



CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS
DEL INSTITUTO POLITÉCNICO NACIONAL
UNIDAD ZACATENCO
DEPARTAMENTO DE COMPUTACIÓN

Ambiente computacional para compañeros
digitales orientado al cuidado de la salud
empleando infraestructura pervasiva

TESIS

Que presenta
Guillermo Monroy Rodríguez

Para obtener el grado de
Doctor en Ciencias en Computación

Directora de tesis
Dra. Sonia Guadalupe Mendoza Chapa

Ciudad de México

Agosto, 2024

Resumen

Según la Organización Mundial de la Salud, en México, cerca del 16% de la población padece diabetes. Por otro lado, el mundo se encuentra en la cuarta revolución industrial, lo que significa que existen dispositivos con poder de procesamiento computacional casi por todos lados. Estos dispositivos, llamados dispositivos de Internet de las Cosas, son capaces de sentir las magnitudes físicas del entorno y generar una cantidad enorme de datos e información. Adicionalmente, el desarrollo de la inteligencia artificial ha logrado revolucionar el procesamiento de información, incluso información masiva. También, los avances alcanzados en el procesamiento de lenguaje natural son significativos, logrando contar con conversores de audio a texto o reconocedores de habla que han permitido a los ingenieros de software proponer asistentes virtuales capaces de entender a un hablante, e.g., *Alexa*, *Google Assistant*, entre otros. Nos encontramos en un momento ideal para usar los avances en la técnica y las ciencias computacionales para resolver problemas que impacten en el bienestar de las personas, incluyendo la calidad de vida de pacientes con enfermedades crónicas como la diabetes. A pesar de encontrarnos en un momento ideal para desarrollar asistentes virtuales que apoyen al paciente, esto no ha sucedido. Las propuestas del estado del arte no aportan una integración de los avances mencionados y se encuentran deficiencias en su planteamiento para ser implementados o actualizados adecuadamente. En el presente trabajo se denomina compañero digital a un asistente virtual que acompaña al usuario y no le hace daño. Este trabajo propone la creación de un compañero digital para ayudar a los pacientes con enfermedades crónicas a mejorar su calidad de vida. Para lograr esto, se plantea la creación de un arquitectura de *software* que, por medio de la integración de dispositivos de Internet de las Cosas, crea un ambiente computacional pervasivo e inteligente. Para tomar un enfoque adecuado, se han considerado los deseos de los pacientes que padecen una enfermedad crónica, resultando en un marco conceptual para diseñar la interacción entre el sistema propuesto y los pacientes. El ambiente computacional propuesto está centrado en el usuario, lo que permite al sistema, además de sentir el ambiente, tener una comunicación e interacción constante e incremental con el paciente para cumplir con las tareas que más sobrecarga emocional le producen. En el marco del ambiente computacional, también se propone un sistema para guiar a los pacientes en el cumplimiento de sus objetivos en un ambiente basado en áreas jerárquicamente organizadas, denominado AMS (*Area Management System*). Se emplea como caso de estudio un hospital que es visitado por un paciente para recibir tratamiento médico.

Palabras clave: *compañero digital, asistente virtual, arquitectura de software, cuidado de la salud pervasivo.*

Abstract

On one hand, according to the World Health Organization, about 16% of the Mexican population has diabetes. On the other hand, the world is in the so-called fourth industrial revolution or Industry 4.0., which means that there are devices with computational processing power almost everywhere. These devices, called IoT devices, can sense the physical magnitudes of their sense the physical quantities of the environment and generate an enormous amount of data and information. Additionally, the development of artificial intelligence has revolutionized the processing of information, even massive information. Also, the advances achieved in the processing of natural language are significant, achieving audio-to-text converters or speech recognizers that have made it possible for software engineers to propose virtual assistants that can understand a speaker, i.e., Alexa and Google Assistant, among others. We are at an ideal time to use advances in technology and computer science to solve problems that impact people's well-being, including the quality of life of patients with chronic diseases such as diabetes. Even though we are at an ideal time to develop virtual assistants to support the patient, this has not happened. The works reviewed in the state of the art need to integrate the advances mentioned above, and there are deficiencies in their approach to be implemented or updated to the present. We call a virtual assistant that accompanies the user a digital companion. This work proposes creating a digital companion to help patients with chronic diseases improve their quality of life. To achieve this, the creation of a software architecture is proposed that, through the integration of Internet of Things devices, creates a pervasive and intelligent computing environment. To take an appropriate approach, the wishes of patients suffering from a chronic disease have been considered, resulting in a conceptual framework to design the interaction between the proposed system and the patients. The proposed computational environment is user-centered, which allows the system, in addition to sensing the environment, to have constant and incremental communication and interaction with the patient to fulfill the tasks that cause the most emotional overload. Within the framework of the computational environment, a system is also proposed to guide patients in meeting their objectives in an environment based on hierarchically organized areas called AMS (Area Management System). A hospital that a patient visits to receive medical treatment is used as a case study.

Keywords: *Digital Companion, Virtual Assistant, Software Architecture, Pervasive Healthcare.*

Dedicatoria

Al amor de mi vida, Lizbeth, por compartir la vida conmigo. Estoy en deuda con el cosmos. Lizbeth, refrendo la promesa de acompañarte tomados de la mano, sorteando el mar de la vida, nadando hacia lo desconocido, con la fuerza y la libertad del alquimista, como el loco que pende de un pie, balanceándose, pero nunca cayendo, descubriendo como el ermitaño los secretos de la existencia ...

A mi gran amigo, José Luis Gudiño Ángeles. Contigo comparto inquietudes, anhelos, penas y éxitos. Estoy agradecido contigo por brindarme tu amistad. Amigo mío, me has regalado grandes tesoros de conciencia.

La vida es un viaje hacia la conciencia, cada acto, cada decisión, cada momento, tejen al cosmos y lo cambian para siempre.

“Cuándo se ha visto que un dragón sucumba ante la mordida de una serpiente.”
Friedrich Nietzsche

Agradecimientos

A mi directora de tesis, la doctora Sonia Guadalupe Mendoza Chapa, por el gran apoyo brindado con enorme generosidad. Cuando todo parecía oscuro y perdido, usted me ayudó a sortear los obstáculos que se presentaron en mi proceso de formación.

A mis sinodales: Dra. Beatriz Adriana González, Dr. Luis Martín Sánchez, Dr. Juan Carlos Ku y Dr. Sergio Víctor Chapa. Gracias por el tiempo y dedicación a la revisión de este trabajo de tesis y por los atinados comentarios.

Al Consejo Nacional de Humanidades Ciencias y Tecnologías por haberme otorgado la beca para realizar mis estudios de doctorado.

Al Centro de investigación y de Estudios Avanzados del Instituto Politécnico Nacional, por los recursos materiales y humanos provistos que me hicieron posible obtener el grado de doctor en ciencias en computación. Una institución que se forma con el esfuerzo cotidiano de seres humanos ocupados en generar conocimiento para el bien del país.

Al indómito y revolucionario pueblo de México, por las luchas sociales, por la educación pública, por la defensa del conocimiento. A ustedes les debo mi formación y les prometo luchar con ética y justicia a favor de las causas que defendieron.

A mi madre, Blanca, quién me enseñó, con su ejemplo, la belleza de la vida, lo hermoso de ser sensible, la fortaleza que existe cuando se siguen principios y, sobretodo, que el amor se ejerce cuando se da.

A mi padre, Guillermo, quién paradójicamente, me enseñó, con su muerte, cuán presente siempre ha estado en mi vida. Gracias papá porque, a tu manera, me manifestaste tu amor a cada paso de mi existencia.

A mi hermana Yadira, con quien compartí la infancia y adolescencia. Eres perseverante, valiente, fuerte e inteligente. Gracias por cuidarme, por protegerme y por estar conmigo en la primera parte de mi vida.

A mi cuñado Nicolás, quién con actos ha demostrado su amor por mi hermana y apoyo a mi familia, su familia.

A mi familia, le agradezco el apoyo, amor y respeto a mi atípica forma de ser. Soy la hoja nutrida por las robustas raíces viejas y ancestrales. ¡Gracias por la libertad para experimentar, fracasar y encontrar mi camino con corazón!

Índice general

Resumen	III
Abstract	V
1 Introducción	1
1.1 Motivación	3
1.2 Planteamiento del problema	5
1.3 Hipótesis	6
1.4 Propuesta de solución	7
1.5 Organización del documento	7
2 Estado del arte	9
2.1 Arquitecturas y revisiones	9
2.2 Compañeros digitales orientados a la salud	11
2.2.1 EVA	11
2.2.2 VPA	11
2.2.3 SimSensei Kiosk	12
2.2.4 Kognit	12
2.2.5 Conversational Virtual Agent	13
2.2.6 MHAS	13
2.2.7 EmmIR	14
2.2.8 EMMA	14
2.3 Análisis comparativo	14
3 Marco conceptual de diseño de interacción para sistemas de salud con Internet de las Cosas	17
3.1 Antecedentes	18
3.2 Internet de las Cosas, salud y diseño de interacción	19
3.3 Metodología de Investigación	21
3.4 Desarrollo	22
3.4.1 Comprender	22
3.4.2 Divergir y Decidir	22
3.4.3 Prototipar	27

3.4.4	Validar	29
4	Arquitectura para compañeros digitales	33
4.1	Requerimientos funcionales y no funcionales	34
4.2	Actores del sistema	36
4.3	Casos de uso	38
4.3.1	Gestión de citas médicas	40
4.3.2	Seguimiento médico	40
4.3.3	Adherencia a la toma de medicamentos	41
4.4	Vista por capas	42
4.4.1	Capa de aplicación	43
4.4.2	Capa Lógica	44
4.4.3	Capa de Servicios	44
4.4.4	Capa de comunicación	44
4.5	Vista de servicios	45
4.5.1	Reconocedor del Habla	49
4.5.2	Locutor Proactivo	52
4.5.3	Observador Biométrico	53
4.5.4	Compañero Digital	55
4.5.5	Buscador Semántico	56
4.5.6	Analizador de Comportamiento	59
4.5.7	Manejador de Conocimiento	60
4.5.8	Mensajero	62
4.5.9	Adaptabilidad y recomendaciones	64
4.6	Vista lógica	66
4.6.1	Interacciones para la gestión de citas médicas	66
4.6.2	Interacciones para el seguimiento médico	68
4.6.3	Interacciones para mejorar la adherencia a la toma de medicamentos	69
5	Implementación del Reconocedor del Habla Multifuente	71
5.1	Análisis de la experiencia anticipada del usuario	72
5.1.1	Pruebas	72
5.1.2	Resultados	73
5.1.3	Implicaciones para la implementación	75
5.2	Módulos de Reconocedor del Habla Multifuente	77
5.3	Identificación de usuarios	79
5.4	Procesamiento de lenguaje natural	80
5.5	Comunicación entre módulos	81

5.6	Desarrollo del microservicio	82
6	Apoyo para el paciente en el tránsito de un hospital	85
6.1	Jerarquía de Áreas	86
6.2	Antecedentes	87
6.3	Administración distribuida de áreas	89
6.3.1	Arquitectura de AMS	90
6.3.2	Organización estructurada por áreas	91
6.4	Caso de estudio: visita de un paciente al hospital para consulta y realización de estudios	94
6.4.1	Llegada del paciente al área de Recepción del hospital	94
6.4.2	De la Recepción a Medicina Interna	96
6.4.3	De Medicina Interna a Radiología	98
6.4.4	Salida del paciente del hospital	99
6.5	Pruebas con usuarios finales	100
7	Discusión	103
7.1	Diseño centrado en el usuario	103
7.1.1	Un asistente virtual como “compañero”	104
7.1.2	Eficiencia y efectividad	104
7.1.3	Manejo de la multimodalidad	105
7.1.4	Carga cognitiva en usuarios	106
7.1.5	Descubrimiento de uso con interfaces de voz	107
7.2	El compañero digital como sistema ubicuo	108
7.3	El paciente como usuario nómada	110
8	Conclusiones y trabajo futuro	113
8.1	Conclusiones	114
8.2	Aportaciones	116
8.3	Limitaciones	118
8.4	Trabajo futuro	120
	Bibliografía	125

Índice de figuras

1.1	Diagrama de organización del documento	8
3.1	Los tres perfiles desarrollados mediante entrevistas y observación: una persona con enfermedades crónicas (3.1a), un profesional de la salud (3.1b) y un individuo con un estilo de vida activo (3.1c).	23
4.1	Diagrama de casos de uso para el sistema CoDiSPa	39
4.2	Vista por capas de CoDiSPa	43
4.3	Patrón de arquitectura: microservicios	46
4.4	Arquitectura del sistema CoDiSPa: vista de servicios	48
4.5	Microservicio <i>Reconocedor del Habla</i>	50
4.6	Ejemplo de petición capturada por el microservicio <i>Reconocedor del Habla</i>	51
4.7	Ejemplo de respuesta procesada por el microservicio <i>Reconocedor del Habla</i>	52
4.8	Microservicio <i>Locutor Proactivo</i>	53
4.9	Microservicio <i>Observador Biométrico</i>	55
4.10	Microservicio <i>Compañero Digital</i>	57
4.11	Microservicio <i>Buscador Semántico</i>	58
4.12	Interacción del microservicio <i>Buscador Semántico</i> con otros microservicios de CoDiSPa	59
4.13	Microservicio <i>Analizador de Comportamiento</i>	61
4.14	Microservicio <i>Manejador de Conocimiento</i>	62
4.15	Patrón de arquitectura <i>Broker</i>	63
4.16	Microservicio <i>Mensajero</i>	64
4.17	Ejemplo que emplea la taxonomía del conocimiento empleada para el CoDiSPa	65
4.18	Diagrama de secuencia para la gestión de citas médicas	67
4.19	Diagrama de secuencia para el seguimiento médico	68
4.20	Diagrama de secuencia para mejorar la adherencia a la toma de medicamentos	70
5.1	Los resultados muestran que el prototipo fue bien recibido en términos de su utilidad práctica y su atractivo general	75
5.2	Módulos del Reconocedor del Habla Multifuente, comunicación con fuentes de audio y con componentes del sistema CoDiSPa	78
5.3	Diagrama de clases del Reconocedor del Habla Multifuente	78
6.1	Componentes de la arquitectura de AMS	90

6.2	Clases de la arquitectura de AMS	92
6.3	Flujo de actividades asistidas por el sistema AMS para realizar una consulta médica y un estudio en un hospital	96
7.1	Conceptos de diseño centrado en el usuario	107

Índice de tablas

2.1	Comparativa de compañeros digitales para enfermedades crónicas (primera parte)	15
2.2	Comparativa de compañeros digitales para enfermedades crónicas (segunda parte)	15
3.1	Historias de usuario para la persona José	24
3.2	Historias de usuario para la persona Melissa	25
3.3	Historias de usuario para la persona Carlos	26
3.4	Propuestas para posibles dimensiones del marco de trabajo	27
3.5	Marco de diseño de interacción para sistemas de salud con Internet de las Cosas	28
3.6	Conceptos de diseño de interacción de un compañero digital para personas con diabetes	30
4.1	Actores del sistema CoDiSPa	36
4.2	Dispositivos que interactúan con el sistema CoDiSPa	38
5.1	Medias (μ) y desviaciones estándar de los resultados (σ).	75
6.1	Resultados de la carga de trabajo experimentada por usuarios del sistema AMS	101

Introducción

Contenido

1.1	Motivación	3
1.2	Planteamiento del problema	5
1.3	Hipótesis	6
1.4	Propuesta de solución	7
1.5	Organización del documento	7

Gracias al rápido desarrollo de la electrónica en las últimas décadas, actualmente se tienen disponibles dispositivos como: sensores, microcontroladores, sistemas de radiofrecuencias, antenas, entre otros. El desarrollo tecnológico hace posible la creación de sistemas que pueden medir casi cualquier magnitud física del entorno. En consecuencia, el desarrollo tecnológico también se ha acelerado, permitiendo que se cuente con más y mejores dispositivos cada vez. Sin embargo, la mayor oferta de dispositivos y sistemas hace surgir la necesidad de integración entre ellos para lograr diversos objetivos, tales como investigar fenómenos ambientales, explorar el espacio, automatizar procesos, entre muchos otros.

Cada objetivo que se plantea puede implicar varias áreas del conocimiento. Las áreas que más se han visto beneficiadas por el desarrollo de dispositivos son los sistemas distribuidos, cómputo ubicuo y cómputo móvil, áreas que toman sus fundamentos de las ciencias de la computación. En este trabajo de tesis es de particular interés el área de la computación ubicua, que se puede entender como la integración del poder de cómputo en el entorno de la persona, de tal manera que dicho poder de procesamiento no sea percibido como un objeto diferenciado, produciendo así la ilusión de ubicuidad.

El cómputo ubicuo es el marco bajo el cual las tecnologías electrónicas y computacionales (como los sistemas de sensores, los protocolos de comunicación y las arquitecturas de *hardware* y *software*) se integran para desarrollar soluciones que exploren las fronteras de áreas como: electrónica, domótica, Internet de las Cosas (*IoT* por sus siglas en inglés), sistemas de monitoreo para la salud, redes de sensores, sistemas de radar e inteligencia ambiental.

En particular, en el ámbito de los sistemas cognitivos computacionales, se han hecho esfuerzos por construir agentes inteligentes que sean capaces de asistir al ser humano en su calidad de usuario [1]. La evolución de estos agentes, llamados también agentes cognitivos, ha dado lugar a los compañeros digitales, los cuales tienen la capacidad de asistir a los usuarios tomando uno de los roles siguientes:

1. Guardianes
2. Asistentes o Mentores
3. Amigos, Compañeros o Socios

De acuerdo con Barnard [2], los compañeros digitales son el ejemplo ideal de un sistema asistido por computadora que se adapta a sus usuarios humanos y forma una interfaz entre la eficiencia, los procesos estrictamente definidos, los datos masivos y las personas creativas. Barnard también dice que un compañero digital es un *software* que se percibe como un asistente útil. Por su parte, Platz [2] menciona que un compañero digital debe contar con inteligencia (inteligencia artificial) para entender las tareas de que se esté hablando y sólo entonces podrá ayudar al usuario, tomando decisiones y proponiendo soluciones concretas. Los compañeros digitales se encuentran enmarcados en la intersección de varias áreas: cómputo ubicuo, Internet de las Cosas, inteligencia artificial, ingeniería de *software*, interacción humano-computadora y alguna área de conocimiento adicional (dependiendo de la aplicación del compañero digital).

El cómputo ubicuo está cobrando especial importancia como apoyo en el desarrollo de actividades humanas cotidianas en los últimos años. Por ejemplo, en el control de dispositivos para el hogar, en el seguimiento de actividades deportivas, en la selección de películas, series o videojuegos en la industria del entretenimiento, en el seguimiento de condiciones fisiológicas en el cuidado de la salud (*healthcare*), por mencionar algunos.

Por un lado, de acuerdo con el reporte del 2021 de Statista [3], se tienen a nivel mundial 4.88 trillones de teléfonos móviles, de los cuales 3.8 trillones son teléfonos inteligentes. De acuerdo con otro reporte del 2021 de Statista [3], el 48.33% de la población mundial cuenta, por lo menos, con un teléfono inteligente. Adicionalmente se tiene una tendencia de crecimiento, no sólo en la incorporación de más personas al uso de teléfonos inteligentes, sino también en la cantidad de dispositivos y contratos de

conexión que posee cada usuario, llegando a 5.8 trillones de suscriptores a servicios de conexión celular y 10.97 trillones de conexiones celulares. Existe ya un mercado de aplicaciones (*apps*) que están dirigidas a este nicho de mercado, las más notables son las que posibilitan el monitoreo de parámetros biométricos de sistemas *wearables* en forma de pulseras o relojes inteligentes (*smartbands* o *smart whatches*).

Por otro lado, se tienen indicios de que con el uso de *apps* y compañeros digitales, en el ámbito del cuidado de la salud, se puede ayudar a mejorar entre otros aspectos: la adherencia a la toma de los tratamientos medicamentosos, terapias para diversos padecimientos mentales, registro de signos vitales en pacientes con enfermedades crónicas, por mencionar algunos [4, 5, 6, 7].

Cuando se integran dispositivos de Internet de las Cosas, *wearables* o sistemas empotrados a un sistema para que pueda sentir (a través de sensores) su entorno entramos en la definición de un *ambiente computacional*. Un *ambiente computacional* es la relación y la interoperación que se tiene entre infraestructura tecnológica y plataformas de *software* con el propósito de desarrollar, realizar pruebas, desplegar y ejecutar aplicaciones [8]. Entonces, en línea con la necesidad de integrar los avances tecnológicos y científicos, surge nuestro interés en proponer un ambiente computacional para compañeros digitales que ayuden a los pacientes con enfermedades crónicas, como la diabetes, a mejorar su calidad de vida.

1.1 Motivación

El ser humano se encuentra cada vez más inmerso en la llamada cuarta revolución industrial o *industria 4.0*. Esencialmente, ésta consiste en el aprovechamiento de la información para generar conocimiento que ayude a los usuarios a tomar decisiones con un alto grado de certeza o certidumbre. La *industria 4.0* también se nutre de los avances en las disciplinas propias de la inteligencia artificial como: aprendizaje automático (*machine learning*) [9], minería de datos (*data mining*) [10], datos masivos (*big data*) [11], inteligencia de negocio (*bussiness intelligence*) [12], análisis de datos (*data analysis*) [13]; y desarrollos tecnológicos que han hecho posible la existencia de almacenes de datos (*data warehouse*) [14], mercados de datos (*data marts*) [15], bases de conocimiento (*knowledge bases*) [6], sistemas de apoyo a la toma de decisiones (*decision support systems*) [16], sistemas de apoyo ejecutivo (*executive support systems*) [17], por mencionar algunos.

En años recientes, los investigadores han empleado métodos de minería de datos, datos masivos, aprendizaje automático, análisis estadístico, entre otros, para estudiar y entender las problemáticas en el ámbito del cuidado de la salud (*healthcare*) [18]. Sin embargo, los métodos aplicados a los problemas de pacientes reportan resultados

poco eficaces y eficientes debido, entre otras causas, a que los sistemas no están integrados con servicios o agentes computacionales que logren interactuar mediante la comunicación máquina-máquina (*M2M machine to machine communication* por sus siglas en inglés) [6, 14, 15, 19]; tampoco hacen un uso intenso de modelos predictivos por su alto costo computacional para resolver tareas específicas, tales como: gestión de citas médicas o estudios clínicos [20], registro de parámetros biológicos o biométricos para el seguimiento de tratamientos [21] y recordatorios multimodales de citas y medicación [22].

Por otra parte, el área de datos masivos cuenta con técnicas que han sido creadas para trabajar con grandes cantidades de datos, sin embargo estas técnicas son costosas computacionalmente. Por ello, es necesario crear un ambiente computacional que integre y proponga una especificación para incorporar dispositivos e infraestructura, de manera incremental y sostenida, dentro del área del cómputo Ubicuo, empleando diferentes enfoques.

Particularmente, en el ámbito de la salud, se vuelve indispensable el desarrollo de una arquitectura para compañeros digitales, que cumplan con el rol de asistente de un paciente que quiera resolver los problemas inherentes a su condición de salud. Como se ha expuesto, se cuenta con componentes para pensar en una arquitectura de compañeros digitales, se tiene al alcance *wearables*, teléfonos inteligentes, acceso a Internet, dispositivos inteligentes domóticos, pulseras o relojes inteligentes, básculas inteligentes, así como plataformas de software libre como *Home Assistant*¹, *Mosquito MQTT*², *Tasmota firmware*³, *Tuya*⁴, por mencionar sólo algunas, que nos permiten aprenderlas y modificarlas. También se cuenta con plataformas de *hardware* libre tales como: *Raspberry pi*⁵, *Arduino*⁶, *Asus Tinkerboard*⁷, *ESP32*⁸, e incluso existen *FPGA* como *Alhambra II*⁹. Esta oferta de plataformas, tecnologías y dispositivos son los bloques de construcción que pueden aprovecharse bajo una propuesta centrada en el usuario.

La visión que da marco a este trabajo de investigación está en la idea de crear un ambiente computacional centrado en los pacientes con enfermedades crónicas para

¹<https://www.home-assistant.io/>

²<https://mosquitto.org/>

³<https://tasmota.github.io/docs/>

⁴<https://www.tuya.com/>

⁵<https://www.raspberrypi.com/>

⁶<https://www.arduino.cc/>

⁷<https://www.asus.com/mx/networking-iot-servers/aiot-industrial-solutions/all-series/tinker-board/>

⁸<https://www.espressif.com/en/products/socs/esp32>

⁹<https://ultra-lab.net/producto/alhambra-ii-fpga-board-placa/>

resolverles, de manera proactiva, algunos de los problemas asociados con su condición de salud, usando compañeros digitales. Lo anterior se pretende lograr, aprovechando los dispositivos existentes como pulseras inteligentes, balanzas inteligentes, teléfonos inteligentes, sensores domóticos, entre otros. Toda visión es ambiciosa; sin embargo, para este trabajo doctoral, se propone un ambiente computacional (que incluye las arquitecturas de los sistemas que lo componen) para compañeros digitales que resuelvan tareas para ayudar al paciente en sus actividades cotidianas.

1.2 Planteamiento del problema

De acuerdo con la Organización para la Cooperación y el Desarrollo Económicos (OCDE), el estado que guarda la población mexicana en relación al padecimiento de enfermedades crónicas es alarmante. Los datos reportados por la OCDE indican que la diabetes afecta al 15.8% de la población y el 71.88% tiene sobrepeso [23]. En la población mexicana, la diabetes y la obesidad son un problema grave y han sido declaradas como pandemia. Ante este panorama, las áreas de conocimiento que pueden ayudar a controlar, mitigar, paliar o resolver los problemas relacionados a la obesidad y la diabetes requieren trabajar en conjunto para alcanzar objetivos concretos, de manera más eficaz y eficiente. Los pacientes con enfermedades crónicas experimentan una carga de tareas adicionales a sus actividades cotidianas, producto de su padecimiento. Se identifican algunos problemas específicos que se listan a continuación:

- Gestión de citas: es difícil realizar la gestión de citas para análisis clínicos, revisiones médicas, ya sea por parte del paciente o por parte de los familiares o la persona que apoya al mismo.
- Seguimiento médico: normalmente el paciente no registra las acciones que realiza y mucho menos la relación que guarda con algún parámetro biomédico asociado.
- Adherencia a toma de medicamentos: para los pacientes es difícil dar seguimiento a las tomas de sus medicamentos, especialmente cuando son más de tres y con distintos horarios.

Desde la perspectiva del paciente

El paciente con una enfermedad crónica, como la diabetes, es incapaz de llevar, de manera eficiente y eficaz, las acciones derivadas de su padecimiento de salud; además, el paciente requiere de asistencia permanente para reportar avances en su estado de salud.

Desde el ámbito computacional

No existe un ambiente computacional que integre, mediante compañeros digitales, los diversos servicios, aplicaciones, dispositivos e infraestructura disponibles para atender tareas de un paciente con enfermedad crónica.

1.3 Hipótesis

La creación de un ambiente computacional para compañeros digitales, con su especificación correspondiente, hace posible integrar servicios en línea y dispositivos, así como interconectar infraestructura de Cómputo Ubicuo a la nube para poder explotar datos generados en beneficio de los usuarios.

Objetivo general

Crear un ambiente computacional para compañeros digitales, en el campo de la Computación ubicua, que ayude al cuidado de la salud de los pacientes con enfermedades crónicas, delegándole tareas asociadas a su salud.

Objetivos específicos

1. Comprender los trabajos existentes sobre compañeros digitales en el estado del arte para identificar los componentes más importantes
2. Analizar las técnicas aplicadas de los trabajos revisados para proponer mejoras o modificaciones en los componentes
3. Realizar el diseño de un marco conceptual de *diseño de interacción* (IxD por sus siglas en inglés) para identificar conceptos clave que apoyen la creación de aplicaciones para el cuidado de la salud
4. Realizar el diseño de una arquitectura de *software* para compañeros digitales
5. Realizar la implementación del *Reconocedor del Habla Multifuente* para probar el diseño de la arquitectura y los conceptos de *diseño de la interacción* identificados
6. Realizar el diseño de un sistema de apoyo para el paciente en el tránsito de un hospital, con el propósito de extender la asistencia fuera de un ambiente doméstico

Construyendo el ambiente computacional para un compañero digital centrado en el usuario y haciendo uso de la infraestructura pervasiva y los dispositivos asociados a ella, se sientan las bases para poder generar un compañero digital agnóstico¹ de las tecnologías vigentes para que las personas, en condición de enfermedad y tratamiento médico, puedan mejorar su calidad de vida, quitándoles la carga de las tareas adicionales generadas por su condición.

Para lograr una propuesta sólida y cumplir los objetivos planteados, se realizó la búsqueda de artículos de agentes inteligentes, agentes virtuales y compañeros digitales en los motores de búsqueda: *Clarivate*, *ACM*, *IEEE Xplore*, entre otros. Una vez que se realizó el estado del arte para los sistemas que buscan atender a pacientes con enfermedades crónicas, se definieron características para lograr una taxonomía que permitiera contrastar con el ambiente computacional propuesto. También, se realizó una búsqueda de las tecnologías de programación, marcos de trabajo, creación de entornos escalables, patrones de diseño de software y patrones de integración para comprender el estado de la técnica alrededor de los sistemas en el área del cuidado de la salud. Adicionalmente, se revisaron los dispositivos disponibles para Internet de las Cosas, dispositivos *Edge Computing*, microcontroladores, sensores empleados para ambientes pervasivos y computadoras integradas en una sola tarjeta (*SBC*, *Single Board Computing*). Teniendo el estado del arte, estado de la técnica, entornos de desarrollo, marcos de trabajo y dispositivos comprendidos, se planteó una arquitectura para aprovechar estos elementos, dando solución a los requerimientos del problema de investigación y desarrollo tecnológico.

Esta tesis está organizada en ocho capítulos, como muestra en la Figura 1.1. En el **Capítulo 1** se da una breve introducción. En el **Capítulo 2** se presenta el estado del arte. En el **Capítulo 3** se propone un marco de trabajo conceptual de diseño de interacción para sistemas de salud de Internet de las Cosas. En el **Capítulo 4** se expone la arquitectura para un compañero digital en el marco de autocuidado de la salud. En el **Capítulo 5** se describe el microservicio de reconocimiento de habla multifuente. En el **Capítulo 6** se describe el sistema de apoyo para pacientes en tránsito en un hospital.

¹En la jerga computacional, agnóstico se refiere a la capacidad de interoperabilidad y compatibilidad de un componente entre diversos ambientes o sistemas, sin que se necesite adaptarlo para su uso [24].

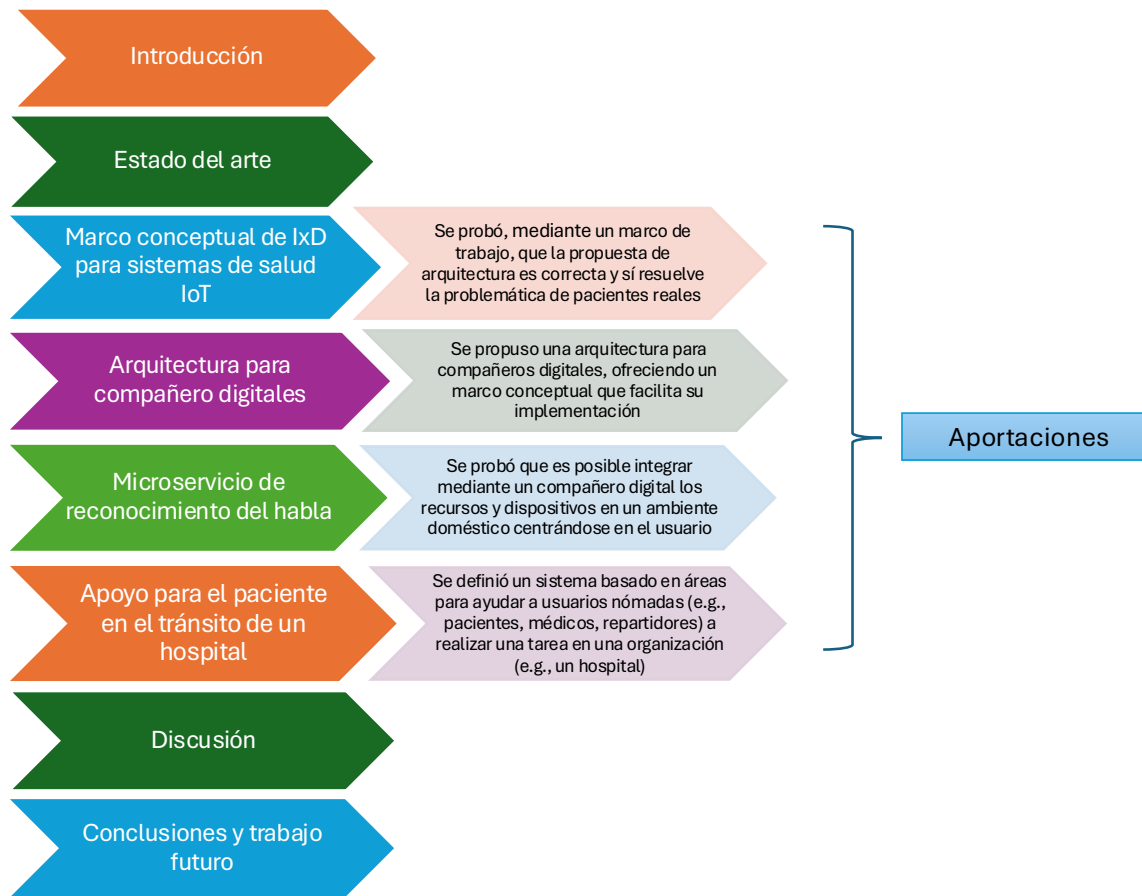


Figura 1.1: Diagrama de organización del documento

En el **Capítulo 7** se presenta la discusión de varios aspectos del trabajo propuesto. Por último, en el **Capítulo 8** se tratan las conclusiones derivadas de esta tesis y el trabajo futuro.

Estado del arte

Contenido

2.1	Arquitecturas y revisiones	9
2.2	Compañeros digitales orientados a la salud	11
2.3	Análisis comparativo	14

2.1 Arquitecturas y revisiones

En su revisión, Ganesan et al. [25] exploran la aplicación de la tecnología de vida asistida ambiental (*Ambient Assisted Living*, AAL, por sus siglas en inglés) para personas mayores con discapacidades físicas y cognitivas, enfatizando su potencial para mejorar la vida diaria y la participación social. El artículo clasifica las tecnologías AAL en: a) herramientas para discapacidades físicas, donde se utilizan tecnologías como sensores de resistencia a la fuerza para monitorear actividades como las posiciones para comer y dormir, y b) sensores portátiles en muñecas y collares para rastrear las actividades diarias. Para los deterioros cognitivos, los sensores portátiles y no portátiles monitorean las actividades de la vida diaria. Los sensores infrarrojos pasivos (*PIR*, por sus siglas en inglés) se utilizan para pacientes depresivos, demostrando cómo las tecnologías AAL pueden ayudar a las poblaciones de edad avanzada, a través de pastilleros digitales y recordatorios *GPS* (*Global Position System*). La revisión también cubre tecnologías domésticas inteligentes que implementan varios sensores para monitorear y ayudar con discapacidades físicas y cognitivas, contribuyendo al bienestar y a la independencia de las personas mayores. Además, el artículo reconoce la necesidad de realizar una

investigación más amplia sobre las tecnologías AAL destinadas a la participación social y a reducir la carga de los cuidadores.

Patro et al. [26] presentan un modelo predictivo de vida asistida ambientalmente para enfermedades cardiovasculares mediante aprendizaje supervisado. La investigación destaca el importante papel del Internet de las Cosas en la atención sanitaria, centrándose en la monitorización de pacientes y el análisis de datos en tiempo real. Para predecir enfermedades cardiovasculares, la arquitectura del sistema propuesta integra técnicas de aprendizaje automático, incluida la regresión lineal múltiple, la regresión *LASSO* (*least absolute shrinkage and selection operator*, por sus siglas en inglés), la regresión de crestas y clasificadores de árboles de decisión. El estudio enfatiza la importancia del preprocesamiento de datos, la extracción de características, el aprendizaje automático y capacidad para desarrollar un modelo predictivo efectivo. El rendimiento del modelo se evalúa en función de la precisión, la exactitud y la sensibilidad, lo que demuestra el potencial del Internet de las Cosas y el aprendizaje automático para mejorar los servicios de atención médica y los diagnósticos predictivos.

Qureshi et al. [27] realizan una revisión detallada de la literatura sobre la vida asistida ambiental, basada en aprendizaje profundo, para el autocontrol de las afecciones cardiovasculares. Explora la integración de tecnologías portátiles, vida asistida por ambiente, sistemas de gestión clínica y métodos de aprendizaje profundo para el diagnóstico cardiovascular. El estudio enfatiza el papel de estas tecnologías en la detección temprana de enfermedades, la mejora del autocuidado del paciente y la intervención médica mejorada. También analiza los beneficios de aprovechar el aprendizaje profundo para analizar datos biomédicos complejos, destacando su importancia para mejorar los servicios de atención médica y el diagnóstico predictivo de enfermedades cardiovasculares.

Si bien los estudios de Ganesan et al. [25], Patro et al. [26], y Qureshi et al. [27] brindan información valiosa sobre la vida asistida ambientalmente y las técnicas de inteligencia artificial (IA) para las enfermedades cardiovasculares y el cuidado de las personas mayores, nuestro trabajo se centra claramente en el cuidado de la diabetes en todos los grupos de edad. La propuesta que se presenta, aprovecha las herramientas de IA existentes para crear compañeros digitales personalizados, integrando sistemas de uso común para soluciones personalizadas. A diferencia de las aplicaciones específicas para el cuidado de personas mayores o enfermedades cardiovasculares, nuestro enfoque prioriza soluciones adaptables y centradas en el usuario, evitando complejidades innecesarias y alineándonos más estrechamente con las diversas necesidades de las personas diabéticas. Esta distinción subraya la originalidad y relevancia de nuestra propuesta en el panorama cambiante de las soluciones de salud digital.

2.2

Compañeros digitales orientados a la salud

En la literatura científica revisada, se encontraron trabajos sobre compañeros digitales con el rol de asistentes en entornos para el cuidado de la salud. A continuación, se presenta un análisis de las características observadas de cada uno de los trabajos.

