

OpenSITE

SISTEMA FEDERADO DE APLICACIONES PARA LA CARACTERIZACIÓN
DE NODOS POTENCIALES DE REDES DE E-SALUD

Lilia Edith Aparicio P. PhD
Paulo César Coronado S. MsC

FASE III
RESULTADOS DE INVESTIGACIÓN

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
CENTRO DE INVESTIGACIONES Y DESARROLLO CIENTÍFICO
Bogotá, D.C.
2018

PRESENTACIÓN

En el marco del proyecto *Telemedicina Bogotá 2K*, el grupo de investigación en telemedicina de la Universidad Distrital Francisco José de Caldas (GITEM), realizó un estudio de campo para caracterizar las entidades de la Red Distrital de Salud. Este estudio permitió dar un diagnóstico del estado de la eSalud en la ciudad y refrendar que las acciones de investigación del grupo estaban alineadas con la propuesta que la Organización Mundial de la salud (OMS) y la Unión Internacional de Telecomunicaciones presentaron en las *Herramientas para el Desarrollo de Estrategias Nacionales de eSalud* [ITU-OMS, 2012].

Como consecuencia de los hallazgos presentados en los informes del Ministerio de Salud y Seguridad Social (MINSALUD) [MINSALUD, 2016], de la Organización para la Cooperación y el Desarrollo Económico (OCDE) [OCDE, 2015] y el Observatorio Así Vamos en Salud [OAVES, 2017]; el grupo GITEM desarrolló la fase III del **Sistema de Información para la Caracterización de Nodos Potenciales de Redes de eSalud** con el que se propone un Portal de inteligencia analítica que apoye los procesos de definición de capacidad en proyectos de eSalud.

OpenSITEM, como se ha denominado el sistema, es fruto del esfuerzo que realiza el grupo para sistematizar su experiencia de diagnóstico. Busca ofrecer una plataforma alternativa, independiente y emergente que colabore en la superación de las limitaciones que en materia de gestión y análisis de información tienen los equipos de trabajo encargados del diseño y desarrollo de redes de e-salud.

En el dominio técnico, OpenSITEM es un sistema federado de aplicaciones de software libre o de código abierto que provee herramientas para analizar datos e información de los siguientes elementos que son de interés para la descripción - y definición de capacidad de, nodos potenciales de redes de e-salud: entidades de salud, servicios médicos, tecnologías de interconexión, operadores de telecomunicaciones, equipos médicos, organizaciones, profesionales, estándares, pacientes, enfermedades, medicamentos y proyectos. OpenSITEM *no es un sistema de eSalud* sino de una plataforma para apoyar la definición de la capacidad que tiene un nodo para potencialmente hacer parte de una red de eSalud.

OpenSITEM propone un mecanismo para la integración de actores en el área de *caracterización y análisis de capacidad* de nodos en redes de eSalud. Provee un escenario ubicuo, basado

en tecnologías de la información y un modelo de trabajo colaborativo en red que propende por la construcción evolutiva de una base de información y conocimiento. Siguiendo los lineamientos de apertura en la información y el conocimiento (open philosophy), busca elaborar un *recurso público* para ayudar al diseño de proyectos de e-Salud en nuestra ciudad.

La arquitectura general del sistema y la experiencia en el desarrollo se presenta en este documento, organizado en las siguientes secciones:

- **Antecedentes:** Enumera los aspectos que motivaron el proyecto y un resumen de las fases anteriores.
- **Estructura del Proyecto:** Muestra un resumen de los aspectos, que a consideración de los autores, son relevantes para medir el impacto presente y la potencialidad del proyecto.
- **Contexto Teórico:** Recopilación de aspectos teóricos que fueron de interés para abordar el desarrollo del trabajo de investigación .
- **OpenSITEM:** Descripción de la arquitectura del sistema resultante haciendo énfasis en sus conceptos y propiedades fundamentales. El producto software que de manera emergente se está creando está centrado en esta descripción y guiado por los casos de uso que se presentan en este informe. Esta sección se complementa con los anexos de los modelos de requisitos, análisis y diseño, implementación, despliegue y datos.
- **Experiencia de Desarrollo:** Teniendo en cuenta que la plataforma de software que soporta OpenSITEM es de código abierto, se presenta una definición del método de trabajo empleado para que los grupos interesados en participar puedan integrarse a la comunidad de desarrollo. Esta sección hace especial énfasis en los aportes originales del grupo y describe las demás herramientas que se integran al modelo.
- **Conclusiones y Recomendaciones:** Aspectos y hallazgos que se deben considerar para la consolidación de OpenSITEM así como el desarrollo de nuevos proyectos de investigación.

Lilia Edith Aparicio Pico. PhD
Directora GITEM+
Grupo de Investigación en Telemedicina
Universidad Distrital Francisco José de Caldas



Capítulo 1

Estructura del Proyecto

Este capítulo presenta un resumen del proyecto denominado *OpenSITEM: Sistema Federado de Aplicaciones para la Caracterización de Nodos Potenciales de Redes de e-salud*.

1.1. Objetivos

El proyecto planteó los siguientes objetivos:

1.1.1. Objetivo General

Describir una arquitectura de un sistema software cuya funcionalidad permita caracterizar nodos potenciales de redes de eSalud, considerando aspectos de adaptabilidad e interoperabilidad.

Objetivos Específicos

- Identificar, a partir del análisis de los resultados del estudio de campo existente, los tipos de nodos nucleares para la definición de redes de eSalud.
- Producir o adaptar un marco de desarrollo de aplicaciones de código abierto que permita a los integrantes del grupo de investigación desarrollar módulos de software alineados con la descripción de la arquitectura.
- Construir prototipos de software que implementen la funcionalidad de los módulos nucleares del sistema.

- Establecer el nivel base de calidad de la descripción de la arquitectura y de los prototipos desarrollados con base en pruebas automatizadas y valoraciones de parte de los interesados.

1.1.2. Base Metodológica

El proyecto se desarrolla con un enfoque mixto cualitativo - cuantitativo, con alcance exploratorio, descriptivo y proyectivo.

Los métodos específicos empleados se decantan de una metodología Architecture Definition Method (ADM) y un método de desarrollo ágil basado en OpenUP, el cual es orientado a la gestión del riesgo, guiado por casos de uso, centrado en la arquitectura, iterativo e incremental.

1.1.3. Metodología de Recolección de Información

El proyecto recolecta información para la definición del modelo de requisitos, para el establecimiento del nivel de calidad y para el soporte de las decisiones significativas para la arquitectura. A continuación se presenta los métodos empleados en cada una de estos dominios:

Modelo de Requisitos

Se emplearon varios métodos de los descritos en el OpenUP [?, ?], es especial las entrevista a interesados, los talleres de requisitos y el estudio de sistemas similares. Todos y cada uno de ellos se realizaron de acuerdo a los lineamientos de OpenUP, siendo los talleres complementados con las técnicas descritas por Gottesdiener [Gottesdiener, 2002].

Establecimiento de nivel de Calidad

Para el establecimiento de nivel de calidad se empleó el análisis de calidad de software empleando la herramienta SonarQube¹, el análisis de logs del sistema y la presentación de prototipos a los interesados - técnica descrita en OpenUP [?, ?].

¹<https://www.sonarqube.org/>

Soporte a decisiones arquitectónicamente significativas

Para recolectar información significativa para este dominio se empleó la técnica de reunión de tormenta de ideas y el *mapeo* de historias de usuario[Patton,2014].

1.1.4. Instrumentos Empleados para Recolección de Información

Varios instrumentos fueron diseñados para recolectar información y se aplicaron en diferentes sesiones de las reuniones de tormentas de ideas, talleres de requisitos, presentación de prototipos y entrevistas a interesados. En , y se presentan los instrumentos empleados, junto con una breve descripción de la metodología de creación, validación e interpretación.

Metodología para la Estructuración y Validación del Documento

La creación del formato de entrevista siguió un método de refinamiento basado en retroalimentación directa a partir de los pasos definidos por Dave Collingridge[Zellerino, 2009]. Cada una de las versiones fue desarrollada por el equipo y presentadas en los talleres de requisitos para aprobación por parte de los interesados.

El proceso consistió en 6 pasos que se aplicaron de manera iterativa hasta obtener una versión estable del documento:

- Validar por revisión de pares. Empleando expertos en el dominio - integrantes del grupo de investigación, y expertos en la elaboración de instrumentos - profesionales adscritos al comité de autoevaluación institucional colaboraron en esta tarea. Principalmente se revisó que las preguntas tuviesen claridad en el enunciado, no tuvieran sesgo, no generaran referencias circulares, no fueran de *doble cañon*, entre otros. En este paso se categorizó cada pregunta de conformidad al contenido, conforme a la clasificación dada por García Córdoba[Garcia, 2008].
- Automatizar el instrumento. Para esto se elaboró un sistema de gestión de encuestas utilizando el marco de desarrollo OpenSARA. Dicho sistema permite crear preguntas de diferentes tipos, agruparlos en instrumentos, definir periodos de aplicación, generar reportes, entre otras funcionalidades².
- Realizar una prueba piloto: Se seleccionó un conjunto que no superara el 20 % de la población de interesados y se les aplicó el instrumento. Se pidió que evaluaran la claridad, extensión y dificultad del instrumento.

²Si bien este módulo no hace parte de los objetivos del proyecto, si sirvió para validar la potencialidad del marco de desarrollo OpenSARA. El sistema de gestión de encuestas está siendo probado a nivel institucional para soportar los procesos de autoevaluación (<http://autoevaluacion.udistrital.edu.co/version3/>)

- Analizar los datos recolectados. Teniendo en consideración que los instrumentos están automatizados, el análisis parte de que se tiene datos íntegros. Se revisa por tanto la relación entre la respuesta y el sentido de la pregunta, la validez de los dominios de datos (en especial en preguntas cuantitativas), la pertinencia del tipo de pregunta (selección múltiple, abierta, selección única, etc) y el tiempo que se tarda en diligenciar el instrumento.
- Análisis de componentes principales. A partir de las respuestas se crea una matriz de correlación de conceptos y se descartan aquellas preguntas que tengan un alto grado de correlación[?].

1.1.5. Declaración de situación esperada

Con el desarrollo del proyecto se obtiene una descripción de arquitectura, los lineamientos para la construcción progresiva de un sistema software que implemente dicha arquitectura; y un conjunto base de prototipos. A partir de la entrega de estos resultados se espera que los estudios de campo y análisis realizados por el grupo de investigación puedan ser accedidos públicamente a través de Internet.

1.1.6. Requisitos Arquitectónicamente Significativos

Para la descripción de la arquitectura se definen los siguientes requisitos arquitectónicos para la adecuada gestión de riesgo:

- Integrar aplicaciones robustas que en la actualidad estén licenciadas como código abierto.
- Se prioriza la adaptación frente al desarrollo desde cero.
- La interoperabilidad utilizan Arquitectura Orientada a Servicios con modelo de Bus de Servicio Empresarial
- A nivel de aplicativo se promueve una arquitectura multicapa- multinivel, en donde la persistencia se tiene en nodos separados.
- El crecimiento de la aplicación se realiza a través de Plugins o utilizando las API definidas.
- Se propende por un sistema federado de aplicaciones
- Funcionalidad emergente. Es decir, prototipos base cuyo acercamiento a la arquitectura se realiza progresivamente. Esto con el fin de gestionar -ser conscientes, del riesgo relacionado con el bajo tiempo de permanencia de personal especializado³.

³Debido a la escasa financiación económica, no es posible mantener el personal que se ha especializado en el proyecto. El grupo - y sus profesores asociados, mantienen un ciclo corto de reclutamiento, capacitación, producción y transferencia

1.1.7. Repositorios de Código Fuente

Los módulos de los prototipos así como el del marco de desarrollo se han liberado bajo una licencia considerada de código abierto. El código fuente se encuentra en repositorios públicos de la plataforma github:

- Marco de Desarrollo de Aplicaciones OpenSARA: [frameworksara/openSARA](https://github.com/frameworksara/openSARA).
- Módulo de nodos medicamentos: [giseproi/OpenHealthVademecum](https://github.com/giseproi/OpenHealthVademecum).
- Módulos de Prototipo OpenSITEM: [giseproi/openSITEM](https://github.com/giseproi/openSITEM).

Los repositorios son de acceso público y están administrados por el equipo de desarrollo. Los integrantes de este equipo se encargan de analizar las contribuciones para agregarlas a las ramas que vayan surgiendo. Con la entrega de este informe y la transferencia desde el servidor git privado a github, se está promoviendo la participación de nuevos actores que contribuyan a mejorar y hacer crecer el sistema.

Capítulo 2

Antecedentes

A principios del milenio, el grupo GITEM realizó un estudio para abordar problemas puntuales en la implementación de servicios médicos prestados a través de medios teleinformáticos en el Distrito Capital[Aparicio,2000]:

“En el momento no existe un diagnóstico real sobre los servicios requeridos en el área de telemedicina, razón suficiente para iniciar un trabajo de campo que establezca la situación actual de servicios médicos y la demanda real, así como la posibilidad de conocer a corto, mediano y largo plazo cuáles serían los costos de inversión que permitirían dar soluciones al problema de cobertura.

La socialización del conocimiento alrededor de las tecnologías aplicadas al desarrollo de la medicina, es uno de los valores que lleva al éxito de soluciones efectivas en el sector salud, por tal motivo es necesario desarrollar un plan de alfabetización en el sector salud y en el sector gubernamental y académico.

En el país no existen estrategias de investigación en esta área del conocimiento para llevar a cabo un estudio real que permita dar el paso a soluciones verdaderas sobre desarrollo tecnológico o experimental para poder implementar centros de investigación en Telemedicina.

La Universidad Distrital tiene el recurso humano, el conocimiento y la experiencia científica y tecnológica, capaz de dar soluciones tangibles a estas necesidades; unida al conocimiento y experiencia de entidades como clínicas y hospitales y con la participación de operadores de comunicaciones, puede desarrollar soluciones efectivas a los problemas de salud que afronta la sociedad colombiana.”

Los resultados obtenidos fueron recopilados en extensos tomos en formato físico y digital. Luego de los primeros análisis, se observó que los reportes no tenían una estructura documental coherente ni un modelo de información que los caracterizara. Este hecho, a la par con el ingreso de entidades al estudio, aumentó la complejidad a la hora de generar estudios comparativos o de apoyo a la toma de decisiones. Se evidenciaron casos de información faltante

o innecesaria para el proyecto. Como consecuencia, al ser el objeto de estudio intrínsecamente dinámico, cualquier cambio de las condiciones iniciales no se ve reflejado y la información queda desactualizada rápidamente. En ese escenario, la labor de articular información de las organizaciones suponía un proceso lento por lo cual la estrategia de gestión de información empleada en el estudio comenzó a mostrar debilidades.

Respondiendo a este contexto problemático se creó al interior del grupo un proyecto denominado **Sistema para la Caracterización de Nodos Potenciales de Redes de e-salud**, que en sus primeras fases de desarrollo ha dado solución parcial al marcar las pautas hacia la integración de información para el **GITEM**. En paralelo el grupo de investigación implementó, en convenio con *Colciencias*, el Sistema de Referencia y Contrarreferencia para el Distrito Capital, utilizando herramientas de desarrollo propietarias específicamente el middleware .NET de Microsoft con lo que el grupo adquirió experiencia en el desarrollo de aplicaciones siguiendo metodologías efectivas para la construcción de software. Así las cosas, el estudio base generó una *instantánea* de la capacidad de la red de salud del distrito para el emprendimiento de proyectos de eSalud pero no se logró un sistema que permitiera *observar* la evolución de dicha capacidad - o tomar otras instantáneas a través del tiempo.

2.1. Sistema para la Caracterización de Nodos Potenciales de Redes de e-salud

El proyecto que se describe en este documento surge de la necesidad de contar con un espacio común de información que permita valorar la capacidad de las instituciones que pertenecen a una red de salud y ofrecer un mecanismo de observación, intercambio, análisis y colaboración entre equipos de consultoría y diseño de redes de eSalud. El proyecto - cuyo nombre de desarrollo es OpenSITEM, pretende brindar una plataforma tecnológica para que grupos de trabajo especializados puedan integrar (cargar, almacenar, analizar, categorizar, comparar) los datos que les permitan responder inquietudes tales como:

- ¿cuáles son las necesidades de servicios de la población que atiende la institución?
- ¿cuáles son las características de las redes eléctricas, de telecomunicaciones, de servicios médicos, de profesionales y de equipos médicos de la institución?
- De un grupo seleccionado de instituciones, ¿cuál es la de mayor potencial para la prestación de un servicio de eSalud?
- ¿cuál es la capacidad de un proveedor de telecomunicaciones para atender la demanda de un determinado servicio en una zona específica?

OpenSITEM se presenta como un sistema federado de aplicaciones de código abierto que, en conjunto, ofrecen herramientas para contestar estas u otras preguntas relacionadas. Si bien el

desarrollo de cada componente no se ha abordado por el grupo, la contribución innovadora radica tanto en la arquitectura como en el motor de integración que conducen a un modelo de utilidad que permite la interoperabilidad novedosa de aplicaciones preexistentes en un nuevo dominio de negocio, con potencial de ser utilizado en procesos de diseño y planeación de sistemas de eSalud.

El proyecto ha pasado por varias fases que han correspondido a etapas dentro de un proceso de investigación emergente. El producto ha evolucionado de un simple sitio web construido 100 % *in - house*, a una solución federada - centrada en una arquitectura y guiada por casos de uso - que recoge los aspectos de calidad definidos por cada uno de sus componentes y del motor de integración. A continuación se describen brevemente cada una de las fases:

2.1.1. SITEM – Fase I

El objetivo de esta fase fue la definición de un modelo de información que permitiera la gestión de resultados del estudio de campo. A partir de dicho modelo se presentó la recomendación de definir un sistema informático que permitiera gestionarlo. Con los resultados de esta fase se encontró *un costo de oportunidad adecuado debido a que en la actualidad ningún portal se especializa en el proceso de diagnóstico de capacidad para la implementación de proyectos de eSalud.*

Los alcances y logros efectivos de esta fase fueron:

- Descripción base del Modelo de Información.
- Propuesta de Desarrollo de portal especializado.
- Bosquejo de la Arquitectura general del Sistema.
- Integración del SITEM dentro de los proyectos del grupo, entendiéndolo como una plataforma tecnológica para soportar grupos de trabajo que constituyan un *observatorio* multitemporal de los nodos que constituyen las redes de salud, con iras a servir de referente para la medición de capacidad para el despliegue - o seguimiento, a proyectos de eSalud.
- Estudios sobre filosofía de Software Libre y el movimiento del código abierto.
- Construcción de un Prototipo de baja funcionalidad conocido como SITEM versión 0.1, bautizada internamente como **Kauil**. Este prototipo se construyó totalmente por los integrantes del grupo y aunque estuvo al aire durante dos años nunca alcanzó una versión estable ni sus características de calidad fueron evaluadas,

Para la construcción del prototipo se experimentó un proceso de desarrollo ágil considerando los principios de Kanban [Anderson,2016]. Con algo de esfuerzo se apropiaron los valores y

algunas técnicas de trabajo propuestas por el método. Sin embargo, ciertas especificidades del proyecto condujeron a un entorno de riesgo que no fue posible gestionar. Principalmente se priorizaron como principales causas:

- Escasos recursos de personal, financieros y técnicos. Durante esta fase el grupo de investigación priorizó el estudio de campo por lo cual SITEM tenía asignado solo dos integrantes.
- Competencias y habilidades no homogéneas entre participantes. Esto condujo a inversión de tiempo en capacitación, aún cuando un supuesto era que al tener estudiantes de últimos semestres estos tendrían formación en ingeniería de software y desarrollo de sistemas.
- Bajo tiempo de permanencia de los integrantes. Con un promedio de retención inferior a 0.5 años y un nivel de inmersión promedio de 2 horas/día.
- Inexistencia de mecanismos de transferencia de conocimiento. Teniéndose como técnica única la capacitación basada en clase magistral por parte de integrantes experimentados.
- Incorrecta disciplina de requisitos y arquitectura.
- Enfoque en donde el código fuente se consideraba el único producto. Esta visión impidió que el grupo de trabajo tuviese un conjunto de artefactos que evolucionara de manera consistente.

2.1.2. SITEM – Fase II

Conscientes de la oportunidad de negocio se evaluaron los resultados y experiencias de la primera fase. Un análisis de riesgo permitió definir estrategias de tratamiento que incluyeron los siguientes aspectos:

- Adopción de un método de desarrollo guiado por casos de uso, centrado en la arquitectura, iterativo e incremental.
- Fortalecimiento del modelo de transferencia de conocimiento a partir de la creación de objetos de aprendizaje.
- Uso de marco de trabajo (frameworks).
- Definición de lineamientos de trabajo.
- Enfoque en la interoperabilidad.
- Limitación del dominio de negocio a los nodos de entidades de salud en sus redes eléctricas y de telecomunicaciones.



Figura 2.1: Fases transcurridas en el desarrollo del SITEM

- Liberación del código para facilitar el desarrollo colaborativo.

Con la experiencia se comprobó la máxima de Larman [Larman,2003] : “Rápido, barato, bueno: elija dos cualquiera”. Dejando a un lado las esperanzas utópicas, sacrificamos la rapidez en que versiones estables del proyecto habrían de ver la luz.

La metodología permitió elegir a OpenUP como el método de trabajo y se declaró la fase II como etapa para desarrollar una visión base, planificar el proyecto, preparar el ambiente de trabajo, definir el modelo base de requisitos y lograr acuerdos acerca del enfoque técnico que debe regular el desarrollo.

Los alcances de esta fase fueron:

- Elaboración de la visión general del proyecto.
- Aprobación del plan general del proyecto para las fases II (inicio), III (elaboración y construcción) y IV (transición).
- Modelo base de Requisitos.
- Bosquejo de la Arquitectura del Sistema.
- Modelo de Información - Segunda Versión.
- Definición de los requisitos mínimos del ambiente de trabajo.

2.1.3. OpenSITEM – Fase III

La fase III se corresponde a las fases de elaboración y construcción definidas en OpenUP [Balduino,2010]. Para concretar la visión el grupo toma una decisión arquitectónica importante y concentra su esfuerzo en un motor de integración de aplicaciones, cuyo objetivo es la federación de sistemas existentes de software libre o código abierto (FLOSS). De esta manera se logra un mayor nivel funcional del que se había poder alcanzado de seguir con un desarrollo 100 % in-house. Con esto se busca el aseguramiento de la calidad en el desarrollo, la interoperabilidad, escalabilidad y el uso extensivo de soluciones consolidadas de software libre o de código abierto.

Para recalcar la adopción de la filosofía del código abierto todo el código generado se gestiona en un repositorio público en la plataforma GitHub. De manera complementaria y considerando el producto de manera integral, se elaboran diferentes artefactos y se formaliza las áreas de capacitación a partir de ciclos genéricos de transferencia de conocimiento apoyados en tecnologías de la información.

Es precisamente esta fase la que da vida a este documento, a una arquitectura emergente y una versión 1.0 del sistema denominada *OpenSITEM*, con la que el grupo entrega un sistema



Figura 2.2: Sistema de Información para la Caracterización de Proyectos de eSalud. Interfaz de Usuario en la Fase III

complejo constituido por un motor de federación (basado en OpenSARA, un framework de desarrollo para aplicaciones web desarrollado por el grupo) que articula las herramientas externas Knowage, ERPNext, Alfresco Community, CAMUNDA y OpenProject, así como varias herramientas propias tales como:

- Sistema Integrado de Evaluación
- Sistema de Gestión de Redes de Práctica y Colaboración
- Sistema de agentes notificadores y de recomendación
- Sistema de Información Geográfica 3D.

De esta forma OpenSITEM propone una solución que abarca diferentes dominios pero con capacidad de adaptación para el propósito específico de las necesidades del proyecto *Sistema Federado de Aplicaciones para la Caracterización de Proyectos de e-salud*.

Capítulo 3

Contexto Teórico

El grupo GITEM considera que dos aspectos del proyecto son de impacto para el ámbito colombiano: el modelo de dominio e información para la caracterización de nodos potenciales de redes de eSalud; y el modelo de federación de aplicaciones que permite el despliegue de soluciones de alta funcionalidad a bajo coste, aprovechando - y colaborando con, el movimiento de Software Libre y Código Abierto (FLOSS).

Este capítulo provee la base teórica que sustenta los modelos de dominio e información, el marco legislativo y demás conceptos que son primordiales para determinar las características (atributos, indicadores, relaciones, axiomas, categorías) que definen un nodo dentro de una red de eSalud. Teniendo en cuenta el carácter evolutivo del proyecto software y como referencia para posteriores desarrollos soportados en OpenSITEM, se incluye el proceso de ingeniería que guía la integración del sistema.

3.1. Redes de eSalud

Conjunto de nodos relacionados con capacidad operativa de brindar servicios de eSalud, garantizando atributos de calidad, estandarización, interoperabilidad, privacidad y seguridad [Wilson, 2017]. La arquitectura base de cada nodo de la red es de tipo agregado con *micronodos* agrupados funcionalmente para brindar un servicios son consumidos por otros nodos, creando estructuras funcionalmente complejas. En este sistema los nodos de más alto nivel de abstracción son los que tienen capacidad de ofrecer servicios de eSalud.

De acuerdo a las recomendaciones del grupo GITEM, la descripción de la arquitectura de cada nodo - así como la de la red, puede obtenerse de manera emergente utilizando métodos tales como el Architecture Development Method (ADM) de The Open Group o similares. Se recalca que OpenSITEM no es una plataforma para prestación de servicios de eSalud sino para la descripción de nodos de la red - a diferentes niveles de abstracción, a partir del modelo

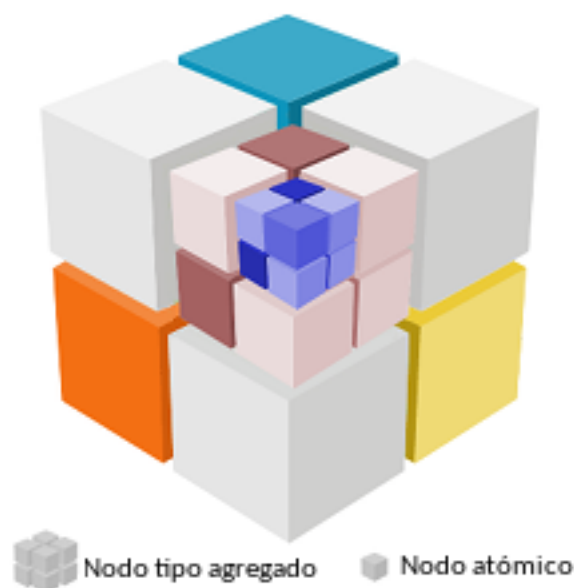


Figura 3.1: Vista conceptual de la arquitectura de una red de eSalud en OpenSITEM

de agregación.

3.1.1. Caracterización de Nodos para Redes de eSalud

Las redes de eSalud incluyen nodos de diferentes tipos, por ejemplo, médicos, tecnológicos, de recurso humano, financiero, estratégico, de organización, de política y de infraestructura [OPS, 2011],[OMS, 2016], [ITU-OMS, 2012]. En el estudio de campo el equipo de diseño y planeación de redes de eSalud, se encontró con el obstáculo de no disponer de información de calidad de los nodos potenciales que harían parte de la red. Por esta razón y con el objetivo de crear una matriz de comparación, se propuso un *reducido* de nodos¹ y una descripción de la arquitectura de cada uno de ellos, de tal manera que se pudiese recopilar datos basados en modelos de información.

Al definir el resultado se planteó que un servicio - con posibilidad de ser implementado en un modelo de eSalud, requiere de la integración de nodos. En OpenSITEM existen dos tipos de nodos: (1) atómicos: aquellos que ofrecen un servicio cohesivo que no abarca un caso de uso de dominio y (2) agregados: que articulan nodos (atómicos o agregados) para brindar capacidad funcional a nivel de caso de uso de negocio. Si se considera que el modelo de dominio gira en torno a los servicios de eSalud, el nodo agregado que lo provee se denominará *macronodo*.

Un nodo atómico está caracterizado por atributos (propiedades) e interfaces. Estas últimas

¹El reducido es un conjunto finito de nodos con capacidad de entregar una funcionalidad contractualmente definida relacionada con la eSalud



Figura 3.2: Componentes en un Sistema de eSalud. Fuente: OMS/ITU

son los puntos de acceso en donde los servicios del nodo son expuestos para ser consumidos [theopengroup,2016]. Un nodo agregado está definido por el conjunto de atributos de los nodos que lo componen y las interfaces que surgen de la interoperabilidad de dichos nodos. Un nodo puede pertenecer a uno o más agregados razón por la cual un nodo - atómico o agregado, goza de alta cohesión y bajo acoplamiento como patrón principal².

Otro aspecto que se utiliza para la descripción de los nodos es la categoría o clase a la cual pertenece. Debido a que la arquitectura de la red puede ser abordada desde distintos puntos de vista, un nodo puede tener asociada uno o varias categorías. Por ejemplo, el modelo de componentes que propone la OMS?? podría definir las categorías de clasificación de los nodos, posterior a ello si surge otro modelo o se edita el actual, los nodos podrían adaptarse para mantener las relaciones y crear unas nuevas. El grupo de investigación desarrolla OpenSITEM enfocado en la caracterización multicategoría de los nodos en redes de salud, tomando como modelo de mayor nivel de abstracción en propuesto por la OMS y uno de baja abstracción con categorías: médica, tecnológicas, administrativa, humana y financiera. Sin embargo, debido al perfil de profesionales que han participado en el desarrollo, los nodos de la categoría tecnológica han sido mejor caracterizados hasta el momento.

3.1.2. Subsistemas en Una Red de eSalud

El grupo GITEM considera que una red de eSalud está compuesta por cuatro subsistemas:

- **Subsistema de Captura de datos:** Conformado por los dispositivos de hardware, los protocolos y aplicaciones software que trabajan conjuntamente para transformar información médica en datos susceptibles de ser administrados usando técnicas digitales.

²En el mayor grado en que se cumplan estos patrones más simple resulta la caracterización de la red.

- **Subsistema de Transmisión de Datos:** Hacen parte de este subsistema los dispositivos de hardware, las tecnologías de interconexión, los protocolos y aplicaciones que permiten estructurar redes de transmisión de datos digitales de una manera fiable en tiempos aceptables para un servicio específico.
- **Subsistema de Gestión de Información:** Dispositivos de hardware - computadores, sistemas de almacenamiento masivo, etc; y sistemas de información que almacenan, procesan, distribuyen, analizan, integran y producen información con base en los datos de los subsistemas de captura, históricos y de pronóstico.
- **Subsistema de Despliegue de información:** Elementos de hardware (pantallas, transductores, sistemas de audio, etc), aplicaciones software y protocolos asociados que permiten recibir y reproducir información médica.

Todos ellos sirven como plataforma para determinar cierta organización para la definición de tipos de nodos.

3.2. eSalud

Teniendo en consideración que el servicio de más alto nivel de la red es la eSalud, es necesario abordar su concepto con el fin de tener un marco para la comprensión del modelo de dominio que se presentará en un capítulo posterior.

La eSalud es definida por la organización Mundial de la Salud (OMS) [OMS, 2016] como

el apoyo que la utilización costo eficaz y segura de las tecnologías de la información y las comunicaciones ofrece a la salud y a los ámbitos relacionados con ella, con inclusión de los servicios de atención de salud, la vigilancia y la documentación sanitarias, así como la educación, los conocimientos y las investigaciones en materia de salud

, en si es un término que define un campo multidisciplinar que integra componentes de diferentes áreas del saber que incluye entre otros a la medicina, la ingeniería electrónica, la telemática, la informática, la ingeniería de sistemas, la inteligencia artificial, la biónica, la psicología, la sociología, la antropología, las geociencias, entre muchas otras. Las redes eSalud consideran elementos que van mucho más allá del simple despliegue de redes tecnológicas de intercomunicación y se plantean como redes de interacción social cuyo objetivo primario - más no el único, es la prestación de servicios médicos y relacionados.

Dependiendo el grado en que se presente cada uno de los elementos mostrados en la figura 3.3 y de la mayor o menor correlación entre ellos, se pueden crear sistemas de eSalud que se acerquen al ideal de proveer servicios de salud de alta calidad.



Figura 3.3: Modelo simple de categorías para nodos de una red de eSalud.

La OMS y la Unión Internacional de Telecomunicaciones (ITU) proveen [ITU-OMS, 2012] un método que podría ser utilizado para abordar el desarrollo de proyectos de eSalud complejos, dinámicos y evolutivos.

3.2.1. Componentes Principales de eSalud

La eSalud comprende áreas que no tienen una división concreta pero se enfocan en ciertos aspectos de interés:

- Telemedicina: Provisión remota de servicio clínicos
- Telesalud: Telemedicina, complementada con la prestación remota de otros servicios tales como entrenamiento médico, monitoreo de pacientes, cuidado médico, gestión ad-

ministrativa, etc.

- mSalud: telesalud con el apoyo de dispositivos móviles.
- Registro Médico electrónico: Conocido como Historia Clínica Electrónica. Comprende los mecanismos que permiten registrar en un entorno digital seguro, la información sobre los eventos de salud de cada paciente.
- eAprendizaje/eEnseñanza: Servicios de enseñanza/aprendizaje de ciencias médica y afines, en modalidad a distancia o virtual.

Cabe la pena aclarar que la pluralidad de componentes no es más que un esfuerzo para abordar en un único modelo, todas las tendencias que se han presentado en la evolución de la eSalud. En alguna literatura los términos son intercambiables dependiendo el enfoque del autor. [OMS, 2010].

3.3. Marco Legislativo y Normativo

OpenSITEM es un sistema que gestiona información sobre diferentes áreas del saber de acuerdo a las categorías de los nodos que se definan. Cada aspecto (atributos, interfaces, interoperaciones) está en un marco legislativo y normativo concreto, o está asociado a estándares y normas de uso extendido o de facto aceptadas en el mundo. La información de OpenSITEM que sea de acceso público no podrá incluir protección por derechos de autor que restrinjan su difusión [Ley 565,2000],[Ley 23,1982]. Debido a esto, el código fuente de OpenSITEM es cubierto por una licencia abierta que se ciñe a la normatividad expresada en la Ley 565 de 2000: adopción del Tratado de la OMPI sobre Derechos de Autor y complementarias para garantizar que todos los aspectos tanto técnicos como conceptuales estén debidamente registrados [Ley 565,2000],[Ley 44,1993],[Dec 1360,1989].

OpenSITEM es una plataforma para la definición de nodos y no es posible *a priori* definir el marco legislativo que regirá cada nodo o categoría. Sin embargo, se describe a continuación un marco relacionado con las categorías base que se han definido en el primer modelo del sistema.

3.3.1. En el Ámbito de los Servicios Médicos

El Derecho a la Salud ha sido reconocido por normas y pactos internacionales contenidos en tratados sobre Derechos Humanos, Económicos, Sociales, y Culturales (DHESC). Esos acuerdos han sido ratificados por Colombia para su cumplimiento como un derecho de los ciudadanos. “La Corte Constitucional; ha señalado que el inciso segundo del artículo 93 de la Carta Política confiere rango constitucional a todos los tratados de derechos humanos,

económicos, sociales y culturales, ratificados por Colombia y referidos a derechos que ya aparecen en la Carta” [Corte1319,2001] como ocurre con el Derecho a la Salud.

Al Ministerio de Salud y Protección Social, le corresponde expedir las normas técnicas y administrativas de obligatorio cumplimiento para las Entidades Promotoras de Salud del régimen contributivo, las Instituciones Prestadoras de Salud del Sistema General de Seguridad Social en Salud, las Administradoras del Régimen Subsidiado y para las Direcciones Seccionales, Distritales y Locales de Salud en cuanto al objetivo de cumplimiento en el desarrollo de actividades de protección específica, detección temprana y atención de enfermedades de interés en Salud Pública.

A continuación se registra la normatividad que se tuvo en cuenta al momento de definir los componentes actuales del subsistema de servicios médicos en cuanto a la relevancia que se tiene tanto para la proyección de nuevas redes de eSalud, como para apoyar los sistemas básicos ya existentes. Es de anotar que lo contemplado en las leyes nacionales es, en su mayoría, derivado de normas internacionales que han sido objeto de detallados estudios y reconocidas técnicamente con base en las experiencias vividas por los profesionales de esta área.

Ley 1751 de 2015 . Por medio de la cual se regula el derecho fundamental a la salud. Es una ley estatutaria que surge a partir de la debacle del proceso de reforma y tiene como efecto positivo el elevar a la salud como un derecho fundamental. Entró en rigor a partir del año 2017 y da lineamientos para reestructurar el sistema de salud a partir del desarrollo de *redes de servicios* públicos, privados o mixtos. También declara la necesidad de establecer políticas relacionadas con la salud tales como la política para la información, la política de innovación, ciencia y tecnología; y la política farmacéutica nacional.

Ley 1419 de 2010 . Por la cual se establecen los lineamientos para el desarrollo de la tele-salud en Colombia. Define las redes de telesalud y el aprendizaje en telesalud como ejes principales de la gestión del conocimiento en salud. Si bien esta ley obliga a desarrollar el mapa de conectividad, aún en el 2017 no se encuentra uno que esté disponible para los ciudadanos.

Resolución 2182 de julio 9 de 2004 Con esta resolución se definían las Condiciones de Habilitación para las instituciones que prestan servicios de salud bajo la modalidad de Telemedicina. Fue derogada por el artículo 11 de la **resolución 1043 de 2006**, con la cual se establecen las condiciones que deben cumplir los Prestadores de Servicios de Salud para habilitar sus servicios e implementar el componente de auditoría para el mejoramiento de la calidad de la atención y se dictan otras disposiciones.

Posteriormente, con la **Resolución 1448 de 8 de Mayo de 2006** se regula la prestación de servicios de salud bajo la modalidad de telemedicina y establece las condiciones de habilitación de obligatorio cumplimiento para las instituciones que prestan servicios de salud. Esta resolución aclara que las actuaciones de los médicos en el ejercicio de la prestación de servicios bajo la modalidad de telemedicina se sujetarán a las disposiciones establecidas en la **Ley 23 de 1981** y demás normas que la reglamenten, modifiquen, adicionen o sustituyan.

Resolución 4678 de noviembre 11 de 2015 Con esta resolución, modificada por la Resolución 1113 de 2017, el Ministerio de Salud y Protección Social adopta una Clasificación Única de Procedimientos en Salud (CUPS) la cual "...corresponde a un ordenamiento lógico y detallado de los procedimientos y servicios en salud que se realizan en el país, en cumplimiento de los principios de interoperabilidad y estandarización de datos utilizando, para tal efecto, la identificación por un código y una descripción validada por los expertos del país." [MINSALUD-4678,2015]. La Clasificación Única de Procedimientos en Salud adaptación para Colombia, se implementa por **Resolución 365 de 1999**. Su primera publicación se presenta en un solo volumen que contiene la Lista Tabular y el Índice Alfabético. A partir de dicha resolución se realizó la primera actualización de la CUPS (1°A-CUPS) mediante la **Resolución 2333 de 2000**. En el año 2016, mediante la Resolución 3804, se establece el procedimiento para la actualización de la CUPS, con lo que el Ministerio espera darle una mayor agilidad al proceso.

Resolución 1830 de junio 23 de 1999 adopta para Colombia, "Las codificaciones únicas de especialidades en salud, ocupaciones, actividades económicas y medicamentos esenciales" para el Sistema Integral de Información del SGSSS - SIIS

Resolución 1895 de noviembre 19 de 2001 por la cual se adopta para la codificación de morbilidad en Colombia, La Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud - Décima revisión.

Considerando que en la 43a. Asamblea Mundial de la Salud llevada a cabo en 1990, fue aprobada por la Conferencia Internacional la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud - Décima revisión -, (CIE-10) en la cual Colombia no expresó objeciones y adquirió el compromiso de implementarla. Resuelve Adoptar para la codificación de morbilidad en Colombia, la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud -Décima revisión-, contenida en la publicación científica No.554 de la Organización Panamericana de la Salud, presentada en tres volúmenes: V1. Lista de Categorías; V2 Manual de Instrucciones; V3 Índice Alfabético.

Resolución 1995 de julio 8 de 1999 por la cual se establecen normas para el manejo de la Historia Clínica.

La Historia Clínica es un documento de vital importancia para la prestación de los servicios de atención en salud y para el desarrollo científico y cultural del sector, *es un documento privado, obligatorio y sometido a reserva*, en el cual se registran cronológicamente las condiciones de salud del paciente, los actos médicos y los demás procedimientos ejecutados por el equipo de salud que interviene en su atención. Dicho documento únicamente puede ser conocido por terceros previa autorización del paciente o en los casos previstos por la ley.

Circular 015 de abril 4 de 2002 estándar de historias clínicas y registros, establece la obligatoriedad de definir procedimientos para utilizar una historia única institucional. Ello implica que la institución cuente con un mecanismo para unificar la información de cada paciente y su disponibilidad para el equipo de salud. No es restrictivo en cuanto

al uso de medio magnético para su archivo, y sí es expreso en que debe garantizarse la confidencialidad y el carácter permanente de registrar en ella y en otros registros asistenciales.

