CAMACUÁ

Sistema para prevención y detección en tiempo real de situaciones de riesgo y ubicación de entidades en hospitales.

Autores

Ignacio Decia

Agustín Farías

Tutor

Franco Simini

Informe de Proyecto de Grado presentado al Tribunal Evaluador como requisito de graduación de la carrera Ingeniería en Computación

Facultades de Ingeniería y Medicina

Universidad de la República

Montevideo, Uruguay

1 de julio de 2016

CAMACUÁ

Sistema de prevención de situaciones de riesgo para pacientes y ubicación en hospitales.

Ignacio Decia
Agustín Farías

Resumen

En los hospitales se dan situaciones que pueden comprometer la integridad física de un paciente. Una de las más notables es el error de medicación, definido como un incidente prevenible que puede causar daño al paciente, relacionado a una mala utilización de medicamentos, u omisión en su administración. Otro problema es la circulación de personas u objetos en zonas indebidas del hospital, por ejemplo, pacientes que se fugan o cambian de cama sin consentimiento, u objetos contaminados en zonas estériles.

Se propone un sistema de tiempo real para la ubicación de personas u objetos y alerta de situaciones de riesgo, que combina la tecnología de código de barras con la identificación por radiofrecuencia. Cuenta con un mecanismo que permite definir a un usuario situaciones de riesgo o interés, por las que el sistema debería emitir alertas.

Se implantó un prototipo en una sala del Hospital Maciel de Montevideo, Uruguay, donde las primeras pruebas de ubicación con los sensores de identificación por radiofrecuencia mostraron resultados correctos en el 90 % de los casos.

Palabras clave: *error de medicación, seguridad del paciente, sistema de ubicación en tiempo real, detección de eventos*

Agradecimientos

Agradecemos a Álvaro Villar, María Piñeyrúa, Déborah Szerman, Silvia Pérez, Diego Briatore, Juan García, Sandra Peraza, María Constantín, Fernando Penone, Marcelo Barbato y al resto del personal del Hospital Maciel que colaboró en el desarrollo del proyecto. También a Pía Artgaveytia, Romina Romero, Pablo Ezzatti y Martín Pedemonte por sus aportes en la mejora de la calidad del informe. A nuestro tutor, Franco Simini, por su disposición y enseñanzas. Finalmente, agradecemos profundamente a nuestras familias y amigos, quienes nos acompañaron durante todo el camino.

Índice general

Índice de figuras

Índice de tablas

Lista de abreviaturas

ADT Admission Discharge Transfer

AP Access Point

CDA Clinical Document Release

CEP Complex Event Processor

COSEPA Comisión de Seguridad del Paciente

ECA Event Condition Action

EPC Electronic Product Code

EPL Event Processing Language

FDA Food and Drug Foundation

GCM Google Cloud Messaging

HCE Historia Clínica Electrónica

HF High Frecuency

HL7 Health Level Seven

IHE Integrating the Healthcare Enterprise

LASA Look Alike Sound Alike

LF Low Frecuency

MPI Master Patient Index

NFC Near Field Communication

OEM Ondas ElectroMagnéticas

PIX Patient Identifier Cross-referencing

QBP Query By Parameter

RBAC Rol Based Access Control

RFID Radio Frecuency Identification

RLTS Real Time Location System

RSSI Received Signal Strength Indicator

SHF Super High Frecuency

SSO Single SignOn

UHF Ultra High Frecuency

UWB Ultra Wide Band

XDS Cross-Enterprise Document Sharing

Capítulo 1

Introducción

1.1. Definición del problema

La operativa de una institución de salud involucra el trabajo conjunto y la interacción de varios actores. Su objetivo principal es prestar atención médica de calidad a los usuarios (pacientes). Es necesario, entonces, reducir la probabilidad de situaciones irregulares que desencadenen contratiempos o afecten la integridad del paciente. Entre los posibles problemas se encuentran [1–4]:

- Errores de medicación.
- Cirugías al paciente equivocado, o del lado contrario al que estaba indicada la operación.
- Demoras para ubicar pacientes, personal, equipamiento médico u otros objetos en situaciones necesarias.
- Circulación indebida de personas u objetos dentro del edificio.
- Dificultades en la gestión operativa, como el desconocimiento de la asignación de camas.

Los errores de medicación refieren a la administración incorrecta de medicamentos a pacientes. Para garantizar el adecuado tratamiento deben considerarse los siguientes aspectos [4]:

- Paciente correcto: debe controlarse la correspondencia medicación-paciente.
- Medicamento correcto: el medicamento prescripto debe ser el correcto para tratar la afección del paciente.
- Dosis correcta: la cantidad de medicación administrada debe ser la adecuada.
- Vía de administración correcta: oral, intravenosa, etc.
- Momento correcto: la medicación debe ser administrada a la hora indicada.

El proceso de administración de medicación se puede dividir en tres etapas:

- Prescripción médica.
- Armado y dispensado de unidosis (empaquetado de medicamentos para un único paciente).

- Administración al paciente.

En cada una de estas etapas se pueden cometer errores que resulten en el incumplimiento de cualquiera de los puntos citados anteriormente.

Existen varios reportes que muestran porcentajes de errores cometidos en cada etapa, y el impacto en pacientes. Un estudio realizado en 2008 en Estados Unidos indica que un 34 % de los errores ocurren durante la administración, de los cuales solo el 2 % son identificados antes de que la medicación sea administrada al paciente [5]. Otro estudio realizado en 2010 por Poon et al. en otro hospital del mismo país concluye que, dentro de los errores serios de medicación [6], un 11 % se da en el armado y dispensado de unidosis, y un 38 % en la administración [7]. A pesar de los esfuerzos realizados para evitarlos, los errores de medicación aún siguen ocurriendo [8].

Se consultó con integrantes de la Comisión de Seguridad del Paciente (COSEPA) del Ministerio de Salud Pública por estadísticas sobre errores de medicación en Uruguay, y la respuesta fue negativa. Los únicos datos disponibles son anecdóticos. Una de las principales dificultades para la adquisición de información es la falta de reportes frente a estas situaciones, por temor a sanciones. De todas formas, los representantes de COSEPA estiman como más frecuentes los errores de prescripción, seguidos de los de administración, y finalmente los de armado y dispensado de unidosis. Mencionaron que muchas soluciones informáticas fallan al no conocer completamente el dominio del problema, o introducen nuevas fuentes de error. Otra de las problemáticas planteadas es que no existe una cultura orientada a la seguridad del paciente, lo que dificulta la instauración de nuevas herramientas centradas solo en procedimientos.

Un estudio realizado en 2006 [3], basado en 13 años de datos históricos pertenecientes a varias organizaciones de salud en Estados Unidos, estimó que, de todos los procedimientos quirúrgicos realizados en dicho país, hay un promedio anual de 1300 a 2700 intervenciones realizadas a pacientes equivocados, del lado contrario, o se efectuó el procedimiento incorrecto [1]. Una consulta a expertos de dominio evidenció que entre las causas principales de este tipo de errores se encuentra la falta de datos de identificación del paciente, u omisiones en el proceso de identificación.

Por otra parte, la circulación de personas u objetos en lugares indebidos podría dar lugar a ciertas situaciones que deben evitarse. Ejemplos de problemas planteados por los expertos de dominio son:

- Pacientes con demencia que podrían fugarse del hospital y perderse.
- Pacientes que cambian de cama sin consentimiento de los funcionarios del centro de salud.
- Objetos extraviados.
- Tránsito de personas y objetos en zonas no permitidas, como la existencia de desechos contaminados en zonas estériles.

Es de interés, también, conocer la ubicación de personal o equipamiento médico para reducir demoras en los procedimientos.

[1] El estudio no indica el total de intervenciones quirúrgicas anuales realizadas en Estados Unidos.

1.2. Motivación

El estudio citado anteriormente - realizado por Poon et al. - muestra que la introducción de la tecnología de código de barras en el proceso de administración de medicación disminuyó la tasa de errores 67 % en la fase de dispensación y 51 % en la administración. Por otra parte, la FDA emitió en 2004 un reglamento que expresa la obligación de emplear códigos de barras en medicamentos y productos biológicos, como muestras de sangre, para su identificación [9]. En consecuencia, resulta prometedora la aplicación de esta tecnología para la resolución de la problemática planteada. Integrantes de la COSEPA creen que la introducción de este tipo de tecnología mitiga el riesgo de error de medicación.

A su vez, es de interés hacer un relevamiento de las tecnologías de ubicación e identificación existentes, tanto en el mercado como a nivel académico. La construcción de un prototipo que resuelva los problemas relacionados a estos aspectos constituye un desafío interesante.

Es deseable contar con un sistema que registre de manera automática situaciones de interés, que permitan ganar visibilidad sobre los procesos operativos del hospital, para poder analizarlos y mejorarlos.

1.3. Objetivos planteados

- Realizar un relevamiento de requerimientos, con expertos de dominio, para lograr una especificación completa de los problemas de interés, y proponer soluciones que permitan resolverlos o mitigarlos.
- Desarrollar un prototipo de bajo costo, que funcione como prueba de concepto de las soluciones.
- Implantar y evaluar el prototipo desarrollado.

1.4. Organización del documento

El resto del trabajo se estructura como sigue:

- El capítulo 2 introduce las tecnologías investigadas y los conceptos necesarios para la comprensión del resto del documento.
- El capítulo 3 presenta una descripción del estado del arte en sistemas comerciales y académicos relevantes para el contexto de este trabajo.
- En el capítulo 4 se realiza un análisis del problema, presentando los requerimientos relevados, y se introduce la solución planteada.
- El capítulo 5 describe la arquitectura del prototipo desarrollado.
- El capítulo 6 detalla las principales decisiones de implementación y tecnologías utilizadas.
- En el capítulo 7 se describen las pruebas funcionales y de rendimiento realizadas, y los

resultados obtenidos.

- Finalmente, el capítulo 8 presenta las conclusiones y trabajo futuro.

Capítulo 2

Conceptos y tecnologías asociados

En este capítulo se introducen los conceptos básicos y tecnologías, necesarios para la comprensión del informe.

2.1. Conceptos médicos

Las siguientes definiciones fueron extraídas del decreto número 148/001 [10], que adopta las terminologías descritas en la resolución GMC 21/00 del Mercosur [11]:

Paciente: usuario de los establecimientos de salud.

Paciente internado: paciente que, admitido en el hospital, pasa a ocupar una cama por un período mayor a 24 horas.

Historia clínica: documento médico-legal constituido por formularios estandarizados o no, destinado al registro de la atención prestada al paciente.

Hospital (o sanatorio): establecimiento de salud destinado a prestar asistencia sanitaria, en régimen de internación, a la población, pudiendo disponer de otros servicios.

Cama hospitalaria: cama destinada a la internación de un paciente en el hospital. Se refiere a las camas incluidas en la capacidad instalada del hospital y localizadas en un espacio, en una habitación o sala de internación, constituidas en la dirección exclusiva de un paciente durante su permanencia en el hospital y vinculadas a una Unidad (sala) de Internación y a uno o más servicios.

Censo diario hospitalario: conteo del número de camas ocupadas, cada 24 horas.

Dado que el sistema presentado en el informe interactúa únicamente con pacientes internados, se utilizarán indistintamente los términos *Paciente* y *Paciente internado*, sin riesgo de generar ambigüedad.

Otros conceptos importantes, no contemplados en la resolución GMC 21/00 son:

Prescripción: documento escrito por un Médico con directivas a un Farmacéutico sobre cómo preparar y dispensar un medicamento [12].

Error de medicación: error producido en la administración de un medicamento a determinado paciente. Puede ocurrir debido a que el paciente no tenía indicada esa unidosis, a que contiene medicamentos incorrectos o en dosis incorrectas, o a que el momento o vía de administración no son los adecuados [4]. El Ministerio de Salud Pública adopta la definición dada por el *National Coordinating Council for Medication Error Reporting and Prevention* [13]: Un error de medicación es «cualquier incidente prevenible que puede causar daño al paciente o dar lugar a una utilización inapropiada de los medicamentos, cuando éstos están bajo el control de los profesionales sanitarios o del paciente o consumidor. Estos incidentes pueden estar relacionados con la práctica profesional, los procedimientos o los sistemas, incluyendo fallos en la prescripción, comunicación, etiquetado, envasado, denominación, preparación, dispensación, distribución, administración, educación, seguimiento y utilización» [14].

Unidosis, o dosis unitaria: conjunto de medicamentos empaquetados, destinados a un único paciente. El Farmacéutico se encarga del armado de la unidosis, a partir de las prescripciones emitidas por un médico.

2.2. Estándares de informática médica

En el proceso de asistencia médica los pacientes suelen transitar por varios prestadores de salud y centros especializados. Cada vez más son las instituciones que deben intercambiar información clínica entre sí, con el fin de cooperar y lograr un diagnóstico temprano y efectivo. La heterogeneidad de los sistemas informáticos involucrados hacen que este intercambio sea cada vez más complejo. Es aquí donde entran en juego los estándares informáticos, que cumplen un papel fundamental para lograr la interoperabilidad de los sistemas. Esta sección introduce brevemente los estándares de informática médica investigados, que se describen con más detalle en el apéndice E.

2.2.1. Identificación de pacientes

Es común que los pacientes tengan múltiples registros o identificadores entre los sistemas de cada institución, y debe ser posible determinar cuáles corresponden a la misma persona. Es necesario, entonces, contar con un método preciso de identificación, independiente de los identificadores locales, que asegure una asociación correcta entre el paciente y su información clínica (estudios, resultados de laboratorio, etc). La correspondencia entre registros o identificadores se conoce como el problema de referencias cruzadas, y se emplean algoritmos que permiten agrupar los que corresponden a una misma persona. Estos operan sobre un conjunto de atributos representativos, es decir, que permiten distinguir una persona de otra. Entre los atributos más utilizados se encuentran: documento de identidad, nombre, apellido, fecha de nacimiento y sexo. Las soluciones tecnológicas más comunes se basan en el estándar *Patient Identifier Cross-referencing* (PIX) o en la utilización de registros maestros de pacientes.

PIX [15,16] es un estándar definido por *Integrating the Healthcare Enterprise* (IHE), cuyo público objetivo son las instituciones que tengan necesidad de soportar referencia cruzada de identificadores de pacientes entre múltiples dominios de identificación. Un dominio de identificación se define como un conjunto formado por uno o varios sistemas interconectados que comparten un esquema de identificación. En general, cada institución tendrá su propio dominio, y los pacientes tendrán asignados identificadores únicos dentro de él. Cuando una institución desea consultar por estudios realizados a uno de sus pacientes en otra institución, primero realiza una consulta al manejador de

referencias (responsable de mantener las referencias cruzadas), enviando como dato su identificador local. El manejador retornará la lista de identificadores en el resto de los dominios. De esta forma la institución podrá determinar el identificador del paciente en la otra institución y realizar las consultas correctamente.

Por otro lado, un *Master Patient Index* (MPI), o Registro Maestro de Pacientes, es un repositorio de pacientes que actúa como un manejador PIX. La diferencia principal es que, además de agrupar registros, asigna un identificador global a cada paciente.

2.2.2. Documentos clínicos

Los documentos electrónicos son la unidad fundamental de intercambio de información entre los distintos prestadores, que describen los actos clínicos o eventos de salud realizados sobre los pacientes. El estándar HL7 CDA R2 [17] define la estructura y semántica de documentos clínicos con el propósito de que sean fácilmente intercambiables. Los documentos CDA son archivos XML que constan de un encabezado y un cuerpo. El encabezado contiene un conjunto de metadatos que establecen información básica y de contexto como el tipo de documento, autor, persona a la que aplica, fecha y lugar de creación. El cuerpo describe el acto clínico que da lugar al documento y puede variar en el nivel de estructuras y codificaciones que se utilizan.

Por otra parte, también es necesario especificar cómo se lleva a cabo este intercambio. El estándar *Cross-Enterprise Document Sharing* (XDS.b) [15, 16, 18, 19], definido por la IHE, facilita el registro, intercambio y acceso de información clínica entre distintos prestadores de salud. Los documentos son organizados en carpetas, y los metadatos se utilizan para realizar búsquedas y lograr recuperarlos. Entre los actores principales que define se encuentran el repositorio XDS y registro XDS. El primero es el encargado de almacenar los documentos, mientras que el segundo almacena los metadatos y mantiene el índice de documentos. Cuando una institución desea acceder a un documento, primero realiza una consulta enviando los metadatos al registro XDS, y este responde con el identificador del repositorio donde se encuentra.

2.3. Arquitectura basada en eventos

A continuación se describe brevemente la teoría detrás de los sistemas informáticos capaces de reaccionar ante la detección de eventos previamente especificados. Estos sistemas cuentan con lenguajes de especificación que permiten al usuario definir eventos simples y complejos (combinación de eventos simples), y reglas para las acciones a ser ejecutadas tras su detección. Los eventos simples constituyen la principal entrada de datos del sistema, y son procesados por el motor de eventos (también llamado procesador de eventos) encargado de la detección y cumplimiento de las reglas y acciones. Las siguientes descripciones se basan en [20–23].

La arquitectura basada en eventos tiene su origen en las bases de datos activas. Estas cuentan con un manejador que contempla continuamente el estado de los registros, y es capaz de iniciar acciones apropiadas en respuesta a modificaciones en los datos, o ante la ocurrencia de estados o transiciones específicas.

Esta característica fundamental de las bases de datos activas se extendió rápidamente a los sistemas de información que tienen como objetivo controlar situaciones de algún tipo. Uno de los

esquemas empleados para la definición de reglas se conoce como *Event–Condition–Action* (ECA). Una regla ECA actúa sobre un evento y establece que la acción *A* tendrá lugar siempre que ocurra el evento *E* y se cumpla la condición *C*.

Según la literatura referenciada, un evento puede ser definido como la ocurrencia de un suceso en el tiempo. En general, un suceso tiene un comienzo y un fin (y por lo tanto una duración), que se representa por un intervalo en el eje de tiempo. Los sucesos modelan situaciones que requieren de un tiempo finito para poder completarse o finalizar. Por el contrario, un evento marca la ocurrencia de un suceso y por lo tanto se representa con un único punto en el eje de tiempo. La ocurrencia puede corresponder al instante que da comienzo al suceso, al que da fin, o cualquiera dentro de su intervalo de duración. Esta distinción, aunque sutil, es necesaria a efectos prácticos de poder seleccionar un punto en el tiempo que denote la ocurrencia del suceso. Un evento es, por lo tanto, algo instantáneo y atómico (ocurre completamente o no ocurre).

Los eventos se clasifican en primitivos y complejos. Los primitivos son generados por actores externos al sistema, a excepción de los temporales, que marcan instantes de tiempo particulares y se generan internamente. Por otro lado, un evento complejo se define de manera recursiva, aplicando un conjunto de operadores sobre otros eventos primitivos o complejos ya definidos. Dentro de los operadores más comunes se encuentran:

- and ($E_1 \wedge E_2$): ocurre cuando E_1 y E_2 ocurren, sin importar el orden.
- or ($E_1 \vee E_2$): ocurre cuando E_1 o E_2 ocurre.
- not ($\neg E$): ocurre cuando ninguna instancia de E ocurre.
- seq (E_1; E_2): ocurre cuando E_1 ocurre seguido de E_2.
- within ([E, t]): ocurre cuando E ocurre dentro de un intervalo menor a t.

La sintaxis utilizada para definir reglas ECA comprende variantes del siguiente esquema:

$$\textbf{ON}\ event\ ;\ \textbf{IF}\ condition\ ;\ \textbf{DO}\ action \tag{2.1}$$

donde la acción (*action*) es ejecutada si y solo si la condición (*condition*) es verdadera cuando ocurre el evento (*event*).

2.4. Tecnologías de identificación y ubicación disponibles

En esta sección se analizan algunas de las tecnologías relevadas, y se realiza una comparación de las ventajas, desventajas, y posibles aplicaciones en el marco del proyecto.

2.4.1. Identificación por radiofrecuencia

La radiofrecuencia comprende a todas las ondas electromagnéticas (OEM) que caen en el rango de frecuencias menores a 3 GHz [24]. El espectro de radiofrecuencias se divide en nueve bandas, según muestra la tabla 2.1.

Una OEM adquiere propiedades físicas diferentes según su frecuencia, y es adecuada para distintos tipos de aplicación [25]: a frecuencias bajas atraviesa mejor los metales y materiales líquidos como

Nombre	Frecuencia
Very Low Frecuency (VLF)	3 - 30KHz
Low Frecuency (LF)	30 - 300KHz
Medium Frecuency (MF)	300 - 3000KHz
High Frecuency (HF)	3 - 30MHz
Very High Frecuency (VHF)	30 - 300MHz
Ultra High Frecuency (UHF)	300 - 3000MHz
Super High Frecuency (SHF)	3GHz - 30GHz
Extremely High Frecuency (EHF)	30GHz - 300GHz
Tremendously High Frecuency (THZ)	300GHz - 3000GHz

Tabla 2.1: Bandas de radiofrecuencia

el agua, mientras que a frecuencias altas puede transportar mayor información, y ser detectada a distancias más largas.

La identificación por radiofrecuencia (RFID, por sus siglas en inglés) comprende a toda tecnología de captura e identificación automática de información almacenada en etiquetas electrónicas (*tags*) adheridas a objetos físicos que utilice la radiofrecuencia para la comunicación [26]. En su forma más básica, los sistemas RFID consisten en tags que se adhieren a objetos, y de lectores (o también llamados sensores) que se encargan de obtener la información almacenada en los primeros.

Existen tres tipos de tags: pasivos, semiactivos, y activos. En el caso de los pasivos y semiactivos, los lectores RFID interrogan a los tags, emitiendo señales de radiofrecuencia que son alcanzadas por los que se encuentren en el área de cobertura del lector. Cuando uno de ellos recibe la señal propagada, comienza a transmitir la información almacenada en su memoria, habitualmente un código de identificación. Por otro lado, los tags activos inician periódicamente la transmisión de datos al lector sin el mensaje previo de interrogación. Se puede encontrar una descripción más detallada en el apéndice A.

La ventaja principal de esta tecnología es que la recuperación de la información contenida en la etiqueta se realiza sin necesidad de que exista contacto físico o línea de visión entre el lector y el tag.

Existen diversos sistemas RFID utilizados en diferentes aplicaciones y configuraciones. Varían en frecuencia de operación, fuente de energía del tag y funcionalidad. Los requerimientos y restricciones que se tengan van a determinar los costos y especificaciones técnicas del sistema de RFID a utilizar. Ejemplos de aplicaciones de esta tecnología son: rastreo e identificación de objetos, control de acceso, sistemas de pago automático y control de robo de objetos [27].

2.4.2. Ultrasonido

Se entiende por ultrasonido al conjunto de ondas sonoras con frecuencias por encima de los 20 KHz (umbral del espectro audible). Los equipos de ultrasonido utilizados en medicina trabajan con frecuencias en el rango de 2 a 15 MHz [28].

Se puede utilizar ultrasonido como base física para sistemas de ubicación en tiempo real (RTLS, por sus siglas en inglés) [29–31]. Los mecanismos diseñados para tal fin en general se basan en

dispositivos emisores de ultrasonido que envían pulsos a receptores en posiciones conocidas. Dichos receptores determinan el tiempo de llegada de los pulsos. Con esta información estiman las distancias a los emisores, y, en base a ellas, las posiciones, con un rango operativo típico de unos 10 m [32].

Una de las desventajas mencionadas en los artículos a los que se hace referencia es que la velocidad del sonido varía con la temperatura del medio en que se propaga, y en particular el ultrasonido es muy sensible a gradientes de temperatura. Además dependen fuertemente del viento.

2.4.3. Bluetooth

Bluetooth es un estándar abierto de comunicación inalámbrica de radio de corto alcance entre dispositivos [33]. Utiliza FH-CDMA [34] para evitar interferencias entre dispositivos bluetooth, y entre estos y otros que emitan señales de radiofrecuencia. Para corrección de errores utiliza FEC y ARQ [35]. Además cuenta con un indicador de intensidad de la señal recibida (RSSI).

Un RTLS basado en esta tecnología, puede valerse de la RSSI para estimar las distancias a dispositivos fijos y triangular la posición [36,37]. Sin embargo, este tipo de implementación es impreciso para ubicar exactamente un dispositivo dentro de un edificio [38]. Otro tipo de implementación ubica nodos con precisión de habitación (esto es, detecta en qué habitación se encuentra un nodo dado), pero los tiempos de estimación de ubicación están en el orden de los minutos, y se necesita una buena densidad de sensores bluetooth para obtener un funcionamiento razonable [39].

2.4.4. Biometría

La identificación a través de biometría ocurre mediante la determinación de patrones fisiológicos o conductuales de una persona. Normalmente, los métodos de biometría basados en la conducta del sujeto detectan: forma de caminar, interacción con una computadora (a través de sus periféricos de entrada), voz (pronunciación y forma de hablar), parpadeos, movimiento de labios, firma, entre otras; mientras que los basados en su fisiología analizan patrones de cara, oídos, iris, huellas dactilares, geometría de la mano, o patrones vasculares de manos y retinas [40,41].

Aunque poco frecuentemente, un sistema basado en identificación por biometría puede confundir a una persona con otra [41, 42]. Además, el éxito de la obtención de los patrones biométricos depende en gran medida de la interacción sensor-usuario [43].

2.4.5. WiFi

WiFi es el nombre comercial dado al mecanismo de enlace inalámbrico entre dispositivos, basado en la familia de protocolos 802.11x [44]. La tecnología WiFi opera a una frecuencia de 2.4 GHz y alcanza rangos de 50 a 100 m. Recientemente ha tomado popularidad como tecnología de soporte para RTLSs, el cual permite estimar la ubicación de los dispositivos conectados a la red [32].

El uso de esta tecnología es ventajoso porque aprovecha la infraestructura de red existente actualmente en la mayoría de lugares. En un sistema de RTLS-WiFi, cada objeto es etiquetado con un tag WiFi que se comunica con los *Puntos de Acceso* (AP) transmitiendo señales de radiofrecuencia. El sistema utiliza el nivel de intensidad de señal recibido, o RSSI, para calcular la distancia del

AP al tag y estimar la ubicación, según la literatura citada previamente.

Por otro lado, esta tecnología presenta algunas desventajas que dificultan su adopción como sistema de RTLS [45]. En primer lugar, el estándar WiFi fue diseñado para sustituir a las redes Ethernet, y no como un protocolo de RTLS. Muchos de los puntos de accesos son instalados en posiciones inadecuadas para RLTS, y no todos permiten obtener el nivel de señal con la precisión requerida por un sistema de RTLS. Por otro lado, la señal a 2.4 GHz experimenta interferencia y deterioro con objetos del ambiente, especialmente, en contacto con el cuerpo humano. Este fenómeno, sumado al rango de lectura extenso, trae como consecuencia que un AP que se encuentra lejos pueda leer un RSSI más alto que un AP que se encuentra próximo al tag, pero que está siendo obstruido por un objeto.

2.4.6. Beacon

La tecnología Beacon (del término *baliza* en inglés) se compone de dos elementos: un beacon transmisor, y un dispositivo que recibe y procesa las señales. Un beacon es un chip radiotransmisor que emite señales periódicamente a dispositivos equipados para recibirlas [46]. En este sentido, un beacon es un dispositivo activo, alimentado por batería, que inicia transmisiones sin ser interrogado. Los beacons emplean la tecnología *Bluetooth Low Energy* (BLE) [47] para transmitir información.

Los dispositivos encontrados en el mercado tienen un costo mínimo aproximado de cinco dólares, una duración de 1 a 5 años, y un alcance de lectura de 30 a 50 m [48].

2.4.7. Códigos de barras y QR

Se entiende por código de barras a «la representación de información legible por una máquina que es formada por combinaciones de zonas de baja y alta reflectancia sobre la superficie de un objeto, las cuales son convertidas en 0 s y 1 s» [49]. Existen tres tipos de código de barras: los lineales, que consisten en barras paralelas y espacios de ancho variable; los bidimensionales, que utilizan diferentes figuras geométricas y espacios entre ellas distribuidos en un plano; y, finalmente, los tridimensionales, que son cualquiera de los tipos anteriores escritos directamente sobre la superficie del objeto, valiéndose de diferencias de profundidad en los símbolos que contienen para conseguir cambios de reflectancia.

Una de las ventajas que presentan los códigos de barras frente a otras tecnologías es el costo. A modo de ejemplo, los lectores de tarjetas RFID UHF de mano cuestan en el orden de los 1000 a 3000 dólares, mientras que el costo de un lector de código de barras ronda en los 20 dólares (ver apéndice B). Los dispositivos móviles inteligentes tienen la capacidad de interpretar códigos de barras, y existen librerías que implementan esta funcionalidad [50]. Además, los datos en un código de barras no pueden ser alterados sin modificar el código mismo, por lo que provee una característica de seguridad interesante (ver sección 2.5).

La desventaja más notable es que la lectura de un código de barras requiere que no haya obstáculos entre este y el sensor, por lo que resulta imposible en la práctica identificar entidades (pacientes, visitantes, instrumental, etc.) de forma automática, a partir de códigos de barras adjuntos a estas, en un ambiente no controlado. Si bien esto implica la necesidad de una persona que opere el sensor, resulta mucho más rápido obtener los datos en el código de barras con un sensor, que leerlos e introducirlos a través de un teclado [51]. A su vez es menos propenso a errores, pues se reduce el

factor humano.

2.4.8. GPS

El Sistema de Posicionamiento Global (GPS, por sus siglas en inglés) es un sistema de navegación basado en 24 satélites distribuidos alrededor de la Tierra, de tal forma que queden determinados seis planos orbitales, con cuatro satélites en cada uno de ellos. En cada punto del planeta hay de cuatro a diez satélites visibles en todo instante; para poder obtener información de posición se necesitan solo cuatro [52].