2.2.1

EVA

El sistema *Educational Virtual Assistant* (EVA) [28] es un compañero digital que tiene como objetivo educar a los pacientes, a partir de preguntas y respuestas, en el manejo correcto de su enfermedad crónica. EVA es un sistema conversacional basado en la comprensión de lenguaje natural, a partir de algoritmos de aprendizaje automático. El sistema hace uso de un modelo entrenado, empleando datos de comportamiento que incluye temas sobre hipoglucemia, nutrición, actividad física y medicación. El paciente puede hacer preguntas al compañero digital, mediante una caja de diálogo contenida en una *app*. El compañero digital intenta identificar el tema de la pregunta para dar una respuesta concreta o hacer una recomendación arrojada por el modelo entrenado, o bien interroga al paciente hasta que el modelo sea capaz de dar una respuesta. Una de las desventajas que tiene EVA es que no es proactiva, i.e., reacciona ante las entradas del paciente, pero no es capaz de predecir o estar pendiente del paciente y proporcionarle las sugerencias pertinentes para mejorar sus hábitos o resolver sus posibles dudas. El sistema no emplea dispositivos adicionales, tales como *wearables*, de contar con ellos, el sistema podría ser orientado al usuario y dejar de ser puramente reactivo, es decir, reaccionar a las solicitudes del usuario únicamente sin proponer acciones derivadas de la información que los *wearables* le proporcionan al sistema.

2.2.2

VPA

De acuerdo con Ahmed [29], el compañero digital que propone es de tipo conversacional, usando una serie de preguntas para diagnosticar el grado de demencia que un paciente de la tercera edad tiene. Ahmed reporta que se hace una prueba cognitiva llamada HDS-R (*Revised Hasegawa's Dementia Scale* por sus siglas en inglés). La prueba se realiza de manera verbal, pues tanto las preguntas como las respuestas son en modo auditivo. Este compañero digital tiene como objetivo ayudar a identificar el grado de

deterioro en las capacidades mentales de los pacientes, con el fin de que el equipo médico les ajuste o planifique un tratamiento adecuado. Si bien el compañero digital proporciona una ágil interacción con el paciente, no es un sistema proactivo o que sea capaz de predecir el deterioro o el comportamiento del paciente. El sistema no es capaz de inferir el estado de salud mental del paciente sin solicitarle información. El sistema no emplea dispositivos adicionales tales como *wearables*. El contar con *wearables* permitirían al sistema pasar de un enfoque reactivo a las peticiones del usuario a estar centrado en el usuario y con ello ser proactivo, i.e., propositivo en función de la información de contexto provista por el dispositivo *wearable*.

2.2.3

SimSensei Kiosk

DeVault et al. [30], reportan en su proyecto denominado SimSensei Kiosk el compañero digital *Ellie*. *Ellie* realiza entrevistas virtuales con humanos para evaluar automáticamente los indicadores de angustia, que son conductas asociadas a depresión, trastorno de estrés postraumático y ansiedad. *Ellie* utiliza cuatro clasificadores para la comprensión del lenguaje natural, la detección de contexto y la generación de respuestas. Si bien el sistema emplea técnicas de clasificación, no es capaz de generar, de manera desatendida por el paciente, un diagnóstico sobre el principal parámetro que mide, la angustia. Ofrece una interacción visual donde analiza las expresiones del rostro de los pacientes y se comunica verbalmente con él. El sistema no emplea dispositivos adicionales tales como *wearables*. Como en los trabajos anteriores, mediante el uso de *wearables* el sistema se transformaría de ser puramente reactivo a estar centrado en el usuario.

2.2.4

Kognit

Este proyecto es una propuesta de compañero digital que hace uso de lentes inteligentes y algunos *wearables* para ayudar a los pacientes, con deterioro mental, a reforzar o recuperar la memoria en sus actividades cotidianas. El prototipo de compañero involucra aprendizaje automático, procesamiento de imágenes y está centrado en el paciente de manera proactiva. Una de las actividades que el compañero atiende es el seguimiento de la toma de los medicamentos del paciente. El sistema es prometedor, sin embargo no

se encontraron reportes de su implementación para saber qué tanto de lo propuesto se logró llevar a cabo [31].

2.2.5

Conversational Virtual Agent

Como su nombre lo indica, es un compañero digital conversacional que es accesible a través de una *app* para teléfono inteligente. El foco del sistema es los pacientes del padecimiento fibrilación cardiaca intratable con un alto riesgo de presentar un ataque. El compañero integra sensores de respuesta galvánica (*GSR, Galvanic Skin Response*) para detectar el ritmo cardiaco, aunque su interacción no es constante, sino bajo eventos. El compañero digital fue probado en 120 pacientes de manera aleatoria y es capaz de dar aviso ante un riesgo de ataque cardiaco a la clínica definida para dar auxilio al paciente. El sistema no es proactivo ni predictivo, ya que los autores reportan que el grado de incertidumbre para hacer una predicción puede traer fatales consecuencias, pero esto se puede deber a una mala elección de la técnica de aprendizaje automático [32].

2.2.6

MHAS

Este compañero digital emplea teléfonos inteligentes y está diseñado para evaluar el grado de angustia de un grupo de rescatistas sometidos a un grado de estrés alto, producto de su actividad. El sistema muestra cómo la identificación del usuario con un compañero digital, que realiza tareas de asesor, tiene un impacto significativo en los aspectos cognitivos y emocionales del usuario. Los servicios de asesoramiento virtual son herramientas muy importantes para el dominio de la autoayuda psicológica. El sistema de asesoramiento virtual MHAS se creó con la colaboración de 120 rescatistas de emergencia que realizaron sus tareas en Mongolia y está basado en medios audiovisuales y cuestionarios especializados. El sistema no es proactivo. El sistema no cuenta con dispositivos extras, como *wearables*, para incorporar más fuentes de datos [33].

Se trata de un robot social empático y un compañero digital que brinda apoyo cognitivo a las personas mayores para las actividades de la vida diaria. Muestra siete emociones para generar una respuesta empática a los usuarios: enojado, asustado, disgustado, feliz, neutral, triste y sorprendido. El sistema tiene una interacción auditiva con el paciente y es capaz de reconocer expresiones y algunas acciones que el paciente realiza. El robot no es proactivo y presenta el inconveniente de que no puede estar acompañando, en todo momento y lugar de la casa, al paciente [34].

Se trata de un compañero digital que se presenta como un asistente personal virtual y que se ejecuta en un teléfono inteligente. EMMA interactúa con el usuario y emplea evaluaciones psicológicas, a partir de preguntas, para rastrear su nivel de energía, positividad y bienestar general para decidir si es necesaria una intervención, debido al riesgo de depresión del paciente. Emma está dirigido a cualquier tipo de paciente, pero en particular pacientes que tienen propensión a deprimirse. El sistema fue probado en una comunidad de 39 pacientes, algunos de ellos internos. El sistema no es proactivo además de que no cuenta con interacción con dispositivos extras tales como *wearables* [35]. El uso de *wearables* es importante, pues posibilita que los sistemas estén centrados en el usuario y cuenten con una fuente de datos adicional que ayude a reconocer la situación, dada por el contexto, en el que los usuarios se encuentran.

2.3

Análisis comparativo ---

A partir de los trabajos analizados, se identificaron las siguientes características de interés: tipo de interacción, si el sistema tiene capacidad proactiva, cuál es la enfermedad tratada, si tiene un ambiente integrado con otros dispositivos, si usa teléfonos inteligentes, si usa *wearables*, cuáles son las técnicas empleadas, si está centrado en el paciente, si reporta resultados experimentales y si cuenta con capacidades educativas. En las Tablas 2.1 y 2.2, se presentan los trabajos usando las características listadas.

Los compañeros digitales se han desarrollado tratando de resolver el problema de la interacción con el usuario de manera inteligente, i.e., simulando que el sistema

Nombre	Interacción	Proactivo	Enfermedad	Integrado	Smartphone	Wearable
EVA [28]	Texto	No	Diabetes	No	Sí	No
VPA [29]	Verbal	No	Demencia	No	Sí	No
SimSensei Kiosk [30]	Verbal	No	Angustia	No	Sí	No
Kognit [31]	Texto	No	Deterioro geriátrico	Sí	No	Sí
Conv. Virt. Agent [32]	Expresiones emotivas	No	Fibrilación intratable	No	Sí	No
MHAS [33]	Video y texto	No	Angustia	No	Sí	No
EmmIR [34]	Expresiones emotivas	No	Apoyo geriátrico	No	No	No
EMMA [35]	Texto, audiovisual	No	Depresión	No	Sí	No

Tabla 2.1: Comparativa de compañeros digitales para enfermedades crónicas (primera parte)

Nombre	Técnicas	Centrado en el paciente	Resultados experimentales	Educativo
EVA [28]	<i>NLP, ML,</i>	No	No	Sí
VPA [29]	<i>NLP, ML,</i>	No	No	No
SimSensei Kiosk [30]	<i>Classifiers, NLP</i>	No	No	No
Kognit [31]	<i>ML, Vision Recognition, NLP</i>	Sí	No	No
Conv. Virt. Agent [32]	<i>ML, Pattern Recognition</i>	Sí	Sí	No
MHAS [33]	Cuestionarios	No	Sí	No
EmmIR [34]	Robótica, <i>NLP, ML</i>	No	No	No
EMMA [35]	Regresión lineal, <i>Random Forest, Classification</i>	No	Sí	No

Tabla 2.2: Comparativa de compañeros digitales para enfermedades crónicas (segunda parte)

cuenta con la capacidad de adaptarse a las peticiones o expresiones del usuario. Para resolver el problema de entender al usuario y dar la impresión de inteligencia, ha sido necesario acotar el universo de posibilidades para no malinterpretar los comandos del usuario [36]. Los sistemas conversacionales han demostrado que son de gran ayuda para los objetivos del autocuidado de la salud; sin embargo, la mayoría de compañeros digitales usan entradas restringidas a opciones múltiples [37]. En tiempos recientes, los compañeros digitales emplean, cada vez con mayor frecuencia, el procesamiento de lenguaje natural para identificar las ideas o comandos expresados por los usuarios. Con la evolución de las técnicas de aprendizaje automático, se han incorporado modelos entrenados al procesamiento de lenguaje natural, dando como resultado sistemas con una mayor flexibilidad para la comprensión de las frases y una cercanía mayor

con el comportamiento humano [38, 39]. En la medida en que el comportamiento del compañero digital es más cercano o parecido al del ser humano genera confianza en el usuario porque se siente comprendido [40]. Uno de los objetivos de los compañeros digitales es ayudar a los usuarios en la toma de decisiones. Por esta razón, es cada vez más frecuente que se desarrollen compañeros digitales con avatares o personalidades que muestren también emociones o se comuniquen con el usuario de formas no verbales, pero si visuales [41, 42].

De los trabajos analizados, se identifica una área de oportunidad para proponer un ambiente computacional que incorpore dispositivos de Internet de las Cosas y *wearables* para aumentar las capacidades del compañero digital [43]. Otra área de oportunidad, es la incorporación de los principios del cómputo ubicuo referentes a la proactividad y a la disponibilidad en todo momento y en todo lugar. Aunque sea de manera parcial, se puede integrar al compañero con las pantallas inteligentes, con los teléfonos inteligentes, con las pulseras inteligentes, con los equipos de audio, entre otros, de tal manera que se aprovechen los dispositivos disponibles en el mercado electrónico actual [44]. También, es posible incorporar procesamiento ligero a algunos dispositivos para reducir el uso de cómputo en un servidor (*offloading*) o el uso de cómputo en la nube [19]. Una área de oportunidad más, es aprovechar en esta propuesta el avance en el uso de sensores [45] para hacer el seguimiento de los parámetros biológicos de los usuarios, tales como oxímetros, glucómetros, manómetros, sensores de respuesta galvánica, cámaras, entre otros. Estos sensores ponen a disposición del compañero digital la información relacionada a la salud y a las actividades de los usuarios, y con esto, dotar al sistema propuesto con la capacidad de aprender y adaptarse a cada paciente [46].

Marco conceptual de diseño de interacción para sistemas de salud con Internet de las Cosas

Contenido

3.1	Antecedentes	18
3.2	Internet de las Cosas, salud y diseño de interacción	19
3.3	Metodología de Investigación	21
3.4	Desarrollo	22

El aumento de la prevalencia de las enfermedades crónicas ha impulsado el desarrollo de numerosos dispositivos y sistemas de Internet de las Cosas para mejorar los resultados de salud de los pacientes. A pesar de la abundancia de estas soluciones, su usabilidad sigue siendo un desafío significativo que dificulta su adopción y eficacia. Este capítulo propone un marco conceptual de diseño de interacción para sistemas de Internet de las Cosas para la atención médica que prioriza los tres componentes críticos de usabilidad: efectividad, eficiencia y satisfacción. Para desarrollar el marco, se llevó a cabo una revisión del estado del arte y se realizaron entrevistas a varios pacientes y expertos en salud utilizando la metodología *Design Sprint*. Esto resultó en la creación de tres personas y sus historias de usuario relacionadas. El marco propuesto consta de tres componentes principales: interfaces de usuario como servicios, interacciones contextuales y gestión de datos centrada en el usuario. Se utilizó un escenario hipotético de un compañero digital para personas con diabetes para demostrar la viabilidad del marco. Los resultados de la propuesta subrayan la importancia de considerar a todos

los interesados al diseñar sistemas de Internet de las Cosas y destacan el potencial para que las personas se beneficien de una experiencia más integrada y personalizada.

El presente capítulo se presentan los antecedentes en la Sección 3.1. En la Sección 3.2 se revisa la literatura científica relevante sobre Internet de las Cosas, salud y diseño de interacción. En la Sección 3.3 explica la metodología de investigación que se siguió. Finalmente, en la Sección 3.4 describe el marco de diseño de interacción propuesto para sistemas de salud con Internet de las Cosas.

3.1

Antecedentes

Para empezar, es necesario revisar qué características tienen los sistemas con Internet de las Cosas orientados a la salud. Con este fin, el trabajo presentado por Chhiba et al. [47] las establece como resultado de una revisión bibliográfica. Este estudio destaca varios puntos clave en relación a los sistemas con Internet de las Cosas, incluyendo su estructura de red dinámica y en constante cambio, recursos limitados de memoria y energía, diversidad en cuanto a tipos de dispositivos, y la necesidad de medidas de privacidad y seguridad cuando se manejan datos sensibles. Además, describen tres dominios prevalentes en este tipo de sistemas: aquellos orientados a la vida saludable, e.g., ayudando a monitorear rutinas de ejercicio, regímenes alimenticios y prevención de enfermedades. Otro dominio incluye el cuidado en el hogar, e.g., monitores de autocuidado, telemedicina y prevención de accidentes. Finalmente, existe el cuidado especializado, como el proporcionado por hospitales y especialistas en salud.

En lo que respecta a los usuarios de tecnologías de Internet de las Cosas para la salud, Hossain et al. [48] presentan un análisis de las posibles influencias en los potenciales usuarios de estos sistemas. Una influencia tiene que ver con las relaciones interpersonales, i.e., las personas tienden a ser incentivadas a usar una tecnología específica por recomendaciones de conocidos. Otro factor es la confiabilidad, ya que los usuarios esperan que los datos sean confiables, precisos y privados. Otro elemento es la intención del usuario de mejorar su salud, ya sea debido a una enfermedad reciente, recomendación médica o simplemente por iniciativa propia; muchas personas buscan comenzar una vida más saludable y ven la tecnología como un medio para ayudarles en su propósito. Por último, otro factor que se puede mencionar es el valor percibido, ya que elementos como el precio bajo, la demanda de un producto, la calidad y lo que los usuarios obtienen por un precio determinado son elementos que ayudan a decidir la adopción de un sistema.

Para entender la relación entre la usabilidad y los sistemas de salud, se consultó la investigación de Yen y Bakken [49], quienes crearon una revisión y categorización exhaustiva de estudios de evaluación de usabilidad. Entre sus conclusiones, encontraron

que no hay un buen modelo o marco teórico para apoyar este tipo de trabajo y que las evaluaciones, a menudo, están estrechamente enfocadas, i.e., evalúan una función de una herramienta con un usuario en particular y luego intentan extender los resultados a varios escenarios, lo que deja muchos detalles fuera y crea una imagen incompleta. Además, encontraron que la retroalimentación de los interesados, a menudo, sólo se integró en una etapa del desarrollo y nunca se consultó de nuevo.

Después de revisar las características de los sistemas de Internet de las Cosas para la salud, los usuarios y varios problemas de usabilidad, se discutirá un trabajo que presenta un modelo para aceptar las tecnologías de Internet de las Cosas. Aunque esta investigación no es específica para los sistemas de salud, es una buena representación de una propuesta que incorpora todos los elementos que se ha descrito anteriormente. En su estudio, Sneesl et al. [50] desarrollaron un modelo de adopción de tecnología de Internet de las Cosas para un campus inteligente. Al revisar exhaustivamente el estado del arte, propusieron un modelo que comienza con dos elementos principales: propagación y valor percibido. El primer elemento engloba factores como la replicabilidad, escalabilidad, confiabilidad, seguridad y privacidad, así como el costo de implementación. El segundo elemento incluye utilidad, disfrute, tecnicidad y confianza. A través de este modelo, se puede observar que los desafíos en la adopción de la tecnología de Internet de las Cosas en un campus inteligente son similares a los de los sistemas de salud.

Finalmente, vale la pena mencionar un trabajo que permite profundizar en las necesidades y desafíos que enfrentan los entornos de Internet de las Cosas para la salud. De Michele y Furini [51] identifican la seguridad, privacidad y confidencialidad como elementos a considerar no solo por los usuarios, sino por todos aquellos involucrados en estos escenarios, i.e., fabricantes, investigadores, técnicos y profesionales de la salud. Además, exponen el comportamiento no intencional, que se refiere a situaciones en las que los usuarios cambian su comportamiento en respuesta a los datos fisiológicos proporcionados por las tecnologías. Esto puede ser peligroso, ya que solo un médico certificado puede indicar un diagnóstico basado en datos y, en función de esto, hacer cambios en los pacientes. De lo contrario, existe la posibilidad de causar daño involuntario.

3.2 Internet de las Cosas, salud y diseño de interacción

El Internet de las Cosas está revolucionando la atención médica, con dispositivos y sistemas conectados que transforman la forma en que se gestiona la salud y bienestar, proporcionando datos en tiempo real sobre el estado de salud de los pacientes, permitiendo la detección temprana de problemas y ofreciendo una atención más perso-

nalizada [52]. Sin embargo, hay varios desafíos para implementar sistemas con Internet de las Cosas en la atención médica, incluyendo preocupaciones de privacidad y seguridad, falta de estandarización y la necesidad de métodos de diseño y evaluación [7]. En general, estos desafíos destacan la importancia de diseñar holísticamente los sistemas de salud en lugar de centrarse en una área específica, como los sensores o las tecnologías de comunicación. Al tener en cuenta todos los aspectos de un ecosistema, desde el hardware hasta el software, las soluciones con Internet de las Cosas pueden ser mucho más efectivas [53].

El problema al desarrollar sistemas de salud con Internet de las Cosas deriva de la necesidad de diseñar para una amplia gama de usuarios, incluyendo pacientes, cuidadores y clínicos [54]. Esto puede ser difícil porque cada grupo tiene diferentes necesidades y expectativas. Por ejemplo, los pacientes pueden querer respuestas rápidas del sistema, mientras que los cuidadores pueden querer más información sobre la condición del paciente. Los clínicos pueden necesitar tomar decisiones complejas rápidamente para proporcionar una atención óptima a sus pacientes [55]. Además, es necesario considerar el complejo entorno regulatorio en el que operan los sistemas de salud. Los proveedores de atención médica deben cumplir con varias regulaciones federales y estatales con respecto a los derechos de privacidad, la seguridad de los datos y los protocolos de ensayos clínicos [56]. Además, muchos hospitales están obligados por ley a compartir registros electrónicos de salud con otros hospitales dentro de su red, para que los pacientes puedan recibir atención de múltiples fuentes al mismo tiempo [57].

Es esencial considerar la experiencia del usuario (UX) para realizar el potencial de los sistemas de salud con Internet de las Cosas. Hay varios factores a tener en cuenta, como las necesidades y preferencias del usuario; el contexto específico en el que se utilizará el sistema; los tipos de tareas que el usuario necesitará realizar; el nivel de complejidad requerido para cada tarea; y cualquier posible preocupación de seguridad o privacidad que deba abordarse [58, 59]. Para lograr esto, el diseño de interacción puede ayudar a crear sistemas con Internet de las Cosas fáciles de usar y efectivos. Un sistema bien diseñado puede mejorar los resultados del paciente al proporcionar atención personalizada y atractiva. Al tener en cuenta las necesidades de los usuarios, los diseñadores de interacción pueden crear soluciones útiles y utilizables. En última instancia, el diseño de interacción puede desempeñar un papel crucial en hacer que los sistemas de salud con Internet de las Cosas sean más exitosos en general [60, 61].

Este capítulo propone un marco de diseño de interacción para sistemas de salud con Internet de las Cosas basado en tres conceptos clave: interfaces de usuario como servicios, interacciones conscientes del contexto y gestión de datos centrada en el

usuario. La información que se recopiló para construir esta propuesta provino de dos fuentes: 1) una revisión del estado del arte para descubrir qué problemas se han encontrado en las evaluaciones de usabilidad y experiencia del usuario de los sistemas con Internet de las Cosas, así como los avances realizados en el campo de diseño de interacción; 2) con base en una técnica de *Design Thinking* [62], *Design Sprint* [63], se realizaron entrevistas a varios usuarios que han integrado sistemas con Internet de las Cosas para cuidar su salud, para recopilar requisitos funcionales y construir personas basadas en esto.

3.3 Metodología de Investigación

Como método de investigación, se eligió adaptar el enfoque conocido como *Design Sprint*, el cual consta de cinco etapas [63, 62]:

1. **Comprender:** los participantes evalúan el problema que pretenden abordar, las personas para las que diseñan y el formato que utilizarán.
2. **Divergir:** se anima a los participantes a dejar de lado los prejuicios y participar en varios ejercicios para generar tantas ideas como sea posible, independientemente de lo factibles o poco realistas que puedan ser.
3. **Decidir:** a través de diferentes actividades, los participantes deciden qué ideas investigar más a fondo.
4. **Prototipar:** los participantes crean rápidamente un prototipo de sus ideas, centrándose en el flujo de la interfaz de usuario.
5. **Validar:** los participantes presentan su producto a los usuarios, lo prueban y se les anima a proporcionar comentarios siempre que sea posible.

Aunque este enfoque no es una metodología de investigación tradicional, se eligió porque es una forma efectiva de crear productos valiosos que no solo funcionan bien y son estéticamente atractivos, sino que también fomentan cambios en habilidades y formas de pensar [62]. El objetivo de este enfoque en particular es llegar a una solución viable en cinco días. Sin embargo, se modificó el tiempo para llegar a una solución porque estábamos más interesados en un enfoque de diseño centrado en el usuario y sentimos que la restricción de tiempo sería limitante.

3.4

Desarrollo

En esta sección, se detallan las distintas etapas de la metodología que se utilizaron en la creación de la presente propuesta. Es importante señalar que el proceso descrito aquí se refiere a la iteración inicial de la metodología.

3.4.1

Comprender

Para recopilar información valiosa sobre posibles requisitos, se investigó contactando a amigos y familiares. Según sus características, estos fueron divididos en tres grupos: 1) personas con una enfermedad crónica o condición de salud específica; 2) profesionales de la salud; 3) personas que utilizan dispositivos electrónicos durante el ejercicio.

A través de entrevistas no estructuradas y observación casual (principalmente entre las personas cercanas a nosotros), fue posible entender sus rutinas diarias. Posteriormente, se desarrollaron tres perfiles que representan los grupos críticos dentro de la población objetivo (ver Figura 3.1).

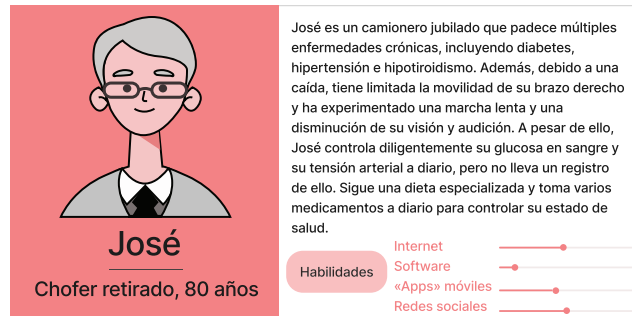
El siguiente paso en esta fase fue llevar a cabo entrevistas más específicas para descubrir las necesidades y características de cada grupo de usuarios, según lo referenciado en los estudios de Keijzer-Broers y Southall [64, 65]. Se realizaron encuestas y entrevistas con los participantes para recopilar información sobre los tipos de dispositivos o sistemas que les gustaría incorporar en su rutina diaria de salud. También se pidió a aquellos que ya estaban utilizando esta tecnología que compartieran su opinión sobre qué características faltaban y cómo podrían mejorarse. En el caso de los expertos en salud, se les preguntó sobre qué tipo de información les gustaría que sus pacientes recopilaran. Luego, se utilizó esta información para crear historias de usuario para cada perfil, como se describe en la metodología de Cohn [66], que se pueden ver en las Tablas 3.1, 3.2 y 3.3.

3.4.2

Divergir y Decidir

Con base en las personas y sus historias de usuario, se pudo crear un escenario integrador que involucra los siguientes elementos:

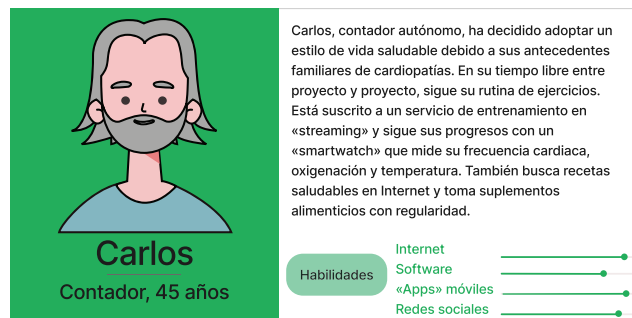
- **Seguimiento de salud personalizado:** cada usuario podría tener un panel de control personalizado que rastrea sus métricas de salud, como los niveles de glucosa y presión arterial de José, la frecuencia cardíaca, la oxigenación y la temperatura de Carlos durante el ejercicio y el progreso de los pacientes de Melissa en el manejo del TDAH.



(a) José - Enfermedades crónicas



(b) Melissa - Profesional de la salud



(c) Carlos - Estilo de vida activo

Figura 3.1: Los tres perfiles desarrollados mediante entrevistas y observación: una persona con enfermedades crónicas (3.1a), un profesional de la salud (3.1b) y un individuo con un estilo de vida activo (3.1c).

- **Integración con dispositivos portátiles:** el panel de control podría integrarse con dispositivos portátiles, como el reloj inteligente de Carlos, que sincronizaría automáticamente los datos de salud en tiempo real, reduciendo el esfuerzo manual requerido para ingresar los datos.

Tabla 3.1: Historias de usuario para la persona José

Requerimientos	Debe tener	Sería bueno tener
Funcional	1. Seguimiento y monitoreo automatizado de los niveles de glucosa en sangre, presión arterial e ingesta de medicamentos. 2. Capacidad para registrar y acceder fácilmente a datos históricos de salud. 3. Opción de establecer recordatorios para medicamentos y restricciones dietéticas.	1. Integración con un dispositivo <i>wearable</i> para rastrear la actividad física y monitorear la frecuencia cardíaca y la oxigenación. 2. Comandos de voz activados para facilitar el uso con movilidad limitada. 3. Ayudas visuales para mejorar la lectura de los datos de salud y las notificaciones.
Interacción del usuario	1. Interfaz simple e intuitiva para rastrear y monitorear datos de salud. 2. Capacidad para compartir información de salud con profesionales médicos. 3. Opción para recibir consejos y recomendaciones personalizados sobre salud basados en datos de salud.	1. Integración con un asistente virtual para acceder fácilmente a información y recordatorios de salud. 2. Opción de personalizar notificaciones y recordatorios para adaptarse a las preferencias personales. 3. Capacidad para establecer metas y realizar un seguimiento del progreso para mantener un estilo de vida saludable.
Contexto social	1. Capacidad para conectarse y comunicarse con una red de apoyo para motivación y aliento. 2. Opción para compartir datos de salud y progreso con amigos y familiares. 3. Integración con redes sociales para rastrear y compartir recetas y planes de comidas saludables.	1. Opción de unirse a comunidades virtuales y participar en desafíos con otras personas que trabajan hacia objetivos de salud similares. 2. Integración con aplicaciones de salud y bienestar para un enfoque holístico de la gestión de la salud. 3. Acceso a una red de profesionales de la salud para consultas y consejos virtuales.

- **Recordatorios y alarmas:** el panel de control también podría proporcionar recordatorios y alarmas para actividades críticas relacionadas con la salud, como el régimen de medicamentos de José y el programa de ejercicios de Carlos.
- **Compartir con profesionales de la salud:** el panel de control podría permitir el intercambio seguro de datos de salud con profesionales de la salud, como Melissa,

Tabla 3.2: Historias de usuario para la persona Melissa

Requerimientos	Imprescindibles	Agradable tener
Funcional	1. Una aplicación que integre un calendario y recordatorios para programar citas y medicamentos. 2. Un diario para que los pacientes puedan hacer seguimiento de su estado de ánimo y pensamientos. 3. Una plataforma que ofrezca recursos educativos y vídeos sobre el Trastorno por Déficit de Atención e Hiperactividad (TDAH).	1. Una función que permita a los pacientes hacer seguimiento de sus patrones de sueño. 2. Una herramienta que ayude a los pacientes a manejar sus niveles de estrés. 3. Una biblioteca de meditaciones guiadas y ejercicios de <i>mindfulness</i> .
Interacción con el usuario	1. Una interfaz fácil de usar con instrucciones claras e iconos. 2. Una plataforma que permita a los pacientes enviar actualizaciones a sus terapeutas. 3. Una herramienta que proporcione una plataforma segura y confidencial para que los pacientes se comuniquen con sus terapeutas.	1. Un chatbot impulsado por inteligencia artificial que brinde apoyo personalizado a los pacientes. 2. Una plataforma que ofrezca servicios de teleterapia. 3. Una herramienta que se integre con dispositivos portátiles y haga seguimiento a la actividad física.
Contexto social	1. Una plataforma que ofrezca una comunidad de apoyo y aliento para los pacientes con TDAH. 2. Una función que permita a los pacientes conectarse con otros que comparten su condición. 3. Una herramienta que proporcione recursos para familiares y cuidadores.	1. Una aplicación que se integre con plataformas populares de redes sociales. 2. Una plataforma que ofrezca grupos de apoyo entre pares. 3. Una herramienta que proporcione recursos para escuelas y profesores.

lo que le permitiría monitorear el progreso de sus pacientes y brindar orientación en consecuencia.

- **Recursos educativos:** el panel de control también podría proporcionar recursos educativos, como recetas saludables e información sobre la técnica Pomodoro y prácticas de atención plena, para apoyar el viaje de salud de los usuarios.
- **Accesibilidad:** el panel de control debe ser accesible y fácil de usar, considerando la movilidad limitada, la marcha lenta y la visión y audición en declive de José.

Tabla 3.3: Historias de usuario para la persona Carlos

Requerimientos	Imprescindibles	Bueno tener
Funcionales	1. Un reloj inteligente que rastree la frecuencia cardíaca, la oxigenación y la temperatura. 2. Una aplicación que rastree las rutinas de ejercicio y el progreso. 3. Una plataforma que proporcione sugerencias de recetas saludables.	1. Una herramienta que se integre con un dispositivo portátil para rastrear la actividad física. 2. Una función que proporcione una base de datos de nutrición y sugiera planes de comida. 3. Una plataforma que se integre con un dispositivo portátil para rastrear los niveles de hidratación.
Interacción del usuario	1. Una interfaz fácil de usar con instrucciones claras e iconos. 2. Una plataforma que se integre con un reloj inteligente. 3. Una aplicación que proporcione retroalimentación en tiempo real sobre el progreso.	1. Una herramienta que proporcione mensajes y recompensas motivacionales. 2. Una aplicación que proporcione señales de audio e instrucciones verbales. 3. Una plataforma que se integre con redes sociales y permita a los usuarios compartir su progreso con otros.
Contexto social	1. Una plataforma que proporcione una comunidad de apoyo y motivación para personas que quieren llevar un estilo de vida saludable. 2. Una herramienta que proporcione recursos para personas con antecedentes familiares de enfermedades del corazón. 3. Una aplicación que se integre con redes sociales para conectarse con amigos y familiares.	1. Una plataforma que ofrezca grupos de apoyo entre pares. 2. Una aplicación que se integre con redes sociales para rastrear el progreso de amigos y familiares. 3. Una función que permita a los usuarios competir con amigos y familiares en desafíos saludables.

- **Privacidad y seguridad de datos:** la plataforma debe garantizar la privacidad y seguridad de los datos, asegurando que la información de salud de los usuarios esté protegida en todo momento.

Este diseño integrador se alinea con las necesidades y requisitos de cada persona y proporciona una solución integral a sus preocupaciones relacionadas con la salud. Por lo tanto, se partió de estos puntos para encontrar conceptos comunes y discutirlos

para ver cuáles eran necesarios para crear un marco conceptual que reflejara las ideas adquiridas. La Tabla 3.4 proporciona una visión general condensada de las ideas generadas sus resultados de evaluación y el razonamiento detrás de ellos.

Tabla 3.4: Propuestas para posibles dimensiones del marco de trabajo

Propuesta	Resultado	Evaluación
Predicción de salud	Rechazada	Esto puede generar preocupaciones sobre la privacidad y seguridad de los datos, así como sobre la precisión de las predicciones, lo que puede tener un impacto negativo en la salud de los usuarios.
Integración con dispositivos médicos	Rechazada	Los dispositivos médicos pueden requerir aprobación regulatoria, lo que puede ser una barrera significativa para la implementación y está fuera del alcance de esta tesis.
Ludificación	Rechazada	Esto puede no ser bien recibido por todos los usuarios, ya que algunos pueden preferir un enfoque más serio para el seguimiento y monitoreo de la salud.
Interfaces de usuario como servicios	Aceptada	Esta característica cumple con el requisito de accesibilidad, asegurando una experiencia de usuario amigable y accesible para aquellos con limitaciones de movilidad e impedimentos sensoriales.
Interacciones sensibles al contexto	Aceptada	Esta dimensión respalda la provisión de recordatorios relevantes, alineándose con el objetivo de la plataforma de ayudar a los usuarios en sus trayectorias de salud. También satisface el requisito de recursos educativos, al ofrecer recursos para apoyar la salud de los usuarios.
Gestión de datos centrada en el usuario	Aceptada	Esta propuesta avanza en el objetivo de la plataforma de permitir a los usuarios compartir de, forma segura, sus datos de salud con profesionales de la salud. También aborda el requisito de privacidad y seguridad de los datos, protegiendo la información de salud de los usuarios.

3.4.3

Prototipar

Después de seleccionar cuidadosamente las dimensiones más aplicables al escenario integrador y discutir las en términos de la revisión de la literatura relevante, se describe

el marco conceptual basado en los ejes clásicos de usabilidad. Inspirados en el trabajo de Gutwin y Greenberg [67], la Tabla 3.5 resume nuestros hallazgos.

Tabla 3.5: Marco de diseño de interacción para sistemas de salud con Internet de las Cosas

Componentes de usabilidad	Interfaces de usuario como servicios	Interacciones sensibles al contexto	Gestión de datos centralizada en el usuario
Efectividad	Proporciona a los usuarios la información y herramientas necesarias para gestionar su salud, de manera efectiva.	Detecta el contexto de los usuarios y les presenta información relevante.	Permite a los usuarios gestionar y entender sus datos de salud, de manera personalizada y significativa.
Eficiencia	Optimiza las interacciones del usuario con el sistema para minimizar la cantidad de pasos necesarios para completar tareas.	Anticipa las necesidades del usuario basándose en el contexto y proporciona información relevante sin que el usuario tenga que buscarla activamente.	Permite a los usuarios acceder y manipular sus datos de salud, de manera fácil y rápida.
Satisfacción	Ofrece una experiencia positiva y amigable que aliente a los usuarios a utilizar el sistema regularmente.	Crea una sensación de conciencia y control para los usuarios sobre su salud y bienestar.	Permite a los usuarios sentirse seguros en la comprensión de su salud y cómo gestionarla efectivamente.

Además de presentar las tres dimensiones según los ejes de usabilidad, también creamos sus correspondientes definiciones:

- **Interfaces de usuario como servicios:** este aspecto del marco se centra en diseñar e implementar interfaces de usuario como servicios independientes, modulares y escalables. El objetivo es proporcionar a los usuarios una experiencia adecuada en múltiples dispositivos y plataformas, manteniendo la interfaz más accesible.
- **Interacciones sensibles al contexto:** esta característica del marco enfatiza la importancia de comprender y utilizar el contexto del usuario para crear interacciones más personalizadas y significativas. Esto puede incluir tener en cuenta

la ubicación del usuario, el tiempo, el dispositivo, las condiciones físicas y otros factores relevantes para proporcionar información y recomendaciones relevantes.

- **Gestión de datos centrada en el usuario:** la dimensión final del marco prioriza la gestión de los datos del usuario en función de sus necesidades y preferencias. Esto incluye garantizar la privacidad, seguridad, control sobre su información personal y ofrecer herramientas para que los usuarios comprendan mejor y tomen decisiones sobre sus datos.

3.4.4

Validar

La validación del marco conceptual se puede demostrar explorando su aplicación en el desarrollo de un compañero digital para personas con diabetes. Para ello, consideremos un escenario hipotético en el que una persona con diabetes utiliza un compañero digital desarrollado, utilizando el marco propuesto. Se eligió la diabetes porque es una de las condiciones crónicas más prevalentes en México [68]. La Tabla 3.6 resume esta propuesta según el marco.

En primer lugar, las Interfaces de usuario como servicios permitirán al compañero digital proporcionar interfaces de usuario personalizadas que se adapten a las necesidades específicas de las personas con diabetes. Esto se puede lograr mediante la integración de dispositivos portátiles, como un reloj inteligente o un monitor de actividad física, y aplicaciones móviles, como aplicaciones que complementan los sensores continuos de glucosa diseñados específicamente para personas con diabetes. El reloj inteligente mostrará métricas de salud esenciales, como los niveles de glucosa, la presión arterial y la frecuencia cardíaca. Al mismo tiempo, la aplicación móvil proporcionará una plataforma para registrar la ingesta de alimentos, hacer un seguimiento de la actividad física y establecer recordatorios para la medicación.

A continuación, las interacciones sensibles al contexto permitirán al compañero digital comprender el contexto en el que la persona está utilizando el sistema y proporcionar información y recomendaciones relevantes. Por ejemplo, suponga que la persona está buscando una opción de comida saludable. En ese caso, el compañero digital puede utilizar datos de *GPS* e Internet para determinar la ubicación y proporcionar una lista de opciones saludables en el menú del restaurante. Además, el compañero digital puede integrarse con plataformas de salud y bienestar para proporcionar recomendaciones personalizadas basadas en las preferencias dietéticas, las condiciones de salud y las metas individuales de la persona.