Decreto 2092 de 2 de Julio de 1986 , Por el cual se reglamenta parcialmente los Títulos VI y XI de la **Ley 09 de 1979**, en cuanto a la elaboración, envase o empaque, almacenamiento, transporte y expendio de Medicamentos, Cosméticos y Similares. Se dan las Disposiciones Generales y Definiciones, el Registro Sanitario de Medicamentos, Cosméticos y Similares.

3.3.2. En el Ámbito del Desarrollo de Redes Teleinformáticas

Documento CONPES 3072 Aunque no es una norma regulatoria, es una declaración oficial del gobierno colombiano acerca de la necesidad de fomentar las Tecnologías de la Información para potenciar la absorción, creación y divulgación del conocimiento por medio del desarrollo sostenible en las infraestructuras física, de información y social. Según [CONPES3072,2000]: “... para que el país pueda ofrecer un entorno económico atractivo y participar en la economía del Conocimiento, resulta indispensable desarrollar una sociedad en la que se fomente el uso y aplicación de las Tecnologías de la Información. A través de estas Tecnologías, se puede efectuar un salto en el desarrollo en un tiempo relativamente breve, mucho menor del que se necesita para superar el déficit de infraestructura física.”.

El documento CONPES brinda un referente válido pues la mayoría de los objetivos estratégicos del OpenSITEM contienen el espíritu expresado en diferentes partes del mismo.

Documento CONPES 3582 Política Nacional de Ciencia, Tecnología e Innovación. En el cual se enfatiza el desarrollo de la salud y la tecnología como mecanismos de generación de valor social.

Resolución 087 de 1997 “Por medio de la cual se regula en forma integral los servicios de Telefonía Pública Básica Conmutada (TPBC) en Colombia.” En donde claramente se expresa que:

Los servicios de TPBC deberán ser utilizados como instrumento para impulsar el desarrollo político, económico y social del país con el objeto de elevar el nivel y la calidad de vida de los habitantes en Colombia. Los servicios de TPBC serán utilizados de manera responsable para contribuir a la defensa de la democracia, a la promoción de la participación de los colombianos en la vida de la Nación y la garantía de la dignidad humana y de otros derechos fundamentales consagrados en la Constitución Política, y para asegurar la convivencia pacífica.

Esta resolución presenta particular importancia para la extracción de elementos semánticos y algunos componentes necesarios en las redes de telecomunicaciones basadas en telefonía conmutada. Algunas heurísticas usadas en el subsistema de consultoría también se basan en apartes de esta resolución.

Manual de Calidad de Servicio “Con este Manual de calidad de servicio (QoS) se especifican los parámetros de calidad de servicio de red que permiten el suministro de servicios a los clientes y los usuarios, satisfaciendo sus expectativas de calidad de servicio. Estos parámetros tienen que ver tanto con la implementación del servicio como con su utilización continua. Asimismo, la calidad de servicio se relaciona con todos los aspectos relativos a la evaluación y gestión de las redes.” [ITU-T, 2004]

Sus principales aspectos son recogidos en [CRT,QoS,2006] y [CRT,Indicadores,2006] las cuales sirven de base para el proyecto de resolución [CRT,QoS,2007] de la Comisión de Regulación de Comunicaciones y que especifica entre otros: las definiciones relativas a la QoS, parámetros de medición, variables y propiedades técnicas de diferentes enlaces. Aunque el objetivo principal de este manual es el de garantizar la QoS en un sistema de telecomunicaciones dado es importante notar que sus indicaciones deben ser tenidas en cuenta al momento de proyectar los servicios de telecomunicaciones en un sistema de telemedicina dado. En estos aspectos también se considera dentro del modelado de ciertos componentes del OpenSITEM las recomendaciones de la UIT en [G1000, 2001] y [G1010, 2001], las cuales aún no tienen un equivalente en la normatividad colombiana pero son de uso extendido alrededor del orbe.

3.3.3. En el ámbito del Tratamiento de Datos e Información

Ley 1581 de 2014 y Ley 1266 de 2008 . Estas leyes están relacionadas con la protección de los datos personales y el aseguramiento de la privacidad de la información personal.

3.4. Aspectos Claves en la Gestión y Dirección del Conocimiento

Como se aclaró anteriormente, OpenSITEM es una federación de diferentes tipos de sistemas software y sirve de soporte para el trabajo de los equipos de diseño de redes de eSalud del grupo de investigación GITEM (o similares). En esa línea, parte de su funcionalidad está enfocada en brindar apoyo a los flujos de trabajo de comunidades de práctica y por tanto es necesario que se aborden aspectos relacionados con la gestión del conocimiento. A continuación se presenta la aproximación conceptual de esta área.

3.4.1. Acerca del conocimiento

En la presente investigación conocimiento es entendido como un concepto que encierra, entre otras, las siguientes definiciones y se usarán de acuerdo a su contexto como partes complementarias de una misma meta – definición.

De acuerdo con Gunter Dueck [Dueck,2001], el conocimiento es un concepto que puede tener

componentes en una o varias de las siguientes dimensiones:

- **Episteme** - *Dimensión abstracta – o metafísica*, en forma de generalizaciones, bases, leyes y principios científicos.
- **Phronesis** - *Dimensión práctica*, relativa al conocimiento pragmático discernido a través de las practicas aceptadas por la sociedad.
- **Techne** - *Dimensión técnica*, relativa a la forma de hacer las cosas, de la realización de actividades concretada en la forma de manuales, procedimientos y comunidades de práctica.
- **Metis** - *Dimensión objetiva*, como forma de volver corpóreo, real y sustancial la conjugación de los otros tipos de conocimiento.

Esta concepción multidimensional extiende y explica la taxonomía del conocimiento dada por los griegos:

“El conocimiento incluye restricciones implícitas y explícitas entre objetos (entidades), operaciones y relaciones, que permiten recoger heurísticas generales y específicas así como los procedimientos de inferencias relacionados con la situación a modelar”. [Sowa,1984]

Otros autores despojan del sentido filosófico y colocan su definición en un plano simple y utilitario: “El conocimiento es información organizada y analizada para hacerla comprensible y aplicable a la resolución de problemas y toma de decisiones”. [Turban,1992]. Si bien esta es una definición reduccionista, sirve de base para las propuestas de representación de conocimiento en documentos XML - comúnmente denominadas *ontologías*.

3.4.2. Ciclo de Conocimiento

Múltiples factores deben ser considerados cuando se trata de capturar, crear y diseminar el conocimiento dentro de un grupo de personas. La no homogeneidad en los medios de almacenamiento de conocimiento es uno de ellos. El conocimiento - en cualquiera de sus formas, puede estar guardado en diferentes partes que van desde entidades biológicas - mente humana, genes; a repositorios de conocimiento estructurado tales como ontologías, grafos de relación o mapas mentales.

Otro factor importante es la capacidad de acceso al conocimiento que es muy diferente al mero hecho de acceder a una fuente de conocimiento dada. En [Nokata,1995] se considera que el conocimiento puede estar en dos estadios con propiedades diferentes: tácito y explícito.

- **Conocimiento tácito:** Este conocimiento se corresponde con el conocimiento obtenido a través de la experiencia, conocimiento simultáneo y conocimiento análogo. Esta forma

de conocimiento usualmente se encuentra en medios de almacenamiento biológicos como la mente humana.

- **Conocimiento explícito:** Se corresponde con el conocimiento racional, conocimiento secuencial, y conocimiento digital y se encuentra almacenado en documentos, bases de conocimiento, ontologías o cualquier otro medio abstracto de representación.

En [Liebowitz,1998] se establece un tercer estadio llamado conocimiento implícito.

- *Conocimiento Implícito:* Acceso directo mediante consulta y discusión. Requiere la localización y comunicación previa de conocimiento informal.

Un *Ciclo de Conocimiento* es el proceso por el cual el conocimiento se transforma de tácito a explícito y viceversa por medio de las siguientes actividades:

- **Socialización:** Compartir conocimiento tácito entre individuos. El conocimiento permanece siendo tácito sin ser transformado en explícito. Este tipo de patrón no es muy interesante debido a su naturaleza tácita (Tácito - Tácito).
- **Articulación:** Alguien transforma el conocimiento tácito en explícito (Tácito - Explícito).
- **Síntesis:** Combinación de conocimiento explícito para crear nuevo conocimiento explícito (Explícito - Explícito).
- **Interiorización:** Proceso de transformar conocimiento explícito en tácito (Explícito - Tácito).

El flujo de conocimiento organizacional más importante es la transformación del conocimiento tácito en explícito [Davies,2011], esto es, la *articulación* que se apoya en procesos avanzados de socialización. Esto permite acumular conocimiento explícito que puede ser compartido y accedido por los miembros de la organización. Por el contrario, la interiorización es el proceso natural llevado a cabo a través del aprendizaje individual por parte de los integrantes de la organización, esto es, la asimilación de conocimiento.

El tercer flujo de conocimiento relacionado con la transformación de conocimiento es la *combinación* o síntesis. En este caso se transfiere conocimiento a otra forma explícita de conocimiento. Un ejemplo sería el cambiar el formato de una base de conocimiento, agrupar ontologías o refinar heurísticas. Este tipo de flujo de conocimiento es importante para seleccionar, combinar y distribuir el conocimiento existente con diferentes fines. Por ser quizás el flujo de conocimiento más formal en OpenSITEM algunos componentes específicos implementan flujos de síntesis de conocimiento.

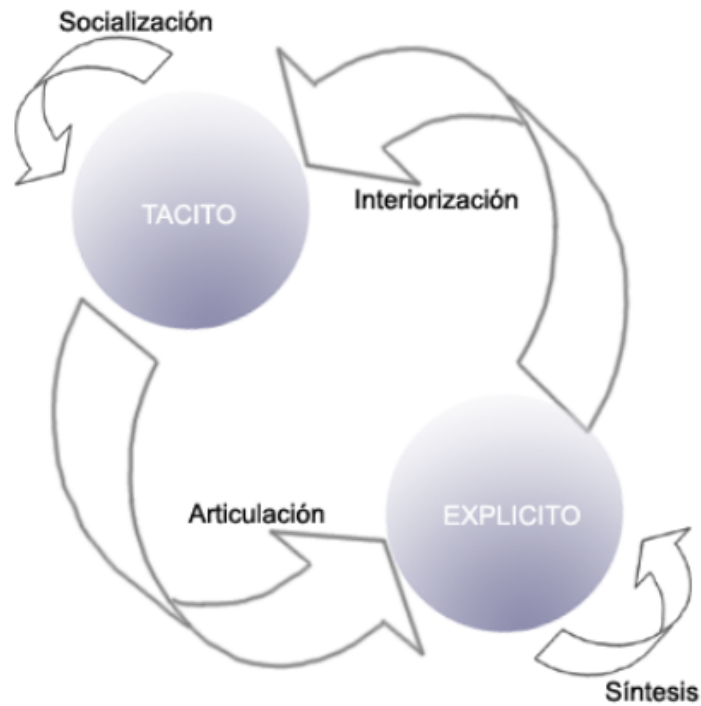


Figura 3.4: Ciclo de Conocimiento

El cuarto flujo del ciclo de vida del conocimiento permite transformar conocimiento tácito en otras formas de conocimiento tácito mediante procesos de socialización. Un ejemplo de esto es cuando se transfiere conocimiento tácito de un experto a un ingeniero de conocimiento en una entrevista personal.

3.4.3. Tecnologías del Conocimiento

En la actualidad el conocimiento se considera un activo fundamental y como lo expresa [Gang,2007], tiene dos propiedades de vital importancia para contribuir al desarrollo de las organizaciones o comunidades de práctica:

- **Es explicable.** Cuando no se evidencia esta propiedad el conocimiento permanece tácito y en ninguna medida puede considerarse como perteneciente a la comunidad o la organización. Es entonces una tesis que solo aquel conocimiento que se ha convertido en explícito es el que posee una organización; de otra manera es propiedad exclusiva del individuo.
- **Se puede comunicar y compartir.** Cuando no se toman medidas que formalicen estas actividades el conocimiento se pierde. Es decir, la organización o comunidad debe

tener procesos conocidos que garanticen flujos de conocimiento basados en la síntesis, interiorización y socialización.

En general para que el conocimiento pueda generar ventajas competitivas debe ser gestionado de alguna manera. Esta necesidad dio surgimiento a dos áreas de investigación comúnmente agrupadas bajo el concepto de Tecnologías de Conocimiento: Ingeniería de Conocimiento y Gestión de Conocimiento.

Uno de los primeros problemas que deben atacar las tecnologías de conocimiento es el asociado con el Modelado de Conocimiento. En [Benafia,2016] se expresan algunos principios a tener en cuenta cuando se modela el conocimiento:

- **Definición de roles de conocimiento.** El conocimiento se puede dividir en unidades atómicas que tienen propiedades irreductibles y que se asocian para lograr funciones que identifican dicha unidad.
- **Identificación de tipos de conocimiento.** Cada unidad de conocimiento debe enmarcarse en uno o varios de los siguientes tipos de conocimiento: de tareas, inferencial, del dominio, ontologías del dominio, modelos del dominio.
- **Capacidad de ser compartido y reutilizado.** Las unidades de conocimiento pueden ser expresadas usando lenguajes y reglas formales. Lo que permite que pueda ser entendido por entidades con roles de conocimiento definidos.
- **Uso de modelos gráficos.** Las unidades de conocimiento pueden ser representadas mediante grafos tipo red en los cuales tanto los nodos como las rutas de interconexión son unidades atómicas de conocimiento.

3.4.4. Gestión de conocimiento

La gestión del conocimiento es de esos conceptos *polisémicos* que no pocos autores pretenden consolidar en una sola definición [Girard,2015]. En este proyecto partimos de una visión sistémica que lo trascienda a lo que en investigación holística se comprende como sintagma[Hurtado,2000]. La *gestión de conocimiento*, como se percibe en la presente investigación, es “una unión sintagmática de diversos paradigmas” [Hurtado,2000]. Fue *Karl Wiig*, quien usó el término de *gestión de conocimiento* por primera vez durante una conferencia en Suiza y a partir de ese momento diversos autores han conceptualizado el término surgiendo definiciones parciales tales como:

“La Gestión de Conocimiento es la construcción y aplicación sistemática, explícita y deliberada de conocimiento para maximizar la efectividad organizacional con respecto al conocimiento, por lo que usa sus activos de conocimiento” [Wiig,1993]

“La Gestión de Conocimiento es el proceso de capturar experiencia colectiva organizacional donde ésta resida (por ejemplo, bases de datos, documentos, mentes humanas) y su distribución allá donde pueda ayudar a mejorar los resultados” [Hibbard,1997]

“La Gestión de Conocimiento es la gestión y control explícito del conocimiento en una organización para lograr los objetivos de la organización” (van der Spek and Spijkervet, 1997)

Consecuente con la concepción de la gestión de conocimiento como un sintagma se puede concebir un paradigma asociado a un proceso con ciertas actividades implicadas:

- Identificación y mapeo de bienes intelectuales de la organización.
- Generación de conocimiento nuevo que permita obtener una ventaja competitiva.
- Recopilación accesible de información organizacional.
- Compartir buenas prácticas y tecnología, incluyendo técnicas de trabajo en grupo.

Este paradigma explora el conocimiento técnico, pragmático y objetivo considerando que la conjugación de leyes rígida e epistemológicas van en contravía de la dinámica misma de los sistemas. La articulación de este conocimiento debe ser mantenido de alguna forma en la organización, y de ahí surge el concepto de Memoria Corporativa. El saber hacer está completamente diseminado en la organización y debe ser integrado de forma coherente para facilitar el acceso al mismo y su reutilización, esto es, expresarlo en forma de memoria corporativa.

Las memorias corporativas se consideran un elemento clave para gestionar el conocimiento porque facilitan su conservación, distribución y reutilización. Van Heijst define la memoria corporativa como una “representación de conocimiento e información organizacional explícita y persistente”, mientras que en (Nagenda y Plaza, 1996) se define como “los recursos colectivos de datos y conocimiento de una compañía, incluyendo experiencias en proyectos, experiencia en resolución de problemas, etc”. En (Abecker, 1998), una memoria corporativa es referida como “un contenedor que integra información contextual, documentos e información no estructurada, que facilita su uso y reutilización”.

Sistemas de Gestión de Conocimiento

En IA, las bases de conocimiento son generadas para ser consumidas en sistemas expertos y basados en conocimiento, donde las computadoras usan inferencias para responder a cuestiones de usuario. Aunque es importante la adquisición de conocimiento para inferencias computacionales, en los desarrollos más recientes en Gestión del Conocimiento, el conocimiento queda disponible para consumo humano directo o para desarrollar software que procese dicho conocimiento.

Históricamente, la Gestión de Conocimiento se ha centrado en un único grupo a través de lo que generalmente se ha conocido como sistema de información ejecutiva (EIS), que contiene un conjunto de herramientas para acceder a bases de datos, generar alertas, etc para apoyar el proceso de toma de decisiones. Más recientemente, se ha comenzado a diseñar sistemas de Gestión de Conocimiento para organizaciones completas. Si los ejecutivos necesitan acceder a la información y al conocimiento, es probable que sus empleados tengan interés en esa información.

De acuerdo con (O'Leary, 1999) citado por (Valencia, 2005), las principales funciones de un sistema de gestión de conocimiento son facilitar:

- La conversión de datos y texto en conocimiento;
- La conversión de conocimiento individual y de grupo en conocimiento explícito;
- La conexión de individuos y conocimiento a otros individuos y conocimientos;
- La comunicación de información entre diferentes grupos;
- La creación de nuevo conocimiento útil para la organización;

Sistemas Integrados para el Soporte de Desempeño

Sistemas que integran múltiples fuentes y herramientas de gestión de conocimiento en un único ambiente de trabajo para apoyar de una manera más efectiva las tareas relacionadas con: (Winslow and Bramer, 1994)

- **Infraestructura:** Organización y estructura del entorno de trabajo.
- **Control:** Monitorización, coordinación y control.
- **Navegación:** Interacción hombre-máquina.
- **Presentación:** Posibilidad para personalizar datos y servicios.
- **Adquisición:** Captura conocimiento, casos, opiniones, aprendizaje y datos sensoriales en diferentes medios y su transformación en formato interno.
- **Consultoría:** Consultar servicios, asistencia y recordatorios.
- **Instrucción:** Ayuda y entrenamiento.
- **Aprendizaje:** Aplicación de técnicas de descubrimiento de conocimiento y minería de datos.
- **Evaluación:** Valoración y certificación basada en medidas del rendimiento y la calidad.

- **Referencia:** Constituyen fuentes de conocimiento y experiencia para la organización.

Estos sistemas se han convertido en la actualidad en una necesidad debido al crecimiento desmesurado de aplicaciones incompatibles y protocolos no estandarizados.

3.5. Ingeniería de Software

A medida que el marco conceptual del OpenSITEM crece, el cúmulo de nuevos requerimientos - no vislumbrados en su planteamiento inicial, fomenta que los riesgos asociados al desarrollo de los componentes también crezcan. Lo que en un principio no era más que una "herramienta" para la administración del acervo documental fruto de una investigación, se convirtió en una propuesta de sistema de información y conocimiento que suponía un reto novedoso al interior del grupo de investigación.

Es evidente que se deben sumar nuevos saberes para procurar manejar formalmente el proceso de elaboración del sistema: un punto inicial y obligado de estudio se centró en la ingeniería de software.

Como rama de la ingeniería comparte la definición fundamental que de la misma brindó a mediados del siglo pasado el **Consejo de Ingenieros para el Desarrollo Profesional - ECPD**, por sus siglas en inglés; y que en general propone que:

Ingeniería Es la aplicación creativa de principios científicos para el diseño o desarrollo de estructuras, máquinas, aparatos, procesos de manufactura o sistemas genéricos, para ser usados de forma independientemente o combinados; o la construcción y operación de los mismos con total conocimiento de su diseño; o el pronóstico de su comportamiento bajo ciertas condiciones de operación; todo aquello respecto a una funcionalidad esperada asegurando economía en el manejo de los recursos y con seguridad para la vida y la propiedad.

3

Se concibe entonces a la ingeniería de software como la aplicación de los principios de ingeniería a los sistemas de software con base a "un acercamiento sistemático, cuantificable y disciplinado del desarrollo, operación y mantenimiento de software" [IEEE, 1990]; y ciertamente se fundamenta en actividades interrelacionadas, propias del ser humano cognoscente y creativo que en [Brugge, Dutoit, 2000] se identifican como:

- **Actividades de Modelado:** Para abstraer la complejidad del dominio del problema en unidades factibles de ser objeto de estudio y análisis. En este contexto las nociones de

³Adaptación de la definición hecha por los autores.

contratos funcionales e independencia conceptual juegan un rol importante. Se pretende con estas actividades obtener modelos de análisis - como representaciones relativamente simples de la realidad y modelos de diseño - como representaciones del dominio de la solución de un problema dado, representado con un modelo de análisis.

- **Actividades de Resolución de problemas:** Siendo la ingeniería de software un proceso guiado de búsqueda de solución a un problema específico del ser humano que es viable de ser apoyado por sistemas software. Se concibe actualmente como un proceso investigativo que, de acuerdo a un acercamiento holístico, contempla flujos de trabajo continuos y evolutivos de exploración, descripción, análisis, comparación, explicación, predicción, proposición, modificación, confirmación y evaluación
- **Actividades de Adquisición de conocimiento:** Durante el desarrollo del sistema software el ingeniero, a partir de un modelo constructivista, recrea constantemente su conocimiento tácito a partir de las nuevas experiencias y el mayor conocimiento del dominio del problema así como de los diferentes paradigmas usados en la consecución de soluciones óptimas. En realidad el ingeniero, así como los demás actores que intervienen con el sistema, ven revalidados o reformados sus conocimientos a medida que los requerimientos son cumplidos y los riesgos minimizados.

También contempla actividades propias de trabajo colaborativo que producen integración de saberes en ambientes inter, trans y multidisciplinarios, lo que potencia efectivamente la creación de ciclos de conocimiento que contribuyen al refinamiento continuo del sistema - como objeto perfectible, y del conocimiento directo, que tanto del sistema como del proceso, tienen los actores vistos como sujetos perfectibles, racionales y cognoscentes; dentro de una dinámica de retroalimentación entre el sujeto que crea el sistema y el sistema mismo.

3.5.1. Proceso de Desarrollo del Software

Un proceso de desarrollo de software puede ser visto como el conjunto de actividades que deben realizar un grupo de personas para dar solución - mediante un sistema software- a un problema cuyas características y condiciones de resolución han sido especificadas. El producto final, el software, es un **sistema** o sea un conjunto de componentes funcionales que se relacionan por medio de interfaces definidas logran el objetivo común de solucionar los problemas determinados.

Para determinar el proceso más apropiado según las necesidades y especificidades del proyecto se condujo una metodología ⁴ centrada en diferentes modelos ampliamente aceptados en el campo de la ingeniería de software que al final dio lugar a un proceso consolidado de guía para el desarrollo del Sistema de Información para Proyectos de Telemedicina. Se aclara que este modelo, como el sistema y los actores, no es indiferente del proceso evolutivo de

⁴La metodología no fue exhaustiva y se limitó a un grupo muy reducido de elementos cuya caracterización se basó exclusivamente en indicadores de tipo cualitativo. Se recomienda remitirse a [McCarty, 2006], [Pressman, 2006], [Jacobson, 2000], [Koch, 2005] - entre otros, para detalles de los diferentes procesos.

adaptación de conocimiento por lo que en realidad no se considera como una fórmula mágica sino simplemente como un caso específico que aporta unos lineamientos interesantes para otros proyectos de software similares y sirve de base para los ingenieros de proceso de fases posteriores en el ciclo de vida del macroproyecto SITEM. En últimas, un sistema software exitoso es aquél en el cual todos sus componentes se refinan constantemente y el proceso de desarrollo es un componentes nuclear que tiene mayor incidencia.

Existen tantos procesos de desarrollo de sistemas en el mundo que la mera enumeración taxativa podría cubrir cientos de páginas. El dilema de cual es el mejor de ellos es irresoluble, sin embargo se puede definir las características óptimas para un contexto en particular teniendo en cuenta múltiples indicadores a partir de aspectos tales como:

- Tamaño del Grupo de Desarrollo
- Presupuesto
- Límites de tiempo.

3.5.2. Principios de Diseño y Desarrollo

A través del tiempo se han decantado ciertas prácticas que son reconocidas como las más óptimas cuando se trata de construir un sistema software de gran magnitud - tanto en líneas de código, como en funcionalidad y recursos involucrados. Estos principios son tenidos en cuenta independientemente del proceso de desarrollo que se siga. Quizás los de mayor difusión son los *patrones GRASP* - acrónimo de General Responsibility Assignment Software Patterns, que se basan en la asignación precisa de responsabilidades a cada uno de los componentes del sistema software.⁵

En el desarrollo del SITEM se recomienda, como estrategia para mantener la calidad del software, que los integrantes del grupo tengan en cuenta y adquieran competencias en el manejo de los siguientes patrones y principios: ⁶

- **Modularidad.** Para facilitar las tareas de mantenimiento, depuración e incremento en la funcionalidad, se requiere que el sistema se implemente con base en componentes que presente propiedades de *alta cohesión* funcional entre sus elementos y tengan *bajo acoplamiento* entre sí. La alta cohesión funcional tiene en cuenta que el componente realiza solo tareas relacionadas y utiliza un conjunto de datos homogéneo. El bajo acoplamiento

⁵Aún el Object Management Group declara el uso de ciertos principios de diseño en el desarrollo del metamodelo que especifica a UML

⁶Debido a que el proceso general adoptado contiene elementos del Desarrollo de Software de Código Abierto [Koch, 2005], no siempre se obtiene un seguimiento preciso de los patrones por parte de todos los participantes. Refinamientos sucesivos y estrategias de capacitación se despliegan al interior del grupo para incrementalmente llegar a este objetivo.

se refiere al hecho de que un componente se relaciona con otro a través de una interfaz estable y definida; dicha relación no está supeditada a la implementación interna de ninguno de los componentes, con bajo acoplamiento un cambio en un componente no requeriría ningún cambio en la implementación del componente asociado.

- **Prueba continua.** Todos los módulos y sus componentes deben ser probados en cuanto su funcionalidad y el cumplimiento de los demás principios y patrones. Las actividades de prueba podrán ser automatizadas o realizadas manualmente pero en cualquier caso deben ser formalmente documentadas. Es una recomendación que en lo posible el personal de prueba sea diferente a aquel que ha diseñado, o construido, el componente.
- **Codificar claramente.** La forma en que se ingresa el código o se agrupa un conjunto de elementos gráficos en un diagrama deben ser hecha de tal forma que se facilite su comprensión. Se recomienda el uso de comentarios para aquellas partes del código cuya funcionalidad no sea evidente o cuando se evite el tener que analizar piezas de código extensas.
- **Abstracción Funcional por capas.** Los diferentes componentes del SITEM deberán centrar su funcionalidad en tres capas principales: datos, aplicación e interfaz. Los unidades que manejen cada una de las capas deben propender por conservar la modularidad.
- **Reutilización.** Los diferentes componentes del SITEM - denominados bloques dentro del modelo de desarrollo, deben estar codificados de tal forma que puedan ser fácilmente adaptados en los diferentes módulos sistema.
- **Re-creación de componentes.** Se debe conocer la estructura interna de un determinado componente para poder sugerir mejoras. Este principio no pretende desplazar a la reutilización sino que debe complementarlo. El contexto definirá cual de los dos deberá ser usado. El tiempo transcurrido desde la creación y la cantidad de uso del componente son indicadores a tener en cuenta.
- **Controlar las versiones.** Debe mantenerse un repositorio que permita recuperar los estados anteriores de cualquier componente dentro del sistema. El incremento general en la funcionalidad, el refinamiento en el desempeño y la experiencia adquirida al desarrollar el software es información que permanece latente en los repositorios. Los repositorios integrados permiten mantener la sincronización de los grupos de trabajo y blindan el hilo estable - “oficial” - de los hilos secundarios en desarrollo o depuración.
- **Documentar.** Ya sea empotrado dentro del código, usando lenguajes de modelado o en artefactos independientes se deben documentar las actividades interesantes que se realicen en el desarrollo del sistema. La documentación debe usar estándares multi-plataforma para que pueda ser transparentemente visualizados, editados y compartidos entre los integrantes del equipo de desarrollo.

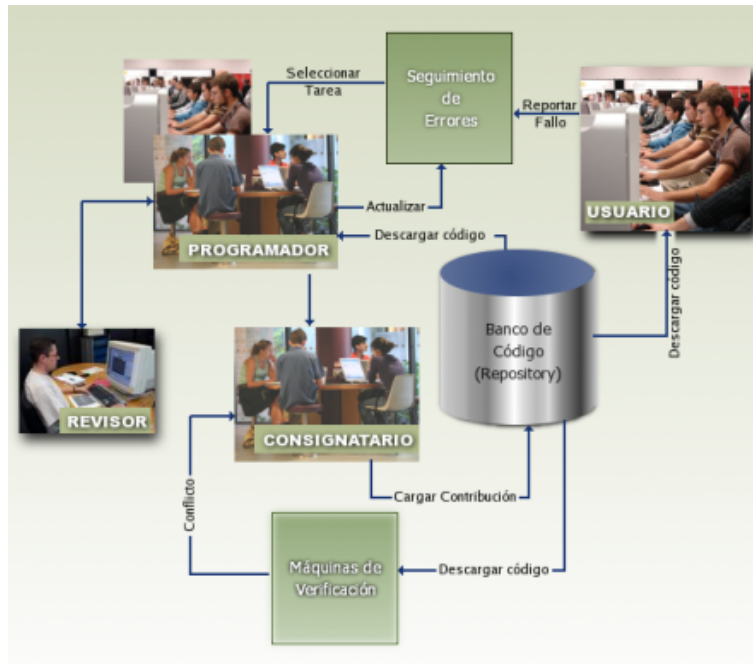


Figura 3.5: Conceptos Básicos en el Desarrollo de Software de Código Abierto

3.5.3. Proceso de Desarrollo de Software de Código Abierto

Es indiscutible el papel preponderante que tiene la planificación en el desarrollo de cualquier tipo de sistema, sin embargo, no debe olvidarse que cuando se requiere solucionar un problema no basta con el mero seguimiento de una receta y es aquel “toque” único que brinda el ser humano el que hace que los sistemas de software se diferencien unos de otros. No es por casualidad que en nuestro medio el software se considera un producto que esta cubierto por la misma legislación que las obras literarias o musicales.

Apelando a ese recurso intangible llamado pasión, que aún hoy solo es característico de los seres humanos, se ha extendido en los últimos años el proceso de desarrollo de software de código abierto. Todos los paradigmas que las grandes empresas de desarrollo de software se han encargado de poner como las *mejores prácticas*, se han reevaluado, trastocado, pisoteado y sin más ni más un gran cisma apareció en el horizonte. Este renacimiento moderno surge como respuesta humana al gran vacío de satisfacción de necesidades que brindaba el software a principios de los años noventa.

Más que una forma de realizar sistemas, se trata de una visión revolucionaria en torno al software como patrimonio de la humanidad y se une a la filosofía del software libre que se expresa en [stallman,2002]:

“Software Libre” se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere

a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2).
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.

Las aplicaciones más representativas del mundo del software libre como Apache, Mozilla, MySQL, PostgreSQL y el mismo sistema operativo Linux, luego de sus etapas primarias, adoptaron como proceso de desarrollo uno que contravenía en gran manera los fundamentos del control riguroso y ponía el futuro del sistema en manos de la anarquía ⁷. Tal como lo propone Linus Torval "la idea es liberar versiones de prueba rápido a menudo, delegar cuanto sea posible, estar abierto hasta el punto de resultar promiscuo".

Algunos principios fundamentales en este tipo de desarrollo los expone [Raymond, 2010]:

1. Todo buen trabajo de software comienza a partir de las necesidades personales del programador. (Todo buen trabajo empieza cuando uno tiene que rascarse su propia comezón).
Esto podría sonar muy obvio: el viejo proverbio dice que "la necesidad es la madre de todos los inventos". Empero, hay muchos programadores de software que gastan sus días, a cambio de un salario, en programas que ni necesitan ni quieren. No ocurre lo mismo en el mundo Linux; lo que sirve para explicar por qué se da una calidad promedio de software tan alta en esa comunidad.
2. Los buenos programadores saben qué escribir. Los mejores, qué reescribir (y reutilizar).
... una importante característica de los grandes programadores es la meticulosidad con la que construyen. Saben que les pondrán diez no por el esfuerzo, sino por los resultados; y que casi siempre será más fácil partir de una buena solución parcial que de cero.
3. Contemple desearlo; de todos modos tendrá que hacerlo. cita encontrada en el capítulo 11 de libro The Mythical Man-Month escrito por el célebre Fred Brooks.

⁷Definida en su sentido positivo como la situación humana en donde es innecesaria e indeseable la autoridad, lo que conlleva a una sociedad libre basada en el respeto mutuo de sus miembros y la cooperación voluntaria entre individuos.[BCE,2003]

Diciéndolo de otro modo: no se entiende cabalmente un problema hasta que se implementa la primera solución. La siguiente vez quizás uno ya sepa lo suficiente para solucionarlo. Así que si quieres resolverlo, prepárate a empezar de nuevo al menos una vez.

4. Si tienes la actitud adecuada, encontrarás problemas interesantes.
5. Cuando se pierde el interés en un programa, el último deber es heredarlo a un sucesor competente.
6. . Tratar a los usuarios como colaboradores es la forma más apropiada de mejorar el código, y la más efectiva de depurarlo.
7. Libere rápido y a menudo, y escuche a sus clientes.
8. Dada una base suficiente de desarrolladores asistentes y beta-testers, casi cualquier problema puede ser caracterizado rápidamente, y su solución ser obvia al menos para alguien.

Dicho de manera menos formal, con muchas miradas, todos los errores saltarán a la vista”. A esto lo he bautizado como la Ley de Linus.

9. Las estructuras de datos inteligentes y el código burdo funcionan mucho mejor que en el caso inverso.

De nuevo Fred Brooks, Capítulo 11: “Muéstreme su código y esconda sus estructuras de datos, y continuaré intrigado. Muéstreme sus estructuras de datos y generalmente no necesitaré ver su código; resultará evidente.”

En general un proceso de desarrollo de software libre se basa en el hecho de que el programa puede ser instalado y el código fuente está disponible para cualquier persona. Es decir, la ausencia de barreras en cuanto a la limitación en el uso hace que muchas personas interesadas en la funcionalidad que brinda el software lo descarguen y empiecen a utilizarlo. Dando inicio al siguiente ciclo:

1. El grupo inicial de programadores mantiene un sitio en la red para obtener retroalimentación de los usuarios los cuales reportan fallos, disfuncionalidades y solicitan nuevas características.
2. Un desarrollador - que puede ser uno de los usuarios, revisa la lista de reportes y decide trabajar en uno específico; para tal efecto descarga la última versión del código fuente la modifica y la envía a un revisor para que este convalide la contribución.
3. La contribución se agrega al código fuente generando una nueva versión del sistema. Esto se realiza sincronizando el código fuente de desarrollo con aquél existente en la bodega de código fuente - repository, la cual normalmente es gestionada por un programa para el control de versiones.
4. Si en algún momento dos programadores están realizando modificaciones a la misma porción de código y pretenden sincronizarlas ocurre un conflicto que deberá ser resuelto siguiendo reglas definidas que habitualmente contemplan el bloqueo de la versión más reciente, el aviso para resolución entre desarrolladores que causan el conflicto o el descarte de las contribuciones.

De esta forma se va refinando el software siguiendo el ciclo mostrado en la figura 3.5. El grupo de desarrollo se ve aumentado cuando usuarios expertos empiezan a proponer y realizar cambios directos en el código; cuando uno de ellos demuestra tener el suficiente interés y respeto hacia los intereses del software se le asigna el permiso para escribir directamente en la bodega de código.

La creación de la documentación así como de los modelos de requerimientos, análisis, diseño y despliegue siguen el mismo proceso.

Métodos Ligeros

Ha principios del milenio un grupo de experimentados desarrolladores, entre los que se encontraban Kent Beck, Alistair Cockburn, Martin Fowler y Dave Thomas, redactaron un manifiesto en el que consignaban los elementos de mayor importancia dentro del desarrollo de sistemas software [Beck,1999]:

Nosotros estamos descubriendo mejores formas de desarrollar software dado que lo creamos y ayudamos a otros a realizar esta tarea. Por medio del trabajo de desarrollo hemos encontrado de gran valor elegir:

- *Individuos e interacciones* sobre *procesos y herramientas*
- *Software ejecutable* sobre *documentación profusa*
- *Colaboración del Cliente en el desarrollo* sobre *contrato de negocios*
- *Respuesta al cambio* sobre *ceñirse a un plan*.

Mientras que existe valor en los elementos de la derecha nosotros valoramos más los elementos de la izquierda.

Con esto sentaban las bases para el despliegue de nuevos métodos de realizar software agrupados bajo el nombre genérico de “ágiles”⁸ que se contraponían a los métodos y procesos tradicionalmente rígidos y altamente planificados.

En [Koch, 2005] se expresan claramente las razones del porqué se desarrollan estos métodos y sus principales características. Los métodos ágiles nacen como respuesta del desarrollador puro al ambiente altamente industrializado y burocrático en el cual transcurren la mayoría de proyectos de desarrollo de software. En estos ambientes es típico el riguroso control que sobre el cumplimiento de cronogramas, planes de trabajo y presupuestos mantienen los denominados *ingenieros de proceso*. El enfoque tradicional se basa en la planificación con la que se trata de *predecir* desde las primeras etapas todos los pormenores del ciclo de desarrollo.

⁸Siendo por definición un método caracterizado por ser liviano y ligero

Debido a que los métodos tradicionales tienen fundamento en la ingeniería civil y mecánica, tratan de mitigar los riesgos poniendo un especial interés a las actividades de modelo en especial en las etapas de análisis y diseño; en general relegan a los desarrolladores a etapas de construcción erróneamente consideradas de *cero esfuerzo* intelectual. Los requisitos del software se tratan de fijar desde los inicios del desarrollo, firmandose usualmente un contrato de aceptación de los mismos por parte del cliente. Estos métodos tradicionales siguen los lineamientos de aseguramiento de la calidad por la cual los procesos son eficientemente documentados, controlados, auditados, vigilados y mejorados. Todos esos aspectos hacen que el elemento clave sea el proceso y se relega a segundo plano el crear productos que en realidad aporten un nuevo valor al cliente.

Para atacar la abrumadora complejidad que añade el proceso al sistema de software, los métodos ágiles proponen cambiar el paradigma *predictivo* - rígido y resistente al cambio; por uno adaptativo que sea flexible y reaccione rápidamente antes cambios inesperados en los requisitos del software. Aquellos que han desarrollado un software de mediana o alta complejidad conocen de primera mano el hecho de que los requisitos no son estáticos, ellos cambian, evolucionan se transforman ya que en sí, no son sino abstracciones de necesidades del mundo real y este no es estático sino que se caracteriza por una fuerte dinámica.

El *cliente también debe ser adaptable* en el sentido de que la mayoría de las veces los requerimientos del sistema los va descubriendo a medida que interactúa con él. Una de las premisas de los procesos ágiles es el mantener un contacto permanente con el cliente e involucrarlo en todas las fases del desarrollo. Con esto se logra que el cliente obtenga un software que realmente cope sus intereses y (el cliente) sea consiente de los costos asociados al desarrollo del mismo.

Así como los requisitos cambian durante el desarrollo también lo hacen los recursos y el escenario en el cual se desenvuelve el equipo de trabajo. Para atacar esta característica de los sistemas software, se recomienda *aferrarse a un presupuesto global* pero distribuyéndolo en pequeños presupuestos que solventen las tareas que a corto plazo realizan los involucrados en el desarrollo.

Es claro con lo expuesto hasta ahora que el enfoque es considerar el desarrollo como una *"carrera de 100 metros planos"* y no como una maratón. En tal sentido se deben gestar planes a corto plazo cuyo objetivo principal sea generar versiones del sistema que puedan ser probadas, corregidas e incrementalmente adicionadas en funcionalidad. Cada plan transcurre en lo que se denomina una iteración la cual usualmente no supera el mes de duración - algunos recomiendan una duración de dos semanas o menos.[Beck,1999]

Otra característica de los métodos clásicos, y que atacan los métodos ágiles, es aquella en la cual se considera a las personas como recursos intercambiables mediante la definición de *roles* con funciones específicas y predictivas. Esto hace que las personas - cuyo comportamiento es poco predecible y no lineal [Cockburn, 1999], tengan una moral baja y disminuya su productividad; en el mejor de los casos trabajan con esfuerzo y, si sus condiciones son excelsas, rápidamente abandonan el grupo perdiéndose un activo intangible que repercute negativa-

mente en la calidad global del sistema. Para los metodólogos ágiles el desarrollo se centra en las personas más que en los procesos, considera a cada miembro del grupo como un *ser creativo e irremplazable*, esto genera un gran cambio en cuanto al método: *no es estático* ni recetario. Evoluciona, se recrea, se adapta y se concerta dentro el grupo de trabajo.

