encargaría de la generación de reporte.

Siguiendo el orden de ideas, lo que se sugirió en primer lugar al líder técnico y tutor de la práctica empresarial, fue la utilización de dos métodos de obtención de información pre- configurados. De manera similar, se realizó una re-ingeniería en lo que respecta a la elaboración de método de reporte, adecuando la lógica de reportes de otra índole para el contexto en construcción, realizando el mayor trabajo para el diseño de la consulta a la base de datos y su implementación.

9.2.3 Configuración de la interfaz de usuario. Cabe destacar que ya existía un módulo en los usuarios administrativos que realizaban la función de eliminación de usuarios haciendo uso en la búsqueda de estos mismos. Por lo tanto las labores en esta fase se dedicaron a los siguientes aspectos:

- La elaboración de un archivo UIM y propiedades adecuadas en lo que fue la vista al usuario administrativo en el cual, con parámetros de búsqueda: EPS y unidad Básica, se puede generar el reporte.
- Configuración del panel de atajos para la inclusión de una nueva sub-pestaña en la sección de reportes, que hace parte de las opciones presentes en la interfaz de usuario para los administrativos.

9.3 REPORTE DE NOTIFICACIONES

Se trata de un desarrollo completo basado en un caso de uso en el cual básicamente requería que el sistema de software Cúram pudiera, a través de un formato de búsqueda basado en ciertos parámetros, generar un formato de reporte para “notificaciones”, siempre y cuando fuera posible dicha generación en el momento en que ingresaran datos correspondientes a los parámetros correctos de búsqueda.

El momento en que se había hecho la asignación con respecto a este caso, la información que se intenta desplegar en el reporte aún no estaba definida. Por lo tanto, el usuario podría a partir del sistema descargar de un archivo Excel (como formato de reporte). Esto sería la base del desarrollo que contaría luego con una consulta a la base de datos para plasmar la información.

9.3.1 Análisis de caso de uso. El caso de uso manifestó la necesidad del cliente de la realización de un nuevo reporte, especificando los campos que serían los parámetros de búsqueda. De esta manera y siendo un desarrollo completamente nuevo, fue necesario realizar el diseño de lo que sería dentro del modelo una nueva clase que contuviera la lógica de la búsqueda con dichos campos. Por otra parte, dada la naturaleza del requerimiento, se pudo identificar que dentro de este software existen reportes pre- configurados.

Sin embargo, dado que se trata de asuntos paralelos y cada uno de ellos con motivos distintos, fue necesario analizar con la documentación qué funcionalidades son probablemente aprovechadas para un proceso de re-ingeniería en pro de la elaboración del módulo de reporte de notificaciones.

También, dentro de la documentación se ha especificado cómo debería estar disponible para cinco roles distintos el módulo de reporte de notificaciones. Por lo cual, fue necesario tomar en cuenta que se verían afectadas las propiedades para cada rol manifestado en el caso de uso, además de la creación de archivos XML para la creación de cajas de texto, botones implicados en el módulo.

9.3.2 Creación del modelo. Sobre el modelo se consideraron dos aspectos:

- Una clase donde se contengan los métodos apropiados: un método asociado a la toma de información para la búsqueda y generación de reporte. El segundo método tiene la responsabilidad del procesamiento de la información. El procesamiento de información se trata de la separación de esta misma que puede ser recopilada a partir de la consulta SQL. Naturalmente, los campos adquiridos de esa manera no se encuentran de manera individual, sino de manera concatenada. El tercero permite la descarga del formato del reporte.
- Modelamiento de entidad nueva en el modelo de datos del software Cúram que pueda realizar la función de alojar la información de la generación de reportes de notificación. Esta tabla, debe alojar datos de usuario que lo genera, fecha y,

por su puesto, número único de identificación como llave primaria. Intuitivamente, esta entidad se modeló, dado que se debe guardar un registro de esta clase de procesos de reportes.

9.3.3 Configuración de la lógica de funcionalidades. A partir de la clase modelada, además del uso de pre-configuración de clases de reportes con respecto a los métodos de manejo de información y descarga de reporte, se realizó el tratamiento particular al método sobre la búsqueda de notificaciones. Son respecto a los métodos que re-utilizaron para estos fines, fue necesario para adecuación necesaria para este reporte.

La lógica que se aprovechó de otros reportes fue a partir de los campos de EPS, Unidad Básica y Equipo Gestor. Para un rol en específico, quien sería un usuario que maneja casos desde un centro de atención particular, solo podría generar reportes a partir del centro donde él o ella pertenezcan. Otros usuarios, más específicamente administradores, pueden tener privilegios para realizar reportes para diversos puntos de atención.