La tecnología GPS es, por sí sola, bastante imprecisa dentro de edificios [53–55]. Las soluciones comerciales de localización en interiores para teléfonos móviles inteligentes basadas en GPS requieren de asistencia de un operador para calibrarlas [56].

2.4.9. Comparativa de tecnologías

La tabla 2.2 muestra un resumen de las características principales de las tecnologías estudiadas.

Característica	a	b	c	d	e	f	g	h	i
Rango de lectura máximo	100m	10m	6m	50m	d/a	100m	50m	10cm	n/a
Requiere línea de visión	No	No	No	No	d/a	No	No	Sí	No
Sensible a metales y líquidos	Sí	Sí	No	Sí	No	Sí	Sí	No	Sí
Admite lecturas simultáneas	Sí	Sí	Sí	Sí	No	Sí	Sí	No	n/a
Tag admite reescrituras	Sí	Sí	Sí	Sí	n/a	Sí	Sí	No	n/a
Tag requiere baterías	Sí	No	Sí	Sí	n/a	Sí	Sí	No	n/a
Requiere interacción con usuario	No	No	No	No	Sí	No	No	Sí	No
Sirve para ubicación dentro de edificios	Sí	Sí	Sí	Sí	No	Sí	Sí	No	No
Sirve para ubicación en espacios abiertos	Sí	No	No	No	No	s/d	s/d	No	Sí
Sirve para identificación	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	No

Tabla 2.2: Resumen de tecnologías estudiadas: a) RFID activo; b) RFID pasivo; c) Ultrasonido; d) Bluetooth; e) Biometría; f) WiFi; g) Beacon; h) Código de barras; i) RTLS basado en GPS. n/a: no aplica; d/a: depende de la aplicación; s/d: sin datos.

2.5. Seguridad informática

El prototipo desarrollado maneja información sobre pacientes, visitantes y personal del hospital. Cada uno cumple un rol determinado dentro de dicha institución, y hay determinadas acciones que podrá o no realizar. Por ejemplo, un visitante no puede administrar medicación a un paciente, y un enfermero no puede preparar una unidosis. A su vez, se debe mantener la confidencialidad de la información sobre los pacientes del hospital, como lo enmarca la Ley N.º 18.335 [57]. Es necesario, entonces, incluir una capa de seguridad al sistema, lo que implica manejar algunos conceptos básicos (definidos en [58]):

Autenticación: consiste en asegurar quién realiza una determinada acción. Las formas típicas de autenticación pueden realizarse a través de algo que el sujeto conoce (por ejemplo una contraseña),

algo que el usuario posee (como una tarjeta de identificación), alguna característica biométrica (iris, huella dactilar, etc.), dónde se encuentra, o qué está haciendo.

Autorización: es el proceso de determinar, luego de que un usuario se autentica, si puede realizar la acción que pretende.

Control de acceso basado en roles (RBAC): a un usuario dado se le asigna uno o más roles, cada uno de los cuales tiene asociado un conjunto de procedimientos. El usuario podrá ejecutar las operaciones del rol que esté desempeñando en ese momento.

Conftdencialidad: implica evitar que sujetos no autorizados tengan acceso a cierta información, o incluso sepan de su existencia.

Integridad: ningún usuario del sistema, autorizado o no, debería poder modificar información de forma tal que quede corrupta. Por otro lado, ningún tercero debería ser capaz de alterar la información que el sistema envía a uno de sus usuarios, o viceversa, sin que pase desapercibido.

No repudio: proveer evidencia irrefutable de que una acción ocurrió.

Single Sign-On (SSO, o inicio de sesión único): para poder acceder a las funcionalidades de un sistema distribuido, un usuario normalmente necesita autenticarse en cada componente. Esto podría implicar tener que ingresar en cada uno su nombre de usuario y contraseña. Un servicio SSO provee al sistema la capacidad de compartir la información de autenticación entre cada componente.

Servidor de SSO: sistema que implementa el servicio SSO.

Salt: información aleatoria que puede ser pública, y se utiliza para mitigar el riesgo de que un atacante pueda obtener información privada [59, 60].

Capítulo 3

Sistemas existentes

El presente capítulo describe aplicaciones académicas y comerciales relevadas que más se aproximan a los requerimientos planteados por los expertos de dominio. Luego presenta el contexto nacional en cuanto a utilización de estas tecnologías.

3.1. Desarrollo académico

A continuación se describen desarrollos académicos existentes en el área de ubicación de pacientes y administración segura de medicación.

RFID pasivo

En [2] se describe un sistema para ubicación de pacientes en hospitales que utiliza RFID pasivo. Se coloca una antena de cada lado de una puerta. La secuencia de señales recibidas se utiliza para determinar el sentido de circulación de la persona a través de ella. Cuando un paciente entra o sale se registra su ubicación, tiempo de llegada y de partida. Los datos capturados son analizados, y ante eventos anormales se genera una alerta por SMS que notifica la situación al personal. Las funcionalidades soportadas son:

- Seguimiento y ubicación de personas.
- Control de cantidad de personas en un lugar.
- Alerta sobre pacientes en zonas inhabilitadas o permanencia en lugares por períodos no permtidos.

Por otro lado, en [1] se propone un sistema que utiliza UHF pasivo para la ubicación de aparatos e instrumental médico, y HF pasivo para la identificación de pacientes. Los objetos son etiquetados y los lectores colocados en puertas del hospital y zonas de interés. Las situaciones anómalas, como las que se dan cuando los objetos son movidos fuera de sus áreas de trabajo o salen del hospital sin autorización, se definen utilizando reglas de control. Cámaras en puertas permiten tomar fotos de las personas en el lugar de los hechos cuando se detecta una situación de riesgo.

La evaluación de la tecnología UHF en el trabajo citado devolvió resultados no satisfactorios. Entre los problemas detectados se encuentran:

- Tags que están cerca uno del otro no pueden ser leídos.
- La orientación del tag tiene un gran impacto en las lecturas.
- Un tag en contacto con el cuerpo o metales no puede ser leído.

RFID activo

El sistema propuesto en [61] utiliza tecnología RFID activa. Cada paciente cuenta con un tag activo embebido en su pulsera de identificación. Entre la información almacenada se encuentra la identificación, una nota sobre la condición del paciente y otros datos relevantes de su historia clínica.

Como los lectores RFID son costosos, para reducir la cantidad requerida emplean generadores de campos distribuidos en lugares estratégicos dentro del hospital. Un generador de campo es un dispositivo encargado de despertar tags activos que circulan a su alrededor [62]. Cuando un tag es activado, transmite su información a los lectores incluyendo el identificador del generador que lo despierta. La posición de los pacientes es calculada utilizando algoritmos de posicionamiento basados en la RSSI que llega del tag al lector, y en los identificadores de los generadores.

Ultrasonido

Bat activo [63] es un sistema que consta de un conjunto de transmisores móviles, receptores fijos en lugares conocidos, y un emisor central de radiofrecuencia en la banda de 418 MHz. El emisor central difunde una señal a los demás elementos del sistema, pero indicando que solo uno de los transmisores debe actuar ante ella. Cuando dicha señal llega al transmisor destino, éste emite un pulso de ultrasonido de 40 KHz que es escuchado por los receptores colocados en el edificio, sin interferir con el instrumental médico, dado que las bandas de frecuencia de trabajo no se superponen. Como estos últimos también recibieron la señal de la central, pueden calcular la distancia al transmisor a partir del retardo del pulso ultrasónico. Los distintos retardos son analizados por un servidor centralizado para estimar la ubicación de cada transmisor, con una precisión en el orden de los 9 cm [64].

Por otro lado, el sistema Cricket [65], al igual que el anterior, consta de un conjunto de emisores y receptores de radiofrecuencia y ultrasonido. Los emisores se fijan en el edificio, mientras que los receptores se colocan en los objetos cuya ubicación se desea obtener. Los emisores difunden periódicamente un pulso de ultrasonido, seguido de una señal de radiofrecuencia. El pulso se utiliza para activar a los receptores, que al escucharlo se preparan para recibir los datos por radiofrecuencia, y calcular distancias en base a la diferencia de tiempo de llegada de esas dos señales. El pulso de ultrasonido no posee información, a diferencia de la señal de radiofrecuencia, que contiene, por ejemplo, el identificador del emisor. El receptor calcula sus coordenadas espaciales a partir de los datos obtenidos de los distintos emisores. Los pulsos de ultrasonido se emiten a 40 KHz, mientras que las señales de radiofrecuencia a 433 MHz. Aplican las mismas consideraciones sobre interferencia con equipamiento médico que en el caso anterior.

WiFi

En [66] se presenta un sistema de ubicación de pacientes que utiliza la RSSI en los puntos de acceso para estimar la posición. Aprovecha la infraestructura de red WiFi del hospital y los dispositivos de control añadidos a los pacientes, que usualmente están equipados con módulos WiFi y se utilizan para vigilar los signos vitales. Para los cálculos de la posición emplean la técnica de *fingerprinting* [67], que construye un mapa del ambiente con la distribución de las RSSI medidas desde diferentes APs. Cada punto del mapa corresponde a un vector de medidas de RSSI, y se asocia a una ubicación.

Esta técnica tiene como desventaja que la construcción del mapa de distribuciones demanda mucho tiempo y debe actualizarse si el ambiente es modificado.

3.2. Aplicaciones comerciales

En esta sección se presentan algunos de los sistemas comerciales investigados, identificándolos por las tecnologías que utilizan.

Ultrasonido

La capa de hardware consiste en emisores de ultrasonido, también llamados tags, adjuntos a personas (pacientes, personal) u objetos del hospital, que envían información a receptores colocados en paredes de las salas o pasillos. El método de funcionamiento no se corresponde con ninguno de los vistos en la sección 3.1, sino que en cada zona se ubica un receptor, el cual demodula las señales de ultrasonido que escucha, e indica a un servidor centralizado que en dicha zona se encuentran los tags identificados. Las funcionalidades mencionadas por el proveedor son [68]:

- Conocer en tiempo real la ubicación de entidades (pacientes, personal, equipamiento, etc.) que lleven consigo un emisor de ultrasonido. El nivel de precisión es de sala o zonas, si aquellas se subdividen con paredes que confinen las emisiones de los tags.
- Proveer infraestructura para que aplicaciones de terceros determinen si hubo contacto entre entidades, lo cual facilita el control de infecciones.
- Cancelar un llamado a enfermeros de forma automática, cuando detecta que hay uno atendiendo al paciente.
- Monitorizar las siguientes condiciones ambientales: temperatura, humedad, presión y movimiento.
- Controlar el tránsito indebido de entidades: permite alertar cuando un paciente, empleado, u objeto se encuentra en una zona no permitida.
- Detectar caídas de pacientes.

Los sensores de ultrasonido se colocan en las paredes de las salas que controlan, y se comunican con el servidor mediante WiFi. La figura 3.1 muestra un esquema de alto nivel que describe la arquitectura del sistema, presentada en [69]. Puede apreciarse que la capa de hardware es

reemplazable por otra tecnología, como RFID, código de barras, etc.

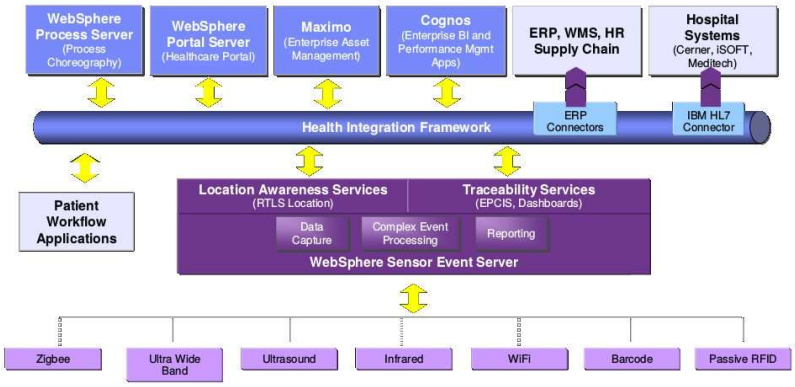


Figura 3.1: Arquitectura de la solución comercial basada en ultrasonido [69].

Se consultó sobre el despempeño de este sistema a un grupo de ingenieros a cargo de su instalación en un hospital del exterior. Citando palabras textuales: «...no estamos muy satisfechos con la tecnología de EEUU. Los tags son caros, y hemos tenido muy mala experiencia respecto a la durabilidad: fallan con mucha frecuencia, y es necesario reenviarlos a la fábrica de EEUU. La gran tasa de fallos nos ha significado no tener datos confiables para analizar. Estamos considerando migrar a otras tecnologías más baratas y confiables...».

WiFi

Se basa en una tecnología propietaria que trabaja sobre los protocolos 802.11b o g. Los tags transmiten mensajes de tipo *probe request* a los puntos de acceso, los cuales reenvían la información relevante (identificador del tag y RSSI) a un servidor centralizado del sistema [70]. Los mensajes *probe request* son los primeros en ser enviados por un nodo, previo a la conexión con un punto de acceso, por lo cual no se requiere que los tags estén autenticados en la red WiFi [44]. Si bien se aprovecha la infraestructura de conectividad inalámbrica existente en el edificio, las necesidades que cubre pueden ser muy distintas a las del sistema para su instalación [71].

Funcionalidades mencionadas [72]:

- Seguimiento y ubicación de personas y objetos.
- Reporte de historial de ubicaciones, utilización de objetos, entre otros.
- Alertas sobre ubicaciones o estados de tags.
- API para integración con sistemas de terceros.

UWB 1

Es un RTLS basado en UWB (6 a 7 GHz), que ubica las coordenadas espaciales de un tag con una precisión de 15 cm, según su proveedor [73]. Se basa en los métodos de *ángulo de llegada* y *diferencia de tiempos de llegada* para estimar la posición del tag [73, 74]. La duración de la batería en un tag se estima de cinco años [75]. Si bien se trata de un sistema diseñado para la industria, se

han hecho pruebas de desempeño en un entorno hospitalario [71,76]. La única funcionalidad con la que cuenta es la ubicación de tags en tiempo real.

UWB 2

Al igual que el sistema anterior, éste es un RTLS que permite ubicar tags en dos o tres dimensiones, y funciona en el rango UWB (de 3,1 a 4,8 Ghz, según el estándar IEEE802.15.4-2011 [77]). Se basa en los tiempos de llegada de señales a receptores fijos para calcular estimaciones sobre la posición del tag, con un error de hasta 30 cm [78]. En la referencia citada, el proveedor afirma que su sistema es hasta diez veces más preciso que los basados en la intensidad de señal WiFi o Beacon.

RFID activo

Cuenta con un RTLS basado en RFID activo con un protocolo propietario, y capacidad de detección de tags con precisión de zona [71,79], esto es, el sistema determina la zona en la que se encuentra pero no el punto exacto dentro de ella. De sus funcionalidades relevadas, las de interés son [80]:

- Ubicar en tiempo real objetos y personas.
- Detectar equipamiento por ubicación.
- Alertar si un objeto o persona sale de una zona asignada.

RFID activo 2

Se basa en RFID activo a 915 MHz con protocolo propietario, y determina la ubicación espacial en dos dimensiones de los tags [71, 81]. Las funcionalidades relevadas de este sistema son [82]:

- Ubicar en tiempo real pacientes, empleados y objetos.
- Alertar ante situaciones de error de medicación.
- Notificar al personal médico si un paciente entra a una sala de operación.
- Gestionar automáticamente las camas: lleva un control de las camas ocupadas y libres.

WiFi y RFID

Este sistema fue ofrecido al Hospital Maciel. Integra varias tecnologías, entre ellas RF a 433,92 MHz [83], WiFi [84], ultrasonido de 40 KHz [85], y biometría (detección de patrones en el iris) para la identificación de personal [86]. El costo de instalación estimado por el proveedor, según comentó personal del hospital, rondaba entre los 200.000 y 500.000 dólares en noviembre de 2015 (costo promedio de 3.000 a 7.700 dólares por sala). Las funcionalidades del sistema son [87]:

- Permitir definir eventos en tiempo de ejecución.
- Prevenir el robo de recién nacidos y controlar la correspondencia con sus respectivas madres.

- Controlar la fuga de pacientes.
- Ubicar en tiempo real personas y objetos del hospital a nivel de zona.
- Enviar diferentes eventos, según el patrón de presión del botón de llamada ubicado en los tags de pacientes.
- Monitorizar temperatura y humedad ambiental.
- Alertar sobre caída de pacientes.
- Controlar cumplimiento de higiene de manos.

Sistema automático de dispensación de unidosis

Existen varios sistemas automáticos de armado y dispensación de unidosis. El más destacado, dentro de los que se investigaron para el presente trabajo, es el presentado por [88]. Su característica principal es la eliminación del factor humano en el proceso de armado de unidosis, lo cual reduce el riesgo de error de medicación. Este sistema es complementario al presentado en el informe, ya que aborda una problemática fuera del alcance definido para este proyecto.

El sistema obtiene la lista de medicamentos que componen la unidosis para un paciente dado a través de un sistema digital de receta médica; los obtiene de un depósito interno y los embala, identificando a cada uno con un código de barras. El conjunto de medicamentos queda unido por una cinta plástica, junto con una placa que contiene la lista completa, como muestra la figura 3.2.

Figura 3.2: Unidosis preparada por sistema automático [88].

Comparativa de sistemas de ubicación

En este apartado se presenta una prueba de desempeño de algunos de los sistemas para ubicación de pacientes y objetos presentados anteriormente (todos menos UWB 2 y WiFi + RFID), realizada en el marco de un trabajo de investigación de la Universidad del Centro Médico de Maryland [71,76]. El mismo estima la tasa de error de los sistemas puestos a prueba en varias situaciones como, por ejemplo, tags cubiertos por otros objetos o cerca del límite entre salas, habitaciones con puertas abiertas, cerradas, etc.

La figura 3.3 muestra un esquema del lugar físico utilizado para las pruebas. El mismo consta de dos salas de operación (*OR1* y *OR2*) y tres habitaciones entre ellas. Los puntos rojos indican los lugares donde se colocaron tags. Las tablas 3.1 y 3.2 detallan los porcentajes de error en la lectura de tags cometidos para cada escenario y los porcentajes de error para cada zona, respectivamente. Una descripción más detallada se puede encontrar en [76].

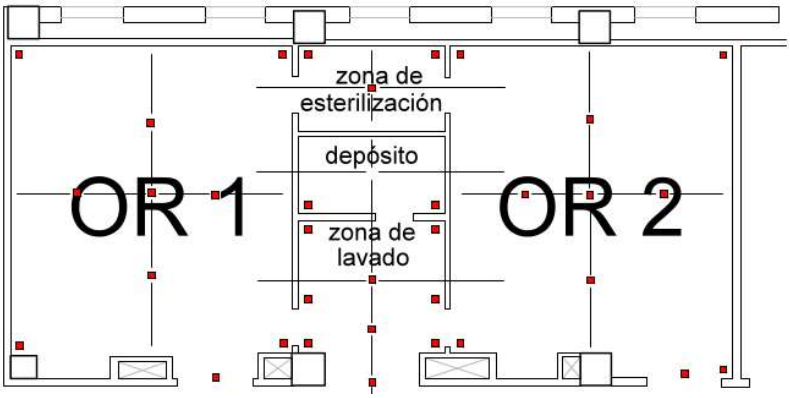


Figura 3.3: Plano de las salas de operación donde se probaron sistemas comerciales [71].

Sistema	a	b	c	d	e	f	g	h	i
Ultrasonido	2,9	20,6	0	26,5	0	2,9	0	n/a	n/a
WiFi	20,6	38,2	55,9	55,9	70,6	82,4	73,5	n/a	68,8
UWB	40	20	20	10	0	n/a	10	n/a	0
RFID activo	50	47,1	44,1	50	64,7	47,1	47,1	47,1	43,8
RFID activo 2	35,3	35,3	17,6	17,6	32,4	22,2	23,5	n/a	23,5

Tabla 3.1: Porcentajes de error de lectura para varios escenarios: a) puertas abiertas; b) puertas cerradas; c) puertas cerradas y manta cubriendo el tag; d) puerta abierta y manta cubriendo el tag; e) tag en bolsa de solución salina, expuesto; f) tag oculto por bolsa de solución salina; g) tag adjunto al exterior de una caja de instrumental; h) tag en interior de cada de instrumental; i) tag en camilla [76].

La tabla 3.3 muestra los costos de instalación en el año 2006 de cada sistema en la sala OR2.

Sistemas existentes en Uruguay

Del relevamiento realizado para el presente trabajo no se encontraron centros de atención que utilicen sistemas de ubicación de este tipo. Varios hospitales utilizan pulseras de identificación con

Sistema	OR1	Depósito	Lavado	OR2
Ultrasonido	2,9	0	22,2	6,7
WiFi	24,7	62,9	78,5	70,1
RF e IR	1,3	100	100	2,5
UWB	n/a	n/a	n/a	14,9
RFID activo	5,7	95	85,1	41,4
RFID activo 2	1,3	42,4	64,1	12

Tabla 3.2: Porcentajes de error de lectura agregados por sala. [76]

Sistema	Costo para OR2 (USD)
Ultrasonido	11000
WiFi	10000
RF e IR	16500
UWB 1	20000
RFID activo	13500
RFID activo 2	4000

Tabla 3.3: Costos de adquisición e instalación de varios sistemas comerciales. Datos del año 2006 [76].

códigos de barras. En el caso del Hospital Maciel, las pulseras de los pacientes cuentan con un código QR con información básica, pero no se le da utilidad actualmente.

Capítulo 4

Análisis del problema

En este capítulo se describen los requerimientos relevados a lo largo del análisis del proyecto, y la solución final planteada.

4.1. Requerimientos funcionales

A continuación se listan los requerimientos funcionales para el sistema, relevados durante las reuniones con expertos de dominio e investigación de la operativa hospitalaria:

1. **Asignación de cama**: el sistema debe ser capaz de indicar la cama asignada a un determinado paciente.

2. **Identiftcación de objetos y personas**: cada objeto o persona del hospital debe ser identificable por el sistema.

3. **Alertas**: el sistema debe emitir alertas ante situaciones de riesgo, o impedir que ocurran. Las situaciones de riesgo que debe contemplar son:

   a) Pacientes en camas no asignadas.

   b) Medicación incorrecta a un paciente.

   c) Pacientes que ingresan o salen de una sala sin previsión. Por ejemplo, un paciente ingresa a una sala de internación no asignada.

4. **Registro de uso**: las acciones de los usuarios, o eventos registrados por el sistema, deben ser recuperables.

5. **Estrategias de alertas conftgurables**: cada usuario podrá elegir qué estrategia debe utilizar el sistema (por ejemplo correo electrónico, notificaciones en dispositivo móvil, etc.) para enviarle alertas. También podrá elegir la prioridad de la alerta (alta o baja).

6. **Modiftcación del conjunto de estrategias de alerta**: el sistema debe permitir quitar o agregar estrategias de alerta sin necesidad de reiniciarlo.

7. **Modiftcación del conjunto de eventos a alertar**: un usuario debe poder agregar nuevas

situaciones (eventos) por los que el sistema debe emitir alertas, sin necesidad de reiniciar el sistema.

8. **Soporte de idiomas**: el sistema debe mostrar los textos correspondientes a los registros que se estén visualizando en distintos idiomas, según el usuario que haya iniciado sesión. Dichos idiomas deben poder ser previamente registrados en el sistema, si aún no existen, y los textos ser creados por el usuario al momento de ingresar el registro correspondiente.

9. **Integración con base de datos del hospital y MPI**: el sistema debe poder tomar datos de pacientes de la base de datos existente en el hospital, y configurarse para almacenar u obtener registros de pacientes de un MPI.

10. **Administración de medicación**: el sistema debe asistir al funcionario de salud en la administración de medicación al paciente, indicando si tomó una unidosis que no es para este último, o si es la correspondiente al turno en que se encuentra. Además, debe llevar un registro de qué medicamentos de esa unidosis se administraron al final del procedimiento y cuáles no.

11. **Integración con repositorio XDS**: el sistema debe registrar el acto de administración de medicación en un repositorio XDS configurable, utilizando el formato CDA.

4.2. Requerimientos no funcionales

A continuación se listan los requerimientos no funcionales del sistema.

1. **No interferencia con procedimientos médicos**: el sistema no debe interferir bajo ninguna forma con los procedimientos médicos realizados. Esto incluye sensores que obstruyen la circulación o dificultan el trabajo del personal.

2. **No interferencia con equipos médicos**: los emisores de radiofrecuencia que posiblemente sean utilizados no deben interferir con el equipamiento electrónico utilizado, o perjudicar a quienes estén expuestos a ellos.

3. **Resistencia al maltrato**: los dispositivos electrónicos deben ser resistentes a condiciones adversas. En particular, puede ocurrir que los equipos deban ser esterilizados periódicamente a altas temperaturas.

4. **Independencia a particularidades de las camas**: existen gran variedad de modelos de cama en el hospital. El sistema debe poder adaptarse a todos ellos, esto incluye poder etiquetar o montar sensores con facilidad.

5. **Base de datos**: El prototipo debe utilizar el DBMS MySql.

6. **Utilización de infraestructura existente**: se deben utilizar los servicios o datos ya existentes en el hospital. Se pretende así evitar el desarrollo de funcionalidades ya implementadas o duplicación de datos, que dan lugar a problemas de sincronización e inconsistencias, y perjudica la mantenibilidad.

7. **Latencia de alertas**: el intervalo de tiempo desde que una determinada situación debe ser alertada hasta que los usuarios interesados reciben una notificación no debe ser mayor a 10

segundos.

8. **Costo de la solución**: la solución debe desarrollarse con un presupuesto de USD 1500. Esto incluye compra de equipos electrónicos y cualquier elemento necesario para la realización del prototipo.

43. Modelo de dominio

A continuación se describen los conceptos más importantes de la realidad estudiada. Entiéndase *concepto* como cualquier entidad o elemento que pertenece al dominio del problema. La figura 4.1 muestra el modelo de dominio, donde para cada entidad se listan sus atributos y relaciones.

Zone: zona o lugar del hospital. Tiene un nombre, descripción y categoría. Una zona pude ser una sala, cama, puerta de acceso, block quirúrgico, etc.

ZoneCategory: Categoría que permite agrupar las zonas. Pueden ser pasillos, salas, puertas, etc.

Room: sala del hospital. Tiene un conjunto camas. Deriva de *Zone*.

Bed: cama del hospital. Se identifica por un número, usualmente fijo a la pared. Define una zona o lugar dentro de la sala que será ocupado físicamente por un objeto cama. Deriva de *Zone*.

Entity: entidad (personas u objetos) que circula por el hospital, identificada por un tag RFID y/o un código de barras. Tiene un conjunto de permisos que habilitan su circulación por determinadas zonas del hospital.

ZonePermission: permiso para circular por zonas del hospital. Tiene un período de validez, definido por una fecha de expiración, hora de comienzo y fin. Los permisos pueden estar activos o inactivos independientemente de su fecha de validez.

HospitalObject : objeto del hospital. Tiene un nombre, descripción y categoría. Puede ser equipo médico, cama, almohada, dispositivo móvil, etc. Deriva de *Entity*.

Person: persona que tiene un vínculo con el hospital. Puede ser paciente, empleado o visitante. Tienen un nombre, apellido, documento de identificación y opcionalmente una foto identificatoria. Tiene asociado un idioma, usualmente su lengua materna. Deriva de *Entity*.

Patient: paciente del hospital. Tiene asignadas una sala y una cama. Interesa saber su fecha de nacimiento, género y observaciones del estado clínico actual. Deriva de *Person*.

Visitor: Visitantes de los pacientes internados. Tienen un conjunto de visitas realizadas. Deriva de *Person*.

Visit: Visitas realizadas a los pacientes. Tienen una fecha de visita.

Empleado: Empleados del hospital. Interesa saber su función: enfermero, médico, farmacéutico, personal de seguridad, etc. Deriva de *Person* .

Unidosis: empaquetado con la medicación de un paciente para determinado turno. Los farmacéuticos son los responsables de armarla y los enfermeros de administrarla. Interesa saber la fecha

en la que fue administrada. Deriva de *Entity*.

MedicineUnidose: medicina empaquetada en unidosis. Se registra la cantidad empaquetada y si fueadministrada.

Medicine: medicamento del hospital. Tiene un nombre y descripción. Deriva de *Entity*.

EventType: evento definido que se desea detectar. Tiene un nombre, descripción y se clasifica en simple o complejo. Para cada evento se registran las instancias u ocurrencias detectadas.

PrimitiveEventType: evento simple definido en el sistema. Es generado por actores externos al sistema. Deriva de *EvenType*.

BasicEventType: evento complejo definido en el sistema. Se define aplicando operadores sobre eventos simples y otros complejos. Deriva de*EvenType*.

Event: ocurrencia de un evento definido. Se clasifica en ocurrencia de evento simple o complejo.

BasicEvent: ocurrencia de un evento simple. Deriva de *Event*.

ComplexEvent: ocurrencia de un evento complejo. Deriva de *Event*.

Subscription: las personas pueden suscribirse a eventos complejos que sean de su interés. Una suscripción tiene una prioridad y un método de notificación. Cuando un evento es detectado se notifica a todas sus suscriptores a través del método que hayan elegido.

Notiftcation: notificación de los eventos que hayan ocurrido. Interesa saber si la notificación fue leída por eldestinatario.

AccountInfo: las personas pueden tener un usuario para acceder a la aplicación. Cada usuario tiene un nombre, contraseña y un conjunto de roles.

Rol: rol del sistema. Tiene asignado un conjunto de permisos y suscripciones por defecto. Un usuario podrá ejecutar acciones de acuerdo a su rol y será suscrito automáticamente a los eventos por defecto.

Permission: permiso del sistema que habilita a un usuario a realizar determinada acción.

DefaultSubscription: suscripción por defecto de un rol. Incluye prioridad y método de notificación.

Language: idioma soportado por el sistema.

Text: texto traducido en distintos idiomas. Será desplegado a los usuarios de acuerdo a su idioma.