Evidentemente el proceso de desarrollo de software libre maneja los principios promulgados por los métodos ágiles los cuales encuentran quizás su máxima expresión en la Programación Extrema [Beck,1999].

Proceso Unificado

Según lo expresa [Alhir,2003]:

El Proceso Unificado (UP) es un proceso de desarrollo de software basado en componentes dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental...que utiliza la especificación UML dada por el Object Management Group (OMG) para preparar los esquemas del sistema. El Proceso Unificado es aplicable a diferentes tipos de sistemas de software, incluyendo proyectos de pequeña y larga escala; proyectos que tengan varios grados de complejidad técnica y administrativa, a través de diferentes dominios de aplicación y culturas organizacionales.

El PU nace de la unificación, en 1995, de la aproximación sugerida por Rational Software Corporation y el proceso orientado a objetos de la empresa Objectory AB. Se puede considerar al Proceso Unificado como un modelo de ciclo de vida del proyecto que incluye contexto, colaboraciones e interacciones. El UP es documentado totalmente en el libro “The Unified Software Development Process” escrito por Booch, Rumbaugh y Jacobson, y publicado por Addison- Wesley en 1999.

Un sistema desde que nace hasta que muere repite el Proceso Unificado en ciclos de desarrollo constituidos por fases secuenciales cuyo objetivo es la producción incremental de liberaciones del sistema, llamadas comúnmente como generaciones del sistema. Cada una de las fases se convierte en un hito principal y esta constituido por pequeños microprocesos denominados *iteraciones*. Habitualmente la numeración de las fases se hace de acuerdo a números enteros mientras que las iteraciones se hacen en números decimales.

Las fases para el desarrollo de proyectos en el Proceso Unificado son cuatro [Jacobson,2000], a saber:

Fase de Concepción. También conocida como de inicio. Se centra en el establecimiento de las fronteras, ámbitos, riesgos asociados y visión del proyecto. Determina la viabilidad y los objetivos del proyecto. En esta fase podría tenerse una arquitectura general del sistema que esboze los subsistemas más importantes.

Fase de Elaboración. Se enfoca en la determinación de la arquitectura y requisitos del sistema; de esta forma se establece su viabilidad técnica. Durante esta fase se construyen los casos de uso críticos y se obtiene una arquitectura refinada del sistema.

Fase de Construcción Es en la que se crea la mayor funcionalidad del producto, finaliza con cierta capacidad operativa. Se centra en la construcción del sistema y la arquitectura del sistema se considera estable.

Fase de Transición . Concluye con la liberación del producto, centrándose en la transición o distribución del sistema a la comunidad o usuario final.

Dentro de las iteraciones el grupo de trabajo deberá distribuir sus esfuerzos en áreas estratégicas que conduzcan a la mitigación temprana de los riesgos, estas áreas son conocidas dentro del PU como disciplinas, figura 3.6:

- **Disciplina de administración de cambios en la configuración**, la cual se centra en la administración de la configuración del sistema y de las peticiones de cambios en la misma.
- **Disciplina de administración de proyecto.**
- **Disciplina de ambiente** que se centra en los ambientes de desarrollo del proyecto, incluyendo los procesos y las herramientas.
- **Disciplina de modelado del negocio**; focalizado en la comprensión del negocio que esta siendo automatizado por el sistema capturando dicho conocimiento en un modelo del negocio.
- **Disciplina de requerimientos**, necesaria para entender los requerimientos del sistema que automatiza el negocio y captura dichos requerimientos en un modelo de casos de uso.
- **Disciplina de diseño y análisis**, centrada en analizar los requisitos y diseñar en sistema capturando tales conocimientos en un modelo de análisis/diseño.
- **Disciplina de implementación** para la implementación del sistema basado en el modelo de implementación.
- **Disciplina de pruebas** que maneja las pruebas (evaluaciones) del sistema comparándolos con los requerimientos basándose primordialmente en el modelo de pruebas.
- **Disciplina de distribución** encargada de la distribución del sistema basado en el modelo de distribución.

Durante la etapa de concepción la mayoría del esfuerzo está distribuido a través del modelo del negocio y la disciplina de requerimientos.

Durante la fase de elaboración el esfuerzo se distribuye entre las disciplinas de implementación, diseño, análisis y requerimientos. Durante la etapa de construcción el esfuerzo se distribuye entre las disciplinas de análisis, diseño, implementación y pruebas.

En la fase de transición el esfuerzo se distribuye a través de las disciplinas de prueba y distribución. Obviamente las disciplinas de soporte se distribuyen entre todas las cuatro fases. El objetivo general es producir el sistema, por lo tanto todas las disciplinas nucleares están comprometidas tanto como sea posible para no introducir riesgos en el proceso; esto es, los practicantes son los responsables de determinar cuales disciplinas comprometer y en que momento hacerlo.

En este punto es necesario definir varios conceptos:

- **riesgo** en el Proceso Unificado se concibe como un obstáculo para alcanzar el éxito en la ejecución de una actividad, este riesgo puede estar determinado por características del negocio, humanas o técnicas.
- **Iteración** es un paso o rama a través de un ruta hasta cierto destino. Dicho de otra forma es un movimiento planeado que puede ser evaluado para demostrar un progreso tangible dentro de una actividad o proceso, además, de acuerdo a lo citado por [Zabala, 2000]: “Una iteración es iterativa en el aspecto de que es un acto repetitivo que propende la mejora continua del trabajo. Aditiva en el caso de que el resultado es siempre superior al alcanzado con un solo trabajo y paralela ya que el trabajo puede ser concurrente dentro de la iteración.”

Cuando se decantan los *requerimientos*, aquello que el sistema debe cumplir, se está declarando explícitamente los casos de uso los cuales, dado que el Proceso Unificado está manejado por ellos, determinan las iteraciones. De la misma forma en que los casos evolucionan en el marco de las disciplinas regidas por un proceso iterativo, los sistemas evolucionan constantemente con base en iteraciones realizados en el marco de su arquitectura. Incluyendo en la arquitectura todos los elementos, sus colaboraciones e interacciones.

Resulta pues obvio que la iteración, dado que es un avance demostrable, tiende en últimas a reducir los riesgos inherentes a cada una de las etapas del proceso de desarrollo. Esto ha sido definido por [Alhir,2003] en su artículo:

.. de esta forma las iteraciones confrontan los riesgos derivados de los casos de uso y la arquitectura para alcanzar el éxito en el proyecto, buscando en todo momento reconciliar las fuerza técnicas y del negocio. Una iteración esta acotada en el tiempo con inicio y final fijos en donde una colección de colaboraciones son planeadas, ejecutadas y evaluadas del tal forma que en todo momento se pueda demostrar progreso en el proceso... un caso de uso evoluciona a través de un gran número de iteraciones y a través de cualquier número de disciplinas nucleares en una iteración. La experiencia y aprendizaje obtenido en una iteración evidentemente conduce la aplicación de las próximas iteraciones dentro del proceso..

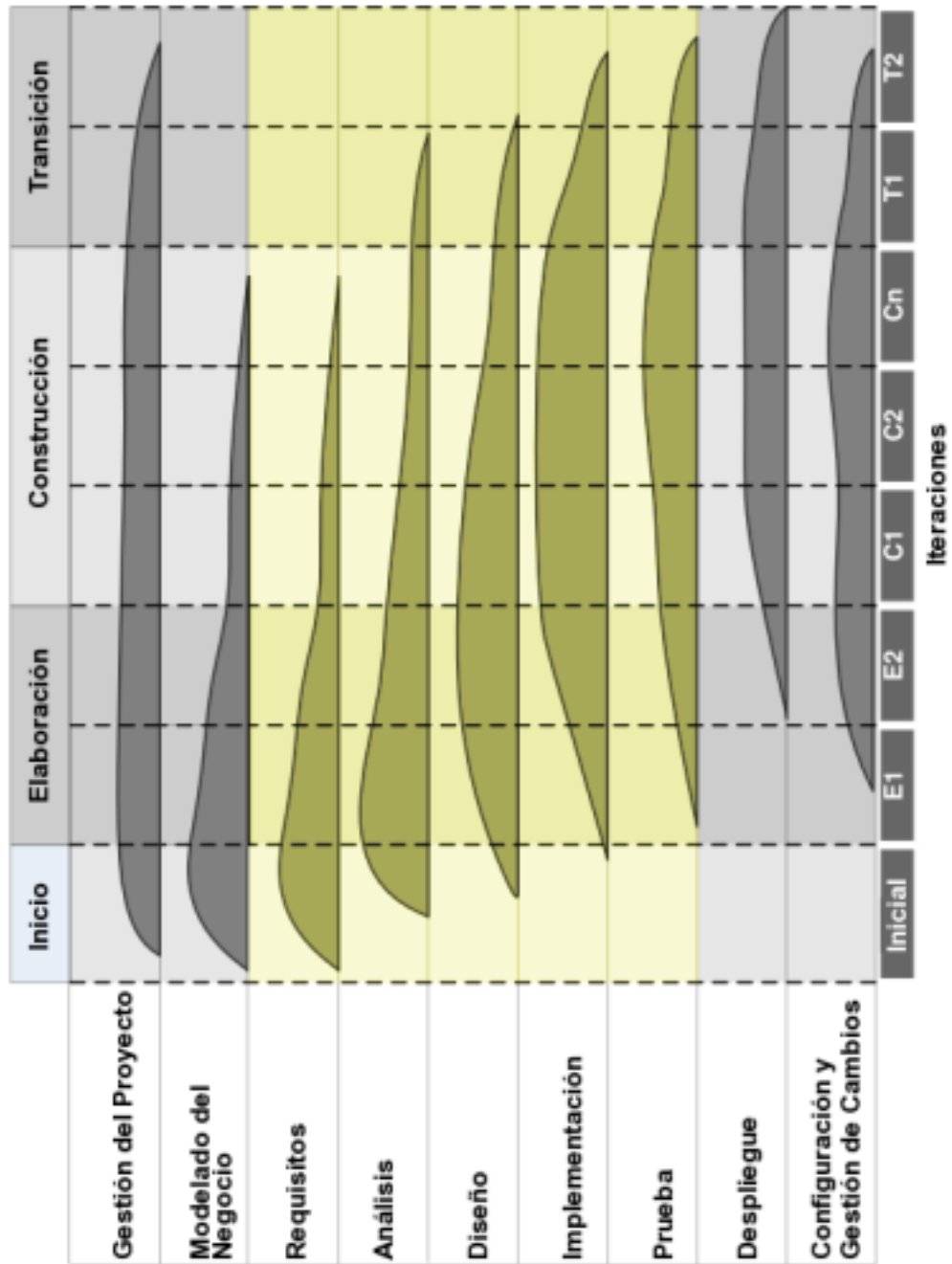


Figura 3.6: Ciclo de Desarrollo de un sistema según el Proceso Unificado

La iteración se convierte en el hito más importante para asegurar el crecimiento continuo y el aseguramiento de la calidad total dentro del sistema que se está desarrollando. Las iteraciones marcan totalmente el ciclo de desarrollo del SITEM que utiliza una aproximación iterativa propuesta por el Proceso Unificado.

3.6. UML: Lenguaje de Modelado Unificado

Independiente del modelo utilizado para la construcción y gestión del desarrollo del sistema se requiere que la comunicación entre los diferentes integrantes del grupo de desarrollo sea efectiva. Se hace indispensable que todo el equipo utilice y entienda un lenguaje consistente y unificado con el cual expresen claramente sus ideas y desde el cual puedan marcar claramente las directrices a seguir. El lenguaje de Modelado Unificado, UML por sus siglas en inglés, brinda las características tanto sintácticas como semánticas para lograr caracterizar lógicamente cualquier tipo de software permitiendo ser utilizado en cualquier etapa del diseño y es especialmente útil en aquellos desarrollos enfocados a objetos.

El Lenguaje de Modelado Unificado es definido por el *Object Management Group*⁹:

UML es un lenguaje visual para la especificación, construcción, y documentación de los artefactos de un sistema. Es un lenguaje de modelado de propósito general que puede ser usado con la mayoría de los métodos orientados a objetos y a componentes; que puede ser aplicado a todos los dominios de aplicación (p.e., salud, finanzas, telecomunicaciones, aeroespacial) y plataformas de implementación (p.e., J2EE, .NET).

En [Jacobson,2005] se recalca que UML es usado para entender, diseñar, buscar, configurar, mantener y controlar la información acerca de los sistemas.

Con UML se crean artefactos con información acerca de la estructura - o vista estática- y el comportamiento - o vista dinámica- de un sistema. Cada vista del sistema se modela como una colección de objetos que interactúan, es decir, que tienen interfaces y relaciones entre ellos perfectamente definidas. Fruto de tal relación entre objetos el sistema ofrece una funcionalidad o cumple un objetivo que es de interés.

⁹La página del OMG (www.omg.org) describe la organización como un consorcio de la industria de la informática, sin ánimo de lucro, con carácter internacional y de membresía abierta. Los diferentes grupos de trabajo del OMG desarrollan estándares en un rango amplio de tecnologías.

3.7. Portales de Información y Conocimiento

Antes de la explosión de servicios a través de la Internet, los portales basados en aplicaciones web estaban recomendados solo para organizaciones que por su complejidad (en tamaño ó geografía) necesitaran de un sistema tecnológico para que todo su personal pudiese tener acceso a la información en forma compartida y simultánea. Sin embargo, fundamentado en el crecimiento del uso de Internet¹⁰ surge la necesidad de la sociedad por mantener cierto orden en la corriente de bytes y grupos de usuarios con intereses de información comunes empiezan a conglomerarse alrededor de portales temáticos no organizacionales.

Un portal de información no es, en esencia, una fuente nueva de información; es una vista de la información existente que dispuesta en una forma ordenada se convierte en una herramienta de conocimiento extraordinariamente poderosa permitiendo poner al descubierto información valiosa que se enmascaraba entre otra no menos interesante – Data Mining. Dentro de las múltiples ventajas que ofrece el portal, es que proporciona la facilidad de obtener información actualizada a muy corto plazo, lo que es indispensable para la óptima toma de decisiones. Dicha información esta disponible, en condiciones óptimas, veinticuatro horas al día, trescientos sesenta y cinco días del año, permitiendo así acceder a los datos que se necesitan de acuerdo con la disponibilidad singular de tiempo y apoyar la toma de decisiones bajo cualquier circunstancia y lugar.

Gran cantidad de organizaciones y grupos de usuarios están explotando el uso de los portales creando y transformando servicios y procesos tradicionales convirtiéndolos en servidores de autoservicio, pudiendo así dedicarse a aquellos de mayor valor agregado a la organización, el personal o el grupo de investigación.

3.7.1. Beneficios y obstáculos para la implementación de portales basados en aplicaciones web.

Las facilidades que proporciona la tecnología, permite que el portal sea accedido a través de numerosas opciones, esto es a través de computadoras de escritorio y portátiles integradas a la red interna de la organización, a través de Internet por redes de banda ancha y estrecha y de los diversos medios inalámbricos como son las tecnologías celular, WiFi, WiMax, BlueTooth por intermedio de PDA, celulares y equipos de cómputo en redes WLAN.

Debido a la estructura del portal, se tiene una fuerte correlación entre diversas aplicaciones que nos permiten analizar interrelaciones que serían realmente complejas y tardadas si no se contara con ellos. Sin embargo, es importante recalcar más que los beneficios los problemas potenciales. De hecho en el éxito de un portal están enfocados factores clave que tienen beneficios y problemas asociados.

¹⁰El porcentaje de la población con acceso a Internet en Colombia a crecido de un 2,1 % en el año 2000 a un 15,8 % en el 2007, según la Comisión de Regulación de Telecomunicaciones.

Factor Humano Los individuos adaptan los procesos de información en diferentes maneras.

Factor Tecnológico Intranets, pueden ser costosas y poco efectivas si la organización no tiene la tecnología necesaria para construirlas.

La principal ventaja obtenida al construir y mantener un portal basado en aplicaciones web es mejorar la eficiencia y efectividad en la comunicación de los miembros de una organización o un grupo de usuarios, lo que aumenta la objetividad en la toma de decisiones y la transferencia de conocimiento. Todo lo anterior se maximiza si el portal se concibe como fruto de un proceso de investigación en donde todos sus componentes y servicios se construyen, mantienen, distribuyen e integran de acuerdo a los requerimientos de los usuarios finales[Sarmiento,2005].

Una de las características importantes de los portales es que en un sólo lugar - y con un mecanismo de acceso unificado, los usuarios pueden acceder a las aplicaciones. Esta integración con aplicaciones y servicios orientados al trabajo colaborativo hacen que trascienda los límites de un mero repositorio organizacional - que permite el autoservicio de requerimientos y extracción de información básica - y lo convierte en una herramienta de administración del conocimiento, útil para la toma de decisiones.

Las novedosas tecnologías que convergen en Internet permiten que la información sea personalizada y dirigida de tal forma que se potencian ciclos de creación, captura y diseminación de conocimiento necesarios para el crecimiento de los activos intangibles de los grupos y organizaciones. Así los portales convierten la información en valor, ya que eliminan las barreras de distancia y disponibilidad de información, reduciendo costos conectando a múltiples personas en diversos sitios al mismo tiempo.

La tabla 3.1 muestra los principales beneficios potenciales de desplegar los portales de información y conocimiento dentro del quehacer de las organizaciones, los grupos de trabajo y las comunidades de práctica.

Los riesgos que se afrontan también son enormes y pueden llevar al traste cualquier política o proyecto de desarrollo; la tabla 3.2 muestra algunos de los más importantes que deben ser minimizados.

3.7.2. Ciclo de Vida de los Portales

Los portales como representación sistemática del quehacer de un grupo humano evoluciona en la medida que dicho grupo mejora su conocimiento de las relaciones entre sus miembros y el entorno que los rodean. En general pueden determinarse cinco macro-etapas [UN, 2005] que aumentan gradualmente su funcionalidad basado en el conocimiento organizacional y la interrelación de usuarios a través del Portal:

Presencia Emergente Esta es la etapa primaria por la que pasa un portal en donde su

Beneficios Humanos (suaves)
Provee estructura de soporte 24 hrs. Servicio centrado en el usuario. Medio ambiente amigable.
Beneficios Físicos y capitales (beneficios fuertes)
Creación medioambiente libre de papel. Mejorar eficiencia y efectividad. Reducción de costos. Menores tiempos en consecución de información.
Beneficios estratégicos
Creación de herramientas innovadoras de apoyo. Proveer información a tiempo real. Apoyo a proceso de negocios de reingeniería Apoyo a los ciclos de creación, captura y diseminación de conocimiento. Aumento de Capital intangible. Formalización del <i>Know-How</i> .

Tabla 3.1: Algunos Beneficios Potenciales al Implementar un Portal

De tipo Humano (Fuertes)
Indiferencia de la administración. Sobrevaloración del papel de las TIC Dirección centrada en el capital financiero Estructuras organizacionales conservadoras. Micropoderes y feudos organizacionales autogestionados. Ignorancia y/o resistencia respecto al uso de TIC. Resistencia a estandarizar la información. Resistencia a compartir información/conocimiento.
Riesgos Físicos/Capital
Procesos orientados a la técnica y no multidimensionales. Carencia de capital(TIC marginales). Dificultad de integración de tecnología nueva y la existente. Ausencia de inter, trans y multidisciplinariedad.
Riesgos tecnológicos (Débiles)
Estándares propietarios. Redes de interconexión da baja velocidad. Alta relación Consumo/Adopción de tecnología.

Tabla 3.2: Algunos riesgos Potenciales al Implementar un Portal

funcionalidad es la de distribuir información interesante para un grupo de usuarios, el cual es totalmente caracterizado por el grupo de personas que construye - en todas sus dimensiones, el portal. Dichos usuarios no intervienen directamente en la estructura del Portal el cual complementa sus servicios por medio de enlaces y dependencia a otros portales temáticamente relacionados.

Presencia Mejorada Los usuarios pueden determinar en cierto grado la navegación a través de búsquedas en el archivo del sitio. Un portal en esta etapa presentará gran cantidad de información que usualmente se agrupa por áreas temáticas. Los mapas del portal se distribuyen profusamente con el fin de guiar a los usuarios en su tránsito por el mismo y usualmente un sistema básico de ayuda prediseñada esta disponible.

Interacción Los portales registran a sus usuarios. Se implementan herramientas en línea como las salas de charla - chat, las listas de correo y los foros; se realiza capacitación básica por medio de seminarios basados en contenidos y se hace uso extensivo de recursos multimediales. La ayuda es síncrona o asíncrona pero ágil lo que fomenta una depuración y actualización de la información contenida en el portal.

Transacción Los usuarios realizan operaciones a través del portal. El comercio electrónico, la gestión de contenidos, la personalización de los ambientes del portal, búsquedas semánticas y el despliegue de servicios avanzados - cursos, blogs, etc; caracterizan esta etapa.

Transformación La etapa más avanzada de los portales en donde se han estructurado comunidades de práctica sobre temas concretos que potencian los ciclos de conocimiento mediante las herramientas brindadas por el portal. Se observa una jerarquía *ad hoc* de usuarios con base en su aporte. Ellos mismos generan contenido que es convalidado por la comunidad y los administradores técnicos limitan sus funciones a aquellas relacionadas con mantener operativa la plataforma tecnológica. Las transacciones y el contacto en tiempo real son rutinarios. Las aplicaciones son de conocimiento general y el nivel de inmersión en el portal es alto.

3.7.3. Aplicaciones Web

También conocidas como **WebApps** son, en su concepción más básica, aplicaciones que responden a peticiones realizadas por un usuario por medio de un navegador (cliente) y ejecutan la lógica del programa en un servidor. Las aplicaciones web usualmente interactúan con sistemas de bases de datos y distribuyen los resultados de sus operaciones en lenguajes estándar tales como HTML, SMIL, XML, RDF, SVG, etc.[Jackson,2005]. Las WebApps también se pueden encontrar en ambientes diferentes al modelo cliente - servidor[Bos,2004].

Entre las características de las aplicaciones web se destacan[Bos,2004]:

- **No requieren instalación.** En general las aplicaciones web no necesitan ejecutar rutinas de instalación en las máquinas cliente. Quizás en algunos lenguajes sea necesario

la preparación de un ambiente específico de trabajo que en la mayoría de las veces es de acceso público.

- **Accesibilidad.** Las aplicaciones web se despliegan desde de una página web. Los protocolos usados son estandarizados y abstraen fácilmente las capas de aplicación de las de diseño y datos.
- **Facilidad en el Desarrollo.** Los lenguajes usados son de alto nivel, con un buen soporte para cadenas de caracteres, diferentes tipos de datos y con facilidades para la programación orientada a objetos. La mayoría de ellos con sintaxis similares y herramientas de desarrollo gratuitas de fácil adquisición.
- **Independencia de la Plataforma.** Las *WebApps engines* implementan el modelo de capa intermedia lo que permite que las diferentes WebApps puedan ser desplegadas sobre diferentes plataformas sin detrimento de su funcionalidad. El uso de métodos genéricos definidos en interfaces de programación (API) ayuda en gran manera a garantizar esta característica.
- **Seguridad.** No obstante la facilidad de acceso de la WebApp, estas pueden implementar rutinas avanzadas que brindan ambientes transaccionales seguros aislados del sistema de archivos y configuración del sistema en donde se alojan. El intercambio de información cifrada por la red y la integración con la seguridad de los servidores de bases de datos forman un contexto de alta seguridad.
- **Privacidad.** Las WebApps pueden operar fácilmente sobre una Plataforma de Preferencias de Privacidad debido a que la mayoría de los motores están habilitados para soportar el protocolo **P3P**.
- **Almacenamiento Persistente.** Tanto en el cliente - a través de archivos texto para el manejo de sesiones; como en el servidor de base de datos.
- **Integración.** Las aplicaciones Web pueden brindar sus servicios - u obtener uno determinado, a través de interfaces claras y definidas en las denominadas redes de servicios Web.

Capítulo 4

OpenSITEM - Sistema Federado de Aplicaciones para la Caracterización de Nodos Potenciales de Redes de e-salud

OpenSITEM es un sistema federado de aplicaciones de software libre o de código abierto que provee herramientas para analizar datos e información que son de interés para la descripción - y definición de capacidad de, nodos potenciales de redes de e-salud tales como: entidades de salud, servicios médicos, tecnologías de interconexión, operadores de telecomunicaciones, equipos médicos, organizaciones, profesionales, estándares, pacientes, enfermedades, medicamentos y proyectos. Implementa un ambiente para apoyar las tareas de las comunidades de práctica involucradas en la investigación, el diseño, mantenimiento, desarrollo e implementación de redes de eSalud. Tuvo su génesis en la primera fase del Proyecto Telemedicina Bogotá, como solución a la necesidad de administrar los resultados del estudio de campo realizado a las entidades e instituciones de salud y los operadores de Telecomunicaciones en la ciudad de Bogotá.

Esta plataforma va generando de manera progresiva un sistema que permita la definición, categorización y caracterización de nodos potenciales de redes de eSalud, para apoyar las actividades básicas de los *trabajadores del conocimiento* en el área de la telemedicina del grupo GITEM¹, ofreciéndoles, además de un repositorio de datos, herramientas que facilitan las tareas de capturar, extraer, organizar, analizar, encontrar, sintetizar, distribuir y compartir información y conocimiento de nodos potenciales. OpenSITEM describe un mecanismo para federar herramientas que sirvan para actualizar el estado de los nodos definidos en el modelo base, así como para la caracterización de nodos y categorías inéditas. El modelo base de nodos

¹Conformado por profesionales y estudiantes de la Universidad Distrital así como por profesionales de las diferentes instituciones que han participado en los diferentes estudios de campo.

Tabla 4.1: Módulos base de OpenSITEM

Módulo	Elaboración	Nombre
Motor de Federación de Aplicaciones	Propia	OpenSITEM-FE
Motor de recomendación	Propia	OpenSITEM-RS
Gestión de Nodos	Propia	OpenSITEM-NM
Diseñador de Redes	Propia	OpenSITEM-ND
Analizador de Redes	Propia	OpenSITEM-NA
Gestión de Encuestas	Propia	OpenSITEM-PM
Inteligencia de Negocio	Tercero	Knowage Cesium
Sistema de Información Geográfica	Tercero/Propia	qGIS Leaflet
Gestión Documental	Tercero	Alfresco
Gestión de Reportes	Tercero/Propio	Reportico Varias bibliotecas

y categorías se construye a partir del estudio de campo realizado por el grupo de investigación y se plantea una plataforma para gestionar los datos de nodos potenciales de redes de eSalud, contribuyendo a disminuir el tiempo de adquisición, análisis y despliegue de la información.

La integración del sistema² está basado en métodos conocidos [Balduino,2010]. Para el diseño de los módulos inéditos se ha tenido en consideración varios principios, patrones y antipatrones, tratando de minimizar riesgos que impacten de manera negativa a la calidad del software.

OpenSITEM hace uso extensivo de aplicaciones existentes, marcos de trabajo, bibliotecas, APIs, servicios web y plantillas. Esto ha permitido lograr un alto grado de funcionalidad específica. Los módulos propios³ - aquellos que hacen parte de la suite desarrollada por GITEM - se implementan sobre el framework OpenSARA⁴.

4.1. Módulos Funcionales

OpenSITEM- incluyendo las aplicaciones de terceros y las desarrolladas por GITEM, ofrece los módulos base listados en la tabla 4.1:

²OpenSITEM es un sistema federado de aplicaciones.

³En referencia a la autoría, no al carácter de código abierto.

⁴Diseñado y construido por GITEM. OpenSARA es un producto de este proyecto y se ha usado en otros dominios tanto en la Universidad Distrital (sistema de gestión de inventarios, sistema de consultas a comunidades, sistema de evaluación para acreditación, entre otros), como por algunas empresas del sector TI (OpenKyOS).



Figura 4.1: Imagen en el año 2017 de la página principal de OpenSITEM

4.2. Características de Interés de OpenSITEM

OpenSITEM define algunas características que son de interés en diferentes dominios:

- **Dominio de Aplicación:** Desarrolla un motor y un proceso para la federación de aplicaciones.
- **Dominio de arquitectura:** Propone un modelo de arquitectura para la caracterización de nodos potenciales de redes de eSalud.
- **Dominio de utilidad:** Implementa una plataforma para soportar flujos de trabajo de investigadores del GITEM. En el mercado no existe una herramienta que de manera unificada cumpliera con el modelo de requerimientos definido.
- **Dominio social:** Tanto el framework (OpenSARA) como los módulos de OpenSITEM están disponibles en repositorios públicos y cobijados por licencia de código abierto. El framework ya se ha utilizado por equipos de trabajo externos al grupo GITEM⁵. Se tiene presupuestado que la fase IV del proyecto (transición) permita que los datos también sean abiertos.

4.3. Descripción de la Arquitectura de OpenSITEM

Como se ha mencionado OpenSITEM es una aplicación federada de arquitectura orientada a servicios, con características como las descritas en [erl, 2017]. La integración es realizada por un motor que permite el intercambio de mensajes y la sincronización de sesiones entre los diferentes aplicativos federados sin llegar a considerar una arquitectura de Bus de Servicio Empresarial (ESB por sus siglas en inglés)⁶

En esta sección se describe la arquitectura general de OpenSITEM y de los módulos inéditos realizados en el marco del proyecto. Las aplicaciones federadas construidas por terceros se abstraen, visibilizando únicamente las interfaces provenientes de las API o de los servicios web utilizados en su adaptación.

La descripción de la arquitectura sistema (AD) se realiza con base al modelo conceptual definido en [ISO,2011]. Dada la extensión de los artefactos, se presentan aquellos apartes de la AD que a juicio de los autores se consideran relevantes para explicar las propiedades fundamentales.

⁵Especialmente en la Universidad Distrital en el sistema de gestión de votaciones, sistema de autoevaluación institucional, sistema de gestión de inventarios, sistema de pago en línea, sistema de costos, entre otro.

⁶Aunque existen ESB de corte empresarial tales como Mule, WSO2, Apache Service Mix o similares, el análisis de utilidad mostró que la mayoría de funcionalidades que ofrecen nunca serían utilizadas y entonces se optó por la simplicidad arquitectónica y evitar la tarea recurrente de configuración y administración del ESB. Ver el anexo I.

4.3.1. Interesados e Intereses

El proyecto considera una aplicación que en su conjunto es inédita y de la cual no se encontró referente de modelo de dominio. En este escenario es importante la identificación de los interesados, los cuales están enfocados en los investigadores del grupo GITEM que participan en proyectos relacionados con las redes de eSalud. También incluye expertos en el diseño de redes de telecomunicaciones y de prestación de servicios médicos.

A continuación se muestra el resultado del proceso de identificación:

- Asistente de Investigación Hospital
 - Ingresar y corregir la información recopilada en el estudio de campo.
 - Elaborar informes a partir de la información ingresada utilizando diferentes filtros y formatos.
 - Generar una plantilla del informe de investigación que cumpla con los parámetros exigidos por la Universidad.
 - Registrar las evidencias de las visitas de campo y demás entregables que exige el proyecto.
 - Gestionar el proyecto de caracterización de los hospitales.
 - Visualizar geográficamente algunos atributos del hospital
- Analista en Proyectos eSalud del GITEM
 - Analizar la información de diferentes nodos
 - Evaluar la potencialidad de los nodos para pertenecer a una red de eSalud.
- Diseñador de redes de eSalud
 - Diseñar redes de eSalud a partir de la interconexión de nodos.
 - Evaluar la potencialidad de las redes diseñadas.
- Experto en nodo
 - Ingresar y corregir información relacionada con el nodo.
 - Elaborar informes a partir de la información ingresada utilizando diferentes filtros y formatos.

Los principales mecanismos utilizados para la identificación fueron:

- Entrevistas Directas: Realizadas en su mayoría durante la primera iteración de la fase de inicio. Se definió una entrevista base, anexo K.

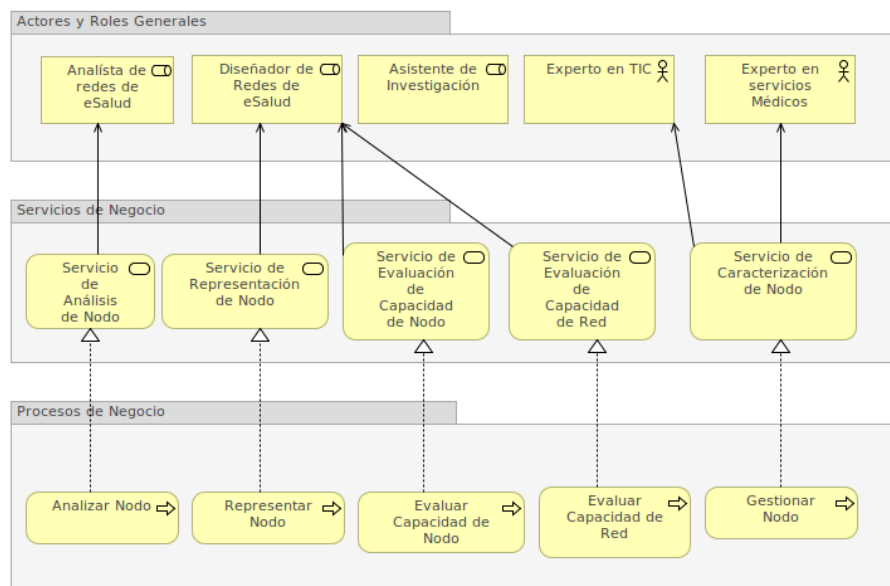


Figura 4.2: Punto de Vista de Negocio - Vista Conceptual por Capas

- Reunión Presencial: De duración fija (30 minutos) y con agenda previa ⁷.
- Matriz de identificación: En donde se consignaron las evaluaciones de los atributos de liderazgo, poder de decisión, nivel de interés en el proyecto, conocimiento, actitud ante el proyecto (oposición, neutralidad, soporte), procedencia (externa o interna - GITEM, Universidad).

4.3.2. Punto de Vista Estructural

La estructura de OpenSITEM se presenta principalmente desde tres puntos de vista:

- Punto de Vista de Negocio (figura 4.2): Con los grupos de actores, roles generales, servicios de negocio y procesos de negocio.
- Punto de Vista de Aplicación (figura 4.4): Con los grupos de Servicios de aplicación y Módulos de Aplicación (federadas y nativas⁸).
- Punto de Vista Infraestructura (figura 4.7): Con los grupos de Servicio de Infraestructura y Componente de Infraestructura.

⁷La agenda en las reuniones de identificación fue simple. Se buscaba tener una idea de quienes eran los interesados representativos de los diferentes roles que existen en el grupo.

⁸Una aplicación federada es aquella que mayormente ha sido desarrollada por equipos externos, mientras que una aplicación nativa es la que mayormente ha sido desarrollada por integrantes de GITEM.

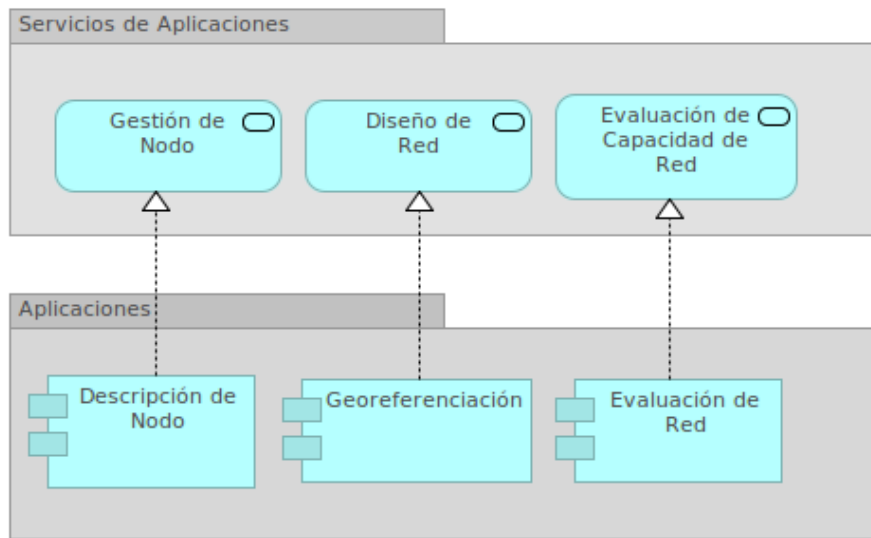


Figura 4.3: Punto de Vista de Aplicación - Vista Conceptual por Capas.

Las aplicaciones nativas se denotan por las iniciales de cada módulo. Por ejemplo el motor de federación (Federation Engine) se conoce como OpenSITEM-FE, el gestor de nodos (Node Management) como OpenSITEM-NM, etc.

4.3.3. Punto de Vista de Comportamiento

OpenSITEM-FE: Motor de Federación de Aplicaciones

OpenSITEM no es un sistema de federación universal de aplicaciones. Integra aplicaciones que expongan su funcionalidad a través de interfaz de programación de aplicaciones (API) o de servicios web, utilizando protocolos de la pila TCP/IP con representación basada en XML o JSON. Por esta razón - y por las mostradas en el anexo I; se prescindió de emplear una solución ESB y se diseñó un motor de integración con la siguiente funcionalidad:

- **Evaluación de Federación:** Servicio que se encarga de recibir el mensaje proveniente de una aplicación - a través de un servicio web general, y definir si dicho mensaje hace parte de una transacción que es necesario ser federada. En el contexto de OpenSITEM, se dice que una transacción es federada cuando luego de su ejecución debe “disparar” transacciones en uno o varios sistemas diferentes.
- **Acceso a Aplicación:** Servicio por el cual un usuario adquiere las credenciales para acceder o consumir un servicio específico de una aplicación federada.
- **Registro de transacción:** Servicio específico por aplicación federada que se encarga de consumir un servicio web. Su objetivo es realizar una transacción en una aplicación.

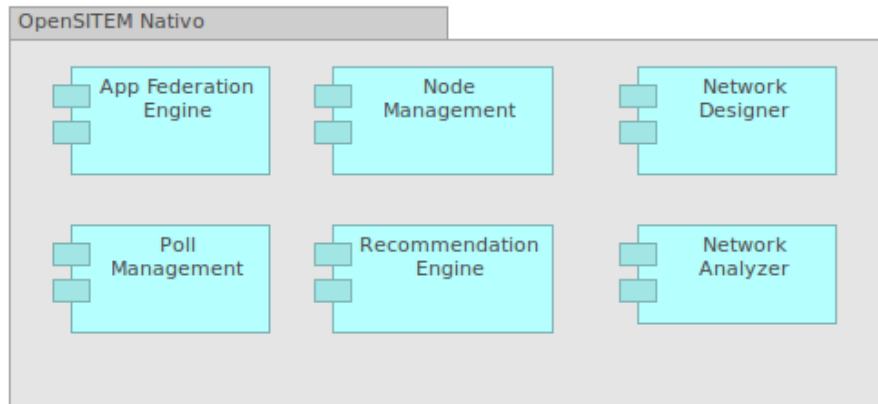


Figura 4.4: Punto de Vista de Aplicación - Aplicaciones nativas de OpenSITEM.

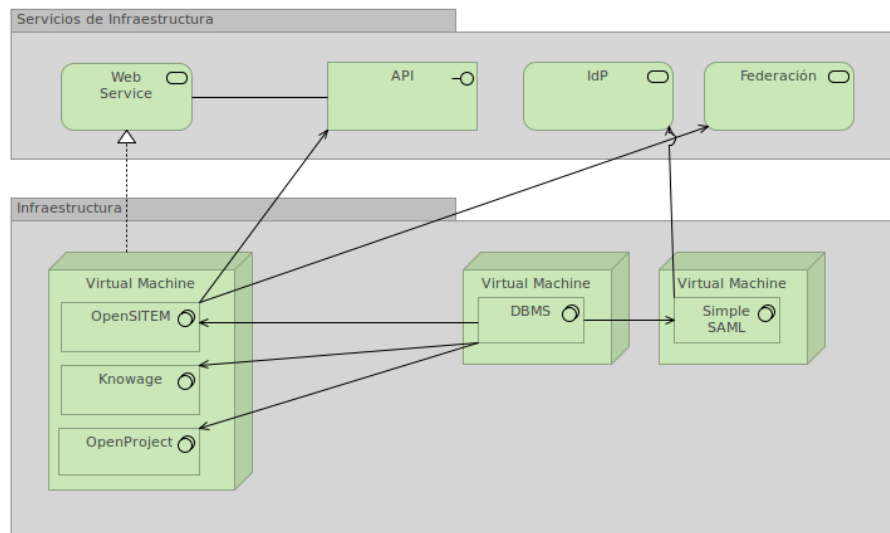


Figura 4.5: Punto de Vista de Infraestructura - Vista Conceptual por Capas.