Realizando un análisis sobre el diseño de las clases implicadas en los reportes, fue necesaria la confección de una clase paralela, no modelada, en la cual se aloja la consulta SQL para la información que posteriormente se plasmará en el formato del reporte y, por otro lado, las validaciones correspondientes a las reglas de negocio expresadas en el caso de uso.

La razón por la cual se realizó esta clase sin ser modelada radica en que dicha clase no va a tener en cuenta campos de entrada y de salida referentes al caso de uso en este contexto.

Teniendo en cuenta las reglas de negocio, se realizaron las respectivas validaciones en los siguientes puntos:

- Las fechas deben ser coherentes en la emisión del reporte. Por tanto, se validó que ninguna fecha sea futura a la actual.
- Se definió un rango apropiado para la búsqueda notificaciones en el caso de uso. Era necesaria por estas circunstancias, la validación de la búsqueda en este sentido.

Dado que en el proceso de modelamiento se incluyó la creación de una nueva entidad dentro del modelo de datos con el objetivo de recoger la información de reportes que se hayan solicitado en los usuarios asignados, era necesario tener en cuenta en la codificación en la toma de información del usuario y fecha actual en la cual se realiza la descarga de reporte.

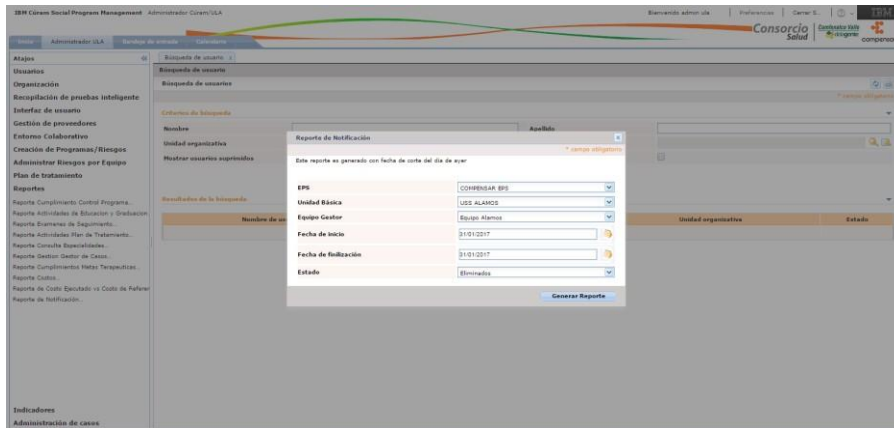
9.3.4 Configuración de la interfaz de usuario. Era imperante la configuración de dos pantallas al usuario dado el análisis del diseño de la funcionalidad: una vista destinada para cuatro de cinco roles que son administrativos y una vistas para un rol que solo tiene alcance en generación de reporte en su centro de atención.

Anteriormente, en el análisis de casos de uso se identificó, siguiendo en el orden de ideas, no solamente las propiedades necesarias para el área de trabajo o bien el módulo de reporte de notificación, sino también para la inclusión de la nueva opción en el panel de cada rol, con el propósito de que, a partir de ello, se pueda hacer la apertura de este módulo.

El botón de descargas fue programado en la interfaz de usuario para que pueda únicamente disponerse de este mismo cuando se haga efectiva una búsqueda exitosa. Dicho botón cuenta con un tratamiento particular dada la función que este mismo contiene. Para que pueda ser funcional, fue necesaria la declaración del método encargado de la descarga del formato de reporte en un archivo especial.

9.3.5 Resultados. Se pueden evidenciar a través de la siguiente imagen, los resultados que se obtuvieron a partir los procesos anteriormente mencionados y cómo se tornan coherentes ante la solicitud de EPS compensar.

Figura N° 15. Módulo de generación de reporte de notificaciones



9.4 INVESTIGACIÓN DE LA FUNCIONALIDAD DE NOTIFICACIONES

Esta investigación partió de la necesidad de evaluar ciertos cambios sobre la presente funcionalidad asociada al envío de notificaciones, ligada a la información que se pretendía plasmar dentro del reporte de notificación.

Las notificaciones son un conjunto de mensajes que son enviados de manera automática a los pacientes registrados y/o usuarios dentro del sistema que, de acuerdo a ciertas actividades dentro del sistema de EPS Compensar, la lógica de la funcionalidad entra en funcionamiento para el envío de mensajes apropiados.