44 Casos de uso

A continuación se describen brevemente los casos de uso relevados.

Nuevo Idioma

Permite crear un idioma nuevo. Los campos del sistema que admitan entradas en varios idiomas lo incluirán.

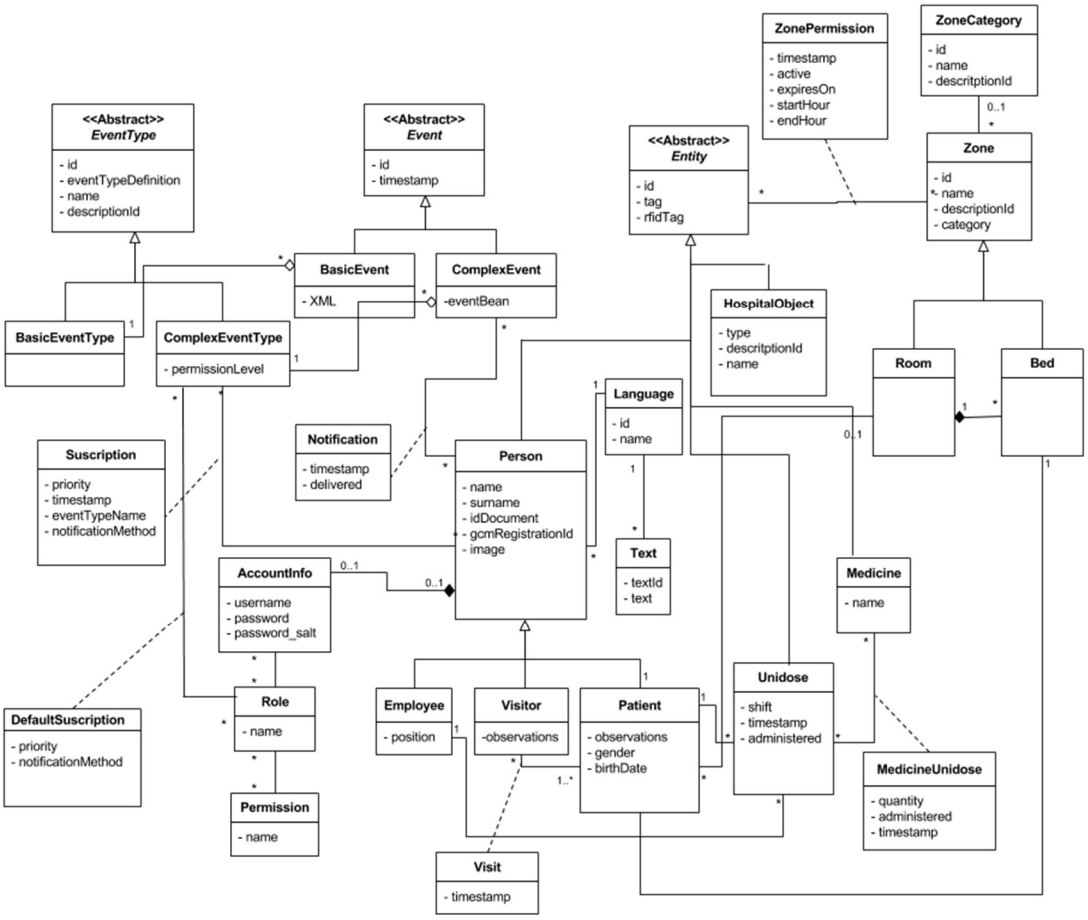


Figura 4.1: Modelo de dominio

Crear nueva categoría de zona

Permite dar de alta una nueva categoría o tipo de zona en el sistema. Se utilizan para agrupar las zonas. Se debe ingresar el nombre y descripción de la categoría.

Crear nueva zona

Permite dar de alta una nueva zona en el sistema. Se debe ingresar el nombre, descripción y categoría.

Nuevo tipo de objeto

Permite crear nuevos tipos de objeto (*monitor*, *silla*, *cesto de basura*, etc.) para clasificarlos.

Nuevo empleado, paciente, visitante u objeto

Permite dar de alta una nueva entidad (persona u objeto) en el sistema. Las personas son empleados, pacientes o visitantes. Se debe ingresar obligatoriamente el nombre, apellido y documento de identidad.

Por otro lado, los objetos pueden ser de cualquier tipo o categoría, y requieren de un nombre y código de barras.

Actualizar empleado, paciente, visitante u objeto

Permite actualizar los datos de una entidad registrada en el sistema. Primero, el usuario debe buscarla, ingresando en el caso de las personas, el documento de identidad, y en los objetos, el código de barras. Una vez seleccionada la entidad, el usuario ingresa los nuevos datos.

Agregar o actualizar zona permitida a entidad

Mediante estos casos de uso se asignan o modifican permisos a una entidad para estar en las zonas indicadas. Se puede establecer un rango de horas del día, o hasta qué fecha es válido un permiso.

Identiftcación de persona, objeto o unidosis

Estos casos de uso permiten a un usuario obtener información sobre una entidad (persona u objeto) determinada. Si se trata de una persona, permite saber el nombre, apellido y otra información básica dependiendo de si es un paciente, visitante o empleado.

Ubicar persona u objeto

Permite a un usuario del sistema ver la última ubicación conocida de una entidad determinada.

Actualizar ubicación

Este caso de uso se dispara de dos maneras: una de ellas es sin necesidad de que la entidad participe activamente, cuando el sistema recibe un evento de identificación de la misma junto con la información de su ubicación (por ejemplo, cuando un sensor de RFID lee su tarjeta de identificación); y la otra forma es realizable solo por un usuario con permisos suficientes, y consiste en la actualización manual a través de la interfaz móvil.

Crear rol nuevo

Mediante este caso de uso se pueden crear nuevos roles del sistema, y dar permisos. Dichos roles podrán ser asignados a un usuario en los casos de uso *dar acceso al sistema* o *actualizar acceso al sistema*.

Además se especificará un conjunto de suscripciones por defecto a tipos de eventos complejos (ver caso de uso *configurar alertas*). De este modo, cuando se asigne un rol determinado a un usuario nuevo en *dar acceso al sistema*, además será suscripto a los tipos de evento especificados aquí.

Dar o actualizar acceso al sistema

Estos casos de uso permiten dar o actualizar el acceso como usuario a una persona registrada del sistema. Se asignan uno o más roles, cada uno de los cuales tiene un conjunto de permisos para realizar ciertas acciones.

Enviar evento simple

Este caso de uso permite enviar un evento simple al sistema. Los eventos recibidos son registrados en el log de eventos y enviados al procesador de eventos para continuar con el procesamiento. Los actores principales son las entidades que participan en el evento, por ejemplo, cuando se envía un evento de identificación, el actor corresponde a la entidad identificada.

Detectar evento complejo

El procesador de eventos se encarga de la detección de eventos complejos. Al detectarse uno, el evento es registrado y se crean las alertas o notificaciones que serán enviadas a los usuarios suscritos.

Notiftcar evento

Cuando el sistema detecta la ocurrencia de un evento complejo, envía alertas a los usuarios suscriptos. Cada alerta es enviada en el idioma del usuario mediante el método de notificación seleccionado.

Conftgurar alertas

Un usuario del sistema verá una lista de tipos de evento por los que puede recibir alertas. Mediante este caso de uso elige a cuáles se suscribirá, qué método de notificación utilizará, y con qué prioridad debe recibir las alertas.

Administrar unidosis

El caso de uso asiste al enfermero en la administración de la unidosis. Primero se identifica al paciente leyendo su código de barras y, el sistema despliega su información. Luego se identifica la unidosis leyendo su código QR, y el sistema verifica la correspondencia entre la unidosis y el paciente. En caso de ocurrir un error se muestra un mensaje de alerta, y en caso contrario, la información de la unidosis. El enfermero irá seleccionando en la aplicación los medicamentos administrados, y al finalizar se registrará el acto creando un documento CDA con la información detallada.

Censo de camas

Diariamente se realiza un censo, el cual consiste en un recorrido por las camas del hospital para identificar a los pacientes que las ocupan. El caso de uso permite al usuario identificar a un paciente y una cama leyendo sus códigos de barras. El sistema lleva el registro de las camas ocupadas, y de quién las ocupa.

45. Solución propuesta

La solución propuesta aborda, entre otras, dos problemáticas planteadas por el hospital: seguridad en la administración de medicación, y alerta sobre situaciones de riesgo.

4.5.1. Seguridad en la administración de medicación

De las entrevistas con expertos de dominio se evidenciaron ciertas situaciones que pueden inducir a error en la administración de medicación a un paciente:

- Las unidosis de varios pacientes se envasan en una misma bolsa plástica, separadas con una máquina selladora manual. Dicha separación se puede romper, y como resultado los medicamentos de dos pacientes pueden mezclarse. Si bien cada unidosis lleva consigo una lista de medicamentos que contiene, se pueden confundir.
- Muchas veces los medicamentos de una unidosis no llegan a ser administrados, por diversas

razones. Esos medicamentos quedan en la sala de internación, en depósitos de lo que el personal llama "medicación sobrante". Cuando, en ocasiones, una unidosis no tiene alguno de los medicamentos prescriptos al paciente (por falta de inventario en farmacia, por ejemplo) el enfermero puede decidir tomarlo del depósito de medicación sobrante, si es que está disponible. Este es un punto de posible introducción de error, porque puede confundir medicamentos, o creer equivocadamente que dos drogas diferentes sirven para el mismo fin.

- Existen medicamentos cuyos nombres suenan parecido, o son de apariencia similar (medicamentos LASA, por sus siglas en inglés), lo cual propicia las situaciones de error en el caso de los puntos anteriores, o mientras la unidosis se está preparando en farmacia.

- Los medicamentos de farmacia se almacenan en estanterías. Al momento de preparar una unidosis, un farmacéutico puede confundir medicamentos indicados en la prescripción, o tomar uno de un estante equivocado.

- Para minimizar la cantidad de recorridas en la sala, algunos enfermeros colocan en una misma bandeja los medicamentos de más de un paciente, amontonados y sin ninguna separación física, pudiéndo mezclarlos.

Además, no existe una cultura establecida de reporte de incidencias (errores o situaciones en las que casi se incurre en ellos), lo que dificulta la visibilidad sobre los procesos del hospital, para poder analizarlos y lograr eventuales mejoras.

Se propone, como solución a la problemática, el reemplazo de las bolsas plásticas por envases cerrados, cada uno conteniendo la unidosis correspondiente a un paciente para un turno determinado. Los mismos se sellan con etiquetas de seguridad con un código QR de identificación impreso y datos básicos (como nombre y apellido), colocadas de forma tal que para abrir el envase sea necesario romperlas, y con ellas el código. El sistema planteado debe encargarse de corroborar la correspondencia entre unidosis, paciente y turno actual.

El funcionario que realice la administración de la unidosis utilizará una aplicación en un dispositivo móvil para corroborar la correspondencia, y como respuesta recibirá la lista de medicamentos que contiene el envase, o un mensaje indicando que el paciente no es el correcto. En el primer caso, el funcionario debe abrir el envase, administrar los medicamentos, e indicar en la aplicación móvil cuáles administró. La administración de medicación se registrará en un respositorio XDS. Además, la aplicación mostrará información detallada sobre el paciente y las medicinas que se estén administrando; esto incluye imagen del paciente para facilitar su identificación, observaciones, dosis a administrar, etc.

El sistema determinará si hay medicación sobrante, y en ese caso puede notificar al personal de farmacia (como se explica en la sección 4.5.2) para prevenir que sean utilizados sin consentimiento de farmacia.

Existe una aplicación para preparar unidosis a partir de prescrpiciones médicas obtenidas, en la cual se proponen los siguientes cambios, a ser realizados por sus desarrolladores:

- Solicitar la lectura de código de barras en los estantes de cada medicamento (o idealmente en sus envases), y verificar la correspondencia con la prescripción, al momento de preparar una unidosis.

- Imprimir la etiqueta de seguridad con el código QR de identificación, nombre, apellido, y turno en que debe administrarse.

- Registrar la unidosis en el sistema presentado en este trabajo, a través de una interfaz desarrollada para tal fin, indicando medicamentos que contiene, para qué paciente es y en qué turno debe administrarse.

Es importante señalar que la solución propuesta se restringe a un escenario simple: no se contemplan aspectos como medicación que no se incluye en la unidosis (como la que debe permanecer refrigerada, o se administra individualmente), aquella que se administra según estado del paciente (si siente dolor, mareo, etc.), o que es preparada en sala (como dilución en suero).

4.5.2. Detección de situaciones de riesgo

Uno de objetivos del sistema es la detección y notificación de eventos, entre ellos, los que representan situaciones de riesgo para el hospital. Para esto, el sistema debe poder identificar y ubicar a las entidades dentro del hospital (empleados, pacientes, visitantes, unidosis, medicinas y objetos físicos). La solución propuesta combina el uso de tecnologías de código de barra y RFID.

Todas las entidades que necesiten ser rastreadas tendrán adherido un código de barras y tag RFID. Por otro lado, al ingreso, los pacientes también recibirán una pulsera con código de barras, que almacenará información básica de identificación. Un conjunto de lectores RFID fijos serán desplegados en entradas de salas, hospital y pasillos, que recibirán las señales de los tags que circulen alrededor.

Los datos recolectados por sensores y lecturas de códigos serán analizados por un procesador de eventos, que alertará en caso de que un evento o situación de riesgo sea detectado. Entre los eventos de interés se encuentran: personas en lugares no autorizados, unidosis administradas a pacientes incorrectos o fuera de tiempo, unidosis no administradas, pacientes en camas incorrectas y medicación sobrante.

Por otro lado, los empleados tendrán un dispositivo móvil, equipado con lector de código de barras, donde ejecutará una aplicación móvil. Se utilizará para identificar y obtener información, recibir notificaciones de alerta de acuerdo a prioridades establecidas por el usuario y para la ubicación de pacientes u objetos. También servirá de asistente a los enfermeros en el proceso de administración de unidosis, para prevenir errores de medicación.

Las notificaciones ante eventos o situaciones de riesgo serán recibidas por usuarios que estén suscritos a ellas. El sistema permitirá que éstos se suscriban especificando una prioridad (baja o alta) y método de notificación (mail o notificación al dispositivo móvil). Por ejemplo, cuando una entidad se encuentre en una zona no autorizada, el personal de seguridad recibirá una alerta detallando el hecho. Por otro lado, la farmacia del hospital podrá recibir notificaciones cada vez que ocurra un error de administración o haya medicación sobrante. Cuando una unidosis no es administrada por olvido, el personal de enfermería recibirá un recordatorio.

Los eventos o situaciones de riesgo se definirán utilizando un lenguaje de especificación de eventos, que será lo suficientemente expresivo para permitir definir nuevos eventos o condiciones de alerta, de acuerdo a los objetivos de gestión segura del hospital.

Capítulo 5

Arquitectura del sistema

En este capítulo se presenta la arquitectura del sistema. La primera sección introduce una vista de alto nivel, y las restantes siguen la representación del modelo de vistas *4+1* [89].

5.1. Descripción de alto nivel

El prototipo desarrollado se divide físicamente en siete módulos distribuidos: *Back-end*, *Front-end*, *Mobile application*, *Database*, servicios de *Salud.uy*, *RFID sensor*, y *Middleware sensor*. Además, las estrategias de notificación desarrolladas utilizan servicios de terceros. La figura 5.1 muestra una esquematización de alto nivel de la arquitectura del sistema. En ella pueden observarse los módulos involucrados, algunos aspectos de su organización interna, y cómo interactúan.

Algunas características a destacar:

- El *Back-end* expone sus servicios a través de una interfaz RESTful.
- Los módulos *Database* y *Salud.uy* corresponden a sistemas desarrollados por terceros.
- Los sensores de RFID se comunican con una API desarrollada por el proveedor, en un protocolo propietario. Dicha API fue desarrollada para el sistema operativo Microsoft Windows.

5.2. Vista lógica

La vista lógica describe principalmente la estructura de soporte a los requerimientos funcionales [89]. La figura 5.2 muestra las principales clases del sistema.

EventEngine

Define la interfaz del motor de eventos, implementada por la clase *EventProcessor*. Esta clase es la responsable de procesar los eventos primitivos, registrarlos en el log de eventos y detectar eventos complejos que representan situaciones de riesgo. Cuando un evento complejo es detectado,

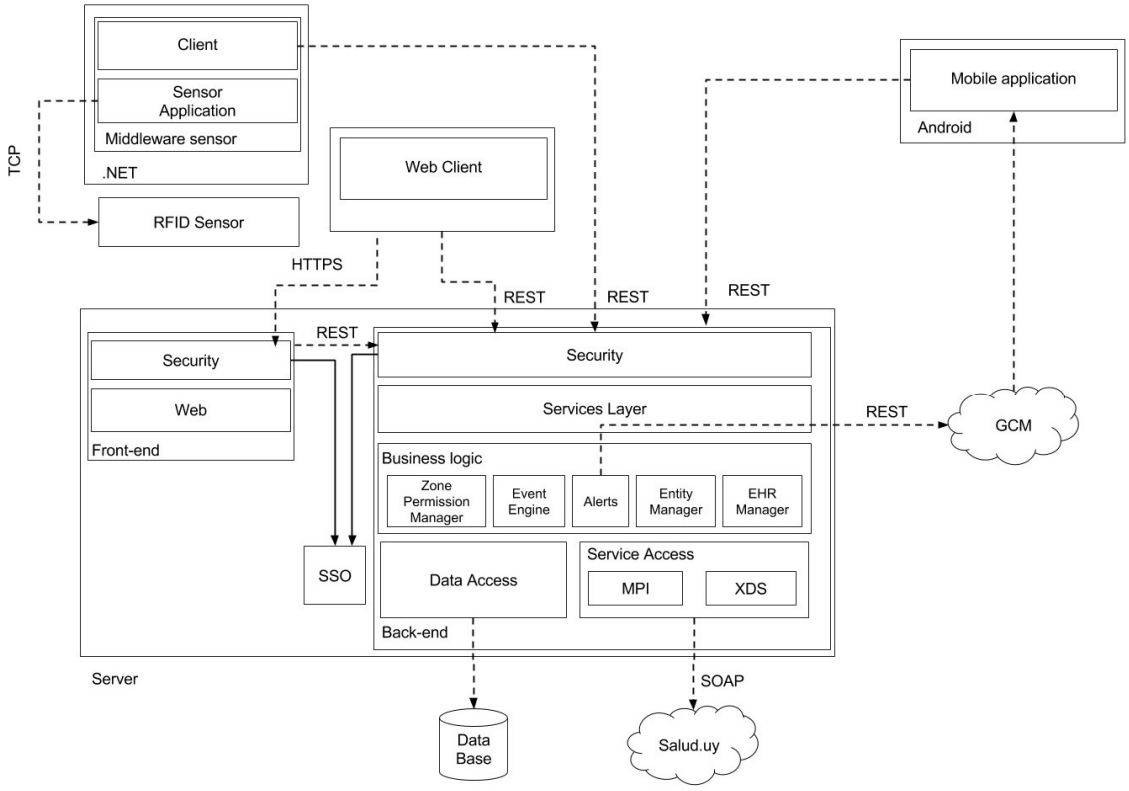


Figura 5.1: Esquema de alto nivel de la arquitectura

invoca a sus listeners para ejecutar las acciones correspondientes.

Listener

Interfaz implementada por los responsables de ejecutar acciones tras la detección de eventos complejos. Un evento complejo puede tener asociado uno o varios listeners que serán invocados cada vez que el evento sea detectado. El prototipo implementa dos listeners: *UpdatePosition*, encargado de actualizar la posición de una entidad cada vez que se recibe un evento de identificación, y *SubscriptionManager*, responsable de notificar la ocurrencia del evento a los interesados.

AlertStrategy

Define la interfaz que deben implementar las estrategias de notificación. Los eventos son notificados por el *SubscriptionManager* utilizando distintos métodos o estrategias de notificación. El prototipo implementa la estrategia *MailStrategy* para notificaciones por mail y la estrategia *NotificationStrategy* para notificaciones al dispositivo móvil. El sistema admite la incorporación de nuevas estrategias en tiempo de ejecución.

Engine, Subscription, Information y Alerts Model

Definen interfaces para el acceso a la persistencia. *EngineModel* permite registrar y consultar eventos primitivos y complejos. *SubscriptionModel* permite registrar y consultar suscripciones de

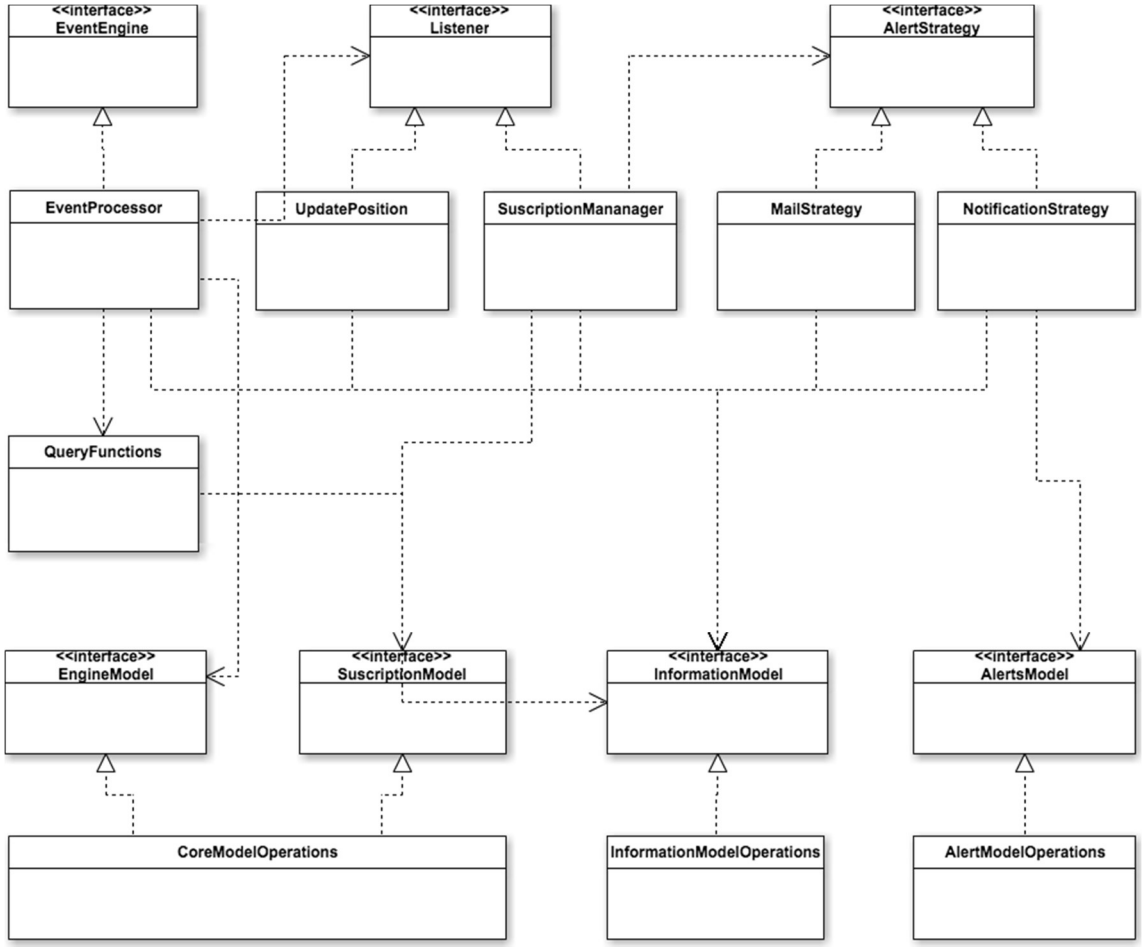


Figura 5.2: Diagrama de clases del Back-end

usuarios a eventos complejos. *InformationModel* permite registrar y consultar infromación sobre las entidades del sistema. Por último, *AlertsModel* permite registrar y consultar notificaciones de alerta.

5.3. Vista de proceso

Esta vista atiende aspectos como la distribución de tareas y concurrencia, además de validar el diseño lógico de la arquitectura, ya que este último debe dar soporte a las interacciones entre procesos y flujo de actividades [89]. Las figuras 5.3, 5.4 y 5.5 representan tres de los diagramas de actividades realizados para los casos de uso relevados.

5.4. Vista de desarrollo

La figura 5.6 muestra un diagrama de componentes, correspondiente a la vista de desarrollo de la arquitectura.

Los componentes se agrupan en tres conjuntos: servidor, cliente y servicios externos. Del lado del servidor se encuentran los principales componentes, responsables de ejecutar la lógica de negocio.

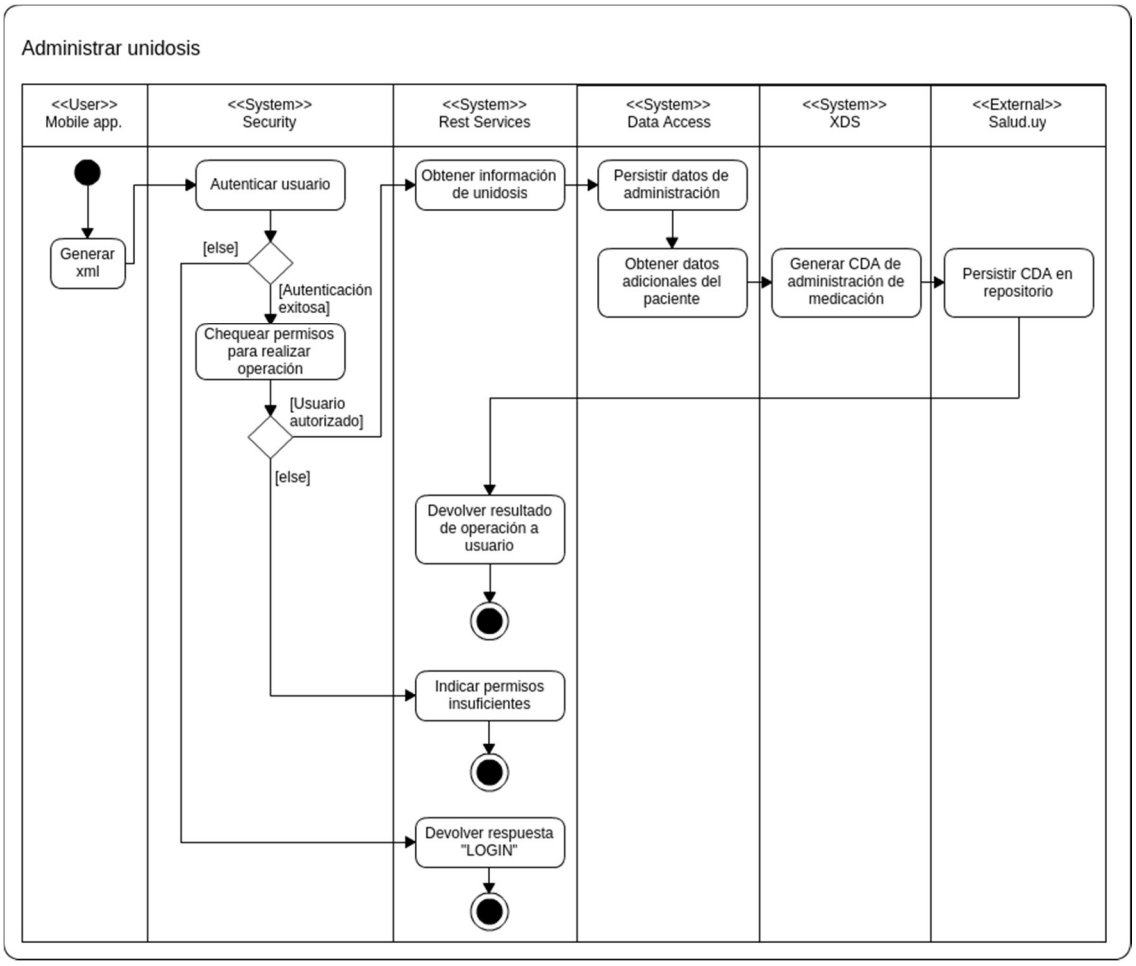


Figura 5.3: Diagrama de actividad para el caso de uso *Administrar unidosis*

Los clientes son las aplicaciones que acceden a los recursos del sistema a través de la interfaz RESTful que provee. Por último, el sistema depende de servicios externos para las notificaciones y acceso al repositorio XDS y MPI de *Salud.uy*.

Back-end

Es el componente principal. Implementa la lógica de negocio del sistema. Lo integran los siguientes subcomponentes:

- **Security**: capa de seguridad para acceso a los servicios que expone el Back-end. Los usuarios se deben autenticar y podrán ejecutar únicamente las operaciones para las que tengan permiso.
- **Rest Services**: interfaz RESTful del sistema. Los clientes acceden a los recursos utilizando el protocolo HTTP.
- **Event Engine**: procesa eventos primitivos y los registra en el log de eventos. Es el responsable de la detección de eventos complejos y la invocación a los listeners correspondientes.
- **Alerts**: implementa distintas estrategias de notificación para el envío de alertas.