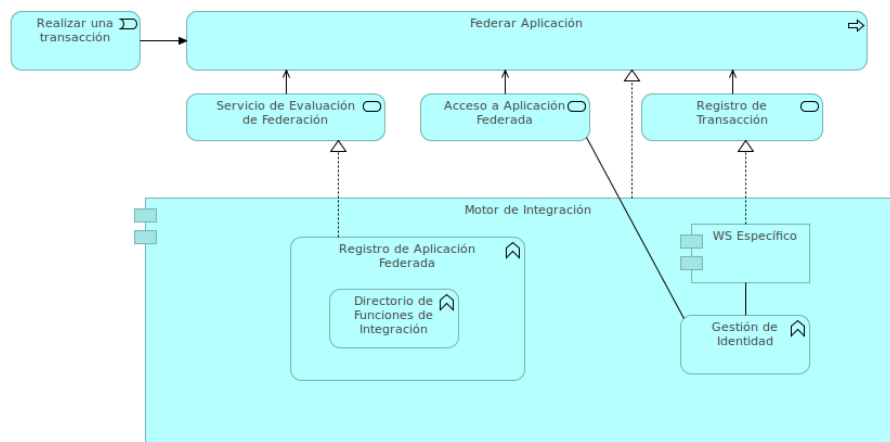


Figura 4.6: Punto de Vista de Comportamiento - Modelo General del Motor de Federación de Aplicaciones.

4.3.4. Punto de Vista de Distribución

OpenSITEM y cada una de las aplicaciones federadas están desplegadas en una infraestructura virtualizada. Aunque al momento de elaborar este informe se está trabajando en migrar a contenedores, la tecnología que se emplea esta 100 % relacionada con máquinas virtuales.

En un plano técnico, cada una de las instancias utiliza un sistema operativo GNU/Linux con distribución Centos.

4.4. Gestión de nodos

El módulo de gestión de nodos se puede considerar como un directorio enriquecido de los objetos con potencialidad de pertenecer a una red de eSalud. Este directorio se considera el nivel ontológico de mayor abstracción y constituye lo que denominamos el *Catálogo de Nodos*. A nivel arquitectónico se trata de una clásica aplicación para crear, leer⁹, actualizar y borrar¹⁰. Los formularios y reportes utilizan plantillas XML, lo que permite hacer cambios sin tener que modificar directamente el código fuente.

⁹Gestionado por el motor de reportes, el cual es una biblioteca externa basada en la aplicación Reportico

¹⁰En realidad en OpenSITEM nunca se borra ni actualiza nada, solamente se replican los registros y se marca su estado. Esto permite tener datos que han demostrado ser útiles en tareas de depuración, control y minería de datos.

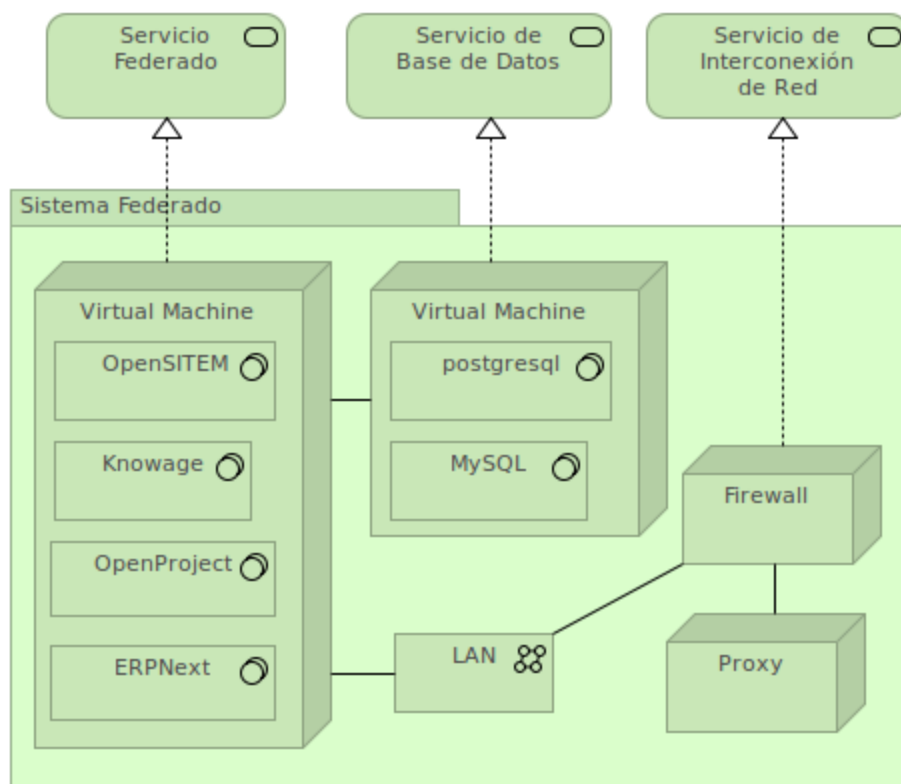


Figura 4.7: Punto de Vista de Distribución.

4.5. Motor de Recomendación

El motor de recomendación de OpenSITEM utiliza dos aproximaciones para determinar cual es el nodo *óptimo* en una determinada búsqueda - cual se muestra primero, o en una determinada estructura de red - cual *encaja*. El motor es un módulo de análisis de contenido con un algoritmo que evalúa la similitud que tienen los atributos del nodo respecto al filtro empleado, y un segundo algoritmo que revisa *lo apropiado* que han sido las clasificaciones anteriores a partir del registro de selecciones, si se trata de búsquedas, o de pertenencia en el caso de estructuras de red.

Ambos algoritmos están dotados de estrategias de variación de entropía con el objetivo de incrementar las probabilidades de aprendizaje.

4.6. Modelo de Nodos

Como se presentó anteriormente, el motor de gestión de nodos permite la definición de cualquier tipo de nodo. No obstante, el equipo de trabajo ha definido un modelo base de nodos que se consideran el conjunto reducido que permite describir los elementos primordiales¹¹ de las redes de eSalud.

Los tipos de nodos del modelo base son:

4.6.1. Prestador de servicio de salud

Esta jerarquía de clases, figura 4.8, se caracterizó a partir de los datos recopilados en el estudio de campo realizado por el grupo GITEM, en el proyecto del Sistema de Gestión de Salud para el Distrito Capital fases I y II. Sus atributos están compuestos por objetos de varias clases: especialidad médica, red de comunicaciones, red eléctrica y red de atención.

La información de estos nodos puede ser administrada por las entidades prestadoras de servicios de salud, de tal forma que se cree gradualmente un catálogo del estado actual de las entidades y su potencialidad para ser parte en redes de eSalud. Las entidades iniciales pertenecen a la red de la secretaría de Salud de Bogotá pero por medio del módulo denominado *redes de atención* se puede crear cualquier prototipo de red jerárquica de atención [Yellowlees, 2005].

¹¹En este caso el carácter de *primordial* fue definido por el grupo de interesados del GITEM.



Figura 4.8: Arquitectura Básica de Prestador de Servicio de Salud

4.6.2. Tecnología de Interconexión

Nodos con datos de las tecnologías y protocolos de interconexión disponibles en las redes de acceso y transporte. Estas tecnologías se ordenan principalmente sobre el modelo de referencia OSI, pudiéndose crear - desde el módulo de *Arquitectura*, cualquier otro tipo de modelo. En la actualidad se tiene como alternativa de clasificación el modelo TCP/IP. El conjunto de nodos de esta clase será una guía técnica que muestra la información de las capas físicas, de enlace y de red en formatos unificados¹².

Los nodos de tipo Tecnología de Interconexión proveen información a los analistas para la revisión sistemática de las opciones que brindan los fabricantes y así proyectar redes que sean técnicamente y económicamente viables¹³. La información se estructura de acuerdo a indicadores cualitativos y cuantitativos que permiten evidenciar el carácter de interoperabilidad, impacto y permanencia de la tecnología en el mercado.

4.6.3. Equipo

Nodos con datos técnicos de equipos con potencial uso en la telesalud - puede ser desde un equipo de resonancia magnética hasta un tablero interactivo. Posee atributos que son objetos de las clases Proveedor y Fabricante, así como la atributos de especificaciones técnicas, funcionales y física.



Figura 4.9: Arquitectura Básica de Operador de Telecomunicaciones

4.6.4. Operador de Telecomunicaciones

Describe nodos de operadores de telecomunicaciones, figura 4.9, con énfasis en las características técnicas de los servicios que ofrecen, su cobertura y tarifas. Estos nodos brindan a los analistas información de operadores que permite determinar comparativas y evaluar las ventajas y desventajas entre diferentes opciones de interconexión. Este módulo se complementa con la información de dominio público que muestra el *Sistema de Información Unificado para el sector de las Telecomunicaciones* mantenido por la Comisión Reguladora de Telecomunicaciones.

4.6.5. Organización

Describe las características de organizaciones y grupos de investigación que trabajen en el área de la eSalud. Estos nodos son claves para vislumbrar potencialidades de trabajo en grupo y de sistematizar experiencias adquiridas en los diferentes proyectos desarrollados en el área.

4.6.6. Servicios Médico

Este tipo de nodo cuya arquitectura se muestra en la figura 4.10, describe servicios médicos. Sus instancias componen un catálogo de servicios y especialidades médicas disponibles. Se pone especial atención en la descripción detallada de los requerimientos técnicos y tecnológicos que requiere cada especialidad así como el perfil de los profesionales y entidades educativas de formación de especialistas.

¹²Aunque existen portales con esta información, por ejemplo tcpipguide, se requiere datos estructurados que faciliten la búsqueda y comparación de tecnologías.

¹³En la construcción del modelo de requisitos se encontraron varios casos en los cuales un equipo no podía accederse de manera remota porque sus interfaces eran incompatibles con las tecnologías de las instituciones.



Figura 4.10: Arquitectura de Servicio Médico

4.7. Aplicaciones Federadas

OpenSITEM integra a su arquitectura soluciones de software libre o de código abierto, para proveer un ambiente integrado y aumentar sus prestaciones - o implementar nuevos casos de uso. Entre las aplicaciones, mostradas en la figura 4.13, que contribuyen a openSITEM se pueden citar:

Knowage Suite de inteligencia analítica y de negocios[Engineering, 2017].

OpenProject Sistema de gestión de proyectos[OpenProject, 2017] .

Alfresco Sistema de Gestión Documental.

Cesium Visor geográfico 3D.

WordPress Sistema de Gestión de Contenido.

SimpleSAML Proveedor de identidad.

Moodle Es un ambiente integrado de aplicaciones para la creación, organización, mantenimiento y seguimiento de cursos en línea. El fin primordial de sus herramientas es dar soporte a un marco de educación social constructivista.

MapServer Es una aplicación desarrollada en Python que proporciona al openSITEM los servicios básicos necesarios para la georeferenciación de la información.

Sistema de Información Unificado del Sector de Telecomunicaciones Conocido como SIUST, es un aplicativo Web desarrollado por la Comisión Reguladora de Telecomunicaciones que contiene información del sector de las telecomunicaciones en Colombia:



Figura 4.11: Aplicaciones de Software Libre o Uso libre que complementan al SITEM

“...información técnica de infraestructura, normatividad del sector, estadísticas comerciales e índices financieros de los prestadores de servicios y los indicadores de gestión del sector entre otros.” ¹⁴

La información, que es de acceso público, permite que openSITEM se nutra de ella para complementar y validar sus propias bases de datos en algunos subsistemas.

Wikipedia Quizás la fuente de información colaborativa más grande en Internet, debido a que sus contenidos son de uso libre, el SITEM se nutre de ellos y a su vez los complementa. A partir de información disponible se han editado más de 50 artículos en Wikipedia que tienen relación temática con el SITEM.

¹⁴Tomado del sitio web del SIUST. <http://www.siuist.gov.co/siuist/>

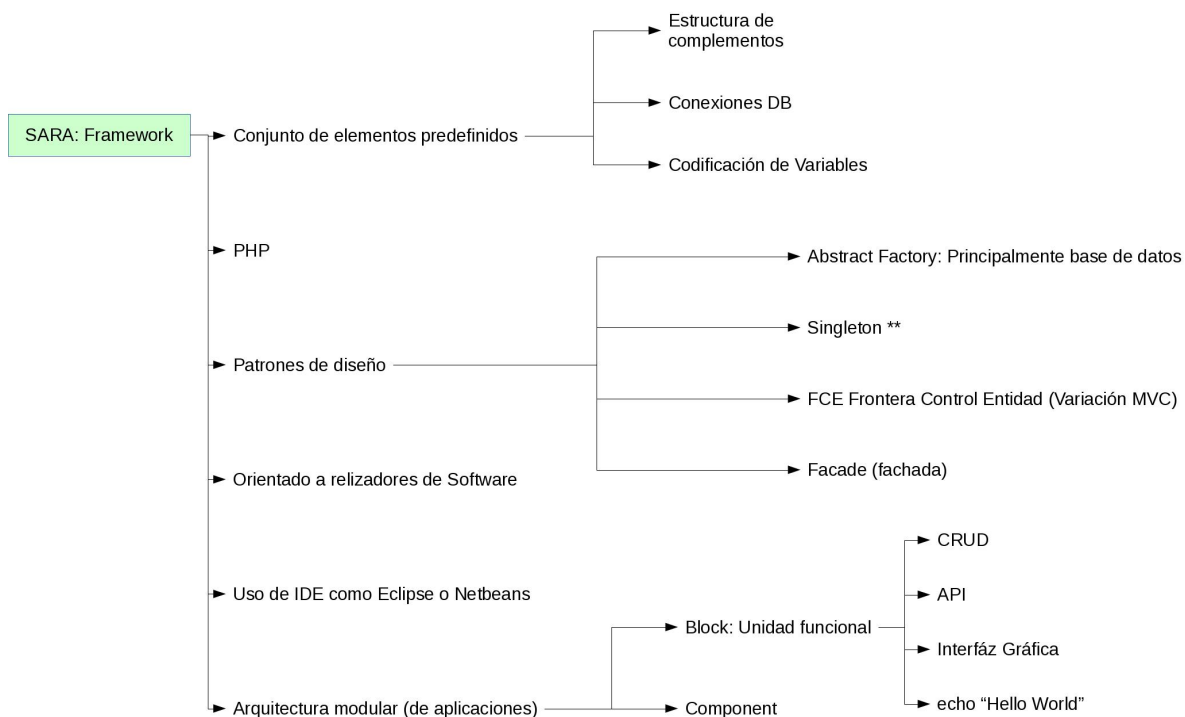


Figura 4.12: Conceptos principales de OpenSARA

4.8. Marco de Desarrollo OpenSARA

openSARA es un marco de desarrollo experimental para aplicaciones PHP orientadas a la web. Se desarrolló con el objetivo de simplificar el desarrollo de openSITEM con una orientación hacia programadores. En ese sentido, está pensado para desarrollo ágil de módulos inéditos y el reciclaje de módulos preexistentes. El código base se distribuye bajo licencia Eclipse Public License o GPL. Los diferentes plugins se distribuyen de conformidad a la licencia definida por sus autores.

openSARA es un marco de trabajo basado en el patrón Frontera - Control - Entidad. La conexión a bases de datos se abstrae por medio de un componente que implementa el patrón *Abstract Factory*.

4.8.1. Patrón Frontera -Control - Entidad

Es una variación del patrón Modelo - Vista - Controlador, en donde se hace un mayor énfasis en el bajo acoplamiento.

- Frontera: Objetos que actúan como interfaces del aplicativo con sistemas cliente ta-

les como: usuario humanos, servicios web, aplicaciones heredadas (legacy applications), Interfaces de Programación de Aplicaciones, etc.

- **Control:** Objetos de interoperación entre los objetos frontera y los objetos entidad. Su misión es orquestar la ejecución de la aplicación y la interacción con los sistemas cliente.
- **Entidades:** Objetos que implementan la lógica de negocio. Incluyen aquellos objetos que representan el modelo de datos de las aplicaciones.

Los objetos Frontera, Control y Entidad tienen símbolos específicos en el estándar UML.

4.8.2. Marco de Trabajo

Bloque

Elemento modular base de openSARA. Puede implementar todo un caso de uso - utilizando patrón FCE, o estar constituido por un simple archivo HTML o PHP. Depende del programador decidir las características del bloque conforme a los requisitos del proyecto.

OpenSARA recomienda tres modelos de bloques: CRUD completo síncrono, CRUD completo asíncrono (vía AJAX) y servicio web.

Página

Punto de interacción con el usuario. Son generadas en tiempo de ejecución con base en la estructura de bloques declarada en base de datos o archivos XML. Una página es creada a partir del código fuente de un conjunto de bloques. Es la interfaz gráfica de la aplicación.

Clase Nuclear

Conjunto de clases que gestionan la integración de aplicaciones, se dividen en siete categorías que se encargan de:

- **Builder:** construir las páginas enlazando los diferentes bloques que las constituyen.
- **Connection:** conectar a bases de datos.
- **Crypto:** codificar y decodificar cadenas de petición.
- **Auth:** autenticación, autorización y control de sesiones.

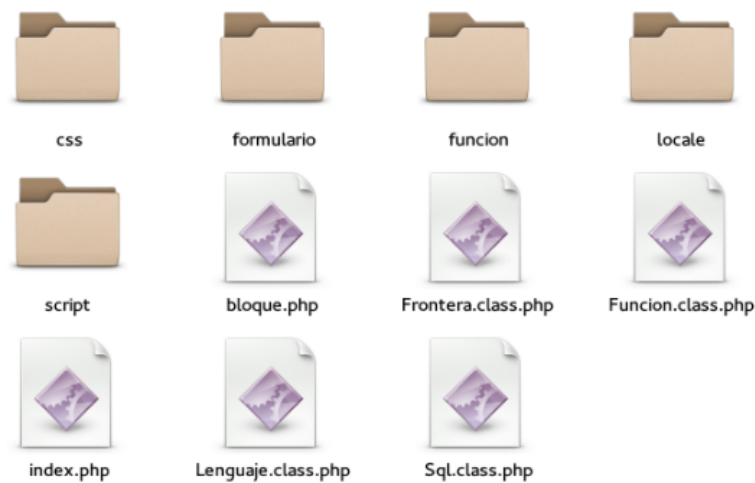


Figura 4.13: Estructura de un bloque en el sistema de archivos

- Manager: gestionar el marco de trabajo.
- Locale: Internacionalización
- General: implementar comportamiento genérico requerido por otras clases nucleares.

Componentes

Bloque especializado. Un componente se construye con el objetivo de que sea consumido por los bloques cuando se requiere un comportamiento genérico (enviar correos electrónicos, generar notificaciones, gestionar logs, etc). Los componentes definen un contrato a través de las interfaces que declaran. Complementos

4.8.3. Estructura

Bloque

Se ubican dentro de la carpeta blocks y se registran en la base de datos o en archivos XML dependiendo de la configuración del marco de trabajo. Un bloque es una carpeta que debe tener, como mínimo, un archivo llamado `bloque.php`, el cual se utiliza como punto de entrada (entry point). Aunque esta es la única restricción que impone openSARA para la estructura, el marco de desarrollo provee dos modelos de bloques enriquecidos que implementan el patrón FCE, el primero de ellos, llamado `bloqueModelo1`, está compuesto por:

- `bloque.php`: Clase `Bloque` que crea los objetos que procesarán las peticiones. Es el ele-

mento de control en el patrón FCE.

- `Frontera.class.php`: Clase Frontera, define la gestión de la interfaz de usuario a partir de los formularios especificados dentro de la carpeta formulario. Implementa el elemento Frontera en el patrón FCE.
- `Funcion.class.php`: Clase Función, define el procesamiento de los formularios basado en la gestión de objetos creados a partir de las clases de la carpeta funcion. Implementa el elemento Entidad en un patrón FCE.
- `Lenguaje.class.php`: Clase Lenguaje, define las funciones de internacionalización específicas para el bloque, describiendo el procesamiento de los archivos de la carpeta locale.
- `Sql.class.php`: Clase SQL. Define las cadenas SQL que serán utilizadas en el bloque.
- Carpeta script: Contiene los archivos js que se utilizarán en el bloque.
- Carpeta css: Archivos CSS que definen los estilos que serán aplicados a los elementos del bloque

Capítulo 5

Experiencia de Desarrollo de OpenSITEM

Como se ha mencionado, openSITEM es centrado en la arquitectura y guiado por casos de uso. Tanto el Director de Proyecto como los equipos de trabajo temporales, están comprometidos con la tarea de generar un aplicativo funcional que esté alineado - de manera emergente, con las vistas arquitectónicas planteadas.

Con el método definido, los artefactos se construyen de manera iterativa e incremental. El “orden” en el que aquí se muestra no implica una secuencia de actividades pues no en pocas ocasiones, abordar la disciplina de *Modelado de Dominio* se realiza a continuación de - o en paralelo a, un taller de requisitos y dicho taller surge de una nueva restricción detectada al momento de despliegue. Esta capacidad de adaptación al cambio es lo que ha permitido que openSITEM vaya realizando la transición desde un aplicativo específico a una suite escalable - potenciada por la arquitectura de federación de aplicaciones, orientada a servicios que está implementando.

En la implementación de cada uno de los módulos de openSITEM se siguen las fases contempladas en el Proceso Unificado¹ y se desarrollan flujos de trabajo en las disciplinas básicas de:

- Requisitos.
- Arquitectura.
- Elaboración.
- Pruebas.
- Despliegue.

¹Inicio, elaboración, construcción y transición.

A partir de iteraciones continuas por las diferentes disciplinas se refinan constantemente los modelos y se presentan resultados que permiten medir los avances, así como comprobar los niveles de calidad y la validez tanto de los entregables como del método empleado.

5.1. Modelo de Requisitos

Un requisito es una declaración explícita y aceptada de la funcionalidad que un usuario espera del sistema. Dicha declaración debe ser coherente con el objeto de estudio de openSITEM, respetando siempre los principios normativos y de propiedad intelectual. Cada requisito se plasma en un documento denominado Caso de Uso, el cual describe un conjunto de escenarios de interacción del usuario con el sistema, con el propósito de obtener algo de valor.

5.1.1. Declaración del Problema

OpenSITEM es una propuesta de solución a los siguientes problemas:

- La información de los estudios de campo no está estructurada, está dispersa, se ha perdido y es difícil de consultar y consolidar.
- El equipo de diseño de GITEM no cuenta con un mecanismo estándar para caracterizar los componentes de las redes de eSalud.
- No es posible determinar el valor de un nodo dentro de una red de eSalud.
- No existe capacidad de análisis multitemporal para determinar la validez de la categorización de nodos de una red de eSalud.

5.1.2. Responsabilidades del Sistema

El sistema debe contribuir a que los investigadores:

- Caractericen nodos con potencialidad de pertenecer a redes de eSalud.
- Creen instancias de nodos a partir de la información de los estudios de campo que desarrollo el grupo.
- Comparen nodos de la misma categoría para definir cual tiene mayor valor relativo.
- Consulten catálogos especializados en cada categoría de nodos.
- Definan modelos de categorías de nodos.
- Evalúen la potencialidad de un nodo para pertenecer a una red de eSalud.

5.2. Alcance

OpenSITEM no implementa todo el modelo del requisitos. Los módulos desarrollados son prototipos funcionales que sirven de guía para el desarrollo y evolución posterior de la solución. Si relacionamos esto con el proceso OpenUP, el grupo en la fase III alcanza el final de la fase elaboración.

5.3. Definición de actores

El sistema es manejado por varios tipos de usuarios, cada uno con características específicas. Este tratamiento especial tiene que ver con el mantenimiento de la integridad de la información, la cual solo puede ser gestionada por un grupo selecto de usuarios.

Algunos actores esperados en el openSITEM son:

- Administrador.
- Consultor.
- Especialista Médico.
- Profesional TIC.
- Usuario General.

Cada uno de ellos con las características mostradas en el anexo A.

5.4. Casos de uso

Los requisitos funcionales son declarados con diferente nivel de detalle empleando casos de uso, diagramas de comportamiento y de interacción. Al caracterizar una interacción por medio de un caso de uso se espera tener información acerca de:

- Nombre del Caso de Uso
- Objetivo que se logra al ejecutarse el caso de uso.
- Código que lo identifique unívocamente dentro del banco de artefactos.
- Actores que intervienen al desarrollarse el caso de uso.

Caso de Uso	
Nombre	Evaluar Capacidad de un Nodo
Objetivo	El actor asigna un valor y un diagnóstico a un nodo que da cuenta de la capacidad que tiene dicho nodo de pertenecer a una red de eSalud .
Código Interno	UC-GENERAL-XXXX
Actores	Consultor
Precondiciones	El nodo se encuentra registrado en el sistema en estado ACTIVO.
Flujo Básico	<ol style="list-style-type: none"> 1. El Consultor selecciona el nodo que desea evaluar. 2. OpenSITEM presenta los datos relacionados con el nodo 3. El Consultor revisa el historial de valoraciones del nodo. 4. El Consultor revisa el valor de los atributos del nodo. 5. El Consultor valora el nodo. 6. El Consultor argumenta la valoración a través de un diagnóstico. 7. OpenSITEM guarda los datos de la valoración. 8. OpenSITEM envía una notificación de la valoración a los interesados.
Postcondiciones	Se agregó una valoración argumentada al sistema.
Casos de uso relacionados	Consultar Historial de Valoración de Nodo

Tabla 5.1: Caso de Uso Evaluar Capacidad de Nodo

- Casos de uso con los que está relacionado.
- Precondiciones. El estado del sistema que debe asegurarse antes de que el caso de uso inicie. Debido a que es responsabilidad del sistema no se verifica en el caso de uso.
- Postcondiciones. Las características y estado del sistema una vez se haya terminado el caso de uso.
- Flujo de Tareas. Flujo principal y flujos alternativos de las actividades que se suceden para lograr la funcionalidad deseada. Se debe mantener claridad en el modelo por lo que se recomienda utilizar diferentes artefactos para los flujos alternativos cuando esto lo amerite.

La tabla 5.1, muestra el flujo principal del un caso de uso de openSITEM.

El modelo actual tiene 80 casos de uso principales definidos² y más de 230 flujos alternativos - los más relevantes incluidos en el anexo A. Con esto se concreta los requerimientos de más alto nivel definido por el grupo GITEM. Se recalca que el grueso de ellos aún no se han desarrollado por lo que se probablemente el modelo se adapte a medida que se progresa en la construcción.

²No se incluyen casos de uso implementados en aplicaciones conexas.

5.5. Arquitectura

Los requisitos son insumo para modelar los elementos del sistema y sus relaciones. El equipo de desarrollo se ha apoyado en diferentes diagramas de estructura y de comportamiento e interacción para describir la estructura y el comportamiento del sistema. El anexo B corresponde a las vistas de la arquitectura que contienen los elementos más interesantes para diferentes partes del sistema. Como puede observarse en dicho anexo, se han aplicado varios patrones de diseño ³ [Larman,2003], [Gamma, E. y Otros, 1994] especialmente se presta atención a mantener los principios de:

- Alta Cohesión
- Bajo Acoplamiento

Asignando responsabilidades teniendo en cuenta:

- Experto en Información.
- Controlador

En la figura 5.1, se muestra el diagrama de interacción correspondiente a la realización del caso de uso no esencial de registrarse en el sistema.

5.6. Modelo de Implementación

El conjunto de diagramas del Anexo B brinda la información fundamental para el modelo de implementación. En el openSITEM se agrupan los diferentes componentes en la jerarquía de carpetas mostrada en la figura 5.2.

Cada una de las carpetas contiene los ficheros de código fuente del producto:

- **Clases:** Contiene los archivos en PHP que implementan las clases.
- **Funciones:** Grupo de funciones en JavaScript para la validación de información en el nodo de usuario. En la actualidad la fase IV contempla complementar esta aproximación con la utilización de AJAX.
- **Configuración:** Alberga el archivo *config.inc.php* que guarda las variables de ingreso a la base de datos. Dichas variables se encuentran codificadas de acuerdo al algoritmo que se seleccione (o implemente) desde la clase *codificar*.

³Basados en General Responsibility Assignment Software Patterns (GRASP) y GoF

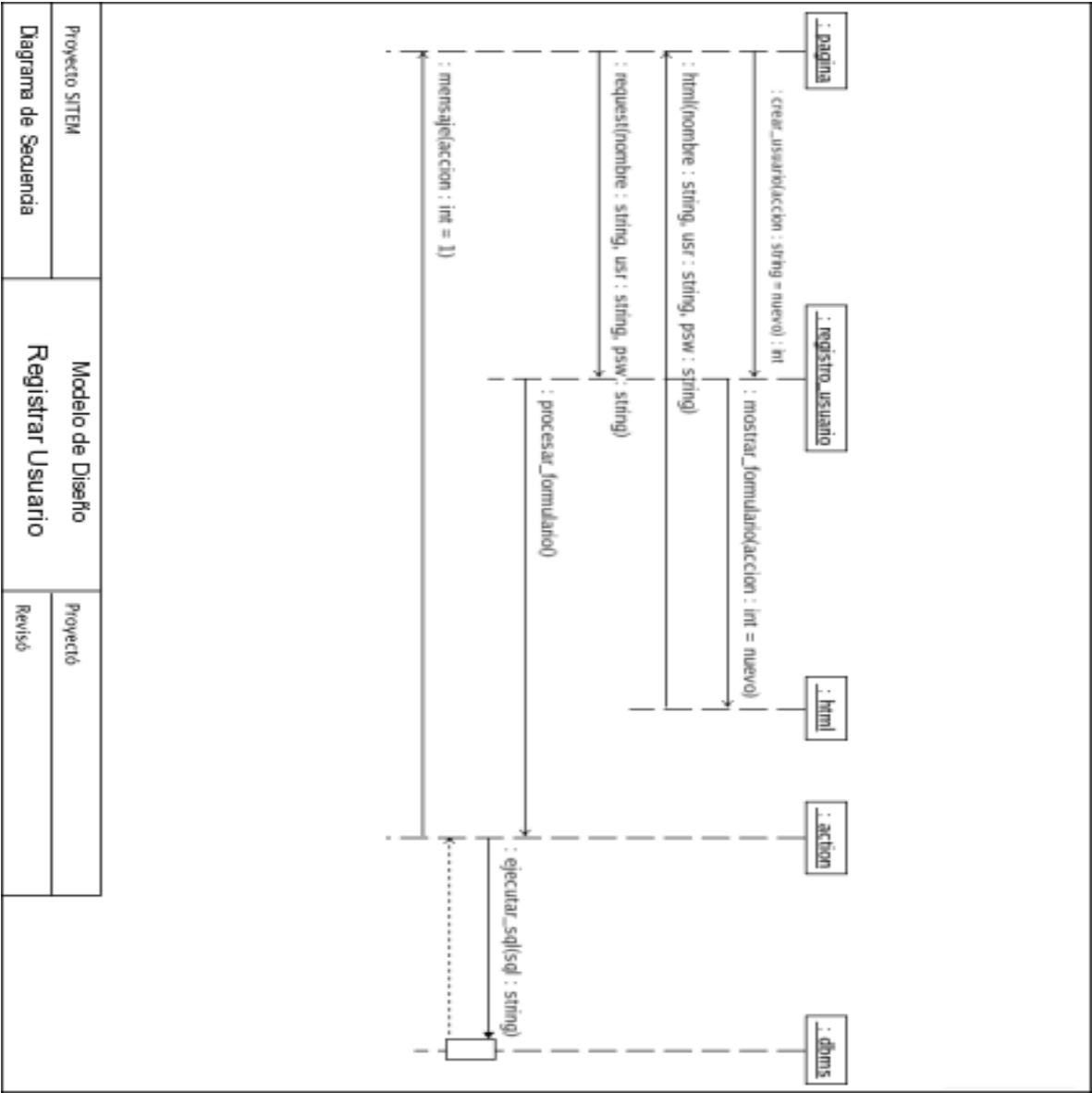


Figura 5.1: Realización del Caso de Uso registrarse en el Sistema

- **Bloques:** Agrupa el código fuente de cada bloque desarrollado en el openSITEM. Un bloque se define como una unidad de funcionalidad independiente que puede utilizarse en cualquier página.
- **Estilo:** Información acerca de los parámetros generales de estilo - tamaño de fuente, color de bordes, fondos, colores de letras, etc; para diferentes componentes del openSITEM. La modificación o inclusión de parámetros afectará la interfaz global del sistema. Actualmente los estilos en el openSITEM se basan en hojas de estilo CSS.
- **Gráficos:** Todos los archivos gráficos usados en el proyecto.
- **Documentos:** Carpeta inicialmente vacía que se utiliza para guardar los archivos que los usuarios carguen a través del protocolo HTTP. Por seguridad se recomienda que esta carpeta se encuentre fuera del directorio en donde se encuentra instalada la aplicación.
- **Instalar:** Contiene el instalador del producto. Esta carpeta debería ser retirada una vez el sitio se encuentre en producción.
- **Desarrollo:** Con varios scripts que facilitan la tarea de desarrollo y adaptación de bloques en el sistema. Estos scripts se han construido pensando en plataformas de desarrollo y prueba por lo que se supone no debe encontrarse en plataformas de producción.

En todo caso, durante el proceso de instalación se puede - y recomienda; asociar nuevos nombres a las carpetas por lo que en teoría ningún desarrollo basado en el openSITEM que esté en etapa de producción debería tener nombres de carpetas conocidos.

5.7. Acerca de la Seguridad en el Sistema

Al tratarse de una aplicación web se tiene grandes retos con la seguridad. Se emplea como referencia el top 10 del proyecto OWASP[OWASP, 2017], que para el año 2017 considera que los riesgos más importantes son:

- Inyección.
- Fallos en la Autenticación.
- Exposición de datos sensibles.
- Entidades externas XML.
- Fallos en el Control de Acceso.
- Fallos en la configuración.
- Cross-Site Scripting (XSS).

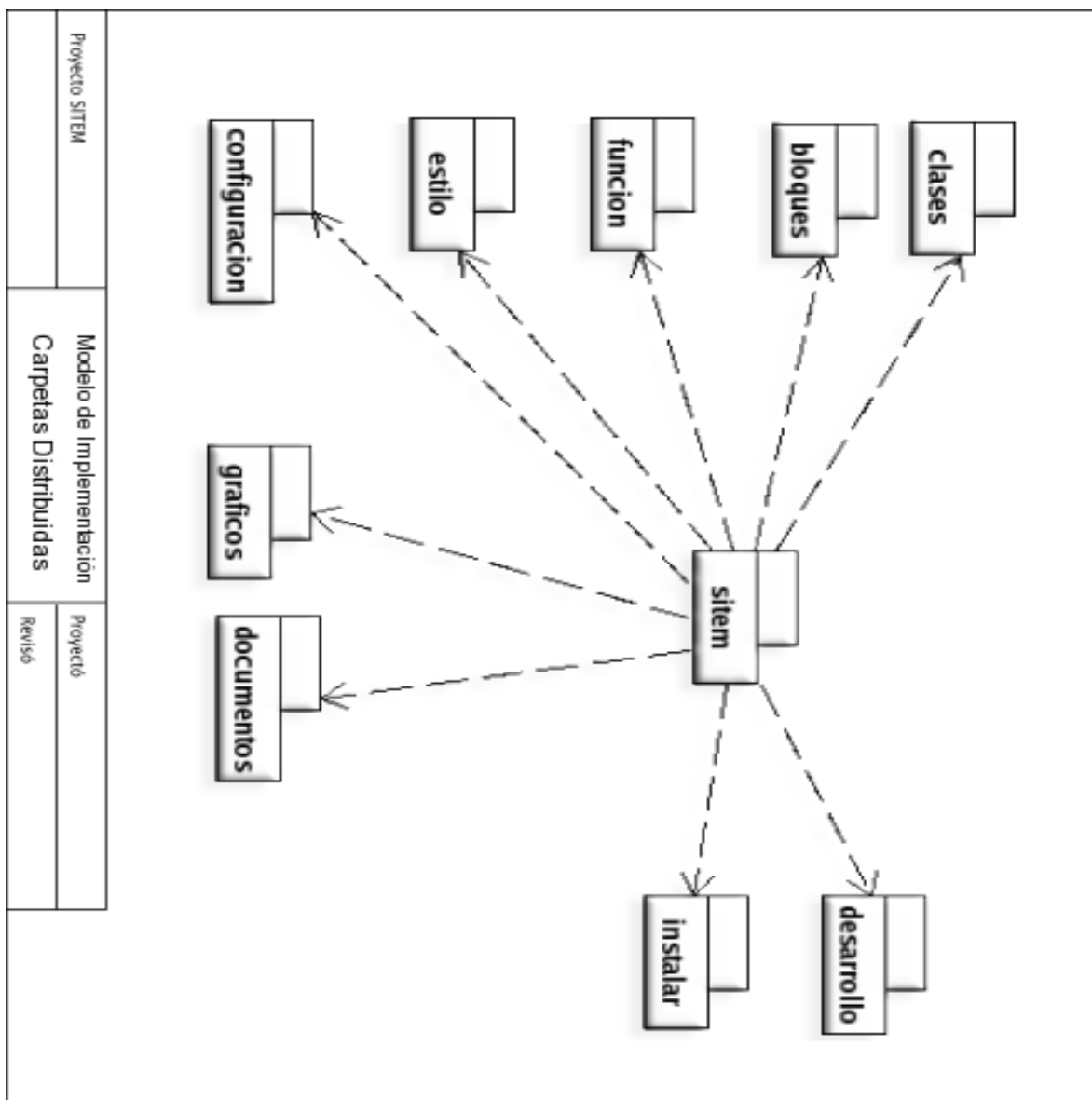


Figura 5.2: Carpetas que se distribuyen con el openSITEM

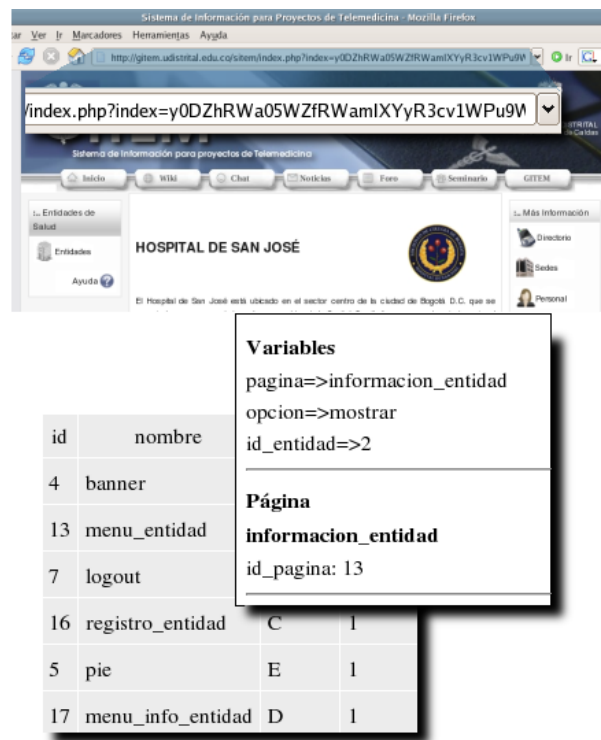


Figura 5.3: URL encriptada. Con la herramienta *Desenlace* el desarrollador puede descifrar los datos

- De-serialización insegura.
- Uso de componentes con vulnerabilidades conocidas.
- Gestión y seguimiento ineficiente a los registros del sistema.

Para cada una de estos riesgos el equipo de desarrollo ha estudiado e implementado las técnicas y controles que la misma guía de la OWASP declara. La documentación del modelo de seguridad está siendo elaborada en el marco de una tesis de pregrado, la cual se encuentra en sus primeras etapas.

Como ejemplo, la figura5.3 muestra la manera que OpenSITEM implementa un control sobre A2:2017, usando el gestor de sesiones de lado de servidor para generar un identificador aleatorio para cada una de las peticiones de página. Dicha técnica se asegura de crear cada cierto periodo de tiempo⁴ una llave única que garantiza un alto nivel de entropía en la cadena codificada - lo que implica que cada recurso tendrá un tiempo de expiración.

En cuanto a la integridad de los datos se tiene un modelo de comparación de contenidos que se activa cada periodo de tiempo, el cual es programable; proponiéndose mantener una copia

⁴Predefinido a 5 minutos

de respaldo verificada y avalada por el administrador. En caso de corrupción o pérdida de datos se mantiene una lista completa de los usuarios del sistema de hasta 1'000.000 de sesiones de tipo desplazamiento en donde el usuario más antiguo es descartado para la inclusión del nuevo cuando el tamaño asignado es completado, asegurándose el manejo eficiente de disco.

Por ningún motivo se permite el acceso a sitios restringidos a usuarios que no hayan sido plenamente identificados en el sistema. En sitios críticos se hace una revisión de los datos de acceso guardados en cookies o se constata los datos de inicio de sesión. Se han evitado al máximo el uso de carácter comodín y todos los accesos a la base de datos son validados en su sintaxis. El openSITEM actualmente propone el uso de protocolos seguros tales como SSL.

Vale la pena destacar el uso de métodos de autenticación de usuario basado en sesiones y codificación de datos que permiten ofrecer un contenido personalizado de acuerdo al perfil de cada uno de los clientes del sistema.

5.8. Modelo de Datos

Dentro del proceso de desarrollo del openSITEM el modelado y elaboración del sistema de bases de datos es una de las partes fundamentales de la propuesta. Se ha venido estructurando el modelo de acuerdo a las necesidades de cada módulo en particular para garantizar la independencia entre ellos en cada una de las capas, incluyendo la de persistencia.