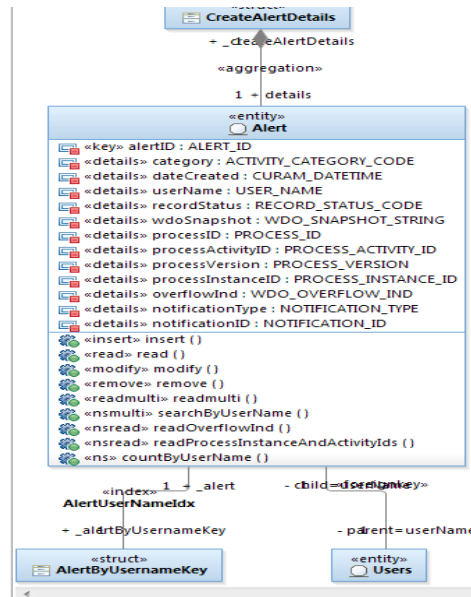
A partir del conocimiento de una entidad presente en el modelo de datos, Alert, expresada por el líder para los asuntos de EPS Compensar, fue el punto de partida para dicho examen.

9.4.1 Objetivos de la investigación

- Conocer la estructura de la funcionalidad y su complejidad.
- Conocer las alternativas para la intervención de la funcionalidad.

9.4.2 Implicaciones en el modelo

Figura N° 16. Estructura de la entidad asociada a las notificaciones.



A partir del examen dentro del modelo presente, se identificaron dos aspectos importantes: Esta entidad resulta un artefacto configurado desde el origen del producto Cúram. Por otro lado, existen tres relaciones con respecto a la entidad.

La importancia de la investigación radicaba en la posible réplica de este escenario, con el fin de aumentar su capacidad en términos de información basado en los asuntos preliminarmente comentados y dados, implica el hecho que en estas áreas no es posible intervenir a la funcionalidad directamente.

De esta manera, se intentó medir, basado en las circunstancias encontradas en el desarrollo de la investigación, la complejidad de la réplica de la funcionalidad dado los resultados en términos del modelo.

En esta fase se ha tomado en cuenta que si resultaba necesario realizar esta toma de configuración previa para un desarrollo personalizado, no derivaba en mayores complicaciones.

9.4.3 Implicaciones con respecto a la interfaz de usuario y lógica de la funcionalidad. Se comenzó esta fase de investigación intentando conocer todas las relaciones, en términos las clases y métodos, por los cuales esta entidad se encuentra en uso.

Para conocer dichos métodos, era necesario identificar el módulo de notificaciones. A partir de la interfaz gráfica y la configuración local de software Cúram, se otorga la posibilidad de conocer bajo qué archivos .uim se encuentra la comunicación entre el módulo de notificación con respecto a la lógica de la funcionalidad.

Abordando dichos formatos .UIM, otorgan información fundamental para el conocimiento de las clases ya que, las herramientas prestas en el formato de construcción XML para el producto Cúram, permite el acople entre interfaz la interfaz gráfica y la lógica a partir de la especificación de la clase y método que se harán cargo la funcionalidad al momento en que el usuario realice alguna actividad dentro de la interfaz. A partir de esta situación, es posible conocer qué métodos se encuentran involucrados en la funcionalidad de notificación.

Dentro de la investigación en esta fase, se encontró con un red de comunicación entre clases asociada a la fachada encargada del envío de notificaciones. Sin embargo, se pudo encontrar a través de la investigación que corresponde típicamente al uso del patrón del diseño. De esta manera, no comprende más aspectos por agregar para la evaluación de la complejidad.

En conclusión, de la investigación se pudieron evaluar todos los aspectos que conciernen a una funcionalidad y a partir de cada fase de su evaluación, se pudo notar como estuvo inclinada al favorecimiento de una réplica de esta misma para la realización de un tratamiento personalizado a los requerimientos de la EPS Compensar, por lo cual era viable este tratamiento para posterior re-ingeniería para dichos fines

9.5 REAPERTURA DE CASO CERRADO MANUALMENTE

A partir de la creación del caso de uso “Reapertura de caso cerrado manualmente” EPS Compensar manifestó la necesidad de un desarrollo para un módulo en el cual se pueda realizar una búsqueda de casos que en el momento de dicha acción se encuentren cerrados, basado en ciertos parámetros de búsqueda aclarados en la documentación, para que se pueda desplegar una lista de estos mismos y, a partir del criterio del usuario, se pueda seleccionar en la lista los casos en los cuales se pretende realizar la re-apertura.