Por último, la Gestión de datos centrada en el usuario garantizará que los datos del compañero digital sean almacenados, gestionados y controlados por el usuario. El

Tabla 3.6: Conceptos de diseño de interacción de un compañero digital para personas con diabetes

Componentes de usabilidad	Interfaces de usuario como servicios	Interacciones sensibles al contextos	Gestión de datos centralizada en el usuario
Efectividad	Interfaces de usuario personalizadas y adaptadas a las necesidades específicas de las personas con diabetes, como un reloj inteligente que muestra los niveles de glucosa, la presión arterial y la frecuencia cardíaca, y una aplicación móvil para registrar la ingesta de alimentos y realizar un seguimiento de la actividad física.	Información y recomendaciones relevantes basadas en la ubicación, preferencias dietéticas, condiciones de salud y objetivos del individuo.	Capacidad para que los usuarios compartan sus datos con proveedores de atención médica si así lo desean.
Eficiencia	Integración de dispositivos portátiles, como relojes inteligentes o monitores de actividad física, y aplicaciones móviles, para proporcionar métricas de salud en tiempo real y recomendaciones personalizadas.	Utilización de <i>GPS</i> y datos de Internet para proporcionar información y recomendaciones relevantes en tiempo real.	Soluciones de almacenamiento seguras y encriptadas para almacenar datos de salud.
Satisfacción	Interfaces personalizadas que se adaptan a las preferencias individuales, lo que hace que el sistema sea más fácil de usar y comprender.	Recomendaciones conscientes del contexto que brindan valor al usuario, aumentando su sensación de control sobre su salud.	Propiedad y control de los datos, lo que brinda tranquilidad y confianza a los usuarios en el sistema.

compañero digital se diseñará para utilizar soluciones de almacenamiento seguras y encriptadas, a través de plataformas compatibles con la Ley de Portabilidad y Responsabilidad del Seguro de Salud (HIPAA) [69] para almacenar datos de salud y proporcionará

a los usuarios la capacidad de compartir sus datos con proveedores de atención médica si así lo desean.

De esta manera, combinar el marco propuesto, con dispositivos y plataformas de Internet de las Cosas existentes, puede proporcionar a los usuarios una experiencia fluida e integrada y representa una intención de amalgamar el creciente cuerpo de investigación en esta área.

Arquitectura para compañeros digitales

Contenido

4.1	Requerimientos funcionales y no funcionales	34
4.2	Actores del sistema	36
4.3	Casos de uso	38
4.4	Vista por capas	42
4.5	Vista de servicios	45
4.6	Vista lógica	66

Actualmente, se tienen avances en tecnologías de sensores, microcontroladores y sistemas de radiofrecuencia, los cuales han facilitado el desarrollo de sistemas capaces de medir diferentes parámetros físicos en diversos entornos. En paralelo a los avances tecnológicos, se tienen avances en la creación de sistemas centrados en el usuario para asistir a los usuarios a realizar alguna tarea específica, particularmente se tienen avances en asistentes virtuales de voz como *Siri*, *Google Assistant* y *Alexa*. Estas progresiones subrayan la necesidad de una integración perfecta de estos sistemas para lograr objetivos específicos y sacar el mayor provecho de los dos aspectos: el tecnológico y el de *software*. Los compañeros digitales, que personifican esta tendencia, son sistemas asistidos por computadora, que están diseñados para crear sinergia con los usuarios humanos. Son un nexo entre la eficiencia de los procesos rigurosos, la gestión de datos voluminosos y las interacciones humanas.

En respuesta al imperativo de integrar estas tecnologías y innovaciones científicas, ha surgido una propuesta convincente para desarrollar una arquitectura de *software* para compañeros digitales que pretende ayudar a pacientes con enfermedades crónicas

como la diabetes. La arquitectura propuesta comprende varios servicios, incluido el reconocimiento de voz y el control de seguimiento biométrico. Estos elementos son parte integral de la capacidad del sistema para brindar respuestas personalizadas y efectivas al usuario, con el objetivo de mejorar la calidad de vida de las personas que padecen esta enfermedad.

Este capítulo está estructurado de la siguiente manera. Primero, se presentan los requerimientos funcionales y no funcionales en la Sección 4.1, los actores que interactúan con el sistema en la Sección 4.2 y los casos de uso en la Sección 4.3. Posteriormente, se presentan diferentes vistas de la arquitectura: en la Sección 4.4 la vista por capas, en la Sección 4.5 la vista de servicios y, por último, la vista lógica en la Sección 4.6.

4.1 Requerimientos funcionales y no funcionales

Se define como CoDiSPa (Compañero Digital para apoyo a la Salud para Pacientes) al sistema y como compañero digital al agente inteligente para distinguir entre el sistema completo y el componente.

Se han identificado algunos problemas recurrentes que las personas con enfermedades crónicas tienen debido a su condición de salud. Estos problemas conforman los requerimientos funcionales que el sistema CoDiSPa realiza:

- RF1: Gestión de citas. Para la persona con enfermedad crónica, los familiares o persona que apoya a la persona, es difícil realizar la gestión de citas con el objetivo de realizarse análisis clínicos o revisiones médicas y, con ellas, tener un correcto tratamiento médico.
- RF2: Seguimiento médico. La persona con enfermedad crónica no registra las acciones que realiza y mucho menos la relación que guarda con algún parámetro biomédico asociado, de manera sistemática o regular.
- RF3: Adherencia a toma de medicamentos. Para las personas con enfermedad crónica es difícil dar seguimiento a las tomas de sus medicamentos, especialmente cuando son más de tres y con distintos horarios de ingesta.

También se han considerado los siguientes requerimientos no funcionales para la creación de CoDiSPa:

- RNF1: Adaptabilidad. La capacidad de adaptación se refiere al proceso de aprendizaje que el sistema tiene y que puede modificar su comportamiento para adaptarse, cada vez con mayor precisión, al usuario. Se puede descubrir conocimiento

empleando los datos recolectados de las interacciones del usuario con el sistema CoDiSPa.

RNF2: Personalización. CoDiSPa cuenta con una personalización de carácter tal que aparenta ser un humano. Esto tiene por objetivo que el usuario se sienta con empatía y con ello lograr una adherencia al empleo del sistema para mejorar hábitos de salud. Si se desarrolla la empatía entre el usuario y el sistema se puede mantener un monitoreo de los parámetros medibles a partir del uso de *wearables*, cámaras de vídeo empotradas, escuchas de audio, entre otras fuentes que nutran o retroalimenten al sistema y mejore la interacción.

RNF3: Usabilidad y experiencia de usuario. El concepto proviene del inglés *usability* y hace referencia a la facilidad con que un usuario puede utilizar un sistema de *software* con el fin de alcanzar objetivos específicos. La usabilidad, se define también, como un aspecto controlado del diseño de la experiencia de usuario que garantiza que el usuario final no se esfuerce y no se encuentre con problemas al usar la interfaz de usuario de un sistema de *software*. Para medir el grado de usabilidad de un sistema, en este caso CoDiSPa, se cuentan con tres dimensiones básicas: efectividad, eficacia y satisfacción. Considerando los resultados del trabajo descrito en el [Capítulo 3](#), se requiere que los conceptos resultantes sean usados para la creación del sistema CoDiSPa.

RNF4: Seguridad de la información. La seguridad de la información es el conjunto de medidas preventivas y reactivas de las organizaciones y sistemas tecnológicos que permiten resguardar y proteger la información buscando mantener la confidencialidad, la disponibilidad e integridad de datos. CoDiSPa sigue el principio de cuidar el uso y almacenamiento de los datos personales, así como cuidar el almacenamiento de la información recolectada empleando mecanismos de cifrado de datos.

RNF5: Escalabilidad. La escalabilidad es un atributo que describe la capacidad de un proceso, red, *software* u organización para crecer y gestionar una mayor demanda de uso. En el caso de CoDiSPa, nos interesa la capacidad de crecimiento del sistema con los menos cambios posibles en su configuración e incluso en su programación. Para alcanzar la escalabilidad, CoDiSPa emplea la arquitectura de microservicios, el principio de separación de responsabilidades y la buena práctica de diseño denominada alta cohesión y bajo acoplamiento.

RNF6: Modularidad. La modularidad es una forma de diseñar el *software* para que esté formado por módulos independientes. Cada módulo tiene una interfaz bien

definida y puede utilizarse de forma aislada de los demás módulos. Para CoDiSPa los módulos, en granularidad gruesa, son los microservicios que tienen bien definidas las tareas a realizar y que exponen sus funcionalidades por medio de la especificación *RESTful Services*.

La propuesta del sistema CoDiSPa, además de atender los requerimientos funcionales, atiende los requerimientos no funcionales como: la modularidad y la escalabilidad, y observa la experiencia del usuario procurando ser un sistema amigable y accesible para personas con movilidad limitada o impedimentos sensoriales. Adicionalmente, el sistema CoDiSPa tiene cuidado con la emisión de notificaciones para que éstas se alineen a los objetivos particulares del usuario y que se tenga, en todo momento, el objetivo de ayudar en la mejora de su estado de salud, así como proporcionar información educativa que fomente la mejora en sus hábitos. Por último, CoDiSPa tiene cuidado del manejo de la información compartida del usuario, cuidando la seguridad y privacidad de su información.

4.2 Actores del sistema

Cada actor debe ser entendido como un rol que puede tomar un ente que interactúa con CoDiSPa, pudiendo ser una persona, un sistema o un agente inteligente. En la Tabla 4.1 se puede observar la lista de roles y el tipo de ente que puede personificarlos.

Identificador	Actor	Tipo
H1	Paciente	Humano
H2	Cohabitante o Asistente	Humano
H3	Visitante	Humano
M1	Médico	Humano o Agente
M2	Técnico de análisis clínico	Humano o Agente
M3	Nutriólogo, Terapeuta o Psicólogo	Humano o Agente
MS	Farmacia o Farmacéutica	Humano o Agente
DC	Compañero Digital	Agente

Tabla 4.1: Actores del sistema CoDiSPa

Para el diseño del sistema CoDiSPa se definieron los siguientes ocho actores:

Paciente. Este actor es capaz de hacer peticiones al sistema, en todo momento, por distintas interfaces, principalmente por voz. Este actor puede interactuar con

CoDiSPa mediante diversos dispositivos como: una aplicación móvil, el micrófono de una bocina inteligente, las pulseras inteligentes, los relojes inteligentes y transferencias de otros subsistemas, como se muestra en la Tabla 4.2.

Cohabitante o Asistente. Este actor es capaz de hacer peticiones a CoDiSPa en todo momento por distintas interfaces, principalmente por medio de voz. Este actor puede interactuar con CoDiSPa mediante diversos dispositivos como: una aplicación móvil, el micrófono de una bocina inteligente, las pulseras inteligentes, los relojes inteligentes y transferencias de otros subsistemas.

Visitante. Este actor sólo tiene permitido realizar consultas al sistema CoDiSPa o búsquedas vía Internet.

Médico. Este actor puede ser el personal médico o un sistema automatizado de la clínica o consultorio médico. Las responsabilidades que puede tomar este actor son de receptor de reportes, alertas o alarmas emitidos por CoDiSPa; el Médico también puede interactuar con el Compañero Digital mediante una aplicación móvil o un formato de intercambio de datos para integrarse con el sistema propio del hospital, laboratorio, clínica o consultorio médico. El Médico puede hacer preguntas para que CoDiSPa se las haga llegar al Paciente, registre las respuestas y las haga llegar al Médico.

Técnico de análisis clínico. Este actor puede ser una persona o un agente. Las responsabilidades del actor son de receptor de solicitudes de agenda de citas y de resultados de los análisis clínicos del Paciente.

Nutriólogo, Terapeuta o Psicólogo. Este actor puede ser una persona o un agente. Sus responsabilidades son las de recibir peticiones de cita o consultas directas en forma de preguntas breves. Este actor puede también enviar mensajes o información importante al Paciente.

Farmacia o Farmacéutico. Este actor puede ser un agente o una persona. Sus responsabilidades son las de recibir solicitudes para el envío de medicamentos con base en la prescripción médica.

Compañero digital. Este actor interactúa con el sistema CoDiSPa y puede desencadenar distintas funcionalidades como si de una persona se tratase. Adicionalmente, este actor se encarga de procesar la información que le llega por los varios canales de CoDiSPa y es capaz de comunicarse, de manera multimodal, con los demás actores mediante texto, voz o datos, como se muestra en la Tabla 4.2.

Dispositivo	Descripción	Tipo	Modalidad
D1	Bocina inteligente	Entrada y salida	Voz
D2	Cámara inteligente	Entrada	Gráfico, voz y vídeo
D3	Teléfono inteligente	Entrada y salida	Texto, gráfico, vibración, voz y vídeo
D4	Pulsera inteligente	Entrada y salida	Texto, vibración y voz
D5	Báscula inteligente	Entrada y salida	Texto, gráfico
D6	Pantalla inteligente	Entrada y salida	Texto, gráfico, vídeo y voz
D7	Computadora personal	Entrada y salida	Texto, gráfico, voz, vídeo

Tabla 4.2: Dispositivos que interactúan con el sistema CoDiSPa

4.3 Casos de uso

Producto del análisis de requerimientos se generaron los casos de uso siguientes:

CU 1: Gestión de citas médicas

- CU 1.1: Registro de cita médica ordinaria
- CU 1.2: Registro para revisión médica programada
- CU 1.3: Registro para atención médica de emergencia

CU 2: Seguimiento médico

- CU 2.1: Envío de parámetros biométricos
- CU 2.2: Análisis y visualización de parámetros
- CU 2.3: Retroalimentación de resultados clínicos
- CU 2.4: Recomendaciones para el bienestar

CU 3: Adherencia de toma de medicamentos

- CU 3.1: Registro de prescripción médica
- CU 3.2: Recordatorio de toma de medicamentos
- CU 3.3: Recomendación educativas sobre el tratamiento
- CU 3.4: Registro de emociones del Paciente

Para atender cada caso de uso, que se muestran en la Figura 4.1, se realizó la separación de responsabilidades con lo que se diseñaron ocho componentes de granularidad gruesa. Los componentes generados, al articularse entre sí, atienden a los requerimientos funcionales, como se muestra en la Figura 4.2. Para atender los requerimientos no funcionales, se eligieron las estructuras de datos, las diferentes pilas de tecnologías de desarrollo y los elementos para el despliegue.



Figura 4.1: Diagrama de casos de uso para el sistema CoDiSPa

4.3.1

Gestión de citas médicas

La gestión de citas médicas comprende tres actividades, como puede verse en la Figura 4.1, a saber: 1) registro de cita ordinaria, 2) registro de cita para revisión programada y 3) registro para atención médica de emergencia.

El registro de una cita ordinaria se puede dar cuando el Paciente hace la solicitud mediante voz o texto. También, el registro de una cita ordinaria puede darse cuando el compañero digital le propone al Paciente solicitar una cita ordinaria debido a la detección de parámetros biométricos que sugieran la necesidad de atención médica no urgente.

El registro para revisión médica programada, se puede dar cuando el Médico, después de una revisión, le indica al Paciente que deberá acudir después de un periodo determinado o en una fecha dada. El registro también puede darse cuando, en complemento con la atención médica, se requiere acudir a la cita con resultados de análisis clínicos para ayudar al seguimiento o diagnóstico por parte del médico.

El registro para atención médica de emergencia, se puede dar por dos vías. Primero, si el Paciente le manifiesta por medio de voz o texto al compañero digital que se encuentra mal y requiere atención médica de emergencia. Segunda, si el compañero digital detecta que los parámetros biométricos dan indicios de que el Paciente se encuentre en mal estado puede solicitar los servicios de emergencia. El compañero digital, después de la detección del estado del Paciente, puede preguntarle al Paciente si requiere servicios de emergencia, si el Paciente no le contesta el compañero digital deberá solicitar los servicios.

Este caso de uso, para la gestión de citas médicas, tiene como responsabilidades: 1) agendar las citas con diferentes entidades o actores del sistema, 2) recordar la cita próxima al Paciente y 3) tener en observación el estado biométrico del Paciente para solicitar atención de emergencia.

4.3.2

Seguimiento médico

El seguimiento médico comprende cuatro actividades, como puede verse en la Figura 4.1, a saber: 1) envío de parámetros biométricos, 2) análisis y visualización de parámetros, 3) retroalimentación de resultados y 4) recomendaciones para el bienestar.

El envío de parámetros biométricos se da periódicamente para observar el estado general del Paciente. El sistema lanza la petición de obtención de los valores biométricos cada periodo de tiempo, previamente definido en la configuración del sistema. De manera independiente, el Paciente o el Médico puede solicitar el estado más reciente para observar el estado de los parámetros biométricos.

El análisis de los parámetros es una tarea que procesa los datos obtenidos y almacenados para contar con una visualización de los parámetros o proporcionar, en audio, una respuesta del análisis al Paciente o Médico.

La retroalimentación de los resultados clínicos es una tarea que se da cuando un sistema externo o el Médico alimenta al sistema propuesto con los resultados clínicos. El formato proporcionado con los resultados es analizado y se obtienen de él, los valores para cada parámetros clínico. Una vez que se tienen los pares, parámetro clínico-valor, se hace una valoración del estado general y, ésta es enviada al médico para su validación y comentarios. Una vez que el médico valida y agrega sus comentarios, éstos son notificados al Paciente, de manera auditiva o en texto.

Las recomendaciones para el bienestar son tareas que se realizan periódicamente desde el sistema propuesto. A partir de información validada, se tienen directrices o recomendaciones para el Paciente basadas en las mejores prácticas para tratar el padecimiento específico. Se generan las recomendaciones y se consulta el estado de parámetros biométricos y el último estado emocional, registrado en el sistema, para identificar si el Paciente puede recibir las recomendaciones en ese momento. Si no es el momento adecuado para que el Paciente reciba la notificación, se intentará nuevamente cuando las condiciones emocionales y biométricas sean adecuadas. Las recomendaciones están conformadas por información nutricional, técnicas de manejo emocional, técnicas de actividad física, información sobre el padecimiento, entre otros temas.

Este caso de uso, para el seguimiento médico, tiene como responsabilidades: 1) Mantener todos los registros de los parámetros biométricos actualizados en el sistema propuesto, 2) tener disponible el análisis y visualización para los parámetros biométricos y los resultados clínicos, 3) tener disponible los resultados clínicos validados y comentados por el Médico y 4) tener la información protegida ante el acceso no autorizado.

4.3.3 Adherencia a la toma de medicamentos

La adherencia a la toma de medicamentos comprende cinco actividades, como puede verse en la Figura 4.1, a saber: 1) registro de prescripción médica, 2) recordatorios de toma de medicamentos, 3) recomendaciones educativas sobre el tratamiento y 5) registro de emociones del Paciente.

El registro de la prescripción médica es una tarea que realiza el médico cuando envía al sistema propuesto la prescripción médica. El sistema toma el formato de prescripción y obtiene de él los medicamentos, horarios de toma y recomendaciones.

Con la información el sistema propuesto genera los recordatorios para cada toma de medicamento con la información correspondiente.

Los recordatorios de toma de medicamentos son tareas que se realizan periódicamente para recordar las tomas e indicaciones, que el Médico hace al Paciente, para llevar adecuadamente su tratamiento. El sistema propuesto pregunta al Paciente si ha realizado la toma del medicamento correspondiente y registra el evento para dejar una traza que el Médico puede consultar en todo momento.

Las recomendaciones educativas sobre el tratamiento son notificaciones informativas para el Paciente, con información validada por el Médico específicamente sobre el tratamiento medicamentoso. Estas recomendaciones, se dan de manera periódica buscando un estado emocional y biométrico adecuado para que generen un impacto positivo en el Paciente.

El registro de emociones del Paciente es una tarea generada por el sistema propuesto. El sistema periódicamente busca el momento, biométricamente y emocionalmente adecuados, para abordar al Paciente y preguntarle sobre las emociones que está experimentando. El sistema almacena las respuestas para obtener una interpretación de la misma. El estado emocional del Paciente puede ser consultado por el Médico, siempre que éste lo requiera.

Este caso de uso, para mejorar la adherencia a la toma de medicamentos, tiene como responsabilidades: 1) tener actualizado el tratamiento detallado del Paciente, 2) tener el registro detallado de las notificaciones necesarias para que el Paciente atienda la toma de medicamentos y recomendaciones del Médico, 3) el sistema tendrá notificaciones sobre el tratamiento médico validadas por el Médico que puedan ser transmitidas al Paciente para la mejora de su comprensión del padecimiento y el tratamiento correspondiente, 4) tener registro de los estados emocionales para contar con una escala que permita valorar al Paciente por el Médico y 5) resguardar la información obtenida con las medidas de protección al acceso no autorizado.

4.4

Vista por capas

La arquitectura de CoDiSPa se compone de cuatro capas en las que se agrupan sus microservicios, a saber: Capa de Aplicación, Capa Lógica, Capa de Servicios y Capa de Comunicación.

En la Capa de Aplicación se tienen los microservicios: *Reconocedor del Habla*, *Locutor Proactivo*, *Observador Biométrico*. En la Capa Lógica se tiene el microservicio *Compañero Digital*. En la Capa de Servicios se tienen los microservicios: *Buscador Semántico*, *Analizador de comportamiento* y *Manejador de Conocimiento*. En la Capa Comunicación se tiene el microservicio *Mensajero*. Thinking En la Figura 4.2 se puede

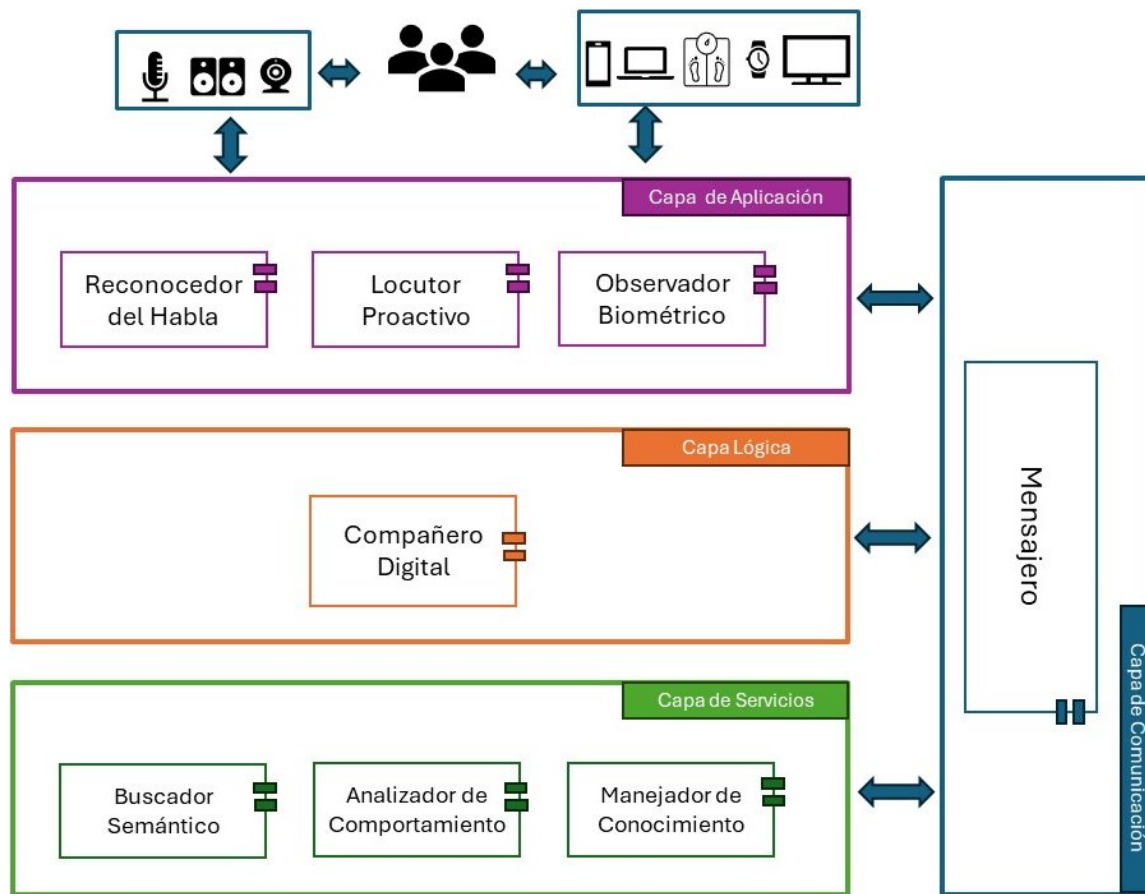


Figura 4.2: Vista por capas de CoDiSPa

observar que la comunicación entre capas es a través de la Capa de Comunicación. El usuario se comunica con los diversos dispositivos, que conforman el entorno de infraestructura pervasiva, a través de la Capa de Aplicación.

4.4.1 Capa de aplicación

En esta capa se agrupan tres microservicios: *Reconocedor del Habla*, *Locutor Proactivo* y *Observador Biométrico*. Estos microservicios son los encargados de recolectar la información del entorno pervasivo incluyendo las peticiones del usuario para alimentar, con esta información de contexto, a los demás microservicios del sistema CoDiSPa. En esta capa se llevan a cabo operaciones como la identificación del Paciente, su estado

biométrico, así como las respuestas multimodales que se le dan a dicho usuario. Adicionalmente, en esta capa se cuenta con los mecanismos adecuados para comunicarse con los diversos dispositivos, ya sea de entrada o de salida del sistema. Cada microservicio en esta capa tiene sus propias responsabilidades y se encuentran estratégicamente cohesionados para ser eficientes en los servicios que realizan.

4.4.2 Capa Lógica

En esta capa se cuenta con un microservicio: *Compañero Digital*. Este es el encargado de coligar y explotar todos los servicios proporcionados por los microservicios de las otras capas en la arquitectura. Cuenta con la lógica que atiende a las reglas de negocio para satisfacer las funcionalidades a partir de las que se diseñó a CoDiSPa.

4.4.3 Capa de Servicios

En esta capa se agrupa a los servicios que dan soporte a las búsquedas de información y conocimiento para la operación del sistema CoDiSPa. Los microservicios se encuentran separados en tres rubros: manejo de conocimiento, búsquedas de información y generación de conocimiento. Cada microservicio es independiente de los demás y las operaciones que puede realizar se exponen a través de una *API*. La comunicación entre los microservicios es a través de la capa de comunicación, ya que es la encargada de ser la intermediaria con capacidades de gestionar concurrentemente las peticiones y respuestas de todos los microservicios de la arquitectura de CoDiSPa. La comunicación se da empleando el esquema de mensajes propio de un enrutador y transformador de mensajería, propio del patrón de integración *Broker*.

4.4.4 Capa de comunicación

En esta capa se cuenta con un microservicio: *Mensajero*. Esta capa es la encargada de comunicar a todos los microservicio de la arquitectura de CoDiSPa. Esencialmente, es una capa *middleware* que se encarga de enrutar las peticiones entre todos los microservicios, facilitando la operación concurrente y permitiendo la escalabilidad del sistema. Si es necesario, se pueden crear más ejemplares para cada microservicio y con esto CoDiSPa será capaz de atender un número masivo de información y contextos. Esencialmente, esta capa es un sistema de enrutamiento y transformación de mensajes que se envían y reciben entre los microservicios de CoDiSPa.

Para satisfacer el requerimiento no funcional de diseñar modularmente mediante servicios, se revisó el estado de la técnica de aplicaciones Web [70, 71, 72, 73] y se tomó como base la arquitectura de microservicios [74, 75] para ser la encargada de resolver la comunicación entre microservicios. En los últimos años se ha transitado de sistemas con una arquitectura monolítica hacia una arquitectura distribuida, como resultado se tienen las arquitecturas basada en el patrón de microservicios.

El patrón de arquitectura de microservicios, como se muestra en la Figura 4.3, se puede definir como la estructura lógica de componentes que hace que se abstraigan las funcionalidades deseadas en la aplicación final, para agrupar en distintos componentes que se van a encontrar de manera independiente y distribuida en el contexto del propio sistema. Los componentes exponen las funcionalidades, a través de algún mecanismo de comunicación como puede ser una *API*, para ser usadas por los clientes del sistema o componentes del mismo. La arquitectura basada en el patrón de microservicios tiene algunas ventajas, e.g., se pueden generar varios ejemplares para cada microservicio para aumentar la capacidad de atención de peticiones de los clientes o componentes, se tiene tolerancia a fallos al poder replicar los microservicios, se puede balancear la carga si la demanda es alta, entre otras. Algunas desventajas para los sistemas basados en microservicios son: el control de versiones de los microservicios, la posible complejidad para comunicar los microservicios entre sí, la latencia de los medios de comunicación, entre algunas otras. Sin embargo, una de las ventajas principales del uso de microservicios, es que hacen posible articular aplicaciones complejas usándolos como componentes. El uso de microservicios para componer sistemas hace eficiente el poder de cómputo y ayuda a la reutilización de componentes de *software*. También, los microservicios se pueden desplegar de manera local, a través de la infraestructura de la nube en Internet y una mezcla de ambos, dando como ventaja de esto, la alta disponibilidad de las funcionalidades.

Actualmente, *REST* es la lógica más utilizada para la construcción de *APIs* de aplicaciones de *software* interactivas que trabajan sobre un esquema cliente servidor [76]. Es muy importante tener en cuenta que hablar de *APIs*, hablar de *REST* y hablar de *RESTful API* significa hablar de tres conceptos totalmente diferentes. Una *API* es una abstracción de funciones y procedimientos. *REST* es una lógica de restricciones y recomendaciones bajo la cual se puede construir una *API*, i.e., es una especificación o guía de principios. Finalmente, una *RESTful API* es una *API* ya implementada que está construida utilizando la lógica de *REST*.

Las *RESTful API* funcionan estrictamente bajo una arquitectura cliente servidor utilizando *HTTP* como protocolo de comunicación. En *REST* cada procedimiento de una

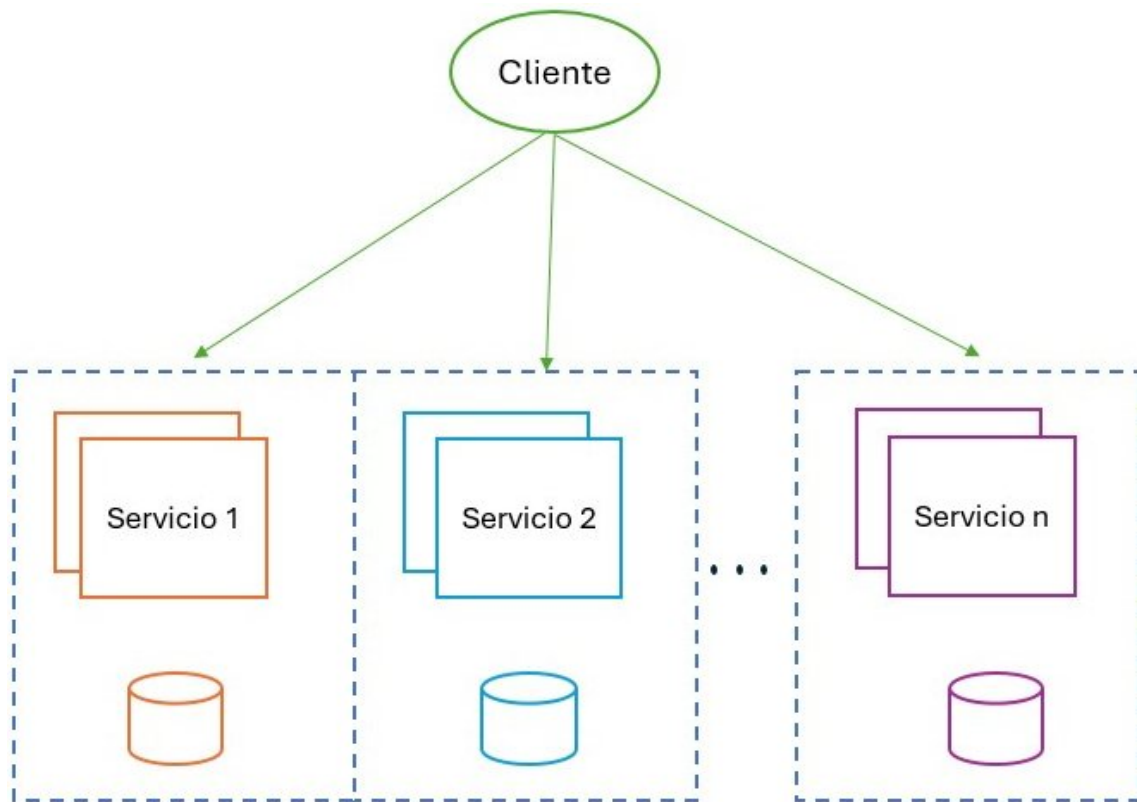


Figura 4.3: Patrón de arquitectura: microservicios

aplicación está compuesto por tres partes importantes: un verbo *HTTP*, una dirección *URI* única y la información necesaria que requiere servidor para satisfacer la petición, incluyendo la información de autenticación del cliente. Los verbos *HTTP* son identificadores que determinan el objetivo de una petición del cliente. Entre los verbos *HTTP* más utilizados tenemos: *get*, *post*, *put*, *update*, *patch* y *delete*. Cada uno de estos cumple con un propósito específico. El verbo *get* se utiliza para los procedimientos que devuelven información al cliente, el verbo *post* se utiliza para que el cliente pueda crear recursos dentro del servidor, i.e., agregar un registro dentro de una base de datos, el verbo *put* y el verbo *patch* se utilizan para que el cliente pueda editar recursos ya existentes dentro del servidor y, finalmente, el verbo *delete* se utiliza para eliminar recursos dentro del servidor. Cuando se usa la especificación *REST*, la implementación de estos procedimientos del servidor no le interesa, al cliente ni al servidor, la tecnología empleada para su codificación. Tampoco importan las implementaciones dentro del cliente para realizar las peticiones. El cliente y el servidor son dos cajas negras intercambiando

información entre sí y a ninguno le importa en qué lenguaje está programado el otro, o qué *frameworks* utilizan. El servidor debe recibir toda la información para poder ejecutar cada petición, para ello se debe definir un formato de intercambio de datos entre las dos partes. Este formato debe ser flexible y comunmente se usa *JSON*, *XML*, formato binario o texto plano. Debido a que el servidor necesita que cada petición tenga toda la información necesaria para poder ejecutar un procedimiento, se dice que *REST* es *Stateless*, i.e., sin estado, ya que no se necesita guardar información o el estado de peticiones anteriores para poder satisfacer peticiones nuevas. Cada petición es independiente de otras, de allí el significado de sus siglas *Representational State Transfer*, sin embargo, la arquitectura *REST* si admite la implementación de *cache* para guardar la respuesta de peticiones hechas con anterioridad, y de esta forma, utilizarlas para poder satisfacer nuevas peticiones de los mismos recursos de manera rápida. Cuando se utiliza el mecanismo de *cache* en los peticiones con el verbo *get*, la ventaja de implementar nuestras *APIs* bajo la lógica *REST*, es que obtendremos *APIs* de fácil mantenimiento y tendremos una arquitectura modular y escalable.

Para el diseño de los microservicios para CoDiSPa, se decidió que se segmentarían las funcionalidades y que serían expuestas bajo la especificación *RESTful API* [77]. También se decidió, que cada microservicio estaría autocontenido, i.e., que cada uno tendría su propio proceso servidor que atiende peticiones mediante la *API* expuesta para ser consumida. Por último, se decidió emplear el patrón de integración conocido como *Broker* [78] para integrar y orquestar a los distintos microservicios del sistema CoDiSPa.

Una vez que se realizó la abstracción de los microservicios y se asignaron las responsabilidades para cada uno, se definió que la comunicación entre los distintos microservicios sería mediante el uso de mensajes. Como componente coligador en la arquitectura, tenemos al microservicio *Mensajero*, que es el encargado de gestionar los mensajes provenientes de todos los microservicios del sistema, incorporando en la arquitectura el desacoplamiento y respuesta asíncrona de los microservicios.

Se evaluaron algunas implementaciones de sistemas de enrutamiento de mensajes, derivado del patrón de integración *Broker*, tales como *MS-Message Queuing*¹, *Apache ActiveMQ*² y *Apache Kafka*³, y se decidió emplear este último para integrar los distintos microservicios. La última pieza para realizar el sistema CoDiSPa es un orquestador:

¹Sistema de cola de mensajes de Microsoft. [https://learn.microsoft.com/en-us/previous-versions/windows/desktop/msmq/ms711472\(v=vs.85\)](https://learn.microsoft.com/en-us/previous-versions/windows/desktop/msmq/ms711472(v=vs.85))

²Apache ActiveMQ es un sistema intermediario que maneja colas de procesamiento de mensajes entre sistemas. <https://activemq.apache.org/>

³Apache Kafka es una plataforma de eventos para *streaming* que permite la integración de datos, *pipelines* de datos de alto rendimiento y uso en aplicaciones de misión crítica. <https://kafka.apache.org/>

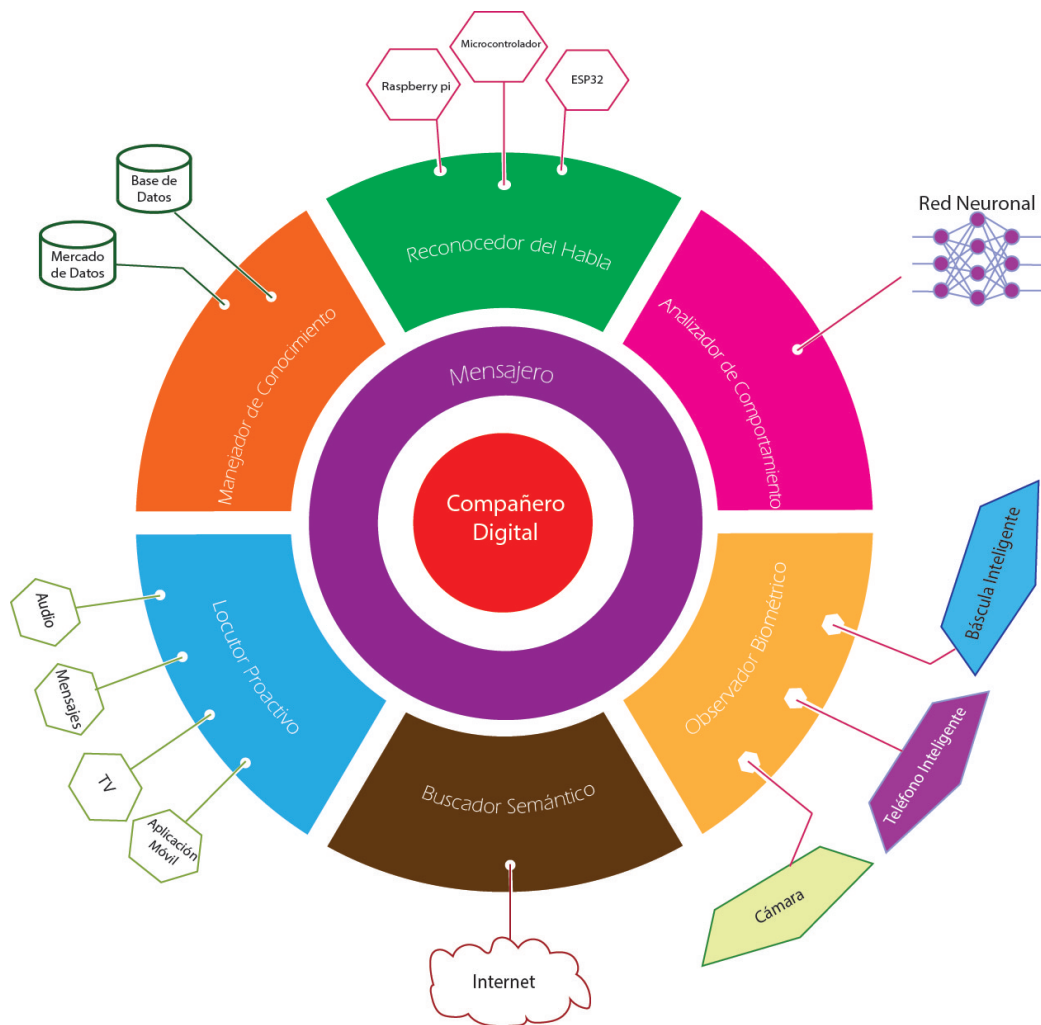


Figura 4.4: Arquitectura del sistema CoDiSPa: vista de servicios

el *Compañero Digital*, que tiene en su flujo las reglas necesarias para saber a quién delegarle cada petición realizada, ya sea por los usuarios o por él mismo.