Dependiendo de las características de cada subsistemas se implementan o no políticas transaccionales. El modelo de seguridad en los datos hereda todos los elementos del servidor tales como bloqueos de puertos, ocultación de ventanas, manejo de sockets, etc; además, un esquema lógico de validación por conexiones persistentes complementa estas características.

El anexo D contiene los diagramas de clases que describen la arquitectura de datos del sistema y cada uno de sus subsistemas asociados.

En la actualidad el openSITEM acepta bases de datos PostgreSQL, MySQL y ORACLE. La capa de persistencia del hilo principal se despliega sobre un servidor MySQL.

5.9. Interfaz gráfica.

De acuerdo a los diagramas conceptuales del portal GITEM y de openSITEM. Se han utilizado para la creación de las páginas los conceptos de diseño web enumerado por [Krug,2014], intentando evitar al máximo las páginas sobrecargadas de información. La navegación es guiada mediante enlaces, los cuales están agrupados temáticamente logrando una coherencia en el contenido.

Los gráficos han sido optimizados y su inclusión es necesaria para dar ayuda visual al contenido basado en texto. Teniendo en cuenta que el openSITEM está diseñado para interactuar permanentemente y por periodos prolongados de tiempo con el cliente, se han evitado deliberadamente la utilización de componentes dinámicos. No obstante, el diseño no pierde atractivo ya que su implementación se fundamenta en bibliotecas de uso extendido.

Para reducir el tiempo de acceso al portal, sobre todo cuando se trabaja con conexiones lentas, se da la posibilidad en algunos subsistemas de descargar en formato PDF todo el contenido del grupo de páginas en donde se este ubicado.

5.9.1. Arquitectura de la página

Independiente del subsistema que nos encontremos las páginas siempre están compuestas por cinco secciones denominadas genéricamente con las letras A, B, C, D y E. En ellas se distribuyen los diferentes bloques que conforman la página en una arquitectura Top - Down. Las páginas que no tienen bloques en todas las secciones colapsan aquellas que no se utilizan para dar una impresión visual consistente. Las figuras 5.4 y 5.5 muestran gráficamente el manejo de las secciones en cada página.

5.10. Entregables del proyecto

Los siguientes artefactos – documentos, son generados y utilizados por el proyecto. En el anexo H se tiene extractos importantes de algunos de ellos.

- **Plan General de Trabajo**

- **Modelo de Casos de Uso.** El cual especifica los requerimientos que debe cumplir el módulo de software y que en últimas constituye los contratos que éste, el módulo, tiene con actores externos. Este artefacto estará hecho en su totalidad usando el Lenguaje de Modelado Unificado.

Dentro de este modelo se tienen dos vistas claras: la del negocio y la del sistema. El modelo de Casos de Uso del Negocio ilustra el ámbito del negocio que está siendo modelado. El diagrama contiene actores del negocio y los servicios o funciones que ellos requieren del negocio. El modelo de casos de uso del sistema representa el ámbito de una aplicación. De esto se tiene que un solo modelo de casos de uso del negocio puede tener muchos Modelos de casos de uso asociados, donde cada modelo de casos de uso representa una única aplicación.

- **Modelo de Objetos.** El cual describe la forma en que cada requerimiento – o contrato, es cumplido. Establece las entidades internas, la información que intercambian y los flujos de trabajo que logran el cumplimiento de los requisitos. Los grafos correspondientes podrán incluir diagramas estáticos o interactivos expresados en UML.



Figura 5.4: Arquitectura de una página en el openSITEM

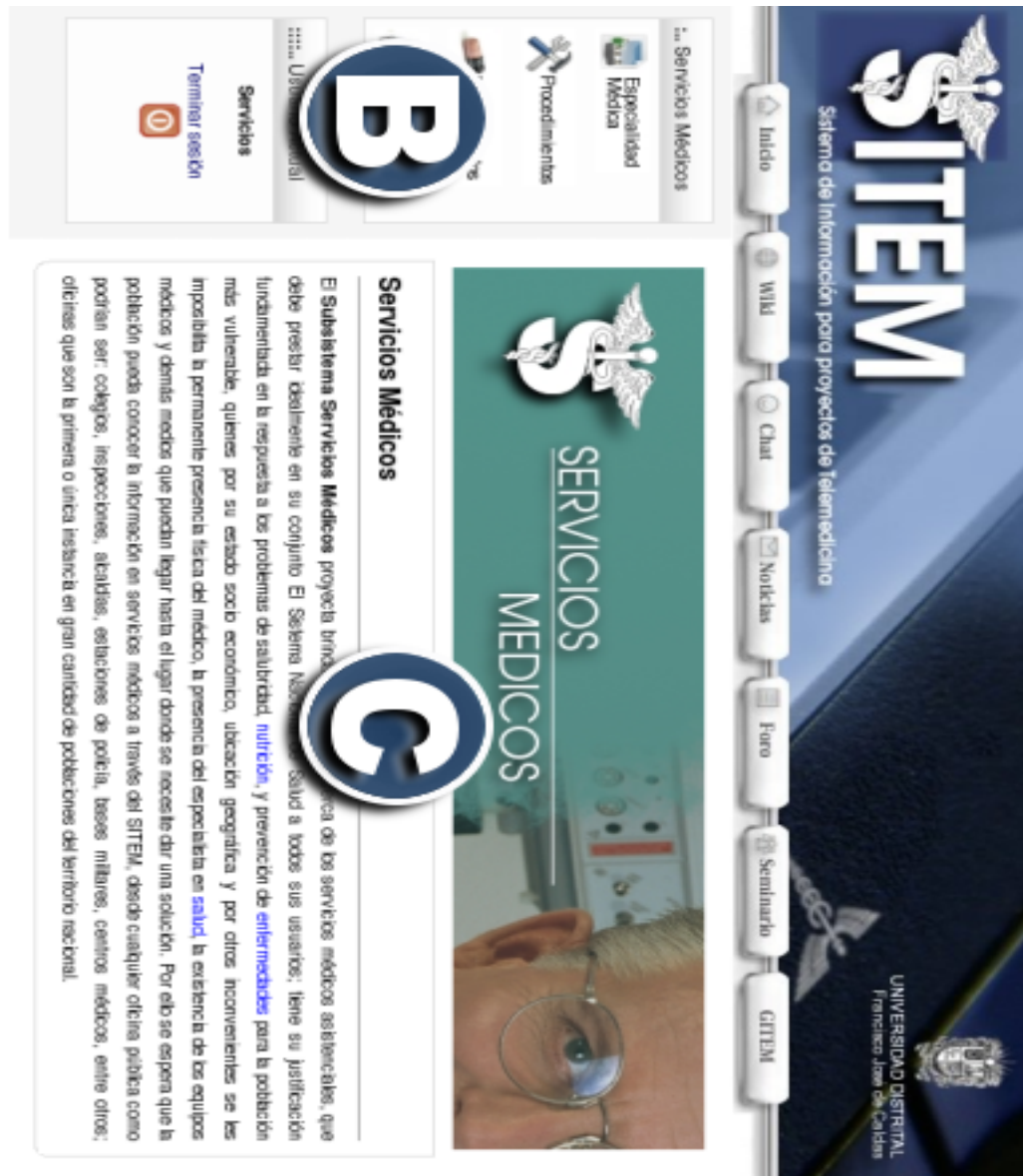


Figura 5.5: Página del openSITEM en donde la Sección D se ha colapsado.

- **Glosario.** Que es el único artefacto válido de consulta para la terminología usada en el desarrollo del openSITEM. Ver el anexo 5.10.
- **Visión.** Este documento define la visión del openSITEM. Es de todos el que marca las pautas conceptuales.
- **Especificaciones Adicionales.** Este documento captura los requisitos que no han sido incluidos como parte de los casos de uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc.
- **Prototipos de Interfaces de Usuario.** Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.
- **Arquitectura.** Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto. Consultar el anexo B.
- **Modelo de Datos.** Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.). El anexo D contiene los artefactos más importantes del modelo de datos.
- **Modelo de Implementación.** Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema.
- **Modelo de Despliegue.** Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes. Ver anexo E
- **Casos de Prueba.** Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

- **Solicitud de Cambio.** Los cambios propuestos para los artefactos se formalizan mediante este documento. Mediante este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. Así se provee un registro de decisiones de cambios, de su evaluación e impacto, y se asegura que éstos sean conocidos por el equipo de desarrollo. Los cambios se establecen respecto de la última baseline (el estado del conjunto de los artefactos en un momento determinado del proyecto) establecida. En nuestro caso al final de cada iteración se establecerá una baseline.
- **Plan de Iteración.** El conjunto de actividades y tareas se orden temporalmente y se le asignan recursos a corto plazo. Se realiza para cada iteración y en todas las fases.
- **Lista de Riesgos.** Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación. El anexo 3.2 contiene la declaración de los riesgos más importantes detectados en el proyecto así como estrategias para minimizarlos.

Bibliografía

- [Acero y Ariza,2012] Acero, D. Ariza, M. (2012)*Requerimientos tecnicos y financieros para la implementacion de una red piloto de Telemedicina en el Hospital Simón Bolívar*.Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [Alhir,2003] Alhir, D. (2003) *Understanding the Unified Process*. Methods and Tools, Martinig Associates.
- [Anderson,2016] Anderson, D. y Carmichael, A. (2016). *Essential Kanban Condensed*. Seattle, WA: Lean Kanban University Press.
- [Aparicio-Ramirez,2003] Aparicio,L. y Ramírez, J. (2003).*Arquitectura de Red de Telemedicina*, Centro de Investigaciones y Desarrollo Científico, Universidad Distrital F.J.C.
- [Aparicio,2000] Aparicio,L.*Propuesta de Estudio Red de Telemedicina Bogota*, Grupo GI-TEM, Universidad Distrital F.J.C, 2000.
- [Ardila,2011] Ardila,J y Ardila, M. (2011)*Evaluación y diagnóstico de los servicios básicos y especializados al servicio de la salud de la clínica San Pedro Claver para el desarrollo de la red de Telemedicina de Bogotá*.Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [Balduino,2010] Balduino, R. (2010).*Introduction to OpenUP (Open Unified Process)*.Fundación Eclipse. Disponible en: <https://www.eclipse.org/>. Accedido el: 2017-06-20
- [Bashshur,1977] Bashshur, R., Lovett J. (1977) *Assessment of telemedicine: results of the initial experience*. Aviation, Space and Environmental Medicine.
- [Bashshur,1995] Bashshur, R. (1995). *On the Definition and Evaluation of Telemedicine*. *Telemedicine Journal*. Volume 1, Number 1, Mary Ann Liebert, Inc., Publishers.
- [Beck,1999] Beck, K. (1999) *Extreme Programming Explained: Embrace Change* . Addison-Wesley.
- [Benafia,2016] Benafia, A., Mazouzi, S. y Maanru, R. (2016). *From Linguistic to Conceptual: A Framework Based on a Pipeline for Building Ontologies from Texts*. Journal of Advanced Computational Intelligence and Intelligent Informatics.
- [Bergeron,2003] Bergeron, B. (2003). *Essentials of knowledge management*. Jhon Wiley & Sons, New Jersey.

- [Bos,2004] Bos,B. (2004).*A proposal for Webapps*. W3C Consortium.
- [Boggs,2002] Boggs,W. Boogs, M. (2002).*UML with Rational Rose 2002*. SyBex.
- [BCE,2003] Britannica Concise Encyclopedia.*Anarchism..* Obtenido en julio 18, 2003, desde: <http://search.eb.com/ebc/article?eu=380585>.
- [Brugge, Dutoit, 2000] Brugge, B. y Dutoit, A. (2000).*Object-Oriented Software Engineering*. Prentice Hall.
- [McCarty, 2006] Carty, J. (2006). *Dynamics of Software Development*.Microsoft Press.
- [Cockburn, 1999] Cockburn, A. (1999). *The Impact of Object-Orientation on Application Development*. IBM Systems Journal 38. Páginas 308-332.
- [Ley 23,1982] Congreso de la República. (1982). *Ley 23 de 1982: Sobre Derechos de Autor*. República de Colombia.
- [Dec 1360,1989] Congreso de la República. (1989). *Decreto 1360 de 1989: Inscripción del soporte lógico (software) en el Registro Nacional del Derecho de Autor*. República de Colombia.
- [Ley 44,1993] Congreso de la República. (1993). *Ley 44 de 1993: Modifica y adiciona la Ley 23 de 1982 y modifica la Ley 29 de 1944*. República de Colombia,.
- [Ley 565,2000] Congreso de la República. (2000). *Ley 565 de 2000: adopción del Tratado de la OMPI sobre Derechos de Autor*. República de Colombia.
- [CRT,QoS,2006] Comisión de Regulación de Telecomunicaciones. (2006). *Condiciones de Calidad en Servicios de Telecomunicaciones*. República de Colombia.
- [CRT,Indicadores,2006] Comisión de Regulación de Telecomunicaciones. *Indicadores de Calidad en Servicios de Telecomunicaciones*. República de Colombia, 2006.
- [CRT,QoS,2007] Comisión de Regulación de Telecomunicaciones.*Proyecto de Resolución para Indicadores de Calidad de Servicio*. República de Colombia, 2007.
- [Corte1319,2001] Corte Constitucional de Colombia.*Sentencia T-1319 de 2001*. República de Colombia, 2001
- [Craig, 2005] Craig J, Patterson V. *Introduction to the practice of telemedicine*. Journal of Telemedicine and Telecare. 2005. Págs 3-9.
- [Currie y otros,2014] Currie W. y Seddon J. A cross-national analysis of eHealth in the European Union: some policy and research directions. Inf Manage. 2014.
- [Davenport,1998] Davenport, T. y Prusak, L. *Working Knowledge: How Organizations Manage What They Know*, Harvard Business School Press, Boston.1998
- [Davies,2011] . Davies, M. (2011). Knowledge – Explicit, implicit and tacit: Philosophical aspects. , International Encyclopedia of Social and Behavioral Sciences, Elsevier.

- [Dueck,2001] Dueck,G. (2001).*Views of knowledge are human views*.IBM Systems Journal, Volumen 40, Número 4.
- [Duque *et al*,2012] Duque,J. García J, Caicedo, D. (2012).*Estudio sobre los requerimientos técnicos y financieros para implementar los servicios telemédicos en el Hospital El Tunal, para el proyecto telemedicina Bogotá*..Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [UN, 2005] Department of Economic and Social Affairs. (2005).*UN Global E-government Readiness Report 2005 From e-government to e-inclusion*. Naciones Unidas.
- [Eclipse, 2012] Fundación Eclipse.*OpenUP Versión 7*. Disponible en <http://epf.eclipse.org/>. Consultado en 2016-02-14.
- [erl, 2017] Earl, T. (2017).*Service-Oriented Architecture: Analysis and Design for Services and Microservices*. 2da Edición. Prentice Hall.
- [Engineering, 2017] . Engineering. (2017). Knowage Portal. Consultado el 2017-08-14. Disponible en <https://www.knowage-suite.com/>.
- [Firestone,2001] Firestone, J. (2001)*Key Issues in Knowledge Management. Knowledge and Innovation*. Knowledge Management Consortium International. Volumen 1.
- [Gamma, E. y Otros, 1994] Gamma, E., Helm, R., Johnson, R y Vlissides, J. (1994). Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional Computing Series.
- [Gang,2007] Gang, L. y Yi, L. (2007). *A Relational Model of Knowledge Share, Knowledge Acquisition and Product Innovation*. Universidad Xi'an Jiaotong. China..
- [Garcia, 2008] Garcia, F. (2008). El cuestionario: Recomendaciones metodologicas para el diseno de un cuestionario. Editorial Limusa.
- [Girard,2015] Girard, J. (2015). *Defining knowledge management: Toward an applied compendium*. Online Journal of Applied Knowledge Management. International Institute for Applied Knowledge Management.
- [González y Torres,2012] González, O. Torres, J. (2012).*Evaluación y Diagnóstico de los Servicios Básicos y Especializados al Servicio de la Salud del Hospital de San José para el Desarrollo de la Red de Telemedicina de Bogotá*. Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [Gottesdiener, 2002] Gottesdiener, E. (2002). Requirements by Collaboration: Workshops for Defining Needs. Addison-Wesley Professional.
- [Guarin *et al*,2013] Guarín, S. García, M. Torres, L. (2013)*Diagnóstico de las Redes Eléctrica, Telefónica y de Datos del Hospital Rafael Uribe Uribe E.S.E*. Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [Hibbard,1997] Hibbard, J.(1997).*Knowledge management—knowing what we know*. Information Week.

- [Huang *et al*,2011] uang, L., Milne, D., Eibe, F. y Witten,E. (2011). Learning a Concept-based Document Similarity Measure. Departamento de Ciencias de la Computación. Universidad de Waikato. Nueva Zelanda.
- [Hurtado,2000] Hurtado,J. (2000).*Metodología de la Investigación Holística*. Sygal, Caracas, Venezuela.
- [IEEE, 1990] IEEE Institute. (1990).*IEEE Standard Glossary of Software Engineering Terminology - IEEE std 610.12-1990*. IEEE.
- [Koch, 2005] Koch, S. (2005) *Free Open Source Software Development*. Idea Group.
- [Liebowitz,1998] Liebowitz, J. y Beckman, T. (1998). *Knowledge Organizations: What Every Manager Should Know*. Boca Raton, St. Luci press.
- [ISO,2011] International Organization for Standardization. (2011) *ISO 42010:2011 Systems and software engineering - Architecture description*.
- [ITU-T, 2004] ITU-T. (2004). *Manual Calidad de servicio y calidad de funcionamiento de la red*. Unión Internacional de Telecomunicaciones.
- [ITU-OMS, 2012] ITU - OMS. (2012). *National eHealth Strategy Toolkit*. Unión Internacional de Telecomunicaciones.
- [G1000, 2001] ITU-T. (2001). *Recomendación G.1000 - Calidad de servicio en las comunicaciones: Marco y definiciones*. Unión Internacional de Telecomunicaciones.
- [G1010, 2001] ITU-T. (2001).*Recomendación G.1010 - Categorías de calidad de servicio para los usuarios de extremo de servicios multimedios*. Unión Internacional de Telecomunicaciones.
- [Jacobson,2000] Jacobson,I. Booch,G y Rumbaugh,J. (2000).*El Proceso Unificado de Desarrollo de Software*. Addison-Wesley, Madrid.
- [Jacobson,2005] Jacobson,I. Booch,G y Rumbaugh,J. (2005). *The Unified Modeling Language Reference Manual*, segunda edición. Addison-Wesley, Boston.
- [Jackson,2005] Jackson,D.(2005).*The W3C Workshop on Web Applications and Compound Documents*. W3C Consortium.
- [Krug,2014] Krug,S. (2014)*Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*. New Riders.
- [Larman,2004] Larman,C. (2004). *Agile and iterative development: a manager 's guide*. Addison Wesley.
- [Larman,2003] Larman, C. (2003) *UML y Patrones. Una introducción al análisis y diseño orientado a objetos y al Proceso Unificado*. Pearson Educación S.A. Madrid.
- [Lewis, 2012] Lewis T, Synowiec C, Lagomarsino G y Schweitzer J. (2012). E-health in low-and middle-income countries: findings from the Center for Health Market Innovations. Bull World Health Organ.

- [Martínez,1998] Martínez, R. Martín, F. (1998). *LANDSCAPE: A Knowledge-Based System for Visual Landscape Assessment..* IEA/AIE, Volumen 2. Springer. Páginas 849-856.
- [CONPES3072,2000] Ministerio de Comunicaciones. (2000). *Documento CONPES 3072 - Agenda de Conectividad*. República de Colombia.
- [MINSALUD,2016] Ministerio de Salud y Protección Social. (2016). *Estudio Exploratorio de la Situación de la Telemedicina en Municipios Priorizados - Colombia*. República de Colombia.
- [MINSALUD-4678,2015] Ministerio de Salud. (2015). *Resolución Número 4678 de 2015*. República de Colombia.
- [MINSALUD-1448,2006] Ministerio de Salud. (2006). *Resolución Número 1448 de 2006*. República de Colombia.
- [Barrero,2010] Muñoz, F. Barrero, J. (2010). *Evaluación y Diagnóstico de los Servicios Básicos y Especializados al Servicio de la Salud del Hospital La Victoria para el Desarrollo de la Red de Telemedicina de Bogotá*. Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [Nokata,1994] Nonaka, I.(1994). *A dynamic theory of organizational knowledge creation*. Organization Science,5. pp. 14-37.
- [Nokata,1995] Nonaka,I. y Takeuchi, H. (1995). *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press.
- [OAVES, 2017] Observatorio Así Vamos en Salud. (2017). *Tendencias de la Salud en Colombia. Informe Anual 2016*.
- [OCDE, 2015] Organización para la Cooperación y el Desarrollo Económico. (2015). *OECD Reviews of Health Systems: Colombia 2016*. OECD Publishing, Paris.
- [OMG, 2007] Object Management Group. (2007). *Unified Modeling Language: Specification. Versión 2.1.1*. Object Management Group.
- [OMG, 2007a] Object Management Group. (2007). *Unified Modeling Language: Superstructure. Versión 2.1.1*. Object Management Group.
- [OpenProject, 2017] OpenProject. *Portal del Sistema OpenProject*. Disponible en: <https://www.openproject.org>.
- [OMS, 2010] Organización Mundial de la salud. (2010). *Telemedicine: opportunities and developments in Member States: report on the second global survey on eHealth 2009*.
- [OMS, 2016] Organización Mundial de la salud. (2016). *Global diffusion of eHealth: making universal health coverage achievable. Report of the third global survey on eHealth*.
- [OPS, 2011] . Organización Panamericana de la Salud. (2011). *Estrategia y Plan de Acción sobre eSalud*. 51 Consejo Directivo.

- [OWASP, 2017] . Open Web Application Security Project. (2017). OWASP Top 10 - 2017 The Ten Most Critical Web Application Security Risks.
- [Patton,2014] . Patton, J y Economy, P. (2014). User Story Mapping: Discover the Whole Story. O'Reilly Media.
- [Pressman, 2006] Pressman, R. (2006). *Ingeniería del Software*. Sexta Edición. Mc Graw Hill, México.
- [Raymond, 2010] Raymond,E.(2010) *The Cathedral and the Bazaar*, Revision 1.87.
- [AIM,1993] *Research and technology development on telematics systems in health care: AIM 1993; Annual Technical Report on RTD: Health Care*. Comisión Europea: Dirección General XIII, 1993.
- [Ross y Otros,2016] Roos, I y Otros. (2016). *Factors that influence the implementation of e-health: a systematic review of systematic reviews*. Research Department of Primary Care and Population Health, University College London.
- [Rozo et al,2002] Rozo, O. Valencia, S. Barahona, F. (2003). *Estudio diagnóstico de las condiciones técnicas y financieras en instrumentos y equipos médicos y de servicios de la clínica San Pedro Claver para la implentación de los servicios telemédicos*. Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [Ruiz y Niño,2012] Ruiz, M. Niño, D. (2012). *Evaluacion y Diagnostico de los Servicios Basicos y Especializados al Servicio de la Salud del Hospital El Tunal para el Desarrollo de la Red de Telemedicina de Bogotá* Grupo de Investigación en Telemedicina, Universidad Distrital Francisco José de Caldas.
- [Salazar,2002] Salazar,J. y Kopec, A. (2002). *Aplicaciones de Telecomunicaciones en Salud en la Subregion Andina - Telemedicina*, Organismo Andino de Salud, OPS.
- [Sarmiento,2005] Sarmiento,A. (2005). *Issues of human computer interaction*.IRM Press, Londres.
- [Sowa,1984] Sowa, J. (1984). *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley.
- [stallman,2002] Stallman,R. (2002). *Free Software, Free Society:Selected Essays of Richard M. Stallman*. GNU Press, Bosotón.
- [Tatnall, 2003] Tatnall,A. (2005). *Web portals:The New Gateways to Internet Information and Services*. Idea Group, Londres.
- [BDT,1999] *Telemedicine And Developing Countries - Lessons Learned*. Document 2/116-E. ITU-D STUDY GROUPS. Question 14/2: Fostering the application of telecommunication in health care. Identifying and documenting success factors for implementing telemedicine. 1999.
- [Bangemann,1994] Bangeman, C. (1994).The Bangemann Report: Europe and the global Information Society. Recomendaciones al Consejo Europeo. Bruselas.

- [theopengroup,2016] The Open Group (2016). *ArchiMate 3.0.1 Specification*, Van Haren Publishing.
- [Turban,1992] Turban, E. *Decision Support and Expert Systems - Management Support Systems*. Collier Macmillan, Sydney,1992.
- [Wiig,1993] Wiig,K. (1993). *Knowledge Management Foundations*.Schema Press, Arlington.
- [Wielinga *et al*, 1992] Wielinga, B. (1992). *et al. KADS: A Modelling Approach to Knowledge Engineering*.Knowledge Acquisition Journal, 4(1). Páginas 5-53.
- [Wilson, 2017] Wilson, R., (2017). *Successful Digital Health Systems: Guidelines for Healthcare Leaders and Clinicians*. Memorias de e-Health 2017 Virtual Meeting. Canada.
- [Yellowlees, 2005] Yellowlees P. (2005). *Successfully developing a telemedicine system*. Journal of Telemedicine and Telecare. Págs:331-335.
- [Zabala, 2000] Zavala R. (2000) *Diseño de un Sistema de Información Geográfica sobre internet*. Tesis de Maestría en Ciencias de la Computación. Universidad Autónoma Metropolitana-Azcapotzalco. México, D.F.
- [Zellerino, 2009] Zellerino, B., Milligan, S., Brooks, R., Freedenberg, D. y Collingridge, D. (2009). *Development, testing, and validation of a patient satisfaction questionnaire for use in the clinical genetics setting*, American Journal of Medical Genetics Part C: Seminars in Medical Genetics. Wiley Subscription Services, Inc.

Glosario

ADSL - (Asymetric Digital Subscriber Line). Línea digital asimétrica del suscriptor. Tecnología que permite la transmisión de información digital a altas velocidades de descarga utilizando medios de transmisión convencionales.

ALGORITMO - Conjunto de reglas o procedimientos que representa la solución a un problema. Un programa de software es la transcripción de un algoritmo.

ANCHO DE BANDA - Cantidad de datos que pueden transmitirse en determinado periodo de tiempo por un canal de transmisión.

APACHE - Servidor Web de distribución libre, poderoso y flexible. Implementa los últimos protocolos HTTP. Altamente configurable y extensible con módulos de terceros. Fue desarrollado en 1995 y ha llegado a ser el más usado de Internet.

API - (Application Program Interface). Interfaz de programación de Aplicaciones.

APRENDIZAJE POR COMPUTADOR - Hace referencia al uso de computadores como parte clave en la enseñanza y el aprendizaje haciéndolo parte integrante del entorno educativo.

APRENDIZAJE EN LINEA - Hace referencia a aquel tipo de aprendizaje que se lleva a cabo mediante la utilización de una red o INTERNET.

ARTEFACTO - Cualquier tipo de información que es producida o usada en el proceso de desarrollo de software.

BASE DE DATOS (MOTOR) - Aplicación informática que permite la gestión de datos y el manejo de la información.

CALL (Computer-assisted language learning) - Es un pseudo lenguaje que permite aproximar la enseñanza y aprendizaje en el que tecnología computacional es usada ya sea como presentación, refuerzo y acceso a material para aprender. Usualmente incluye elementos interactivos y contenidos multimedia.

CASO DE USO – Artefacto utilizado para describir la funcionalidad de un sistema desde el punto de vista de los usuarios.

CBR - (Constant Bit Rate). Tasa de bits constante.

CGI - (Common Gateway Interface) Estándar para tender interfaces entre aplicaciones externas y servidores de información, tales como los servidores HTTP.

COOKIE - Pequeño archivo de texto que se almacena en el disco duro al visitar una pagina Web y que sirve para identificar al usuario cuando se conecta de nuevo a dicha página.

CLASE - En el desarrollo de software es la representación mediante un nombre, atributos y operaciones de un objeto o conjunto de objetos.

CRUD - Caso CRUD, caso que agrupa los casos de crear, leer, actualizar y borrar registros.

CSS - (Cascading Style Sheet - Hojas en estilo de cascada). Reglas de estilo para documentos HTML.

DAEMON - (Disk And Execution MONitor) Un programa no invocado explícitamente pero que permanece esperando a que una situación especial ocurra Los sistemas Unix ejecutan muchos daemons, principalmente para manipular solicitudes de servicios desde otro host o desde una red.

DBMS - (Data Base Management System – Sistema de Gestión de Bases de Datos). Sistema de software que permite a los usuarios guardar y modificar información.

DEPURAR - Relacionado con el software, detectar, localizar, corregir los problemas en un programa informático.

DESARROLLO ITERATIVO - Método de construcción de productos cuyo ciclo de vida está compuesto por un conjunto de iteraciones, las cuales tienen como objetivo entregar versiones del software.

DISCIPLINA - Conjunto de reglas para mantener el orden y la subordinación entre miembros de un cuerpo.

DOMINIO - Parte del nombre jerárquico con que se conoce a cada entidad conectada a Internet. Un dominio se compone de una serie de etiquetas o nombres separados por puntos.

DSL - (Línea digital de abonado). Tecnología de red pública que proporciona ancho de banda amplio sobre el cableado tradicional de cobre en distancias limitadas. Hay cuatro tipos de DSL: ADSL, HDSL, SDSL y VDSL. Todos ellos se aprovisionan a través de pares de módems, estando un módem situado en la oficina central y el otro en el domicilio del abonado.

ENTORNO DE APRENDIZAJE - Un entorno de aprendizaje es el conjunto de conocimientos, herramientas de enseñanza y aprendizaje, espacios y personas involucradas en un proceso de aprendizaje dado.

ENTORNO VIRTUAL DE APRENDIZAJE - Un entorno virtual de aprendizaje (virtual learning enviroment) es un paquete de software cuya función es reemplazar o complementar un entorno de aprendizaje formal y tradicional.

eSALUD - De acuerdo con el Instituto de Estándares de Comunicaciones Europeo, es la aplicación de tecnologías de la información y la comunicación en todo el rango de funciones que afecta el sector salud.

FILTRO - Cualquier función o programa que seleccione información automáticamente con un criterio preestablecido.

GNU – (GNU's Not Unix). Sistema operativo, compuesto de pequeñas piezas individuales de software totalmente libre.

HIPERTEXTO - Método de preparar y publicar un texto ideal para el computador, con el que los lectores pueden escoger su propia ruta a través del material. La información se descompone en pequeñas unidades y después los hipervínculos se insertan en el texto.

HIPERVINCULO - Enlace. En un sistema de hipertexto, palabra subrayada o destacada que, cuando se pulsa en ella lleva a otro documento.

HTML - (Hyper text markup language – Lenguaje de marcado de hipertexto). Lenguaje de marcado que define la estructura de paginas Web. Utiliza etiquetas para denotar los diferentes objetos que componen una página.

HTTP - (Hyper Text Transfer Protocol - Protocolo de Transferencia de Archivos). Estándar para la transferencia de mensajes entre navegadores web utilizando texto plano.

INCREMENTAL - Que puede instalarse por fases, cada una de ellas añadiendo nuevas funcionalidades.

INFRAESTRUCTURA - Conjunto de medios necesarios para el desarrollo de una actividad.

INGENIERÍA DE SOFTWARE - Aplicación de un enfoque sistemático al diseño, desarrollo, operación y mantenimiento de software.

INGENIERÍA INVERSA – Conjunto de etapas desarrolladas para obtener las bases del diseño y la forma de implementación de algún producto de ingeniería a partir de su estado final.

INTERCONEXIÓN - Unión o conexión entre sí de dos o más elementos.

INTERFAZ - Parte de un sistema en la que se desarrolla la comunicación o interacción entre el programa y el usuario (siendo este humano u otro sistema).

INTEROPERABILIDAD - Según IEEE, es la habilidad de dos o más sistemas para intercambiar mensajes y utilizar la información intercambiada.

IP – (Internet Protocol - Protocolo Internet). Protocolo de la capa de red de la pila TCP / IP que ofrece un servicio de intercambio de paquetes de datos entre host.

ITERACION - Repetición de un conjunto de pasos.

Kbps - Kilobits por segundo.

KBps - KiloBytes por segundo.

MÉTODO - Modo ordenado de proceder para llegar a un resultado o fin determinado.

METODOLOGÍA - Parte de la lógica que estudia los métodos.

MITIGAR - Moderar, suavizar.

MODELADO – Descripción, en un lenguaje de máquina, de la forma o características de un objeto o conjunto de objetos.

MÓDULO - Una aplicación software que presenta una funcionalidad específica dentro de OpenSITEM. En el contexto del proyecto es sinónimo de subsistema.

MULTIPLATAFORMA - Las aplicaciones pueden ser vistas en cualquier computador del mundo a través de múltiples plataformas (sistemas operativos, navegadores y hardware).

MySQL - Sistema de administración para bases de datos relacionales (rdbms).

NRT-VBR - (Non-Real-Time Variable Bit Rate). Tasa de bits variable en tiempo no real, Esta categoría de servicio está encaminada para aplicaciones en tiempo no real, las cuales tienen características de tráfico en ráfaga.

NODO - Componente de una red de eSalud. Tiene atributos que definen su tipo y capacidad de interconectarse e interoperar con otros nodos de igual o diferente tipo. Ejemplos de nodos en OpenSITEM: Un electrocardiógrafo (tipo Equipo Médico), un hospital (tipo Entidad de Salud), un radiólogo (tipo Especialista), una toma eléctrica (tipo Componente Eléctrico), un smartphone (tipo Dispositivo de Comunicación), etc.

NTP - (Network Time Protocol – Protocolo de Tiempo de red) - Sistema de sincronización de tiempo para relojes de computadores a través de Internet.

OBJETO - Un objeto es una instancia, un ejemplar, de una clase. Por extensión del término, en openSITEM se considera que un nodo es un ejemplar de una categoría de nodos.

OMG - Object Management Group. Organización que se encarga de crear y actualizar varios estándares de tecnologías orientadas a objetos incluyendo UML y BPMN.

OSI - Internetworking de sistemas abiertos. Programa internacional de normalización creado por ISO e ITU-T. Para desarrollar estándares para las redes de datos que faciliten la interoperabilidad entre equipos de varios fabricantes.

PASSWORDS - Contraseña. Método de seguridad empleado para identificar a un usuario. Sirve para dar acceso a personas con determinados permisos. En la actualidad se están generando de manera dinámica y distribuyendo vía correo electrónico o mensajes de texto.

PCR - (Peak Cell Rate). Tasa pico de celda.

PHP - (Hypertext Preprocessor"). Es un lenguaje interpretado de alto nivel ejecutado al lado del servidor. Según w3techs.com, para octubre de 2017 cerca del 80 % del total de sitios web del mundo empleaban este lenguaje.

PLANTILLAS - Serie de archivos que definen la apariencia de una aplicación Web, y permiten cambiar la presentación de la aplicación con sólo modificar unos cuantos archivos sin necesidad de modificar el código que le da funcionalidad a la aplicación.

PLATAFORMA - Base o elemento de apoyo. Se refiere a una combinación específica de hardware y sistema operativo.

PROTOCOLO - Conjunto de instrucciones o lenguaje que utilizan los computadores para comunicarse entre sí.

PROTOTIPO - Original ejemplar o primer molde en que se fabrica una figura u otra cosa. Modelo original que sirve como ejemplo para futuros estados o formas.

QoS - (Quality Of Service - Calidad de servicio). Una medida de rendimiento en el sistema de transmisión que refleja la calidad de la transmisión y la disponibilidad del servicio.

RDSI - (Red digital de servicios integrados). Protocolo de comunicaciones que permite a las redes de las compañías telefónicas transportar datos, voz y otro tráfico.

REGISTRO - Pequeña unidad de almacenamiento que contiene cierto tipo de datos. En el ámbito de las bases de datos es cada una de las fichas que forman un conjunto.

RT-VBR - (Real-Time Variable Bit Rate). Está encaminada a aplicaciones que requieren restricción en la variación de retardo (lo mínimo) y variación de retardo, que podrían ser apropiados para voz y vídeo en tiempo real.

SCRIPT - Pequeños archivos de texto embebidos dentro del documento, que ofrecen una forma de extender los documentos HTML convirtiendo el contenido del documento en una página dinámica, pudiendo ser ejecutados al presentarse un evento como el movimiento del ratón.

SCORM (Sharable Content Object Reference Model) - Es una colección de normas y especificaciones para aprendizaje basado en la Web. Define como debe ser la comunicación entre el contenido de lado al cliente y un sistema servidor, tratando aspectos como el empaquetamiento de información en archivos ZIP.

SERVIDOR - Dispositivos de un sistema que resuelve las peticiones de información de otros elementos del sistema a los que se denomina clientes.

SISTEMA DE INFORMACIÓN - Conjunto de componentes interrelacionados que permiten capturar, procesar, almacenar y distribuir la información para apoyar la toma de decisiones y el control en una institución.

SQL - (Structured Query Language Lenguaje Estructurado de Consultas). Lenguaje de programación interactivo y estándar para obtener y actualizar información de una base de datos.

TASA DE BITS - Velocidad en Kilobits por segundo a la que puede transmitir un circuito virtual.

TCP/IP - TCP("Transmisión Control Protocol") e IP("Internet Protocol"). Protocolo para el control de la transmisión / Protocolo Internet. Nombre común que se da al paquete de protocolos desarrollados por el DoD en los años 70's con el fin de soportar la construcción de interworks a escala mundial.

TECNOLOGÍA - Conjunto de los conocimientos técnicos. Conjunto de los instrumentos y procedimientos industriales de un determinado producto.

TELEASISTENCIA - Los sistemas de teleasistencia permiten a los pacientes recoger datos acerca de su enfermedad, transmitir la información a un lugar remoto y usar la videoconferencia para discutir sus padecimientos y tratamientos con el profesional de la salud.

TELECONFERENCIA - Es la telecomunicación que se establece entre dos o más especialistas, para tratar diversos temas que en el caso de la Telemedicina, generalmente tienen que ver con los pacientes o están relacionados con capacitación especializada.

TELEMEDICINA - Herramienta tecnológica para brindar servicios médicos a distancia mediante el intercambio de imágenes, voz, datos y video, haciendo uso de las tecnologías de la información y de las comunicaciones, permitiendo el diagnóstico, la opinión de casos clínicos y la provisión de cuidados de salud y educación médica.

TRANSICIÓN - Tiempo en el cual se cambia de un estado a otro.

UBR - (Unspecified Bit Rate). Esta categoría está encaminada para aplicaciones en tiempo no real, por ejemplo aquellas que no requieren restricciones en la variación de retardo (mínimo) y variaciones de retardo.

UML - (Unified Modeling Language - Lenguaje Unificado de Modelado). Brinda todas las características, tanto sintácticas como semánticas, para lograr caracterizar lógicamente cualquier tipo de software permitiendo ser utilizado en cualquier etapa del diseño y especialmente útil en aquellos desarrollos enfocado a objetos.

UP - Proceso Unificado. El Proceso Unificado puede concebirse como una idea, una plantilla que provee una infraestructura para ejecutar proyectos pero que no tiene en cuenta todos los detalles requeridos en dicha ejecución.

USUARIO - Dicho de una persona: que tiene derecho de usar de una cosa con cierta limitación.

VIDEOCONFERENCIA - Comunicación simultánea bidireccional de audio y vídeo.

WEB - Sistema de comunicación para el intercambio de mensajes a través de Internet.

WWW - (World Wide Web). Sistema mundial de hipertexto que utiliza Internet como mecanismo de transporte. Conjunto de páginas de información con texto, imágenes sonido, etc., enlazadas entre sí.

XML - (Extended Markup Language). Lenguaje de marcado extendido. Permite definir estructuras de información empleando etiquetas organizadas de manera jerárquica.

Anexo A

Modelo de Requisitos

El modelo de requisitos es un pilar importante para garantizar que la funcionalidad implementada es de interés para un actor específico. se emplea el mecanismo de casos de uso para describir la funcionalidad para describir la funcionalidad con un alto grado de abstracción técnicas.

A.1. Modelo de Casos de Uso

A.1.1. Características Generales de los Actores

El sistema ofrece funcionalidad que es de interés para alguno de los siguientes actores (figura A.1):

- **Administrador:** Se encarga de la gestión de usuarios, el registro de nuevas categorías de nodos y tareas de administración tales como copias de seguridad, corrección de errores, edición de páginas, etc.
- **Entidad de Salud:** Para la gestión específica de un nodo tipo Entidad de Salud. Un usuario de este tipo está confinado a la institución que crea. Para poder gestionar información de otra entidad deberá solicitar autorización a la instancia de actor dueña del registro.
- **Servicios Médicos:** Gestiona la información de nodos tipo Servicio Médico.
- **Especialista:** Con permisos para gestionar nodos tipo Especialista (profesionales de la salud).

- **Profesional TIC:** Actor abstracto que agrupa los actores que gestionan información de los nodos tipo: Equipo Médico, Operador de Telecomunicaciones y Tecnología de Interconexión.
- **Usuario General:** Usuario general de consulta. Puede generar informes pero no se le permite modificar ningún registro.
- **Consultor:** Especificación del usuario general en donde además de consultar información tiene acceso a los módulo de tablas de análisis, agenda y blog.