Un caso hace referencia a la apertura del seguimiento de un paciente en el cual se ha podido identificar una enfermedad crónica.

9.5.1 Análisis de casos de uso. Se identificó que este desarrollo iba hacer especificado para el uso de un rol administrativo.

En el caso de uso se manifestó un prototipo en el cual se intenta desplegar una lista en la misma vista de usuario donde se realiza la búsqueda de los casos. Dada la experiencia adquirida dentro de la práctica empresarial, se pudo identificar que este prototipo propuesto no podría ser viable.

El motivo por el cual el diseño está por fuera del dominio de la tecnología Cúram se encuentra específicamente en la programación de la interfaz de usuario dado que se espera un botón “Resignar Equipo gestor” al momento en que se encuentre algún caso cerrado en este módulo. En estas circunstancias no es posible replicar, dado que la programación XML no contempla el escenario en cuestión.

A partir la situación comentada anteriormente, se realizó la sugerencia sobre la separación de la interfaz en dos vistas distintas. De esta manera se puede realizar la búsqueda de los casos cerrados, re-direccionando a otra vista la lista de los casos encontrados.

Por otra parte, a través del conocimiento del sistema que la EPS Compensar maneja, por medio del caso de uso, se identificó cómo en la construcción de la lógica del módulo era necesario considerar dos asuntos: Casos automáticos y casos manuales.

Es necesario aclarar que en este escenario se hace referencia a la re-apertura manual de casos. No apertura de casos que fueron creados de manera manual. Por lo cual, también era necesario considerar los casos automáticos. En este escenario, lo que imperaba era el reconocimiento en la base de datos de Cúram dado que son comprendidos en tablas distintas.

Aunque en el caso de uso explícitamente no plantea el almacenamiento de la re-apertura de casos, es decir, la información que concierne al usuario, paciente, fecha, inicio de tratamiento y fecha de finalización de tratamiento, fue necesario el planteamiento de una tabla en la base de datos que pudiera servir de historial para estas aperturas.

9.5.2 Creación del modelo. Como fue comentado anteriormente, en el modelo se consideró una entidad para ser incluida en el modelo de datos en cual tiene una relación inherente con las tablas de casos manuales y casos automáticos.

En el modelo fue necesario considerar, en primer lugar, una fachada o clase en la cual pueda alojar tres métodos: El método fundamental para este contexto, se trató de uno que pueda tomar los campos de la búsqueda para devolver una lista particular donde se encuentre la información de los casos encontrados.

En consecuencia, se modelaron dos estructuras que pueden alojar los campos de entrada y de salida respectivamente, para ser utilizados en el método de búsqueda. El segundo y tercer método se encargan básicamente de realizar la re-apertura de los casos. Para cada uno de ellos, una función distintiva: de casos automáticos y manuales respectivamente. Los valores de entrada para estos métodos, son los campos de salida del método de búsqueda.

9.5.3 Configuración de la lógica de funcionalidades. Los métodos fueron contruidos principalmente por una consulta a la base de datos respectiva a la función de búsqueda, y dos operaciones UPDATE de cada uno de los métodos restantes.

Dado que la información que se necesita para la búsqueda de los casos involucra, no solamente las tablas prestas a la información de estos escenarios automáticos y manuales, sino también las tablas especificadas a la información del paciente, implicó el cruce de tablas a base de igualdades de ciertos campos comunes, de manera que se pudo originar un resultado que se intenta mostrar en la interfaz del usuario.

En este caso existió la particularidad en lo que respecta a la programación para señalar los casos que tendrán su re-apertura. Como parte del planteamiento del modelo, es necesario destacar en este punto cómo este campo consistió en asignarle una variable booleana.

Si bien el usuario realiza la selección de alguno de ellos, el sistema toma el valor de verdadero o falso para que pueda reconocerse la solicitud del usuario. Adicionalmente, para ambos casos de los métodos de re-apertura, se instauró dentro de la lógica una operación INSERT con los valores apropiados de

los campos de la tabla construida en primera instancia en la fase de modelamiento para el registro de aperturas de ambas índoles.

En el contexto de las validaciones, se identificó a través de las reglas de negocios, cómo era necesario asegurar que la operación de búsqueda no se realizara sin algún criterio de búsqueda.

9.5.4 Configuración de la capa de presentación. Por medio de la situación que se había presentado, en la cual el prototipo presentado en el caso de uso no podía corresponder a una interfaz factible en el diseño para el usuario, en esta fase se elaboró dos archivos .UIM, con sus propiedades respectivas, en los cuales en la primera se dedicó para la búsqueda de los casos.