La arquitectura que se propone está conformada por ocho microservicios. El sistema está diseñado bajo el principio de alta cohesión y bajo acoplamiento y empleando patrones de diseño probados para garantizar la flexibilidad ante los cambios futuros. Los patrones usados para el desarrollo de CoDiSPa son: Observador, Adaptador, Factory, Proxy, Decorador, Fachada, Estrategia, Inyección de dependencias, Modelo-Vista-Controlador, Singleton, Broker, Microservicios, entre otros. Los microservicios que conforman el sistema son: 1) *Reconocedor del Habla*, 2) *Locutor Proactivo*, 3) *Observador Biométrico*, 4) *Compañero Digital*, 5) *Buscador Semántico*, 6) *Analizador de Comportamiento*, 7) *Manejador de Conocimiento* y 8) *Mensajero*. Se describen a continuación los microservicios del sistema CoDiSPa de la Figura 4.4.

4.5.1

Reconocedor del Habla

Este servicio, mostrado en la Figura 4.5, recibe las señales de audio para identificar algunas frases clave, comandos u ordenes que pueden denotar acciones a realizar por solicitud del usuario.

Responsabilidades

Las responsabilidades de este servicio son reconocer órdenes o comandos del usuario y enviar las peticiones identificadas al microservicio correspondiente mediante el envío de un mensaje a través del microservicio *Mensajero*.

Interacciones

Este microservicio recibe audio del entorno y, en primera instancia, trata de identificar las frases emitidas por el Paciente, el Médico o el Cohabitante. Sin embargo, a partir de identificar de qué usuario se trata, se decide si la orden o comando será enviado o descartado. También, se cuenta con comunicación directa con el microservicio *Mensajero* y de manera indirecta con los demás microservicios de CoDiSPa.

Componentes del microservicio

Este microservicio de CoDiSPa fue diseñado para satisfacer los siguientes requerimientos funcionales:

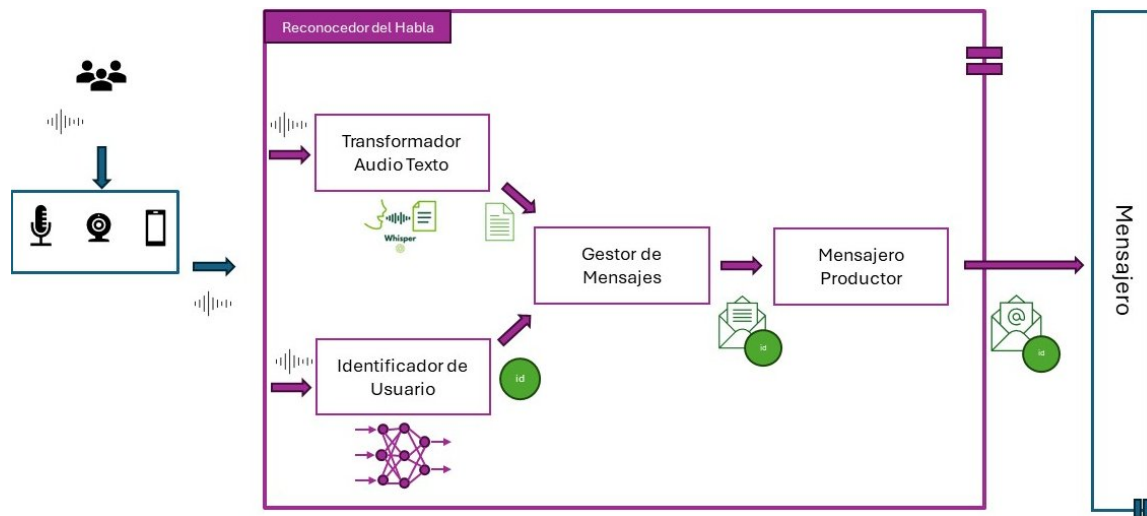


Figura 4.5: Microservicio *Reconocedor del Habla*

1. Fue implementado como un servicio para interactuar con los demás servicios de CoDiSPa
2. Puede recibir audio multi-fuente, i.e., los dispositivos de captura son múltiples y con ello se cuenta con conciencia de su contexto
3. Recibe el audio, identifica al usuario y traduce la orden para enviarla al microservicio que la atiende

También, el microservicio satisface los requerimientos no funcionales siguientes:

1. Mantiene la privacidad de datos personales que permitan a otros microservicios de CoDiSPa identificar plenamente al usuario
2. No requiere salir a Internet para operar
3. No comparte datos con servicios externos a CoDiSPa
4. El tiempo de respuesta hacia el usuario es del orden de menos de 5 segundos
5. Es capaz de identificar al usuario y sus ordenes (comandos) a una distancia menor a los 2 metros de cualquiera de los dispositivos asociados con CoDiSPa

En la Figura 4.6 y 4.7 se puede ver un ejemplo de la información que se envía y recibe, respectivamente, por el microservicio *Reconocedor del Habla*.

Los microservicios que conforman el microservicio son: *Transformador de Audio Texto*, *Identificador de Usuario*, *Gestor de Mensajes* y *Mensajero Productor*.

Funcionamiento del microservicio

Como se observa en la Figura 4.5, el microservicio *Reconocedor del Habla* obtiene señales de audio empleando los transductores electrónicos de un teléfono inteligente, pulsera inteligente, bocina inteligente o directamente de una *appliance* construida con el propósito de obtener el audio del entorno. La aplicación específicamente construida está basada en un *ESP32* y en una placa de desarrollo *Raspberry pi*. El audio es obtenido y segmentado para pasarlo por el microservicio de transformación de audio a texto. Se emplea procesamiento basado en una red neuronal para transformar el audio a texto y éste es entregado al gestor de mensajes. De manera simultánea, el mismo audio segmentado es procesado por el identificador de usuario para decidir a qué perfil de usuario pertenece el audio. Para realizar la identificación del usuario, es necesario que previamente se tengan muestras de voz de los usuarios y con ello mediante un procesamiento de muestreo, con esto se logra identificar al usuario. Una vez que el identificador de usuario sabe quién es el usuario, se entrega un *id* digital al gestor de mensajes para que éste pueda asociar la orden junto con la identificación del usuario. En ese momento, se decide si se procesa el mensaje o se descarta. Se procesa el mensaje si la orden es de usuarios con los roles de Paciente, Médico o Asistente, de lo contrario se descarta el mensaje, aunque se deja el registro en la bitácora del microservicio para posterior análisis y depuración. Si el mensaje es generado correctamente por el *Gestor de Mensajes*, se entrega al microservicio *Mensajero Productor* para que se empaquete y se envíe al *Compañero Digital* mediante el microservicio *Mensajero*. En las Figuras 4.6 y 4.7 se muestra un mensaje enviado y su respuesta.

raw	Recuerdame tomar mi medicamento mañana a las 7 y media de la noche
action	Recordatorio de toma de medicamento
search	
place	
physician	
date	mañana
time	7:30
timestamp	15:23:43
user-id	H1
issue	112

Figura 4.6: Ejemplo de petición capturada por el microservicio *Reconocedor del Habla*.

user-id	H1
issue	112
message	Se ha agendado recordatorio de toma de medicamentos

Figura 4.7: Ejemplo de respuesta procesada por el microservicio *Reconocedor del Habla*.

4.5.2

Locutor Proactivo

Este microservicio es el encargado de realizar la interacción con el usuario para comunicar información, preguntar o para dar respuesta a las peticiones del mismo.

Responsabilidades

Las responsabilidades de este microservicio son la de ubicar dónde se encuentra el usuario en el entorno y decidir cómo hacerle llegar el mensaje correspondiente. Los mensajes pueden ser comunicados en distintos modos como pueden ser en audio, texto o ambos. Se debe asegurar que el mensaje sea visto por el usuario mediante un mecanismo de verificación o *ACK* (*acknowledge*).

También, se notifican sugerencias o información de manera proactiva para apoyar al usuario en el cambio progresivo de hábitos de comportamiento que le permitan avanzar en su control sobre su estado de salud.

Interacciones

Este microservicio tiene interacción con todos los roles asociados con las personas y con el microservicio *Mensajero*.

Componentes del microservicio

Este microservicio, como puede observarse en la Figura 4.8, tiene los siguientes componentes: *Transformador Texto Audio*, *Localizador del Usuario*, *Gestor Multimodal de Mensajes* y *Mensajero Consumidor*.

Funcionamiento del microservicio

Cuando un mensaje es entregado por el microservicio *Mensajero*, el *Mensajero Consumidor* lo recibe para extraer la carga útil o información y delegar su procesamiento al *Gestor Multimodal de Mensajes*. Cuando se tiene por lo menos un mensaje en el

Gestor Multimodal de Mensajes se solicita, al elemento *Localizador del Usuario*, la ubicación del usuario y, en función de dicha ubicación, se decide si se debe transformar el mensaje de texto o si se entregará directamente en ese formato. El *Gestor Multimodal de Mensajes* decide también si entrega el mensaje solo a un dispositivo y el formato de entrega. Si no tiene la ubicación del usuario puede esperar para la entrega hasta que se ubique o si es urgente puede realizar una entrega a todos los dispositivos de salida de CoDiSPa. Una vez realizada la entrega a los dispositivos, el microservicio registra la entrega en la bitácora y avisa, mediante el microservicio *Mensajero*, que ya se atendió la entrega del mensaje.

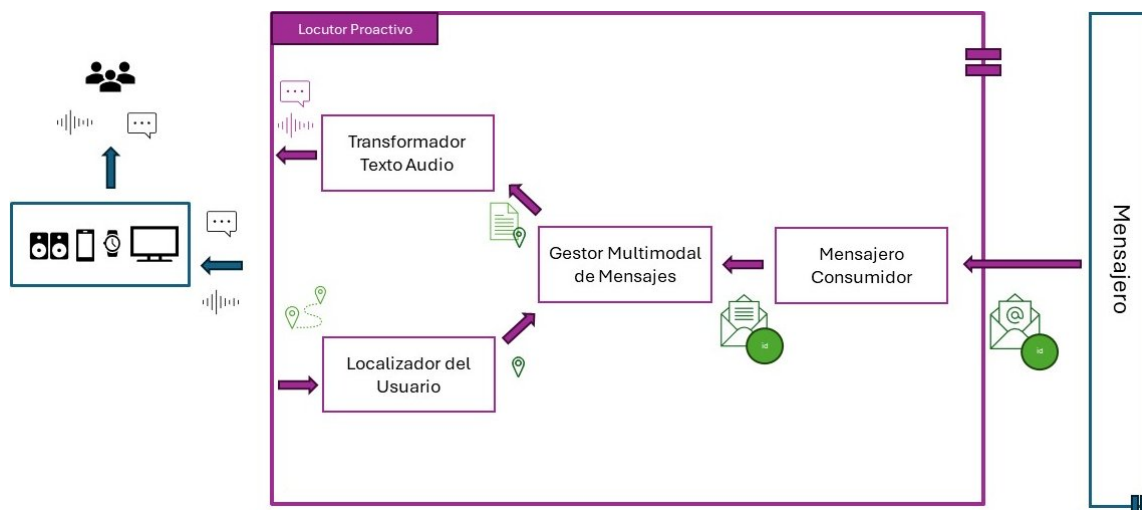


Figura 4.8: Microservicio *Locutor Proactivo*

4.5.3

Observador Biométrico

Este microservicio está basado en el concepto de *middleware* que nos ayuda a integrar los diversos dispositivos en la infraestructura disponible al entorno del usuario. De tal manera que, recolecta y estructura los patrones biométricos y de comportamiento que se usan para ser analizados y explotados. Los datos recolectados se envían al microservicio *Analizador de Comportamiento* para su análisis y para generar conocimiento para su futura explotación en la toma de decisiones del Paciente y el equipo médico.

Responsabilidades

Las responsabilidades del microservicio son las de permitir que se conecten todos los dispositivos con capacidades biométricas del entorno, obtener los datos biométricos del usuario, organizar los datos biométricos para realizar su limpieza, análisis, visualización y exportación. El procesamiento de los datos biométricos se realiza de acuerdo con las reglas para el intercambio de la información. También, el microservicio tiene como responsabilidad entregar la información estructurada para su análisis y explotación al microservicio *Analizador de Comportamiento*.

Interacciones

La interacción de este microservicio es con el usuario Paciente, para recolectar los datos biométricos, y con el usuario Médico para notificarle periódicamente el reporte de dichos parámetros. Asimismo, el microservicio tiene interacción con el microservicio *Mensajero* directa e indirectamente con los demás microservicios de CoDiSPa.

Componentes del microservicio

Los componentes del microservicio, que pueden observarse en la Figura 4.9, son: *Receptor de Datos Biométricos*, *Procesador Biométrico*, *Analizador de Estado Biométrico* y *Mensajero Productor*.

Funcionamiento del microservicio

El *Observador Biométrico* recibe los datos biométricos desde distintos orígenes como son: una pulsera inteligente, una báscula inteligente, baumanómetros, glucómetro o una aplicación móvil que concentre algunos parámetros que sean tomados desde dispositivos que no se pueden conectar al sistema. Una vez recibidos los datos, se identifica el origen en el *Receptor de Datos Biométricos* y se registran en el *Procesador Biométrico* para su transformación al formato adecuado que permitirá su interpretación y análisis. Cuando se entrega la información al *Analizador de Estado Biométrico*, este elemento concentrará o analizará los datos para dar un estado biométrico del Paciente. El estado biométrico del Paciente es producto del conocimiento del Médico. El Médico define cuáles son los umbrales para darle la interpretación resumida de los datos procesados al Paciente. El procesamiento incluye el uso de modelos de clasificación de aprendizaje automático entrenados para este propósito. Una vez que se cuenta con el estado biométrico, este se envía para ser visualizado o comunicado en texto o audio al Médico o al Paciente. El

estado biométrico se envía mediante el *Mensajero Productor* que se comunica con el microservicio *Mensajero*.

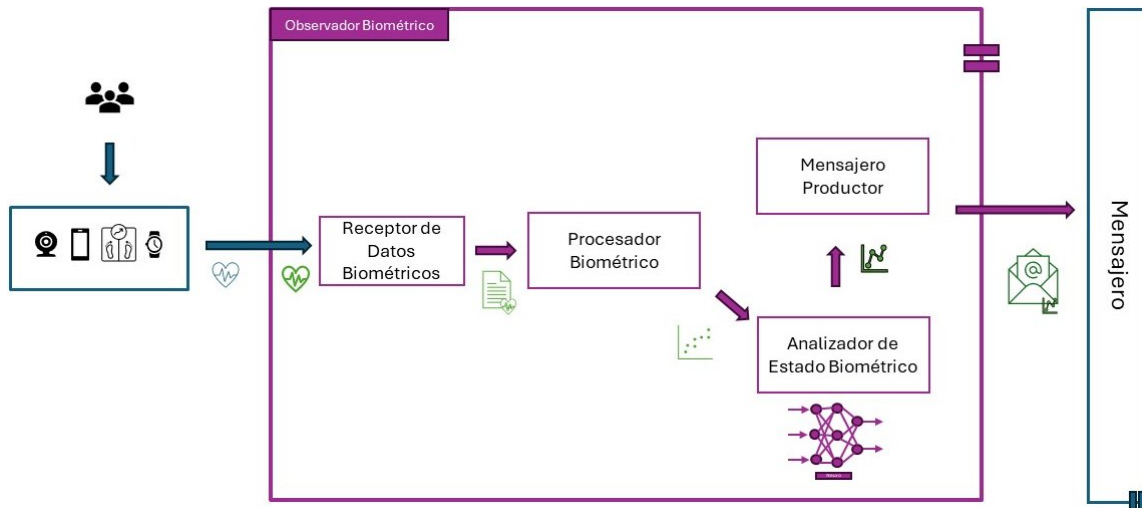


Figura 4.9: Microservicio *Observador Biométrico*

4.5.4 Compañero Digital

Este microservicio es el receptor de la información generada por los demás microservicios de CoDiSPa y cuenta con las reglas de negocio que definen el comportamiento del sistema CoDiSPa. El microservicio se puede describir como un orquestador que delega las tareas de acuerdo con las responsabilidades de cada componente del sistema CoDiSPa, incluido el propio microservicio Compañero Digital.

Responsabilidades

Este microservicio es el encargado de reunir y coordinar los datos, información y conocimiento para generar un modelo de retroalimentación, con el objetivo de adaptar para el uso del Paciente al compañero digital. También es el encargado de establecer comunicación con los demás microservicios de CoDiSPa para realizar los casos de uso para la gestión de citas, seguimiento médico y adherencia a la toma de medicamentos.

Interacciones

Este microservicio tiene interacción con los demás microservicios que conforman al sistema CoDiSPa a través del microservicio *Mensajero*.

Componentes del microservicio

Los componentes de este microservicio, como puede verse en la Figura 4.10, son: *Mensajero Consumidor*, *Mensajero Productor*, *Analizador de Mensajes*, *Productor de Mensajes*, *Agente Inteligente* y *Calendarizador de Tareas*.

Funcionamiento del microservicio

Este microservicio es el corazón que articula y orquesta a los demás microservicios que componen al sistema CoDiSPa, de tal manera que recibe las peticiones de los microservicios de la capa de aplicación, i.e., *Reconocedor del Habla* y *Observador Biométrico*. Adicionalmente el microservicio se comunica con el usuario, a través del microservicio *Locutor Proactivo*. El microservicio *Compañero Digital* procesa las peticiones dependiendo de su origen y la tarea que se esté solicitando procesar. Una vez que identifica la tarea a realizar, la delega a alguno de los microservicios de la capa de servicios, i.e., al *Buscador Semántico*, al *Analizador de Comportamiento* o al *Manejador de Conocimiento*. De acuerdo con la respuesta de los otros microservicios, se envía la respuesta al microservicio solicitante. Cuando se trata de registrar tareas que se deben atender en un futuro, se hace uso del *Calendarizador de Tareas* para lanzarlas en el horario que se establece en la solicitud. Este microservicio tiene en el microservicio *Calendarizador de Tareas*, una serie de tareas programadas que son generadas por él mismo, para realizar preguntas al usuario Paciente. Las preguntas son realizadas de manera proactiva. La interacción con el usuario es bajo el principio de sistema centrado en el usuario, además de tener la capacidad reactiva como cualquier sistema tradicional. Los parámetros que se adaptan al Paciente se gestionan en el componente *Agente Inteligente*. Estos parámetros se adaptan en el sistema CoDiSPa de acuerdo con la interacción con el Paciente.

4.5.5

Buscador Semántico

Este microservicio es capaz de recibir consultas y responderlas a partir de la *Base de Conocimiento* hecha para el sistema CoDiSPa, como se muestra en la Figura 4.11. En caso de que el microservicio no encuentre respuesta en la *Base de Conocimiento* se

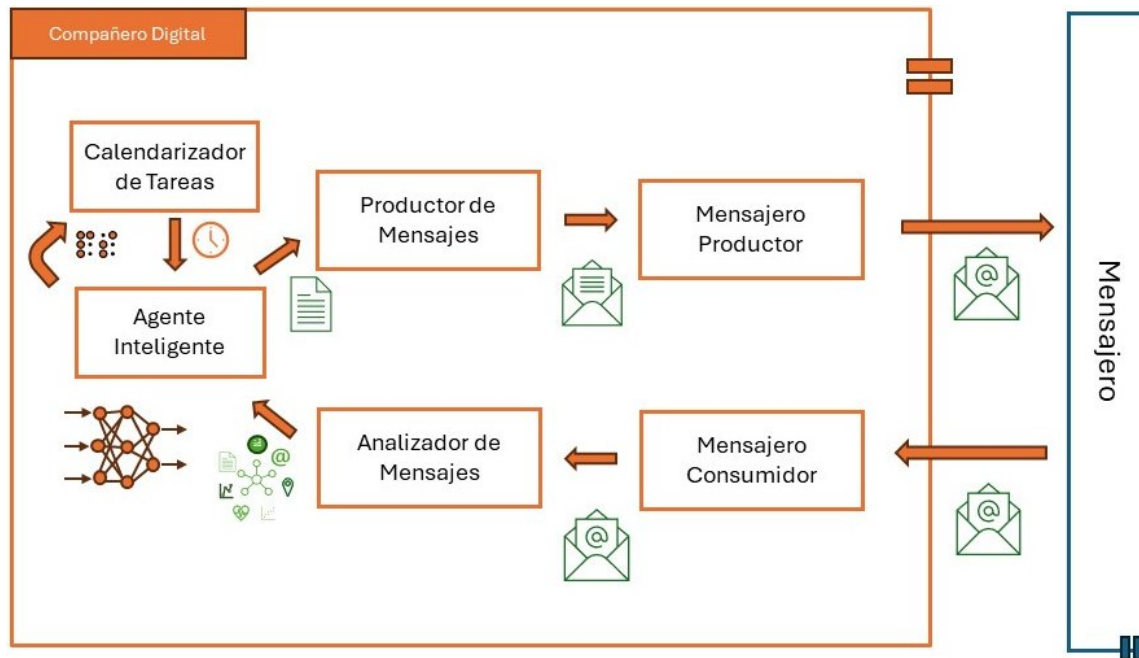


Figura 4.10: Microservicio *Compañero Digital*

realizará una búsqueda en la Web, de preferencia en el grafo de conocimiento, para responder y retroalimentar al usuario.

Responsabilidades

Este microservicio atiende solicitudes de búsqueda y da respuesta a ellas. En el caso de que no tenga una respuesta, la pregunta puede ser delegada a otros usuarios, como puede ser el Médico, el Farmacéutico o el Técnico de análisis clínico.

Interacciones

Las interacciones de este microservicio con los demás microservicios es indirecta y, de manera directa, con el microservicio *Mensajero*.

Componentes del microservicio

Los componentes de este microservicio, como puede verse en la Figura 4.12, son: *Mensajero Consumidor*, *Mensajero Productor*, *Analizador de Preguntas*, *Productor de*

Respuestas, Buscador Ontológico, Generador de Propuestas Educativas y Gestor de Conocimiento.

Funcionamiento del microservicio

El microservicio recibe una petición de búsqueda, por medio del *Mensajero Consumidor*. Se analiza la pregunta recibida, en el *Analizador de Preguntas* y se decide si se usa una ontología o se hace una búsqueda directa de la pregunta en la *Base de Conocimiento* del sistema CoDiSpa. Una vez obtenido el resultado de la consulta, se genera la respuesta correspondiente mediante el *Productor de Respuestas* y se entrega de regreso al microservicio *Mensajero*, usando el *Mensajero Productor*. Las preguntas pueden ser también solicitudes de propuestas educativas desde el microservicio *Compañero Digital* para realizar una tarea proactiva, que se le haga llegar al usuario, con información para la mejora de hábitos o de conocimiento de algún aspecto de salud de su padecimiento.

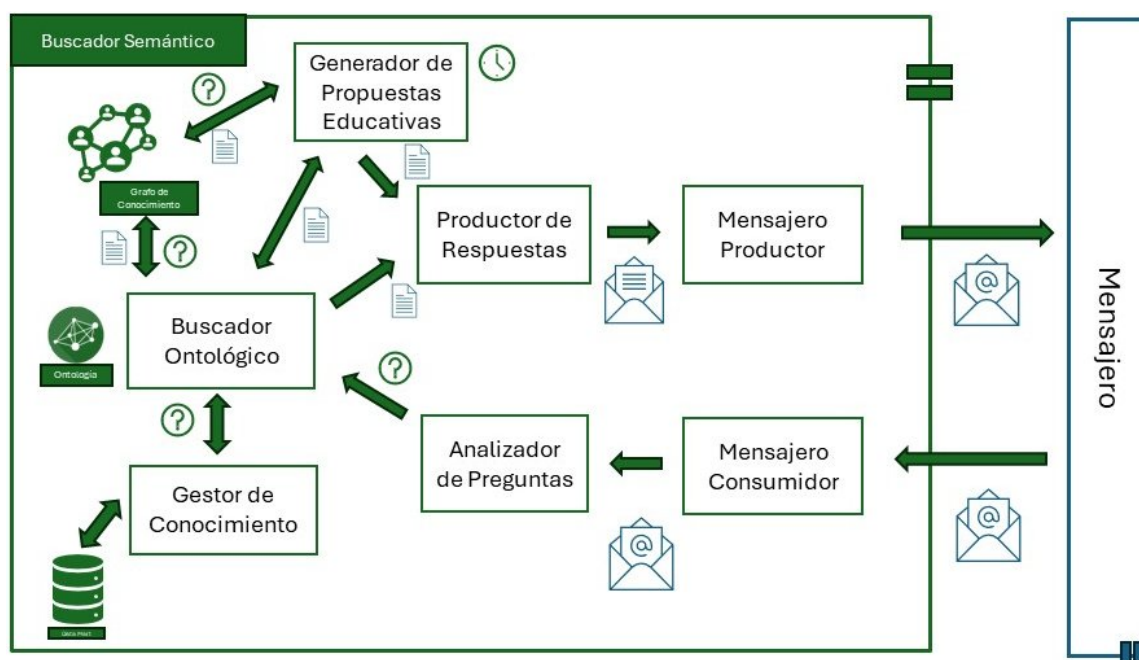


Figura 4.11: Microservicio *Buscador Semántico*

Como parte de sus responsabilidades, este microservicio debe alimentar la *Base de Conocimiento* con entradas a partir de patrones descubiertos por el microservicio *Analizador de Comportamiento*. Cuando se realiza una consulta a la *Base de Conocimiento* y la respuesta no se encuentra en ella, se cambia de fuente de información.

Las fuentes de información pueden ser *Base de Conocimiento* en Internet, el grafo de conocimiento u ontologías especializadas. El resultado de la consulta es insertada en la *Base de Conocimiento* del sistema CoDiSPa, como se muestra en la Figura 4.12.

Este microservicio interactúa con la *Base de Conocimiento*. La *Base de Conocimiento* almacena información contextualizada lista para ser consultada por cualquier otro microservicio. En particular, la *Base de Conocimiento* cuenta con temas específicos relacionados con la salud del Paciente con origen en el microservicio *Analizador de Comportamiento*.

El microservicio puede consultar dos principales fuentes: 1) algún servicio de grafo de conocimiento alojado en Internet o 2) la *Base de Conocimiento* generada por el sistema CoDiSPa.

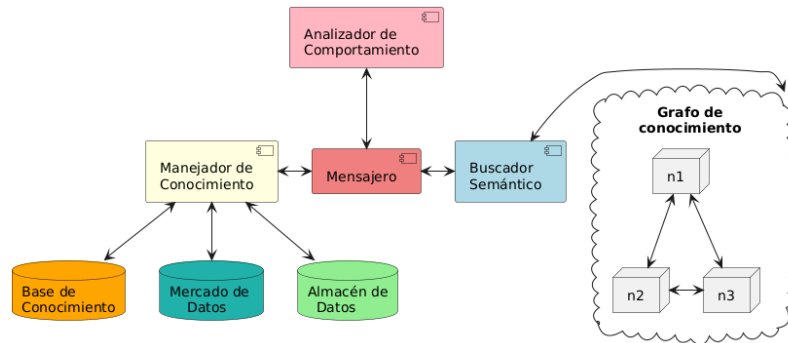


Figura 4.12: Interacción del microservicio *Buscador Semántico* con otros microservicios de CoDiSPa

4.5.6 Analizador de Comportamiento

Este microservicio recibe datos del microservicio *Observador Biométrico* y del microservicio *Compañero Digital* para realizar análisis de datos usando métodos de aprendizaje automático tales como agrupamiento (*clustering*), clasificación (*classification*) o detección de anomalías (*anomaly detection*) [79].

Responsabilidades

Este microservicio tiene la responsabilidad de sintetizar la información que se le proporciona y generar, a partir de ella, conocimiento, i.e., se trata de un razonador para generar conocimiento.

Interacciones

Este microservicio se comunica con los microservicios: *Compañero Digital*, *Observador Biométrico* y *Manejador de Conocimiento*.

Componentes del microservicio

Los componentes del microservicio, como puede observarse en la Figura 4.13, son: *Mensajero Consumidor*, *Mensajero Productor*, *Selector de Origen de Datos*, *Gestor de Modelos Inteligentes*, *Ejecutor de Modelos de Datos* y *Productor de Conocimiento*.

Funcionamiento del microservicio

Este microservicio recibe los datos que se generan, periódicamente, en los microservicios *Observador Biométrico* y *Compañero Digital*. Estos paquetes de datos, de acuerdo con su origen, se encuentran identificados para que en el selector se pueda delegar, el procesamiento, al modelo más adecuado, de acuerdo con el tipo de conjunto de datos. Una vez que se seleccionó el modelo a usar, este es ejecutado por el *Ejecutor de Modelos de Datos* en un proceso asíncrono. Cuando el proceso asíncrono termina, el resultado es enviado al *Productor de Conocimiento* para que sea formateado adecuadamente y se almacene en la *Base de Conocimiento*. También, se envía una respuesta al microservicio *Compañero Digital* para que lleve el control de los procesamientos realizados. Los mensajes hacia los microservicios del sistema CoDiSPa son enviados a través del *Mensajero Productor*.

En este microservicio se tiene la posibilidad de acelerar el entrenamiento de los modelos de aprendizaje automático, sobretodo de la ejecución de modelos de *Clustering* y *Classification*, usando los dispositivos de *Edge AI* o una tarjeta *GPU* con soporte para *CUDA*. El resultado del análisis es enviado al microservicio *Buscador Semántico* mediante el microservicio *Mensajero*. El resultado también es enviado al microservicio *Compañero Digital*.

4.5.7

Manejador de Conocimiento

Este servicio es el encargado de la gestión del conocimiento. Cuenta con una *Base de Conocimiento* que almacena el conocimiento para su explotación, posterior, en el sistema CoDiSPa.

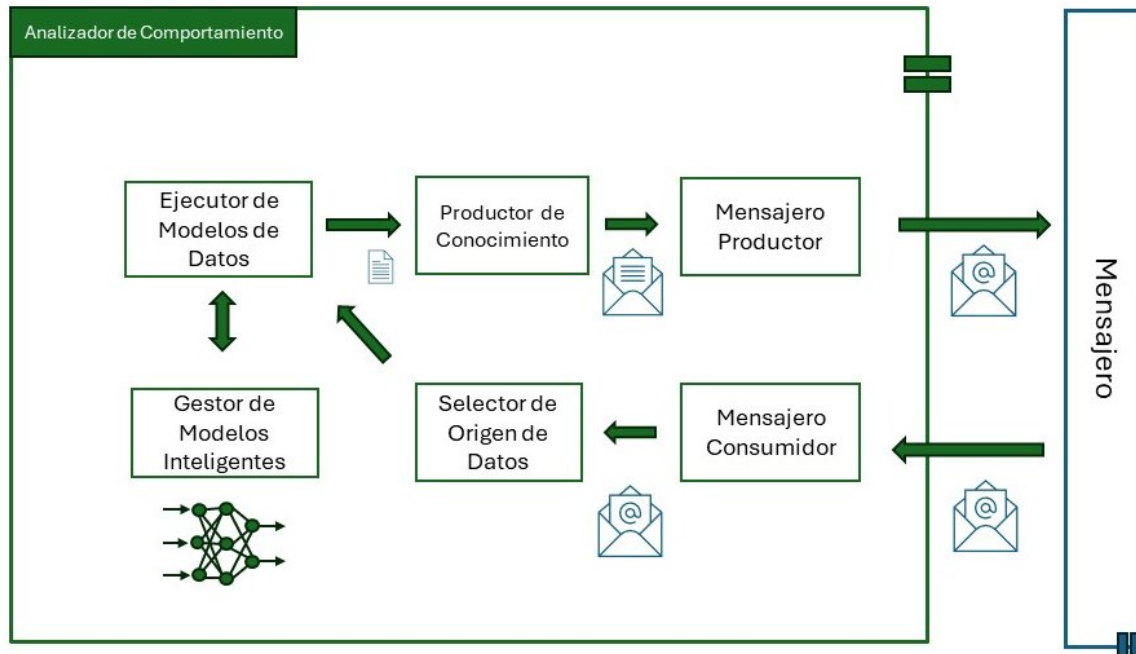


Figura 4.13: Microservicio *Analizador de Comportamiento*

Responsabilidades

La responsabilidad del microservicio es almacenar, de manera estructurada, el conocimiento y tenerlo disponible usando indización. La estructura básica del conocimiento es con la formación de tuplas pregunta-tema-conocimiento en un grafo.

Interacciones

La interacción de este microservicio con los demás microservicios de CoDiSPa es de manera indirecta, a través del microservicio *Mensajero*.

Componentes del microservicio

Los componentes del microservicio, como se muestra en la Figura 4.14, son: *Mensajero Consumidor*, *Mensajero Productor*, *Analizador de Tuplas*, *Productor de Conocimiento*, *Gestor de la Base de Conocimiento*.

Funcionamiento del microservicio

A partir de la recepción de un mensaje en el *Mensajero Consumidor* se analiza la petición en el *Analizador de Tuplas* y se identifica la operación a realizar. Las operaciones a realizar pueden ser: 1) registro de nuevo conocimiento o 2) recuperación de conocimiento. El *Gestor de la Base de Conocimiento* consulta en las fuentes disponibles: *grafo de conocimiento*, *almacén de datos*, *mercado de datos* o *base de datos*. Usando las respuestas obtenidas por el *Gestor de la Base de Conocimiento* se envían al *Productor de Conocimiento*. El *Productor de Conocimiento* selecciona o sintetiza las respuestas para generar una respuesta única y la envía al *Mensajero Productor*. Por último, el *Mensajero Productor* genera el mensaje que será enviado al *Mensajero* del sistema CoDiSPa.

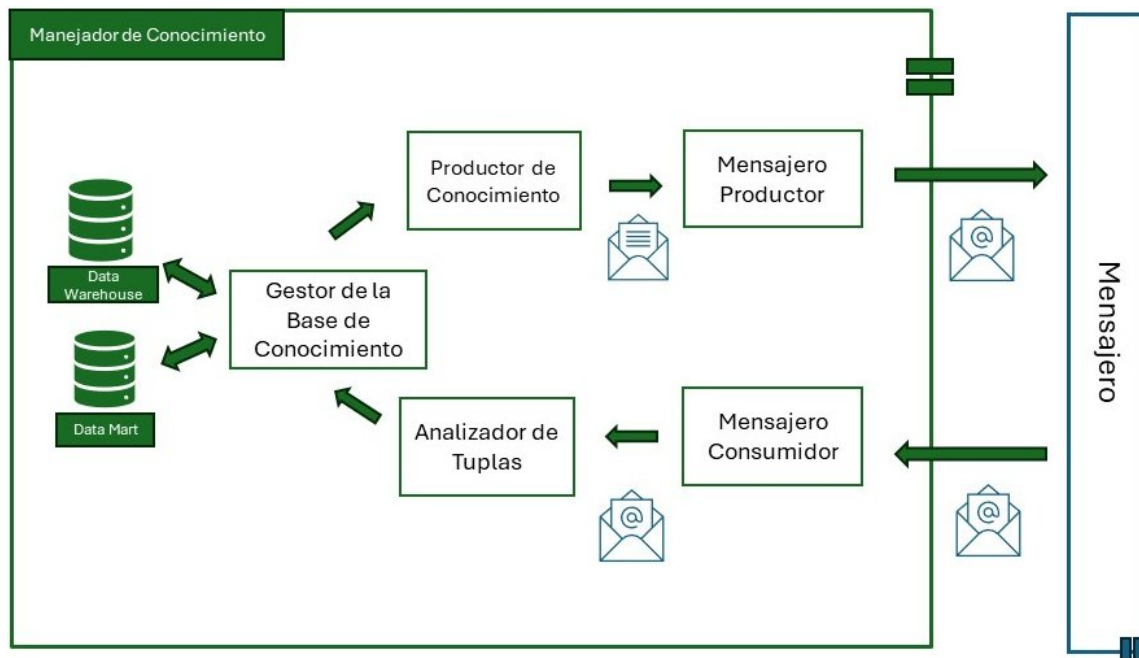


Figura 4.14: Microservicio *Manejador de Conocimiento*

4.5.8

Mensajero

Este microservicio es el encargado de enrutar los eventos, a través de un sistema de colas de peticiones. El microservicio gestiona los mensajes para facilitar el intercambio

entre los diversos microservicios usando la técnica de ensobretado del patrón de integración *Message Queue* [80]. Este microservicio está basado en el patrón *Broker* [78], como se muestra en la Figura 4.15.

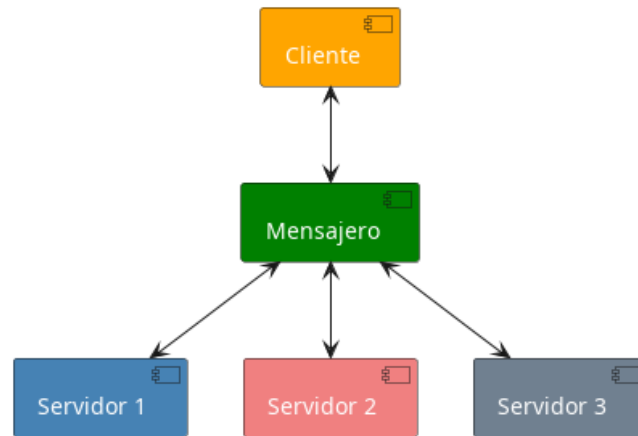


Figura 4.15: Patrón de arquitectura *Broker*

Responsabilidades

El microservicio *Mensajero* es el responsable de coordinar la comunicación, despachando a otros microservicios o redirigiendo las peticiones, así como transmitiendo los resultados de dichas peticiones.