A.1.2. Casos de Usos

Los artefactos correspondientes a los casos de uso se han empaquetado de acuerdo al tipo de nodo en el cual se desarrollan. Se incluyen solo los casos que se consideran nucleares y las especificaciones de mayor relevancia e impacto.

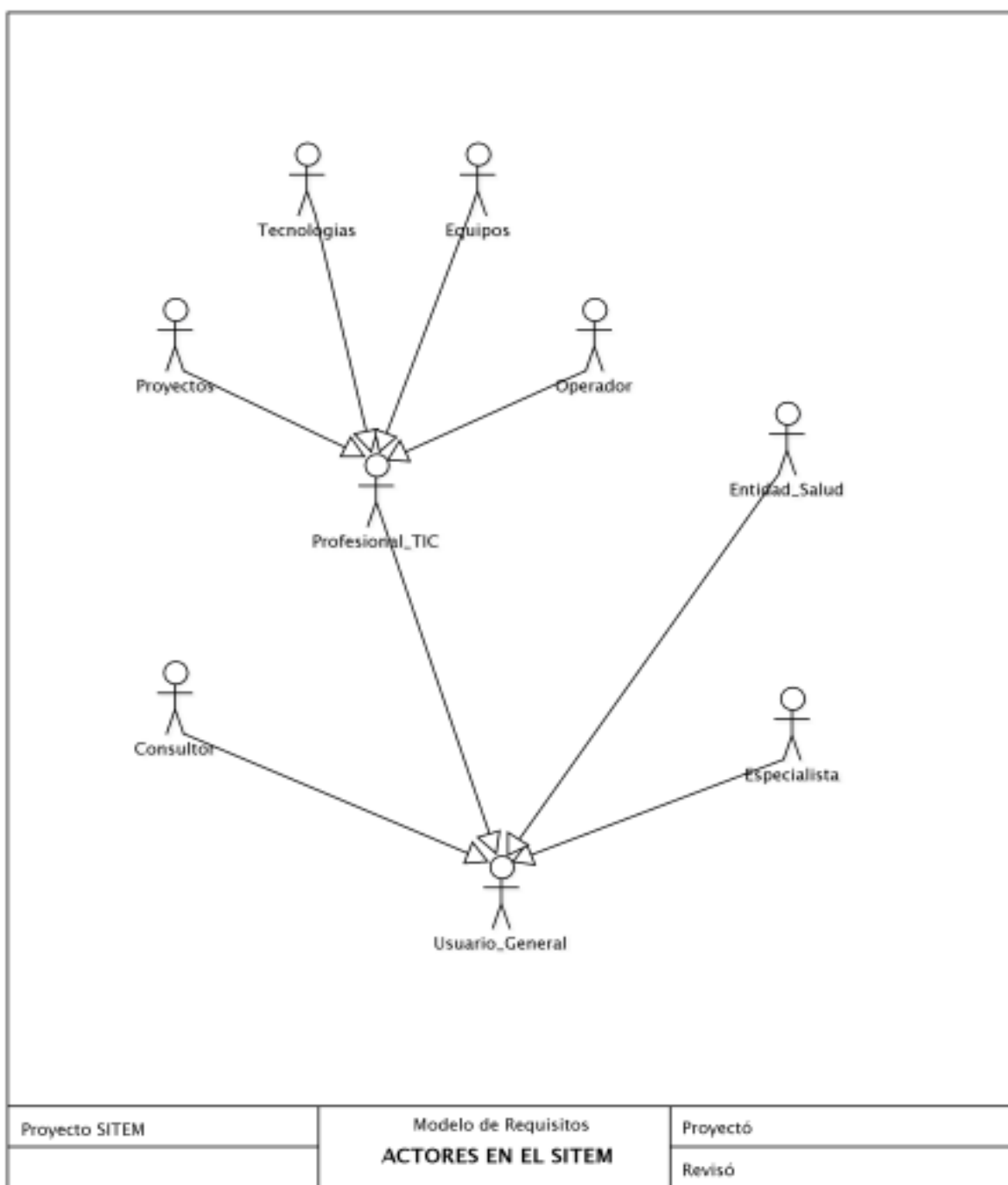


Figura A.1: Conjunto de Actores de OpenSITE;

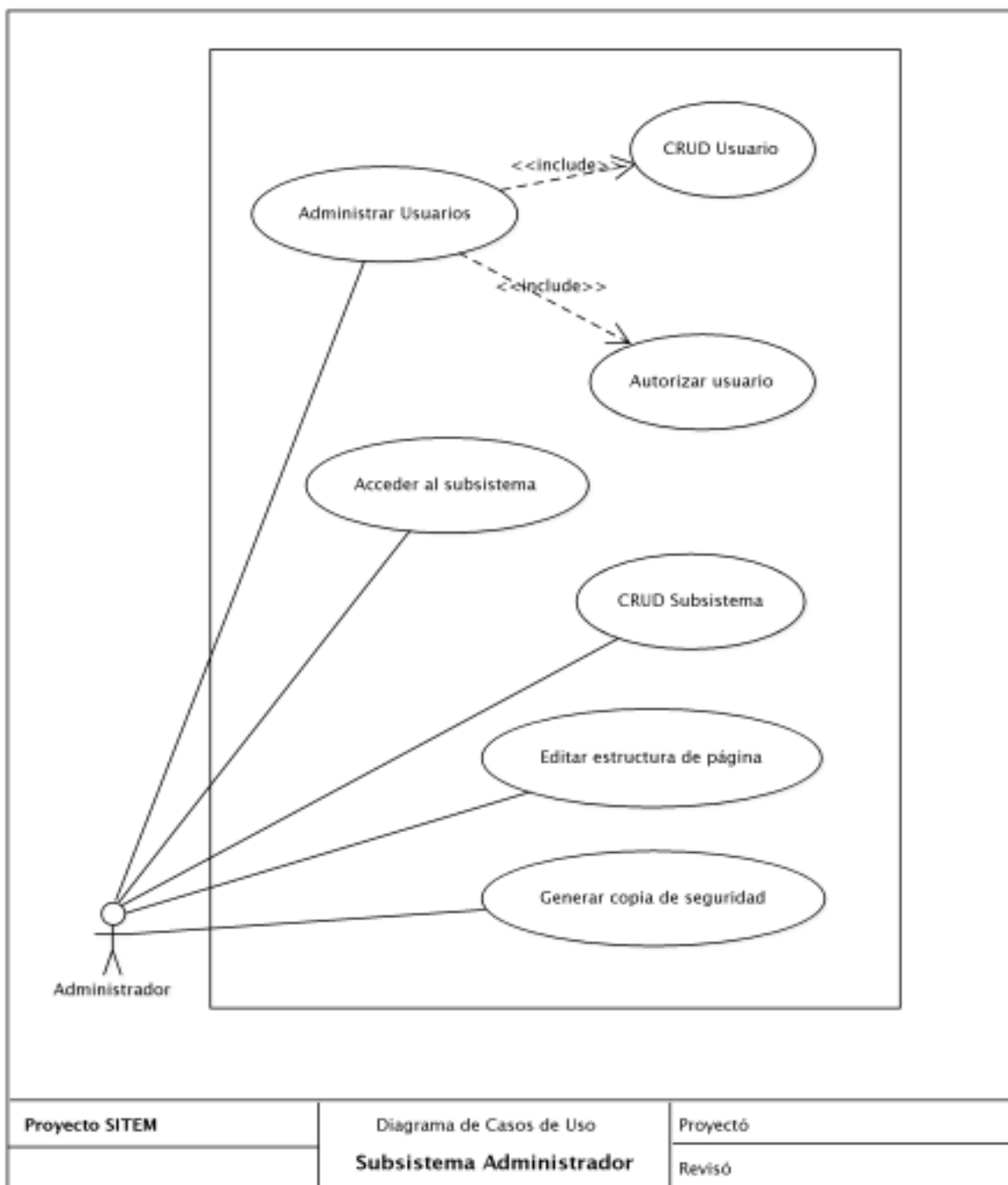


Figura A.2: Casos de uso principales del Actor administrador



Figura A.3: Casos de uso principales del Actor Consultor/Usuario General

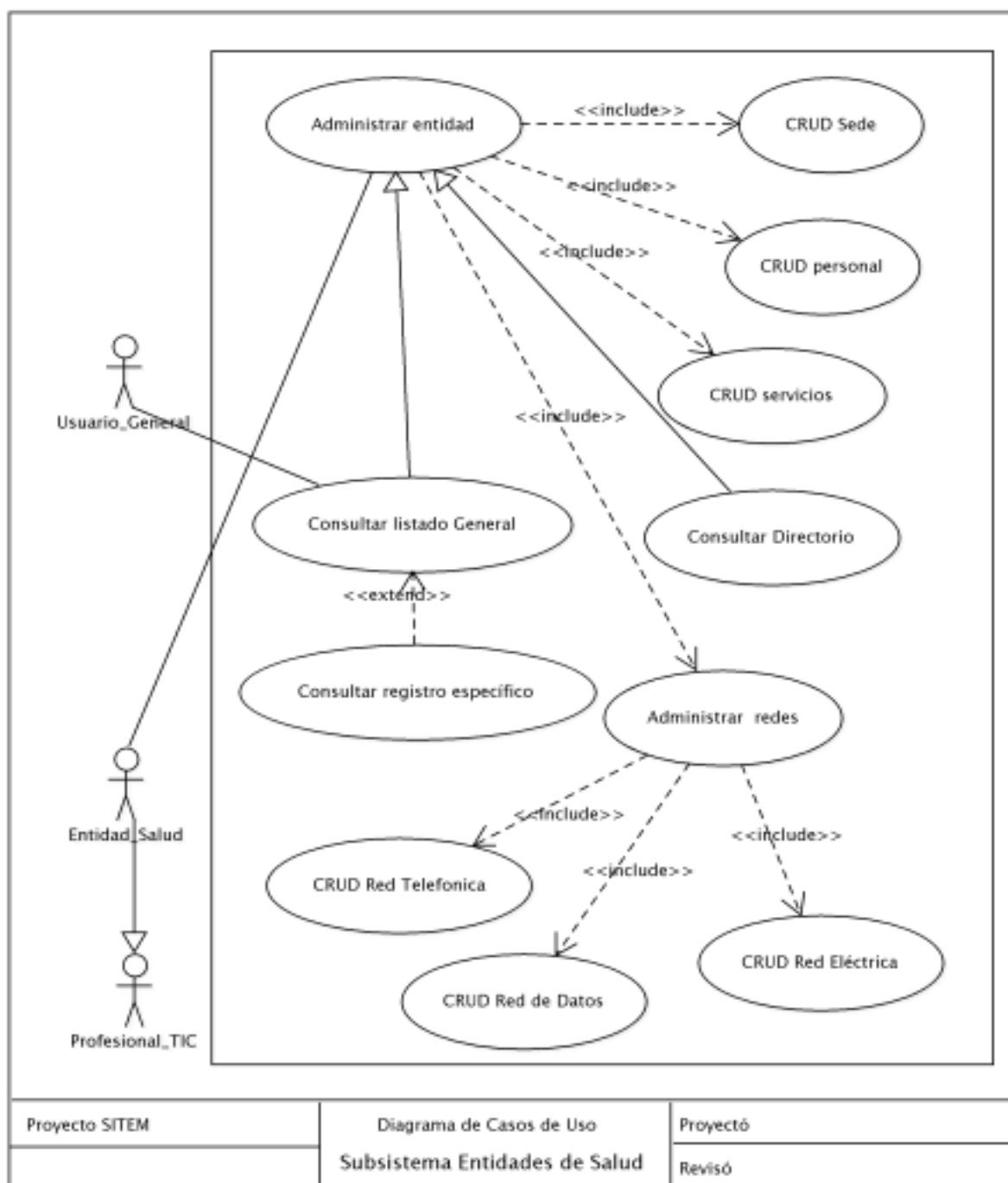


Figura A.4: Casos de uso principales del Actor Entidad de Salud

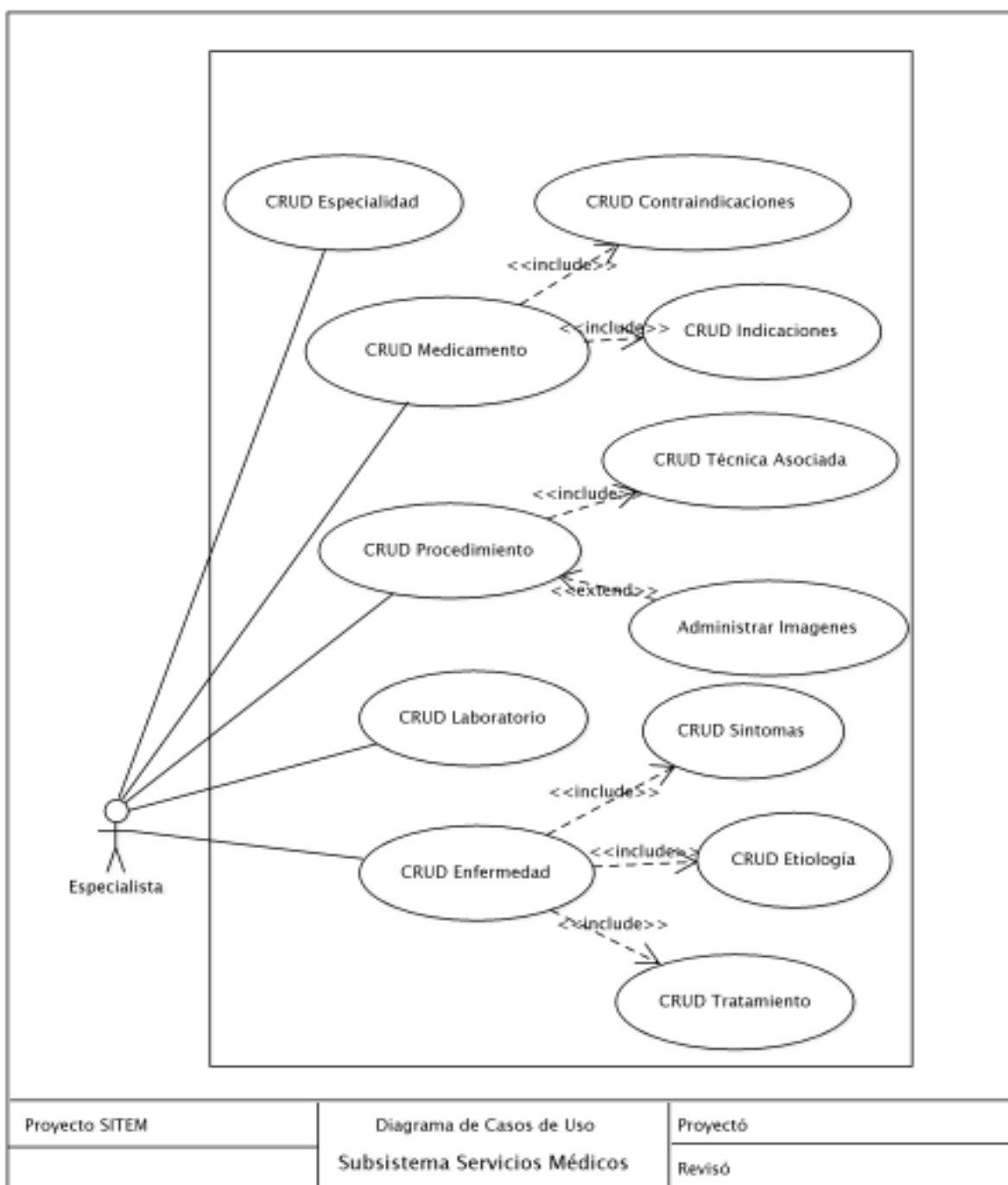


Figura A.5: Casos de uso principales del Actor Servicios Médicos

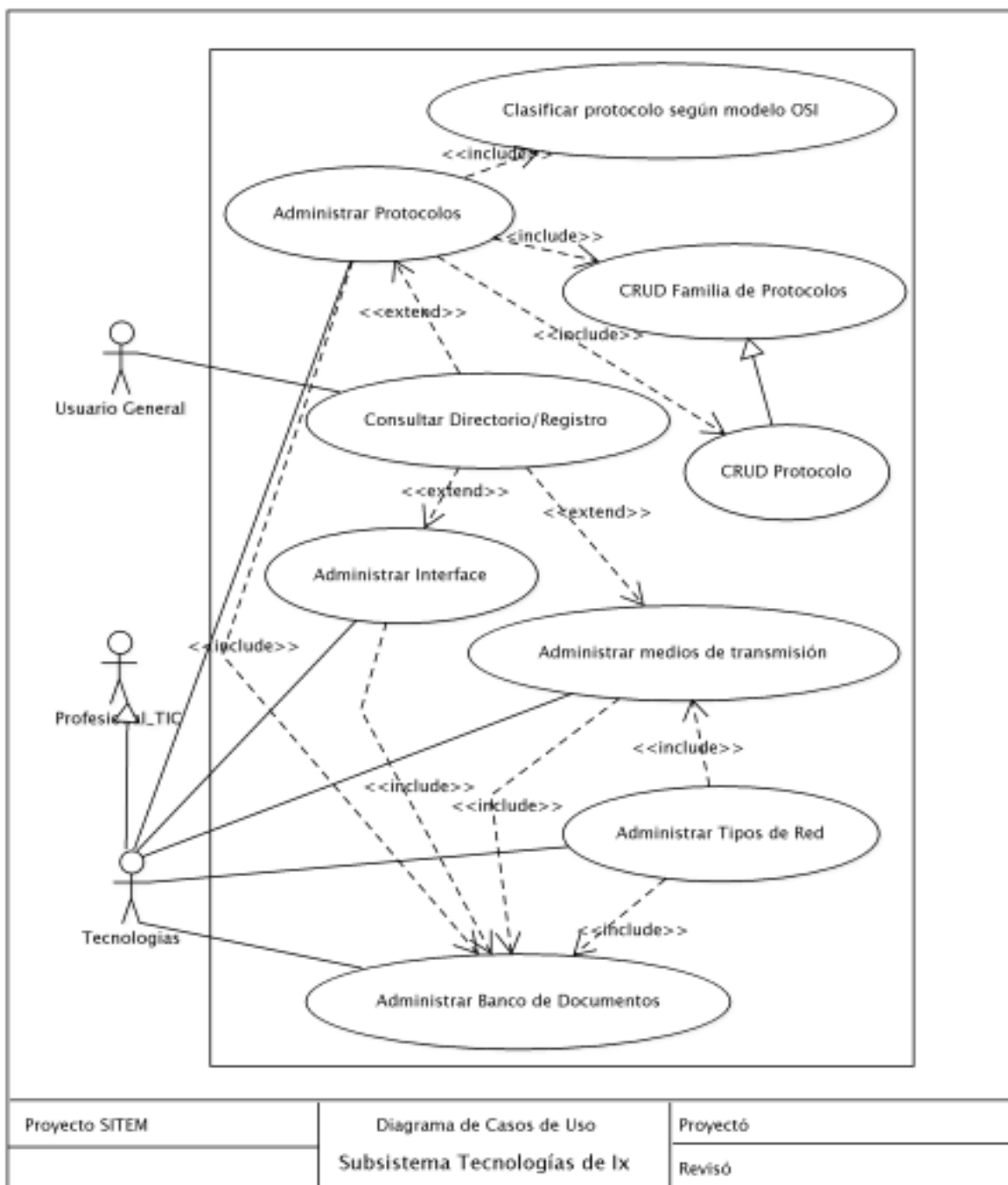


Figura A.6: Casos de uso principales del Actor Tecnologías de Interconexión

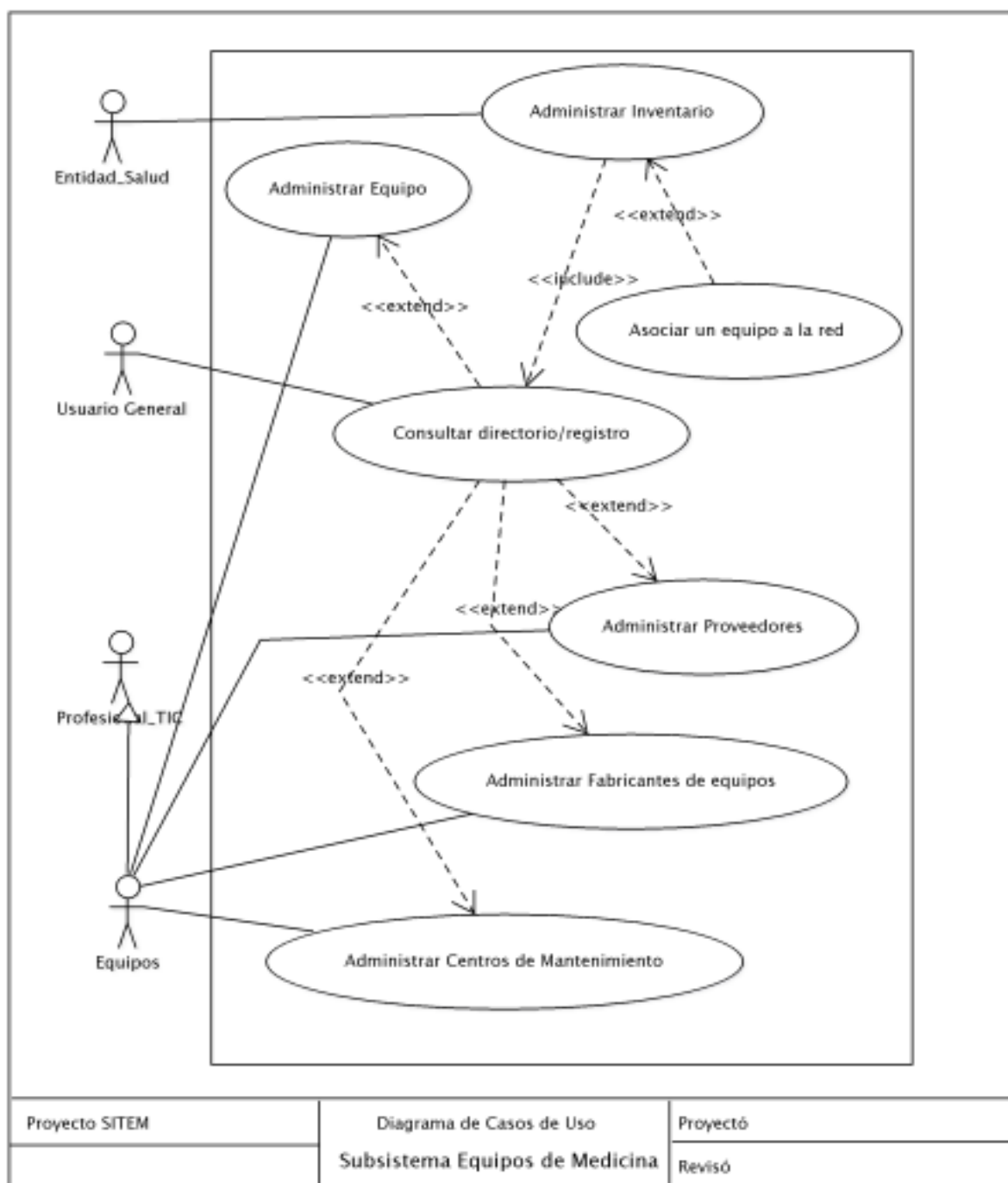


Figura A.7: Casos de uso principales en el Subsistema Equipos de Medicina

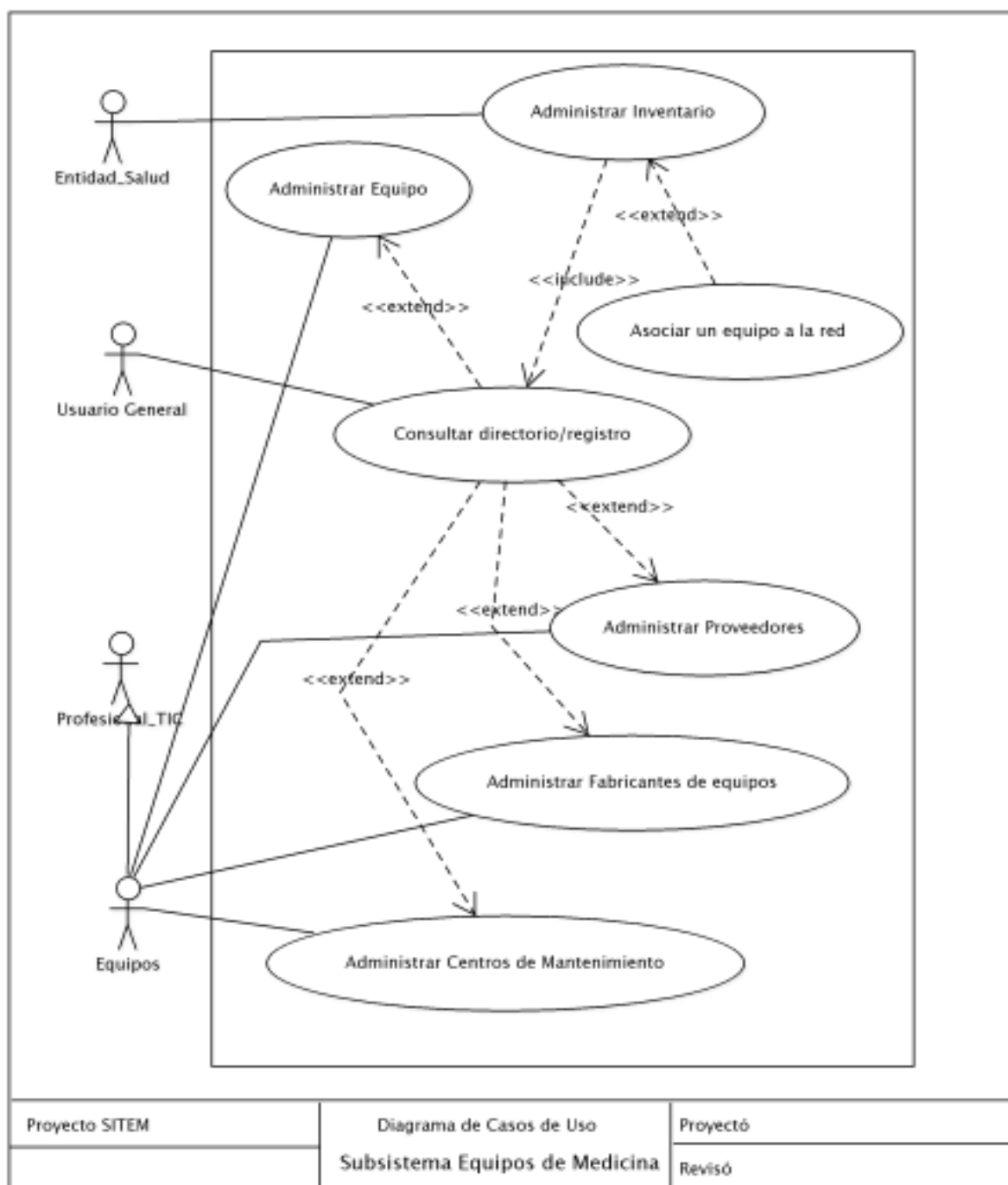


Figura A.8: Casos de uso principales en el Subsistema de Operadores de Servicios de Telecomunicaciones

Anexo B

Modelo de Análisis y Diseño

B.1. Especificaciones de Casos de Usos

Para analizar en detalle los requerimientos del sistema se especifica mediante plantillas aquellos casos de uso que se consideran de importancia para el desarrollo base - conocidos como *Casos de Uso Nucleares*. La tabla B.1 muestra las secciones básicas que contiene un caso de uso. En aquellos casos de uso que una o varias secciones carezcan de contenido relevante se omiten por completo su declaración.

Caso de Uso	
Nombre	Nombre que identifica el caso de Uso usualmente es el mismo utilizado en el diagrama de Casos de Uso
Objetivo	Beneficio que obtiene el actor con la ejecución de este caso de uso.
Código Interno	Código único que identifica al Caso de Uso dentro del repositorio de artefactos.
Actores	Usuarios que intervienen en el caso de uso
Precondiciones	Estado en que debe encontrarse el SITEM antes de ejecutarse el caso de uso.
Flujo Básico	Flujo principal de actividades. Ambiente ideal.
Flujo Alternativo y de error	Actividades que bifurcan el flujo básico. Si existe más de un flujo alternativo este debe colocarse en una nueva fila.
Postcondiciones	Estado en que queda el SITEM después de ejecutado el caso de uso.
Puntos de Extensión	Secuencias de acciones que extienden el flujo del caso de uso.

Tabla B.1: Plantilla Genérica para la Especificación de los Casos de Uso.

Caso de Uso	
Nombre	Registrarse en el SITEM
Objetivo	El actor logra crear una cuenta en el SITEM con un rol específico para poder trabajar en un subsistema dado.
Código Interno	UC-GENERAL-001
Actores	Usuario General
Precondiciones	Ninguna.
Flujo Básico	<ol style="list-style-type: none"> 1. El usuario general selecciona la opción de nuevo usuario desde la página principal del SITEM. 2. El SITEM muestra un formulario con los campos: Nombres Apellidos Correo Electrónico Teléfono Nombre de Usuario Clave Reescriba la clave Acceso Requerido 3. El usuario diligencia uno a uno los campos requeridos y opcionales. 4. El usuario envía los datos al SITEM. 5. El SITEM verifica que los datos tengan los formatos esperados. 6. El SITEM ingresa el registro a la base de datos colocando el campo de estado en 1 - registrado sin autorización. 7. El SITEM redirecciona a la página de registro exitoso. 8. El usuario acepta el mensaje.
Postcondiciones	Se agregó un registro en la base de datos con el campo de estado en 1.
Casos de uso relacionados	Seleccionar Rol en el SITEM

Tabla B.2: Caso de Uso Registrarse en el SITEM

Caso de Uso	
Nombre	Administrar autorizaciones de Usuario
Objetivo	Proveer un mecanismo eficaz para que el administrador general del SITEM pueda gestionar el estado de autorización de los usuarios en los diferentes subsistemas.
Código Interno	UC-GENERAL-002
Actores	Administrador
Precondiciones	Debe existir por lo menos un usuario registrado en el sistema diferente al administrador. El administrador se encuentra correctamente autenticado y autorizado en el subsistema de administrador.
Flujo Básico	<ol style="list-style-type: none"> 1. El Administrador selecciona la opción Usuarios desde el menú principal del subsistema Administrador. 2. El SITEM muestra un listado con los datos básicos de diferentes usuarios registrados: Nombres Apellidos Correo Electrónico Acceso Requerido 3. <i>Punto de Extensión 1.</i> 4. <i>Punto de Extensión 2.</i> 5. El administrador verifica que los datos del usuario son reales. 6. El administrador seleccionada la casilla de verificación y acepta el trámite. 7. El SITEM procesa el formulario colocando el estado del usuario en valor 2 - Registrado y Autorizado. 8. El SITEM indica con un mensaje el éxito en la operación de autorización. 9. El SITEM envía un mensaje de texto al usuario indicando que ha sido autorizado. 8. El Administrador acepta el mensaje de éxito.
Postcondiciones	Se cambia el valor en el campo estado del registro correspondiente al usuario.
Puntos de Extensión	1:direccion="avanzar" o direccion="retroceder" extend Navegar en listado. 2:Opción="buscar" extend Busqueda condicional de registro.

Tabla B.3: Caso de Uso Autorizar un usuario para acceder al SITEM

Caso de Uso	
Nombre	Generar Copia de Seguridad
Objetivo	Generar una copia de seguridad de los datos contenidos en la base de datos asociada al SITEM.
Código Interno	UC-GENERAL-003
Actores	Administrador
Precondiciones	El administrador se encuentra correctamente autenticado y autorizado en el subsistema de administrador.
Flujo Básico	<ol style="list-style-type: none"> 1. El Administrador selecciona la opción Copia de Seguridad el menú principal del subsistema Administrador. 2. El SITEM muestra un listado con las tablas opcionales para la copia de seguridad. 3. El administrador selecciona la casilla de verificación de las tablas que desea sean copiadas. 4. <i>Punto de Extensión 1.</i> 5. El administrador acepta la selección. 6. El SITEM muestra un listado con las tablas que serán copiadas y un formulario con los campos: Nombre del Archivo. Ruta de Descarga. 7. El usuario diligencia uno a uno los campos requeridos. 8. El usuario envía los datos al SITEM. 9. El SITEM realiza una copia de los registros escribiéndolos uno a uno en los archivos de destino. 10. El SITEM redirecciona a la pagina de operación exitosa. 11. El usuario acepta el mensaje.
Postcondiciones	Se crea n archivos en la carpeta de destino con el contenido de las n tablas seleccionadas para copia de seguridad.
Puntos de Extensión	1:opcion="todo" extend Seleccionar todos los cuadros.

Tabla B.4: Caso de Uso realizar Copia de Seguridad

Caso de Uso	
Nombre	Elaborar Tablas de Análisis
Objetivo	Obtener un repositorio de análisis de algún componente del modelo jerárquico de seguimiento a proyectos.
Código Interno	UC-GENERAL-004
Actores	Consultor
Precondiciones	Existe en el SITEM un modelo jerárquico de seguimiento a proyectos.
Flujo Básico	<ol style="list-style-type: none"> 1. El Consultor selecciona la opción Seguimiento desde el menú principal del subsistema Consultor. 2. El SITEM muestra el modelo de seguimiento a proyectos con sus componentes de primer nivel. 3. <i>Punto de Extensión 1.</i> 4. El Consultor selecciona la opción de Analizar para un componente. 5. El SITEM muestra un formulario con los campos: Valoración Cualitativa. Valoración Cuantitativa. Juicio. Diagnóstico. Fortalezas. Oportunidades. Debilidades. Amenazas. Directrices de Mejoramiento. Directrices de Acción. 6. El usuario diligencia uno a uno los campos requeridos y opcionales. 8. El usuario envía los datos al SITEM. 5. El SITEM verifica que los datos tengan los formatos esperados. 6. El SITEM ingresa el registro a la base de datos. 7. El SITEM redirecciona a la página de registro exitoso. 8. El usuario acepta el mensaje.
Postcondiciones	Existe un registro de análisis asociado a un componente y un consultor.
Puntos de Extensión	1:opcion="mas" extend Mostrar Componentes de Nivel Inferior

Tabla B.5: Caso de Uso Elaborar Tablas de Análisis

Caso de Uso	
Nombre	Ingresar una Entidad
Objetivo	Registrar una nueva entidad de Salud en el SITEM.
Código Interno	UC-GENERAL-005
Actores	Entidad Salud
Precondiciones	El usuario Entidad Salud se encuentra autorizado y autenticado en el subsistema Entidades de Salud.
Flujo Básico	<ol style="list-style-type: none"> Entidad Salud selecciona la opción Nueva Entidad desde el menú principal del subsistema Entidades de Salud. El SITEM muestra un formulario con los campos: Nombre de la Entidad. Nombre Corto. Logosímbolo. NIT. Fecha de Fundación. Dirección Principal. Teléfono Principal (PBX). Misión. Visión. Descripción. Comentarios. El usuario diligencia uno a uno los campos requeridos y opcionales. El usuario envía los datos al SITEM. El SITEM verifica que los datos tengan los formatos esperados. El SITEM comprueba que no exista otra Entidad registrada con el mismo NIT. El SITEM ingresa el registro a la base de datos. El SITEM redirecciona a la página de registro exitoso. El usuario acepta el mensaje.
Flujo Alternativo	<ol style="list-style-type: none"> Los datos no tienen el formato adecuado. El SITEM informa el error. <i>Punto de Extensión 1.</i>
Flujo Alternativo	<ol style="list-style-type: none"> Existe una entidad registrada con el mismo NIT. El SITEM informa el error. <i>Punto de Extensión 1.</i>
Postcondiciones	Existe un registro de una entidad de salud.
Puntos de Extensión	1:opcion="corregir" extend Mostrar Formulario Corrección.

Tabla B.6: Caso de Uso Ingresar una Nueva Entidad de Salud.

Caso de Uso	
Nombre	Consultar información básica de una entidad de Salud.
Objetivo	Obtener en pantalla los datos básicos de una entidad de salud.
Código Interno	UC-GENERAL-006
Actores	Entidad Salud, usuario general
Precondiciones	El usuario se encuentra autorizado y autenticado en el subsistema Entidades de Salud.
Flujo Básico	<ol style="list-style-type: none"> 1. Entidad Salud selecciona la opción Entidades desde el menú principal del subsistema Entidades de Salud. 2. El SITEM muestra un listado de 25 entidades ordenadas alfabéticamente por nombre. 3. <i>Punto de Extensión 1.</i> 4. El usuario selecciona una entidad de salud desde el listado. 5. El SITEM realiza una búsqueda con el id de la entidad. 6. El SITEM muestra en pantalla el menú secundario para solicitar edición y los datos de la entidad: Nombre de la Entidad. Nombre Corto. Logosímbolo. NIT. Fecha de Fundación. Dirección Principal. Teléfono Principal (PBX). Misión. Visión. Descripción. 7. El usuario acepta los datos.
Puntos de Extensión	1:direccion="avanzar" o direccion="retroceder" extend Navegar en listado.

Tabla B.7: Caso de Uso Consultar información básica de una entidad de Salud.

Caso de Uso	
Nombre	Editar un registro en el SITEM.
Objetivo	Editar la información que se encuentra en un registro del SITEM. La actualización puede involucrar más de una entidad en la capa de persistencia
Código Interno	UC-GENERAL-007
Actores	Profesional TIC, entidad salud, administrador, usuario general
Precondiciones	El usuario se encuentra autorizado y autenticado en el subsistema.
Flujo Básico	<ol style="list-style-type: none"> 1. El usuario selecciona la opción Editar Registro desde el menú secundario del subsistema. 2. El SITEM muestra un formulario relleno con los datos del registro correspondiente. 3. El usuario edita los valores dentro de los controles del formulario. 4. El usuario envía el formulario al SITEM. 5. El SITEM verifica que los datos editados no violen alguna regla de integridad referencial. 6. El SITEM actualiza los registros en la capa de persistencia. 7. El SITEM muestra al usuario un mensaje de éxito. 8. El usuario acepta el mensaje.
Flujo Alternativo	<ol style="list-style-type: none"> 5.A. Existe un error de integridad referencial. 7.A. El SITEM informa el error. 8.A. <i>Punto de Extensión 1.</i>
Puntos de Extensión	1:opcion="corregir" extend Mostrar Formulario Corrección.
Postcondiciones	Se actualizan los registros correspondientes.

Tabla B.8: Caso de Uso Editar un registro en el SITEM

Caso de Uso	
Nombre	Asociar un protocolo de comunicaciones al modelo OSI.
Objetivo	Asociar un protocolo de comunicaciones al modelo de referencia OSI.
Código Interno	UC-GENERAL-008
Actores	Profesional TIC, Tecnologías.
Precondiciones	El usuario se encuentra autorizado y autenticado en el subsistema. Existe registrado por lo menos un protocolo de comunicaciones en el SITEM.
Flujo Básico	<ol style="list-style-type: none"> 1. El usuario selecciona la opción Más Información desde el menú secundario del subsistema. 2. El SITEM muestra la información del protocolo asociada por áreas temáticas. 3. El usuario selecciona la opción Clasificar OSI. 4. El SITEM muestra el grafico de siete capgráficomodelo OSI. 5. El usuario selecciona una o varias capas del modelo. 6. El SITEM asocia el id del protocolo a cada una de las capas del modelo OSI seleccionadas por el usuario. 7. El SITEM muestra el modelo OSI extendido con los demás protocolos registrados en cada capa. 8. El usuario acepta el registro.
Flujo Alternativo	<ol style="list-style-type: none"> 5.A. El usuario no selecciona ninguna capa. 7.A. El SITEM regresa al punto 2 del caso de uso.
Postcondiciones	El conjunto de protocolos asociados a una capa del modelo OSI se incrementa.

Tabla B.9: Caso de Uso Asociar un protocolo de comunicaciones al modelo OSI.

Caso de Uso	
Nombre	Borrar un registro del SITEM.
Objetivo	Eliminar un registro en algún subsistema del SITEM garantizando que solo el experto en información lo realice y se mantenga la integridad referencial en los datos.
Código Interno	UC-GENERAL-009
Actores	Profesional TIC, Entidad Salud, Consultor, Administrador.
Precondiciones	El usuario se encuentra autorizado y autenticado en el subsistema. Existe un registro en el SITEM. El usuario a creado el registro y este no tiene información asociada.
Flujo Básico	<ol style="list-style-type: none"> 1. El usuario selecciona la opción Eliminar Registro desde el menú secundario del subsistema. 2. El SITEM muestra un mensaje de confirmación de eliminación con los datos básicos del registro. 3. El usuario selecciona la opción de Confirmar Borrado. 4. El SITEM elimina el registro cumpliendo las restricciones de claves foráneas. 5. El sistema muestra un mensaje indicando que el registro se ha borrado del sistema. 6. El usuario acepta el mensaje. 7. El SITEM redirige a la página en donde se encontraba el usuario antes del proceso de borrado.
Flujo Alternativo	<ol style="list-style-type: none"> 3.A. El usuario no acepta borrar el registro. 4.A. Continúa en el punto 7 del flujo principal.
Postcondiciones	El registro se borra del sistema.