Por otra parte, en la segunda vista, se dedicó lo que respecta a la lista de los casos encontrados. Como este módulo es dedicado a un usuario administrativo, se realizaron los cambios respectivos con la programación del panel de atajos para ser desplegar la opción de búsqueda de casos.

9.5.5 Resultados A continuación se puede evidenciar los resultados sobre el desarrollo.

Figura N° 17. La re-apertura de casos.

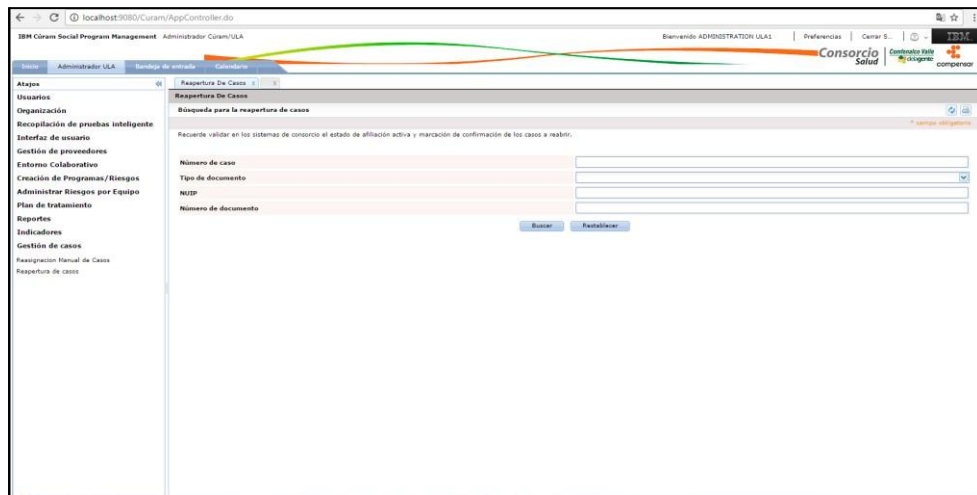


Figura 18. Lista producida por la búsqueda de casos

No. Caso	Nombre y Apellidos	Riesgo
2049	Wiston King	EPOC
2052	Wiston King	HIPERTENSION ARTERIAL
2304	Wiston King	OSTEARTROSIS
774	Wiston King	OSTEARTROSIS

9.6 AMPLIACIÓN DE LA FUNCIONALIDAD “ASIGNAR CASO”

Se trata de una labor que se realizó a partir de la funcionalidad del sistema Cúram con el propósito en que un usuario asignado en un unidad dentro de un centro de la EPS Compensar pueda tomar un caso de un paciente, dependiendo de su riesgo y régimen: Plan Obligatorio de Salud (POS) y/o Plan Adicional de Salud o mixto, a partir del criterio de la usuarios administrativos de unidades.

La asignación de caso requería de un flujo de acciones por parte de los usuarios anteriormente dichos:

- Creación de unidad básica (Sub-unidad de EPS compensar)
- Creación de equipos (Sub-unidad de unidad básica que consiste en un conjunto de profesionales prestos en la atención de pacientes con riesgos específicos)
- Registro del paciente
- Asignación

En lo que respecta a los dos primeros momentos de la asignación de casos, la creación de unidades y equipos, se requería el uso de la interfaz gráfica o bien un módulo presente en esta funcionalidad que se encargara del registro de nueva unidad y equipo. No obstante, en dicho módulo todavía no existía una manera en

la cual una unidad asociada a la EPS Compensar tuviera una diferencia entre las unidades en términos de los programas de salud.

9.6.1 Análisis de caso de uso. Si bien el caso de uso dejaba entrever que básicamente se quería añadir una lista que iba a reunir a los programas POS, PAS y mixto, se debía englobar el conjunto de implicaciones para el desarrollo:

- Dado que el módulo en cuestión realiza una función de registro para unidades básicas, inherentemente, debe contar con una entidad en el modelo de datos por el cual se pueda alojar dicha información. Sin embargo, dado el diseño de la base de datos para el software Cúram, se debía considerar analizar el contexto por parte del grupo técnico de IBM para concluir que debe construir una nueva entidad que se encargue únicamente de englobar los programas de salud y establecer una relación con la entidad de registro de unidades
- Como se requiere una lista dentro de un módulo que no existía por ese periodo de tiempo, era necesario construir un archivo especial que se encargaría de reunir los tipos de programas de salud. Esta clase de archivos están ligados íntimamente a los artefactos de domino en el modelo, en el cual puede asociarse e incorporarse al modelo en el cual representa el primer paso para garantizar que pueda verse posteriormente en la interfaz del usuario.
- En el orden de ideas, era necesario localizar dentro del modelo los artefactos relacionados a la funcionalidad del registro de usuarios. Esto, naturalmente, se trataría para abordar el modelo en modificaciones adecuadas.
- La adecuación del modelo representa una reestructuración de los métodos involucrados dentro del módulo del registro. Además de la necesidad de localización de archivos .UIM, fue contemplado los cambios de sus propiedades respectivas para la inclusión de la lista. Por otra parte, en la inclusión de los programas se tuvo en cuenta la pre-existencia de unidades. Dada la actualización de estas mismas, debía estar contemplado.

9.6.2 Configuración del modelo. Preliminarmente se creó un archivo especial que tuviera la configuración de los programas de salud. A partir de lo anteriormente analizado, resultó en la adecuación de la estructura adecuada que

hacía parte de la configuración del modelo que hacer parte del registro de unidades.

Como la estructura mencionada ya se encuentra acoplada con la fachada que hace parte la funcionalidad, se realizó la adición del nuevo tipo de dato que corresponde para a los programas de salud.

Por otro lado, se realizó la construcción de la entidad tomando en cuenta la imperiosa relación que debe tener entre esta misma con la tabla encargada de los registros de unidades básicas. Las operaciones configuradas incluían aspectos de inserción para nuevos registros y sobre escritura, además de una búsqueda para obtención de información presente en la entidad en función de la llave primaria de esta misma.

9.6.3 Configuración de la lógica de la funcionalidad. El reto que se presentaba en este escenario consistía en el acople de los programas de salud en la lógica del registro de unidades. En términos generales, dado que había que contemplar que los nuevos registros que incluían dichos programas y que fue creada una nueva entidad relacionada con el registro de unidades, era necesaria la utilización de las operaciones por las que fueron configuradas dentro de la entidad más reciente.

La inserción de nuevos registros y sobre escritura de unidades pre-existentes eran los puntos por traducir en términos de codificación. La dificultad que acarrearán estas acciones se encuentra en la configuración adecuada ante una anteriormente desarrollada.

La clase dedicada a los registros de unidades contaba con tres métodos: Nuevos registros, modificación de registros y vista de información de unidades. Se abordó cada método dado el análisis que realizó en la configuración pre-existente:

- Vista de información: Al proveer información para otras clases en funcionalidades diversas en el software Cúram, se decidió replicar la presente configuración y crear una sobrecarga de métodos en este sentido, donde se puede utilizar este último, incluyendo información de los programas de salud, para el uso exclusivo de la clase especial para el registro de unidades. La decisión se tomó en torno al hecho en que no se conoce todo el amplio conjunto de clases en las cuales se encuentra invocando el método de vista de información. Como una solución ágil ante el requerimiento se definió utilizar un método exclusivo para la clase.

Nuevos registros: Intuitivamente se incluyó al final del método dado que ya lógica cuenta con la información respectiva de registro de unidades. Por lo cual, restaba la inclusión de la operación de inserción de nueva unidad con el programa de salud.

- **Modificación de registro:** Se utilizó el sistema “try- catch” como práctica adecuada en la codificación del método. Dado que este mismo se encuentra presto para modificación, únicamente que tiene apenas alcance en la tabla respectiva a registros de nuevas unidades, se hizo este tratamiento para incluir las operaciones de sobre escritura dentro del try, si ya existen unidades con programas de salud asociados y dentro del catch si bien no existe el registro con programa, utilizando de nuevo la operación de inserción.

9.6.4 Configuración de interfaz de usuario. Se incluyó una lista desplegable en términos del lenguaje XML para que el usuario administrativo pueda elegir el programa al cual va a asociar la unidad que intenta registrar. Por otro lado, también se tienen en cuenta las variables dentro de las propiedades del módulo adecuadas.

9.7 BÚSQUEDA DE TAREAS

Software Cúram contiene un módulo especificado para la búsqueda de tareas a partir de parámetros descritos a partir del caso de uso respectivo. Una tarea en este contexto, corresponde a un conjunto de actividades que deben realizar los pacientes de acuerdo a los riesgos que presentan estos mismos. Las tareas pueden corresponder por ejemplo, a exámenes y terapéuticas, según el plan de tratamiento del riesgo.