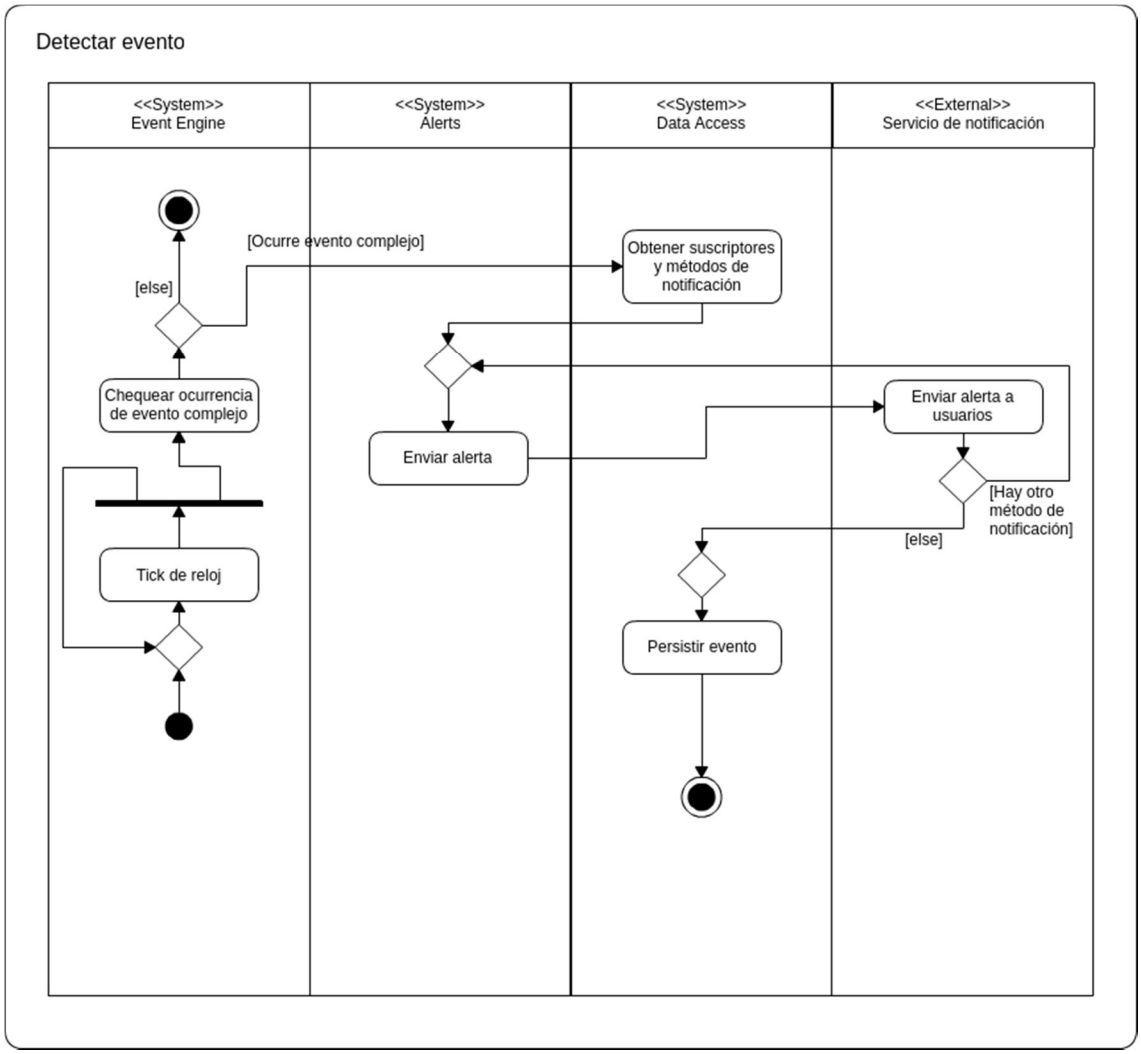


Figura 5.4: Diagrama de actividad para el caso de uso *Detectar evento*

- **Zone Permission Manager**: responsable de la asignación de permisos para circular por zonas del hospital.
- **Entity Manager**: responsable del registro y modificación de entidades del hospital.
- **EHR Manager**: responsable de la creación de documentos clínicos CDA y registro de pacientes en el XDS y MPI de *Salud.uy*.
- **Service Access**: capa de acceso a los servicios de *Salud.uy*.
- **Data Access**: capa de persistencia. Define las entidades persistentes y un conjunto de interfaces para recuperar y almacenar información.

Front-end

Corresponde a la interfaz web de administración del sistema. Es utilizada por el personal de administración y sus funcionalidades principales son el registro de entidades (pacientes, empleados, objetos, etc) y la asignación de permisos de zonas. Los integran los siguientes subcomponentes:

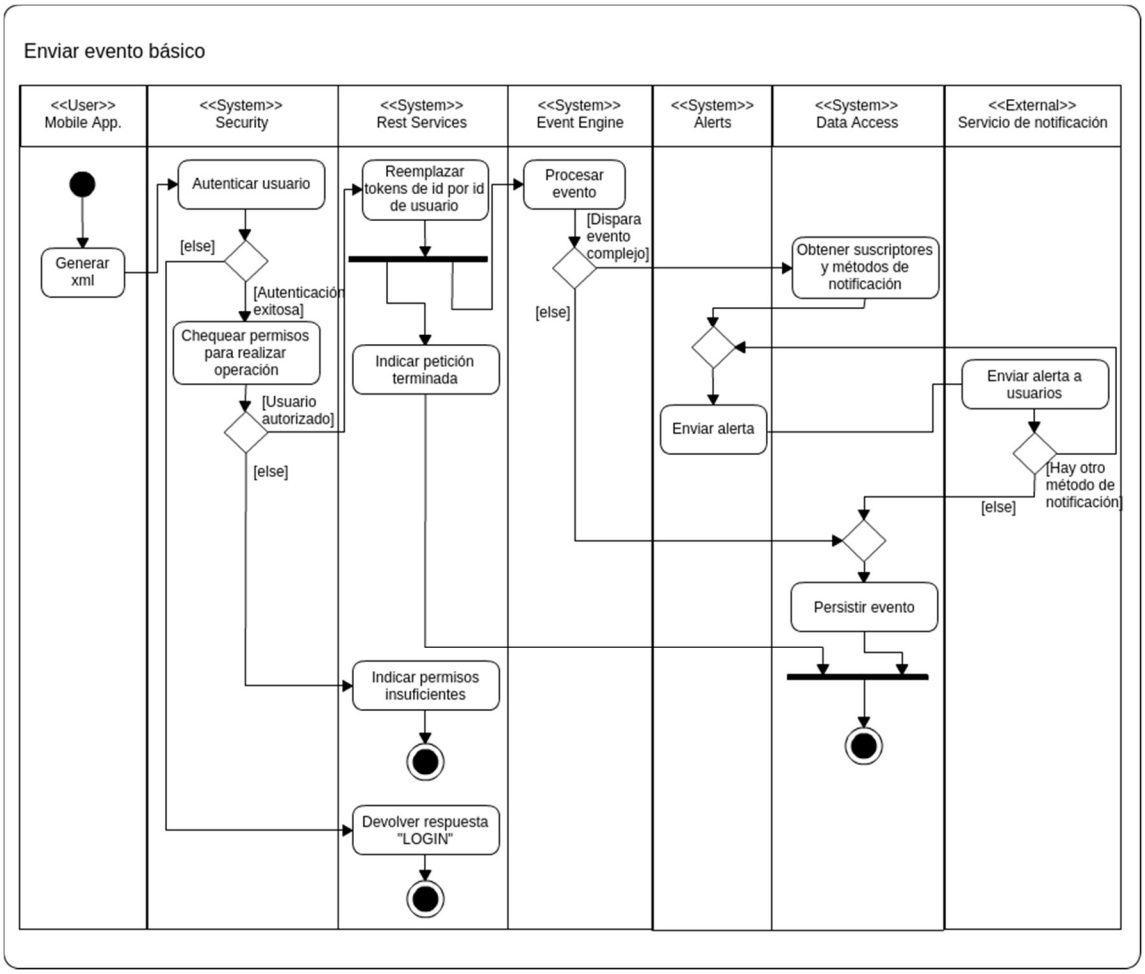


Figura 5.5: Diagrama de actividad para el caso de uso *Enviar evento básico*

- **Web**: capa de presentación web, compuesta por archivos HTML, CSS y Javascript.
- **Security**: capa de seguridad para acceso al *Front-end*.

SSO

Implementa el servicio *Single Sign-On*. De esta forma, un usuario que inicie sesión en uno de ellos podrá acceder a los servicios de los otros. En particular es necesario para poder acceder a operaciones del servidor central desde el módulo web. Para poder desplegar los módulos web y servidor central en servidores web diferentes, es necesario utilizar un servidor de SSO.

Web Browser

Los clientes acceden a la interfaz del *Frond-end* utilizando un navegador web.

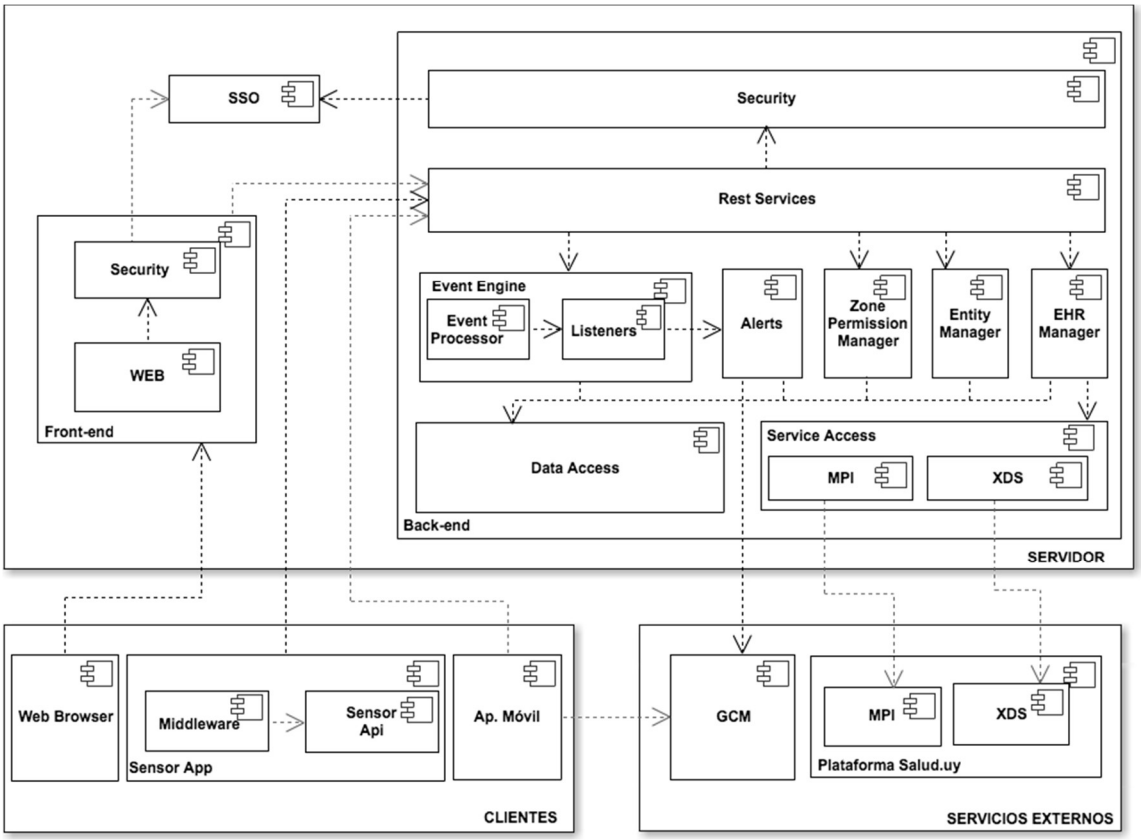


Figura 5.6: Diagrama de componentes

Aplicación móvil

Ejecuta en dispositivos móviles que serán utilizados por el personal del hospital. Entre sus funciones principales se encuentran la identificación de entidades y la recepción de notificaciones de alerta en base a prioridades definidas por el usuario. Los enfermeros utilizarán la aplicación como asistente del proceso de administración de unidosis, para evitar errores de medicación.

Sensor App

Se encarga de recibir y filtrar los datos que llegan en crudo de los sensores RFID. Luego de la fase de filtrado, los datos de salida son enviados al *EventEngine* para continuar con su procesamiento.

GCM Services

El servidor envía mensajes de sincronización a los dispositivos móviles, que marcan la presencia de datos sin leer, en particular alertas de situaciones de riesgo. Cuando el dispositivo móvil de un usuario recibe un mensaje de sincronización, inmediatamente consulta al servidor para obtener la información.

Plataforma *Salud.uy*

Responsable del repositorio XDS y MPI nacionales. Aquí se almacenan los pacientes de todo el país junto a sus documentos clínicos.

5.5. Vista física

La figura 5.7 muestra el diagrama de despliegue del prototipo realizado, correspondiente a la vista física de su arquitectura.

5.6. Vista de casos de uso

A continuación se describen los casos de uso más relevantes que determinan la arquitectura del sistema. La figura 5.8 muestra la vista de casos de uso del mismo. Para una descripción de los casos de uso del sistema, ver sección 4.4;

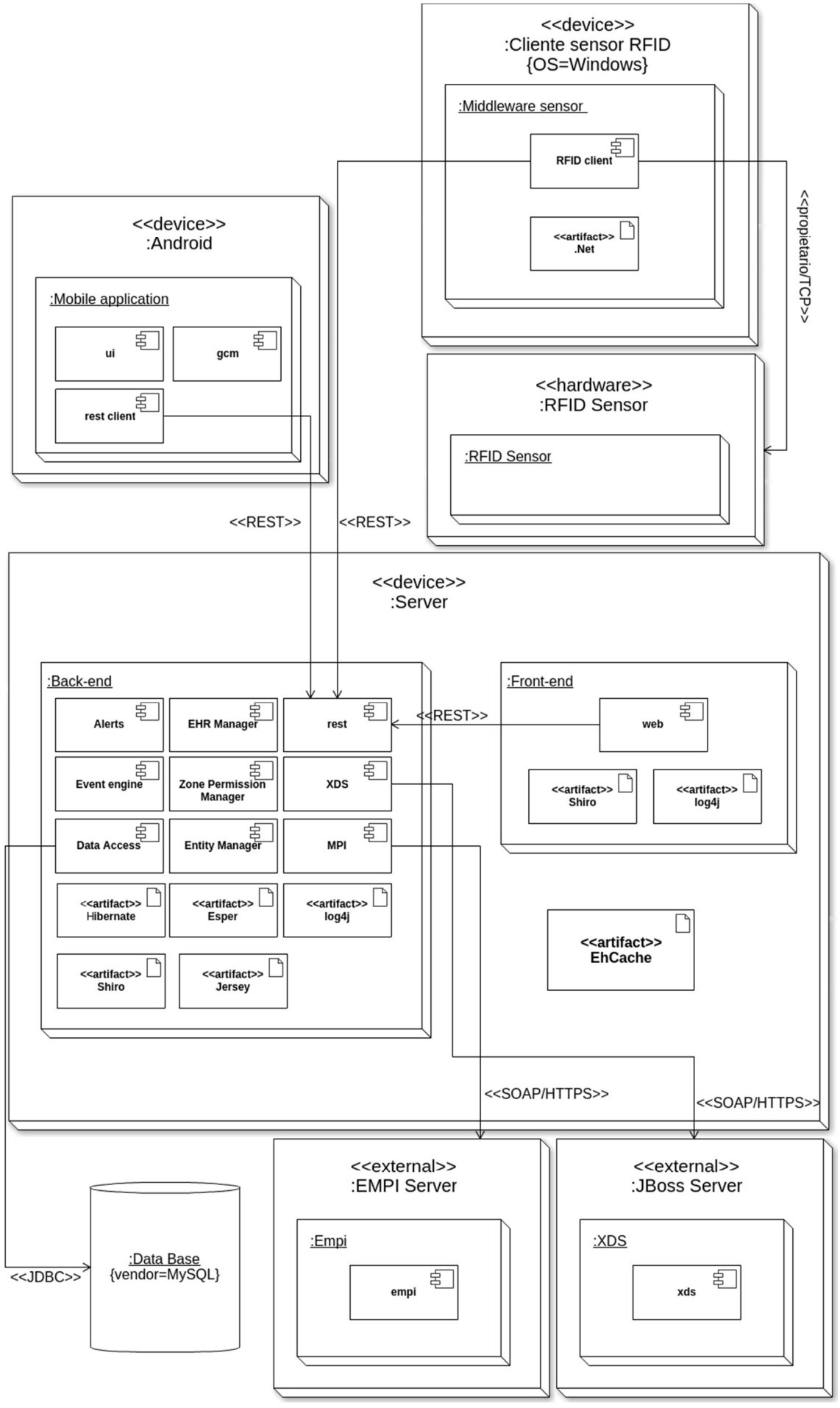


Figura 5.7: Diagrama de despliegue

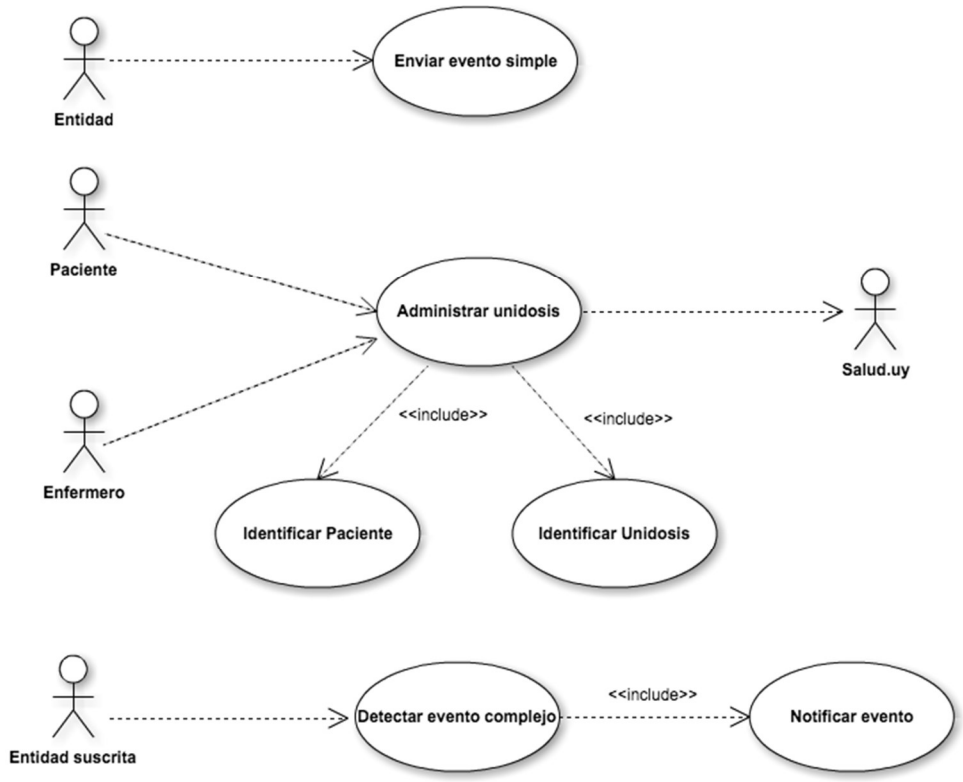


Figura 5.8: Diagrama de casos de uso

Capítulo 6

Implementación

6.1. Aspectos físicos

En esta sección se describen los detalles de los elementos físicos con los que cuenta el prototipo.

RFID

Se eligió RFID pasivo sobre las demás tecnologías introducidas en la sección 2.4 para la ubicación de entidades del hospital porque:

- No requiere interacción directa con la entidad. El sensor detecta el tag de manera automática. No introduce demoras en los procesos actuales.
- No requiere línea de visión con los tags. Esto permite detectar tags en entornos no controlados, por ejemplo cuando está cubierto por algún objeto.
- Los dispositivos que se comercializan se basan en estándares. No se depende de un proveedor particular.
- Hay una gran diversidad de tipos de tag, y de menor costo que el resto de las tecnologías. Por ejemplo algunos tags soportan altas temperaturas y humedad. Esto los hace apropiados para objetos que deben esterilizarse.
- La mayoría de los sensores trae consigo una API para integración con el sistema que los utiliza.
- Puede utilizarse para la identificación de objetos, a diferencia de la biometría.
- La tecnología GPS es, por sí sola, bastante imprecisa dentro de edificios [53–55].
- Se descartó la biometría dado que puede incurrir en errores de identificación no aceptables (ver sección 2.4.4).

Los sensores de RFID utilizados cuentan con las siguientes características[1]:

[1]Para más información sobre las características mencionadas, ver apéndice A.

- Antena con polarización circular incluída.
- Frecuencias de operación en el rango UHF.
- Soporte a protocolos EPC Class 1 Gen 2, e ISO 18000-6B.
- Interfaces de comunicación de datos leídos a través de un protocolo propietario sobre WiFi.

Se colocaron dos sensores RFID en la puerta de acceso de una sala del hospital, según muestra la figura 6.1. Estos se conectan a la red local del edificio mediante un punto de acceso WiFi. La instalación se realizó en conjunto con el departamento de intendencia y el centro de cómputo. El departamento de intendencia colocó soportes y tomas de energía eléctrica para los sensores, y el centro de cómputo hizo las configuraciones de red necesarias para lograr conectividad entre éstos y el servidor central. Cada sensor se asocia a la zona del hospital donde se encuentra.

(a) Esquema de instalación **(b)** Lector exterior a sala **(c)** Lector en interior de sala

Figura 6.1: Instalación de lectores RFID en sala y pasillo del hospital

Tarjetas RFID

Los tags utilizados para el prototipo se muestran en la figura 6.2. Sobre ellos se imprime una etiqueta con los datos básicos del paciente, visitante o empleado del hospital. Un número aleatorio único almacenado en la memoria de la tarjeta se asigna en la base de datos del prototipo como identificador de la persona.

Las características principales indicadas por el proveedor son:

- Protocolo de comunicación EPCglobal UHF Class 1 Gen 2.
- Distancia de lectura menor o igual a 3 metros.
- Número aleatorio único pregrabado.
- 85 milímetros de largo, 54 de ancho y 10 de espesor.

Envases plásticos y etiquetas de seguridad para unidosis

Se propone la utilización de recipientes plásticos cilíndricos para el envasado de unidosis. Sobre estos se coloca una etiqueta de seguridad de tipo *piel de cebolla*, con la propiedad de destruirse al intentar quitarse de la superficie donde se adhiera (ver figura 6.3). Sobre la etiqueta debe imprimirse

(a) Datos de paciente ficticio **(b)** Código de barras en reverso

Figura 6.2: Tarjeta RFID con datos de paciente ftcticio

un código QR que identifique a la unidosis. Además, la etiqueta debe colocarse de forma tal que sea necesario romperla para abrir el envase.

(a) Envase con etiqueta de seguridad **(b)** Etiqueta de seguridad rota

Figura 6.3: Envase con etiqueta de seguridad colocada

En el caso de que la etiqueta se rompa antes de leer el código QR con la aplicación móvil durante la administración, el prototipo notificará que la unidosis no fue administrada (ver sección 6.2.2). Se cree que el reporte de estas irregularidades influirá positivamente en el proceso de identificación del paciente, dado que obligará al personal a leer la etiqueta antes de administrar la dosis.

6.2. Procesamiento de eventos

Se evaluaron dos procesadores de eventos complejos (CEP, por sus siglas en inglés) que implementan el paradigma ECA descrito en la sección 2.3.

Esper

Es un CEP de código abierto, desarrollado en Java, diseñado para aplicaciones que requieren analizar y reaccionar ante eventos en tiempo real [90]. Entre las funcionalidades principales se

encuentran:

- Definición y detección de patrones de eventos.
- Filtrado sobre la serie de eventos recibidos.
- Funciones de agregación (COUNT, DISTINCT, etc) aplicadas a ventanas de tiempo.
- Ejecución de acciones tras la detección de eventos complejos.

Algunas de las características más importantes de Esper son:

- Baja latencia: las aplicaciones pueden reaccionar en tiempo real (desde milisegundos hasta unos pocos segundos) ante el cumplimiento de una condición.
- Alto desempeño: permite procesar grandes volúmenes de eventos, en el orden de los 100.000 por segundo.
- Define un lenguaje simple (EPL), similar a SQL, que permite analizar y realizar consultas sobre la serie de eventos recibidos.
- Los eventos primitivos pueden representarse como POJOs (Plain Old Java Object) o documentos XML.
- Tiene documentación extensa y clara.

Drools Fusion

Droll [91] es un procesador de reglas, de código abierto, desarrollado en Java. Las reglas se declaran como un grupo de enunciados (*if - then*), y son evaluadas con los datos de entrada. En caso de cumplirse todas las condiciones de una regla, se ejecutan las acciones definidas por el *then*.

Las funcionalidades provistas son similares a las de Esper.

Algunas de las características más importantes de Drools Fusion son:

- El lenguaje de definición de reglas es simple.
- Soporta la especificación de reglas en lenguaje natural. Previamente es necesario definir un DSL (Domain Specific Language) que haga la traducción al lenguaje de reglas nativo.
- Los eventos primitivos se representan con POJOs.
- Soporta un plugin para Eclipse para facilitar el desarrollo.
- Tiene documentación extensa y clara.

CEP elegido

Se eligió Esper para la implementación del prototipo. La ventaja principal es la representación XML de los eventos primitivos. Esto promueve la portabilidad y facilita la incorporación de nuevos eventos en tiempo de ejecución y el envío por la red (a diferencia de un POJO, no requiere serialización).

6.2.1. Eventos primitivos

Los tipos de evento primitivo están definidos por esquemas XSD (Xml Schema Definition), mientras que las ocurrencias corresponden a documentos XML que cumplen las restricciones impuestas por dichos esquemas. Se definieron los siguientes tipos de evento primitivo:

- **Identiftcación de entidad**: se envía cuando una entidad del hospital es identificada.
- **Unidosis empaquetada**: se envía cuando una unidosis es armada en farmacia.
- **Inicio y cierre de sesión**: se envía cuando un usuario inicia o cierra sesión en la aplicación móvil.
- **Medicación sobrante**: se envía cuando hay medicación sobrante en la administración de una unidosis.

Cada evento primitivo tiene un conjunto de atributos; por ejemplo, el evento de identificación de entidad tiene como atributos los identificadores de la entidad, la zona donde fue identificada y el usuario que la identifica.

6.2.2. Eventos complejos

Los tipos de evento complejo se definen utilizando el lenguaje EPL definido por Esper y se almacenan en un archivo con extensión .epl, que puede ser modificado en tiempo de ejecución para agregar nuevos tipos. Se definieron los siguientes tipos de evento complejo:

- **Persona en lugar no autorizado**: se detecta cuando una persona se identifica en una zona no autorizada. Por ejemplo, pacientes que salen del hospital, se cambian de cama, entran por equivocación al block quirúrgico o visitantes en salas que no corresponden.
- **Objeto en lugar no autorizado**: análogo al anterior. Permite controlar la circulación de objetos y detectar posibles hurtos.
- **Unidosis no corresponde al paciente**: se detecta cuando se intenta administrar una unidosis a un paciente incorrecto.
- **Unidosis no administrada**: se detecta cuando una unidosis prevista para un turno no es administrada dentro de un intervalo de tiempo establecido.
- **Unidosis administrada fuera de turno**: se detecta cuando una unidosis es administrada en un horario incorrecto.

Por cada tipo de evento complejo puede haber uno o más listeners encargados de ejecutar alguna acción tras la detección delevento.

El apéndice G muestra el archivo .epl con la definición formal de los tipos de eventos complejos descritos.

6.3. Estrategias de notificación

El servidor central cuenta con un módulo donde se implementan las estrategias de notificación del prototipo. Un archivo de configuración indica el nombre de la clase que implementa cada estrategia. Los cambios realizados en ese archivo se verán reflejados en el servidor central en tiempo de ejecución, por lo que no es necesario reiniciarlo si se desea modificar alguna estrategia concreta. La implementación sigue el patrón de diseño *Strategy*, como muestra la figura 6.4. Además, una fábrica se encarga de crear instancias de las estrategias concretas, mediante la API *Reflection* de Java. De esta forma se consigue desacoplar completamente la lógica de soporte de las notificaciones de cada estrategia concreta. También permite agregar nuevas estrategias en tiempo de ejecución, sin necesidad de compilar y desplegar nuevamente el servidor completo.

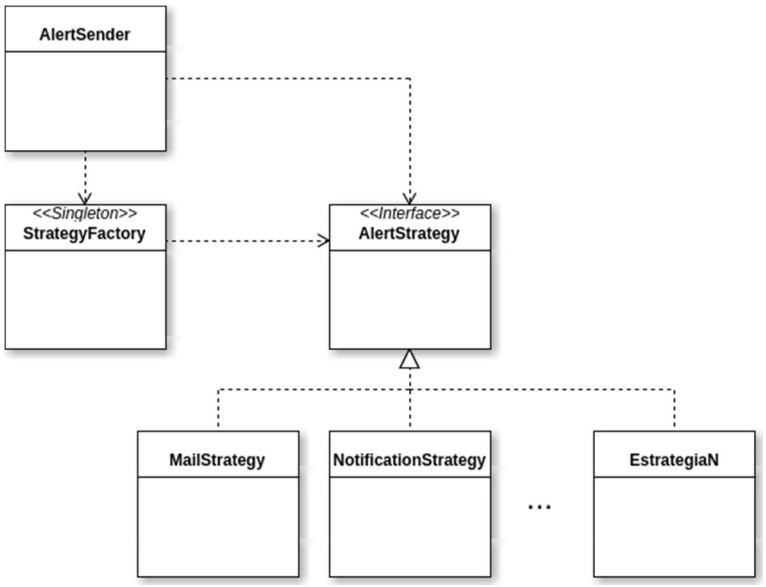


Figura 6.4: Diseño del módulo de alertas

Se implementaron dos estrategias concretas de notificación: al dispositivo móvil y mediante correo electrónico.

Para la primera estrategia se utilizó la infraestructura GCM[2], debido a que es el mecanismo estándar de Android para notificaciones. Para mantener la confidencialidad en los datos manejados, las notificaciones a dispositivos móviles solo indican que ocurrió un evento nuevo, y luego la aplicación debe encargarse de obtener la información extra directamente del servidor central a través de la red local del hospital.

Para las notificaciones mediante correo electrónico se utilizó un servicio de terceros, por tratarse de un prototipo y no utilizar datos reales. El sistema en producción debería ser configurado para utilizar un servicio de correo institucional.

[2] https://developers.google.com/cloud-messaging/

6.4. Seguridad

Se evaluaron algunas librerías y frameworks[3] de seguridad para Java.