El microservicio tiene la responsabilidad de gestionar los mensajes de manera concurrente y llevar un seguimiento de la comunicación.

Interacciones

El microservicio tiene interacción con todos los microservicios de CoDiSPa debido a que es el medio de comunicación entre los microservicios en las distintas capas.

Componentes del microservicio

Los componentes que tiene el microservicio, como puede verse en la Figura 4.16, son: *Mensajero Consumidor*, *Mensajero Productor*, *Enrutador de Mensajes*, *Analizador de Mensajes* y *Transformador de Mensajes*.

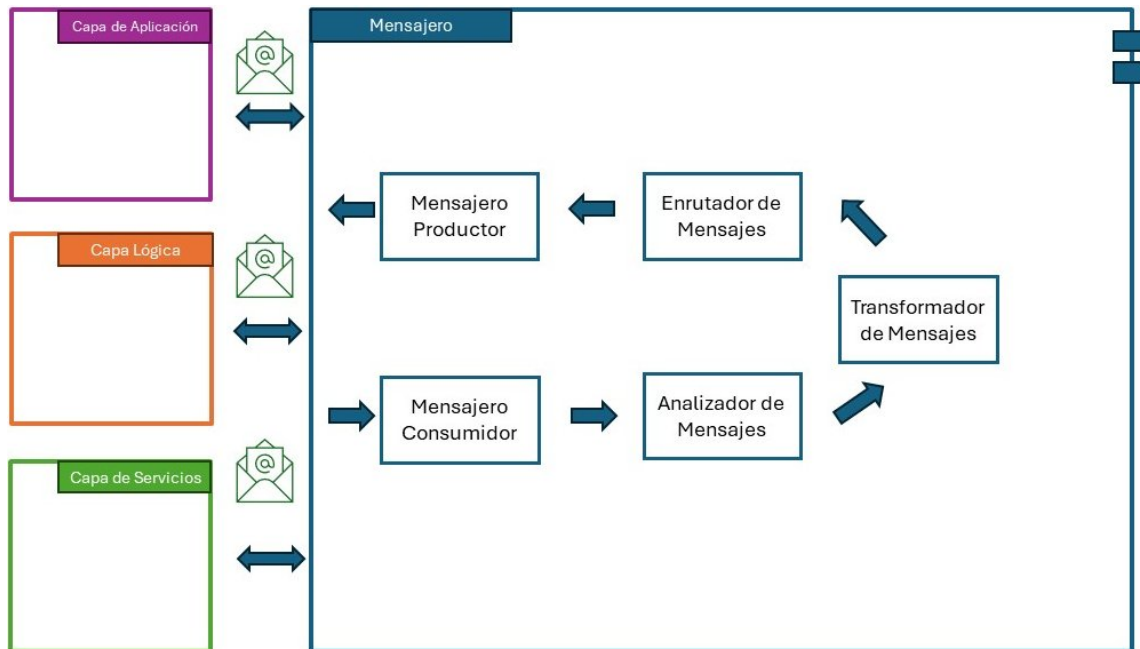


Figura 4.16: Microservicio *Mensajero*

Funcionamiento del microservicio

El microservicio recibe mensajes y los organiza en colas de mensajes para cada microservicio. Gracias a ello, puede gestionar los mensajes y atender asincrónicamente a la velocidad que sea capaz cada microservicio de CoDiSPa.

4.5.9 Adaptabilidad y recomendaciones

Para que el sistema CoDiSPa pueda proporcionar recomendaciones que ayuden al usuario, es necesario contar con una *Base de Conocimiento*. La *Base de Conocimiento* es un almacén organizado de información contextualizada que cuenta con un sistema de gestión que se alimenta desde dos vías, la primera, a partir de bases de conocimiento expuestas vía *Web* desde Internet, y la segunda, es producto del razonamiento de los servicios planteados para el sistema CoDiSPa.

La gestión del conocimiento es el proceso mediante el cual se recopila y mantiene información producida dentro de una organización. La información se mantiene y se hace accesible a las partes relevantes mediante un microservicio disponible en todo momento. La gestión incluye, también, mantener la *Base de Conocimiento* actualizada

y asegurar su integridad. Asimismo, se debe tener acceso al conocimiento y hacerlo asequible para el personal de administración de la base llegando hasta los Pacientes. Una tarea imprescindible es la recopilación de la información relevante y verificación de su exactitud y veracidad. Para contar con un sistema capaz de adaptarse al usuario y generar empatía en él, se requiere hacer que el comportamiento del sistema se reconfigure a partir de la *Base de Conocimiento*, en particular, con lo aprendido del comportamiento del usuario. Todo el conocimiento recopilado continuamente se agrega al sistema CoDiSPa con un estado por verificar para que posteriormente un experto en el área ayude con su dominio sobre los temas. Es necesario que la *Base de Conocimiento* se valide con los especialistas y, mediante el microservicio *compañero digital*, se le hagan llegar la dupla pregunta-respuesta para ser validada o descartada por dicho especialista, o realizar las preguntas pertinentes para clasificar el conocimiento en la *Base de Conocimiento*. Se tiene con ello la propuesta de una taxonomía *ad hoc* con nuestros objetivos como se muestra en la Figura 4.17. Es de suma importancia que la base de conocimiento sea actualizada y depurada para que vaya madurando el conocimiento y esté sea útil. Se usan técnicas de clasificación y agrupamiento de aprendizaje automático. También se conforma un *grafo de conocimiento* que es empleando para la validación del especialista y del Paciente también.

not_validated	☐ false
specialist_validated	☒ true
patient_validated	☒ true
institution_validated	☒ true

Figura 4.17: Ejemplo que emplea la taxonomía del conocimiento empleada para el CoDiSPa

La estructura del conocimiento relacionada con el Paciente tiene la información siguiente:

- Perfil del Paciente
- Estado de seguimiento médico
- Estado biométrico
- Estado emocional
- Problema con solución o recomendación
- Conocimiento encontrado para responderse al Paciente

Las entradas de conocimiento que han sido empleadas, validadas y que han tenido éxito resolviendo el problema o duda bajo la que fueron consultadas son ligadas en el grafo de conocimiento conformado en la *Base de Conocimiento*. El sistema CoDiSPa esta dotado, también, de parámetros de comportamiento, i.e, cuenta con parámetros específicos para cada Paciente que cambian de acuerdo con su comportamiento, e.g., si un Paciente prefiere que no se le den recomendaciones sino hasta que él pregunte o solicite, el orden de operaciones del *compañero digital* registrará cual es la preferencia del Paciente. Si por el contrario al Paciente le gusta que el sistema CoDiSPa lo aborde para conversar y compartirle recomendaciones esto quedará registrado en sus parámetros de configuración. Los parámetros para cada usuario irán cambiando periódicamente haciendo con ello que el sistema se adapte cada vez más a la personalidad del Paciente.

A partir de los eventos que el sistema CoDiSPa le pregunta al Paciente cómo se siente emocionalmente y registrar su respuesta junto con los parámetros biométricos disponibles para que esto posibilite realizar procesos de análisis e identificar patrones de comportamiento que acerca al sistema a un funcionamiento más asertivo y que produce empatía en el Paciente. Entre más se acierte en cada interacción con el Paciente, se gana la confianza del Paciente hacia el sistema CoDiSPa y se aproxima al éxito de la asistencia.

4.6 Vista lógica

El sistema CoDiSPa articula los distintos componentes, en forma de microservicios, para atender los requerimientos y satisfacerlos. Atendiendo también los requerimientos no funcionales, gracias a la selección de las tecnologías apropiadas. A continuación se describe la interacción entre los distintos microservicios para atender los casos de uso del sistema CoDiSPa.

4.6.1 Interacciones para la gestión de citas médicas

Los actores que participan en el caso de uso, para la gestión de los citas médicas, son el Paciente y el *compañero digital*. En la Figura 4.18, se muestra un diagrama basado en un diagrama de secuencia; sin embargo, el símbolo de ejemplar de clase se debe entender como un componente de CoDiSPa y el actor Compañero Digital es a su vez actor y componente. En la Figura 4.18 se puede observar que el *compañero digital* comienza con el flujo de acciones en el sistema preguntando al Paciente, mediante el microservicio *Locutor Proactivo*, si el Paciente quiere agendar una cita en algún servicio médico. Si el Paciente responde afirmativamente, se analiza la respuesta para identificar el tiempo y lugar para agendar la cita médica. Se realiza el registro de la

cita comunicándose con el sistema o persona del servicio médico y se almacena la cita en el sistema del *Compañero Digital*. Se notifica verbalmente al Paciente, a través del microservicio *Locutor Proactivo*, que se ha agendado correctamente la cita solicitada. Asíncronamente, se recordará al Paciente cuando la cita esté próxima, mediante un mensaje de audio que se enviará al Paciente, por medio del microservicio *Locutor Proactivo*.

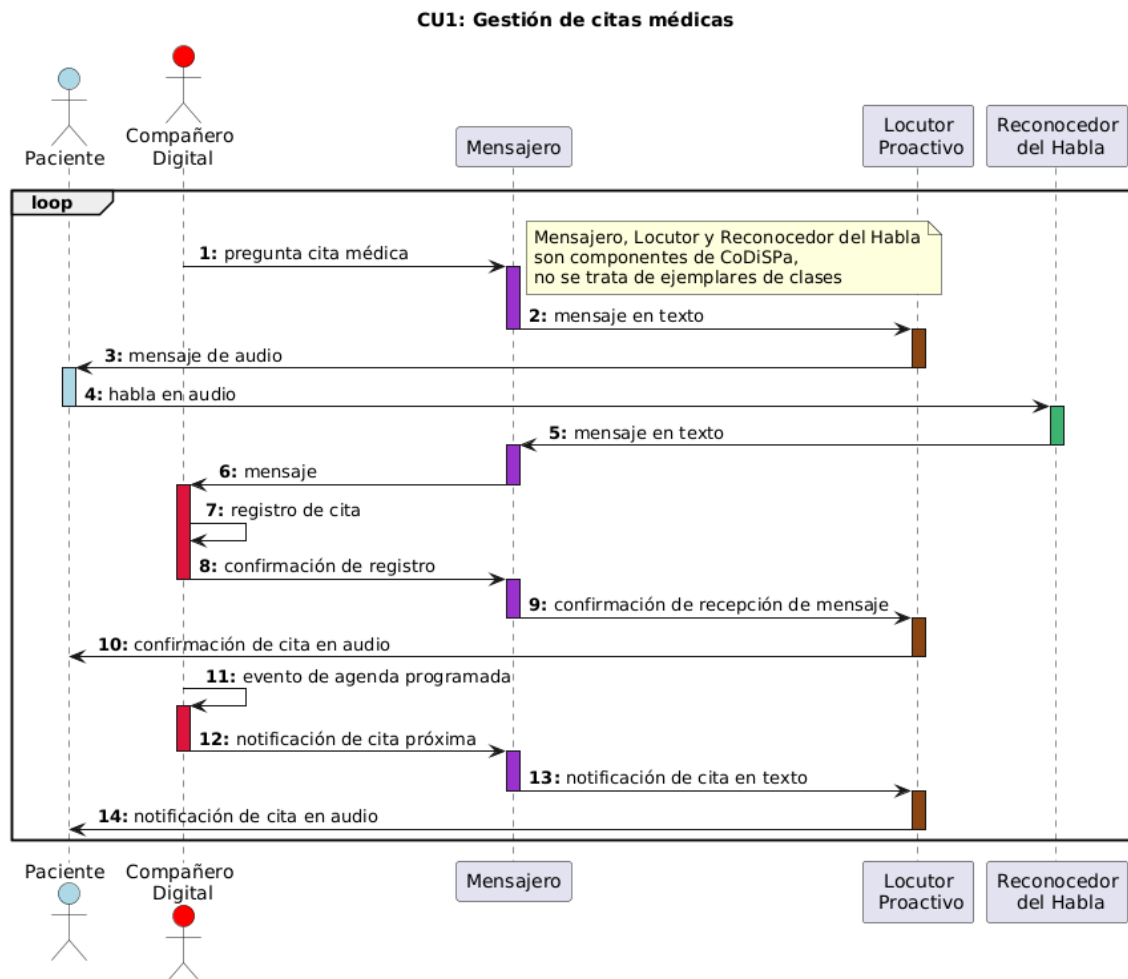


Figura 4.18: Diagrama de secuencia para la gestión de citas médicas

Es posible realizar el registro de una cita solicitada por alguno de los servicios médicos. Esta acción es realizada de manera convencional, i.e., si alguno de los servicios

médicos lo solicita el *compañero digital* agendará y le notificará al Paciente la cita con el servicio médico correspondiente.

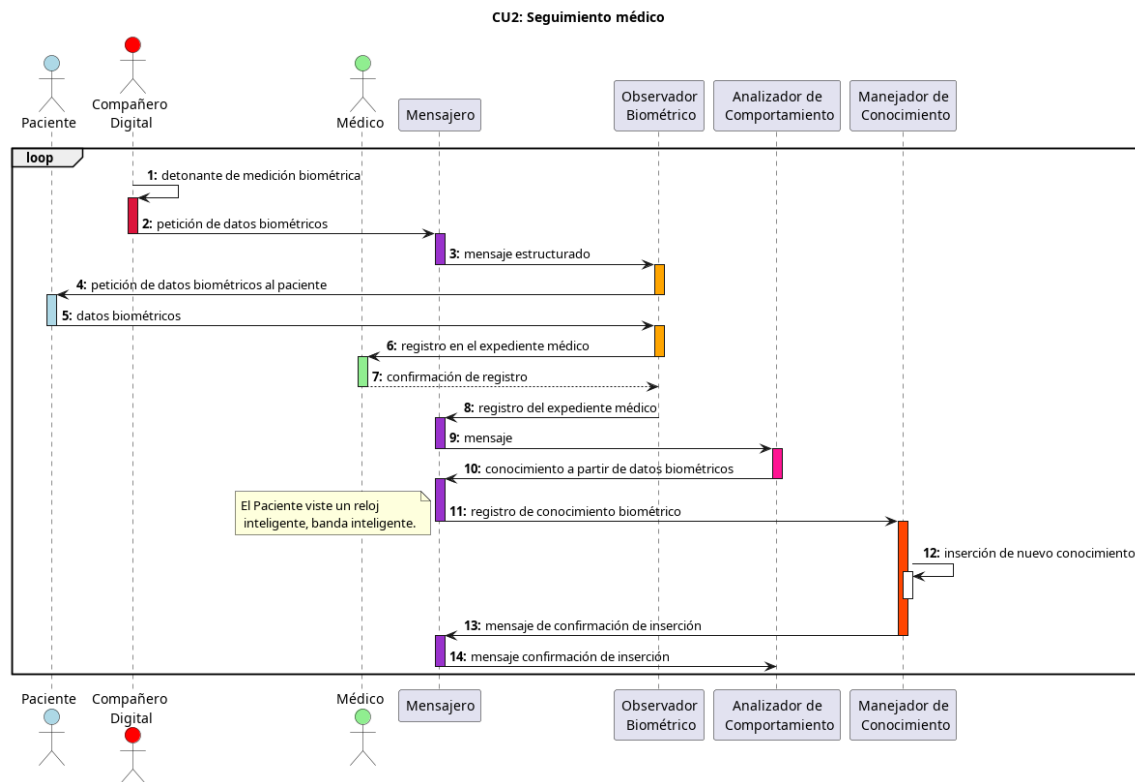


Figura 4.19: Diagrama de secuencia para el seguimiento médico

4.6.2 Interacciones para el seguimiento médico

El caso de uso para el seguimiento médico, como se muestra en Figura 4.19, muestra que el actor *Compañero Digital* inicia el flujo haciendo una petición al microservicio *Observador Biométrico* para obtener los datos recolectados por los diversos dispositivos en contacto directo con el Paciente. En la Figura 4.19, se muestra un diagrama basado en un diagrama de secuencia; sin embargo, el símbolo de ejemplar de clase se debe entender como un componente de CoDiSPa y el actor *Compañero Digital* es a su vez actor y componente. Una vez que el *Compañero Digital* obtiene los datos biométricos estos son organizados y procesados por el microservicio *Observador Biométrico* para ser entregados por dos vías, la primera es al Médico; la segunda, es al microservicio

Analizador de Comportamiento para su inserción y procesamiento. Una vez que se tiene un nuevo elemento en el microservicio *Analizador de Comportamiento* este es procesado para generar conocimiento a partir del conjunto de elementos disponibles, si se genera nuevo conocimiento de este proceso se insertará enviándolo al microservicio *Manejador de Conocimiento*.

El sistema CoDiSPa, basándose en la lectura de los parámetros biométricos del Paciente, es capaz de encontrar un momento ideal para conversar con él y hacerle saber información conductual y de tratamiento para ayudar a la comprensión de su condición médica, y así, fomentar un cambio en algunos hábitos (adherencia a toma de medicamentos tal como la alimentación, ciclos de descanso, sesiones de ejercicio, entre otros. También es capaz de ayudar al Paciente en la búsqueda de información o resolución de preguntas que el Paciente le plantee. Esta resolución será en varias vías, priorizando la *Base de Conocimiento* que el sistema CoDiSPa posee y después delegando las dudas al Médico que sean capaces de contestar. La dupla pregunta respuesta será almacenada en la *Base de Conocimiento* de manera estructurada.

4.6.3

Interacciones para mejorar la adherencia a la toma de medicamentos

El caso de uso para mejorar la adherencia a la medicación del Paciente comienza con el registro de la prescripción de los medicamentos ya sea por parte del Médico o por parte del Paciente, como se muestra en la Figura 4.20. En la Figura 4.20, se muestra un diagrama basado en un diagrama de secuencia; sin embargo, el símbolo de ejemplar de clase se debe entender como un componente de CoDiSPa y el actor Compañero Digital es a su vez actor y componente.

Una vez que se registró la prescripción médica en el *Compañero Digital* se realizan todas las inserciones de recordatorios para cumplir con la toma de los medicamentos, asimismo, se almacenan los detalles de la aplicación de los medicamentos como fueron prescritos. La responsabilidad de los recordatorios es para el microservicio *Compañero Digital* que realizará las notificaciones a los diversos dispositivos disponibles para avisar al Paciente. Por defecto se avisará al Paciente mediante el microservicio *Locutor Proactivo*. El microservicio *Compañero Digital* hará la revisión periódica de los eventos almacenados para notificar todas y cada una de las prescripciones que el Paciente debe tomar.

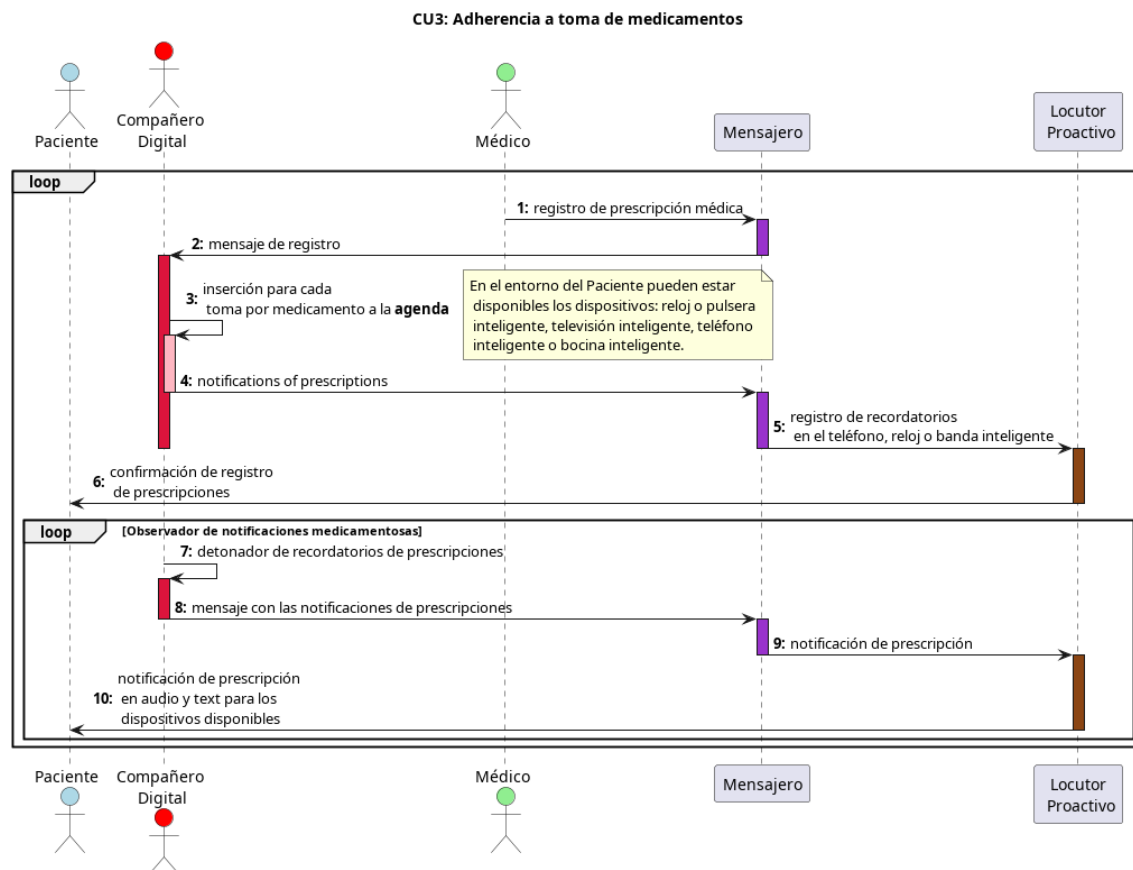


Figura 4.20: Diagrama de secuencia para mejorar la adherencia a la toma de medicamentos

Implementación del Reconocedor del Habla Multifuente

Contenido

5.1	Análisis de la experiencia anticipada del usuario	72
5.2	Módulos de Reconocedor del Habla Multifuente	77
5.3	Identificación de usuarios	79
5.4	Procesamiento de lenguaje natural	80
5.5	Comunicación entre módulos	81
5.6	Desarrollo del microservicio	82

Primeramente, en la Sección 5.1, se describen las pruebas y los resultados de un estudio de experiencia de usuario anticipada, que permitió sentar las bases para la implementación. Enseguida, en la Sección 5.2, se describen los módulos que conforman al Reconocedor del Habla Multifuente. En la Sección 5.3 se describe el módulo de identificación de usuarios en el sistema. En la Sección 5.4 se describe el módulo de procesamiento de lenguaje natural que traduce audio a texto para el sistema propuesto. Por último, en la Sección 5.5 se describe el módulo de comunicación con los mecanismos que el sistema propuesto emplea para interoperar entre los módulos del sistema propuesto.

En esta sección, se describen y comentan los resultados de las pruebas de experiencia de usuario anticipada realizadas con personas que padecen diabetes, quienes realizaron un conjunto de tareas derivadas de los casos de uso del sistema CoDiSPa.

La técnica del Mago de Oz (WoZ, siglas en inglés para *Wizard of Oz*) , que toma su nombre de la novela homónima, es un método seminal en el diseño de sistemas interactivos, especialmente útil en las primeras fases de desarrollo. Consiste en que un operador humano controle clandestinamente un sistema mientras los usuarios interactúan con él como si fuera totalmente autónomo [30]. Esta técnica tiene un valor incalculable para escenarios como el del presente capítulo, donde es crucial comprender la interacción del usuario en contextos del mundo real. Facilita la recopilación de datos para el entrenamiento de la inteligencia artificial, permite observar cómo articulan los usuarios sus peticiones y sus expectativas de respuesta y ayuda a identificar posibles escenarios de interacción para futuros desarrollos [81].

Las pruebas se realizaron en un entorno controlado en el que los participantes interactuaban con un compañero digital, creyendo que era totalmente autónomo. Se pidió a los usuarios que hablaran a un micrófono y recibieran las respuestas del compañero digital a través de altavoces. Sin que los participantes lo supieran, estas respuestas las elaboraba un humano mediante un generador de texto a voz. Esta configuración permitió observar y grabar interacciones, reacciones y comentarios auténticos de los usuarios en tiempo real.

Las tareas derivadas de los casos de uso propuestos fueron evaluadas por una cohorte de 20 individuos con diabetes. Este grupo de participantes estuvo formado por colegas, amigos y familiares, lo que representa una muestra de oportunidad. La distribución demográfica incluía 11 mujeres y nueve hombres, con edades comprendidas entre los 38 y los 81 años con una media de 55.15 años.

Antes de iniciar las sesiones de prueba, se les explicó a los participantes la finalidad del compañero digital y la idea general que subyace en su apoyo a los pacientes diabéticos. Se les explicaron las tres tareas que debían realizar: concertar citas médicas, introducir datos de salud y pedir consejos dietéticos. Sin embargo, para garantizar la autenticidad de sus interacciones y obtener información genuina, deliberadamente no se les ofrecieron instrucciones detalladas sobre la realización de estas tareas. Este enfoque se adoptó para observar la forma natural en que los usuarios interactuaban con el sistema, incluida la manera en que formulaban sus peticiones y consultas. Así

podimos captar una serie de comportamientos y expresiones verbales espontáneas de los usuarios, fundamentales para crear un compañero digital intuitivo y centrado en el usuario.

Durante las pruebas, a los participantes se les encomendaron actividades diseñadas para reflejar casos de uso típicos de pacientes diabéticos, no para evaluar la funcionalidad del sistema, sino para obtener información de cara a la implementación del compañero digital. Por ejemplo, al reservar una cita con el médico, se observaron los patrones de lenguaje de los participantes y cómo especificaban los detalles de la cita, como el tipo, la fecha y la hora. Estas observaciones ayudaron a comprender la necesidad de que el sistema solicitara la información que faltaba e interpretara frases vagas como “por la mañana” o “por la tarde”.

En las tareas que implicaban la introducción de datos sanitarios, como las lecturas de glucosa en sangre y presión arterial, este estudio se centró en cómo los participantes comunicaban estas mediciones. La tendencia a omitir las unidades explícitas puso de manifiesto la importancia de tener en cuenta el contexto en el diseño del sistema.

Por último, a la hora de pedir consejo dietético, se prestó mucha atención al tipo de lenguaje y a los detalles que preferían los usuarios. Normalmente eran directos y esperaban el mismo tono del compañero digital.

Tras las sesiones de interacción, se pidió a los participantes que rellenaran el cuestionario AttrakDiff¹. Así se obtuvieron datos cuantitativos sobre su percepción de la estética, la funcionalidad y la experiencia general del usuario. La combinación de la información cualitativa obtenida con la técnica WoZ y los datos cuantitativos del cuestionario AttrakDiff permitieron comprender mejor los puntos fuertes y los aspectos mejorables de las interacciones.

5.1.2

Resultados

Se utilizó la herramienta AttrakDiff porque es un cuestionario ampliamente reconocido en la investigación sobre experiencia de usuario y usabilidad, especialmente en el diseño de interfaces de usuario y productos digitales. El cuestionario AttrakDiff consta de 28 pares semánticos, i.e., pares de palabras que presentan un fuerte contraste entre sí (e.g., bueno-malo), cuyas respuestas se miden en una escala Likert que va de 1 a 7 [82]. A través de estos pares semánticos, el cuestionario mide los siguientes aspectos [82]:

- **Calidad Pragmática:** se refiere a la calidad percibida de manipulación, i.e., la efectividad y eficiencia de uso.

¹<https://www.attrakdiff.de/index-en.html>

- **Calidad Hedónica - Identidad:** indica la identificación del usuario con el producto.
- **Calidad Hedónica - Estimulación:** significa la necesidad humana de desarrollo individual, i.e., la mejora de conocimientos y habilidades.
- **Atractivo:** reporta el valor general de un producto basado en la calidad percibida.

Las dimensiones hedónicas y pragmáticas son autónomas entre sí y contribuyen equitativamente a la evaluación de la experiencia del usuario [82].

La Figura 5.1 muestra una visión general de la evaluación de las cuatro dimensiones principales de AttrakDiff. Es notorio que las calidades Pragmática y de Atractivo alcanzaron las evaluaciones más altas, con puntuaciones de 5.4 y 5.6 respectivamente. Esto sugiere que las técnicas de interacción implementadas en el prototipo de WoZ fueron efectivamente recibidas por los participantes. Dichas técnicas están alineadas con las expectativas de un asistente virtual convencional, logrando no sólo cumplir con los objetivos de los usuarios, sino que también resultan atractivas, lo cual es esencial para mantener el interés de los mismos. Por otro lado, las calidades de Identidad y Estimulación obtuvieron puntuaciones ligeramente inferiores, de 4.9 y 4.6 respectivamente. Este resultado indica que, en su versión final, el compañero digital debería fomentar una mayor empatía con los usuarios, posiblemente mediante un sintetizador de voz que resulte más natural y agradable, así como un estilo de respuesta que sea claro, confiable y útil. Estas respuestas deben ser prácticas y motivadoras, sin condescendencia ni aprehensión, incentivando a los usuarios a mantener un estilo de vida saludable.

La Tabla 5.1 es la contraparte numérica de la gráfica presentada en la Figura 5.1. Además de los promedios ya analizados (μ), lo que resalta aquí es la desviación estándar de los resultados (σ). Por una parte, para las calidades Pragmática y de Atractivo son relativamente bajas y sugieren una percepción más uniforme entre los usuarios. Por otra parte, las desviaciones para Identidad y especialmente Estimulación son más altas, lo que indica una mayor dispersión en las opiniones de los usuarios. Estos valores no son extremadamente altos, pero sí sugieren áreas donde las experiencias y opiniones de los usuarios varían más, lo cual podría ser un punto focal para la implementación del compañero digital.

La evaluación de la consistencia interna de los pares semánticos utilizando el coeficiente de alfa de Cronbach, que resultó ser $\alpha = 0.74$, proporciona una confirmación adicional de la fiabilidad de la escala AttrakDiff utilizada en este estudio. Un valor de α superior a 0.7 es indicativo de una buena fiabilidad, sugiriendo que los pares del cuestionario están midiendo de manera efectiva y coherente las cuatro dimensiones. Esta consistencia interna robusta es crucial para asegurar que las inferencias hechas

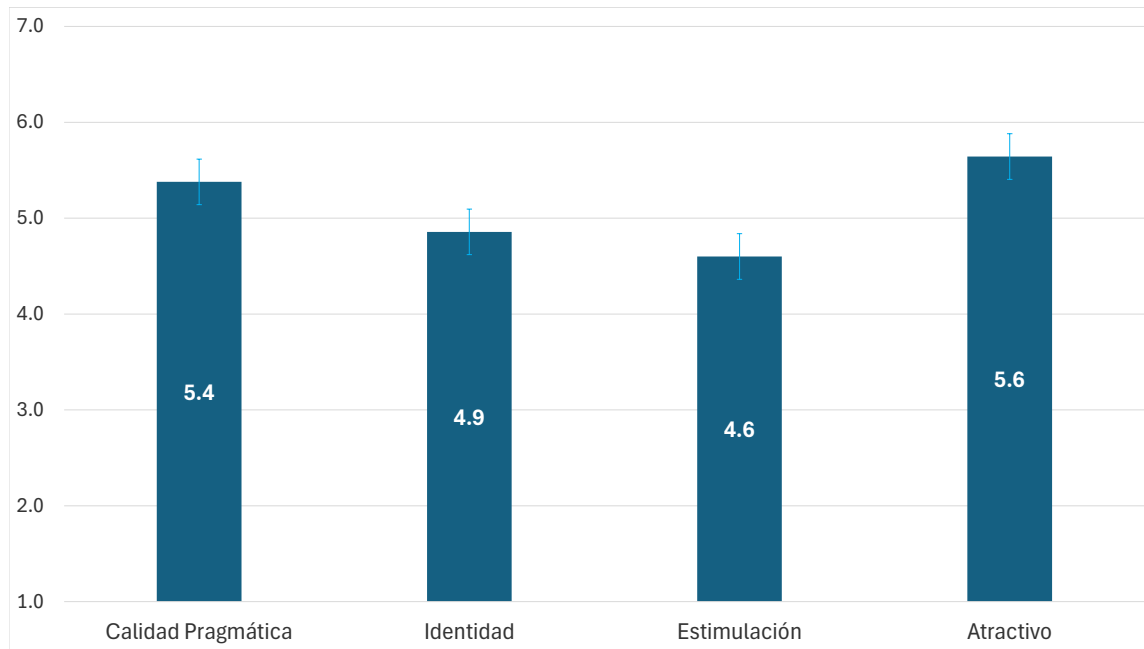


Figura 5.1: Los resultados muestran que el prototipo fue bien recibido en términos de su utilidad práctica y su atractivo general

Dimensiones de AttrakDiff	μ	σ
Calidad Pragmática	5.4	1.5
Identidad	4.9	1.7
Estimulación	4.6	2.0
Atractivo	5.6	1.4

Tabla 5.1: Medias (μ) y desviaciones estándar de los resultados (σ).

sobre las preferencias y percepciones de los usuarios sean basadas en una medición confiable.

5.1.3 Implicaciones para la implementación

La aplicación de la técnica WoZ, en esta fase de pruebas, ofreció información significativa sobre las interacciones y expectativas de los usuarios, que son fundamentales para la implementación del compañero digital enfocado a pacientes diabéticos. Los matices

observados en la forma en que los participantes articulaban sus peticiones y el tipo de información que omitían, durante la realización de las tareas, subrayaron la necesidad de un sistema consciente del contexto capaz de manejar datos incompletos e interpretar el lenguaje natural. Por ejemplo, la frecuente omisión de unidades explícitas en las tareas de introducción de datos sanitarios pone de relieve la necesidad de que el sistema infiera esos detalles con precisión. Este ejercicio fue inestimable para implementar un sistema que comprenda y se anticipe a las necesidades del usuario.

Otra observación fundamental fue la preferencia de los usuarios a la hora de recuperar datos sanitarios. Las preguntas sobre la duración de la respuesta de audio y el manejo de varias lecturas, a lo largo del día, fueron fundamentales para comprender las preferencias de los usuarios, en cuanto a recuperación de datos e interacción. Este aspecto es crucial para garantizar que el compañero digital proporcione información de forma accesible y fácil de usar. Los resultados obtenidos sugieren que los usuarios prefieren respuestas concisas y directas, lo que indica la necesidad de un sistema que proporcione información de forma eficiente sin abrumar al usuario.

La tarea de asesoramiento dietético reveló que, en general, los usuarios encontraban confusa la compleja jerga médica y preferían consejos más sencillos y directos. Esto supone una gran oportunidad para que el compañero digital salve la distancia entre la información médica y la comprensión del usuario. El sistema puede mejorar la comprensión y el cumplimiento de las directrices médicas por parte de los usuarios, ofreciéndoles consejos en términos sencillos.

Además, esta fase de pruebas sirvió como ejercicio de Experiencia de Usuario Anticipada [83]. Permitió calibrar las expectativas e ideas preconcebidas de los usuarios sobre los compañeros digitales, formadas por sus experiencias con asistentes virtuales comerciales como *Alexa* y *Siri*. Este conocimiento es crucial para adaptar el sistema propuesto a las expectativas de los usuarios, sobre todo teniendo en cuenta la naturaleza especializada de la asistencia sanitaria en comparación con los compañeros digitales de uso general.

Una observación notable durante las pruebas fue el retraso entre las peticiones de los usuarios y las respuestas del sistema. A pesar de que el operador humano era capaz de entender las peticiones de los usuarios mejor que cualquier sistema de inteligencia artificial actual, el tiempo de respuesta era notablemente más lento de lo que cabría esperar de un sistema totalmente automatizado. Esta discrepancia pone de relieve la importancia de la velocidad de respuesta en la satisfacción del usuario y la eficacia general de un compañero digital. A medida que se avance en la implementación de un sistema autónomo, la optimización del tiempo de respuesta será un área de atención crucial para igualar, o incluso superar, las expectativas de los usuarios fijadas por sus experiencias con los asistentes virtuales existentes. La información obtenida en las

pruebas WoZ es fundamental para dar forma a la implementación de una arquitectura de compañero digital que no sólo sea técnicamente competente, sino que también se ajuste a las preferencias y expectativas de los usuarios. Estos resultados nos guiaron en la creación de un sistema centrado en el usuario, intuitivo y eficaz para ayudar a los pacientes diabéticos en su gestión sanitaria diaria.

5.2 Módulos de Reconocedor del Habla Multifuente

Para implementar el Reconocedor del Habla Multifuente, se creó un servicio *RESTful* bajo la arquitectura de microservicios, el cual demostró versatilidad para adaptarse a diferentes ámbitos sin necesidad de estar relacionados. Se logró comunicar el Reconocedor de Habla Multifuente, de manera bidireccional, con tres sistemas: 1) microservicios que funcionen como intermediarios para delegar las tareas a otros microservicios del sistema CoDiSPa, 2) dispositivos relacionados con el Internet de las Cosas y 3) sistemas que requieren la transformación de audio capturado a un formato de texto para su posterior procesamiento, mediante una *RESTful API*.

Para ayudar en la captura permanente del audio del entorno, se creó un sistema, de apoyo para la implementación del Reconocedor del Habla Multifuente, cuyo objetivo es tomar las peticiones realizadas por el usuario, mediante su voz, como se muestra en la Figura 5.2. Este sistema está construido para la captura continua de audio, lo que significa que los dispositivos, que forman parte de este sistema, siempre están grabando el sonido del ambiente en donde están localizados. Adicionalmente, se construyó una aplicación para el sistema operativo *Android* para capturar audio del usuario bajo demanda de éste. La idea principal de la utilización de estos dispositivos es que el usuario sea capaz de colocar varios de ellos en diferentes ubicaciones de su entorno y con ello cubrir la mayoría del espacio disponible en su hogar. Como puede observarse en la Figura 5.2, el Reconocedor del Habla Multifuente tiene la posibilidad de comunicarse con diferentes fuentes de datos que le envían los audios a procesar. El Reconocedor de Habla Multifuente también es capaz de comunicarse con diferentes sistemas como lo son: la app para *Android* y el Mensajero que es un componente del sistema CoDiSPa, como puede observarse en la Figura 5.3.

Para cumplir con las tareas que el microservicio debe llevar a cabo, se planteó la implementación de los siguientes módulos, como se muestra en la Figura 5.2: 1) Módulo de identificación de usuarios, 2) Módulo de procesamiento de lenguaje natural y 3) Módulo de comunicación. En las siguientes secciones se describen los detalles técnicos.