Tabla B.10: Caso de Uso para Borrar un registro en el SITEM.

Caso de Uso	
Nombre	Acceder a una página del SITEM.
Objetivo	Ingresar a una página específica dentro del sistema.
Código Interno	UC-GENERAL-010
Actores	Profesional TIC, Entidad Salud, Consultor, Administrador.
Precondiciones	El usuario está autorizado para acceder a la página. La página se encuentra registrada en el SITEM. La página tiene uno o más bloques asociados.
Flujo Básico	<ol style="list-style-type: none"> 1. El usuario elige un enlace a una página dentro del SITEM. 2. El SITEM verifica que el usuario tiene una sesión válida 3. El SITEM rescata los valores de la página desde la base de datos. 4. El SITEM verifica que el usuario tenga los privilegios necesarios para ingresar a la página. 5. El SITEM consulta la estructura de la página desde la base de datos. 6. El SITEM envía secuencialmente el código HTML correspondiente a cada una de los bloques que constituyen la página. 7. El SITEM registra el acceso del usuario en la base de datos. 8. El SITEM actualiza la información de sesión.
Flujo Alternativo	<ol style="list-style-type: none"> 4.A. El usuario no tiene los privilegios para ver la página. 5.A. El SITEM registra un intento de ingreso ilegal. 6.A. El SITEM muestra un mensaje informando el error.
Postcondiciones	El usuario ingresa a una página dentro del SITEM.

Tabla B.11: Caso de Uso para Acceder a una página del SITEM.

Anexo C

Modelo de Implementación

El modelo de implementación del SITEM esta compuesto básicamente por los siguientes artefactos que en su conjunto representan lo que comunmente se denomina el aplicativo:

- **Bloques:** Unidad Básica de funcionalidad. Se pueden pensar en ellos como “instancias” persistentes de clases abstractas. En especial se tienen tres clases:
 - Administración: Con atributos y métodos para mostrar directorios de datos.
 - Menú: Con métodos especializados en la administración de enlaces dentro del SITEM.
 - Registro: Para la realización de casos CRUD.
- **Clases:** Descriptores para varios tipos de objetos entre las cuales se tiene:
 - DBMS: Interacción con la bases de datos.
 - Página: Describe objetos que realizan la construcción en tiempo de ejecución de las páginas.
 - Encriptar: Descriptor para objetos que se encargan de codificar y decodificar los datos en el SITEM. El conjunto de operaciones debe ser manipulado en cada implementación del SITEM para garantizar un alto nivel de seguridad.
 - Autenticacion: Con descripción de atributos y operaciones que controlan las rutinas de AAA en el SITEM.
 - Config: Clasificador de objetos que manipulan las variables de configuración globales.
 - Html: Descriptor de controles de formularios en el SITEM.
 - Sesión: Operaciones y atributos para el control de sesiones en el SITEM luego del proceso de AAA.
 - Mensaje: Para objetos que administran mensajes de interacción con los actores.

- **Sql:** Clase para describir objetos especializados en gestionar archivos con extensión SQL.
- **Navegacion:** Con operaciones específicas para el control de desplazamiento entre conjuntos de registros.
- **Función:** Métodos JavaScript para la validación y control de navegación en el lado del cliente.
- **Estilo:** Para el control de la capa de Interfaz Gráfica.

C.1. Código Fuente del SITEM

La siguiente porción de código fuente representa el formato general que se encuentra en el SITEM. Por patrón general se recomienda a todos los grupos que participan en el desarrollo que mantengan un esquema de codificación claro y documentación *in situ* suficiente para aclarar secciones.

C.1.1. Clase Página

La clase Página tiene las operaciones:

- **pagina:** Constructor.
- **especificar pagina:** Inicializar variables privadas.
- **procesar pagina:** Controlar redireccionamientos.
- **ancho seccion:** Implementa control de secciones colapsadas.
- **armar seccion:** Para seleccionar los bloques que contiene cada página.

Algunos métodos instancian la clase DBMS y HTML. Estas por lo tanto deben ser visibles.

```
class_pagina
{

//Metodo_constructor
function_pagina(id_pagina,configuracion)
{
//Declaracion_de_variable_parta_controlar_accesos_indebidos
```

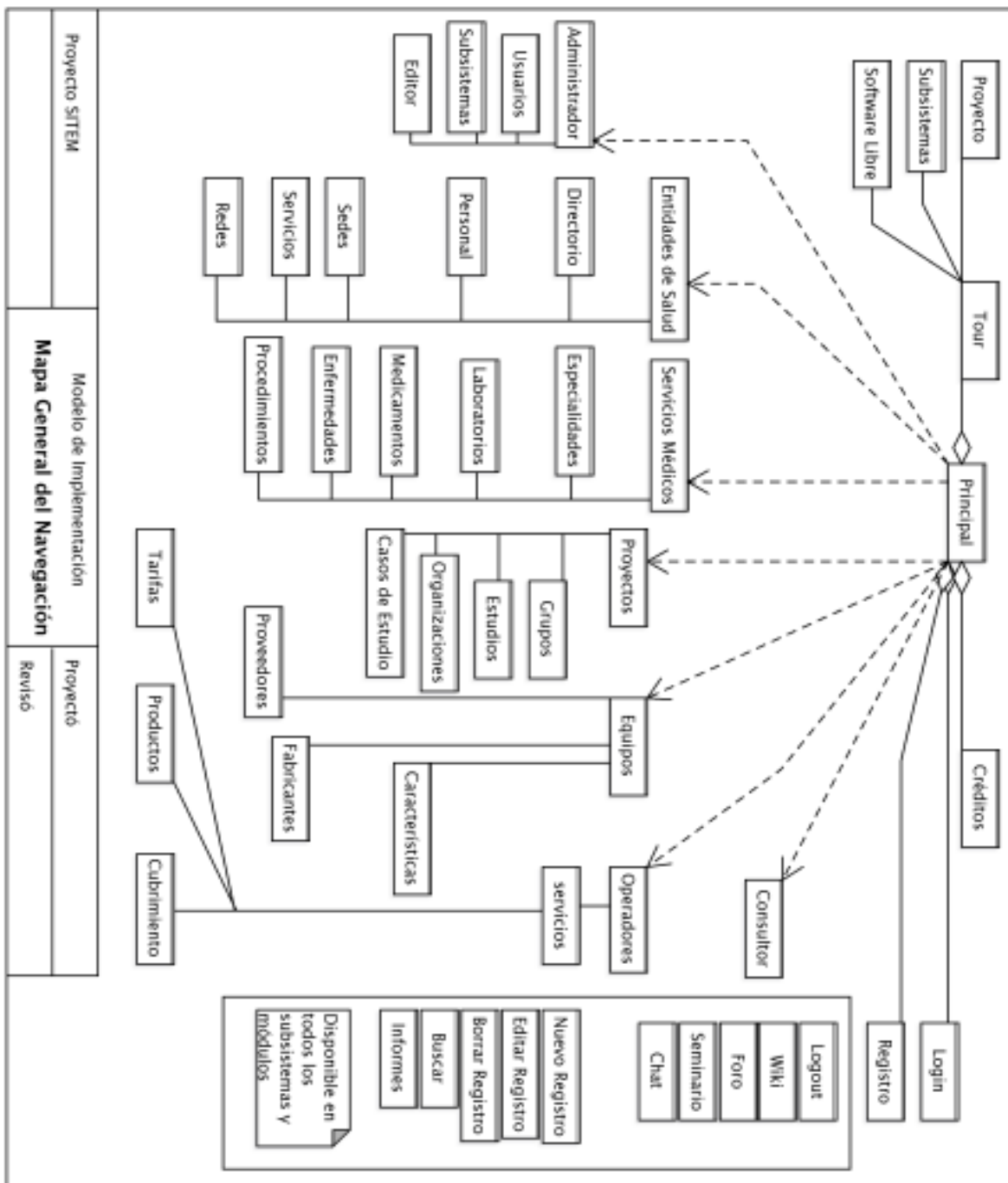


Figura C.1: Modelo General de Navegación

```

GLOBALS["autorizado"]=TRUE;
this->especificar_pagina(id_pagina);

if(!isset($_POST['registro_compuesta']))
{
if(!isset($_REQUEST['action']))
{

this->mostrar_pagina(configuracion);
}
else
{
//echo 'Procesamiento de la pagina';
this->procesar_pagina(configuracion);
}
}
else
{
this->mostrar_pagina(configuracion);
}
}
//Fin del metodo constructor

//Metodo especificar_pagina
function especificar_pagina(id_pagina)
{

this->id_pagina=id_pagina;

}
//Fin del metodo especificar_pagina
.
.
.
}

```

Anexo D

Modelo de Datos

La capa de persistencia en el SITEM esta soportada en una estructura de base de datos de modelo Relacional. La forma normal básica - 3NF, se garantiza mientras que otras formas normales pueden ser omitidas en casos puntuales donde se demuestre, por pruebas de desempeño, que no seguirlas redundaría en una mejora significativa de la velocidad en el acceso a los datos sin detrimento de la calidad e integridad de los mismos. A continuación se listan las formas normales tenidas en cuenta como patrón de diseño en el modelo de datos del SITEM:

- Primera Forma Normal: Cada renglón-columna contiene valores atómicos.
- Segunda Forma Normal: 1NF y todo campo que no sea clave primaria depende de los campos clave.
- Tercera Forma Normal: 2NF y no hay dependencias transitivas.
- Forma Normal de Boyce-Codd: Todos los determinantes de la tabla son clave candidata.
- Cuarta Forma Normal: Una fila no debe contener dos o más campos multi-valorados.
- Quinta Forma Normal: Una tabla puede almacenar atributos dependientes a la clave sólo por unión.

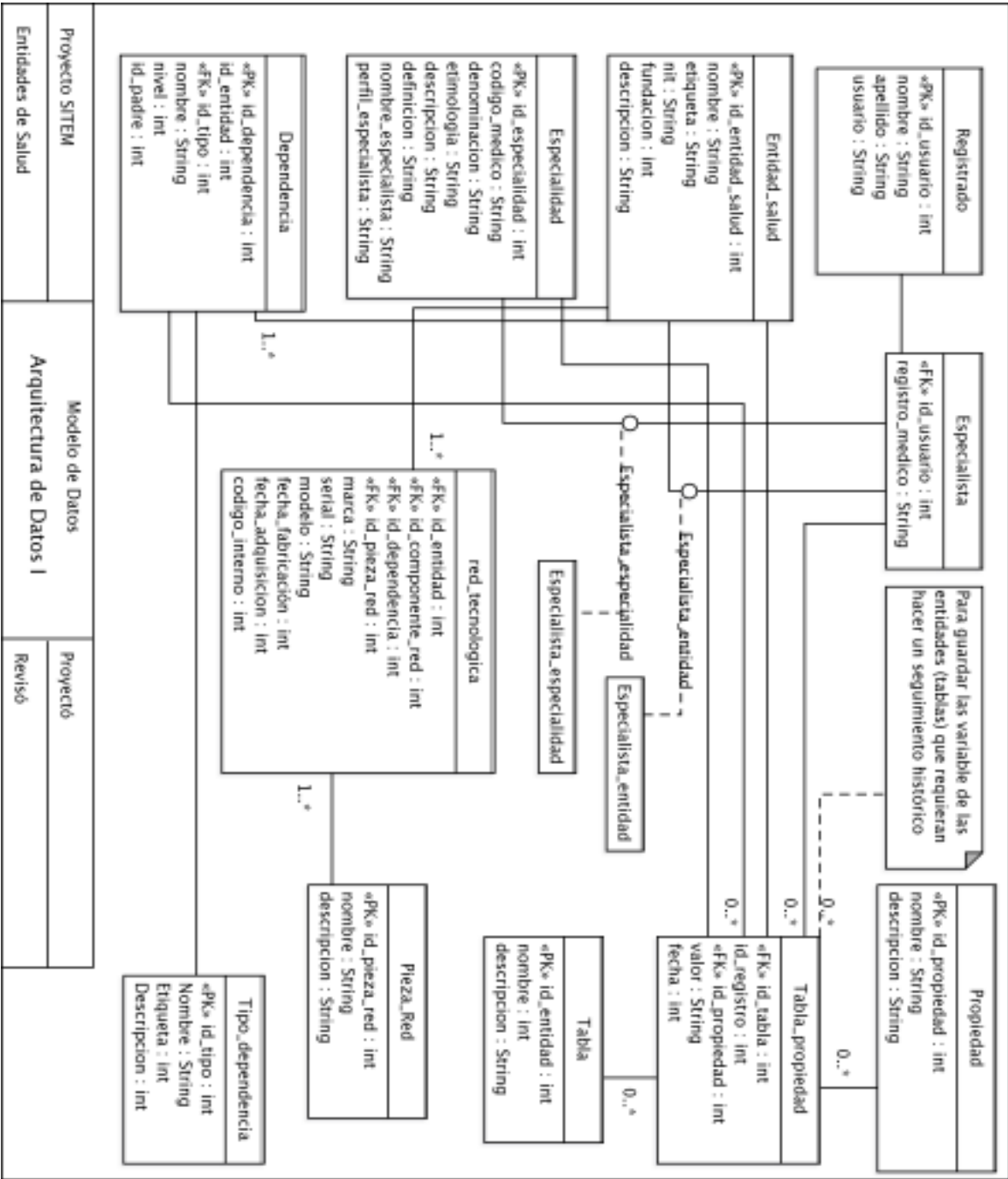


Figura D.1: Arquitectura de datos Subsistema Entidades de Salud

Anexo E

Modelo de Despliegue

El SITEM se despliega sobre un servidor que tenga instalado un programa que acepte peticiones usando el protocolo HTTP - Servidor Web, tenga soporte para el interprete de PHP y un servidor de bases de datos MySQL o PostgreSQL. La figura E.1 muestra el diagrama de componentes correspondiente a la arquitectura propuesta.

La arquitectura mostrada se reproduce en tres instancias del servidor web, cada una correspondiendo a los niveles de:

- **Desarrollo:** Con las versiones más recientes del sistema que se consideran en versión alfa o inestables. Dentro de la codificación son aquellas que muestran incrementos en el cuarto segmento: AA.XX.YY.**ZZ**
- **Prueba:** Versiones que tienen desarrollos estables pero que aún no han pasado la etapa de revisión de casos de prueba. En la codificación son aquellas que muestran un incremento en el tercer segmento: AA.XX.**YY**.ZZ
- **Producción:** Versiones que han sido probadas y no tienen errores detectados. De acuerdo a la naturaleza de la funcionalidad alcanzada estas versiones representan un incremento en los segmentos primero o segundo dentro de la codificación.

Si es el primer segmento el que incrementa se alcanza un ciclo de desarrollo y la versión es bautizada con un nombre específico. Para el SITEM se usan los nombre de los Dioses maya que participaron en alguno de los tres intentos por crear la humanidad.

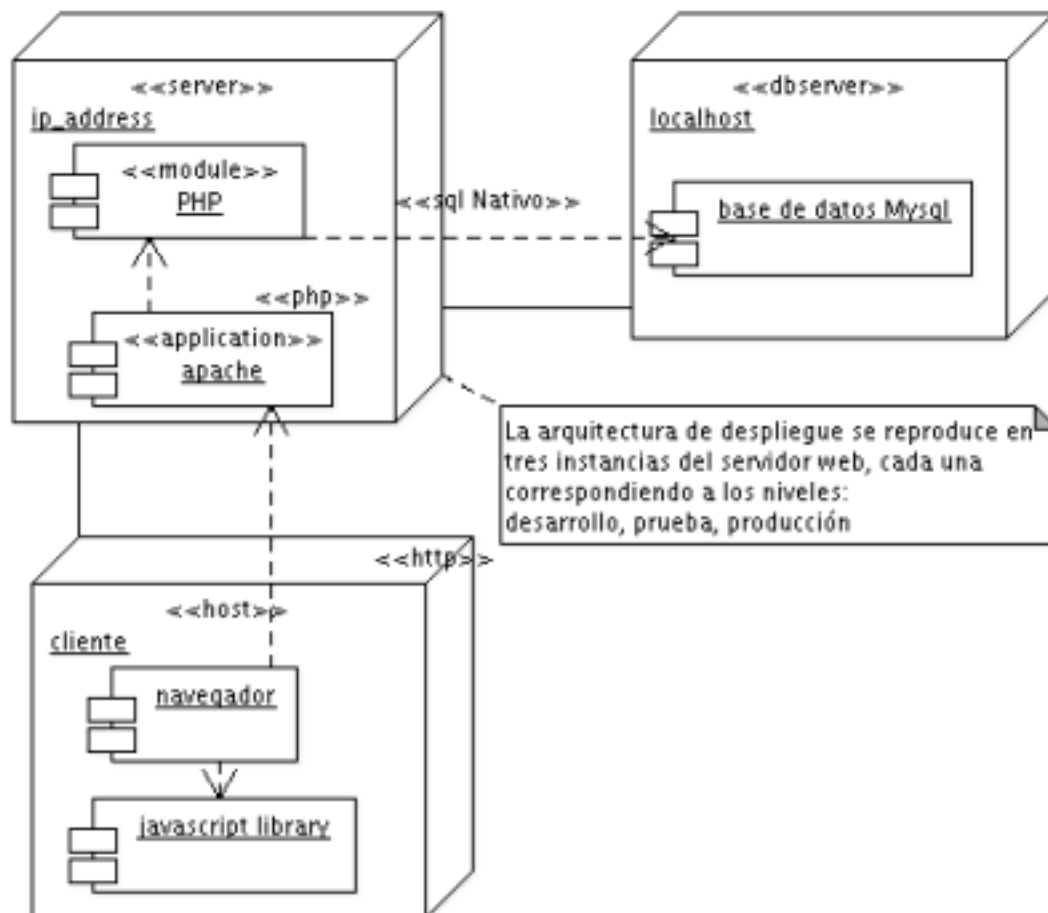


Figura E.1: Modelo de Despliegue

Anexo F

Declaración de Riesgos

El proyecto está expuesto a una serie de contingencias que de no ser correctamente manejadas atentan la consecución de sus metas. Estos riesgos deben estar de la forma más óptima detectados, evaluados y medido su impacto. Cada vez que se avanza en el ciclo de desarrollo unos se vuelven más importantes que otros. A continuación se declaran los riesgos más importantes.

- **Baja disponibilidad de Tiempo.**

La falta de tiempo por parte de los integrantes del proyecto debido al trabajo en paralelo de otros proyectos es un riesgo inminente que de seguro puede desencadenar en la no consecución de los objetivos. El reclutamiento de integrantes cuyo objetivo primario es el lleno de un requisito para grado pone en riesgo la continuidad en el desarrollo de algún hilo.

Métodos para la reducción del riesgo: Es aquí cuando los métodos ágiles son de gran ayuda. El grupo de desarrollo fomenta la retroalimentación continua entre los diferentes hilos creando una especie de competencia sana en donde los componentes y prácticas más eficientes son detectadas y propagadas en las sesiones de capacitación. A partir de la versión 3.0, actualmente en desarrollo, los grupos adscritos no se les imponen un objetivo específico sino que las personas interesadas simplemente deciden que tarea es más relevante y su participación solo se tiene en cuenta a través de la producción - posicionamiento basado en el esfuerzo.

- **Pérdida de tiempo debido a procesos administrativos.**

Las cargas administrativas inherentes a los procesos del proyecto consumen gran cantidad de tiempo y en ciertas circunstancias puede tomar más del esperado obstruyendo el desarrollo de las actividades en el proyecto. La dependencia de algunas actividades al formalismo burocrático desenfoca al grupo de trabajo.

Métodos para la reducción del riesgo: El trabajo se realiza en varias áreas y disciplinas de manera paralela brindando siempre la posibilidad de redistribuir recursos si se requiere esperar a que algún trámite administrativo se concrete. El jefe del proyecto se encarga de

realizar todos los trámites y generación de documentos necesarios con el fin de agilizar los procesos administrativos.

- **Falta de recursos para trabajo.**

El mayor riesgo para la elaboración del proyecto y la consecución de los objetivos del mismo se encuentra en el no disponer de recursos para dichas tareas. Principalmente un equipo de computo capaz de soportar sistemas operativos Linux así como de un espacio físico de trabajo dónde se encuentre dicho equipo y en dónde los integrantes del proyecto puedan realizar sus actividades y centralizar información.

Métodos para la reducción del riesgo: Cada uno de los integrantes del proyecto está en capacidad de realizar las tareas que le corresponden por su cuenta fomentando el teletrabajo. El jefe del proyecto se encargo de realizar la centralización del trabajo y la información por medio del préstamo al proyecto de recursos propios. En fases posteriores el grupo de investigación recibe apoyo del Centro de Investigaciones y Desarrollo Científico de la Universidad Distrital con lo que se adquiere un servidor con acceso público a Internet.

- **Falta de información y demora en la consecución de la misma.**

El proyecto requiere y se basa en cierta información proveniente de diversas fuentes como publicaciones, manuales y estudios realizados dentro y fuera del grupo GITEM; la demora o imposibilidad de consecución de información calificada se convierte en un riesgo para el desarrollo normal del proyecto.

Métodos para la reducción de riesgo: Se realiza desde el comienzo del proyecto un listado de información necesario e indispensable y se inicia la búsqueda y consecución de la misma desde el inicio. El jefe de proyecto realiza los procedimientos requeridos con el fin de contar con los resultados de estudios previos en el interior del GITEM. Un cúmulo documental se mantiene disponible en todo momento para la comunidad de desarrollo, las jornadas de capacitación y el despliegue de presentaciones focalizadas es también de gran ayuda.

- **Falla en la dirección del proyecto.**

Existe la posibilidad de que el director del proyecto o de alguno de sus ayudantes cometa algún error o falla atentando de ésta manera con el proceso natural de desarrollo.

Métodos para la reducción de riesgo: Se realiza un seguimiento continuo a todas las tareas asociadas a la dirección del proyecto. El concepto y objetivos del SITEM plasmados en su documento de Visión se han convertido en un artefacto de consulta continua y recurrente.

- **Valoración nula o errónea de los resultados.**

Al terminar una iteración dentro del proceso de desarrollo es posible que no se obtengan los resultados esperados, el exceso de planificación y la rigidez frente al cambio puede generar que se subvaloren los resultados obtenidos y, con la excusa de no cambiar un documento impreso y empastado, sacrificar nuevas ideas y mejoras.

Métodos para la reducción de riesgo: Se ha difundido claramente en el grupo de desarrollo el hecho de que todos los artefactos están en construcción; nunca se valora uno de

ellos como la versión definitiva. El uso de términos correctos como las versiones estables permite que los entes de control al interior del grupo puedan realizar métricas a corto plazo que en sumatoria pueden arrojar datos valiosos en cuanto al avance del proyecto. Los objetivos de los trabajos de grado pueden ser modificados respecto a los anteproyectos siempre y cuando se conserve el espíritu de los mismos. En la práctica se utiliza Cervisia, un sistema de CVS, que permite el seguimiento del proyecto, sus resultados y sus errores.

- **Falta de motivación de los integrantes del proyecto.**

Debido a la posibilidad de que los integrantes del grupo en ciertos momentos y etapas del desarrollo del proyecto carezcan de la motivación necesaria, se corre el riesgo de que las actividades asignadas no se realicen o se realicen de una forma no satisfactoria. Usualmente esto es fruto de reclutamientos forzosos de personal.

Métodos para la reducción de riesgo: A integrantes de fases preliminares se les dicta capacitaciones en los métodos de desarrollo ágiles y se les descarga en gran medida la responsabilidad de documentar el desarrollo - enfoque 100 % centrado en el código. Para el cumplimiento de las tareas y actividades se asigna un tiempo adicional con el objetivo de que los integrantes del grupo resolvieran sus dudas y preocupaciones sobre el proyecto y así atacar una determinada tarea entre todos los integrantes.

En la actualidad el proyecto es de carácter abierto y cualquier persona puede; sin una intermediación directa del grupo de investigación, realizar trabajos sobre el mismo. Los desarrollos que deseen ser validados podrán solicitar dicha certificación al grupo de desarrollo principal el cual sopesará los resultados y en algunos casos concretos el proceso de desarrollo. Es decir, a partir de la versión 3.0 del SITEM ningún nuevo integrante es regulado por las políticas del grupo principal de desarrollo, es una decisión individual el seguimiento o no de los lineamientos.

- **Imposibilidad total de trabajo.**

Posiblemente por motivos externos al proyecto como paros, vacaciones o cierres, es posible que el trabajo se detenga totalmente.

Métodos para la reducción de riesgo: Teletrabajo, depreciación de la comunicación persona a persona y uso intensivo de herramientas de trabajo en grupo dentro de la plataforma del SITEM. Especialmente las de primer nivel como correo electrónico, canales IRC y Foros. Las de segundo nivel tales como chat y videoconferencia en tiempo real han sido poco utilizadas para minimizar el riesgo colateral de segregación tecnológica.

- **Capacitación deficiente.**

Es posible que algún miembro del proyecto asignado a alguna tarea no posea todos los conocimientos necesarios para realizarla.

Métodos para la reducción de riesgo: Al interior del SITEM se han creado cursos en línea con contenidos actualizados a los que los integrantes podrán acceder de forma asíncrona a través de Internet. En los foros se guarda información relevante a los problemas en el desarrollo de tareas con el objetivo de que todo el equipo de trabajo pueda encontrar soluciones basadas en la colaboración. El jefe del proyecto puede asignar recursos enfocados en la capacitación de ciertas áreas específicas.

- **Mal funcionamiento de herramientas de software y hardware.**

La posibilidad de fallo en los equipos de trabajo y de los recursos utilizados puede causar la para y perdida de tiempo.

Métodos para la reducción de riesgo: En el grupo de desarrollo principal existe un plan de mantenimiento a corto plazo que contempla actividades para el aseguramiento de la integridad de la plataforma de hardware y de los datos asociados al proyecto. Con la autorización para desplegar el hilo principal en servidores de aplicaciones se tendrá un mayor respaldo.

- *No seguimiento del plan de trabajo.*

Las libertades propias de cada integrante del grupo además del conjunto formado por todos los otros riesgos pueden causar un desarrollo no planeado. En sí este riesgo solo es evidente cuando se captan recursos que van enfocados al cumplimiento de objetivos específicos, a partir de la versión 3.0 el grupo principal será el encargado de dictaminar los alcances de los nuevos releases.

Método para la reducción de riesgo: Uno o más de los integrantes debe actuar como ingeniero de procesos y se encargará de verificar el cumplimiento del plan de trabajo.

- **Desacuerdo entre los integrantes del grupo.**

La posibilidad de desacuerdos de trabajo o personales puede darse al interior del proyecto.

Método para la reducción de riesgo: En el caso de desarrollo novedoso los integrantes en desacuerdo deberán seguir las reglas de contribución concurrente. Para todos los demás casos - procedimiento depreciado - el jefe de proyecto y el ingeniero de procesos deberán encargarse de tomar cualquier decisión con miras al cumplimiento de los objetivos.

- **Desviación de los recursos de trabajo.**

Es posible que los recursos obtenidos para el desarrollo del proyecto sean utilizados para otras actividades que no involucren la obtención de objetivos específicos.

Método para la reducción de riesgo: El ingeniero de procesos verifica el correcto cumplimiento del plan de trabajo así como la correcta utilización de los recursos. Se realiza un historial de utilización de los recursos.

- **Otros riesgos.**

Existen otros riesgos como la renuncia al proyecto de algún miembro, el cambio de los objetivos del proyecto, cambio en las características del mercado y eventos externos que afecten el normal desarrollo del proyecto.

Método para la reducción de riesgo: El proceso de desarrollo ágil - con elementos claves del proceso unificado, la utilización de artefactos, el desarrollo centrado en la persona y el compromiso con la filosofía general de la libertad y el derecho a la información.

Anexo G

Instrumento Preliminar para la Fase de Transición

G.1. Presentación

El presente instrumento agrupa de una manera lógica los aspectos necesarios para identificar el estado y potencialidad de los servicios de salud que hacen uso intensivo de las TIC. El contenido está basado en aquel utilizado por el proyecto HERMES, modelo completamente depurado que ha servido de base para la recolección de información de varios proyectos de telemedicina emprendidos en la Comunidad Económica Europea (CEE). Así mismo se le han adicionado algunos ítem aplicables al contexto colombiano.

G.1.1. Indicaciones Generales

El cuestionario deberá ser respondido como fase posterior a la etapa de divulgación del proyecto y una vez se haya conformado un comité, o en su defecto, se haya designado un encargado, por parte de la institución objetivo. Es necesario que en el proceso los miembros del proyecto sirvan de apoyo resolviendo de una manera exacta las dudas que surjan en el ámbito tecnológico (telesalud, estándares de telecomunicaciones y otros). De igual forma se deben realimentar de la información especializada que puedan capturar de los comités. Para apoyar estas tareas podrán hacer uso de las herramientas de trabajo en grupo que se encuentran en el Portal SITEM.

La profundidad con que se respondan las cuestiones garantiza que la institución evalúe su situación actual frente a la posible implementación de servicios de salud prestados en la modalidad de Telemédicina.

El cuestionario esta disponible para ser diligenciado en línea de tal forma que podrá ser resuelto en cualquier orden y anexando cualquier tipo de información que se considere conveniente para aclarar y/o complementar una respuesta.

Las cuestiones deberán ser manejadas por un especialista en la materia aunque esto conduzca a que los encargados del proyecto jueguen un papel meramente coordinador de actividades. Todas las cuestiones deben ser tenidas en cuenta aunque se hayan tratado tácitamente en respuestas anteriores.

G.1.2. Contexto Clínico

Las preguntas que tienen que ver con los servicios clínicos están incluidos en este cuestionario dado que uno de sus objetivos es dar un conocimiento global del sistema de telemedicina y sus características.

El cuestionario puede ser usado para ver el estado actual de los aspectos clínicos y para evaluar cuales mejoras deben ser hechas. Las preguntas no pretenden definir competencias clínicas y son por lo tanto abiertas y basadas en la familia de estándares de calidad ISO 9000. Estas características están en consonancia con el concepto de Administración con Calidad Total (TQM):

1. Definición de necesidades
2. Planificación para cubrir necesidades.
3. Controles para cubrir los estándares.
4. Mejora continua

Un gran número de factores deben ser tomados en mente en consideración a los tópicos relacionados en esta sección:

Los estándares médicos deben definir el contexto sobre el cual el sistema de telemedicina ha de ser introducido, no solamente las necesidades que han de ser atendidas por dicho servicio. Es decir, la definición de estándares médicos y de un conjunto de directrices para asegurar la más alta calidad en la atención debe ser una actividad independiente del sistema de telemedicina a implementar.

Cualquier servicio medico en la modalidad de telemedicina que se quiera implementar en un ambiente donde existe un conjunto aceptado y aplicado de estándares médicos y directrices, debe estar completamente estructurado para que sea compatible. Si no lo es debe estar lo suficientemente documentado y la necesidad de su aplicación claramente dilucidada.

Es necesario tener en cuenta las siguientes definiciones:

Evento de Atención en Salud Un contacto para el cuidado de la salud que requiere que un servicio clínico sea aplicado.

Servicio Clínico Uno a más planes de salud autorizados e implementados para poder llenar los requerimientos del evento de atención en salud.

Plan de salud Uno o más procedimientos clínicos asociados a estándares médicos y directrices que se utilizan para poder cubrir un servicio médico.

Procedimiento clínico Una o más actividades de tratamiento en salud.

Directriz médica Un informe sistemáticamente desarrollado, el cual asiste en la toma de decisiones acerca de los procedimientos a seguir para una condición clínica específica.

Estándar médico Un documento, establecido por consenso y aprobado por un cuerpo consultor organizado, el cual provee para el uso común y repetido de las pautas o las características de actividades de cuidados clínicos (o sus resultados) y su objetivo principal es el de alcanzar el grado óptimo de orden en un contexto de medicina dado.

Unidades de Recolección de Información

1. ¿Cuales son los servicios de salud por modalidad telemedicina que deben ser implementados?

Describe sus ideas/necesidades para el uso de la telemedicina en su ambiente de trabajo

- ¿Cuales son los servicios clínicos que serán afectados o reemplazados por el nuevo sistema de telemedicina?
- ¿Cuales organizaciones/departamentos clínicos están directamente involucrados en el servicio?

2. ¿Cuales son los requerimientos de los usuarios?

- ¿Quiénes son los usuarios?
Quien está involucrado.
- ¿Qué hacen ellos? roles/actividades Usuarios Cliente (pacientes) Usuarios participantes (doctores, enfermeros, paramédicos, personal de soporte, etc.)
- ¿Que método era usado para obtener los requerimientos clínicos de los usuarios?
Ejemplo: ¿ Existe un grupo de usuarios, incluyendo un representante del paciente, donde se use un cuestionario o se realice una encuesta?
- ¿Cuales son los eventos/episodios clínicos que serán dirigidos en el sistema de telemedicina?
- ¿Que señales, incluyendo signos vitales, necesitarán ser monitorizados?
- ¿Qué equipo es necesario para monitorizar dichas variables?
- ¿Qué procedimientos (incluyendo entrenamiento) son requeridos para implementar la monitoria de las variables?

- ¿Quién interpretará la información monitorizada? (Especificar el nivel profesional)
 - ¿Que factores son requeridos para garantizar la seguridad del paciente, exactitud y funcionalidad de los equipos, y aceptación por parte del usuario en la monitoria y otros procedimientos?
3. ¿Cuales son los procedimientos de servicios clínicos que podrán ser afectados por la implementación de la telemedicina?
- Ejemplo: El acceso al servicio, referencia, admisión, descarga y procedimientos de seguimiento.
- Describir los eventos y procedimientos existentes, además de las particularidades del ambiente local (específicas)
- Describir una aproximación al nuevo sistema de telemedicina (tomando en consideración condiciones y directrices locales)
- Describir la estrategia con la cual se obtendrá la aprobación local y extendida de los cambios.
4. ¿Cuales son los estándares/ pautas relacionadas con el sistema de telemedicina?
- Pautas existentes que deben ser conservadas en el sistema de telemedicina.
- Pautas que deben ser reemplazadas.
- ¿Existen nuevas directrices que deban ser creadas para implementar el sistema de telemedicina?
 - ¿Que proceso de desarrollo de directrices debería ser usado?
5. ¿Cuales son los recursos humanos requeridos por el sistema de telemedicina?
- ¿El sistema de telemedicina puede ser implementado y operado por el personal existente?
 - ¿Nuevo personal debe ser entrenado?
6. ¿Cuales son las necesidades de entrenamiento para el sistema de telemedicina?
- En telemática, telemedicina, procedimientos de calidad, análisis de requerimientos, indicadores de resultados y desempeño, etc.
- Tipo de entrenamiento requerido.
- Tipo y grado del personal.

G.1.3. Contexto de servicio / relación con el entorno

Es necesario definir una estrategia de desarrollo para poder garantizar que la solución en Telemedicina encaje perfectamente en el ambiente existente de servicios en salud. Los servicios Telemédicos usualmente se montan sobre alguno de los siguientes escenarios: Creación de nuevos servicios Telemédicos ó servicio telemédico que reemplazará un servicio no telemédico existente.

Unidades de Recolección de Información

■ Restricciones legales.

La pregunta explora las restricciones legales, éticas y sociales sobre el servicio y aquellos que participan en él. Las restricciones éticas pueden estar enmarcadas en leyes pero no siempre sucede, ej. Los médicos tienen límites éticos promulgados por organismos internacionales pero razones culturales pueden flexibilizar o endurecer dichas directrices,

1. ¿Cuales son las restricciones legales de los servicios telemédicos?
 - Con respecto a las transmisiones de datos.
 - Con respecto a la protección de datos.
 - Con respecto a la responsabilidad en los productos y el seguimiento de estándares.
 - Con respecto a la libertad de información / privacidad
 - Con respecto a la responsabilidad personal y organizativa.
 - Con respecto a las entidades en las cuales se siguen responsabilidades legales.
 - ¿Que organismos son responsables de la dirección legal?
2. ¿Cuales son las consideraciones éticas
 - ¿Para quien están dirigidas las restricciones éticas?
 - ¿Con respecto a qué se definen dichas restricciones?
 - ¿Que organismos son los encargados de la dirección en cuanto a cuestiones éticas?
3. ¿Cuales son los factores culturales y sociales a ser considerados?
 - Consideraciones relacionadas con las raza.
 - Consideraciones relacionadas con creencias culturales.
 - Consideraciones relacionadas con el nivel educativo de los usuarios.
 - Consideraciones relacionadas con el grupo socio económico al cual pertenece el usuario.
 - Consideraciones relacionados con la ubicación geográfica de los usuarios.
 - Consideraciones relacionadas con la edad de los usuarios.

■ Consecuencias organizativas del sistema de telemedicina

La siguiente pregunta tiene que ver con los efectos que tendrá la introducción de un servicio de salud por modalidad de Telemedicina en la organización.

1. ¿Que consecuencias tendrá en su organización la introducción del sistema de telemedicina?
 - Sobre la practica actual del trabajo.
 - Sobre los estándares locales.
 - Sobre las inversiones existentes.
2. ¿Que características técnicas u organizativas requieren adaptación?

- Practica actual del trabajo.
 - Estándares locales.
 - Inversiones existentes.
 - Políticas administrativas.
 - Maquinaria consultiva.
- ¿Que organizaciones/ contratistas deben interactuar para proveer el servicio de salud por modalidad de Telemedicina?
 - Organizaciones gubernamentales
 - Administración regional de salud.
 - Organizaciones públicas.
 - Organizaciones profesionales.
 - Organizaciones voluntarias (cruz roja, defensa civil, etc)
 - Compañías de telecomunicaciones (Proveedores de servicios de interconexión)
 - Compañías de seguros.
 - Otras compañías.

G.1.4. Consideraciones Tecnológicas

Esta sección esta relacionada con la definición de los recursos técnicos y de infraestructura necesarios para la provisión del sistema de telemedicina.

Unidades de Recolección de Información

1. Requerimientos de hardware

- ¿ Cuales plataformas de hardware son requeridas? (tanto como para proveer el servicio de salud en la modalidad de Telemedicina como para la utilización del mismo)
Ej. PC, UNIX,
- ¿Que tipo de almacenamiento se requiere?
 - De acuerdo a la cantidad de información.
 - De acuerdo a las políticas de seguridad.
 - De acuerdo a la velocidad de búsqueda de la información.
- ¿Que tecnología de visualización se requiere?
- ¿Cuales son los requerimientos de procesamiento y como puede ser medido?
Ejemplo: Velocidad, procesamiento en paralelo. Medición en MFlops, Mips, Whets-tones, Dhrystones o otras medidas estándar.

- ¿Que requerimientos especiales deben tener los equipos del usuario final?
Ej. Unidades móviles u otros dispositivos de usuario específicos (para captura de información, análisis, etc)
 - ¿Que puertos de comunicaciones deberán estar disponibles?
Ejemplo: Ethernet, V32, SCSI, RS 232, USB, conectores especiales a otros equipos.
 - ¿Que tecnologías de adquisición, entrada o salida de datos son requeridas?
Ej. Dispositivos de monitoria, modems, impresores, OCR, etc.
 - ¿Existen en su institución dispositivos que cumplan los requerimientos?
2. Software
- ¿Cuales sistemas operativos se requiere?
 - ¿ Que herramientas de desarrollo son requeridas para desarrollar aplicaciones específicas para el sistema de telemedicina?
Ejemplo: Lenguajes de programación, compiladores, enlazadores, depuradores, editores, herramientas CASE.
3. ¿ Que infraestructura de comunicaciones se requiere? Ejemplo: Hardware de red, Software de red,
4. ¿ Que enlaces de comunicaciones se requiere? Ejemplo: Hardware (modems, bridges, switch, router,etc)
- ¿ Que formatos?
 - ¿ Que velocidad en los tiempos de respuesta?
 - ¿ Que estándares / protocolos?
 - ¿ Que ancho de banda?
5. Software aplicativo.
- ¿ Cuales son los requerimientos para KBS, DSS?
 - ¿ Cuales son los requerimientos para los sistemas de bases de datos?
 - ¿ Cuales son los requerimientos para otro tipo de software aplicativo? Ejemplo:
¿Existe software comercial, GNU, GPL que cubra las necesidades?
6. Requerimientos en infraestructura.
- ¿ Que mobiliario se requiere?
 - ¿ Que requerimientos eléctricos deben ser cumplidos?
Voltaje, corrientes, regulación, aislamiento, sistemas de tierra, etc.
 - ¿ Que temperatura, iluminación o humedad se requiere?
 - ¿ Que cambios deben realizarse a los sitios existentes? Ejemplo: Cableado, requerimientos de espacio, etc.
7. ¿ Cuales metodologías están implementadas (o deben ser adoptadas) para cumplir los requerimientos tecnológicos y de infraestructura del servicio?

G.1.5. Consideraciones de Calidad

Esta sección contiene preguntas que deben ser contestadas para garantizar un servicio que tome en cuenta los factores de calidad en todas las etapas del proyecto.

Unidades de Recolección de Información

1. Aseguramiento de la calidad.

- ¿Cuales son las consideraciones de calidad a tener en cuenta en el sistema tele médico?