Spring Security

Provee mecanismos de autenticación, autorización, y posibilidad de extender sus funcionalidades [92].

Aspectos favorables:

- Constituye un proyecto activo.
- Presenta amplia documentación oficial y actividad (en blogs, foros de discusión, etc.) de sus usuarios.

Aspectos desfavorables:

- La mayoría de la información existente refiere al desarrollo junto al framework para aplicaciones web Spring.

Último reporte de vulnerabilidad encontrada: 29-07-2015 [93].

OWASP Enterprise Security API

Define interfaces para aplicar controles de seguridad en una aplicación, junto con implementaciones de referencia para autenticación y autorización, entre otros [94].

Aspectos favorables:

- Proyecto realizado por organización con experiencia en el área de seguridad informática.

Aspectos desfavorables:

- Actualmente inactivo. Los responsables del proyecto recomiendan utilizar otras librerías o frameworks.

Último reporte de vulnerabilidad encontrada: 05-05-2016 [95].

Apache Shiro

Ofrece mecanismos de autenticación y autorización, control de acceso basado en roles, integración con servidores de inicio de sesión único, etc. [96].

Aspectos favorables:

- La documentación disponible y bien organizada.

[3] Conjunto de herramientas, librerías, y otros artefactos que sirven como soporte para resolver una determinada problemática durante el desarrollo de un sistema informático.

- Es de uso general. No está restricto a una plataforma (web, de escritorio, etc.) particular.

Aspectos desfavorables:

- No implementa el uso de salts para la autenticación, aunque si provee las interfaces y ejemplos necesarios para hacerlo.

Último reporte de vulnerabilidad encontrada: 10-07-2014 [97].

Apache Santuario

Es una librería que implementa estándares para encriptación y firma de documentos XML [98–100].

Aspectos favorables:

- Puede ser utilizada como complemento a otros frameworks, como *Spring Security* o *Apache Shiro*.
- Constituye un proyecto activo.

Aspectos desfavorables:

- No hay mucha actividad en la comunidad de usuarios.
- Es incompleta en términos de los requerimientos del proyecto (por ejemplo, no ofrece mecanismos de autenticación).

Último reporte de vulnerabilidad encontrada: 20-02-2015 [101].

Framework elegido

Se eligió Apache Shiro para el diseño e implementación del prototipo, debido a que la documentación es muy clara, de fácil comprensión, y las funcionalidades que provee son suficientes para las necesidades del proyecto. Se extendió el framework para que las contraseñas de usuarios del sistema se almacenen en una base de datos, aplicándoles la función de hash SHA-256, y utilizando salts pseudoaleatorias de 64 bits, generadas por Shiro.

6.5. Persistencia

Se utilizó la implementación de JPA de Hibernate. La base de datos utilizada es MySQL (versión 5.5.49), a petición del centro de cómputo del hospital.

Las ventajas más importantes de la implementación del módulo de persistencia con JPA son:

- Se consigue abstracción del modelo relacional de las bases de datos. A nivel de implementación se trabaja con el paradigma de orientación a objetos.
- En particular Hibernate provee un sistema de varios niveles de memoria caché que en general

mejoran el rendimiento del módulo de persistencia [102] (aunque ante una cantidad grande de datos, la de segundo nivel podría perjudicarlo [103]).

- Permite configurar fácilmente un repositorio de conexiones a la base de datos, que Hibernate gestiona de forma transparente. Para el prototipo se utilizó uno.
- La validación del formato de los parámetros para generar consultas a la base de datos es responsabilidad de Hibernate. Esto mitiga el riesgo a introducir errores en el código u omitir controles, lo que podría resultar en un problema de seguridad[4].
- Los cambios en las clases que desean persistirse (en sus relaciones o atributos) se ven reflejados en las tablas de la base de datos de forma transparente.

Se eligió Hibernate sobre otras implementaciones de JPA (OpenJPA, EclipseLink y DataNucleus). Según una prueba de desempeño[5] con MySQL investigada, OpenJPA obtiene un mejor resultado general con respecto a Hibernate, seguidos por EclipseLink y DataNucleus, en ese orden [105]. Otra prueba arroja mejores resultados generales de Hibernate [106]. Además, ya se contaba con conocimiento previo de Hibernate, por lo que la elección de esta implementación mitiga riesgos técnicos.

6.6. Interfaz Web y REST

Los casos de uso relacionados a aspectos administrativos de las entidades del sistema (altas, bajas, modificaciones de usuarios, personas, objetos, permisos, etc.) se realizan a través de una interfaz web.

De lado del servidor se utiliza JSP y JSTL [107] para el manejo de parámetros y traducciones de textos. Se integra con Apache Shiro para mostrar u ocultar contenido según los permisos del usuario que inició sesión.

De lado de cliente se utiliza AngularJS [108], que permite desacoplar el manejo de datos de su presentación.

Por otro lado, el sistema expone sus servicios mediante una interfaz RESTful. A diferencia de SOAP, los servicios web RESTful son simples, están diseñados para operar en clientes "livianos"(como dispositivos móviles y navegadores web), requieren únicamente soporte del protocolo HTTP y utilizan XML o JSON para la representación de recursos u objetos, siendo el último, el formato nativo de JavaScript. Del revelamiento de tecnologías realizado se desprende también que la plataforma Android no tiene soporte nativo para los servicios SOAP.

6.7. Aplicación móvil

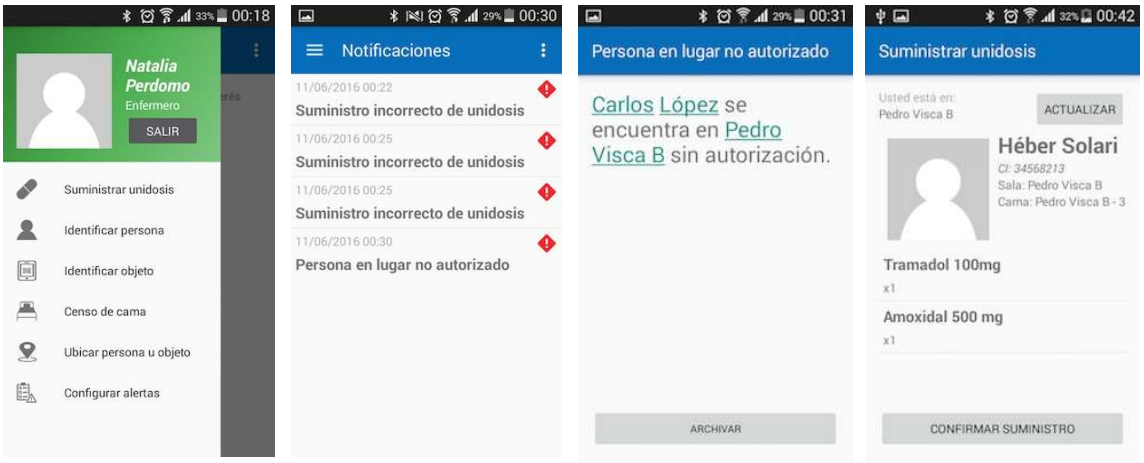
La aplicación accede a los servicios del servidor a través de la interfaz RESTful y es utilizada por el personal del hospital para realizar los siguientes casos de uso:

- Administrar unidosis a los pacientes.

[4] Ver [104]
[5] Se miden tiempos empleados en realizar consultas, insertar o borrar datos, etc.

- Identificar y ubicar objetos y personas.
- Realizar el censo de camas diario.
- Recibir notificaciones y alertas de situaciones de riesgo.
- Configurar suscripciones a eventos.

La figura 6.5 muestra algunos detalles de su interfaz gráfica.



(a) Menú principal de la aplicación

(b) Listado con las notificaciones recibidas.

(c) Alerta de persona en lugar no autorizado.

(d) Caso de uso administración de unidosis.

Figura 6.5: Vistas de la aplicación móvil

Se eligió Android como plataforma para el desarrollo de esta aplicación dado que, entre los dispositivos móviles adecuados para el prototipo, existe una mayor variedad (en prestaciones y costo) de los basados en este sistema operativo.

6.8. Middleware para sensores RFID

Se implementó un middleware[6] que establece dos conexiones: una con el servidor central y otra con un sensor de RFID. Primero inicia sesión en el servidor (con un usuario con permisos únicamente para enviar eventos básicos), y luego comienza a consultar al sensor de RFID qué tags leyó. En caso de obtener datos, genera un hilo de ejecución nuevo donde envía un evento de identificación de entidad al servidor central, incluyendo la zona a la que pertenece el sensor RFID y el identificador del tag.

La API provista por el proveedor del sensor es para C#, por lo cual la aplicación fue desarrollada en dicho lenguaje.

6.9. Integración con MPI y XDS

Para seguir la línea de interoperabilidad e intercambio de datos clínicos, el prototipo se integra al MPI y repositorio XDS de Salud.uy.

[6]Software diseñado para actuar como adaptador entre dos sistemas

Cuando se registra un paciente nuevo, el sistema envía un mensaje para registrarlo a su vez en el MPI de Salud.uy (específicamente, un mensaje ADT04, ver sección E.2). De esta forma, cada paciente tendrá un identificador único dentro del dominio de identificación nacional. Cada vez que se modifican los datos de un paciente ya registrado, el sistema propaga los cambios al MPI (enviando un mensaje ADT08).

Por otro lado, se implementó el template CDA, *Informe Administración de Unidosis*, con el propósito de documentar detalladamente los actos clínicos de administración de medicación. Cuando un enfermero termina de administrar una unidosis, el sistema genera un documento CDA y lo envía al repositiorio XDS de Salud.uy iniciando una transacción *Provide and Register Document Set* (ver sección E.5.2). El template fue construido siguiendo las guías de implementación [109–112] y validado por Salud.uy. Por más detalles, ver apéndice F

Capítulo 7

Pruebas y resultados

Este capítulo presenta las pruebas realizadas sobre el sistema.

7.1. Pruebas de hardware

Se realizaron pruebas para medir la capacidad del prototipo de determinar correctamente la ubicación de un tag. El entorno de prueba es el descrito en la sección 6.1. La figura 6.1a muestra un esquema de ubicación de los sensores en sala y pasillo.

Se considera que el prototipo determina la ubicación de un tag cuando es detectado únicamente por un sensor, durante un intervalo de tiempo de 0,5 s. En este caso, la zona reportada por el prototipo es la de dicho sensor. Si además el sensor se ubica en la misma zona donde se encuentra el tag, se dice que la ubicación determinada es correcta. Si, por el contrario, el sensor se encuentra en una zona diferente a la del tag, la ubicación determinada es errónea. Por otro lado, si ningún sensor detecta al tag durante un intervalo de tiempo mayor o igual a 0,5 s, o lo hacen ambos en un intervalo de tiempo menor a 0,5 s, se dice que la ubicación es indeterminada.

A modo de ejemplo, se espera que el sensor colocado en el pasillo (SP) no detecte a un tag que se encuentra en la sala, y viceversa. Si tanto SP como el sensor ubicado en la sala (SS) detectan al tag en un intervalo menor a 0,5 s, la ubicación del tag es indeterminada.

Cada una de las pruebas fue realizada con una persona llevando una tarjeta de RFID en un colgante. Se probó:

- **Escenario 1:** persona en pasillo, a más de un metro de distancia de la entrada, con la puerta de la sala cerrada.
- **Escenario 2:** igual a escenario 1, pero con puerta abierta.
- **Escenario 3:** persona en pasillo, a menos de un metro de distancia de la entrada. Puerta cerrada.
- **Escenario 4:** igual a escenario 3, pero con puerta abierta.
- **Escenario 5:** persona en sala, a menos de un metro de distancia de la entrada. Puerta

cerrada.

- **Escenario 6:** igual a escenario 5, pero con puerta abierta.
- **Escenario 7:** persona en sala, a más de un metro de la entrada. Puerta cerrada.
- **Escenario 8:** igual a escenario 7, pero con puerta abierta.

Se tomaron en total 400 muestras, divididas en 50 por escenario. La tabla 7.1 presenta los resultados obtenidos por escenario. Se observa que, en un 90 % del total de las pruebas, el prototipo logró ubicar correctamente al tag.

			Indeterminados (%)	
Escenario	**Correcto (%)**	**Incorrecto (%)**	No se detecta tag	Ambos detectan tag
Escenario 1	100	0	0	0
Escenario 2	100	0	0	0
Escenario 3	90	0	10	0
Escenario 4	90	0	10	0
Escenario 5	70	0	0	30
Escenario 6	70	0	0	30
Escenario 7	100	0	0	0
Escenario 8	100	0	0	0

Tabla 7.1: Resultados de pruebas de ubicación de tags. Para cada escenario se muestra el porcentaje de resultados de ubicación correctos, incorrectos, indeterminados porque ningún sensor detectó el tag, e indeterminados porque ambos sensores lo detectaron.

Es importante mencionar que si la tarjeta está en contacto con el cuerpo de la persona (por ejemplo, con la palma de la mano o el torso), ningún sensor la detecta. Este aspecto debe ser tenido en cuenta, de implantarse completamente el sistema.

7.2. Pruebas funcionales del software

A continuación se describe el conjunto de pruebas funcionales realizadas.

Pruebas unitarias y de integración

Se utilizó la herramienta JUnit[1] para realizar pruebas unitarias y de integración. La tabla 7.2 muestra el cubrimiento de código alcanzado para cada módulo.

También se generaron pruebas unitarias para todos los módulos del servidor central con la herramienta Evosuite[2]. El cubrimiento de código alcanzado con ella fue muy bajo en casi todos los casos. La tabla 7.3 muestra los resultados obtenidos para cada módulo. Es importante señalar que los errores reportados por Evosuite no fueron tales.

[1] http://junit.org/
[2] http://www.evosuite.org/

Módulo	Cubrimiento alcanzado
model	70 %
engine	83 %
empi	89.7 %
xds	89.1 %

Tabla 7.2: Resultados de pruebas unitarias y de integración con JUnit

Módulo	Cubrimiento alcanzado
alerts	44.3 %
commons	32.8 %
empi	72.7 %
engine	20.4 %
model	31.9 %
rest	7.60 %
web	no aplica
xds	16.8 %

Tabla 7.3: Resultados de pruebas unitarias con Evosuite

Revisión automática de código

Se utilizó la herramienta FindBugs[3], que realiza análisis estático del código fuente. Con ella se encontraron y corrigieron dos errores en el servidor.

Pruebas basadas en casos de uso

Se diseñaron pruebas a partir de los casos de uso. El dominio de cada uno de los parámetros de entrada para cada caso de uso fue dividido en clases de equivalencia, donde dos entradas son equivalentes si el resultado esperado del sistema es del mismo tipo (error u operación exitosa). Para cada caso de prueba se tomó un elemento por clase de equivalencia de los parámetros de entrada, se determinó la salida esperada a partir de las especificaciones del sistema, y se comparó con el resultado obtenido. Puede verse una lista de algunos de los casos de prueba diseñados en el apéndice D. En total se realizaron 69 pruebas de este tipo. Se corrigieron todos los errores encontrados en esta etapa.

7.3. Pruebas de rendimiento del software

La funcionalidad crítica del sistema es la de emitir alertas ante eventos que representan situaciones de riesgo para el hospital. Estas alertas deben ocurrir en tiempo real; es decir, el intervalo entre que ocurre dicho evento y se recibe la alerta (latencia) debe ser pequeño (definido como 10 s. Ver sección 4.2, requerimiento no funcional *Latencia de alertas*). Es interesante medir este tiempo ante escenarios de carga, con múltiples usuarios conectados enviando eventos concurrentemente. La siguiente ecuación corresponde a una estimación de este tiempo.

[3] http://findbugs.sourceforge.net/

$$T_{total} = T_{envio} + T_{proc} + T_{reg} + T_{notif} + T_{cons} \quad (7.1)$$

donde:

- T_{envio} es el tiempo entre que el cliente crea y envía el evento básico hasta que el servidor lo recibe.
- T_{proc} es el tiempo que demora al servidor en registrar el evento en el log, procesarlo y detectar la situación de riesgo.
- T_{reg} es el tiempo que demora el servidor en generar y almacenar las notificaciones de alerta.
- T_{notif} es el tiempo que demora el servicio GCM en notificar a los clientes por datos a sincronizar.
- T_{cons} es el tiempo desde que un cliente consulta por notificaciones al servidor hasta que las recibe.

Observar que la ecuación 7.1 involucra cuatro actores: el cliente que envía el evento, el servidor, el servicio GCM y los clientes que reciben las notificaciones de alerta. Para simplificar la estimación de T_{total}, la ecuación anterior se divide en tres términos, donde cada uno se mide de manera independiente:

$$T_{total} = \underbrace{T_{envio}}_{A} + \underbrace{T_{proc} + T_{reg}}_{B} + \underbrace{T_{notif} + T_{cons}}_{C} \quad (7.2)$$

El hardware del servidor utilizado para las pruebas cuenta con las siguientes características:

- Procesador: Intel Core i7-2670QM de 4 núcleos.
- RAM: 6 GB.
- Disco duro: 200 GB de estado sólido.
- Sistema operativo: Ubuntu 14.04 LTS 64 bits.
- Red local: WiFi.

Los datos de prueba utilizados incluyeron:

- 100 empleados.
- 300 pacientes.
- 100 visitantes.
- 100 cuentas de usuario, correspondientes a los empleados.
- 380 zonas.
- 300 camas.

- 500 tipos de medicamentos.
- y 600 unidosis no administradas.

Estimación de T_A

Se empleó la herramienta Jmeter[4], que permite realizar pruebas de rendimiento, simulando escenarios de carga con múltiples usuarios concurrentes.

El escenario de prueba definido consiste en un conjunto de usuarios que envían concurrentemente eventos de identificación de entidades, abarcando las distintas zonas del hospital. Cada usuario envía al servidor un total de 10 eventos, seleccionando aleatoriamente las entidades y zonas que participan. No envía el siguiente hasta recibir la respuesta al evento previo. Se toma en cuenta el intervalo de tiempo que toma cada respuesta (latencia).

Se realizaron 10 repeticiones del escenario descrito, variando la cantidad de usuarios. En cada escenario se configuró Jmeter para generar todas las peticiones al servidor tan pronto como pueda[5]. La tabla 7.4 muestra los tiempos obtenidos experimentalmente.

Cantidad usuarios	10	20	30	40	50	60	70	80	90	100
T_A **(ms)**	46,0	72,8	95,6	130,8	149,4	184,4	207,6	241,7	263,9	294,6
σ_A **(ms)**	3,4	4,8	5,2	7,5	10,8	9,4	4,8	12,4	22,0	31,7

Tabla 7.4: Latencias de envío, procesamiento y registro de eventos

Utilizando la técnica de mínimos cuadrados, se encontró la recta de mejor ajuste. El coeficiente de correlación fue de 0,999, por lo que se puede concluir que estas latencias son de orden $O(n)$, siendo n la cantidad de usuarios.

La figura 7.1 muestra una gráfica con los tiempos y el ajuste por mínimos cuadrados.

Estimación de T_B

El servicio GCM permite enviar notificaciones a un dispositivo móvil. En el sistema presentado en este informe, la aplicación móvil de un usuario recibe un mensaje de sincronización, e inmediatamente realiza una consulta al servidor central para obtener la información del evento.

Es importante determinar el tiempo promedio que demoran en llegar los mensajes de sincronización. Para esto, se implementó una pequeña aplicación que envía una notificación a un único dispositivo móvil, utilizando el servicio GCM. El mensaje contiene la fecha del momento de envío, con precisión de milisegundos. Al ser recibida por la aplicación móvil, se resta el tiempo actual con el tiempo recibido en la notificación, y de esa forma se estima el tiempo consumido por el servicio GCM. Se tomaron 40 medidas de latencia con este método. La tabla 7.5 muestra los resultados obtenidos.

Cabe mencionar que la introducción de un servicio externo que no ofrece garantías sobre la

[4] http://jmeter.apache.org/

[5] Se configuraron los parámetros *Ramp-Up Period* de Jmeter en 0, y *Number of Threads* de acuerdo a la cantidad de usuarios concurrentes.

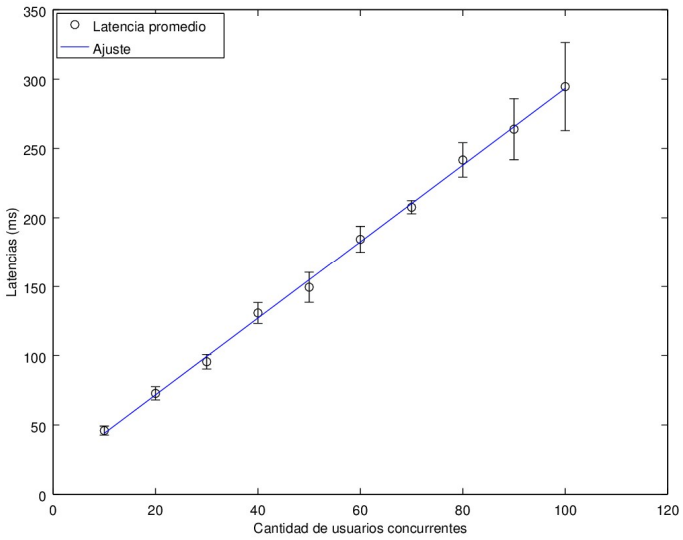


Figura 7.1: Latencias de envío, procesamiento y registro de eventos, y ajuste por mínimos cuadrados

$\overline{T_B}$ (ms)	σ_B (ms)
2829	265,3

Tabla 7.5: Latencia de notiftcación

latencia de notificación no es adecuada para un sistema de tiempo real.

Estimación de T_C

En este escenario, utilizando Jmeter, usuarios concurrentes consultan al servidor por notificaciones sin leer, y se mide el intervalo de tiempo desde que se envía la consulta hasta que se reciben las notificaciones. Al igual que para estimar T_A, se varió la cantidad de usuarios concurrentes accediendo al servidor central. Cada usuario envió a su vez 10 consultas al servidor.

La tabla 7.6 muestra los resultados obtenidos para esta parte. Se realizó un ajuste por mínimos cuadrados, obteniéndose un coeficiente de correlación lineal de 0,995.

Cantidad usuarios	10	20	30	40	50	60	70	80	90	100
T_C **(ms)**	38,90	75,70	113,8	134,8	169,1	202,9	236,6	263,7	337,6	350,9
σ_C **(ms)**	3,72	6,96	13,6	4,94	6,67	7,49	8,51	11,2	33,8	21,7

Tabla 7.6: Latencias de consulta

La imagen 7.2 muestra una gráfica de los datos, junto con la recta de ajuste lineal.

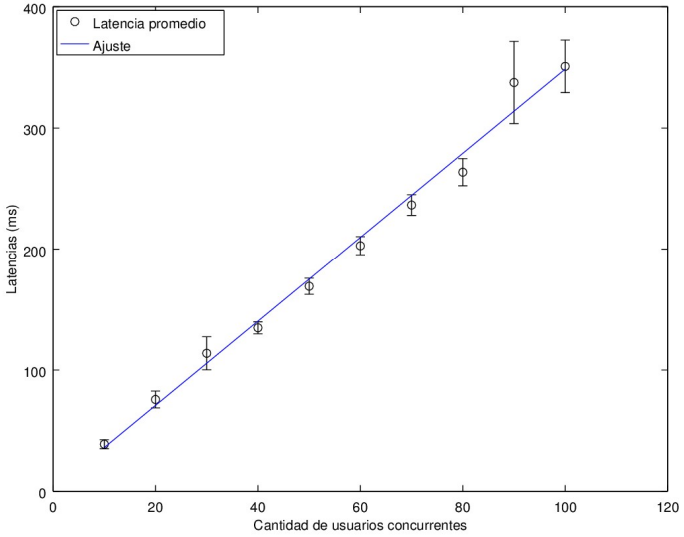


Figura 7.2: Latencias de consulta y ajuste por mínimos cuadrados

Estimación de T_{total}

Finalmente, sumando los resultados de arriba, se obtiene una estimación de la latencia total. La tabla 7.7 muestra estos resultados. Los desvíos estándar resultantes se calculan mediante propagación de errores. Se incluyen además los errores relativos porcentuales (ε_{rel} %).

Cantidad usuarios	$\overline{T_{total}}$ **(ms)**	σ_{total} **(ms)**	ε_{rel} %
10	2914	272	9,35
20	2978	277	9,31
30	3038	284	9,35
40	3095	278	8,97
50	3148	283	8,98
60	3216	282	8,77
70	3273	279	8,51
80	3334	289	8,66
90	3431	321	9,36
100	3475	319	9,17

Tabla 7.7: Latencias totales entre evento y notiftcación

Los resultados obtenidos se grafican en la figura 7.3. El coeficiente de correlación lineal es 0,998, lo que indica un buen ajuste.

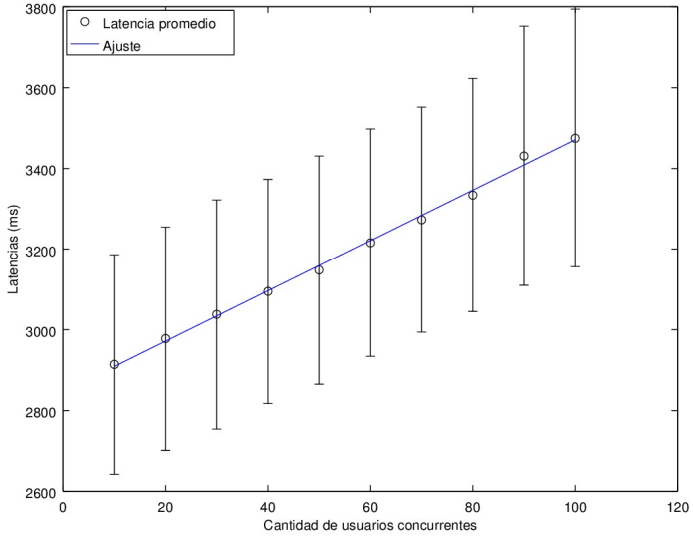


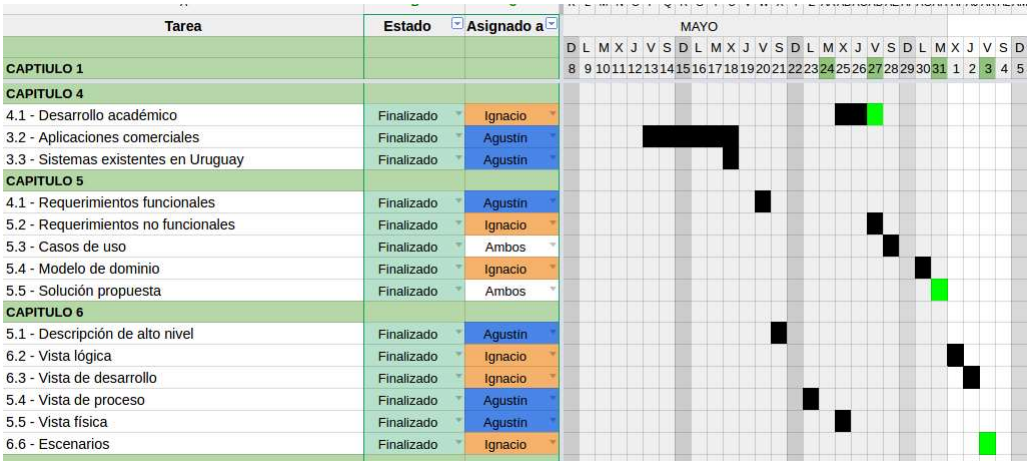
Figura 7.3: Latencias totales y ajuste por mínimos cuadrados. Los desvíos estándar son menores al 10 % en todos los casos.

Capítulo 8

Gestión del proyecto

8.1. Planificación

Con cierta regularidad se realizaron reuniones para planificar la realización de tareas concernientes al proyecto. Se utilizaron diagramas de Gantt como el que muestra la figura 8.1.



Tarea	Estado	Asignado a
CAPTIULO 1		
CAPITULO 4		
4.1 - Desarrollo académico	Finalizado	Ignacio
3.2 - Aplicaciones comerciales	Finalizado	Agustin
3.3 - Sistemas existentes en Uruguay	Finalizado	Agustin
CAPITULO 5		
4.1 - Requerimientos funcionales	Finalizado	Agustin
5.2 - Requerimientos no funcionales	Finalizado	Ignacio
5.3 - Casos de uso	Finalizado	Ambos
5.4 - Modelo de dominio	Finalizado	Ignacio
5.5 - Solución propuesta	Finalizado	Ambos
CAPITULO 6		
5.1 - Descripción de alto nivel	Finalizado	Agustin
6.2 - Vista lógica	Finalizado	Ignacio
6.3 - Vista de desarrollo	Finalizado	Ignacio
5.4 - Vista de proceso	Finalizado	Agustin
5.5 - Vista física	Finalizado	Agustin
6.6 - Escenarios	Finalizado	Ignacio

Figura 8.1: Diagrama de Gantt para planiftcar la escritura del informe. Las celdas marcadas en verde claro corresponden a hitos establecidos.

8.2. Esfuerzo realizado

Para el registro de esfuerzo se utilizó una planilla electrónica. La figura 8.2 muestra una gráfica del esfuerzo realizado mensualmente, categorizado por disciplina.

La tabla 8.1 muestra el esfuerzo total realizado, dividido por disciplina.

Se estima el total de tiempo dedicado por parte del Hospital Maciel a reuniones en 65 h, y el del Tutor de Proyecto en 40 h.