La labor dentro de este desarrollo fue la modificación de la funcionalidad agregando campos nuevos según los cambios que se realizaron en la documentación. Básicamente, los cambios que se necesitaban realizar estaban en función a los campos “Asignado por” tomando en cuenta nombre y apellido de usuario administrativo. Anteriormente, en la imagen se puede evidenciar que solamente en los parámetros de búsqueda contenían el campo “Asignados a”

9.7.1 Análisis de casos de uso. Dada la naturaleza de los campos en que se debían incluir a incorporar al módulo no representaba un aspecto en cual imperan procesos de validación en el uso del nombre y apellido del usuario en el módulo

para ciertas circunstancias. El uso de estos campos es según lo requiera el usuario, sin restricción alguna.

9.7.2 Modificación del modelo. En una intervención de un desarrollo pre-existente, se debió identificar en el modelo la fachada y principalmente las estructuras que pertenecen a la funcionalidad, dado que prima la inclusión de los nuevos campos que serán tenidos en cuenta para la lógica y la interfaz gráfica coherentes con el requerimiento. No obstante, el primer paso, naturalmente, parte del modelamiento de estos mismos.

9.7.3 Modificación en la lógica de la funcionalidad. Se tuvo en cuenta inclusión de dichos datos a la consulta que provee la búsqueda de tareas. Por su puesto, se esperaba que la funcionalidad se edificara en torno a una consulta la base de datos presente en la lógica de la funcionalidad.

Por lo cual, se debía ubicar dentro de la condición de la consulta los datos de nombre y apellido en una condición "if". Esta práctica se realizaba para cada campo presente en la consulta dado que, si algún dato no fue suministrado, y se intentaba realizar la consulta, el sistema encontraría una excepción. Esta condición evitaba un excepción NullPointerException.

9.7.4 Modificación de la interfaz de usuario. Estos cambios consistían en adecuar los campos presentes en la interfaz actual para conseguir los resultados proyectados en el caso de uso. De esta manera se hizo la intervención del archivo .UIM identificado como parte de la funcionalidad, y las propiedades inherentes a estos campos.

9.7.5 Resultados

Figura N° 19. Módulo de búsqueda de tareas luego de la intervención

Criterios de búsqueda	
ID de tarea	<input type="text"/>
Asignado a	Unidad organizativa <input type="text"/> Unidad POS <input type="text"/>
Asignado por: Nombre	admin <input type="text"/>
Prioridad	Alto <input type="text"/>
Tipo de objeto de negocio	<input type="text"/>
Estado	Abierta <input type="text"/>
Apellido	user <input type="text"/>
Fecha límite	07/01/2017 <input type="text"/>

Buscar Restablecer

Resultados de la búsqueda (Número de elementos: 1)

Asunto	Prioridad	Fecha límite
Prueba Unidad	Alto	07/01/2017 00:00

9.8 SOLUCIÓN A LA PROBLEMÁTICA ANTE LA CARGA DE PACIENTES EN UNIDAD BÁSICA CON RIESGO ASOCIADOS

Esta asignación fue una investigación en términos lógicos, es decir, sobre el examen, intervención y solución de la codificación de una funcionalidad que desencadena un comportamiento específico basado en un estudio preliminar que teóricamente describe los hechos erráticos de dicha funcionalidad, a falta de conocer, de manera más profunda, el porqué del comportamiento y cómo se debe cambiar.

El escenario que se presentó, luego de evidenciar una investigación teórica, gira en torno a la toma de datos automática de registros de un paciente para asignarse a una unidad básica en la cual tiene un riesgo particular asociado, coherente con los riesgos que dicho paciente registra.

La pluralidad de unidades básicas en las cuales coincide en riesgos configurados, permitió que el sistema con fallas, en ese período de tiempo desconocidas, se vincularan pacientes a otras unidades básicas de la EPS Compensar aunque ya se encuentran vinculadas inicialmente a una. Esto corresponde, teóricamente, a una violación de la regla de negocios especificado en la documentación de la funcionalidad de “asignación de casos”

9.8.1 Desarrollo de la investigación y solución. A partir del conocimiento técnico de IBM para la EPS Compensar, la investigación partió de una clase conocida que previamente fue desarrollada, ajustada al sistema del cliente en la cual intuitivamente partía la falla. Dado que apenas se conocía la clase

involucrada, el examen del código de realizó a partir de la depuración del código, simulando escenarios de prueba para descartar aspectos. La clase de asignación contenía un método particular que tomaba información del paciente para buscar una unidad básica disponible. Se conoció en su contenido cómo existe una entidad referente a los centros de atención o unidades básicas, donde se modeló una operación no estándar.