Ejemplo:. Mejoramiento de los mecanismos de cuidados para el paciente, mejoramiento de la eficiencia, reducción de costos, etc.

- ¿Cuales métodos de administración con calidad total (TQM) existen o deben ser implementados?
- ¿Quien es responsable de la supervisión de los servicios médicos / de enfermería, servicios técnicos, servicios administrativos y cuales son sus roles en el servicio telemédico?
- ¿Como podría ser validado el sistema de telemedicina?
Definir factores, variables, criterios e indicadores.
- ¿Existen manuales y políticas de calidad implementadas para el sistema de telemedicina?
- ¿Como podría obtenerse aprobación de la autoridades?
Ejemplo: Legislación Local, Nacional o certificación ISO

2. Grupo de usuarios

- ¿Está o estará un grupo de usuarios establecido para el sistema de telemedicina?
- ¿Esta o estará disponible un manual de usuario para el sistema de telemedicina?
- ¿Existirá enlaces entre los diversos grupos de usuarios nacionales/mundiales de servicios de salud similares brindados por la modalidad de telemedicina?

3. Describa las estrategias que se seguirán para dar conocimiento al público del nuevo servicio telemédico

- Estrategias para promocionar los objetivos del servicio y la forma de acceder a él.
- Estrategias para el material educativo que será usado en el servicio.
- Estrategia para la mejora, promoción de resultados e impacto del servicio.

4. ¿Como podrían evaluarse las variables de calidad?

5. ¿Cuales son los valores aceptables para las variables de calidad seleccionadas ?

6. ¿Quien es el responsable de definir las estrategias de calidad?
7. ¿Que tipo de documentación y procedimientos son usados (o se necesitan) para asegurar que la calidad total esta implementada?
8. ¿Que criterios pueden ser utilizados para demostrar que la calidad en el servicio es máxima?

G.1.6. Aceptación del servicio de salud brindado en la modalidad de telemedicina

Unidades de Recolección de Información

1. ¿Qué tipos de intereses (beneficios) son inherentes al establecimiento de un sistema de telemedicina y que variables deben ser consideradas?

Tipos de interés cuyos valores deben ser considerados:

- Clínicos : Doctores, enfermeras, etc.
 - Tecnológicos: Técnicos, investigadores, doctores, desarrolladores, etc.
 - Éticos.
 - Económicos.
 - Industriales: Empresas, institutos de investigación/universidades, sociedad en general.
2. ¿Qué hace que el sistema de telemedicina sea aceptado por las personas envueltas en el proyecto?
 3. ¿Cual es el costo proyectado para establecer el sistema de telemedicina?
 - Costos Capitales Tangibles: Equipos, instalación/pruebas, Cambios estructurales (Edificios/cuartos), cambios en la organización (Procedimientos), depreciación.
 - Costos operativos: Personal, Servicio de Comunicaciones, mantenimiento y servicio, control de calidad, capacitación de personal, jornadas de socialización.
 4. ¿Quien debería pagar por el establecimiento / implementación del sistema de telemedicina?

Entre otros puede estar:

- Hospital, centros atención de salud.
- Universidad.
- Empresas de Telecomunicaciones.
- Fundaciones de investigación.

- Proveedores de servicios de salud bajo la modalidad de telemedicina, compañías de aseguramiento.
5. ¿Cómo debe ser evaluado el sistema de telemedicina?
- Evaluación económica.
 - Evaluación de impacto social.
 - Evaluación de impacto médico.
 - Evaluación clínica.
 - Evaluación a las comunidades de práctica.
6. ¿Que técnicas de evaluación están disponibles?
- Investigación Evaluativa.
 - Evaluación Integral.
 - Etnografía.
 - Análisis de costos.
 - Análisis costo/beneficio.
 - Ingeniería económica.

G.1.7. Aspectos de Construcción y Persistencia del sistema de telemedicina

Unidades de Recolección de Información

1. Tiempo de vida del proyecto
 - ¿ Cual es el tiempo de vida proyectado para el sistema de telemedicina?
 - ¿ Que etapas pueden identificadas en dicho tiempo de vida?
 - ¿Cuánto tiempo tardaría en implementarse el servicio en una nueva ubicación?
 - ¿ Cuando y cuales mejoras pueden ser anticipadas / planeadas?
2. ¿ Que recursos son o deben estar disponibles para establecer el sistema de telemedicina?
 - Recursos de información (manuales, educación, soporte)
 - Recursos financieros.
 - Personal (Habilidades, incentivos, educación, entrenamiento)
 - Espaciales (arquitectónicos)
3. ¿ Que recursos deben ser re – evaluados?
 - Financieros.

- De personal.
 - Espaciales.
 - De cultura en la organización.
 - De mecanismos de interacción con el entorno.
4. Disponibilidad de los recursos para la operación del sistema de telemedicina.
- ¿En que sitios?
 - ¿En que tiempo?.
 - ¿Quien los suple?
 - ¿A quien o que van dirigidos?

G.1.8. Políticas de la Organización

Esta sección es relacionada con la formulación de políticas administrativas con respecto a la prestación de servicios Tele médicos.

Las políticas versan sobre los objetivos de la compañía, prácticas estandarizadas, regulaciones, estatutos, manuales de procedimientos, manuales de seguridad, códigos de ética, etc.

1. ¿ Que políticas / regulaciones administrativas pueden tener efecto sobre el sistema de telemedicina?
 - Con relación con la disponibilidad y distribución de recursos.
 - Con relación a las estructuras de la organización.
 - Con relación a las personas que son afectadas
 - Con relación a la calidad (norma ISO 9000 o similares)
2. ¿Que implicaciones pueden tener para el sistema de telemedicina?
 - Con relación con la disponibilidad y distribución de recursos.
 - Con relación a las estructuras de la organización.
 - Con relación a las personas que son afectadas.
3. ¿ Cómo deben ser definidas las políticas de seguridad, manejo de información y calidad para el sistema de telemedicina?
 - Con respecto a políticas administrativas.
 - Con respecto a la disponibilidad de fuentes de información.
 - Con respecto a la libertad de uso de la información, a la protección de datos y la confidencialidad.

- Con respecto a la legislación.
- Con respecto a consideraciones éticas y sociales.
- Con respecto a la distribución de privilegios en el acceso al sistema de telemedicina.
- Con respecto a las políticas de calidad definidas.

G.1.9. Consideraciones acerca de la información

Esta sección cubre la información que ha de ser parte del sistema de Telemedicina, incluye cuestiones sobre flujo de datos, estructura de datos y archivos, almacenamiento de datos, entre otras.

Las necesidades de información del sistema y el procesamiento que se haga de ella tienen una gran implicación cuando se pretende elegir la tecnología a ser utilizada.

Unidades de Recolección de Información

1. ¿Que tipos de información son necesarios para el sistema telemédico?
 - texto, numérico, imagen, video.
 - Imágenes de rayos X, MRI.
2. Formatos de archivos y estándares de codificación.
 - ¿Cuales estándares de codificación deberán ser usados para tipos genéricos de datos?
Ejemplo. MPEG, JPEG, MHEG, IA5, etc.
 - ¿Que estándares de codificación médica deberán ser usados para los elementos de información?
Ej. Códigos READ, ICD9, ICD10, WHO.
3. ¿ Cuales son las actividades en las cuales la información está involucrada? ¿ Como y donde está siendo utilizada?
4. Calidad de la información
 - ¿ Con que calidad la información debe ser utilizada en las diferentes etapas del servicio. (recolección, procesamiento, transmisión y visualización)?
 - ¿Que tan relevante es la calidad en esas etapas para la exactitud de la conclusión final?
 - ¿Como puede ser maximizado el nivel de exactitud del diagnóstico final?
5. ¿Que procedimientos de control de calidad en la información son necesarios?

6. ¿ Como es el flujo de la información en el sistema de Telemedicina (dentro de las fronteras del sistema)?
7. ¿Cual es la información que fluye desde y hacia el sistema de Telemedicina?
8. ¿ que pasos deben ser tomados en cuenta para garantizar la seguridad de los datos?

G.1.10. Procesamiento de información

Unidades de Recolección de Información

1. ¿Que conversiones en la información son (o deberían ser hechos) antes de su presentación ¿ (Si los datos originales son transmitidos entonces ninguna conversión deberá ser necesaria)
Ej. De-comprensión de imágenes, decodificación, etc.
2. ¿Que procesamiento de señal es requerido?
3. ¿Cuales son las implicaciones de tales transformaciones / procesamientos?
 - Con respecto a responsabilidades legales.
 - Con respecto a consideraciones de calidad.
4. ¿ Cómo deberá ser presentada la información al usuario?
 - Requerimientos en interfase de usuario.

Anexo H

Otros Entregables Nucleares del SITEM

H.1. Visión - Resumen Ejecutivo

H.1.1. Propósito

SITEM es un *Portal Web* especializado en la gestión de datos e información de diferentes componentes estructurales de los sistemas de telemedicina. Provee un ambiente de apoyo a las tareas de las comunidades de práctica involucradas en la investigación, el diseño, mantenimiento, desarrollo e implementación de redes de Telemedicina. Tuvo su génesis conceptual en el año 2000, en la primera fase del Proyecto Telemedicina Bogotá, como solución a la necesidad de administrar los resultados del estudio de campo realizado a las entidades e instituciones de salud y los operadores de Telecomunicaciones en la ciudad de Bogotá.

Su principal objetivo es apoyar las actividades básicas de los denominados *trabajadores del conocimiento* en el área de la telemedicina ofreciéndoles, además de un repositorio de datos, herramientas que facilitan las tareas de capturar, extraer, organizar, analizar, encontrar, sintetizar, distribuir y compartir información y conocimiento. El ideal es actualizar el estado de ciertos nodos interesantes del Sistema de Salud de Bogotá Distrito Capital, haciendo énfasis en la posibilidad de interacción a nivel nacional e internacional y en los requerimientos que en Telemedicina tengan las diferentes entidades que participan o no en el proyecto de Telemedicina auspiciado por el grupo GITEM.

H.1.2. Alcance

El SITEM es principalmente *un concepto*, su estado actual es una representación del potencial real del sistema que debe ser socializado y entregado a la comunidad. La base de desarrollo principal es el grupo GITEM y será responsable de la versión oficial del producto. Sin embargo, dada la dinámica en el mundo del software libre, el grupo GITEM no limitará el trabajo independiente que sobre su desarrollo realice cualquier persona o grupo de personas. En este sentido la funcionalidad original del sistema podrá ser modificada pero no avalada directamente por el grupo.¹.

El SITEM ha sido creado con el fin de apoyar a los grupos de trabajo que realizan labores en el área de proyección de sistemas de Telemedicina. La información que en él se encuentra debe ser ingresada por personas autorizadas para asegurar en un alto grado la veracidad e idoneidad de la misma. Sin embargo no se puede garantizar, y no se garantiza, la exactitud, disponibilidad, integridad y oportunidad de dicha información: LA INFORMACIÓN CONTENIDA EN EL SITEM NO ES UNA FUENTE OFICIAL DE DATOS. El uso de la misma es responsabilidad de quien lo realiza. La información que se encuentre en el SITEM no ha sido necesariamente revisada por expertos profesionales. Todos los contenidos que se ingresen al SITEM deben ser de licencia pública o de libre uso; los contenidos que no cumplan estos criterios serán eliminados.

H.1.3. Posicionamiento

- **Definición del problema**

La mayoría de los estudios base de conocimiento se encuentran disgregados y en idiomas diferentes al español por lo cual su consulta es compleja y no existe un mapa seguro de navegación que guíe al investigador hacia las fuentes confiables de información.

- **Afecta a**

Investigadores, consultores, usuarios y proveedores de servicios en el área de la salud.

- **El impacto asociado es**

Estudios abandonados, e inconclusos, junto con la complejidad innecesaria del proceso de determinación del estado del arte, están abocando a los grupos universitarios a competir codo a codo - a pesar de todas sus limitaciones - contra grandes empresas multinacionales interesadas en “sacar del camino” a estos facilitadores de procesos.

- **Una solución parcial adecuada sería**

Un Sistema Informático que en un ambiente integrado ofrezca posibilidades a los usuarios para la administración de información sobre varios componentes tecnológicos de las redes de telemedicina así como la posibilidad de realizar seguimiento al cumplimiento de ciertos indicadores en los proyectos de Telemedicina.

¹Salvo en casos en que no se trasgredan directamente los objetivos primarios del desarrollo. En tales casos las contribuciones serán asociadas al hilo oficial de desarrollo.

Un sistema que sea fácilmente adaptable a las necesidades novedosas y que este basado en software libre para concentrar la inversión en su desarrollo y no en le pago de licencias de uso o de compra de herramientas de programación.

- **Para**

Investigadores, estudiantes, usuarios, prestadores de servicios de salud, prestadores de servicios de telecomunicaciones, programadores.

- **Quienes**

Son los beneficiarios directos del despliegue de servicios médicos por la modalidad de Telemedicina.

- **Nuestro producto**

Sistema de Información para el Apoyo de Grupos de Trabajo en Proyectos de Telemedicina. SITEM

Disminuye el tiempo de adquisición, análisis y despliegue de la información. Es construido guiado por adaptaciones de procesos de desarrollo ampliamente conocidos y siguiendo el paradigma de la orientación a objetos con lo que se garantiza su facilidad de mantenimiento, escalabilidad e indirectamente su permanencia en el medio.

Contiene módulos para la generación de estadísticas e informes pormenorizados de cada uno de los componentes y logra obtener en unos pocos segundos los datos necesarios para apoyar la labor de análisis, diseño e implementación de proyectos telemédicos o de telesalud. Usa un esquema modular de crecimiento a la medida en donde el esfuerzo para la creación de instrumentos nuevos de consulta se minimiza por el uso de plantillas prediseñadas. En lugar de ser un Sistema estático, SITEM contiene características de adaptación dinámica para cubrir las necesidades que tengan los próximos proyectos emanados del GITEM y otras entidades que hagan uso del sistema.

H.1.4. Participantes en el Proyecto y Usuarios

Perfil de los participantes del SITEM. En nuestro desarrollo nos unimos al manifiesto de los metodólogos ágiles manteniendo ciertas pautas del Proceso Unificado para poder dar fe de la calidad en el proceso y el producto:

H.1.5. Entorno de usuario

El usuario opera una interfaz web a través de un navegador HTTP, con soporte para HTML 4.0, XML 2.0, javascript, XSL y Cascada Style Sheet 1.0.

Para acceder a las diferentes secciones del SITEM se requiere que el usuario ejecute un proceso de Autenticación, Autorización y Registro (AAA) – asociados a una sesión. Hasta la versión

Nombre	Descripción	Responsabilidades
Director de Proyecto.	Directora Grupo GITEM	Garantiza el flujo de recursos para el desarrollo del proyecto. Seguimiento del desarrollo del proyecto. Aprueba requisitos y funcionalidades Generales
Arquitectos del Sistema	Se encarga de Definición, modelado del Problema – Arquitectura del Sistema Solución Engloba las funciones de los antiguos analistas, diseñadores e ingenieros de Proceso	Caso de desarrollo aplicando en parte el Proceso Unificado. Determinar las necesidades de los usuarios del Sistemas. Generar los niveles más altos de requerimientos del sistema. Asegurar los criterios de consistencia, pertinencia y completitud del modelo de requisitos. Particionar el SITEM en subsistemas y componentes. Generar artefactos del modelo de requisitos, análisis y diseño.
Ingenieros de Prueba	Se encargan de desplegar los casos de prueba para garantizar que los ejecutables cumplen con los requisitos de los usuarios.	Diseñar Casos de Prueba Realizar pruebas. Proponer modificaciones. Depurar componentes.
Programador	En el SITEM representa el integrante de mayor jerarquía dentro del proceso de desarrollo. Engloba las funciones asociadas a los demás participantes.	Desarrollar componentes

Tabla H.1: Perfil de los participantes del SITEM.

3.0 se mantendrá un entorno gráfico básico centrado especialmente en hipervínculos y diseño gráfico mínimo.

H.1.6. Suposiciones y dependencias

- La plataforma tecnológica sobre la que se implementa el módulo tiene una disponibilidad superior al 99 por ciento del tiempo.
- Las herramientas de desarrollo son Software Libre.
- El SITEM podrá integrar en su arquitectura otras aplicaciones de Software Libre o Público.
- El hilo principal de desarrollo estará en la Universidad Distrital pero no se restringirá la distribución del producto a usuarios interesados.
- El grupo de participantes en el SITEM es indefinido. Los procesos se potenciarán en la medida que se produzcan “explosiones” de desarrollo fomentadas por usuarios interesados.

H.1.7. Descripción Global del SITEM

SITEM es implementado sobre una arquitectura multicapa que distribuye los diferentes componentes en tres capas principales: Presentación, aplicación y datos, estando presente una capa transversal tácita de seguridad. A nivel de usuario el SITEM está compuesto por siete subsistemas autónomos que prestan servicios a sus pares. Estos agrupan seis componentes claves en todo proyecto de telemedicina: entidades de salud, operadores de telecomunicaciones, tecnologías de interconexión, equipos y tecnologías de captura de datos, proyectos e instituciones relacionadas con la telemedicina y servicios médicos - incluyendo módulos de vademécum, consulta de procedimientos, enfermedades y especialidades médicas.

H.1.8. Otros Requisitos del Producto

Estándares Aplicables

- Unified Process
- Unified Modeling Language versión 2.0
- Extensible Markup Language Versión
- SOAP

- OWL

El sistema debe ser:

- Multiplataforma.
- Multiusuario.

Requisitos de Desempeño

- Velocidad de acceso promedio interior a 10 s.
- Ayudas contextuales y contenidos autoexplicativos.
- Disponibilidad superior al 99 por ciento.
- Manejo de conexiones concurrentes.
- Integridad referencial en la capa de persistencia.

H.1.9. Lineamientos de codificación para la organización de los módulos

Para asegurar una codificación eficiente, que permita realizar búsquedas rápidas dentro de la organización documental, se tienen las siguientes reglas de obligatorio cumplimiento en todos los artefactos:

El nombre del artefacto deberá estar antecedido de un identificador del tipo:

aaa-bbb-ccc inicialesmodulo-tipoartefacto-versiónartefacto

Así, para la primera versión del documento de especificaciones de casos de uso, del módulo de administración de instrumentos para la recolección de información ha de tenerse una codificación similar a:

MAI-ECU-001

Siendo MAI y ECU los identificadores únicos tomados del artefacto Códigos

Anexo I

Análisis de Necesidad de un Bus de Servicio Empresarial

Para determinar si el proyecto se beneficiaba de un Bus de servicio Empresarial, se respondió la Lista de Chequeo propuesta por Ross Mason del proyecto Mule:

I.1. Capacidad de Integración

¿Está integrando tres (3) o más aplicaciones o servicios?

Respuesta: SI.

¿Tiene más de diez (10) aplicaciones para integrar?

Respuesta: NO

I.2. Protocolo de Comunicación

¿Requiere usar más de un protocolo de comunicación?

Respuesta: NO.

I.3. Capacidades especiales de mensajería

¿Necesita capacidad para el enrutamiento de mensajes tales como bifurcación o agrupación de flujos de mensajes; o enrutamiento basado en el contenido?

Respuesta: NO.

I.4. Orientación a Servicios

¿Necesita publicar servicios que deban ser consumidos por otras aplicaciones?

Respuesta: SI

I.5. Escalabilidad vs Simplificación

¿Realmente necesita la escalabilidad provista por un ESB?

Respuesta: NO DETERMINADO

¿Conoce el alcance de su aplicación, puede iniciar sin requerir un ESB?

Respuesta: SI

Anexo J

Aspectos Relativos al Despliegue

El despliegue de la solución requiere una plataforma tecnológica adecuada tal como se muestra en el anexo E para garantizar un óptimo servicio a los potenciales usuarios ¹. Las herramientas básicas se encuentran disponibles para que el grupo de investigación convoque a las entidades de salud, profesionales en el área de la tecnología, especialistas en medicina y fabricantes - distribuidores - de dispositivos médicos que participaron en el Estudio Red de Telemedicina Bogotá [Aparicio,2000] para que de forma conjunta enriquezcan la base de información en el subproceso de prueba piloto (presupuestada como Fase IV).

Basado en el módulo de generación de herramientas para la recolección de información, anexo ??, el grupo aplicará diferentes instrumentos entre los que se incluyen:

- Entrevistas: Con el objeto de acordar las directrices a seguir tanto con los operadores de telecomunicaciones, como los prestadores del servicio de salud que participaron de la fase I de recolección de información con el propósito de socializar los resultados del proyecto e invitarlos para que editen y actualicen sus datos obteniendo beneficios incrementalmente. Dichos servicios van desde la disponibilidad de un mapa de sedes, servicios y profesionales hasta la consolidación de información para la gestión de sus redes tecnológicas primarias.
- Encuestas: Para identificar nuevos requerimientos de servicios que puedan ser desplegados en la plataforma propuesta. Además, estos instrumentos permitirán medir el impacto en la cobertura de los servicios y de participación de las instituciones objeto de la investigación de campo.

Las unidades de recolección de información mostradas en el anexo G, se han adaptado de aquellas propuestas por el proyecto europeo HERMES. ² El modelo investigación evaluativa

¹La versión actual esta desplegada en servidores de la intranet universitaria

²Proyecto de investigación a tres años, financiado por la Comunidad Económica Europea y que cumplió sus

que continua en la fase IV del proyecto pretende medir la evolución del nivel de servicios de salud prestados con el apoyo de TIC comparando sucesivamente el modo de operación encontrado entre el año 2000 y 2005 con aquel encontrado entre el año 2008 y 2011 luego que algunas entidades interactúen con el SITEM.

J.0.1. Fuentes de Información Primaria

Para la carga de información inicial en el SITEM se utilizan los resultados del estudio de campo realizado en varias instituciones de carácter público y privado. Dicho resultados se encuentran consignados en sendas tesis en formato digital e impresos disponibles en la biblioteca de la Universidad Distrital y archivo del grupo de investigación:

- Hospital Rafael Uribe Uribe.[Guarin *et al*,2013]
- Hospital San Pedro Claver.[Ardila,2011],[Rozo *et al*,2002]
- Hospital Simón Bolívar. [Acero y Ariza,2012]
- Hospital El Tunal. [Ruiz y Niño,2012][Duque *et al*,2012]
- Hospital La Victoria.[Barrero,2010]
- Hospital San José.[González y Torres,2012]

objetivos hacia principios del milenio dejando como resultado un conjunto de preguntas básicas que apoyan los procesos de implementación de soluciones médicas apoyadas en las TIC.

Anexo K

Entrevista Base para Identificación de Interesados

En la elaboración y validación de este instrumento de recolección de la información se siguió el método definido en la sección de *Metodología para la Estructuración y Validación del Documento*, presentado en el capítulo de Estructura del Proyecto. Es necesario recalcar que el formato de entrevista es semi-estructurado y el entrevistador tendrá las preguntas solo como referencia.

K.1. Metodología de Procesamiento

El instrumento debe aplicarse en la plataforma de gestión de encuestas. Cada una de las respuestas se analiza de manera separada y se califican los atributos de claridad, pertinencia, independencia (que no tenga similitud con respuestas dadas por otro entrevistado), impacto y completitud.

Las preguntas que tengan mayores índices se procesarán primero y los usuarios que las respondieron se clasificarán como interesados del proyecto. Cada una de las preguntas viene acompañada de su categoría para facilitar la interpretación.

K.2. Instrumento

K.2.1. Pregunta

Categoría: Hechos. Objetivo: Identificar si el entrevistado posee información específica. Enunciado: ¿Cuáles son los procedimientos que desarrolla al interior del proyecto?

K.2.2. Pregunta

Categoría: Hechos. Enunciado: ¿Qué datos produce al desarrollar su trabajo?

K.2.3. Pregunta

Categoría: Opinión. Objetivo: Determinar la percepción del entrevistado respecto a un aspecto concreto. Comentario: Se utiliza para obtener un criterio base para análisis multi-temporal. Enunciado: ¿Cuáles son los procedimientos que desarrolla al interior del proyecto?

K.2.4. Pregunta

Categoría: Creencia. Objetivo: Adquirir un conocimiento de la creencia, prejuicio o concepto que tiene el entrevistado respecto a un tema. Comentario: Se utiliza como insumo para analizar posibles obstáculos o fortalezas que pueda tener una creencia en el desarrollo del proyecto. Enunciado: ¿Cree que las tecnologías de información pueden contribuir a mejorar la productividad del grupo?

K.2.5. Pregunta

Categoría: Opinión. Enunciado: ¿Las herramientas software que apoyan su trabajo al interior del grupo son adecuadas?

K.2.6. Pregunta

Categoría: Opinión. Enunciado: ¿Las herramientas software que apoyan su trabajo al interior del grupo son suficientes?

K.2.7. Pregunta

Categoría: Intención. Objetivo: Examinar los mecanismos de optimización de tareas que utilizaría el entrevistado. Enunciado: ¿Cómo desarrollaría su trabajo si existiese una herramienta adecuada?

Anexo L

GNU General Public License

Licencia Pública General

Versión 2, Junio de 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Traducción No oficial

Toda persona tiene permiso de copiar y distribuir copias fieles de este documento de licencia, pero no se permite hacer modificaciones.

L.1. PREAMBULO

Los contratos de licencia de la mayor parte del software están diseñados para quitarle su libertad de compartir y modificar dicho software. En contraste, la **GNU General Public License** pretende garantizar su libertad de compartir y modificar el software, esto para asegurar que el software es libre para todos sus usuarios.

Esta licencia pública general se aplica a la mayoría del software de la “FSF Free Software Foundation” (Fundación para el Software Libre) y a cualquier otro programa de software cuyos autores así lo establezcan. Algunos otros programas de software de la Free Software Foundation están cubiertos por la “LGPL Library General Public License” (Licencia Pública General para Librerías), la cual puede aplicar a sus programas también.

Cuando hablamos de software libre, nos referimos a libertad, no a precio. Esta licencia está diseñada para asegurar que:

1. Usted tiene la libertad de distribuir copias del software libre (y cobrar por ese sencillo

servicio si así lo desea)

2. Recibir el código fuente (o tener la posibilidad de obtenerlo si así lo desea)
3. Usted puede modificar el software o utilizar partes de el en nuevos programas de software libre
4. Usted esté enterado de que tiene la posibilidad de hacer todas estas cosas.

Para proteger sus derechos, necesitamos hacer restricciones que prohíban a cualquiera denegarles estos derechos o a pedirle que renuncie a ellos. Estas restricciones se traducen en algunas responsabilidades para usted si distribuye copias del software, o si lo modifica.

Por ejemplo, si usted distribuye copias de un programa, ya sea gratuitamente o por algún importe, usted debe dar al que recibe el software todos los derechos que usted tiene sobre el mismo. Debe asegurarse también que reciban el código fuente o bien que puedan obtenerlo si lo desean. Y por último debe mostrarle a esa persona estos términos para que conozca los derechos de que goza.

Nosotros protegemos sus derechos en 2 pasos: (1) protegiendo los derechos de autor del software y (2) ofreciéndole este contrato de licencia que le otorga permiso legal para copiar, distribuir y modificar el software.

Además, para la protección de los autores de software y la nuestra, queremos asegurarnos de que toda persona entienda que **no existe ninguna garantía del software libre**. Si el software es modificado por alguien y lo distribuye, queremos que quienes lo reciban sepan que la copia que obtuvieron no es la original, por lo que cualquier problema provocado por quien realizó la modificación no afectará la reputación del autor original.

Finalmente, cualquier programa de software libre es constantemente amenazado por las patentes de software. Deseamos evadir el peligro de que los re-distribuidores de un programa de software libre obtenga individualmente los derechos de patente con el fin de volver dicho programa propietario. Para prevenir esto, hemos dejado en claro que cualquier patente deberá ser licenciada para el uso libre de toda persona o que no esté licenciada del todo.

A continuación se describen con precisión los términos y condiciones para copiar, distribuir y modificar el software.

L.2. TERMINOS Y CONDICIONES PARA COPIA, MODIFICACION Y DISTRIBUCION

0. Esta licencia aplica a cualquier programa o trabajo que contenga una nota puesta por el propietario de los derechos del trabajo estableciendo que su trabajo puede ser distribuido

bajo los términos de esta “GPL General Public License”. El “Programa”, utilizado en lo subsecuente, se refiere a cualquier programa o trabajo original, y el “trabajo basado en el Programa” significa ya sea el Programa o cualquier trabajo derivado del mismo bajo la ley de derechos de autor: es decir, un trabajo que contenga el Programa o alguna porción de él, ya sea íntegra o con modificaciones y/o traducciones a otros idiomas. De aquí en adelante “traducción” estará incluida (pero no limitada a) en el término “modificación”, y la persona a la que se aplique esta licencia será llamado “usted”.

Otras actividades que no sean copia, distribución o modificación no están cubiertas en esta licencia y están fuera de su alcance. El acto de ejecutar el programa no está restringido, y la salida de información del programa está cubierta sólo si su contenido constituye un trabajo basado en el Programa (es independiente de si fue resultado de ejecutar el programa). Si esto es cierto o no depende de la función del programa.

1. Usted puede copiar y distribuir copias fieles del código fuente del programa tal como lo recibió, en cualquier medio, siempre que proporcione de manera consiente y apropiada una nota de derechos de autor y una declaración de no garantía, además de mantener intactas todas las notas que se refieran a esta licencia y a la ausencia de garantía, y que le proporcione a las demás personas que reciban el programa una copia de esta licencia junto con el Programa. Usted puede aplicar un cargo por el acto físico de transferir una copia, y ofrecer protección de garantía por una cuota, lo cual no compromete a que el autor original del Programa responda por tal efecto.
2. Usted puede modificar su copia del Programa o de cualquier parte de él, formando así un trabajo basado en el Programa, y copiar y distribuir tales modificaciones o bien trabajar bajo los términos de la sección 1 arriba descrita, siempre que cumpla con las siguientes condiciones:
 - A. Usted debe incluir en los archivos modificados notas declarando que modificó dichos archivos y la fecha de los cambios.
 - B. Usted debe notificar que ese trabajo que distribuye contiene totalmente o en partes al Programa, y que debe ser licenciado como un conjunto sin cargo alguno a cualquier otra persona que reciba sus modificaciones bajo los términos de esta Licencia.
 - C. Si el programa modificado lee normalmente comandos interactivamente cuando es ejecutado, usted debe presentar un aviso, cuando el programa inicie su ejecución en ese modo interactivo de la forma más ordinaria, que contenga una noticia de derechos de autor y un aviso de que no existe garantía alguna (o que sí existe si es que usted la proporciona) y que los usuarios pueden redistribuir el programa bajo esas condiciones, e informando al usuario como puede ver una copia de esta Licencia. (Excepción: si el programa en sí es interactivo pero normalmente no muestra notas, su trabajo basado en el Programa no tiene la obligación de mostrar tales notas)

Estos requerimientos aplican al trabajo modificado como un todo. Si existen secciones identificables de tal trabajo que no son derivadas del Programa original, y pueden ser razonablemente consideradas trabajos separados e independientes como tal, entonces esta Licencia y sus términos no aplican a dichas secciones cuando usted las distribuye

como trabajos separados. Pero cuando usted distribuye las mismas secciones como parte de un todo que es un trabajo basado en el Programa, la distribución del conjunto debe ser bajo los términos de esta Licencia, cuyos permisos para otras personas que obtengan el software se extienden para todo el software, así como para cada parte de él, independientemente de quién lo escribió.

No es la intención de esta sección de reclamar derechos o pelear sus derechos sobre trabajos hechos enteramente por usted, en lugar de eso, la intención es ejercer el derecho de controlar la distribución de los trabajos derivados o colectivos basados en el Programa. Adicionalmente, el simple agregado de otro trabajo NO basado en el Programa al Programa en cuestión (o a un trabajo basado en el Programa) en algún medio de almacenamiento no pone el otro trabajo bajo el alcance de esta Licencia.

3. Usted puede copiar y distribuir el Programa (o un trabajo basado en él, bajo la Sección 2) en código objeto o en forma de ejecutable bajo los términos de las secciones 1 y 2 arriba descritas siempre que cumpla los siguientes requisitos:

A. Acompañarlo con el correspondiente código fuente legible por la máquina, que debe ser distribuido bajo los términos de las secciones 1 y 2 y en un medio comúnmente utilizado para el intercambio de software, o

B. Acompañarlo con una oferta escrita, válida por al menos 3 años y para cualquier persona, por un cargo no mayor al costo que conlleve la distribución física del código fuente correspondiente en un medio comúnmente utilizado para el intercambio de software, o

C. Acompañarlo con la información que usted recibió sobre la oferta de distribución del código fuente correspondiente. (Esta alternativa está permitida sólo para distribución no-comercial y sólo si usted recibió el Programa en código objeto o en forma de ejecutable con tal oferta de acuerdo a la subsección b anterior)

El código fuente de un trabajo significa la forma preferida de hacer modificaciones al mismo. Para un trabajo ejecutable, un código fuente completo significa todo el código fuente de todos los módulos que contiene, mas cualquier archivo de definición de interfaces, mas los programas utilizados para controlar la compilación y la instalación del ejecutable.

Sin embargo, como excepción especial, no se requiere que el código fuente distribuido incluya cualquier cosa que no sea normalmente distribuida con las componentes mayores (compilador, kernel, etc.) del sistema operativo en el cual el ejecutable corre, a menos de que una componente en particular acompañe al ejecutable.

Si la distribución del ejecutable o del código objeto se hace ofreciendo acceso a copiar desde un lugar designado, entonces el ofrecer acceso equivalente para copiar el código fuente desde el mismo lugar se considera distribución del código fuente, aunque las demás personas no copien el código fuente junto con el código objeto.

4. Usted no puede copiar, modificar, sub-licenciar ni distribuir el Programa a menos que sea expresamente bajo esta Licencia, de otra forma cualquier intento de copiar, modificar, sub-licenciar o distribuir el programa es nulo, y automáticamente causará la pérdida de sus derechos bajo esta Licencia. Sin embargo, cualquier persona que haya recibido copias o derechos de usted bajo esta Licencia no verán terminadas sus Licencias ni sus derechos perdidos mientras ellas continúen cumpliendo los términos de esta Licencia.

5. Usted no está obligado a aceptar esta Licencia, dado que no la ha firmado. Sin embargo, nada le otorga el permiso de modificar o distribuir el Programa ni sus trabajos derivados. Estas acciones están prohibidas por la ley si usted no acepta esta Licencia. Sin embargo, modificando o distribuyendo el Programa (o cualquier trabajo basado en el Programa) indica su aceptación de esta Licencia y de todos sus términos y condiciones para copiar, distribuir o modificar el Programa y/o trabajos basados en el.
6. Cada vez que usted redistribuye el Programa (o cualquier trabajo basado en el Programa), la persona que lo recibe automáticamente recibe una licencia del autor original para copiar, distribuir o modificar el Programa sujeto a estos términos y condiciones. Usted no puede imponer ninguna restricción adicional a las personas que reciban el Programa sobre los derechos que en esta Licencia se les otorga. Usted no es responsable de forzar a terceras personas en el cumplimiento de esta Licencia.
7. Si como consecuencia de un veredicto de un juzgado o por el alegato de infringir una patente o por cualquier otra razón (no limitado solo a cuestiones de patentes) se imponen condiciones sobre usted que contradigan los términos y condiciones de esta Licencia, éstas no le excusan de los términos y condiciones aquí descritos. Si usted no puede distribuir el producto cumpliendo totalmente con las obligaciones concernientes a la resolución oficial y al mismo tiempo con las obligaciones que se describen en este contrato de Licencia, entonces no podrá distribuir más este producto. Por ejemplo, si una licencia de patente no permitirá la distribución del Programa de forma libre de regalías (sin pago de regalías) por parte de quienes lo reciban directa o indirectamente, entonces la única forma de cumplir con ambas obligaciones es renunciar a la distribución del mismo.

Si cualquier parte de esta sección resulta inválida, inaplicable o no obligatoria bajo cualquier circunstancia en particular, la tendencia de esta es a aplicarse, y la sección completa se aplicará bajo otras circunstancias. La intención de esta sección no es la de inducirlo a infringir ninguna ley de patentes, ni tampoco infringir algún reclamo de derechos, ni discutir la validez de tales reclamos; esta sección tiene el único propósito de proteger la integridad del sistema de distribución del software libre, que está implementado por prácticas de licencia pública. Mucha gente ha hecho generosas contribuciones a la amplia gama de software distribuido bajo este sistema favoreciendo así la constante aplicación de este sistema de distribución; es decisión del autor/donador si su Programa será distribuido utilizando este u otro sistema de distribución, y la persona que recibe el software no puede obligarlo a hacer ninguna elección en particular.

Esta sección pretende dejar muy en claro lo que se cree que será una consecuencia del resto de esta Licencia.

8. Si la distribución y/o el uso del Programa se restringe a algunos países ya sea por patentes, interfases protegidas por derechos de autor, el propietario original de los derechos de autor que ubica su Programa bajo esta Licencia puede agregar una restricción geográfica de distribución explícita excluyendo los países que aplique, dando como resultado que su distribución sólo se permita en los países no excluidos. En tal caso, esta Licencia incorpora la limitación como si hubiera sido escrita en el cuerpo de esta misma Licencia.
9. La "FSF Free Software Foundation" puede publicar versiones nuevas o revisadas de la "GPL General Public License" de uno a otro momento. Estas nuevas versiones man-

tendrán el espíritu de la presente versión, pero pueden diferir en la inclusión de nuevos problemas o en la manera de tocar los problemas o aspectos ya presentes.

Cada versión tendrá un número de versión que la distinga. Si el Programa especifica un número de versión para esta Licencia que aplique a él y cualquier versión subsecuente", usted tiene la opción de seguir los términos y condiciones de dicha versión o de cualquiera de las posteriores versiones publicadas por la "FSF". Si el programa no especifica una versión en especial de esta Licencia, usted puede elegir entre cualquiera de las versiones que han sido publicadas por la "FSF".

10. Si usted desea incorporar partes del Programa en otros Programas de software libre cuyas condiciones de distribución sean distintas, deberá escribir al autor solicitando su autorización. Para programas de software protegidas por la "FSF Free Software Foundation", deberá escribir a la "FSF" solicitando autorización, en ocasiones hacemos excepciones. Nuestra decisión será guiada por dos metas principales:

Mantener el estado de libertad de todos los derivados de nuestro software libre

Promover el uso comunitario y compartido del software en general

INEXISTENCIA DE GARANTÍA

11. DEBIDO A QUE EL PROGRAMA SE OTORGA LIBRE DE CARGOS Y REGALÍAS, NO EXISTE NINGUNA GARANTÍA PARA EL MISMO HASTA DONDE LO PERMITA LA LEY APLICABLE. A EXCEPCIÓN DE QUE SE INDIQUE OTRA COSA, LOS PROPIETARIOS DE LOS DERECHOS DE AUTOR PROPORCIONAN EL PROGRAMA COMO ES SIN NINGUNA GARANTÍA DE NINGÚN TIPO, YA SEA EXPLÍCITA O IMPLÍCITA, INCLUYENDO, PERO NO LIMITADA A, LAS GARANTÍAS QUE IMPLICA EL MERCADEO Y EJERCICIO DE UN PROPÓSITO EN PARTICULAR. CUALQUIER RIESGO DEBIDO A LA CALIDAD Y DESEMPEÑO DEL PROGRAMA ES TOMADO COMPLETAMENTE POR USTED. SI EL SOFTWARE MUESTRA ALGÚN DEFECTO, USTED CUBRIRÁ LOS COSTOS DE CUALQUIER SERVICIO, REPARACIÓN O CORRECCIÓN DE SUS EQUIPOS Y/O SOFTWARE QUE REQUIERA. 12. EN NINGÚN CASO NI BAJO NINGUNA CIRCUNSTANCIA EXCEPTO BAJO SOLICITUD DE LA LEY O DE COMÚN ACUERDO POR ESCRITO, NINGÚN PROPIETARIO DE LOS DERECHOS DE AUTOR NI TERCERAS PERSONAS QUE PUDIERAN MODIFICAR Y/O REDISTRIBUIR EL PROGRAMA COMO SE PERMITE ARRIBA, SERÁN RESPONSABLES DE LOS DAÑOS CORRESPONDIENTES AL USO O IMPOSIBILIDAD DE USAR EL PROGRAMA, SIN IMPORTAR SI SON DAÑOS GENERALES, ESPECIALES, INCIDENTALES O CONSECUENTES CORRESPONDIENTES AL USO O IMPOSIBILIDAD DE USAR EL PROGRAMA (INCLUYENDO PERO NO LIMITADO A LA PERDIDA DE INFORMACIÓN O DETERIORO DE LA MISMA AFECTANDOLO A USTED, A TERCERAS PERSONAS QUE SEA POR FALLAS EN LA OPERACIÓN DEL PROGRAMA O SU INTERACCIÓN CON OTROS PROGRAMAS) INCLUSIVE SI TAL PROPIETARIO U OTRAS PERSONAS HAYAN SIDO NOTIFICADAS DE TALES FALLAS Y DE LA POSIBILIDAD DE TALES DAÑOS.

FIN DE TÉRMINOS Y CONDICIONES