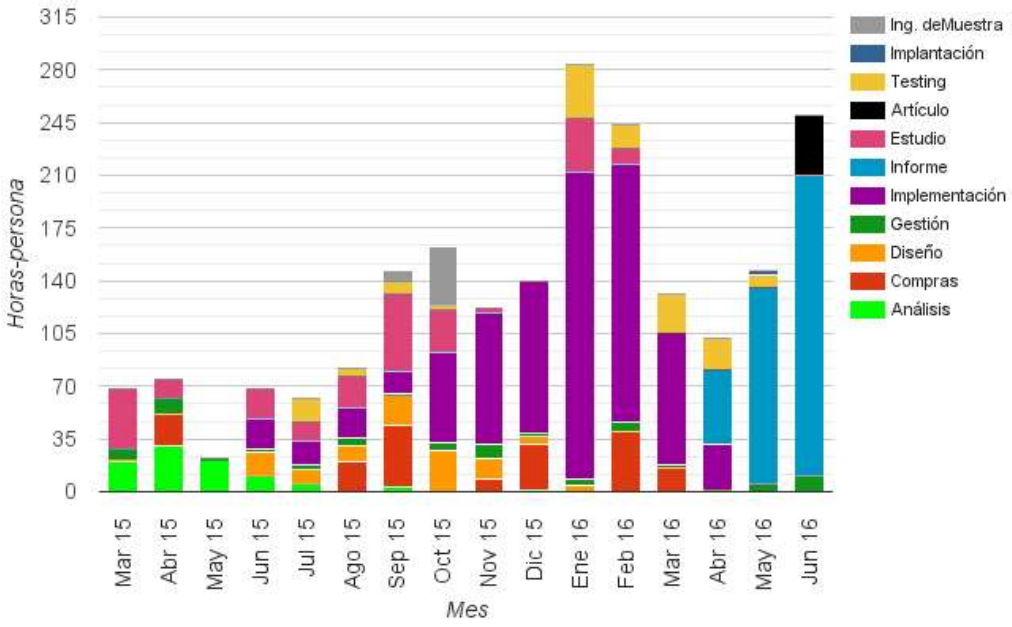


Figura 8.2: Gráftca de esfuerzo mensual total realizado

Disciplina	Esfuerzo (horas-persona)
Implementación	811
Informe	380
Investigación	236
Compras	176
Testing	140
Reuniones	116
Diseño	108
Análisis	92
Gestión	74
Ingeniería deMuestra	47
Artículo	40
Implantación	16
Total	2.236

Tabla 8.1: Esfuerzo total realizado por disciplina

Capítulo 9

Conclusiones

9.1. Resultados obtenidos

Se logró construir un prototipo funcional, de acuerdo al alcance definido. En relación con los errores de medicación, se estudió el procedimiento actualmente empleado por enfermeros para administrar medicamentos, se identificaron los puntos débiles, y se propuso una solución que pretende evitarlos. Se plantearon cambios en el sistema utilizado por Farmacia para la dispensación de unidosis - aún pendientes -, y una solución basada en código de barras y dispositivos móviles para verificar la correspondencia paciente-unidosis. Si bien aún no se han obtenido resultados sobre la efectividad de la solución desarrollada, la investigación realizada en el área sugiere una reducción sustancial en la frecuencia error de administración de medicación.

Por otra parte, se empleó la identificación por radiofrecuencia para conocer en qué zona se vio por última vez una entidad determinada. Un relevamiento del mercado permitió adquirir los equipos necesarios para el desarrollo del prototipo a un bajo costo, acorde al presupuesto. Se coordinó la instalación de los sensores RFID en una de las salas de internación del hospital, y se realizaron pruebas de ubicación mediante lectura de tags. Los resultados mostraron que, en el 90 % de los casos probados, el prototipo pudo determinar correctamente la ubicación. Se concluye que esta tecnología es adecuada para los requerimientos planteados.

Para la detección de situaciones de riesgo se utilizó un motor de eventos que, a través de un lenguaje flexible, permite definir cómo identificarlas y especificar las alertas que serán enviadas a los usuarios interesados. Los métodos de notificación son configurables, y se pueden agregar o quitar del sistema en tiempo de ejecución. Se realizaron pruebas de rendimiento, y se observó que la latencia de alerta no superó el umbral definido en los requerimientos no funcionales, y respondió de forma lineal a la cantidad de usuarios enviando eventos primitivos concurrentemente.

Uno de los problemas que se encontró es que el hospital no cuenta con estadísticas de errores de medicación o de otras situaciones de riesgo. En este sentido, el sistema será también un recolector de datos que podrán utilizarse para identificar debilidades en la operativa del hospital.

Como corolario de la realización del proyecto se adquirió experiencia en el área de informática médica. Se estudiaron estándares internacionales y servicios que los implementan, los cuales se integraron al prototipo desarrollado. La experiencia de trabajar en conjunto con el hospital fue muy enriquecedora, mostrando la importancia de la cooperación de varias disciplinas para resolver

problemáticas de gran impacto social.

9.2. Dificultades encontradas

La mayor dificultad encontrada tiene relación con el presupuesto del proyecto: las tecnologías para sistemas de ubicación son en general costosas, y resultó impracticable experimentar con cada una de ellas. La elección se basó en material teórico y artículos de investigación en el área. La compra de los sensores insumió un esfuerzo mayor al esperado, debido a trámites de importación y de homologación, por ser dispositivos emisores de ondas electromagnéticas.

Se encontraron dificultades para comprender completamente las necesidades del hospital, dado que sus procesos operativos involucran conocimiento técnico específico del área médica y farmacológica.

9.3. Trabajo futuro

Como primer paso, se deberían realizar los cambios propuestos al sistema existente en Farmacia, y seguir profundizando en posibles escenarios de administración de medicación no contemplados. Luego, utilizar el prototipo durante un período suficiente para obtener datos que permitan conocer su impacto en el procedimiento de administración de medicación. Naturalmente, el siguiente paso será extender el sistema al resto de las salas del hospital, de confirmarse, a juicio de la institución, un resultado beneficioso.

Es necesario sustituir el servicio GCM por un método de notificación alternativo, posiblemente basado en las mismas funcionalidades, pero sin depender de servicios externos. De esta forma se podrá reducir y controlar la latencia media de alerta.

Finalmente, sería deseable diseñar una interfaz simple para definición de eventos, más accesible a un usuario sin conocimientos técnicos, y un visualizador de los documentos CDA generados.

Referencias

[1] P. Najera *et al.*, "Real-time location and inpatient care systems based on passive RFID," *Journal of Network and Computer Applications*, vol. 34, no. 3, pp. 980–989, 2011.

[2] C. L. Yeung *et al.*, "An Investigation of an RFID-based patient-tracking and mobile alert system," *International Journal of Engineering Business Management*, vol. 3, no. 1, pp. 50–56, 2011.

[3] S. SC y B. P, "Wrong-side/wrong-site, wrong-procedure, and wrong-patient adverse events: Are they preventable?" *Archives of Surgery*, vol. 141, no. 9, pp. 931–939, 2006. Internet: http://dx.doi.org/10.1001/archsurg.141.9.931

[4] M. R. Cohen, *Medication errors: causes, prevention, and risk management*, M. R. Cohen, Ed. Jones & Bartlett Learning, 1999.

[5] ISMP, "Pharmaceutical Bar Coding to Improve Patient Safety Options for Technical Standards in the Canadian Environment Roundtable Discussion Paper," Institute for Safe Medication Practices Canada, Canada, Tech. Rep., 2008.

[6] H. L. Folli *et al.*, "Medication Pharmacists Error Prevention by Clinical in Two Children's Hospitals," vol. 79, no. 5, pp. 718–722, 1987.

[7] E. G. Poon *et al.*, "Effect of Bar-Code Technology on the Safety of Medication Administration," *New England Journal of Medicine*, vol. 362, no. 18, pp. 1698–1707, 2010.

[8] M. L. Durham *et al.*, "Reducing Medication Administration Errors in Acute and Critical Care: Multifaceted Pilot Program Targeting RN Awareness and Behaviors." *The Journal of nursing administration*, vol. 46, no. 2, pp. 75–81, feb 2016.

[9] Food and Drug Administration, "Bar Code Label Requirement for Human Drug Products and Biological Products," 2004. Internet: http://www.fda.gov/ohrms/dockets/98fr/04-4249.htm [2016-06-13]

[10] "Decreto número 148/001." Internet: http://www.impo.com.uy/bases/decretos/148-2001/2 [2016-05-18]

[11] "Resolución GMC 21/00." Internet: http://www.impo.com.uy/bases/decretos-internacional/148-2001 [2016-05-18]

[12] E. Martin y O. U. Press, *Concise Medical Dictionary*, ser. Oxford paperback reference. Oxford University Press, 2015.

[13] "About Medication Errors | NCC MERP." Internet: http://www.nccmerp.org/ about-medication-errors [2016-05-18]

[14] MSP, "Errores de medicación." Internet: http://www.msp.gub.uy/sites/default/files/archivos_adjuntos/IT.PRM_.02__Errores_de_Medicacion.pdf [2016-05-18]

[15] IHE International Inc, "IHE IT Infrastructure (ITI) Technical Framework Volume 1 Integration Profiles," *International Journal of Healthcare Technology and Management*, vol. 1, pp. 1–177, 2008.

[16] IHE, "IT Infrastructure Technical Framework: Volume 2a (ITI TF-2a): Transactions Part A -Sections 3.1 - 3.28," pp. 6–212, 2011. Internet: http://www.ihe.net/Technical_Framework/upload/IHE_ITI_TF_Rev8-0_Vol2a_FT_2011-08-19.pdf

[17] Health Level Seven International (HL7), "HL7 Clinical Document Architecture, Release 2.0." Internet: http://www.hl7.org/implement/standards/product_brief.cfm?product_id=7 [2016-05-14]

[18] IHE, "IT Infrastructure Technical Framework: Volume 2x (ITI TF-2x) Volume 2 Appendices," pp. 1–153, 2016.

[19] IHE, "IT Infrastructure Technical Framework Volume 3 (ITI TF-3) Cross-Transaction Specifications and Content Specifications," vol. 3, pp. 1–173, 2016.

[20] F. Wang *et al.*, "Complex RFID event processing," *VLDB Journal*, vol. 18, no. 4, pp. 913–931, 2009.

[21] S. Chakravarthy *et al.*, "Composite Events for Active Databases: Contexts and Detection Semantics,," *Proceedings of the 20th International Conference on Very Large Data Bases, (VLDB'94), September 12-15, 1994, Santiago de Chile, Chile.*, pp. 606–617, 1994. Internet: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.2501&rep=rep1&type=pdf

[22] Vidhya Krishnaprasad, "Event Detection in Active Databases," Master Thesis, University of Florida, 1994.

[23] D. Mishra, "SNOOP: An Event Specification Language for active Databses," Tesis de Maestría, University of Florida, 1991.

[24] ITU, "Radio Regulations Articles," ITU, Tech. Rep., 2012.

[25] B. Glover y H. Bhatt, *RFID Essentials*. O'Reilly Media, Inc., 2006.

[26] AETIC, "La tecnología RFID: Usos y oportunidades," pp. 15–29, 2009.

[27] S. A. Weis, "RFID (Radio Frequency Identification): Principles and Applications," 2007. Internet: http://www.eecs.harvard.edu/cs199r/readings/rfid-article.pdf [2016-06-13]

[28] F. M. Abu-Zidan *et al.*, "Clinical ultrasound physics," pp. 501–503, 2011.

[29] M. Hazas y A. Ward, *UbiComp 2002: Ubiquitous Computing: 4th International Conference Göteborg, Sweden, September 29 – October 1, 2002 Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, vol. 2498, ch. Broadband Ultrasonic Location System, pp. 264–280.

[30] M. Hazas y A. Hopper, "Broadband ultrasonic location systems for improved indoor positio-

ning," *IEEE Transactions on Mobile Computing*, vol. 5, no. 5, pp. 536–547, may 2006.

[31] O. J. Woodman y R. K. Harle, "Concurrent scheduling in the active bat location system," *2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2010*, pp. 431–437, 2010.

[32] R. Mautz, "Indoor Positioning Technologies," Disertación de Ph.D., ETH Zurich, 2012.

[33] C. S. R. Prabhu y A. P. Reddi, *Bluetooth technology, and its aplications with Java and J2ME*, 1st ed. PHI Learning, 2004.

[34] R. van Nee y R. Prasad, *OFDM for Wireless Multimedia Communications*, 1st ed. Norwood, MA, USA: Artech House, Inc., 2000.

[35] J. Haartsen, "Bluetooth: The universal radio interface for ad hoc, wireless connectivity," *Ericsson review*, no. 3, pp. 110–117, 1998. Internet: https://www8.cs.umu.se/kurser/TDBD16/VT07/bluetooth.pdf

[36] S. Feldmann *et al.*, "An indoor Bluetooth-based positioning system: concept, implementation and experimental evaluation," *International Conference on Wireless Networks*, no. January 2003, pp. 109–113, 2003.

[37] Y. Gu *et al.*, "Fast Indoor Localization of Smart Hand-Held Devices Using Bluetooth," in *2014 10th International Conference on Mobile Ad-hoc and Sensor Networks*. IEEE, dec 2014, pp. 186–194.

[38] A. Kwiecień *et al.*, *Reliability of Bluetooth Smart Technology for Indoor Localization System*. Cham: Springer International Publishing, 2015, pp. 444–454.

[39] M. Bargh, R. Groote, "Indoor localization based on response rate of bluetooth inquiries," *Acm*, pp. 49–54, 2008.

[40] L. Wang, *Behavioral biometrics for human identification: Intelligent applications*, ser. Premier reference source, L. Wang, Ed. IGI Global, 2009.

[41] I. Buciu y A. Gacsadi, "Biometrics Systems and Technologies : A survey Biometrics - Introduction," *International journal of computers communications & control*, vol. 11, no. June, pp. 315–330, 2016.

[42] D. M. Rankin *et al.*, "Iris recognition failure over time: The effects of texture." *Pattern Recognition*, vol. 45, pp. 145–150, 2012.

[43] D. White *et al.*, "Error rates in users of automatic face recognition software," *PLoS ONE*, vol. 10, no. 10, pp. 1–15, 2015.

[44] J. Kurose y K. Ross, *Computer networking: A top-down approach*, 6th ed., Pearson, Ed. Pearson, 2013, vol. 1.

[45] Kevin P. Corbley, "WiFi Inadequate as Real-Time Asset and Patient Tracking Solution," *Wireless Design and Development*, pp. 30–32, dec 2008.

[46] G. Sterling *et al.*, "Understanding Beacons: A Guide to Beacon Technologies," Localsearch Association, Tech. Rep. December, 2014.

[47] C. Gomez *et al.*, "Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology," *Sensors*, vol. 12, no. 9, p. 11734, 2012. Internet: http://www.mdpi.com/1424-8220/12/9/11734

[48] "Beacon station CC2640 | Alibaba Group." Internet: http://goo.gl/nm5an1 [2016-05-26]

[49] H. Kato *et al.*, *Barcodes for Mobile Devices*. Cambridge University Press, 2010.

[50] "Librería ZXing - Github." Internet: https://github.com/zxing/zxing [2016-05-18]

[51] A. Sears *et al.*, "Investigating touchscreen typing: the effect of keyboard size on typing speed," *Behaviour & Information Technology*, vol. 12, no. 1, pp. 17–22, 1993. Internet: http://dx.doi.org/10.1080/01449299308924362

[52] A. El-Rabbany, *Introduction to GPS: The Global Positioning System*, 2nd ed., ser. Artech House mobile communications series. Artech House, 2002.

[53] A. Casandra *et al.*, "Estimators of the indoor channel for GPS-based pseudolite signal," in *Electronics and Telecommunications (ISETC), 2010 9th International Symposium on*, nov 2010, pp. 233–236.

[54] F. Dovis *et al.*, "An assisted high-sensitivity acquisition technique for GPS indoor positioning," in *2008 IEEE/ION Position, Location and Navigation Symposium*, may 2008, pp. 1350–1361.

[55] S. Nirjon *et al.*, "COIN-GPS: Indoor Localization from Direct GPS Receiving," in *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '14. New York, NY, USA: ACM, 2014, pp. 301–314.

[56] Google, "Use indoor maps to view floor plans - Google Maps Help." Internet: https://support.google.com/maps/answer/2803784?co=GENIE.Platform%3DDesktop&hl=en [2016-05-05]

[57] "Ley número 18.335 - MSP." Internet: http://www.msp.gub.uy/sites/default/files/18.335.pdf [2016-05-18]

[58] D. Gollmann, *Computer Security*, 2nd ed. John Wiley & Sons, Ltd, 2006.

[59] "Glossary - OWASP." Internet: https://www.owasp.org/index.php/Glossary#S [2016-06-05]

[60] P. Gauravaram, "Security analysis of salt||password hashes," in *Advanced Computer Science Applications and Technologies (ACSAT), 2012 International Conference on*, Nov 2012, pp. 25–30.

[61] J. Chao, "Patient Tracking System Using RFID," Tesis de Maestría, Univertity of California, Los Angeles, 2007.

[62] "Products: RFID Product Datasheet - Standard Field Generator." Internet: http://www.activewaveinc.com/products_datasht_standardfieldgen.php [2016-05-28]

[63] A. Ward *et al.*, "A new location technique for the active office," *IEEE Personal Communications*, vol. 4, no. 5, pp. 42–47, Oct 1997.

[64] J. Hightower y G. Borriello, "Location systems for ubiquitous computing," *Computer*, no. 8, pp. 57–66, 2001.

[65] N. B. Priyantha *et al.*, "The Cricket Location-support System," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '00. New York, NY, USA: ACM, 2000, pp. 32–43.

[66] P. A. Catherwood *et al.*, "Cost-effective RSSI Wi-Fi positioning solution for ambulatory patient monitoring devices," *2010 Loughborough Antennas and Propagation Conference, LAPC 2010*, pp. 557–560, 2010.

[67] E. Navarro *et al.*, "Wi-Fi Localization Using RSSI Fingerprinting," BS in Computer Engineering Thesis, California Polytechnic State University, 2010. Internet: http://digitalcommons.calpoly.edu/cpesp/17

[68] "Sonitor RTLS Technologies." Internet: http://www.sonitor.com/ [2016-05-15]

[69] G. Buchan, "Real Time Location Services & Asset Management," 2009. Internet: http://www-07.ibm.com/solutions/au/healthcare/presentations/downloads/Real_Time_Location_Services_Asset_Management.pdf [2016-05-14]

[70] "How Wi-Fi RTLS Works." Internet: http://www.ekahau.com/real-time-location-system/technology/how-rtls-works [2016-05-16]

[71] D. Clarke, "Evaluation of Active RFID in the Perioperative Context," Camden, South Carolina. Internet: http://goo.gl/y4f7DB [2016-05-17]

[72] "Ekahau real-time location systems." Internet: http://www.ekahau.com/userData/ekahau/documents/solution-brochures/Ekahau-RFID-over-WiFi-RTLS-Solutions.pdf [2016-05-16]

[73] "Ubisense: Dimension4." Internet: http://ubisense.net/en/products/Dimension4 [2016-05-17]

[74] A. Jagoe, *Mobile Location Services: The Definitive Guide*. Prentice Hall, 2003, vol. 1, no. 1.

[75] "Dimension4 tool tags." Internet: http://ubisense.net/application/files/6114/6205/0887/D4_Tool_Tag_-_Fact_Sheet.pdf [2016-05-17]

[76] D. Clarke y A. Park, "Active-rfid system accuracy and its implications for clinical applications," in *19th IEEE Symposium on Computer-Based Medical Systems (CBMS'06)*, 2006, pp. 21–26.

[77] IEEE Computer Society, *Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*, 2011, vol. 2011, no. September.

[78] "KIO real time location system product sheet." Internet: http://d6qq37vs6ar1f.cloudfront.net/wp-content/uploads/2016/01/KIO-RTLS-Product-Sheet.pdf [2016-05-15]

[79] "GE Healthcare Finds Opportunity, Acquires RTLS Firm - RFID Journal." Internet: http://www.rfidjournal.com/articles/view?7062 [2016-05-18]

[80] "Asset Optimization - Hospital Operations Management - Services." Internet: http://www3.gehealthcare.com/en/services/hospital_operations_management/asset_optimization [2016-05-18]

[81] "Harmon Hospital Implements RFID to Track Assets - RFID Journal." Internet: http://www.rfidjournal.com/articles/view?2933 [2016-05-18]

[82] "Exavera Technologies Incorporated." Internet: http://www.exavera.com/healthcare/eshepherd.php [2016-05-18]

[83] "Datasheet: RoamAlert Pendant Tag." Internet: http://www.stanleyhealthcare.com/sites/stanleyhealthcare.com/files/documents/951-000081-000Rev07RoamAlertPendantTag.pdf [2016-05-18]

[84] "Datasheet: T2s Tag." Internet: http://www.stanleyhealthcare.com/sites/stanleyhealthcare.com/files/documents/T2sTagDataSheet0971-093-000RevC.pdf [2016-05-18]

[85] "Datasheet: AeroScout EX4000 Series." Internet: http://www.stanleyhealthcare.com/sites/stanleyhealthcare.com/files/documents/EX4000Exciter.pdf [2016-05-18]

[86] "STANLEY Healthcare Announces Expanded Security Solutions Offering with STANLEY Security for Hospitals, Health Systems and Senior Living Organizations | STANLEY Healthcare." Internet: http://www.stanleyhealthcare.com/company/newsroom/press-releases/stanley-healthcare-announces-expanded-security-solutions-offering [2016-05-18]

[87] "STANLEY Healthcare." Internet: http://www.stanleyhealthcare.com/ [2016-05-18]

[88] "PillPick® Automated Packaging and Dispensing System." Internet: http://www.swisslog.com/en/Products/HCS/Medication-Management-Systems/PillPick-Automated-Packaging-and-Dispensing-System [2016-05-28]

[89] P. Kruntchen, "Architectural blueprints–the "4+1" view model of software architecture," *IEEE Software*, vol. 12, no. November, pp. 42–50, 1995.

[90] Esper, "Esper Datasheet." Internet: http://www.espertech.com/download/public/EsperTechtechnicaldatasheet.pdf [2016-06-18]

[91] T. J. D. Team, "Drools documentation," pp. 1–672, 2008. Internet: https://docs.jboss.org/drools/release/6.1.0.Final/drools-docs/pdf/drools-docs.pdf

[92] "Spring Security." Internet: http://projects.spring.io/spring-security/ [2016-06-05]

[93] "Springsource Spring Framework : List of security vulnerabilities." Internet: https://www.cvedetails.com/vulnerability-list/vendor_id-9664/product_id-17274/Springsource-Spring-Framework.html [2016-06-05]

[94] "Category:OWASP Enterprise Security API - OWASP." Internet: https://www.owasp.org/index.php/OWASP_ESAPI [2016-06-05]

[95] "Owasp : Security vulnerabilities." Internet: https://www.cvedetails.com/vulnerability-list/vendor_id-4880/Owasp.html [2016-06-05]

[96] "Apache Shiro | Java Security Framework." Internet: http://shiro.apache.org/ [2016-06-05]

[97] "Apache Shiro: List of security vulnerabilities." Internet: https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-20193/Apache-Shiro.html [2016-06-05]

[98] "Apache Santuario – Index." Internet: http://santuario.apache.org/ [2016-06-05]

[99] "XML Encryption Syntax and Processing." Internet: https://www.w3.org/TR/2002/

REC-xmlenc-core-20021210/Overview.html [2016-06-05]

[100] "XML Signature Syntax and Processing (Second Edition)." Internet: https://www.w3.org/TR/xmldsig-core/ [2016-06-05]

[101] "Apache Santuario Xml Security For Java : List of security vulnerabilities." Internet: https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-31117/Apache-Santuario-Xml-Security-For-Java.html [2016-06-05]

[102] "Hibernate: Improving performance." Internet: https://docs.jboss.org/hibernate/orm/3.3/reference/en/html/performance.html#performance-cache [2016-06-05]

[103] W. Jing y R. Fan, "The research of hibernate cache technique and application of ehcache component," in *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, May 2011, pp. 160–162.

[104] D. Cherry, *Securing SQL Server: Protecting Your Database from Attackers*, 3rd ed. Elsevier Science, 2015.

[105] "JPA performance benchmark," 2012. Internet: http://www.jpab.org/MySQL.html [2016-06-15]

[106] L. Sembera, "Comparison of JPA providers and issues with migration," Disertación de Ph.D., Masaryk University, 2012.

[107] "JavaServer Pages Technology." Internet: http://www.oracle.com/technetwork/java/javaee/jsp/index.html [2016-06-05]

[108] "AngularJS." Internet: https://angularjs.org/ [2016-06-05]

[109] AGESIC, "Guía de implementación estructura mínima nacional del documento clínico HL7 V3 CDA-R2 para uso en el dominio de Salud," pp. 1–28, 2013.

[110] A. Associates, "HL7 Implementation Guide: For Simple CDA Release 2 Documents," Alschuler Associates, Tech. Rep., 2007.

[111] Health Level Seven International (HL7), "HL7 Implementation Guide for CDA § Release 2: Consolidated CDA Templates for Clinical Notes (US Realm) Volume 1 - Introductory Material," Tech. Rep., 2015.

[112] Health Level Seven International (HL7), "HL7 Implementation Guide for CDA § Release 2 : Consolidated CDA Templates for Clinical Notes (US Realm) Volume 2 — Templates and Supporting Material," vol. 2, pp. 1–64, 2015.

[113] S. Burd, *Systems Architecture*, 7th ed. Cengage Learning, 2015.

[114] "Active RFID vs. Passive RFID: What's the Difference?" Internet: http://blog.atlasrfidstore.com/active-rfid-vs-passive-rfid [2016-05-22]

[115] J. J. Carr, *Practical Antenna Handbook*, 4th ed., McGraw-Hill, Ed. McGraw-Hill, 2001.

[116] "Circular Polarization vs. Linear Polarization: Which is the right RFID Antenna?" Internet: http://blog.atlasrfidstore.com/circular-polarization-vs-linear-polarization [2016-05-23]

[117] "Near Field Communication: What is Near Field Communication?" Internet: http://www.nearfieldcommunication.org/ [2016-05-21]

[118] International Organization for Standarization-ISO, "ISO/IEC 14443 - Identification cards." Internet: http://www.iso.org/iso/catalogue_detail.htm?csnumber=50942 [2016-05-10]

[119] International Organization for Standarization-ISO, "ISO/IEC 18000-3:2010 Information technology – Radio frequency identification for item management – Part 3: Parameters for air interface communications at 13,56 MHz," 2010. Internet: http://www.iso.org/iso/catalogue_detail.htm?csnumber=53424 [2016-05-10]

[120] "EPC/RFID | GS1." Internet: http://www.gs1.org/epc-rfid [2016-05-21]

[121] GS1, "EPC Tag Data Standard," Tech. Rep., 2014.

[122] GS1, "Specification for RFID Air Interface EPC ™ Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID," Tech. Rep., 2007.

[123] "Health Level Seven International - Homepage." Internet: http://www.hl7.org/ [2016-05-11]

[124] "IHE.net." Internet: http://www.ihe.net/ [2016-05-11]

[125] AGESIC, "Funcionalidades HCEO," pp. 1–14, 2015.

[126] P. Gibbons *et al.*, "Scoping Interoperability for Health Care," HLE, Tech. Rep., 2007.

[127] G. Morris *et al.*, "Patient Identification and Matching Final Report," Audacious Inquiry, Baltimore, Tech. Rep., 2014.

[128] Health Level Seven International (HL7), "HL7 Implementation Guide for CDA ® Release 2: Consolidated CDA Templates for Clinical Notes (US Realm) Volume 2 — Templates and Supporting Material," vol. 1, pp. 1–64, 2015.

Glosario

demodular Extraer información que porta una señal.

esquema de identiftcación Conjunto de identificadores y un procedimiento para la asignación de los mismos [15].

estándar abierto Estándar de especificación pública.

experto de dominio En Ingeniería de software, se conoce como experto de dominio a la persona conocedora de los detalles del contexto y requerimientos para un sistema.

librería Conjunto de funciones u operaciones compiladas que se empaquetan en un archivo y distribuyen como una unidad.

metadato Son datos que se utilizan para describir otros datos.

motor de eventos Software que analiza y reacciona ante eventos.

probe request Mensaje enviado por un dispositivo como parte del protocolo 802.11 para descubrir redes WiFi [44].

protocolo propietario Protocolo privado.

RESTful Que implementa la especificación de la arquitectura REST.

Salud.uy Iniciativa de la AGESIC con el fin de fortalecer el Sistema Nacional Integrado de Salud (SNIS) apoyando la conformación de la red asistencial a través del uso de las Tecnologías de la Información y la Comunicación (TIC), creando herramientas que contribuyan a mejorar el acceso de los ciudadanos a servicios de salud de calidad [1].

sistema distribuido Sistema compuesto de módulos físicamente separados, que intercambian mensajes [113].

tiempo real Restricción temporal en la ejecución de tareas de un sistema.

[1] http://www.agesic.gub.uy/innovaportal/v/4422/19/agesic/que-es.html?idPadre=4425

Apéndice A

Tecnología RFID

A.1. Componentes de un sistema RFID

Los sistemas RFID se componen de tags, lectores y antenas. Adicionalmente, son necesarios elementos como cables, fuentes de alimentación y soportes para el montaje de los lectores y antenas. Deben considerarse factores ambientales: materiales como agua y metales, pueden causar interferencias o problemas en la lectura, en especial en los sistemas RFID que operan a frecuencia alta [27]. La información que sigue se basa en los trabajos de [25,27].

Tag

Un tag, o etiqueta RFID, es un dispositivo compuesto por un circuito integrado y una antena para transmitir y recibir señales de radiofrecuencia. El circuito integrado almacena la información que identifica al objeto. También puede contener un microprocesador y un sensor para recolectar y transmitir datos. Los tags se añaden a los objetos que se desean identificar, y la comunicación con el lector se efectúa enviando señales de radiofrecuencia moduladas con la información codificada.

Los tags varían en forma, tamaño, frecuencia de operación, estándares, etc. La tabla A.1 resume algunas de las características más importantes que se deben tener en cuenta para una elección correcta.