Cabe aclarar que esta clase de consultas más complejas pueden ser creadas a partir del modelo. Por lo cual en la lógica de la funcionalidad sólo estaría actuando para la instanciación necesaria de la entidad que contiene la operación. La consulta en este contexto asigna constantemente a una unidad que se encuentra disponible con el riesgo, sin tener en cuenta que el paciente que se carga al sistema puede ya contar con una asociación con otra unidad. Sin embargo, la naturaleza de la consulta contenía una asociación con lógica de la funcionalidad.

La lógica proveía el código de una unidad particular, identificador único de una unidad, para la operación configurada en la entidad, al ser un parámetro que necesitaba para la ejecución de esta misma. Por ende, esta sería la razón por la cual se omitía alguna verificación de unidades con registro del paciente.

De suerte que el funcionamiento de la consulta permitía la asignación de casos particularmente por el hecho de que fue diseñada para garantizar el registro de un paciente técnicamente nuevo. De tal manera que la solución partió de la corrección de la consulta configurada en el modelo, eliminando la línea de la operación que recibía el código de la unidad.

Además se hizo un cambio dentro de la lógica incluyendo una línea en la cual se pueda obtener la unidad básica donde se encuentra el paciente, si se encuentra presente. Este hecho pasó desapercibido dado la lógica nunca reportaba una excepción.

10.CONCLUSIONES

La práctica empresarial dentro de compañía International Business Machine fue el escenario idóneo para el crecimiento profesional a partir de la formación adquirida en la Universidad Industrial de Santander dado que se realizaron funcionalidades, mantenimientos e investigaciones que involucraron conocimientos adquiridos a través de la carrera de ingeniería de sistemas con respecto a la ingeniería de software, programación orientada a objetos, bases de datos y programación web.

Por lo cual, fue necesaria la aplicación de estos conceptos en el entorno real del mundo corporativo orientado a las operaciones de la empresa con énfasis a la industria de salud, que proporcionó a partir de estos escenarios la oportunidad de adquirir experiencia como ingeniero en esta industria a base de la exigencia del mundo laboral.

Particularmente en el manejo del software IBM Cúram, adquirí conocimientos sobre la programación distribuida, aspectos que se abordaron con dicha temática teórica y práctica con el fin de desarrollar los requerimientos y mantenimientos para las empresas EPS Compensar y Seguros Bolívar.

Adicionalmente, a partir de múltiples escenarios abordados a través de la práctica empresarial incrementé mis conocimientos con respecto a la programación orientada a objetos en énfasis en la construcción y soporte de software, además sobre la construcción de interfaces de usuario o bien “front-end” a partir del manejo del lenguaje XML y aspectos inherentes a por medio de la tecnología presente en IBM Cúram.

BIBLIOGRAFÍA

BROCK, Jhon. GUPTA, Aron. Java EE and HTML5 Enterprise Application Development (Oracle Press). 1st Edition, Kindle Edition. Mar 28, 2014 ISBN-13: 978-0071823098

CEBALLOS, Javier. Java 2 curso de programación. España: RA-MA editorial, 4 edición, 128 pag. ISBN978-84-9964-032-7

GRENNAN, Paul. Fundamentals of the IBM Cúram SPM Platform for Developers 6.0.4 9D32G – Part two, [En Línea.] 2012, 20p. , [Consultado 1 marzo 2017]. Disponible en https://www-03.ibm.com/services/learning/pdfs/curam_spm_v6.0.4.pdf

RODRIGUEZ, Elizabeth. La Historia de IBM. [En línea]. [Consultado 15 abril de 2017] Disponible en <http://www.maestrosdelweb.com/la-historia-de-ibm/>

Newsletter ISOTools Excellence. ¿En qué consiste el ciclo PHVA de mejora continua? [consultado 20 abril de 2017] Disponible en : <https://www.isotools.org/2015/02/20/en-que-consiste-el-ciclo-phva-de-mejora-continua/>.

IBM Cúram solutions. Entrega de programas sociales y sanitarios que mejoran los resultados y reducen los costes. [consultado 10 abril de 2017] Disponible en <http://www-03.ibm.com/software/products/es/social-programs>

NORMAS ICONTEC Bogotá D.C.: Universidad Nacional de Colombia, 2017. p.10-20 ...).