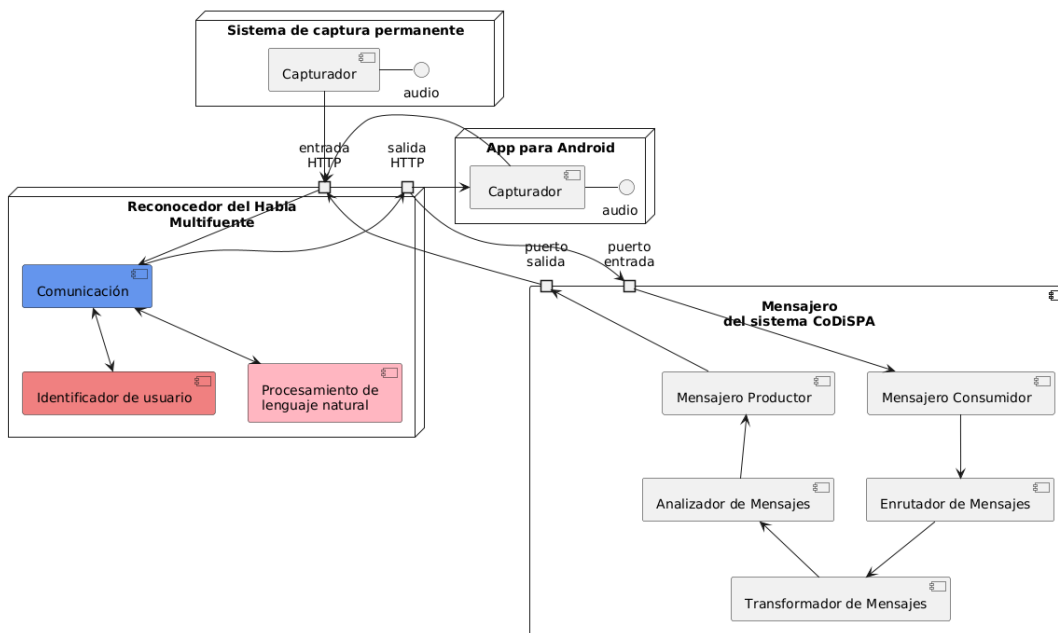


Figura 5.2: Módulos del Reconocedor del Habla Multifuente, comunicación con fuentes de audio y con componentes del sistema CoDiSPa

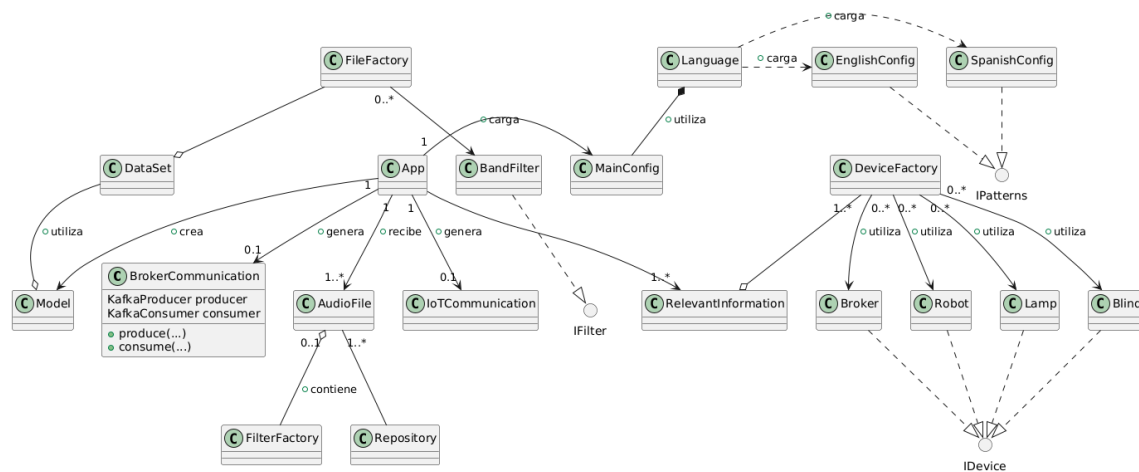


Figura 5.3: Diagrama de clases del Reconocedor del Habla Multifuente

Este módulo está encargado de la identificación de usuarios previamente registrados mediante su voz, que realizan peticiones al microservicio, así como de la agregación de nuevos usuarios. El módulo cuenta con un repositorio por cada usuario registrado, en donde se localizan las muestras de audio proporcionadas para su registro. Para el procesamiento de lenguaje natural se emplea *SpaCy*¹. Las muestras de audio son preprocesadas y utilizadas para el entrenamiento de una red convolucional que ayuda a la identificación de los usuarios [84]. El registro de un usuario en el microservicio se realiza mediante el uso de un teléfono inteligente, utilizando una aplicación *Android* creada para tal propósito.

Para la detección del tipo de petición solicitada por el usuario, se hace uso de una función de *SpaCy* denominada Similaridad Semántica o *Semantic Similarity* que, en español, puede ser utilizada solamente por los modelos *es_core_news_md* y *es_core_news_lg*. Esta funcionalidad permite medir la similitud entre dos textos en términos de su significado. Utiliza modelos de lenguaje pre-entrenados para representar las palabras y frases en un espacio vectorial semántico, donde las similitudes se calculan mediante medidas como la similitud de coseno². Mientras más cercanos estén los vectores de representación de dos textos, mayor será su similitud semántica.

Para probar el procesamiento y reconocimiento de ordenes, se generaron tres oraciones correspondientes a cada tipo de petición, contenidas en el archivo de configuración *SpanishConfig.JSON*, como se muestra en el Código 5.1. Las oraciones se generaron tomando en consideración la información requerida para la atención de dicha petición. En el caso de la tarea “Agendar una cita médica”, los datos a obtener son: 1) nombre del médico, 2) hora de la cita y 3) día de la cita. Para la petición de la tarea “Recordatorio de toma de medicamento”, los datos solicitados son: 1) fecha y 2) hora. Finalmente, para la tarea “Búsqueda de información”, se toma en cuenta el texto que se encuentra después de alguna palabra que refiera a la búsqueda de algo, e.g., en la oración “Busca información sobre la Diabetes”, la palabra “Buscar” indica el tipo de acción y se toma, como elemento a enviar al *Microservicio Mensajero*, el texto posterior a dicha palabra.

¹SpaCy es una biblioteca *open source* para el Procesamiento de Lenguaje Natural Avanzado, <https://spacy.io/usage/spacy-101>

²Los modelos se presentan en la liga: <https://spacy.io/models>

```

1 {
2   "SENTENCES":{
3     "APPOINTMENTS":{
4       "1":"Agendar una cita para el día jueves a las siete y media de la mañana con el ↵
        doctor Martinez",
5       "2":"Cita para mañana con el doctor Ivan Valdespin a las tres y cuarto de la tarde",
6       "3":"Hazme una cita con el medico temprano el quince de enero"
7     },
8     "REMINDERS":{
9       "1":"Recuerdame tomar mi medicina el día de mañana a las ocho y cuarto de la mañana",
10      "2":"Acuerdame que tengo que tomar mi medicina hoy a las siete veinte",
11      "3":"Recordar tomar mi medicamento a las siete de la noche"
12    },
13    "SEARCH":{
14      "1":"Busca información sobre la diabetes",
15      "2":"Dime cómo debo tomar mi medicamento para la diabetes",
16      "3":"¿Cómo se toman las pastillas de paracetamol?",
17      "4":"Buscame información sobre las personas con diabetes"
18    }
19  }
20 }

```

CÓDIGO 5.1: Oraciones utilizadas para la identificación del tipo de peticiones

5.4 Procesamiento de lenguaje natural

Este módulo tiene como tareas principales la traducción de archivos de audio a texto y la obtención de la información necesaria para la ejecución de la petición del usuario. Dependiendo de la configuración, este módulo se activa después de la identificación del usuario o directamente al recibir una petición.

El módulo tiene la capacidad de traducir, a formato de texto, las peticiones realizadas por la aplicación de *Android* o el sistema de captura permanente. Una vez realizada la traducción, se emplea *SpaCy* para el procesamiento de lenguaje natural, de manera local, i.e., sin la necesidad de utilizar el cómputo en la nube. El procesamiento local de lenguaje natural garantiza la privacidad de la información del usuario, cumpliendo con el principio de *Gestión de Datos Centrada en el Usuario* (planteado en el Capítulo 3).

Este módulo es capaz de identificar al sistema con el que necesita interactuar para ejecutar la petición del usuario. Para interactuar con los sistemas y se poder ejecutar la petición del usuario, se emplean técnicas de procesamiento de lenguaje natural para generar y guardar los datos necesarios en un archivo *JSON* que se envía al sistema correspondiente mediante el módulo de comunicación.

El módulo de Procesamiento de lenguaje natural cuenta también con la capacidad de identificar si la petición solicitada es posible ser ejecutada o no, tomando en conside-

ración el idioma en que se ha inicializado el microservicio y los sistemas a interactuar (e.g., el microservicio sólo puede interactuar con dispositivos *IoT* si la petición se realiza en inglés). En caso de que la petición no sea viable, se le informa al usuario que la petición realizada no ha podido ser identificada. Asimismo, si el módulo no logra obtener los datos requeridos para la generación del archivo *JSON*, se le informa al usuario que la petición no fue realizada correctamente.

Para validar las funcionalidades, se realizaron varias pruebas para la detección del tipo de petición con base en la similaridad semántica. Se inicio definiendo cinco oraciones por cada tipo de petición y el procedimiento a seguir es el siguiente. Primero se obtiene la similaridad de la petición generada por el usuario por cada oración predefinida y se calcula el promedio de la similaridad de cada tipo de petición, generando así, tres promedios de similaridad, uno por cada tipo de petición. Se usa el promedio más grande para la identificación del tipo de petición. Sin embargo, esto genera dos problemas: 1) el tiempo de cómputo es considerablemente alto (aproximadamente tres segundos) y 2) la identificación del tipo de petición no es precisa. Este problema es debido a que el promedio obtenido por cada petición puede variar, de manera significativa, si alguna de las oraciones obtiene un valor bajo aún si las demás oraciones tienen valores altos, causando que el promedio se reduzca por una similaridad baja.

Para la identificación de la información requerida se generaron patrones, bajo el concepto de coincidencias basadas en reglas. Se generaron un total de 24 patrones encargados de la identificación de la información necesaria para su envío al *Microservicio Mensajero* del compañero digital.

Los patrones generados posibilitan la identificación de la información necesaria para cada tipo de solicitud. La implementación de estos patrones en el microservicio permite la identificación eficaz de la información requerida, incluso cuando se proporciona de diversas maneras. Esta capacidad aporta robustez al microservicio, ya que los usuarios no están obligados a formular sus solicitudes de manera estandarizada, e.g., los usuarios pueden especificar la fecha de múltiples formas, e incluso usando expresiones imprecisas como “mañana”.

5.5

Comunicación entre módulos

El objetivo principal de este módulo es enviar la información extraída y estructurada en formato *JSON*, proveniente del módulo de procesamiento de lenguaje natural, a sistemas externos, como se muestra en la Figura 5.2, que permiten la ejecución o respuesta de las peticiones del usuario. Este módulo es capaz de facilitar la interoperabilidad con sistemas externos, tanto para la recepción como para el envío de datos en formato *JSON*.

Por esta razón, cumple con la flexibilidad debido a que puede adaptarse a diversas necesidades y requisitos.

La comunicación entre módulos es directa, mediante el paso de mensajes del paradigma orientado a objetos, pero para comunicarse con sistemas externos se utiliza el protocolo *HTTP/S*, tanto para la recepción como el envío de la información extraída por el módulo de procesamiento de lenguaje natural. También, el módulo de comunicación permite la utilización de métodos alternativos de comunicación, como es el uso de colas de procesamiento y envío de mensajes, que le permitan interactuar con sistemas de mensajería y eventos, brindando la flexibilidad necesaria para adaptarse a casi cualquier sistema externo.

Una vez que se ha generado el archivo en formato *JSON* con la información requerida para la atención de la petición por el compañero digital, se utiliza el método *produce* de la clase *BrokerCommunication*, implementada en el *Reconocedor del Habla Multifuente*, para el envío de la información mediante la *framework Apache Kafka*. Para validar este mecanismo de comunicación, se realizaron pruebas del envío de la información contenida en el archivo en formato *JSON* a un *Broker Service* implementado de manera local, con el fin de comprobar que el sistema reconocedor del habla multifuente es capaz de funcionar de manera adecuada.

5.6 Desarrollo del microservicio

Se implementó el microservicio en colaboración el proyecto de tesis: “Sistema de reconocimiento de voz y habla multi-fuente orientado a compañeros digitales” [85]. Para la implementación del microservicio, se emplearon los *Stacks* de desarrollo y el *hardware* siguientes:

- *Python* y *JAVA* como lenguajes de desarrollo
- *Whisper* y *Jasper* como procesadores de lenguaje natural
- *Python Flask* y *BootStrap* como marcos de trabajo para implementar *RESTful Services*
- *OpenVino* y *NVidia CUDA* como marcos de trabajo para la aceleración de los modelos de procesamiento de lenguaje natural
- *ActiveMQ* y *Kafka* como sistemas de manejo de mensajes
- *Raspberry pi* y *ESP32* como plataformas para el despliegue del sistema
- *Neural Compute Stick* como acelerador de ejecución para los modelos de *NPL*

Se desarrollaron las pruebas de concepto de *software* para validar el uso de las tecnologías.

1. Creación del entorno de desarrollo para *Continuous Integration*
2. Creación de aplicaciones como prueba de concepto de las tecnologías propuestas
 - a) Creación de un cliente RESTful que reciba un archivo de audio y probarlo usando *POSTMAN* como servidor *Dummy* y viceversa.
 - b) Creación de un servidor RESTful que emplee *Whisper* y *Python Flask* para procesar archivos de audio y generar archivos de texto para ser transmitidos
 - c) Creación de una aplicación con *Python* para realizar pruebas de rendimiento activando la aceleración con *CUDA*, *OpenVino* y dispositivos de *Edge Computing* como *Neural Computa Stick*
 - d) Una vez teniendo las aplicaciones cliente y servidor, se realizaron pruebas de integración *front-end* y *back-end* para las pruebas de concepto
3. Creación de una aplicación cliente y servidor que use ActiveMQ y Kafka

Se realizó el siguiente proceso para desarrollar el microservicio *Reconocedor del Habla*:

1. **Crear clientes RESTful:** como aplicación móvil y aplicación en dispositivos de Internet de las Cosas. El cliente debe hacer el envío de un archivo de audio, en el formato correspondiente, usando el verbo *POST* del protocolo *HTTP* a una *API* que procesa dicho archivo y da una respuesta. La respuesta se contiene los campos: *filename*, *process_time* y *transcription*. El cliente aplicación móvil para *Android* se evolucionará iterativamente hasta lograr el objetivo del proyecto. Adicionalmente, el sistema se *Dockeriza* para que se puedan agregar más clientes que realizan la misma tarea. Los clientes se despliegan en la plataformas de desarrollo de *IoT*: *ESP32*, *Raspberry pi pico*, *Raspberry pi zero* (1 o 2).
2. **Backend.** Servicio *RESTful* capaz de procesar audio para entender comandos definidos, con mecanismo para procesar archivos de configuración, que emplea varios modelos o motores de reconocimiento de voz: *Whisper* y *Jasper*. La selección del modelo para el reconocimiento de comandos de voz es configurable. En la implementación del *Backend* se emplean los patrones: *Strategy*, *Singleton*, *Factory* y *Wrappers* (*Adapter*, *Proxy*, *Decorator*, ...).

3. **Integración y pruebas de rendimiento.** Se realizaron pruebas unitarias de los puntos 1 y 2. Se realizó la integración del *back-end* y los Clientes para asegurar el funcionamiento correcto del sistema completo. Una vez que se garantiza el funcionamiento correcto se mide el rendimiento del sistema modificando los parámetros de los modelos, envío de audios de distinta duración, variando el medio de conexión (*WIFI, Ethernet, Bluetooth*) para encontrar los parámetros adecuados.
4. **Aceleración del *back-end* empleando *Edge Computing* y *Openvino*.** Se experimentó si con el uso de dispositivos de *hardware* con los *frameworks* apropiados nos ayudan a mejorar el desempeño del *back-end*.
5. **Iteración sobre *back-end* y Clientes.** A partir de esta iteración se logró evolucionar el microservicio. Se incorporó el soporte de envío de mensajes cliente-servidor para los sistemas *ActiveMQ* y *Kafka* desplegados como servicios *Dockerizados* en el laboratorio de computación ubicua del Departamento de Computación. Se realizaron varios hitos: 1) Aceleración del desempeño del servicio mediante *OpenVino, CUDA* y *Edge Computing devices*, 2) Implementación de llamadas a una *API* de encolamiento de un sistema tipo *Broker* que se encuentra en el microservicio de intercomunicación para el Compañero Digital (microservicio Mensajero).

Apoyo para el paciente en el tránsito de un hospital

Contenido

6.1	Jerarquía de Áreas	86
6.2	Antecedentes	87
6.3	Administración distribuida de áreas	89
6.4	Caso de estudio: visita de un paciente al hospital para consulta y realización de estudios	94
6.5	Pruebas con usuarios finales	100

Las organizaciones navegan constantemente por un panorama complejo de usuarios, información y políticas de seguridad. Los trabajadores y visitantes nómadas se mueven por toda la organización, accediendo a recursos y servicios para lograr sus objetivos. Sin embargo, las soluciones de control de acceso existentes, a menudo siguen, un enfoque centralizado de propósito general, que puede conducir a limitaciones como cuellos de botella y fallas del sistema, malentendidos y confusión en las personas.

En algunos casos, el control de acceso se realiza de forma manual por los propios trabajadores. En este capítulo, proponemos una solución innovadora: un Sistema de Gestión de Área (*Area Management System*, AMS, por sus siglas en inglés) que aborda las necesidades únicas de los usuarios nómadas. AMS tiene en cuenta los objetivos y requisitos de los usuarios nómadas, las políticas de seguridad de las organizaciones y los permisos y restricciones de áreas específicas. Al distribuir la gestión de control de acceso, la solución propuesta puede proporcionar un enfoque más eficiente e integral

para administrar recursos, servicios y información dentro de un área.

Este capítulo está organizado como sigue. En la Sección 6.1, se presenta el concepto de Jerarquía de Áreas. En la Sección 6.2, se ofrece una visión general de los trabajos relacionados con los temas principales de la propuesta de gestión de áreas, incluidos los sistemas de control de acceso y los flujos de trabajo. En la Sección 6.3, se presenta el modelo y las principales aportaciones de esta propuesta. La Sección 6.4 incluye un caso de estudio para mostrar la aplicación de los conceptos destacados en los apartados previos. Por último, la Sección 6.5 detalla las pruebas del prototipo con usuarios finales.

6.1 Jerarquía de Áreas

Las personas se mueven entre diferentes áreas dentro de cualquier tipo de organización (e.g., universidades, hospitales, empresas o entidades gubernamentales). Al interactuar con otras personas y sistemas informáticos, pueden acceder a servicios y realizar acciones específicas en estas áreas para lograr sus objetivos parciales y globales [86]. Cada área aporta una parte de la solución final proporcionando la información correcta a las personas adecuadas, según las políticas de administración y seguridad de esa zona. Así, los usuarios nómadas conservan piezas de una solución a través de las zonas que visitan para cumplir con las tareas asignadas [87]. Sin embargo, en algunos casos, estas tareas no se llevan a cabo de forma eficiente, ya que no existen directrices claras para la gestión de recursos y servicios.

Bajo esta perspectiva, toda organización puede estructurarse y modelarse como una jerarquía de áreas [88], i.e., una estructura de árbol en la que cada área está representada por un nodo, ya sea encapsulando (ascendiente) o encapsulado (descendiente) [89]. Este modelo representa las relaciones organizativas entre áreas más que las características físicas, como el tamaño o la forma. Las áreas situadas en niveles superiores ayudan a los usuarios a realizar tareas generales, mientras que las situadas en niveles inferiores, les ayudan a realizar tareas más específicas o especializadas [90].

Como ejemplo del enfoque de modelado anterior, suponga un hospital. En el nodo raíz, se encuentra la Administración General, que establece tanto las políticas generales de seguridad de la organización como los permisos de acceso para los departamentos incluidos como¹: áreas médicas (Anestesiología, Cuidados intensivos, Hematología, Nefrología, Oncología, Neumología, Rehabilitación, Cardiología, Pediatría y Medicina interna), áreas quirúrgicas (Cirugía general, Dermatología, Cirugía ortopédica y traumatología, Oftalmología, Obstetricia y Ginecología, Urología y Otorrinolaringología) y áreas

¹<https://infraestructuramedica.mx/areas-de-un-hospital-como-se-dividen-las-areas-de-un-hospital/>

de diagnósticos y de apoyo clínico (Laboratorios clínicos, Farmacia, Radiodiagnóstico y Medicina preventiva). Estos departamentos, al ser nodos inferiores, heredan las normas generales de la Administración General, sin embargo, establecen políticas propias, ya que también pueden tener áreas específicas (incluso nodos inferiores) con normas más estrictas, e.g., el interior de la Farmacia tiene acceso restringido, ya que alberga medicamentos peligrosos. Todas estas políticas pueden repercutir negativamente en el aprovechamiento de los recursos y servicios del hospital, ya que, aunque pretenden mantener la seguridad, su aplicación es engorrosa y puede resultar confusa, sobre todo si deben aplicarse de forma temporal, e.g., a un repartidor de medicamentos.

A través de nuestra propuesta AMS (*Area-based Management System*), se pretende dar soporte al desarrollo de aplicaciones para usuarios nómadas (e.g., personal médico, pacientes y repartidores) dentro de organizaciones lógicamente subdivididas en áreas. Esta propuesta presenta una solución descentralizada en la que cada área gestiona, de forma autónoma, el acceso a sus recursos y flujos de trabajo, que establecen el plan de actividades que los usuarios nómadas llevarán a cabo dentro de un área. En lugar de definir un flujo de trabajo global centralizado para cada rol de usuario, la solución propuesta define un sistema de flujo de trabajo distribuido basado en áreas, en el que cada área tiene la responsabilidad de definir los roles desempeñados por los distintos usuarios nómadas a los que desea dar soporte. Además, las normas de seguridad del flujo de trabajo de cada área se heredan de las áreas superiores, siguiendo la estructura institucional. El flujo de trabajo general no está predeterminado estáticamente y el subflujo de trabajo de cada área puede redefinirse dinámicamente y adaptarse a casos específicos (posiblemente temporales). Las distintas áreas colaboran compartiendo información para garantizar la consecución de los objetivos de los usuarios nómadas, respetando al mismo tiempo las políticas de seguridad de la organización.

6.2 Antecedentes

La mayoría de los trabajos de investigación abordan el control de acceso y los flujos de trabajo de forma separada, i.e., estos temas no suelen combinarse en una única propuesta de solución como lo hace el trabajo propuesto en este capítulo. En lo que respecta a los sistemas de control de acceso, el modelo RBAC (*Role Base Access Control*) [91, 92] ha sido reconocido como un modelo fundamental para la construcción de sistemas de control de acceso, tanto en el ámbito académico como en la comunidad de ingeniería de software. Aunque existen múltiples trabajos centrados en el modelo RBAC (como los propuestos por Kim y Park [93], van der Laan [94], Cao et al. [95], Pasquale et al. [96], y Ben Fadhel et al. [97]) este modelo no aborda el esquema de gestión de áreas autónomas. Por otro lado, en aplicaciones informáticas móviles y

nómadas, el modelo RBAC se ha ampliado para tratar los requisitos de los usuarios dentro de espacios inteligentes [98].

En los últimos años, se han propuesto varias extensiones del modelo RBAC para tratar restricciones y requisitos específicos. Por ejemplo, T-RBAC [99, 100] extiende el modelo RBAC para soportar restricciones temporales en la habilitación y deshabilitación de roles. Este modelo se extiende también en T-RBAC Generalizado [101] para manejar condiciones temporales en las asignaciones usuario-rol y rol-permisos. Otra extensión, S-RBAC [102, 103], admite restricciones espaciales en la habilitación y deshabilitación de roles, pero no aborda la estructura jerárquica de áreas. GEO-RBAC [104] amplía el modelo RBAC para incorporar información espacial y basada en la ubicación, introduciendo el concepto de “función organizativa delimitada geográficamente”. LoT-RBAC [89, 105] agrega las dimensiones temporal y espacial al modelo RBAC e introduce el concepto de jerarquía de localización. El modelo Or-BAC (*Organizational Based Access Control*) [90, 106] aborda la gestión de comunicaciones virtuales de forma segura, adaptando el modelo RBAC para tratar con suborganizaciones. RBAC-ADS (*RBAC Administration in Distributed Systems*) [107] y dRBAC (*Distributed Role-Based Access Control for Dynamic Coalition Environments*) [108] proponen mecanismos de control de acceso distribuido. Aunque estos modelos han añadido dimensiones útiles a RBAC, no abordan adecuadamente los requisitos específicos del esquema de gestión de áreas propuesto en el presente capítulo.

Se han realizado algunos trabajos sobre sistemas de flujo de trabajo y el modelo RBAC. Bertino et al. [109] desarrollaron un método para asignar roles y usuarios a tareas en un flujo de trabajo. *Task-Role Based Access Control* [110] se centra en los permisos que no se asignan directamente a los roles, sino primero a las tareas relacionadas y después a los roles. El modelo W-RBAC [111] presenta dos modelos que incorporan RBAC a los sistemas de flujo de trabajo: WO-RBAC, que incluye la separación de preocupaciones, y W1-RBAC, que amplía el modelo básico incorporando capacidades de gestión de excepciones. Sun et al. [112] proponen el diseño y la arquitectura de un sistema de flujo de trabajo flexible que incorpora RBAC. AW-RBAC [113] es un modelo RBAC ampliado para sistemas de flujo de trabajo adaptativos que incluye operaciones de cambio y varios objetos sujetos a cambio dentro de los sistemas de flujo de trabajo. A pesar de su contribución, estos trabajos no abordan las necesidades específicas de las áreas autogestionables en las que se lleva a cabo una gestión autónoma.

Por último, se han llevado a cabo varios estudios sobre la intersección de los sistemas de flujo de trabajo y la computación móvil y ubicua. Maurino y Modafferi [114] proponen un modelo para dividir un único flujo de trabajo en múltiples flujos de trabajo autónomos, soportando funciones desconectadas para redes inalámbricas en escenarios móviles. uFlow [115] es un marco de servicios de flujo de trabajo que integra, gestiona y ejecuta

servicios en entornos ubicuos. A pesar de sus contribuciones, estos trabajos no abordan los requisitos específicos de un enfoque de estructura jerárquica de áreas, en el que la gestión de los flujos de trabajo asignados a usuarios nómadas se realiza de forma autónoma.

6.3

Administración distribuida de áreas

En esta sección se describe AMS, un sistema que sirve de guía al usuario nómada (e.g., personal médico, pacientes, repartidores) para desplazarse en una organización (e.g., un hospital) en aras de lograr sus objetivos. AMS sigue un enfoque de gestión autónoma de áreas para permitir que cada una realice las tres acciones siguientes: 1) gestionar sus servicios y recursos; 2) intercambiar información con otras áreas; y 3) tomar decisiones basadas en el contexto del usuario nómada para proporcionarle las funciones adecuadas que le permitan alcanzar sus objetivos. Este enfoque de gestión autónoma de áreas aborda las conocidas desventajas de la mayoría de los trabajos relacionados que proponen una solución centralizada [116, 117, 118], como la lentitud de respuesta, la sobrecarga de la red, las restricciones para acceder y gestionar recursos y servicios compartidos, la actualización tardía de los recursos, entre otras. Esta autonomía constituye una piedra angular para soportar, de forma eficiente, las actividades de los usuarios en entornos organizacionales.

AMS se basa en un modelo distribuido que considera una autogestión de los servicios y recursos proporcionados por cada área de una organización. Cada área administra los objetos bajo su dominio (e.g., información de contexto del área en donde se encuentra el usuario nómada, roles, flujos de trabajo e intercambio de datos entre áreas) mientras observa las políticas generales de seguridad de la organización. Por consiguiente, en la solución propuesta se tienen en cuenta las siguientes cuestiones:

- Acciones y tareas que debe realizar el usuario nómada;
- Funciones que se le pueden asignar;
- Recursos que el usuario nómada puede manipular;
- Permisos que autorizan la ejecución de operaciones sobre los recursos;
- Restricciones que especifican los tipos de acceso, como obligaciones, prohibiciones y permisos;
- Almacenamiento y recuperación de información contextual de área;

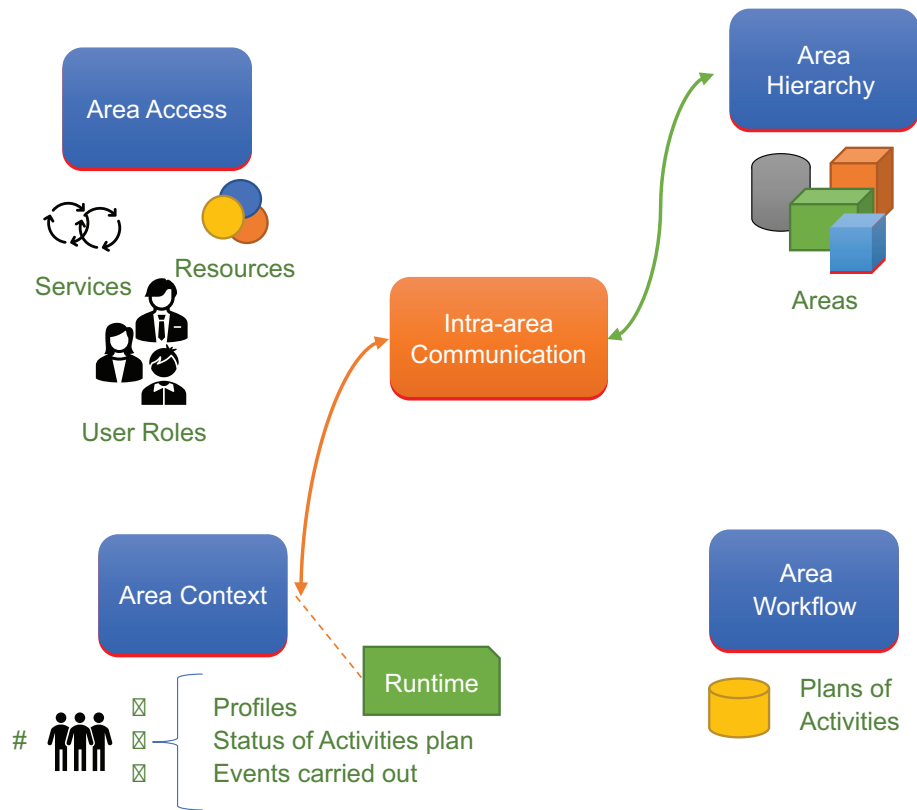


Figura 6.1: Componentes de la arquitectura de AMS

- Interrelaciones e interacciones entre áreas para distribuir y compartir información, servicios y recursos.

Hasta donde se tiene conocimiento, no existe ninguna investigación en las áreas de la Computación Ubicua y la Interacción Humano-Computadora que cumpla estos requisitos de forma exhaustiva.

6.3.1 Arquitectura de AMS

Un área es una entidad lógica que suministra servicios y recursos especializados. Además, un área gestiona su propia información, de forma independiente, para ayudar a los usuarios nómadas a cumplir sus objetivos parciales o globales. La Figura 6.1 muestra la arquitectura de AMS con varios componentes como: *Area Access*, *Area Hierarchy*, *Area Workflow*, *Area Context* y *Intra Area Communications*.

El componente *Area Hierarchy* establece la relación entre el área actual y la estructura jerárquica de áreas y proporciona operaciones que modifican esta estructura, como crear subáreas (descendientes), unir una subárea a otra, eliminar una subárea, copiar una subárea y mover una subárea. El componente *Area Context* se encarga de gestionar la información de contexto del área, que incluye los usuarios nómadas que están dentro de un área determinada, su perfil, el estado de su plan de actividades y la lista de eventos que se están llevando a cabo dentro del área. El componente *Intra Area Communications* envía y recibe información sobre el contexto del área hacia y desde las demás áreas de la jerarquía.

Por otro lado, el componente *Area Access* se encarga de permitir que los usuarios nómadas accedan a los recursos y servicios gestionados por un área. El sistema de control de acceso se basa en el modelo RBAC. Las principales tareas que realiza este componente son: asignar y modificar el rol del usuario nómada; activar y desactivar el rol del usuario nómada; asignar un conjunto de permisos a cada rol; asignar un conjunto de operaciones a cada recurso; y establecer las obligaciones de los usuarios dentro del área. El componente *Area Workflow* gestiona los planes de actividades. Cada vez que un usuario nómada entra en un área, este componente recupera el flujo de trabajo adecuado para dicho usuario, i.e., la lista de actividades que el usuario nómada realizará en esta área.

6.3.2 Organización estructurada por áreas

Se propone un modelo orientado a objetos para el diseño de AMS. En esta propuesta, las áreas y sus entidades contenidas (e.g., recursos, servicios, usuarios, permisos, roles y flujos de trabajo) se representan como objetos que tienen tanto una interfaz pública para interacciones, como una sección privada que oculta detalles internos. Este modelo define un grupo de clases para manejar la organización estructurada en áreas, incluyendo las clases *User*, *Area Hierarchy*, *Access* y *Workflow*. Además, se definen algunas clases de apoyo, como se muestra en la Figura 6.2.

6.3.2.1. Clase “User”

La clase *User* modela al usuario nómada dentro de la organización estructurada por áreas, gestionando información como la identificación del usuario, rol actual, perfil de usuario e identificación de su teléfono inteligente. Es responsable de asignar flujos de trabajo tanto externos (a nivel de organización) como internos (a nivel de área), guardar y recuperar roles, determinar el área actual, identificar la actividad actual realizada por el usuario nómada y obtener el plan de actividades actual asignado a dicho usuario.

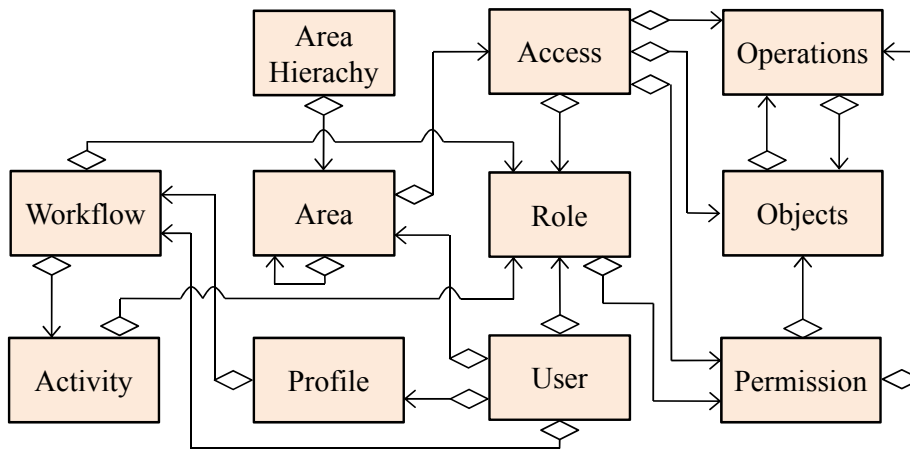


Figura 6.2: Clases de la arquitectura de AMS

El perfil de usuario es una instancia de la clase `Profile`, que se basa en el modelo teórico BDI (*Beliefs, Desires, and Intentions*) [119]. Esta clase proporciona información sobre los objetivos del usuario dentro de la organización (deseos), sus planes de actividad para alcanzar estos objetivos (intenciones) y su comportamiento e intereses (creencias).

Teniendo en cuenta los intereses del usuario, AMS puede hacer recomendaciones, como invitar a un paciente a una plática de salud mental. Si el usuario acepta la invitación, AMS le guiará hasta el lugar del evento. El comportamiento del usuario también es un aspecto importante considerado por la clase `Profile`. Por ejemplo, al final de una visita al hospital, el comportamiento del usuario puede calificarse de “excelente”, “regular” o “malo”, en función de si cumplió o no con las políticas de seguridad establecidas por la Administración General. Esto podría repercutir en el futuro acceso a las instalaciones del hospital.

6.3.2.2. Clase “Area Hierarchy”

La clase `Area Hierarchy` modela la estructura jerárquica de áreas de la organización, abarcando todas las jerarquías de áreas definidas dentro de dicha organización. Esta clase define un nodo (area o subarea) como la raíz de la estructura en el área actual. Permite crear, borrar, copiar, mover un nodo y visualizar la jerarquía de áreas. La estructura jerárquica de áreas está formada por instancias de la clase `Area`.

La clase `Area` se enlaza con su nodo padre y sus nodos hijos y almacena información sobre planes de actividades (a través de la clase `Workflow`), roles, permisos, operaciones y recursos (a través de la clase `Access`) dentro de su dominio. Es responsable de asignar

tanto los flujos de trabajo como los roles a los usuarios nómadas (mediante la clase User).

6.3.2.3. Clase “Access”

La clase Access modela la gestión del acceso dentro de un área, basándose en el modelo RBAC, que incluye usuarios, roles, permisos, operaciones, objetos, separación de funciones y jerarquía de roles. Esta clase utiliza múltiples estructuras de datos para representar los elementos del modelo RBAC. Por ejemplo, utiliza la clase Object para especificar los recursos y servicios dentro de un área, como una red inalámbrica, tensiómetro y balanza. La clase Operation define las operaciones que se pueden realizar sobre estos objetos, como conectarse a una red inalámbrica, medirse la presión arterial o pesarse. La clase Permission contiene una estructura de datos que define los permisos disponibles dentro de un área, vinculando operaciones y objetos. La clase Role define los roles disponibles dentro del área y establece una lista de permisos que admite cada rol. También considera la separación estática y dinámica de funciones dentro de las políticas de seguridad de la organización.

Adicionalmente, la clase Access mantiene un registro del rol de un usuario dentro del área y su rol transferido desde un área vecina. Esta clase ofrece operaciones para gestionar el acceso, incluida la creación de permisos, la asignación de permisos a funciones, la asignación de usuarios a funciones, la conversión de funciones externas en funciones de área interna y la comprobación de la separación de funciones. También proporciona operaciones para recuperar permisos específicos de un rol y roles específicos de un usuario.

6.3.2.4. Clase “Workflow”

La clase Workflow gestiona el plan de actividades de un usuario nómada dentro de un área. La clase Activity define una única tarea y proporciona las herramientas para crearla y completarla. Las actividades múltiples pueden agruparse mediante la clase SetOfActivities (no representada en la Figura 6.2), que describe el orden de ejecución de las actividades y si deben realizarse en paralelo, secuencialmente o de forma combinada.

La clase Activity contiene información sobre la lista de tareas, los roles autorizados para ejecutar el flujo de trabajo y el tiempo máximo asignado al usuario para completar su plan de actividades. Esta clase ofrece funciones para asignar un flujo de trabajo a un rol, iniciarlo y detenerlo, eliminarlo de un rol y determinar la actividad actual del usuario.