Tags pasivos

Para funcionar, los tags pasivos obtienen toda la energía del lector. La energía es transportada por las señales de radiofrecuencia emitidas por el lector, las cuales permiten energizarlo y efectuar la transmisión. La energía recibida se divide para alimentar al microprocesador, la memoria interna y el resto para la transmisión de la información. Al depender totalmente de la energía del lector, los tags pasivos tienen el rango de lectura más corto entre los tres tipos de tag. Son los más económicos, pequeños y duraderos. Su tamaño pequeño facilita la incorporación y adhesión a objetos de todo tipo, y por está razón son los más comunes en la industria.

Característica	Descripción
Kill/Disable	Permiten al lector enviar un comando para que el tag deje de funcionar permanentemente.
Write Once	Permite una única escritura del identificador. Muchas veces el identificador se introduce en en el proceso de fabricación.
Write Many	Permite múltiples escrituras del identificador.
Seguridad y encriptación	Permiten encriptar la comunicación con el lector. Algunos tags requieren de autenticación para acceder a la información.
Fuente de energía	Un tag activo dispone de una batería interna para energizarse mientras que un tag pasivo recibe la energía de las ondas emitidas por el lector. Los tags activos tienen un rango de lectura mayor que los pasivos.
Frecuencia	Los tags pueden operar a frecuencia LF, HF, UHF, SHF y UWB. Las frecuencias altas tienen un rango de lectura mayor y más sensibilidad a materiales como metales y líquidos.
Ciclo de vida	Un tag activo está limitado por la duración de la batería, mientras que un tag pasivo tiene duración ilimitada.
Forma y tamaño	Existen tags con forma de tarjeta, llavero, pulsera, y plegables. En general, los tags activos tienden a ser más grandes que los pasivos.
Material	Algunos tags soportan altas temperaturas y condiciones extremas. Otros están diseñados para adherirse a metales.

Tabla A.1: Características de los tipos de tag RFID

Tags activos

Los tags activos utilizan una batería interna como fuente de alimentación. Al no depender del lector para obtener la energía, tienen un rango de lectura y capacidad de almacenamiento mayor. La duración del tag se encuentra limitada por la batería, y tienen una duración aproximada de cinco años. Los tags activos no esperan a ser interrogados por un lector para comenzar a transmitir. Por el contrario, se caracterizan por enviar periódicamente la información. Usualmente, se utilizan en conjunto con un sensor embebido, el cual recolecta y transmite datos periódicamente; y tienen un costo más elevado que los pasivos.

Tags semiactivos

Los tags semiactivos también disponen de una batería interna, pero continúan dependiendo del lector para la transmisión de la información. La batería interna se utiliza únicamente para alimentar al microprocesador y la memoria. Están activos únicamente cuando son interrogados por un lector. Ya que la energía recibida del lector se destina totalmente a la transmisión, los tags semiactivos tienen un rango de lectura mayor que los pasivos.

Figura A.1: Comparación de tamaños entre un tag activo (azul) y uno pasivo (blanco y plateado) [114].

Características de los tipos de tags

La tabla A.2 resume las características principales de los distintos tags discutidos.

Lectores

Los lectores se comunican con los tags a través de sus antenas, emitiendo y recibiendo señales de radiofrecuencia, y son los responsables de decodificar la información recibida de los tags, la cual permite identificar a los objetos etiquetados. La comunicación se realiza utilizando protocolos generalmente estándares, como NFC o UHF EPC Gen2, y otras veces propietarios.

En ambientes donde ocurren lecturas de múltiples tags en simultáneo, los lectores deben implementar algoritmos anticolisión para evitar conflictos y serializar las lecturas.

Característica	Pasivo	Semiactivo	Activo
Fuente de energía	Lector	Batería	Batería
Comunicación	Solo responde	Solo responde	Responde y/o inicia
Rango de lectura máximo (m)	~10 m	>100 m	>100 m
Ciclo de vida	Ilimitado	Limitado por batería	Limitado por batería
Sensibilidad al ruido	Considerable	Escasa	Escasa
Tamaño relativo	Pequeño	Grande	Grande
Costo relativo	Económico	Costoso	Costoso
Variedad de materiales y formas.	Mucha	Poca	Poca

Tabla A.2: Comparativa de tipos de tag RFID

Los lectores se encuentran en distinta forma, frecuencia de operación y función ofrecida. En general, se clasifican en lectores fijos y de mano. Los primeros, usualmente se montan en lugares fijos e identifican objetos que circulan cercanos a ellos. Por otro, lado los lectores de mano son dispositivos móviles que pueden ser transportados fácilmente mientras se utilizan.

Otra característica importante del lector es la interfaz de comunicación para el envío de los capturados al computador. En general, se encuentran lectores con interfaz cableada Wiegand, RS 232 y LAN, o con interfaz inalámbrica WiFi (ver apéndice B).

Antenas

Las antenas son utilizadas por los lectores para generar campos electromagnéticos que permiten transmitir y recibir señales. Las antenas varían en tamaño, ganancia, resistencia a condiciones adversas, polaridad y tipo de conector. Los lectores pueden tener una antena integrada o conectores para antenas externas.

Ganancia

En términos simples, cuanto más ganancia tiene una antena, el campo electromagnético generado será mayor, y por lo tanto el rango de lectura será más extenso. Algunos lectores permiten regular la ganancia emitida por la antena; esto puede ser muy importante si se desea restringir el alcance de lectura.

Polarización

La polarización es una propiedad que pueden presentar las ondas transversales, como las electromagnéticas. Cuando la dirección de oscilación de una onda transversal es fija, se dice que tiene polarización lineal, al igual que la antena emisora. Al plano que determinan las direcciones de oscilación y propagación de la onda se lo llama plano de polarización. En el caso de las ondas electromagnéticas, se elige la dirección del campo eléctrico (en lugar del magnético) para determinarlo [115]. Cuando el plano de polarización rota periódicamente, se dice que tanto la onda como la antena emisora tienen polarización circular.

La polarización de la antena emisora influye directamente en la capacidad de inducción de energía

en un tag, y por tanto, en su detección. En el caso de que no se conozca a priori el tipo de antenas en los tags, o su ubicación y orientación relativas a la antena emisora, es preferible que esta última tenga polarización circular [115,116].

A.2. Frecuencia de operación

Los sistemas RFID operan a distinta frecuencia. Cada frecuencia de operación ofrece un rango de lectura distinto e impone requerimientos de energía y desempeño diferentes. El nivel de frecuencia determina los materiales por los cuales podrán propagarse las señales de radiofrecuencia. Las dimensiones físicas, formas y materiales de los tags y antenas también dependen de la frecuencia de operación. Los rangos de frecuencia (o bandas) utilizados por la tecnología RFID comprenden la frecuencia baja (LF), frecuencia alta (HF), frecuencia ultra alta (UHF), microondas o frecuencia super alta (SHF) y frecuencia de banda ultra ancha (UWB).

Frecuencia baja

Los tags LF funcionan en el rango de 120 a 140 KHz. Son pasivos y tienen un rango de lectura de hasta 10 cm. Pueden operar en la proximidad de metales y líquidos y son muy utilizados para identificación de animales, donde los tags son implantados bajo la piel. Tienen una tasa de lectura muy baja (cantidad de tags leídos por segundo).

Frecuencia alta

Los tags HF trabajan a 13.56 MHz. Son pasivos y tienen un rango de lectura de hasta 30 cm. Tienen mayor dificultad para operar entre metales y líquidos que los LF pero logran tasas de lectura mayores. Son muy utilizados para el control de acceso.

Frecuencia ultra alta

Los tags UHF operan en el rango de 868 a 928 MHz. Se encuentran tags pasivos y activos. Los pasivos logran un rango de lectura de hasta 10 m y los activos un rango de hasta 100 m. Experimentan interferencia en la proximidad de metales y líquidos. Permiten el rastreo e identificación de objetos como palés y autos, por lo que son utilizados en el control automático de inventarios y pago automático en peajes o estaciones de servicio. Alcanzan una elevada tasa de lectura.

Frecuencia super alta

Los tags que operan a frecuencia super alta están en el rango de 2.5 a 5.8 GHz. Este rango es conocido como el rango de las microondas. Los tags SHF son generalmente semiactivos y activos. Tienen una tasa y rango de lectura mayor que los tags UHF.

Banda ultra ancha

Los tags UWB operan enviando señales débiles en un rango ancho de frecuencia. Estas señales sumadas hacen que la comunicación sea muy robusta, alcanzando rangos de lectura de hasta 200 m y sin interferencias con metales y líquidos. Los tags UWB son semiactivos o activos.

Resumen de frecuencias de operación

Banda	Rango de frecuencia	Rango de lectura máximo	Sensibilidad a los materiales	Tasa de lectura
LF	120 – 140 KHz	~10 cm	No	Baja
HF	13.56 MHz	~30 cm	Sí	Baja
UHF	868 – 928 MHz	~10 m (pasivo) >100 m (activo)	Sí	Alta
SHF	2.5 – 5.8 GHz	>10 m (pasivo) >100 m (activo)	Sí	Alta
UWB	3.1 – 10.6 GHz	>200 m (activo)	No	Alta

Tabla A.3: Comparativa de los rangos de frecuencia de RFID

A.3. Estándares RFID

En esta sección se describen dos de los estándares más importantes en la tecnología RFID: el estándar NFC que opera a una frecuencia HF, y el estándar EPCGlobal UHF Gen2, que lo hace en UHF.

Near Field Communication

Near Field Communication (NFC) es un protocolo de comunicación por radio frecuencia, de corto alcance, diseñado para el intercambio de información entre dispositivos [117]. Los dispositivos NFC operan en HF, y tienen un rango de lectura menor a 20 cm, según la referencia previa. Pueden ser pasivos o activos. Los activos intercambian información con tags u otros dispositivos activos. Por otro lado, los dispositivos pasivos, como un tag, únicamente almacenan información que es consultada por los dispositivos activos.

La tecnología NFC está definida por los estándares ISO/IEC 14443 [118] e ISO/IEC 18000-3 [119]. El estándar define dos tipos de comunicación: *Type A* y *Type B*.

Para funcionar, el dispositivo lector interroga al tag enviando una señal de radiofrecuencia. Si los dispositivos están lo suficientemente cercanos, el tag será energizado y podrá responder al lector. El primer mensaje que envía el lector pregunta por la comunicación soportada por el tag, que puede ser *Type A* o *Type B*. Una vez que el tag envía su respuesta, el lector comienza a enviar los primeros comandos en la especificación adecuada. Cuando el tag recibe un comando, verifica si es válido, y en caso afirmativo responde con la información solicitada.

Existen cuarto tipos de tags NFC:

- **Type 1**: puede configurarse para ser de solo lectura o lectura/escritura. Tiene una memoria de 96 bytes expandible a 2 KB. Es muy utilizado para almacenar URLs. La tasa de comunicación es de 106 Kb/s.
- **Type 2**: es similar al *Type 1*, pero con memoria de 48 bytes, expandible a 2 KB.
- **Type 3**: puede configurarse para ser de solo lectura o lectura/escritura. Tiene hasta 1 MB de memoria y una tasa de comunicación entre 212 y 424Kb/s.
- **Type 4**: soporta ambos tipos de comunicación: *Type A* y *Type B*. Puede configurarse para ser de solo lectura o lectura/escritura. Tiene hasta 32 KB de memoria y una tasa de comunicación de 424Kb/s.

EPCGlobal UHF Gen2

EPCGlobal UHF Gen2 es un protocolo de comunicación por radiofrecuencia, de largo alcance, desarrollado por GS1 [120] para la identificación e intercambio de información de objetos físicos utilizando tags RFID Gen2. Opera en la banda UHF y alcanza un rango de lectura de 3 a 6m. La especificación del estándar se divide en los documentos *Tag Data Standard* (TDS) [121] y *UHF Gen2 Air Interface* [122].

El documento TDS define el *Electronic Product Code* (EPC), un identificador universal que permite la identificación unívoca de cualquier tipo de entidad. La asignación de un código EPC es permanente y preservada, incluso si la entidad deja de existir. No se recomienda reutilizar códigos EPC que ya fueron asignados. Los códigos EPC son comúnmente empleados por sistemas de información que deben llevar un seguimiento de objetos físicos que se desplazan constantemente por distintos lugares o zonas.

Por otro lado, el documento *UHF Gen2 Air Interface* define la comunicación entre el lector y el tag, incluyendo la capa física, algoritmos anticolisión, comandos, tipos de mensajes y métodos de modulación. Define cuatro clases de tag RFID Gen2:

- **Clase 1**: tag pasivo de solo lectura con código EPC asignado de fabrica. Soporta función *kill* para dejar inactivo el tag.
- **Clase 2**: tag pasivo de lectura y escritura. Extiende al de nivel 1 mediante el agregado de soporte para reescrituras del código EPC, control de autenticación y memoria libre para agregar datos del objeto que sean de interés.
- **Clase 3**: tag semiactivo de lectura y escritura. Extiende al de nivel 2 y agrega una batería como fuente de energía.
- **Clase 4**: tag activo de lectura y escritura. Extiende al de nivel 3 y admite comunicación tag a tag.

Apéndice B

Comparativa de precios de tecnologías estudiadas

Se presenta a continuación una serie de tablas con precios relevados de varios proveedores locales e internacionales. Los valores indicados no incluyen costos de envío, ni impuestos o trámites de importación, aunque fueron tomados en cuenta para decidir las tecnologías a utilizar.

Mercado	Modelo	Origen	Conector antena	Interfaz	Costo (USD)
Uruguay	Motorola FX 9500	EE.UU.	4 conectores	Ethernet, Serial	5000
	RU5100	Chino	1 conector	Serial	985
Argentina	Motorola FX 9500	EE.UU.	4 conectores	Ethernet, Serial	2864
	Genérico	Chino	Antena inlc.	Serial	1799
EE.UU.	Motorola FX 9500	EE.UU.	4 conectores	Ethernet, Serial	2085
	ThingMagic M6	EE.UU.	4 conectores	Ethernet, WiFi	1440
China	Safe&Sound SS-UF06W	China	Antena incl.	Serial, WiFi	266
	Safe&Sound SS-UF08	China	4 conectores	Ethernet, Serial	554

Tabla B.1: Costos de lectores UHF pasivos

Mercado	Modelo	Interfaz	Lector cód. de barras	Costo (USD)
Uruguay	Motorola MC3190-Z	WiFi, Bluetooth	No	4270
EE.UU.	TSL 1128	USB, Bluetooth	Sí	1639
	Zebra DS9808-R	USB, RS-232, RS-485	Sí	1638
	Alien ALH-9011	Wi-Fi, Bluetooth, USB	Sí	3224
	Invengo XC-2903	WiFi, GSM/GPRS, USB	Sí	2195

Tabla B.2: Costos de lectores manuales UHF pasivos

Mercado	Modelo	Origen	Polarización	Costo (USD)
Argentina	Genérico	China	Circular	225
EE.UU.	Rfmax 9025PL/8655PL	EE.UU.	Circular	125
	Rfmax PAL90209H	EE.UU.	Circular	135
China	Invelion YR9028	China	Circular	65
	Vikitek, V	China	Circular	100

Tabla B.3: Costos de antenas de RFID

Mercado	Modelo	Origen	Tipo	Costo (USD) 100 unidades
Uruguay	Alien ALN-9730	EE.UU.	Etiqueta	180
EE.UU.	Smartrac Belt Monza 5	EE.UU.	Etiqueta	97
	Alien/Higgs3	EE.UU.	Tarjeta	79.9
China	Kingjoin	China	Tarjeta	19.08
	Impinj H47	China	Etiqueta	13

Tabla B.4: Costos de tags pasivos UHF EPC Gen2

Mercado	Modelo	Interfaz	Costo (USD)
Uruguay	Symbol LS4278	Bluetooth	159
	Symbol LS4004i	PS2	43
	Netum NT-2028	USB	79
China	Omark	USB (inalámbrico)	13.20
	ALANDA CT007S	USB (inalámbrico)	30.50
	Juxing	USB (inalámbrico)	31.33

Tabla B.5: Costos de lectores de códigos de barras

Mercado	Modelo	Origen	Interfaz	Costo (USD)
Uruguay	ZK u560id	EE.UU.	USB	313
EE.UU.	TapTrack Tappyble	EE.UU.	USB, WiFi, Bluetooth	225
	SCM SCL011	EE.UU.	USB	49.99
China	LH-ICR1	China	USB	13.18
	Ehuoyan ER30302	China	USB	25.48

Tabla B.6: Costos de lectores NFC

Mercado	Modelo	Origen	Tipo	Costo (USD) 100 unidades
EE.UU.	Genérico	China	Tarjeta	77
China	Genérico	China	Tarjeta	22
	WSD-MW1	China	Tarjeta	59.99

Tabla B.7: Costos de tags NFC

Apéndice C

Gastos para el desarrollo del prototipo

La tabla C.1 muestra los gastos realizados en insumos y trámites necesarios para el desarrollo del prototipo.

Descripción	**Monto (USD)**
Despachante de aduana para sensores RFID	228
Etiquetas adhesivas 46 mm * 63 mm (caja de 1800 unidades)	27
Rotuladora Brother QL-700	220
2 sensores RFID marca Safe & Sound + Tarjetas RFID (100 unidades)	648
Homologación de sensores RFID en URSEC	17
Trámites para permiso de importación en VUCE	5
Tablet Asus Nexus 7	397
Gasto total	1542

Tabla C.1: Gastos necesarios para el desarrollo del prototipo

Apéndice D

Pruebas basadas en casos de uso

Aquí se muestran algunos ejemplos de los casos de prueba más relevantes diseñados en base a los casos de uso relevados.

Escenario	Ingresa todos los datos obligatorios	Ingresa atributos identiftcadores únicos	Ingresa todos los campos con formato válido	Resultado esperado
a	V	V	V	OK
b	V	F	V	ERROR
c	V	V	F	ERROR
d	F	V	V	ERROR
e	F	F	V	ERROR
f	V	F	F	ERROR
g	F	V	F	ERROR
h	F	F	F	ERROR

Tabla D.1: Casos de prueba para *Nuevo paciente*. a) El paciente se registra exitosamente; b) Ya existe un usuario con los atributos identificadores ingresados; c) Se ingresa algún campo con formato inválido; d) No ingresa un dato obligatorio; e) No ingresa un dato obligatorio e ingresa un atributo identificador repetido; f) Ingresa un atributo identificador repetido y un campo con formato inválido; g) No ingresa un dato obligatorio e ingresa un campo con formato inválido; h) No ingresa un campo obligatorio, ingresa un atributo identificador repetido y un campo con formato inválido

Escenario	Iden. del paciente exitosa	Iden. de unidosis exitosa	Unidosis corr. a paciente	Unidosis corr. al turno actual	Hay med. sobrante	Hay conex. con XDS	Res. esperado
a	V	V	V	V	F	V	OK. Se genera CDA
b	V	V	V	V	V	V	OK. Se genera CDA y evento de med. sobrante
c	F	N/A	N/A	N/A	N/A	N/A	ERROR
d	V	F	N/A	N/A	N/A	N/A	ERROR
e	V	V	F	V	N/A	N/A	ERROR. Se genera evento complejo.
f	V	V	V	F	N/A	N/A	ERROR. Se genera evento complejo.
g	V	V	V	V	F	F	OK. Se registra error de conexión.

Tabla D.2: Casos de prueba para *Administración de unidosis*: a) Administración correcta de unidosis a paciente. Sin medicación sobrante; b) Administración correcta de unidosis a paciente. Medicación sobrante; c) Lectura incorrecta de código de barras de paciente; d) Lectura incorrecta de código QR de unidosis; e) Unidosis no corresponde a paciente; f) Unidosis no corresponde a turno actual; g) Se pierde conexión con XDS.

Escenario	XML enviado con errores	XML enviado válido	Resultado esperado
a	F	V	OK. Se procesa el evento
b	V	N/A	El sistema retorna un código de error.
c	F	F	El sistema retorna un código de error.

Tabla D.3: Casos de prueba para *Enviar evento primitivo*: a) Escenario de éxito. Se envía un XML sintácticamente correcto que corresponde a un evento primitivo válido; El XML envíado contiene errores sintácticos; c) El XML enviado no corresponde a un evento primitivo válido.

Apéndice E

Estándares de informática médica

E.1. Organizaciones desarrolladoras de estándares

En esta sección se describen las principales organizaciones dedicadas al desarrollo de estándares en salud informática que permiten la interoperabilidad de los sistemas de información.

Health Level Seven

Health Level Seven (HL7) es una organización sin fines de lucro dedicada al desarrollo de estándares para el intercambio, integración y recuperación de información clínica electrónica [123]. Define cómo debe ser empaquetada y comunicada la información, especifica la mensajería a intercambiar, y los tipos de datos utilizados.

Integrating the Healthcare Enterprise

Integrating the Healthcare Enterprise (IHE) es una iniciativa tomada por diferentes prestadores de salud y profesionales de la industria con el objetivo de mejorar y optimizar la forma en la que los sistemas de información de salud intercambian información [124]. Promueve la adopción de estándares ya establecidos (HL7, DICOM) para solucionar necesidades clínicas específicas y lograr un cuidado óptimo del paciente. Para esto, IHE define un conjunto de perfiles de integración donde cada uno propone una solución a un caso de uso particular. Los perfiles son agrupados en dominios, siendo el más importante en el contexto de este trabajo, el de infraestructura tecnológica (*IT Infrastructure Domain*). En él se proponen soluciones al problema de identificación de pacientes (perfil PIX) y el de intercambio de documentos clínicos (perfil XDS.b).

E.2. Mensajería HL7

El estándar de mensajería HL7 [17] define un conjunto de transacciones y mensajes para el intercambio de información clínica. La mensajería HL7 ha sido adoptada por un gran número de estándares como protocolo de comunicación e intercambio de información. Entre las transacciones

más frecuentes se encuentran: *Patient Administration Transaction* (ADT), utilizada para transmitir información demográfica de pacientes nuevos, o cuyos datos han sido modificados; y *Query By Paramater* (QBP), que permite obtener información demográfica de un paciente ya registrado.

Aquí se describen únicamente los mensajes ADT y QBP utilizados por el perfil PIX, que será descrito más adelante.

Los mensajes HL7 se componen de segmentos, y a su vez cada uno de un conjunto de campos delimitados por el caracter |, que conforma los datos del mensaje. Entre los segmentos más comunes se encuentran:

- **Message Header (MSH)**: describe el tipo de mensaje (ADT04, ADT08, QBP22, etc.), la versión del estándar, la fecha de creación del mensaje, y los nombres de las instituciones y sistemas que participan en la transacción.
- **Patient Identiftcation (PID)**: contiene la información del paciente, entre ella, nombres y apellidos, identificadores locales, sexo, fecha de nacimiento y domicilio.
- **Query Parameter Deftnition (QPD)**: contiene información de la consulta. Las consultas pueden realizarse a partir del identificador o nombre del paciente.

Mensaje ADT 04

Notifica la llegada de un nuevo paciente a la institución de salud. Puede ser utilizado para propagar la información del paciente de un sistema de la institución a otro, o para almacenarla en un MPI compartido por varias instituciones.

MSH|^~\|INST1_SERVICES|INST1|EMPI_SERVICES|EMPI|201509091452||ADT^A04^ADT_A01|360-571b-11e5-9990-0056010825|P|2.3.1
PID|||0000^^^INST2&2.16.858.0.1&ISO||BLINN^LAURA||19500202|F|||Av.Italia 1347|||||||44922331

Figura E.1: Ejemplo de mensaje ADT 04

Mensaje ADT 08

Notifica cambios en la información de un paciente. Puede ser utilizado para propagar los cambios de un sistema a otro o para actualizar los datos en el MPI.

MSH|^~\|INST1_SERVICES|INST1|EMPI_SERVICES|EMPI|201509091452||ADT^A04^ADT_A01|360-571b-11e5-9990-0056010825|P|2.3.1
PID|||0000^^^INST2&2.16.858.0.1&ISO||BLINN^LAURA||19500202|F|||Soca 1112|||||||44922331

Figura E.2: Ejemplo de mensaje ADT 08

Mensaje QBP 23

Se utiliza para consultar información demográfica de un paciente a partir de su identificador. Puede ser utilizado por un sistema para consultar información de pacientes que se encuentran registrados en otros sistemas o en un MPI.

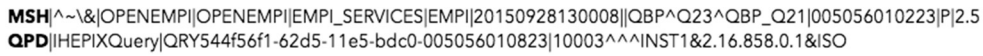


Figura E.3: Ejemplo de mensaje QBP 23

Mensaje QBP 22

Es análogo al mensaje QPB 23, pero la consulta se realiza a partir del nombre del paciente.

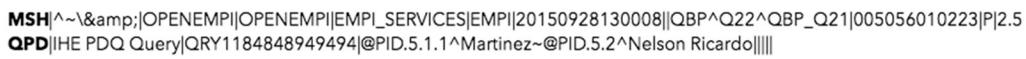


Figura E.4: Ejemplo de mensaje QBP 22

E.3. Historia Clínica Electrónica

La salud es un elemento clave en nuestra sociedad, siendo cada vez más objeto de discusión y de políticas estatales que intentan mejorar la calidad de los servicios de atención. Hoy día existen múltiples prestadores de salud, tanto en el ámbito privado como público. Algunos de ellos forman una red de servicios que se extiende a nivel nacional, con diversos centros de atención distribuidos en todo el país.

Cada vez más son las instituciones y centros especializados que deben intercambiar información entre sí, con el fin de cooperar y lograr un diagnóstico temprano y efectivo. La diversidad de prestadores de salud y heterogeneidad de los sistemas informáticos involucrados hacen que este intercambio de información se torne cada vez más complejo. Para una mejor atención médica es fundamental lograr acceder de manera oportuna a los datos clínicos de pacientes, para así poder tomar mejores decisiones, y reducir riesgos o demoras innecesarias.

En este contexto, el objetivo de la Historia Clínica Electrónica (HCE) es mejorar la atención médica del paciente, integrando la totalidad de la información clínica de las instituciones que participan en el proceso asistencial, independientemente de su localización geográfica e institución de atención, dando así una visión integral del paciente [125]. La HCE comprende la representación, almacenamiento, procesamiento e intercambio de datos clínicos, sociales y financieros referidos a la salud de una persona a través de medios informáticos.

Algunos beneficios a destacar de la historia clínica electrónica son:

A los pacientes:

- Mejora del proceso de asistencia.
- Continuidad asistencial.
- Información clínica accesible en el momento oportuno.

A los médicos:

- Legibilidad de la historia clínica.
- Vista integral del paciente.

- Terminología estandarizada.
- Acceso a la historia completa del paciente, como son imágenes, informes, diagnósticos, etc.

A las instituciones:

- Uso más eficiente de los recursos asistenciales.
- Posibilidad de realizar estadísticas de enfermedades.
- Explotar el uso de datos clínicos.

Desde el punto de vista tecnológico los sistemas que integran la HCE deben poder interoperar, y adherirse a estándares para el intercambio y uso de la información. El grupo de interoperabilidad del HL7 define tres tipos de interoperabilidad que deben cumplir los sistemas de salud informáticos [126]:

- **Interoperabilidad técnica**: dos sistemas A y B pueden comunicarse e intercambiar información. Los datos enviados entre los sistemas A y B llegan en orden y sin errores.
- **Interoperabilidad semántica**: asegura que dos sistemas A y B entienden y son capaces de interpretar y utilizar sin ambigüedad los datos intercambiados. Implica el uso de códigos, identificadores y terminología común, así como la adopción de estándares.
- **Interoperabilidad en los procesos**: los procesos de negocio de dos instituciones A y B pueden trabajar en conjunto y coordinarse entre sí.

E.4. Estándares para la identificación de pacientes

En el proceso de asistencia médica los pacientes suelen transitar por varios prestadores de salud y centros especializados. En cada uno de ellos son registrados e identificados con un identificador local. Típicamente entre los datos registrados se incluye información demográfica como documento de identidad, nombres, apellidos, fecha de nacimiento, sexo y domicilio. La información solicitada por cada prestador de salud suele ser distinta y la forma de representarla también (por ejemplo, para fechas de nacimiento se pueden dar ambas representaciones: 1 de diciembre de 1981 o 1/12/81).

La necesidad de intercambiar información entre los distintos prestadores surge naturalmente con el objetivo de tener una visión integrada del estado y acciones realizadas sobre un paciente. Para esto es necesario contar con un método preciso de identificación de pacientes, independiente de los identificadores locales, que asegure una asociación correcta entre el paciente y su información clínica (estudios, resultados de laboratorio, etc). De lo contrario pueden ocurrir intercambios de información incorrecta, o con datos faltantes, teniendo como consecuencia diagnósticos errados, estudios clínicos duplicados y pérdidas financieras, corolario de una operativa ineficiente.

E.4.1. Algoritmos de referencias cruzadas

Como se anticipó en la sección anterior, un mismo paciente suele tener múltiples registros o identificadores entre los sistemas de cada institución, y debe ser posible determinar cuáles corresponden a la misma persona. Esta correspondencia entre registros o identificadores se conoce como

el problema de referencias cruzadas, y se emplean algoritmos que agrupan los registros de una misma persona. Estos operan sobre un conjunto de atributos que son considerados representativos, y que permiten distinguir una persona de otra. Algunos de los más utilizados son: documento de identidad, nombre, apellido, fecha de nacimiento y sexo. Es deseable que los atributos seleccionados se mantengan estables y varíen lo menos posible con el tiempo.

Si bien se podría pensar que es suficiente utilizar únicamente el documento de identidad como atributo, esto no es siempre posible. Existen países donde una persona puede utilizar más de un documento de identidad, un ejemplo de esto son los Estados Unidos, donde un paciente puede ser registrado en un hospital por su número de seguridad social (SSN por sus siglas en inglés) y en otro por su número de carné de conducir [127].

Existen distintos tipos de algoritmos de correspondencia: los puramente deterministas (debe haber correspondencia caracter a caracter en todos los atributos), los deterministas con ponderación (correspondencia caracter a caracter, pero con atributos con más relevancia más que otros) y probabilísticos (tienen en cuenta errores en el ingreso de datos, de deletreo y otros más).

En general, para que un algoritmo de correspondencia sea eficaz, debe contemplar los siguientes casos:

- **Variabilidad en los atributos**: los sistemas utilizan diferentes atributos para representar a las personas, por ejemplo: un sistema puede utilizar el domicilio, mientras que otro el código postal.
- **Variabilidad en la representación**: los sistemas representan los atributos de diferentes formas, por ejemplo: fechas, identificadores con guiones, nombres compuestos, etc.
- **Errores en la transcripción**: errores de digitación, por ejemplo: introducción y transposición de caracteres, faltas de ortografía, etc. Errores de interpretación al escuchar.
- **Falta de información**: no siempre se conocen los datos requeridos, por ejemplo: una persona accidentada de la cual se conoce solo su nombre y apellido y debe ser registrada en el servicio de emergencia.
- **Información obsoleta**: los datos dejan de ser válidos, por ejemplo: domicilios o apellidos de matrimonio.
- **Información inconsistente**: información contradictoria, por ejemplo: dos registros de una misma persona con fechas de nacimiento distintas.
- **Registros duplicados**: una persona registrada más de una vez en el mismo sistema.

Un algoritmo de referencias cruzadas debería ser capaz de agrupar los registros, como muestra la tabla E.1.

En la tabla E.1 puede observarse para el primer paciente un registro duplicado en la institución A y fechas de nacimiento en formatos diferentes.

Por otro lado, la institución B ha ingresado su documento de identidad con un error en el último dígito. Para el segundo paciente puede verse que la institución B no dispone de su domicilio y que la institución C ha escrito mal el apellido.

El tercer paciente es extranjero y ha sido registrado por su SSN en la institución C y D y por su

número de carné de conducir en la institución E. También puede observarse que la institución D ha escrito mal el apellido.

En el cuarto paciente puede verse una inconsistencia en la fecha de nacimiento que registran las instituciones E y F. También puede verse que la institución G desconoce su documento de identidad. Para el último paciente se da que uno de los domicilios es obsoleto.

E.4.2. Perfil IHE PIX

Patient Identifier Cross-referencing (PIX) [15, 16] es un perfil del IHE cuyos objetivos son las instituciones que tengan necesidad de soportar referencia cruzada de identificadores de pacientes entre múltiples dominios de identificación. Un dominio de identificación se define como un conjunto formado por uno o varios sistemas interconectados que comparten un esquema de identificación. En general cada institución tendrá su propio dominio de identificación, y sus pacientes tendrán asignados identificadores únicos dentro de ese dominio. El perfil define los siguientes actores principales:

- **Patient Identity Source**: responsable de registrar o modificar información de los pacientes en el *Patient Identifier Cross-reference Manager*.
- **Patient Identifter Cross-reference Manager**: responsable de crear y mantener las referencias cruzadas de los pacientes registrados por los *Patient Identity Source*.
- **Patient Identifter Consumer**: realiza consultas al *Patient Identifier Cross-reference Manager* para determinar los identificadores de un paciente en los diferentes dominios de identificación.

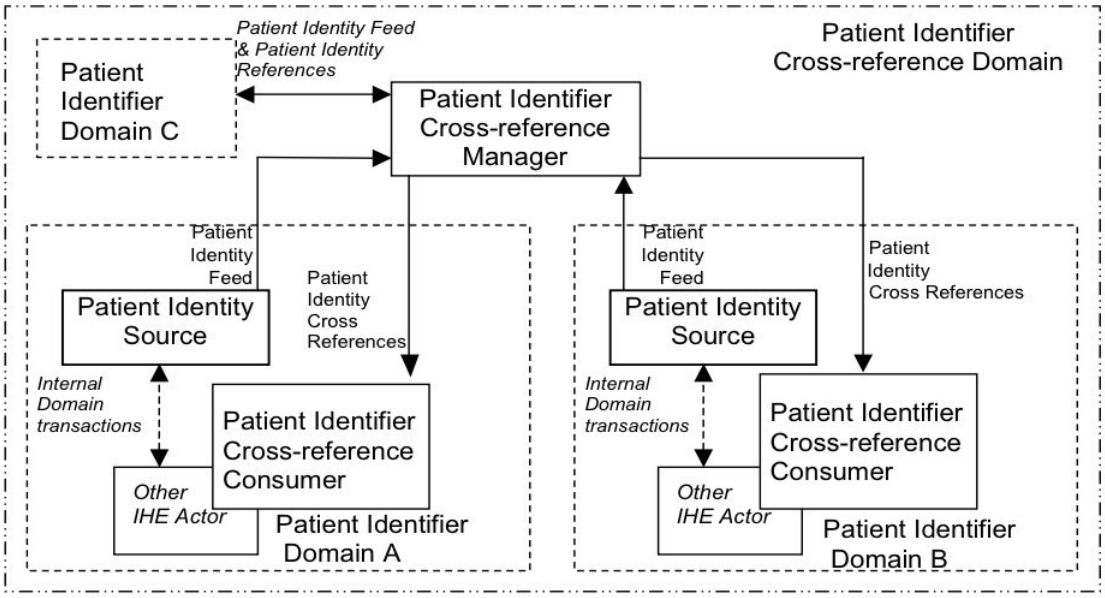


Figura E.5: Actores del perftl PIX. Las instituciones A, B y C utilizan un *Cross Reference Manager* para el manejo de referencias cruzadas. De esta forma logran intercambiar correctamente información de sus pacientes [15].

Las transacciones definidas por el perfil emplean la mensajería HL7 ADT y QBP, discutidas anteriormente y entre las principales se encuentran:

- **Patient Identity Feed**: iniciada por un *Patient Identity Source*, permite registrar o modificar información de un paciente en el *Patient identifier Cross-reference Manager*. Esta transacción se genera cuando se da de alta un paciente nuevo o se realizan modificaciones sobre uno ya registrado. La transacción incluye la información del paciente, entre ellas, el identificador local en la institución que inicia la transacción y el dominio de identificación. Una vez que el *Patient Identifier Cross-reference Manager* recibe los datos de la transacción, comienza el procedimiento para determinar si el identificador puede ser cruzado con otros de dominios diferentes.

- **Pix Query**: iniciada por un *Patient Identifier Consumer*, permite consultar a un *Patient Identifier Cross-reference Manager* por la lista de identificadores de un paciente en todos los dominios de identificación en los que está registrado. La institución que realiza la consulta envía como datos el identificador del paciente en su dominio de identificación y el *Cross-reference Manager* retorna la lista de identificadores en el resto de los dominios. De esta forma una institución A podrá acceder a los estudios del paciente P_a que se ha realizado en otra institución B, en la cual se identifica como P_b. Para esto, A realiza una consulta al manejador de referencias cruzadas, enviando el identificador P_a. El manejador retorna los identificadores de P en el resto de las instituciones, en particular P_b. De esta forma A puede consultar a B por los estudios realizados al paciente P_b.

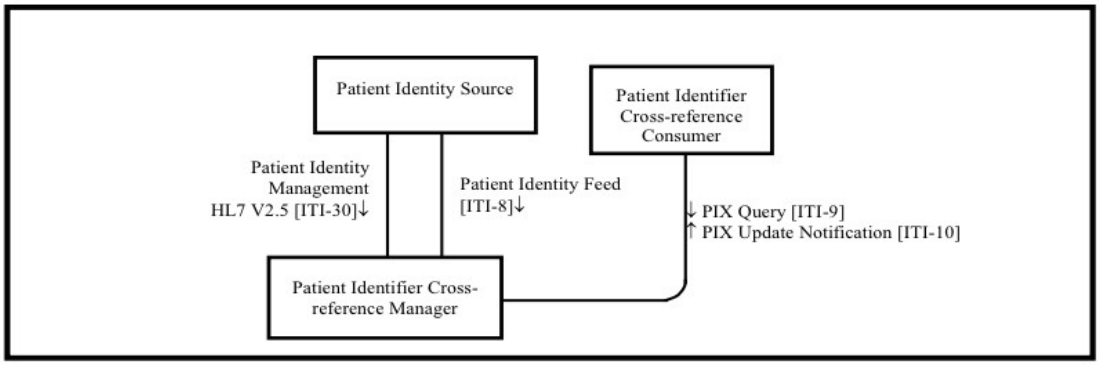


Figura E.6: Transacciones del perftl PIX [15].

E.4.3. Master Patient Index (MPI)

Otra solución comúnmente empleada para la identificación de pacientes es el *Master Patient Index* (MPI) o registro maestro de pacientes. Un MPI es un repositorio de pacientes que actúa como un *Patient Identifier Cross-reference Manager*, manteniendo agrupados los registros que se corresponden entre sí. Esta agrupación la realizan ante la presencia de errores y omisión de atributos. Para esto, implementan algoritmos de correspondencia, y generalmente el usuario tiene la posibilidad de especificar los atributos a ser tenidos en cuenta, y las relevancias de cada uno de ellos. La diferencia principal entre un MPI y un manejador de referencia cruzadas (tal como es definido por el perfil PIX) es que un MPI, además de agrupar registros, asigna un identificador único y global a cada paciente. En este sentido, un MPI no es más que un caso particular de un manejador de referencias PIX, donde la diferencia se encuentra en que el MPI define un dominio de identificación maestro o global al que pertenecen todos los pacientes. Los identificadores de un paciente en los distintos dominios van a tener una referencia cruzada con el identificador global en el dominio maestro.

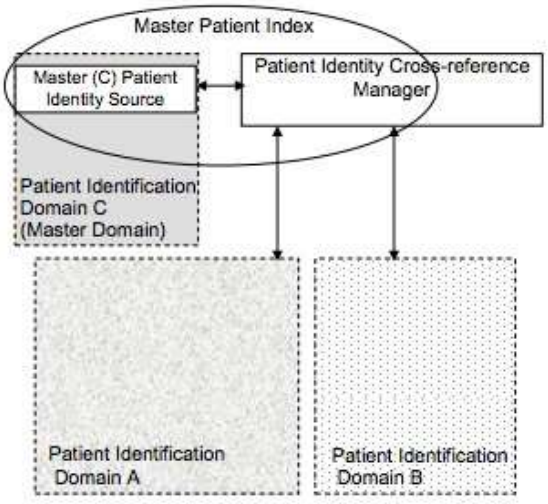


Figura E.7: Diagrama de arquitectura de un MPI [128].

E.5. Estándares de documentos clínicos

El elemento fundamental de la HCE son los documentos electrónicos que describen los actos clínicos o eventos de salud registrados por las instituciones de salud, y que pueden ser compartidos o intercambiados entre ellas.

E.5.1. HL7 CDA R2

El estándar HL7 CDA R2 [17] define la estructura y semántica de documentos clínicos con el propósito de que estos sean intercambiables. Los documentos CDA son archivos XML que constan de un encabezado y un cuerpo. El encabezado contiene un conjunto de metadatos que establecen información básica y de contexto como son el tipo de documento, autor, persona a la que aplica, fecha y lugar de creación. El cuerpo describe el acto clínico que da lugar al documento y puede variar en el nivel de estructuras y codificaciones que se utilizan. El estándar define tres niveles de CDA y cada uno introduce un nivel de interoperabilidad semántica mayor:

- **CDA Nivel 1**: Contiene un encabezado básico con metadatos y un cuerpo de contenido no estructurado. El cuerpo contiene un bloque narrativo que describe el acto clínico y puede ser texto plano, un documento con formato pdf, o una imagen.
- **CDA Nivel 2**: Mismo encabezado que el de nivel 1, pero se diferencia en que admite cuerpos estructurados. Para mantener compatibilidad también admiten los no estructurados. Los cuerpos estructurados se componen de secciones, y cada una contiene un identificador que define el tipo de sección y un bloque narrativo que la describe. Ejemplos de secciones frecuentemente utilizadas son: sección de procedimientos clínicos realizados, sección de medicación administrada, sección de alergias e intolerancias.
- **CDA Nivel 3**: Extiende al de nivel 2, permitiendo que cada sección contenga además del bloque narrativo elementos estructurados denominados ítems clínicos, que puedan ser procesados automáticamente por una computadora.

Encabezado del CDA

El encabezado es el elemento común de los tres niveles de CDA definidos. Su propósito es brindar información acerca del propio documento.

Los metadatos del encabezado pueden ser clasificados por categoría de acuerdo a su propósito en: los que describen el tipo de documento, los que describen a los participantes, y los que describen dónde y cuándo fue creado. En general son utilizados como registros en los índices de las bases de datos para efectuar búsquedas y poder recuperarlos.

A continuación se describen los metadatos más importantes agrupados por categoría.

Metadatos que describen el tipo de documento

El tipo de documento es definido por el elemento <*code*> del encabezado. Ejemplos de tipo de documentos utilizando la codificación LOINC son *11502* (Informe de Laboratorio) y *34848-2* (Hoja de cirugía).

Por otro lado el elemento <*id*> contiene el identificador del documento y generalmente es un OID[1].

Metadatos que describen los participantes

Todos los documentos CDA tienen al menos tres participantes: el autor del documento, el paciente involucrado en el acto médico, y la institución donde ocurre.

El elemento <*recordTarget*> contiene la información del paciente como su nombre, apellido y su identificador local en la institución.

El elemento <*autor*> es análogo al del paciente, pero contiene la información del autor o responsable del documento.

Por último, el elemento <*custodian*> tiene la información de la institución, como el nombre y su identificador global (por ejemplo, OID de la institución).

Metadatos que describen fechas

La fecha en la cual se genera el documento se establece en el elemento <*effectiveTime*>. Se debe respetar el formato de fecha definido por el estándar CDA, que corresponde al ISO 8061.

Por otro lado el elemento <*time*> define la fecha en la cual el autor aprueba o firma el documento y generalmente coincide con el valor de <*effectiveTime*>.

Cuerpo del CDA

El elemento <*nonXMLBody*> se utiliza cuando el cuerpo del documento no es estructurado y puede contener cualquier dato que sea legible como documentos de texto (txt, rtf, html o pdf) o imágenes (gif, jpeg, png, tiff).

Por otro lado el elemento <*structuredBody*> define un cuerpo estructurado, y consta de uno o más

[1] Un OID es un identificador global de objetos y se define de acuerdo a una asignación jerárquica, establecida por la ISO. Ver http://unaoid.gub.uy/

elementos <*section*>. Cada sección contiene el elemento <*text*>, que corresponde al bloque narrativo de la sección, y en él se describe su contenido. Por ejemplo, un documento CDA que describa una intervención quirúrgica podrá contar con secciones para describir la anestesia administrada, las complicaciones surgidas, el preoperatorio, el procedimiento en sí mismo y el posoperatorio. Un documento CDA que describe la administración de medicación a un paciente tendrá una sección que detalla la medicación administrada.

A partir de la definición del CDA de nivel 3, las secciones pueden contener un conjunto de elementos <*entry*> que definen un ítem de información clínica dentro de la sección. Por ejemplo, la sección de anestesia puede contener por cada medicamento administrado un <*entry*> o ítem <*substanceAdministration*> que describe en detalle la dosis, vía de administración, y el medicamento administrado.

Templates CDA

Un template CDA define un tipo de documento específico para un caso de uso o acto clínico particular. Imponen la estructura del documento, como elementos del encabezado, secciones del cuerpo y entradas de cada sección. Se han desarrollado guías de implementación y plantillas [109–112] que pueden ser utilizados o tomados como base para el desarrollo de nuevos documentos.

E.5.2. Perfil IHE XDS.b

Cross-Enterprise Document Sharing (XDS.b) [15, 16, 18, 19] es un perfil de integración del IHE que facilita el registro, intercambio y acceso de información clínica entre distintos prestadores de salud. El estándar define cómo se lleva a cabo el intercambio de documentos XDS entre cualquier institución.

El concepto de documento XDS es sumamente amplio y no se limita únicamente a documentos de texto. Cualquier tipo de información clínica como: imágenes (JPEG, DICOM), textos planos, y documentos CDA pueden ser un documento XDS.

Los documentos son organizados en carpetas, y tienen un conjunto de metadatos estandarizados que permiten realizar búsquedas sobre ellos. Entre los metadatos más comunes se encuentran: identificador del paciente, tipo de documento, autor y lugar. El estándar define los actores y transacciones involucradas. Se definen 4 actores con responsabilidades bien separadas:

- **Document Source**: institución que produce y publica documentos. Es la responsable de enviar los documentos al *Document Repository*.
- **Document Repository**: almacena los documentos y registra los metadatos en el *Document Registry*. Los documentos son almacenados de manera transparente y segura, y se les asigna un identificador único que permite a los *Document Consumer* recuperarlos.
- **Document Registry**: almacena los metadatos de cada documento registrado por un *Document Repository*. Esto incluye un enlace al repositorio donde se encuentra el documento. Responde a búsquedas y consultas de los *Document Consumer*.
- **Document consumer**: realiza búsquedas y consultas al *Document Registry*, y recupera los documentos de los *Document Repository*.

Separar los documentos de sus metadatos ofrece la flexibilidad de tener un único *Document Registry* centralizado compartido por múltiples *Document Repository*. Este esquema promueve la privacidad de la información donde cada institución será el repositorio de sus propios documentos, registrándolos y compartiéndolos de acuerdo a sus políticas de seguridad.

Las transacciones definidas por el estándar son las siguientes:

- **Provide And Register Document Set**: iniciada por un *Document Source* permite almacenar un conjunto de documentos y sus metadatos en un *Document Repository*. El repositorio es el responable de persistir el documento y de reenviar sus metadatos al *Document Registry* a través de la transacción *Register Document Set*.
- **Register Document Set**: iniciada por un *Document Repository* permite alamacenar los metadatos de un documento en el *Document Registry*. El registro crea una nueva entrada con los metadatos y asigna el enlace al repositorio.
- **Registry Stored Query**: iniciada por un *Document Consumer* permite realizar consultas sobre documentos al *Document Registry*. Si existen entradas en el registro que cumplan los criterios, el *Document Registry* retorna una lista que incluye los identificadores y repositorios donde se almacenan los documentos.
- **Retrieve Document Set**: iniciada por un *Document Consumer* permite recuperar documentos de un *Document Repository*.

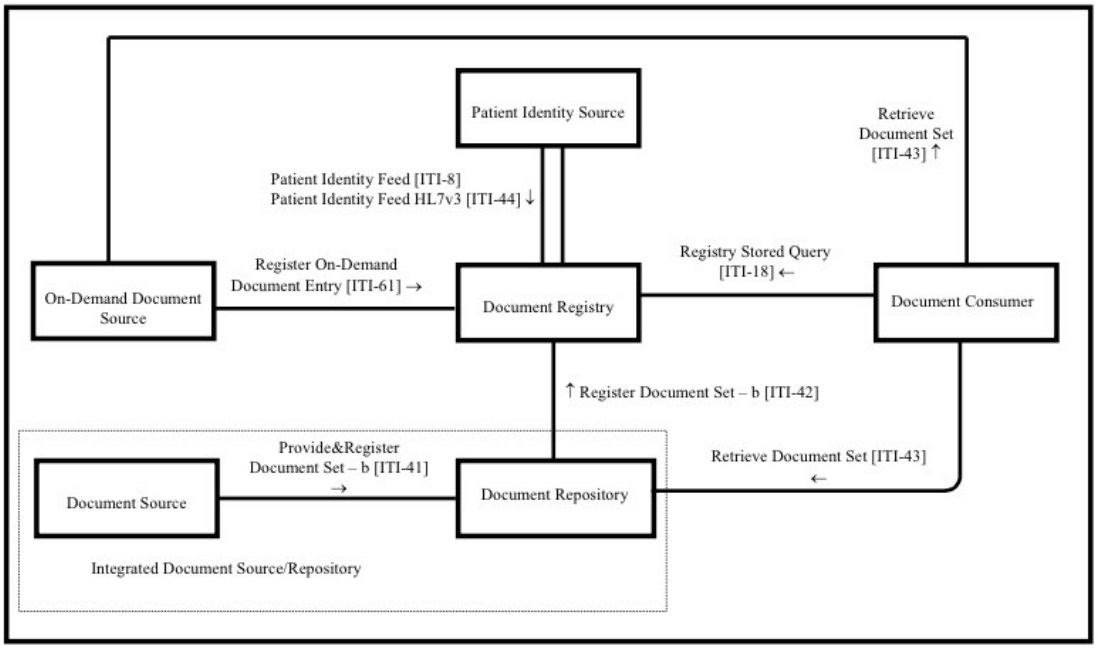


Figura E.8: Actores y transacciones del perftl XDS.b [128].

E.6. Integración de MPI, CDA y XDS

Existen diversos esquemas de integración, por lo que la discusión se limita a un escenario de dos instituciones (*InstA* e *InstB*), con dominios de identificación dados (*DomA* y *DomB*), que desean intercambiar documentos clínicos CDA. Previamente las instituciones deben acordar un dominio

de afinidad en el cual definen:

- Los tipos de documentos a intercambiar y su estructura, en este caso, los tipos de documentos CDA.
- La terminología y esquemas de codificación utilizada en los metadatos y documentos.
- Un dominio de identificación maestro que permitirá mantener las referencias cruzadas y asignar un identificador global a los pacientes de ambas instituciones.

Cada institución contará localmente con un repositorio XDS para almacenar sus propios documentos CDA, y compartirán un registro XDS y un MPI.

Cuando *InstA* desea almacenar un documento CDA de un paciente P, primero debe realizar la referencia cruzada de P_a con el identificador global en el dominio maestro. Para esto realiza una consulta al MPI y obtiene el identificador global. Por último, almacena el documento en el repositorio XDS local, y este reenvía los metadatos (que incluyen el identificador global) al registro XDS centralizado.

Cuando *InstB* desee acceder al documento del paciente P generado por la *InstA*, también deberá realizar la referencia cruzada, esta vez, será del identificador P_b con el identificador global en el dominio maestro. Luego, consulta al registro XDS por documentos del paciente *P* enviando como metadato el identificador global y finalmente accede al repositorio XDS de la *InstA*.

Institución	Tipo de Doc.	Doc.	Nombre	Apellido	Sexo	Fecha de Nac.	Domicilio
Inst. A	CI	44922440	Juan	Fernandéz	M	18-12-1991	Av. Italia 1723
Inst. A	CI	44922440	Juan	Fernandéz	M	18/12/91	Av. Italia 1723
Inst. B	CI	44922441	Juan	Fernandéz	M		Av. Italia 1723
Inst. B	CI	31225660	Cecilia	Perez	F	24-10-1986	
Inst. C	CI	31225660	Cecilia	Peres	F	24-10-1986	Gallinal 1243
Inst. C	SSN	008-36-1234	John	Smith	M	11-09-1976	
Inst. D	SSN	008-36-1234	John	Smythe	M	11-09-1976	
Inst. E	Nro. Carné de Conducir	122492ABC	John	Smith	M	11-09-1976	
Inst. E	CI	22776347	Martín	Ruíz	M	17-13-2000	
Inst. F	CI	22776347	Martín	Ruíz	M	17-12-2000	
Inst. G			Martín	Ruíz	M	17-12-2000	
Inst. G	CI	76539912	Laura	Moreno	F	01-12-1979	Av. Rivera 1140
Inst. H	CI	76539912	Laura	Moreno	F	01-12-1979	Soca 3202

Tabla E.1: Resultado de aplicar un algoritmo hipotético de referencias cruzadas. Puede observarse cómo los registros con alta similitud son agrupados, incluso ante la presencia de errores y falta de información.

Apéndice F

CDA Administración Unidosis

A continuación se presenta el documento XML que define el CDA de nivel 3 *Informe Administración de Unidosis*.

```
<ClinicalDocument
  xmlns="urn:hl7-org:v3"
  xmlns:mif="urn:hl7-org:v3/mif"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:voc="urn:hl7-org:v3/voc"
  xsi:schemaLocation="urn:hl7-org:v3 CDA.xsd" >

  <!-- CDA HEADER -->
  <realmCode code="UY"/>
  <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040" />
  <id root= "2.16.858.2.[idOrganizacion].67430.AAAAMMDDHHMMSS"/>
  <code codeSystem="2.16.840.1.113883.6.1" code="34746-8"
    codeSystemName="LOINC" displayName="Registro de enfermeria"/>
  <title>Informe administracion de unidosis</title>
  <effectiveTime value="AAAAMMDDHHMMSS"/> <!-- ISO 8601-->
  <confidentialityCode code="N" codeSystem="2.16.840.1.113883.5.25"/>
  <languageCode code="es-UY"/>
  <recordTarget>
    <patientRole>
      <id root="2.16.858.1.858.68909.[CI]"/>
      <patient>
        <name>
          <given>Nombre Paciente</given>
          <family>Apellido Paciente</family>
        </name>
        <administrativeGenderCode code="1|2"
          displayName="Masculino|Femenino"
          codeSystem="2.16.858.2.10000675.69600" />
        <birthTime value="AAAAMMDD"/>
      </patient>
    </patientRole>
```

```
</recordTarget>
<author>
  <time value="AAAAMMDDHHMMSS"/>
  <assignedAuthor>
    <id  root="2.16.858.1.858.68909.[CI]"/>
      <!-- Tambien puede utilizarse el identificador asignado
      en el registro de profesionales provisto por el SNIS -->
    <assignedPerson>
      <name>
        <given>Nombre  Enfermero</given>
        <family>Apellido  Enfermeo</family>
      </name>
    </assignedPerson>
  </assignedAuthor>
</author>
<custodian>
  <assignedCustodian>
    <representedCustodianOrganization>
      <id   root="2.16.858.0.[SufijoOIDHospital]"/>
      <name>Nombre  Hospital</name>
    </representedCustodianOrganization>
  </assignedCustodian>
</custodian>

<!-- CDA BODY -->
<component>
  <structuredBody>
    <component>
     <!-- Medications Section -->
      <section>
        <code code="10160-0" codeSystem="2.16.840.1.113883.6.1" />
        <text>Listado de las sustancias administradas al     paciente</text>
        <!-- Substances Administration Acts -->
        <entry>
          <substanceAdministration classCode="SBADM" moodCode="EVN">
            <effectiveTime  value="AAAAMMDDHHMMSS"/>
            <doseQuantity unit="X"/>
              <!-- Medication Information -->
            <consumable>
              <manufacturedProduct>
                <manufacturedMaterial>
                  <code code="IdMedicina"
                    codeSystem="2.16.858.0.0.1.10.2.3.1.11.2"
                    displayName="NombreMedicina"/>
                  <name>Nombre Medicina</name>
                </manufacturedMaterial>
              </manufacturedProduct>
            </consumable>
          </substanceAdministration>
```

```
            </entry>
          </section>
        </component>
      </structuredBody>
    </component>

</ClinicalDocument>
```

Apéndice G

Definición de tipos de evento complejos

A continuación se muestra el archivo .epl con la definición de tipos de evento complejo, utilizando la sintaxis EPL definida por Esper.

```
/*
 * Identificacion de una entidad en una zona del hospital.
 * El listener UpdatePosition actualiza la zona donde se
 * vio por ultima vez la entidad.
 */

@Name('EntityPositionEvent')
select EntityID, ZoneID
from
  EntityIdentificationEvent;

/*
 * Identificacion de una persona en una zona no autorizada.
 * El listener SuscriptionManager notifica al personal de
 * seguridad.
 */

@Name('PersonInUnauthorizedZoneEvent')
select
  PersonID , ZoneID
from PersonIdentificationEvent as e
where
  (not isInAuthorizedZone(PersonID, ZoneID));

/*
 * Identificacion de un objeto en una zona no autorizada.
 * El listener SuscriptionManager notifica al personal de
 * seguridad.
```

```
 */

@Name('ObjectInUnauthorizedZoneEvent')
select
  ObjectID, ZoneID
from ObjectIdentificationEvent as e
where
  not  isInAuthorizedZone(ObjectID,  ZoneID);


/*
 * Administracion de una unidosis a un  paciente.
 * Ocurre cuando se identifica a un paciente seguido de una
 * unidosis (5 minutos despues como  maximo).
 */

@Name('UnidoseAdministrationEvent')
insert into UnidoseAdministrationEvent
select
  e1.UserID, e1.PatientID, e2.UnidoseID
from pattern [
  (every  e1=PatientIdentificationEvent)  ->
     ( not PatientIdentificationEvent(UserID = e1.UserID) and
      every  e2=UnidoseIdentificationEvent(UserID  = e1.UserID)
     )
     where  timer:within  (5 minutes)
];

/*
 * Administracion  incorrecta  de  unidosis  a  un paciente.
 * El listener SuscriptionManager notifica al personal  de
 * farmacia.
 */

@Name('WrongPatientUnidoseEvent')
select
  PatientID, UnidoseID
from UnidoseAdministrationEvent as e
where
  Unidose(e).getRecipientId() != PatientID;

/*
 * Unidosis no  administrada.
 * Ocurre cuando la administracion de una unidosis  tiene
 * un retraso de mas de media  hora.
 * El listener SuscriptionManager notifica al personal  de
 * enfermeria.
 */
```

```
@Name('UnidoseNotAdministeredEvent')
select
  e1.PatientID, e1.UnidoseID
from pattern [
  (every (e1=UnidosePackagingEvent)) ->
     (timer:interval((e1.AdministrationDateTime -
        current_timestamp) / 1000.0 + 30*60) and
    (not e2=UnidoseAdministrationEvent(UnidoseID = e1.UnidoseID,
       PatientID  =  e1.PatientID)
          )
     )
];

/*
 * Unidosis administrada fuera de  tiempo.
 * Ocurre cuando unidosis se administra media hora antes
 * o despues de lo previsto.
 * El listener SuscriptionManager notifica al personal  de
 * farmacia.
 */

@Name('WrongUnidoseTimeEvent')
select
  PatientID, UnidoseID
from UnidoseAdministrationEvent as e
where
  Math.abs(current_timestamp -
      Unidose(e).getDateTimeToBeAdministered().
         toMillisec)  >  30*60*1000;
```