6.4 Caso de estudio: visita de un paciente al hospital para consulta y realización de estudios

Para ilustrar cómo el sistema AMS proporciona a los usuarios un soporte basado en áreas para el trabajo o la asistencia nómada, considere el caso de un paciente que primero tiene consulta en Medicina Interna y después un estudio de ultrasonido de las arterias en Radiología. Además del área médica donde se encuentra Medicina Interna, el hospital cuenta con áreas quirúrgicas y áreas de diagnósticos y de apoyo clínico. Por supuesto, el hospital alberga otras áreas no relacionadas directamente con la salud, tales como Administración, Recepción, Contabilidad y Caja, Intendencia, además de salas de espera, un auditorio, una cafetería para los visitantes y otra para el personal, pasillos, estacionamientos y lugares al aire libre. Estas áreas están representadas y lógicamente estructuradas en una jerarquía que está directamente relacionada con la organización del hospital.

6.4.1 Llegada del paciente al área de Recepción del hospital

Inicialmente, el paciente llega a la Recepción del hospital. En esta área inicial, primero tiene que proporcionar una identificación oficial que será escaneada por un miembro del personal de seguridad. Una vez identificado, el subsistema AMS del área de Recepción le asigna el rol de *Visitante* y activa los permisos asociados, así como las obligaciones requeridas por dicho rol. En consecuencia, el paciente debe llenar un formulario de registro en una pantalla táctil, donde debe seleccionar el motivo de su visita y, en función del mismo, se le mostrará una serie de opciones. En este caso, estas opciones son las áreas a las que se dirigirá el paciente y las personas a la que visitará. Así, el subsistema AMS del área de Recepción identifica su propósito como *Consulta de Rutina* con el médico *Tórres* en el área de *Medicina Interna* y *Ultrasonido de Arterias* con el médico *Jiménez* en el área de *Radiología*.

A continuación, el subsistema AMS del área de Recepción envía una consulta a las áreas implicadas (Medicina Interna y Radiología) para conocer la presencia y disponibilidad de personas y recursos necesarios para cumplir el propósito del paciente. Si todo está bien, el subsistema AMS del área de Recepción muestra las políticas globales de seguridad impuestas por el área de Administración del hospital, para que pueda conocerlas. Ejemplos de políticas de seguridad son: “Los vendedores no pueden entrar en el dominio del hospital” o “Los visitantes sólo pueden entrar si tienen una cita programada”. Para asegurarse de que ha leído las políticas de seguridad, el subsistema AMS del área de Recepción no le dejará ir al siguiente paso antes de que transcurra el

tiempo medio que tarda una persona en leerlas. Una vez transcurrido este tiempo, el paciente debe aceptarlas, utilizando su huella dactilar como firma de reconocimiento. La identificación y huella dactilar del paciente, así como el nombre de los médicos a los que visitará, se envían a los subsistemas AMS de Medicina Interna y Radiología, para que estas áreas identifiquen al paciente y le den el soporte necesario para conseguir sus objetivos parciales.

Para determinar si las personas solicitadas, i.e., los médicos Tórres y Jiménez, están presentes en las áreas de Medicina Interna y Radiología, respectivamente, cada miembro entra y sale de los edificios correspondientes utilizando su huella dactilar para registrar su estancia en un área. Este mismo proceso se sigue en las demás áreas del hospital. Si se confirma la presencia del médico Tórres en las instalaciones de Medicina Interna, también es necesario saber si él se encuentra en su consultorio. Utilizando uno de los detectores de presencia situados en los interruptores de luz de los consultorios, el subsistema AMS de Medicina Interna no sólo obtiene información de presencia, sino que además las luces se encenderán o apagaran automáticamente en presencia o ausencia de una persona. Para asegurarse de que la persona solicitada es la que se encuentra en dicha área médica, es necesario verificar el registro de huellas dactilares y la presencia de la persona al interior del consultorio para enviar una respuesta positiva al subsistema AMS del área de Recepción.

Una vez confirmada la presencia del médico Tórres, se proporciona al paciente un dispositivo móvil que cuenta con todos los accesorios necesarios, como una cámara con lectura de códigos QR y acceso a Internet, para que pueda realizar sus actividades dentro del hospital (paso #0 de la Figura 6.3). En caso de que el paciente entre con un vehículo, una cámara situada en la zona de entrada de vehículos identifica el número de matrícula.

Una vez fuera del área de Recepción, el subsistema AMS del hospital toma el control para guiar al paciente hasta su primera parada: el área de Medicina Interna. Este subsistema AMS establece su perfil asignándole el rol de *Paciente* y el flujo de trabajo global que define el plan general de actividades. En este caso, este flujo de trabajo descargado en el dispositivo móvil del paciente se reduce a tres actividades secuenciales: el trayecto del paciente al área de Medicina Interna (paso #1 de la Figura 6.3), el trayecto del paciente al área de Radiología (paso #9 de la Figura 6.3) y el regreso del paciente a la Recepción (paso #14 de la Figura 6.3). Estas áreas ayudarán al paciente a alcanzar su objetivo global. Las áreas no incluidas en este flujo de trabajo (es decir, otras áreas del hospital) no concederán acceso al paciente. De este modo, estas áreas participan en la aplicación de las políticas de seguridad, denegando el acceso a este usuario nómada temporal.

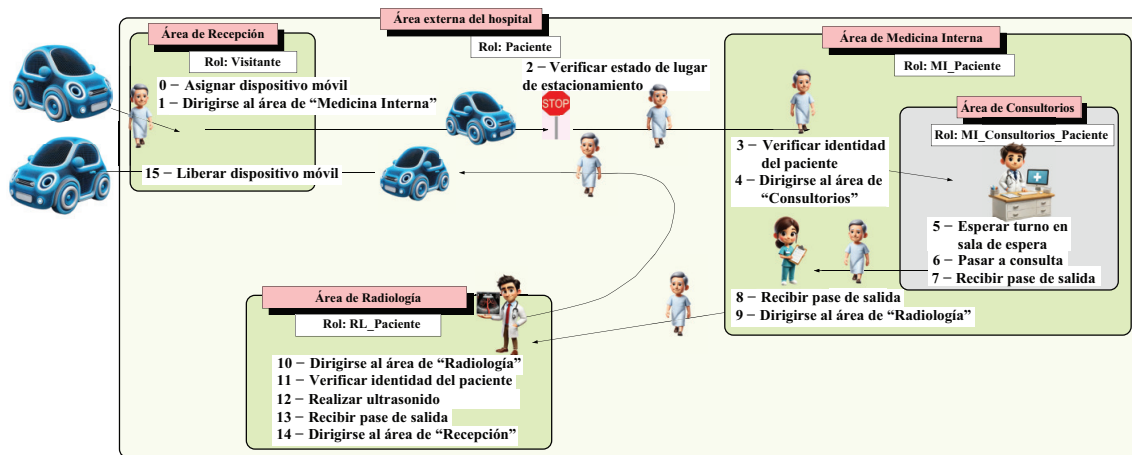


Figura 6.3: Flujo de actividades asistidas por el sistema AMS para realizar una consulta médica y un estudio en un hospital

6.4.2 De la Recepción a Medicina Interna

Como se ha mencionado anteriormente, AMS asigna al paciente un flujo de trabajo global, cuya primera parte consiste en llegar al área de Medicina Interna. El recorrido entre estas dos áreas incluye todo el itinerario (posiblemente pasando por otras áreas) que el paciente puede hacer. Para realizar el trayecto, el paciente utiliza como guía un mapa interactivo del hospital enriquecido con información contextual relevante.

El subsistema AMS del hospital comprueba si alguna de las plazas designadas para el estacionamiento de vehículos de pacientes está disponible, enviando una consulta al subsistema AMS de Medicina Interna (paso #2 de la Figura 6.3). Esta información se obtiene mediante la monitorización de las plazas de estacionamiento, que están equipadas con sensores de proximidad colocados estratégicamente alrededor de cada plaza, además de un identificador.

A cualquier nivel, la tarea de cada área consiste en: a) activar los permisos asociados, b) establecer las obligaciones del usuario nómada con el hospital, c) informar a otras áreas del hospital sobre el perfil del usuario y d) recuperar un subflujo de trabajo adecuado para gestionar la tarea del usuario nómada dentro de esta área. Así, el flujo de trabajo global se divide en subflujos de trabajo independientes organizados jerárquicamente, de forma que cada área se encarga de su propia administración heredando y aplicando las normas, derechos y deberes impuestos por las áreas superiores. Por esta razón, el subflujo de trabajo que debe realizar el paciente dentro de Medicina Interna es completamente específico de esta área y no puede ser conocido ni gestionado

por ninguna otra área para, así, poder conseguir una administración autónoma de los servicios y recursos.

Cuando el paciente llega a la entrada del edificio de Medicina Interna, coloca su huella dactilar en el sensor, para que pueda ser comparada con la huella enviada por el subsistema AMS del área de Recepción. Una vez verificada la identidad del paciente (paso #3 de la Figura 6.3), el subsistema AMS de Medicina Interna le otorga el rol *MI_Paciente* y crea el correspondiente subflujo de trabajo, que incluye las siguientes actividades: obtener la autorización para acudir al consultorio del médico Tórres (paso #4 de la Figura 6.3), obtener un pase de salida una vez realizada la consulta médica (paso #8 de la Figura 6.3) y salir del edificio para dirigirse al área de Radiología (paso #9 de la Figura 6.3).

El subsistema AMS de Medicina Interna también envía la huella dactilar del paciente al subsistema AMS de consultorios, que notifica su llegada al médico Tórres. Esta notificación se realiza enviando un mensaje de texto al dispositivo móvil del médico Tórres y mostrando la foto de identificación del paciente, junto con las opciones de autorización y denegación en una pantalla táctil, colocada en un lugar accesible únicamente al personal de Medicina Interna.

Una vez que el médico Tórres ha confirmado la opción de autorización, el subsistema AMS de Medicina Interna permite al paciente acceder a las instalaciones y muestra el plano del edificio en su dispositivo móvil para guiarlo. Cuando el paciente llega al área de consultorios de Medicina Interna, debe utilizar nuevamente su huella dactilar para identificarse. En ese momento, el subsistema AMS de consultorios de Medicina Interna le otorga el rol *MI_Consultorios_Paciente* y crea el subflujo de trabajo correspondiente, que contiene las siguientes actividades: esperar su turno en la sala de espera (paso #5 de la Figura 6.3), pasar al consultorio del médico Tórres cuando se le indique (paso #6 de la Figura 6.3) y obtener un pase de salida (paso #7 de la Figura 6.3).

Concluida la consulta médica, el subsistema AMS de consultorios muestra, en la pantalla táctil mencionada anteriormente, un icono positivo y otro negativo para que médico Tórres indique si el paciente requiere o no estudios adicionales, respectivamente. En caso de que él seleccione el icono positivo, se muestran los diferentes estudios posibles para que el médico elija los que el paciente debe realizare. Posteriormente, la recepcionista del área de consultorios se encargará de apoyar al paciente en agendar citas para realizarse los estudios requeridos. Después, el paciente recibirá en su dispositivo móvil un código QR que contiene el pase de salida del área de consultorios de Medicina Interna.

En caso de que el paciente no requiera estudios adicionales, recibirá dicho código QR que le permitirá concluir sus actividades en esta área, la cual le retirará los permisos adquiridos. A continuación, el subsistema AMS de consultorios envía al subsistema AMS

de Medicina Interna una aprobación que permitirá al paciente abandonar el edificio (paso #8 de la Figura 6.3). En caso de que el paciente no disponga de dicha aprobación, no podrá abandonar el edificio y el subsistema AMS de Medicina Interna enviará al subsistema AMS del área de Recepción un mensaje de aviso, para que los miembros del personal de seguridad puedan encargarse del caso. En cuanto el paciente reciba la notificación de aprobación, podrá salir del edificio utilizando su huella dactilar para dirigirse a su siguiente parada: el área de Radiología (paso #9 de la Figura 6.3).

Es importante señalar que las distintas áreas anidadas definen su propio flujo de trabajo, especificando lo que necesitan y lo que producen. El subsistema AMS de cada área asigna un rol específico al usuario nómada, permitiéndole obtener permiso para acceder a algunos recursos locales (e.g., al baño del área de consultorios donde trabaja el médico Tórres). Además, la seguridad se refuerza denegando el acceso a algunas partes restringidas de estas áreas.

6.4.3

De Medicina Interna a Radiología

Al estar fuera del edificio de Medicina Interna, el paciente tiene que llegar al área de Radiología para realizarse un ultrasonido de arterias solicitado por su médico tratante del área de Cardiología. En este caso, el subsistema AMS del hospital le ayuda, de nuevo, a llegar a este destino y guía sus movimientos dentro del dominio del hospital.

Dado que se ha supuesto que el paciente llegó en un vehículo, cuando abandona la plaza de estacionamiento de Medicina Interna, el sistema de sensores de proximidad detecta la liberación de dicha plaza y lo notifica al subsistema AMS de Medicina Interna para que cambie el estado de la plaza de estacionamiento a libre.

Cuando el paciente se acerca al edificio de Radiología, el sistema AMS del hospital solicita al subsistema AMS de Radiología que asigne una plaza al vehículo del paciente. Como en el caso del estacionamiento de Medicina Interna, cada plaza tiene el mismo sistema de sensores de proximidad y un identificador. Así, la aplicación móvil proporciona al paciente el número de plaza de estacionamiento que debe utilizar, según sus condiciones médicas. En caso de que no haya plaza de estacionamiento libre, se procesa como una excepción, como en cualquier sistema de flujo de trabajo.

Ya en la entrada del edificio de Radiología, al igual que hizo el paciente en el edificio de Medicina Interna, coloca su huella dactilar en el sensor para acceder. Una vez verificada la identidad del paciente, el subsistema AMS de Radiología le permitirá conocer las políticas de seguridad del edificio a través de su dispositivo móvil. Al tratarse de un edificio donde existen aparatos potencialmente peligrosos, la seguridad en su interior es mayor, por lo que se prohíbe el uso de artículos como sombreros o gafas oscuras.

Una vez que el paciente ha aceptado estas políticas de seguridad, una cámara colocada en la entrada verificará el cumplimiento de las políticas de seguridad para entrar en el edificio. Esta cámara forma parte de un sistema que utiliza un algoritmo de inteligencia artificial para reconocer los objetos no permitidos. Una vez confirmado esto, el subsistema AMS de Radiología le otorgará el rol de *RL_Paciente*, que le permite acceder al edificio, y crea el subflujo de trabajo correspondiente, que incluye las siguientes actividades: ir a la zona restringida de ultrasonido (paso #10 de la Figura 6.3), poner el código QR en el escáner y su huella dactilar para verificar que es la persona correcta a la que se le hará el estudio (paso #11 de la Figura 6.3), realizarse el estudio (paso #12 de la Figura 6.3) y finalmente obtener un pase de salida y abandonar el edificio para volver al área de Recepción (paso #13 de la Figura 6.3).

En el dispositivo móvil, el paciente visualizará también el plano del edificio que contiene solamente el recorrido hasta la zona restringida de Ultrasonido, ya que no tendrá acceso a ninguna otra parte del edificio, excepto a los baños. Una vez que se encuentra en esta zona, el paciente deberá escanear el código QR (obtenido en el área de Medicina Interna) pues funje como un permiso de acceso a esta zona restringida y colocar su huella dactilar en el sensor para ser identificado y recibir su turno para el estudio. Una vez completado el ultrasonido de arterias, el paciente debe firmar un aviso de realización de estudio en una pantalla táctil y el médico Jiménez debe aprobar su salida del edificio, al igual que se hizo en Medicina Interna. Este subflujo de trabajo finaliza cuando el paciente sale del edificio de Radiología, utilizando su huella dactilar.

6.4.4

Salida del paciente del hospital

Una vez fuera del edificio de Radiología, el subsistema AMS del hospital actualiza, de nuevo, su rol y subflujo de trabajo, para que el paciente pueda dirigirse al área de Recepción y completar su objetivo general. Este subsistema presenta nuevos servicios en el dispositivo móvil del paciente, principalmente un mapa interactivo para guiarle de vuelta a recuperar su vehículo y regresar al área de Recepción.

Dado que el propósito de su visita se ha cumplido cuando el paciente regresa a la Recepción, devolverá el dispositivo móvil (paso #15 de la Figura 6.3) que se le proporcionó a la entrada y volverá a colocar su huella dactilar en el sensor para registrar el final de sus actividades (i.e., el flujo de trabajo global). Es importante mencionar que el uso de la huella dactilar también permite al sistema AMS determinar si un visitante intentó entrar en alguna otra zona no especificada en los distintos flujos de trabajo.

Si todo es correcto, el paciente recupera su identificación oficial, mientras que el subsistema AMS del área de Recepción desactiva el rol de *Visitante* y los flujos de

trabajo asociados a este usuario temporal y almacena la información de esta visita en el sistema de persistencia. Finalmente, el paciente puede abandonar el hospital.

6.5 Pruebas con usuarios finales

Para evaluar la eficacia del sistema AMS, se realizó un estudio de usuarios utilizando la herramienta Índice de Carga de Tareas (NASA-TXL) [120] con cinco participantes. Esta herramienta califica la carga de trabajo percibida por los usuarios finales para evaluar la eficacia de una tarea, un sistema, un grupo u otros aspectos del rendimiento. El cuestionario NASA-TXL se compone de seis escalas:

1. *Demanda mental*: ¿Cuanta actividad mental fue necesaria? (e.g., pensar, decidir, calcular, recordar, buscar e investigar) ¿Se trata de una tarea fácil o difícil, sencilla o compleja, pesada o ligera?
2. *Demanda física*: ¿Cuanta actividad física fue necesaria? (e.g., empujar, tirar, girar, presionar y activar) ¿Se trata de una tarea fácil o difícil, lenta o rápida, relajada o cansada?
3. *Demanda temporal*: ¿Cuanta presión temporal sintió el usuario debido al ritmo al que se desarrollaron las tareas o sus elementos? ¿El ritmo era lento y deliberado o rápido y frenético?
4. *Esfuerzo*: ¿Hasta qué punto ha tenido que trabajar el usuario (física o mentalmente) para alcanzar el nivel de resultados?
5. *Rendimiento*: ¿Cual es el grado de satisfacción con el nivel de ejecución?
6. *Frustración*: Durante la tarea ¿en qué medida sintió el usuario inseguridad, desanimo, irritabilidad, estrés, preocupación o, por el contrario, seguridad, agrado, relajación y satisfacción?

Para llevar a cabo estas pruebas, se explicó brevemente a los participantes el objetivo del sistema, así como la intención del cuestionario y el significado de cada escala. Se pidió a los participantes que realizaran una serie de tareas relacionadas con sus roles específicos mientras utilizaban el prototipo del sistema AMS y que después completaran el cuestionario NASA-TXL, asignando una puntuación entre 0 y 100 a cada escala. Aunque el sistema AMS fue concebido para un entorno hospitalario, las pruebas del prototipo tuvieron que ser realizadas en CINVESTAV, dado que no se cuenta con las autorizaciones necesarias para desplegar el sistema AMS en un hospital. En consecuencia, los roles asumidos por los participantes fueron:

1. Un repartidor que iba a dejar un paquete en el Departamento de Toxicología.
2. Solicitante que participa en el proceso de acceso al Master en Informatica.
3. Un profesor que fue invitado como jurado a la defensa de una tesis en el Departamento de Matematicas.
4. Participante en el congreso de Ingeniería Eléctrica.
5. Un técnico de aire acondicionado lleva piezas para reparar una unidad en los servicios escolares.

Las tareas se diseñaron para simular situaciones reales a las que pueden enfrentarse los usuarios nómadas, como encontrar un lugar concreto, acceder a documentos importantes y aprovechar los recursos locales.

La Tabla 6.1 muestra los resultados del cuestionario. En particular, se muestra la ponderación por usuario, así como la media ($\bar{x}$) y la desviación estándar (σ) de cada escala. Como se observa en la Tabla 6.1, la puntuación media de la demanda mental, física y temporal es baja, lo que indica que el sistema fue capaz de reducir significativamente las demandas mentales, físicas y temporales experimentadas por los usuarios. Además, los participantes declararon altos niveles de rendimiento, bajos niveles de esfuerzo y una frustración mínima al utilizar el sistema AMS.

Tabla 6.1: Resultados de la carga de trabajo experimentada por usuarios del sistema AMS

Usuario	Demanda Mental	Demanda Física	Demanda Temporal	Rendimiento	Esfuerzo	Frustración
1	5	5	5	100	5	0
2	2.5	0	5	90	2.5	2.5
3	5	2.5	5	100	5	0
4	10	5	10	80	10	5
5	5	0	2.5	95	5	0
$\bar{x}$	5.5	2.5	5.5	93	5.5	1.5
σ	2.73	2.5	2.73	8.36	2.73	2.23

Los resultados de este estudio sugieren que el sistema AMS reduce eficazmente las exigencias mentales, físicas y temporales experimentadas por los usuarios, independientemente de sus funciones específicas. Es probable que esto se deba a la capacidad del sistema para proporcionar un apoyo personalizado basado en la ubicación del usuario y sus necesidades laborales. Las bajas puntuaciones medias de las demandas mental,

física y temporal indican que el sistema fue capaz de aliviar las demandas cognitivas y fisiológicas de las tareas, permitiendo a los participantes centrarse en el logro de su objetivo. Además, los altos niveles de rendimiento y los bajos niveles de esfuerzo comunicados por los participantes indican que el sistema es fácil de usar. La baja puntuación media de frustración también indica que el sistema es intuitivo y sencillo para los usuarios. No obstante, hay que señalar que el estudio se llevó a cabo con una pequeña muestra de cinco participantes, por lo que es necesario seguir validándolo con una muestra mayor. Además, sería interesante ver como funciona el sistema en diferentes escenarios y contextos para comprobar si los resultados son generalizables a otros roles y entornos.

Discusión

Contenido

7.1	Diseño centrado en el usuario	103
7.2	El compañero digital como sistema ubicuo	108
7.3	El paciente como usuario nómada	110

En la era digital actual, la investigación en el campo de la tecnología y la salud ha llevado a una comprensión más profunda de cómo integrar soluciones innovadoras en nuestra vida diaria. Este capítulo se sumerge en la importancia de discutir los temas destacados de esta investigación, explorando tres secciones clave. En primer lugar, el diseño centrado en el usuario se convierte en una piedra angular para asegurar que las tecnologías sean accesibles y efectivas para todos. En segundo lugar, el concepto del compañero digital como sistema ubicuo aborda cómo las tecnologías inteligentes pueden estar presentes en todas partes, ofreciendo apoyo continuo y personalizado. Finalmente, se examina al paciente como usuario nómada, destacando la necesidad de soluciones que acompañen a las personas a lo largo de sus diversas actividades y entornos. Juntas, estas secciones proporcionan un marco comprensivo para entender cómo la intersección entre tecnología y salud puede mejorar significativamente la calidad de vida.

7.1

Diseño centrado en el usuario

Ben Shneiderman es el creador del término “amigable” en el contexto de las interfaces de usuario [121]. Shneiderman afirma que un diseño amigable es aquello que permite

a los usuarios interactuar con la interfaz de manera efectiva y sin problemas. En este sentido, un “compañero”, se puede entender como “amigo” aquel ente que esté cerca del usuario y que no le hace daño.

7.1.1 Un asistente virtual como “compañero”

Un asistente virtual se transforma en “compañero” al diseñar su interfaz de usuario con características amigables que permiten una navegación sencilla y el uso efectivo del *software*. El proponer un diseño con elementos, visuales y auditivos, identificables, organización lógica y acceso rápido a funciones críticas ayuda a que el usuario confíe en el sistema. Por ello, el compañero digital propuesto es capaz de proporcionar una experiencia de usuario agradable y efectiva, que puede mantener al usuario comprometido con su tarea y evitar problemas comunes, como confusiones en la interfaz o desacuerdos en el flujo de trabajo. El sistema CoDiSPa cuenta con dos modos de interacción, modo visual, empleando dispositivos móviles con pantalla, y modo auditivo, empleando voz que llega al paciente a través de las bocinas del dispositivo que tenga a su disposición.

El sistema CoDiSPa fue diseñado para resolver tres tareas que el paciente debe realizar como producto de su padecimiento: gestión de citas médicas, seguimiento médico y mejora de la adherencia a la toma de los medicamentos. Gracias a que el sistema CoDiSPa descarga al paciente de las tres tareas, es que el paciente puede sentir que CoDiSPa es amigable con él. Además de no hacerle daño, CoDiSPa lo ayuda a mejorar su calidad de vida a pesar de padecer una enfermedad crónica.

CoDiSPa se diseñó como un sistema proactivo (también es un sistema reactivo) que está al pendiente de las necesidades del paciente, i.e., el sistema lanza constantemente procesos que sienten el ambiente y al paciente y, con ello, el sistema calcula el momento en que se comunica con él. También, el sistema CoDiSPa realiza procesos de inferencia, a partir de los datos recopilados, para predecir los momentos en que se tendrá una mejor interacción con el paciente. Adicionalmente, CoDiSPa aprende de la interacción con el paciente, de tal manera que, entre más se emplee, más se puede adaptar a sus preferencias. Con este comportamiento del sistema, el paciente, paulatinamente, lo sentirá como un verdadero compañero o amigo virtual que le apoye en sus tareas de salud, hasta el extremo de que, el sistema o partes de este, pasen desapercibidos.

7.1.2 Eficiencia y efectividad

La efectividad y eficiencia son conceptos clave en el diseño de compañeros digitales al cuidado de la salud. La efectividad se refiere a la medida en que un tratamiento o

intervención logra sus objetivos terapéuticos, como reducir los síntomas de una enfermedad o mejorar la calidad de vida de un paciente [122]. Por otro lado, la eficiencia es cómo se logran estos objetivos de manera efectiva y con recursos mínimos necesarios, i.e., maximizando el beneficio y minimizando los riesgos y costos asociados al tratamiento [122].

En este sentido, el compañero digital propuesto tiene en cuenta tanto la efectividad como la eficiencia, para ayudar a los profesionales de la salud a tomar decisiones clínicas informadas y apropiadas, de acuerdo con el estado de salud actual del paciente, e incluso con el histórico de su comportamiento o reacción a los tratamientos prescritos.

El sistema CoDiSPa registra la información de manera desapercibida para el paciente; la información es obtenida de sus acciones cotidianas. CoDiSPa procesa la información para encontrar patrones de comportamiento que detecten anomalías, las cuales notifica al personal médico. Al notificar al personal médico se mejora el flujo de trabajo clínico, se minimizan los errores humanos y los riesgos de una incorrecta interpretación de la información. Esto es, como contar con un equipo médico todo el tiempo, observando el estado del paciente; es por esto que podemos afirmar que CoDiSPa cumple con la eficiencia y la efectividad al incorporarlo al ambiente de un paciente con enfermedad crónica u otra condición de salud permanente.

7.1.3

Manejo de la multimodalidad

La multimodalidad en la interfaz de usuario de compañeros digitales permite a los usuarios utilizar diferentes modos de interacción sensorial-actuadores, según sea la preferencia y necesidad del usuario [123, 124]. Esto puede incluir el uso de audio, vibración, retroalimentación háptica o visual u otro tipo de retroalimentación táctil o sonora que facilite la navegación por la interfaz de usuario y proporcione una experiencia de usuario efectiva.

A partir de emplear escenarios de uso y prototipado de una versión del compañero digital con usuarios potenciales en un ambiente real, se descubrió que no basta con un modo de interacción único, originalmente se pensó sólo emplear audio para interactuar con el usuario, como se tiene en los asistentes virtuales comerciales. Por lo tanto, se agregó el modo de interfaz visual para ayudar a la adopción del sistema por parte de los usuarios. La propuesta del sistema CoDiSPa es multimodal, lo que permite a los profesionales de la salud seleccionar entre diferentes modos según su contexto y tarea. El sistema CoDiSPa es un compañero digital con orientación conversacional, i.e., el paciente tiene un diálogo con el compañero digital indistintamente del modo de interacción: visual (texto y gráficos), auditivo (voz sintetizada y escucha de voz del paciente) o híbrido (visual y auditivo, complementándose). Los modos de interacción tienen: acceso

rápido a funciones importantes, control preciso sobre el *software* y retroalimentación inmediata. Estas características permiten que los usuarios seleccionen entre diferentes modos, según sea necesario o los mezclen de acuerdo a su preferencia o ubicación en el ambiente. El usuario le puede indicar en cualquier momento al sistema, qué modo de interacción prefiere y este será recordado. Eventualmente, el sistema se va adaptando, cada vez más, a los gustos del usuario. Al diseñar la interfaz de usuario con estas características, se maximiza la efectividad del compañero digital y se minimizan los riesgos de confusión o de frustración en el uso o descubrimiento de las funcionalidades.

7.1.4

Carga cognitiva en usuarios

En general, la carga cognitiva en los usuarios se presenta cuando no se cumple con las características de efectividad, eficiencia y satisfacción, i.e., cuando no se cumple con la usabilidad. Por ejemplo, la carga cognitiva en el usuario se da cuando la interacción con el sistema presenta complejidad en la composición de la interfaz, en la navegación de funcionalidades, en el anidamiento de menús, en la presencia de iconografía fuera del contexto del usuario, en la falta de respuesta para informar de las acciones y errores del sistema, entre otras causas.

La carga cognitiva es un reto importante en el diseño de compañeros digitales, y en general en todo sistema de *software*, que puede impedir la satisfacción del usuario y contribuir a la confusión, y producir errores o riesgos asociados con el tratamiento médico [125]. Esto es especialmente grave cuando se trata de pacientes de la tercera edad o analfabetas digitales, quienes pueden tener dificultades para navegar por una interfaz de usuario compleja o no amigable.

A las personas mayores, en especial de generaciones previas a la era de los teléfonos inteligentes, les es difícil hacer uso de las tecnologías comunes de este tiempo, como son: las computadoras, los teléfonos inteligentes, las tabletas, las televisiones inteligentes, los *wearables*, entre otros. Estas personas tienen una inercia a preferir las tecnologías conocidas por ellos y de desconfiar de lo no conocido, i.e., ellos están habituados a emplear tecnologías como: teléfonos fijos, radio, televisión, cine, entre otros. Ellos también gustan de tratar con personas y establecer diálogos con ellas cuando requieren algún bien o servicio. A las personas mayores, no les es natural interactuar con “máquinas” y les causa desconfianza usar: dispositivos móviles, dispositivos de cómputo, cajeros automáticos, máquinas de cobro de servicios, banca electrónica, máquinas expendedoras, entre otras.

El sistema CoDiSPa tiene considerado este reto, porque fue diseñado bajo el concepto de ente conversacional. Por lo tanto, la carga cognitiva en el usuario es mínima, gracias

a que el sistema se comunica con el usuario entablando una conversación, haciendo que eventualmente el usuario confíe en las funcionalidades que le ofrece el sistema.

7.1.5

Descubrimiento de uso con interfaces de voz

Los sistemas tradicionales, los no ubicuos, requieren que el usuario sea preciso en la acción que desea que el sistema realice. Además, en un sistema tradicional, el usuario es quien inicia las acciones. El sistema CoDiSPa es un sistema ubicuo, i.e., el sistema es el ente proactivo y tiene la responsabilidad de adelantarse a las necesidades del usuario porque a medida que interactúa con él, aprende de su comportamiento y se adapta. El objetivo de los sistemas ubicuos es dotar al ambiente de poder de procesamiento y pasar desapercibido por el usuario al grado de no ser notado.

De acuerdo con Norman, un buen diseño, en términos generales, es aquel que tiene dos características: descubrimiento (*discoverability*) y comprensión (*understability*), además de los conceptos relacionados: posibilidades de acción (*affordances*), significantes (*signifiers*), coincidencia (*mapping*), restricciones (*constraints*), retroalimentación (*feedback*) y el modelo conceptual del sistema [126]. El descubrimiento de uso de CoDiSPa y la comprensión de las funcionalidades y acciones que son posibles por parte del usuario, se dan gracias a la interacción conversacional entre el usuario y el sistema. La Figura 7.1 muestra los conceptos del diseño de la interacción centrado en el usuario.

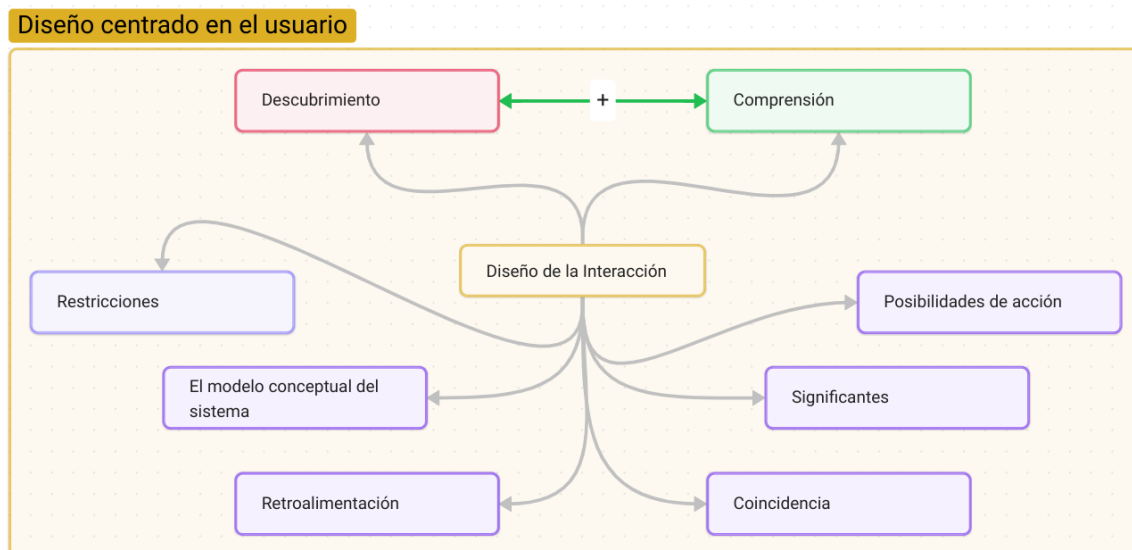


Figura 7.1: Conceptos de diseño centrado en el usuario

Las *affordances* son las posibilidades de acción de un objeto, en este caso de un sistema, que son percibidas por el usuario. No son cualidades del sistema, son una relación determinada conjuntamente por las cualidades propias del sistema y las capacidades del usuario. La relación se nutre de las experiencias, de las estimaciones o de la intuición del usuario. El aprendizaje que tiene CoDiSPa del comportamiento del usuario hace que se adapte a él, esto potencia la experiencia y la comprensión que tiene el usuario sobre las cualidades del sistema. En un proceso de diseño de la interacción tradicional, el objeto es estático en relación a sus *affordances* y el usuario es el que descubre y comprende. En este caso, el objeto es el sistema CoDiSPa, y este también comprende y descubre lo que el usuario puede ordenarle. Si en la interacción de un sistema ubicuo con un usuario, la comprensión y descubrimiento se da en el sistema también, el sistema se acerca a ser un compañero real. Cuando una persona conoce a otra, las dos tienen descubrimiento y comprensión mutuo entre más interactúen. Por esta razón, si el sistema CoDiSPa es capaz de comprender y descubrir a su compañero humano la relación se acerca a una relación entre humanos.

Los *signifiers*, por otro lado, son elementos visuales, o auditivos, que comunican información importante a los usuarios sobre las características del sistema, pero sobre todo del cómo puede ser utilizado. Cuando CoDiSPa entabla una conversación con el usuario, es en ese momento que se dan de manera natural los *signifiers*. Dependiendo del tema de la conversación, CoDiSPa encuentra el momento para dar guías al usuario y este descubre lo que puede hacer el sistema para apoyarlo en su vida cotidiana. Con la interacción cotidiana, el usuario descubrirá el conjunto completo de funcionalidades que el sistema le ofrece y, el sistema, a su vez, descubrirá el comportamiento del usuario.

7.2

El compañero digital como sistema ubicuo

Un sistema ubicuo está fuertemente relacionado con el concepto de *computadora invisible*, sin que invisible se entienda como que las dimensiones físicas del objeto deban ser pequeñas y por eso no son notadas por los sentidos de los usuarios. Weiser [127], postula que las mejores tecnologías son aquellas donde el aspecto físico permanece en segundo plano para el usuario, permitiéndole actuar al realizar una tarea específica, bajo la impresión de que está realizando esa tarea sin emplear una herramienta, i.e., el usuario debe sentir que el sistema es una extensión de sus capacidades y por ello es prácticamente imperceptible. Una herramienta o sistema se convierte en invisible cuando está tan integrada en el ambiente que llega al punto de que el usuario no nota que la está empleando.

El sistema CoDiSPa es un sistema ubicuo que emplea dispositivos del entorno cotidiano de un usuario en el contexto de la vida moderna, i.e., teléfonos inteligentes,

televisiones inteligentes, tabletas, pulseras inteligentes, relojes inteligentes, bocinas inteligentes, por mencionar algunos. De tal manera que el sistema extiende las capacidades de los dispositivos para que realicen funciones adicionales que ayuden al cuidado de la salud del usuario.

Bajo el paradigma de computación ubicua, existen dos tipos de dispositivos, los primarios y los secundarios. Los dispositivos primarios requieren de la atención del usuario y que este lance las tareas que quiere que el dispositivo realice. Los dispositivos secundarios son aquellos que pasan a segundo plano en la percepción del usuario, y cuyas capacidades son extendidas porque se agrega poder de cómputo y comunicación; pero su aspecto original no es modificado. Los dispositivos secundarios se pueden comunicar entre sí y colaborar para lograr una tarea específica.

La arquitectura de CoDisPa integra a los dispositivos entre sí para colaborar en las tareas principales que realiza. Para integrarlos emplea tecnologías de comunicación como: *WiFi*, *bluetooth* y *ZigBee*. El uso de los dispositivos y el sistema CoDisPa se da en un entorno local domótico porque se crea un ambiente computacional para que el compañero digital pueda sentir la presencia y contexto del usuario. CoDisPa aprovecha los avances en modelos de procesamiento de lenguaje natural, y el uso de redes convolucionales pre-entrenadas para llevar a cabo las conversaciones con el usuario. No se emplean servicios en la nube para estas tareas porque se desea mantener la privacidad de la información personal y la información del comportamiento del usuario.

En la lógica de procesamiento de CoDisPa se parte de que el compañero digital siente el ambiente, con el usuario presente, y constantemente lanza procesos para identificar si es el momento adecuado para entablar una conversación y sugerir o preguntar tareas a realizar, adelantándose a sus necesidades. CoDisPa es un sistema proactivo por naturaleza, pues se adelanta a los deseos o necesidades del usuario; sin embargo, CoDisPa puede ser requerido explícitamente por el usuario para realizar alguna tarea o para resolver dudas sobre temas relacionados con el padecimiento o tratamiento médico. Este comportamiento de CoDisPa invierte el control de un sistema tradicional, que son por lo general reactivos, y posibilita una adaptación a las condiciones de uso y preferencias del usuario.

Los sistemas de información empresariales o aplicaciones empresariales, que son por naturaleza los más desarrollados y complejos, fueron convergiendo hacia el uso de servidores especializados para cada tarea requerida en el soporte del despliegue de una aplicación. Por ejemplo, se crearon aplicaciones que daban servicio especializado para manejar bases de datos, servicio de despliegue de aplicaciones Web, servicios de colas de mensajería, servicios para balancear cargas de solicitudes, entre otros. Este tipo de sistemas hicieron que las arquitecturas fueran tendientes a los monolitos

programáticos, hasta el grado, de crear estándares para aplicaciones empresariales que, a pesar de tener módulos, un monolito encapsulaba la lógica del negocio. Las aplicaciones empresariales se hicieron cada vez más costosas de mantener. Entonces, surgió la necesidad de crear otro tipo de arquitectura para mitigar el problema del costo del mantenimiento. La respuesta fue atomizar las funcionalidades y exponerlas mediante la especificación de *RESTful API* en servicios lo más pequeños posible, y con el menor número de responsabilidades. Los servicios trajeron un avance consigo, el desacoplamiento de componentes y la cohesión de funcionalidades, i.e., los sistemas siguieron el principio de alta cohesión y bajo acoplamiento pero a nivel de servicios. El resultado de esta nueva visión para realizar aplicaciones, es la arquitectura de microservicios. Las ventajas de diseñar un sistema basado en microservicios son: escalabilidad, alta disponibilidad, reutilización de componentes, extensibilidad, mantenibilidad, por mencionar algunas. El sistema CoDisPa fue diseñado bajo la arquitectura de microservicios para aprovechar las características antes mencionadas. Adicionalmente, el uso de un sistema que actúe como intermediario para controlar las peticiones entre los microservicios, agrega la ventaja de escalabilidad bajo demanda, porque junto a un manejador de contenedores, puede crear ejemplares de un microservicio cuando este se encuentre saturado de peticiones, y puede eliminarlos cuando se encuentre en inactividad, eficientando el uso de recursos de manera elástica.

7.3 El paciente como usuario nómada

La carga de actividades del paciente no solo se da en su ambiente doméstico, sino también en las instituciones sanitarias a las que tiene que acudir el paciente para sus revisiones o análisis clínicos. La propuesta del sistema CoDiSPa ayuda a mitigar la carga de actividades del paciente pero para crear un ambiente computacional es necesario considerar las actividades que el paciente tiene que realizar cuando se dirige a cumplir con una cita médica o para realizarse los análisis clínicos planeados. Cuando un paciente entra en un hospital se encuentra en un ambiente bastante complejo y esto supone una alteración importante en su estado emocional e incluso de salud porque necesita navegar en la organización para realizar la tarea que lo hizo moverse a dicho ambiente. Por estas razones, fue necesario proponer un sistema basado en áreas, AMS, que convierte al paciente en un usuario nómada, i.e., un usuario que tiene una tarea que cumplir, pero que no permanecerá en el entorno una vez alcanzada la meta. Cuando el paciente se convierte en un usuario nómada, el sistema basado en áreas, le ayuda a realizar la tarea y lo ubica en el ambiente físico o lógico complejo. Para que el usuario nómada pueda lograr su meta, el sistema administra las tareas, los accesos, el mapa de las áreas, y lo guía, mediante pasos claros, para completar la tarea sin agregarle

estrés. El sistema de apoyo para el tránsito de un paciente en un hospital es también un sistema ubicuo que acompaña y ayuda al paciente para cumplir su objetivo en el hospital.

Los sistemas CoDiSPa y AMS colaboran para acompañar al paciente en dos entornos distintos pero conectados por sus necesidades. La naturaleza del sistema CoDiSPa es conversacional y la naturaleza del sistema AMS es un sistema que siente el contexto y la presencia del usuario, aunque no puede adelantarse a las necesidades del usuario nómada, i.e., no es un sistema proactivo. Para agregar los beneficios del sistema CoDiSPa al entorno del sistema AMS, como la característica de proactividad, se espera llevar el ente compañero digital hacia el ambiente creado por el sistema AMS; sin embargo, se deben tomar en cuenta realizar las modificaciones mínimas sobre la lógica del sistema AMS. La manera natural para comunicar los dos sistemas es empleando el mecanismo de mensajería propuesto por CoDiSPa y llevar al compañero digital a enlazarse con el nuevo entorno. Se debe tener cuidado de no compartir los datos personales del paciente, resguardados en el sistema CoDiSPa, hacia el entorno del AMS pues, se debe recordar que, el usuario está temporalmente ligado al entorno del hospital y no tiene sentido que se compartan. También es posible plantear la incorporación del compañero digital como una mascota digital¹ para que el usuario la lleve consigo en un dispositivo para este propósito, i.e., crear un dispositivo portátil donde se encuentre la representación del compañero digital con algún avatar y este pueda entablar conversaciones y le acompaña al paciente en su tránsito hospitalario.

¹En la década de los 80's se propuso comercialmente una mascota digital: el *Tamagotshi*

Conclusiones y trabajo futuro

Contenido

8.1 Conclusiones	114
8.2 Aportaciones	116
8.3 Limitaciones	118
8.4 Trabajo futuro	120

Se describió en este trabajo la necesidad de integrar capacidades computacionales y técnicas para la creación de un ambiente computacional que ayude a pacientes con enfermedades crónicas, como la diabetes, a equilibrar la carga extra de actividades debido a su padecimiento. Para lograr el ambiente computacional, se propuso una arquitectura de *software* para compañeros digitales centrada en el usuario. Asimismo, se propuso una arquitectura para un sistema distribuido, que extendiera el apoyo al paciente en un entorno jerárquicamente organizado para complementar el ambiente computacional. Para lograr que el ambiente computacional atendiera las necesidades del paciente, se exploraron los requerimientos realizando un marco conceptual de trabajo para reflejar sus expectativas y necesidades.

Desde la perspectiva de salud, a partir del estudio de proyectos de investigación en el área, se comprendió que las tres principales tareas que afectan a los pacientes con enfermedades crónicas son: 1) la gestión de citas médicas, 2) el seguimiento médico y 3) la adherencia a la toma de medicamentos. Desde el ámbito computacional, se vio que no existe una plataforma que integre, mediante compañeros digitales, los diversos servicios, aplicaciones, dispositivos e infraestructura necesarias para atender tareas de un paciente con enfermedad crónica. También, se constató que no existe

una especificación para la integración y explotación de los datos generados por los diversos dispositivos que interactúan con asistentes virtuales que hagan la función de compañeros digitales.

Se realizó una amplia revisión de los trabajos relacionados con el uso de tecnologías de Internet de las Cosas, aprendizaje automático y agentes virtuales en el marco de sistemas orientados a la salud. Se comprendió que los pacientes, además de motivarse por un cambio en sus hábitos de autocuidado, se preocupan por la seguridad de su información y la privacidad de sus actividades cotidianas. Adicionalmente, se entendió que los usuarios con enfermedades crónicas requieren sistemas que sean fáciles de usar y que estén altamente disponibles en el ambiente para que los puedan apoyar en el cuidado de su salud.

8.1

Conclusiones

El aumento de la prevalencia de enfermedades crónicas y la necesidad de mejorar los resultados de los pacientes han motivado el desarrollo de diversos dispositivos e implementos de dispositivos de Internet de las Cosas para apoyar a las personas en sus objetivos de salud y bienestar. Se realizó una revisión exhaustiva del estado del arte para comprender las tendencias y desafíos actuales en el diseño y desarrollo de sistemas de Internet de las Cosas orientados a la salud.

Marco de trabajo conceptual

Las mayoría de soluciones existentes necesitan mejorar en términos de usabilidad. Las soluciones que tienen deficiencias de usabilidad no son efectivas y por tanto no son adoptadas por el usuario a largo plazo. El Capítulo 3, propone un marco conceptual de diseño de interacción para sistemas de salud Internet de las Cosas que considera los tres componentes clásicos clave de usabilidad: efectividad, eficiencia y satisfacción. Los resultados de la propuesta destacan la importancia de considerar a todas las partes interesadas al diseñar sistemas con Internet de las Cosas. Al integrar este marco en dispositivos y plataformas de Internet de las Cosas, las personas pueden beneficiarse de una experiencia fluida e integrada, lo que puede ayudarles a alcanzar sus objetivos de salud y bienestar y ayuda en la adopción a largo plazo de estos sistemas.

Para descubrir los conceptos para un diseño apropiado, se realizaron entrevistas con pacientes y profesionales de la salud para desarrollar tres personajes que representan grupos comunes de usuarios para sistemas Internet de las Cosas de salud: a) aquellos con una enfermedad que buscan mejorar su condición, b) profesionales de la salud que requieren más herramientas para apoyar a sus pacientes y c) personas que

quieren una vida más saludable. A través de estos personajes y sus historias de usuario correspondientes, propusimos un marco conceptual que consta de tres componentes principales: 1) Interfaz de usuario como servicios, 2) Interacciones contextuales y 3) Gestión de datos centrada en el usuario. El diseño de la arquitectura del compañero digital (CoDisPa) incorporó los tres conceptos descubiertos. Podemos afirmar que, para diseñar sistemas que proporcionen experiencias personalizadas, conscientes del contexto y centradas en el usuario se deben aplicar estos conceptos. Se demostró la viabilidad del marco conceptual presentando un escenario hipotético de un compañero digital para personas con diabetes que incorpora los conceptos del marco y se tuvieron resultados satisfactorios de la experiencia de usuario con el sistema propuesto.

Arquitectura de CoDisPa

La creación de una arquitectura para compañeros digitales hace posible integrar servicios en línea y dispositivos (de Internet de las Cosas), así como interconectar infraestructura de cómputo ubicuo, de manera local, para poder explotar datos generados en beneficio de los usuarios.

La arquitectura de CoDisPa resuelve las tres tareas, que los pacientes típicamente reportan como carga adicional a su padecimiento, con un enfoque centrado en el usuario. La propuesta aprovecha el desarrollo de *software*, la arquitectura de microservicios, la integración de dispositivos y servicios, así como la posibilidad de crear sistemas empotrados que pasen desapercibidos para los pacientes en su entorno de acción cotidiana.

La arquitectura de CoDisPa aprovecha las herramientas de inteligencia artificial existentes para crear un ente virtual, denominado compañero digital, que además puede ser personalizado para el paciente de enfermedades crónicas. La arquitectura de CoDisPa integra sistemas de uso común que tienen fines específicos como son: los *wearables* que registran actividad física y parámetros biométricos, básculas inteligentes, bocinas inteligentes, entre otros. A diferencia de las aplicaciones específicas para el cuidado de personas mayores, enfermedades cardiovasculares o enfermedades mentales, el enfoque que se empleó prioriza soluciones adaptables y centradas en el usuario, evitando complejidades innecesarias y alineando a CoDisPa con las necesidades propias de un paciente con diabetes.

Mediante la creación de la arquitectura de CoDisPa y la puesta en marcha de algunos componentes con arquitectura de microservicios, como el *Reconocedor del Habla* y el *Mensajero*, se puede afirmar que la arquitectura es viable y correcta, hasta donde se ha implementado. Se ha demostrado, mediante pruebas, que también es viable contar con

el sistema CoDisPa en un entorno doméstico propio de los pacientes con diabetes y que esto ayuda a los pacientes en la descarga de actividades producto de su padecimiento.

Sistema de apoyo para el paciente en el tránsito de un hospital

Con la intención de extender el apoyo a los pacientes cuando salen de un entorno doméstico, hábitat del sistema CoDisPa, se planteó tomar a los pacientes como un usuario nómada. Se propuso el sistema de gestión basado en áreas (AMS) para ayudar a los usuarios nómadas (e.g., personal médico, pacientes y repartidores) en la realización de tareas dentro de organizaciones (e.g., hospitales), divididas lógicamente en áreas. Este enfoque descentralizado permite a cada área gestionar, de forma independiente, el acceso a sus recursos y servicios, así como crear flujos de trabajo que establezcan las actividades que los usuarios nómadas (incluido el paciente de enfermedades crónicas que ha cambiado de entorno) llevarán a cabo dentro de las áreas de interés.

El caso práctico presentado en esta propuesta pone de relieve la importancia de esta solución para organizar y administrar un sistema de flujo de trabajo global. En lugar de definir un flujo de trabajo único para cada rol de usuario, la solución propuesta plantea un sistema de flujo de trabajo distribuido y basado en áreas, en el que cada área es responsable de definir los roles y actividades asociados a los distintos tipos de usuarios nómadas a los que se desea dar soporte.

Las políticas de seguridad de los flujos de trabajo de cada área pueden heredarse de las áreas superiores, siguiendo la estructura social e institucional de la organización. Como resultado, los flujos de trabajo no están predeterminados y pueden redefinirse y adaptarse dinámicamente a casos específicos y temporales. Las distintas áreas colaboran compartiendo información para garantizar el cumplimiento de los objetivos de los usuarios nómadas, respetando en todo momento las políticas de seguridad de la organización.

Se diseñó y validó con éxito un prototipo de AMS con usuarios finales, que encontraron el sistema utilizable, fácil de usar, intuitivo y sencillo.

8.2

Aportaciones

A partir del trabajo de tesis se publicaron artículos científicos indexados. Cada publicación es en sí misma una aportación al estado del arte de las áreas correspondientes. El marco de trabajo conceptual, Capítulo 3, generó el artículo *“Towards an Interaction Design Framework for IoT Healthcare”*. La propuesta de arquitectura, Capítulo 4, generó el artículo *“Wearable and Pervasive Architecture for Digital Companions in Chronica Disease Care”*, esta publicación ganó el premio *“Best Paper Award of DAPI”*.

2024 Affiliated Conference” en el marco del congreso “26th International Conference on Human-Computer Interaction”, Washington DC, USA, celebrado del 29 de junio al 4 de julio. La propuesta de Sistema de apoyo para pacientes en tránsito en un hospital, Capítulo 6, generó el artículo “Achieve Your Goal Without Dying in The Attempt: Developing an Area-Based Support for Nomadic Work”.

Adicionalmente, se presentan más aportaciones del trabajo de tesis en las subsecciones siguientes:

Marco de trabajo conceptual

En la exploración de las opiniones de los pacientes consultados se obtuvieron importantes hallazgos que confirman la intuición de que para realizar una aplicación en el contexto del cuidado de la salud, es necesario partir de características que hagan que la posibilidad de que el paciente adopte la aplicación sea lo más alta posible. Para alcanzar el objetivo de adopción de sistema, por parte de los pacientes, es necesario partir del Diseño de la Interacción. Esto garantiza que los sistemas sean fáciles de usar y sean efectivos para realizar las tareas funcionales bajo las cuales se diseñaron. La principal aportación del marco de trabajo conceptual es la propuesta de los conceptos: Interfaces de usuario como servicios, Interacciones contextuales conscientes y Gestión de datos centrada en el usuario. El diseño de un sistema basado en estos tres conceptos tendrá una alta posibilidad de que el paciente adopte el sistema y con ello el sistema tenga la oportunidad de adaptarse mejor a cada paciente teniendo un círculo virtuoso de evolución del sistema.

Arquitectura de CoDisPa

En el diseño de la arquitectura, dos elementos clave para la propuesta fueron: el uso de los conceptos derivados del marco de trabajo conceptual y el uso de los resultados de aplicar la técnica del mago de Oz para descubrir la experiencia de usuario anticipada en un grupo de pacientes con diabetes. Adicionalmente, el estudio del estado de la técnica en asistentes virtuales, sistemas distribuidos que usan Internet de las Cosas para el cuidado de la salud, hizo posible que se planteara una arquitectura funcionalmente adecuada y robusta para adaptarse al paciente y permitir el crecimiento y evolución del sistema. La arquitectura del compañero digital probó que es posible integrar dispositivos inteligentes e infraestructura pervasiva en un entorno doméstico para ayudar al paciente, empleando transformadores de audio a texto basados en *transformers*, redes neuronales convolucionales para la identificación de personas, por mencionar algunos. La propuesta también aprovecha la disponibilidad de tecnologías de comunicación entre

los dispositivos tales como: *MQTT*¹, *bluetooth LE*, *WiFi*, entre otros. Por lo tanto, se probó que la arquitectura cuenta con las partes necesarias para atender las necesidades de un paciente con diabetes y aliviarle la sobrecarga de tareas producto de su padecimiento. Debido a la naturaleza de los componentes del sistema, CoDisPa es escalable y se puede integrar a casi cualquier entorno donde un paciente con diabetes habite. Un hallazgo notable es que el compañero digital es capaz de descubrir y comprender las acciones del paciente, normalmente el usuario es quién puede descubrir y comprender al sistema que usa, y con esto logra acercarse al comportamiento de un compañero humano. Los resultados de nuestra propuesta destacan la importancia de considerar a todas las partes interesadas al diseñar sistemas con dispositivos de Internet de las Cosas. Al integrar el marco conceptual al uso de dispositivos y plataformas de Internet de las Cosas existentes, las personas pueden beneficiarse de una experiencia fluida e integrada, lo que puede ayudarles a alcanzar sus objetivos de salud y bienestar.

Sistema de apoyo para el paciente en el tránsito de un hospital

Siguiendo con la construcción de un ambiente que ayude a los pacientes, nace la propuesta de un sistema en el que el paciente salga de su entorno doméstico e incursione en instalaciones médicas, como un hospital, siendo dirigido y ayudado por un sistema que gestiona los recursos y tareas basándose en reglas para alcanzar los objetivos del usuario en tránsito (llamado usuario nómada). Este sistema de ayuda basado en áreas jerárquicas hace aportaciones en aspectos del cuidado de la salud y también del manejo de infraestructura mediante el empleo de recursos computacionales, en particular, sistemas distribuidos y colaborativos. Computacionalmente, el enfoque de gestión inteligente de los recursos que se centran en un usuario nómada es innovador y aporta al estado del arte una arquitectura que puede ser implementada en las organizaciones donde un paciente puede sentirse perdido y agobiado.

8.3 Limitaciones

En relación al marco de trabajo conceptual, la limitante principal fue que al realizar el ejercicio con los usuarios potenciales con un grupo acotado que no necesariamente representa a un segmento amplio de la población de pacientes. Esto plantea la posibilidad de encontrar más conceptos importantes si se realiza el mismo estudio en

¹MQTT es un protocolo de mensajería basado en estándares que se utiliza para la comunicación de dispositivos de Internet de las Cosas

muestras representativas de una población mayor, que reflejen con más precisión las características de los pacientes.

En relación con el desempeño de la implementación del compañero digital, una de las limitantes más importantes es el tiempo de respuesta del procesamiento del lenguaje natural y la comprensión de las frases coloquiales que el paciente es capaz de decir en la interacción con el compañero digital. Otra limitante importante, es que se requiere de configuración especializada para ubicar los dispositivos de escucha en las zonas más apropiadas del hogar del paciente, esto para que el compañero digital pueda ser empleado en un entorno doméstico exitosamente. Adicionalmente, el descubrimiento del uso del sistema, por parte de los pacientes, no se pudo realizar a cabalidad, i.e., hizo falta crear un esquema donde el compañero digital guíe al paciente en el descubrimiento del uso de su compañía y apoyo para que en un tiempo corto el sepa los alcances de las funcionalidades del sistema.

En relación con el sistema de apoyo a pacientes en tránsito en un hospital, la principal limitante es la instalación y configuración del sistema en entornos reales, debido a los múltiples sensores y redes de comunicación necesarias para que los componentes interactúen entre sí de manera adecuada. Otra limitante, es el hecho de que el sistema basado en áreas debe convivir con sistemas de información y sistemas de acceso heredados, y que se encuentran en uso en las organizaciones, donde se haga el despliegue del AMS.

Por último, hay limitaciones, de corte tecnológico, en relación a las plataformas Internet de las Cosas, sistemas de computación ligera y medio de comunicación.

- Los dispositivos *Single Board Computer* no pueden ejecutar los modelos de aprendizaje automático en un tiempo aceptable para tener respuestas fluidas con los pacientes o usuarios. Para lograr tiempos de respuesta aceptables se requieren *GPU* o *Edge Computing Accelerators*, además de optimizar los parámetros y bibliotecas que se emplean para cargar la información en ellos.
- Los microcontroladores tienen un poder de cómputo bajo y es por ello que se usan sólo como clientes. Estos clientes delegan u ordenan acciones a los microservicios, i.e., hacen *offloading*.
- Las comunicaciones de los dispositivos de Internet de las Cosas se encuentran limitadas a las velocidades máximas para cada tecnología con la que se implementa y a la versión de cada una de ellas, e.g., *bluetooth* (v 5.2 o LE), *WiFi* (802.11) o *ZigBee* (802.15.4).

Ha sido muy rico el proceso de investigación de este trabajo de tesis; sin embargo, ha sido sólo el principio de una serie de proyectos que pueden, no sólo aportar al estado del arte, sino lograr desarrollar todo un ecosistema que desarrolle los sistemas ubicuos para el apoyo en temas de salud. En ese sentido presentamos algunas ideas de trabajo futuro.

Marco de trabajo conceptual

Para el marco de trabajo conceptual, aún quedan varias áreas por explorar. Por ejemplo, desarrollar interfaces de usuario personalizadas para diferentes tipos de enfermedades crónicas, como enfermedades cardíacas o trastornos de salud mental. Además, es necesario considerar las implicaciones éticas y de privacidad, especialmente en términos de gestión y compartición de datos. Finalmente, el desarrollo de una herramienta o guía de diseño basada en nuestro marco podría apoyar el diseño y desarrollo de sistemas y ayudar a los profesionales a implementar nuestro marco en la práctica.

Arquitectura de CoDiSPa

Para la arquitectura de CoDiSPa, queda pendiente realizar pruebas con pacientes en un entorno controlado para revisar algunos aspectos, principalmente, el análisis del comportamiento y la retroalimentación de la base de conocimiento de CoDiSPa, así como la validación médica del conocimiento generado. Se pretende desplegar el prototipo distintas ubicaciones o instituciones para mostrar que la presente solución, al ser diseñada de manera modular, es escalable, robusta y que puede adaptarse a otros entornos. Puede estudiarse si es viable hacer que el compañero digital pueda transitar entre distintos ambientes computacionales para lograr acompañar a todas partes a su compañero humano, podría explorarse el enfoque de una mascota digital de los años 80's que vivía en un dispositivo tipo llavero, el *Tamagotchi*, que era un ser "vivo" digital que tenía necesidades y que a partir de acciones de su amo lograba sobrevivir o no. Actualmente existe un proyecto de *software* libre que puede darnos un indicio de cómo es posible lograr un dispositivo móvil que aloje al compañero digital como una mascota digital, se trata del proyecto de auditoría de sistemas denominado *Pwnagotchi*¹ que combina herramientas de seguridad informática y dispositivos de Internet de las Cosas.

¹<https://pwnagotchi.ai/>

Sistema de apoyo para el paciente en el tránsito de un hospital

Para el sistema de apoyo para pacientes en tránsito en un hospital, la implantación en un entorno de trabajo nómada realista puede dar cabida a múltiples aplicaciones. Se podrían añadir facilidades de trabajo en grupo, como un sistema para gestionar la disponibilidad de recursos distribuidos a nivel de usuario y un sistema de reconocimiento facial para detectar y localizar a los usuarios, incluso si se separan temporalmente de sus teléfonos inteligentes. Estas implantaciones crearán un entorno de trabajo avanzado para apoyar el trabajo bien organizado, nómada y cooperativo. Por último, se podría hacer la sustitución de los dispositivos móviles de los usuarios nómadas por credenciales inteligentes, utilizando pantallas de tinta electrónica u otras tecnologías de bajo consumo.

Dispositivos de Internet de las Cosas

Para los dispositivos de Internet de las Cosas se pueden extender las capacidades originales de éstos. En particular, se puede realizar un proceso de ingeniería inversa sobre las pulseras inteligentes, relojes inteligentes, básculas inteligentes y dispositivos basados en microcontroladores para extraer el *firmware*, analizarlo y modificarlo para lograr extender sus capacidades y se comunique directamente con el sistema CoDisPa sin necesidad de interceptar la comunicación de manera auxiliar. Al comprender cómo está compuesto el *firmware* de este tipo de dispositivos se puede proponer un *firmware*, que reemplace el original, que soporte la orientación al compañero digital.

Publicaciones del autor

Las publicaciones producto del trabajo de investigación son las siguientes.

1. G. Monroy-Rodríguez, S. Mendoza, L. M. Sánchez-Adame, I. G. Valdespin-Garcia, y D. Decouchant, «Achieve Your Goal Without Dying in the Attempt: Developing an Area-Based Support for Nomadic Work», en HCI in Business, Government and Organizations, F. Nah y K. Siau, Eds., en Lecture Notes in Computer Science. Cham: Springer Nature Switzerland, 2023, pp. 421-438. [doi:10.1007/978-3-031-35969-9_28](https://doi.org/10.1007/978-3-031-35969-9_28).
2. L. M. Sánchez-Adame, G. Monroy-Rodríguez, S. Mendoza, D. Decouchant, y A. P. Mateos-Papis, «Framework for Ethically Designed Microtransactions in the Metaverse», IEEE Access, vol. 11, pp. 140687-140700, 2023, [doi:10.1109/ACCESS.2023.3341057](https://doi.org/10.1109/ACCESS.2023.3341057).
3. G. Monroy-Rodríguez, L. M. Sánchez-Adame, S. Mendoza, I. G. Valdespin-Garcia, y D. Decouchant, «Towards an Interaction Design Framework for IoT Healthcare Systems», en Distributed, Ambient and Pervasive Interactions, N. A. Streitz y S. Konomi, Eds., en Lecture Notes in Computer Science. Cham: Springer Nature Switzerland, 2023, pp. 91-104. [doi:10.1007/978-3-031-34668-2_7](https://doi.org/10.1007/978-3-031-34668-2_7).
4. G. Monroy-Rodriguez, L. M. Sánchez-Adame, y S. G. Mendoza-Chapa, «Un compañero digital al rescate de los pacientes con diabetes», Avance y Perspectiva, Zona Abierta, vol. 8, n.º 4, dic. 2022. Disponible en: <https://avanceyperspectiva.cinvestav.mx/un-companero-digital-al-rescate-de-los-pacientes-con-diabetes/>
5. G. Monroy-Rodriguez, S.G. Mendoza-Chapa, L.M. Sánchez-Adame, I. Valdespin, y D. Decouchant, «Wearable and Pervasive Architecture for Digital Companions in Chronic Disease Care», en: Streitz, N.A., Konomi, S. (eds) Distributed, Ambient and Pervasive Interactions. HCII 2024. Lecture Notes in Computer Science, vol 14719. Springer, Cham. https://doi.org/10.1007/978-3-031-60012-8_5

 Best Paper Award of DAPI 2024 Affiliated Conference

Bibliografía

- [1] K. García, S. Mayer, A. Ricci, and A. Ciortea, "Proactive Digital Companions in Pervasive Hypermedia Environments," in *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*, Dec. 2020, pp. 54–59.
- [2] "AI: My name is Companion – Digital Companion." [Online]. Available: <https://new.siemens.com/global/en/company/stories/research-technologies/artificial-intelligence/artificial-intelligence-digital-companion.html>
- [3] "Statista - The Statistics Portal." [Online]. Available: <https://www.statista.com/studies-and-reports/>
- [4] C. Lisetti, R. Amini, U. Yasavur, and N. Rishe, "I Can Help You Change! An Empathic Virtual Agent Delivers Behavior Change Health Interventions," *ACM Transactions on Management Information Systems*, vol. 4, no. 4, pp. 19:1–19:28, Dec. 2013. [Online]. Available: <https://doi.org/10.1145/2544103>
- [5] A. S. Miner, N. Shah, K. D. Bullock, B. A. Arnow, J. Bailenson, and J. Hancock, "Key Considerations for Incorporating Conversational AI in Psychotherapy," *Frontiers in Psychiatry*, vol. 10, 2019, publisher: Frontiers. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fpsyt.2019.00746/full>
- [6] P. P. Jayaraman, A. R. M. Forkan, A. Morshed, P. D. Haghighi, and Y.-B. Kang, "Healthcare 4.0: A review of frontiers in digital health," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 2, p. e1350, 2020.
- [7] J. Calvillo-Arbizu, I. Román-Martínez, and J. Reina-Tosina, "Internet of things in health: Requirements, issues, and gaps," *Computer Methods and Programs in Biomedicine*, vol. 208, p. 106231, 2021.
- [8] A. A. Di Sessa, "A principled design for an integrated computational environment," *Human-computer interaction*, vol. 1, no. 1, pp. 1–47, 1985.
- [9] I. S. Candanedo, E. H. Nieves, S. R. González, M. T. S. Martín, and A. G. Briones, "Machine learning predictive model for industry 4.0," in *International Conference on Knowledge Management in Organizations*. Springer, 2018, pp. 501–510.
- [10] H. Oliff and Y. Liu, "Towards industry 4.0 utilizing data-mining techniques: a case study on quality improvement," *Procedia CIRP*, vol. 63, pp. 167–172, 2017.

- [11] G. Li, J. Tan, and S. S. Chaudhry, "Industry 4.0 and big data innovations," 2019.
- [12] F.-E. Bordeleau, E. Mosconi, and L. A. Santa-Eulalia, "Business intelligence in industry 4.0: State of the art and research opportunities," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.
- [13] L. Duan and L. Da Xu, "Data analytics in industry 4.0: A survey," *Information Systems Frontiers*, pp. 1–17, 2021.
- [14] C. Gröger, "Building an industry 4.0 analytics platform," *Datenbank-Spektrum*, vol. 18, no. 1, pp. 5–14, 2018.
- [15] S. Barlow, "What is the best healthcare data warehouse model? comparing enterprise data models, independent data marts, and late-binding™ solutions," *Health Catalyst*, 2017.
- [16] I. Kukhtevich, V. Goryunova, T. Goryunova, and P. Zhilyaev, "Medical decision support systems and semantic technologies in healthcare," *Advances in Economics, Business and Management Research (AEBMR)*, vol. 1, no. 48, pp. 370–375, 2020.
- [17] M. Dastbaz and P. Cochrane, *Industry 4.0 and Engineering for a Sustainable Future*. Springer, 2019.
- [18] P. Esmaeilzadeh, "Use of AI-based tools for healthcare purposes: a survey study from consumers' perspectives," *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, p. 170, Jul. 2020. [Online]. Available: <https://doi.org/10.1186/s12911-020-01191-1>
- [19] P. Chomphuwiset, J. Kim, and P. Pawara, "Multi-disciplinary trends in artificial intelligence."
- [20] I. Lukić, M. Köhler, and E. Kiralj, "Appointment scheduling system in multi doctor/multi services environment," *International journal of electrical and computer engineering systems*, vol. 12, no. 3, pp. 171–176, 2021.
- [21] J. Vilchez-Cornejo, L. Romani, S. Reategui, E. Gomez-Rojas, and C. Silva, "Factores asociados a la realización de actividades de autocuidado en pacientes diabéticos en tres hospitales de ucajali," *Revista de la Facultad de Medicina Humana*, vol. 20, no. 2, pp. 254–260, 2020.
- [22] P. Yang, G. Bi, J. Qi, X. Wang, Y. Yang, and L. Xu, "Multimodal wearable intelligence for dementia care in healthcare 4.0: a survey," *Information Systems Frontiers*, pp. 1–18, 2021.

- [23] "Obesidad y sobrepeso." [Online]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/obesity-and-overweight>
- [24] E. Sindoni, "Design System con enfoque agnóstico," Apr. 2021. [Online]. Available: <https://blog.prototypr.io/design-system-con-enfoque-agn%C3%B3stico-5a826af62b43>
- [25] B. Ganesan, T. Gowda, A. Al-Jumaily, K. N. K. Fong, S. K. Meena, and R. K. Y. Tong, "Ambient assisted living technologies for older adults with cognitive and physical impairments: A review," Jan. 2019. [Online]. Available: <https://opus.lib.uts.edu.au/handle/10453/137559>
- [26] S. P. Patro, N. Padhy, and D. Chiranjivi, "Ambient assisted living predictive model for cardiovascular disease prediction using supervised learning," *Evolutionary Intelligence*, vol. 14, no. 2, pp. 941–969, Jun. 2021. [Online]. Available: <https://doi.org/10.1007/s12065-020-00484-8>
- [27] M. A. Qureshi, K. N. Qureshi, G. Jeon, and F. Piccialli, "Deep learning-based ambient assisted living for self-management of cardiovascular conditions," *Neural Computing and Applications*, vol. 34, no. 13, pp. 10 449–10 467, Jul. 2022. [Online]. Available: <https://doi.org/10.1007/s00521-020-05678-w>
- [28] M. Anastasiadou, A. Alexiadis, E. Polychronidou, K. Votis, and D. Tzovaras, "A prototype educational virtual assistant for diabetes management," in *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*. IEEE, 2020, pp. 999–1004.
- [29] S. Ahmed, M. Qaosar, W. Rizka, Y. Sholikah, and Morimoto, *Early Dementia Detection through Conversations to Virtual Personal Assistant*, Mar. 2018.
- [30] D. DeVault, R. Artstein, G. Benn, T. Dey, K. Georgila, J. Gratch, A. Hartholt, M. Lhommet, G. Lucas, S. Marsella, F. Morbini, A. Nazarian, S. Scherer, G. Stratou, A. Suri, D. Traum, R. Wood, Y. Xu, A. Rizzo, and L.-P. Morency, "SimSensei Kiosk: A Virtual Human Interviewer for Healthcare Decision Support," p. 8.
- [31] D. Sonntag, "Kognit: Intelligent Cognitive Enhancement Technology by Cognitive Models and Mixed Reality for Dementia Patients," p. 6.
- [32] "Managing Chronic Conditions with a Smartphone-based Conversational Virtual Agent | Proceedings of the 18th International Conference on Intelligent Virtual Agents." [Online]. Available: <https://dl.acm.org/doi/abs/10.1145/3267851.3267908>

- [33] M. Li, J. Hou, and H. Zhang, "A TALE OF TWO VIRTUAL ADVISORS: AN EMPIRICAL STUDY INVESTIGATING THE EMPOWERMENT EFFECT OF MOBILE MENTAL-HEALTH ADVISORY SYSTEMS ON EMERGENCY RESCUERS," p. 18.
- [34] J. A. Rincon, A. Costa, P. Novais, V. Julian, and C. Carrascosa, "A new emotional robot assistant that facilitates human interaction and persuasion," *Knowledge and Information Systems*, vol. 60, no. 1, pp. 363–383, Jul. 2019. [Online]. Available: <https://doi.org/10.1007/s10115-018-1231-9>
- [35] A. Ghandeharioun, D. McDuff, M. Czerwinski, and K. Rowan, "Emma: An emotionally intelligent personal assistant for improving wellbeing," *arXiv preprint arXiv:1812.11423*, 2018.
- [36] I. Monarca, F. L. Cibrian, A. Mendoza, G. Hayes, and M. Tentori, "Why doesn't the conversational agent understand me? a language analysis of children speech," in *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers*, 2020, pp. 90–93.
- [37] L. Laranjo, A. G. Dunn, H. L. Tong, A. B. Kocaballi, J. Chen, R. Bashir, D. Surian, B. Gallego, F. Magrabi, A. Y. Lau *et al.*, "Conversational agents in healthcare: a systematic review," *Journal of the American Medical Informatics Association*, vol. 25, no. 9, pp. 1248–1258, 2018.
- [38] A. Olthof, P. van Ooijen, and L. Cornelissen, "Deep learning-based natural language processing in radiology: The impact of report complexity, disease prevalence, dataset size, and algorithm type on model performance," *Journal of Medical Systems*, vol. 45, no. 10, pp. 1–16, 2021.
- [39] A. Feder, K. A. Keith, E. Manzoor, R. Pryzant, D. Sridhar, Z. Wood-Doughty, J. Eisenstein, J. Grimmer, R. Reichart, M. E. Roberts *et al.*, "Causal inference in natural language processing: Estimation, prediction, interpretation and beyond," *arXiv preprint arXiv:2109.00725*, 2021.
- [40] A. Darcy, J. Daniels, D. Salinger, P. Wicks, A. Robinson *et al.*, "Evidence of human-level bonds established with a digital conversational agent: Cross-sectional, retrospective observational study," *JMIR Formative Research*, vol. 5, no. 5, p. e27868, 2021.
- [41] M. E. Auer and T. Tsiatsos, *Internet of Things, Infrastructures and Mobile Applications: Proceedings of the 13th IMCL Conference*. Springer Nature, 2020, vol. 1192.

- [42] T. Cohen Rodrigues, T. Reijnders, D. de Buissonjé, P. Santhanam, T. Kowatsch, V. Janssen, R. Kraaijenhagen, D. Atsma, and A. Evers, "Human cues in self-help lifestyle interventions: an experimental field study," *Journal of Medical Internet Research (JMIR) Preprints*, no. 29/04/2021: 30057, 2021.
- [43] T. Ivaşcu and V. Negru, "Activity-aware vital signmonitoring based on a multi-agent architecture," *Sensors*, vol. 21, no. 12, p. 4181, 2021.
- [44] N. Sánchez-Pi and L. Martí, "Towards a green ai: evolutionary solutions for an ecologically viable artificial intelligence," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2021, pp. 1135–1140.
- [45] D. Cleres, F. Rassouli, M. Brutsche, T. Kowatsch, and F. Barata, "Lena: a voice-based conversational agent for remote patient monitoring in chronic obstructive pulmonary disease," in *26th ACM Conference on Intelligent User Interfaces (ACM IUI 2021)*, 2021.
- [46] A. Babu and B. Gupta, "Heart rate monitoring of anesthesiology residents during the airway management of covid-19 suspect patients: An observational study," *Indian Journal of Critical Care Medicine*, pp. S50–S50, 2021.
- [47] L. Chhiba, A. Marzak, and M. Sidqui, "Quality attributes for evaluating iot health-care systems," in *Innovations in Smart Cities Applications Volume 5*, M. Ben Ahmed, A. A. Boudhir, İ. R. Karas, V. Jain, and S. Mellouli, Eds. Cham: Springer International Publishing, 2022, pp. 495–505.
- [48] M. I. Hossain, A. F. Yusof, A. R. C. Hussin, N. A. lahad, and A. S. Sadiq, "Factors influencing adoption model of continuous glucose monitoring devices for internet of things healthcare," *Internet of Things*, vol. 15, p. 100353, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2542660520301840>
- [49] P.-Y. Yen and S. Bakken, "Review of health information technology usability study methodologies," *Journal of the American Medical Informatics Association*, vol. 19, no. 3, pp. 413–422, 08 2011. [Online]. Available: <https://doi.org/10.1136/amiajnl-2010-000020>
- [50] R. Sneesl, Y. Y. Jusoh, M. A. Jabar, and S. Abdullah, "Revising technology adoption factors for iot-based smart campuses: A systematic review," *Sustainability*, vol. 14, no. 8, p. 4840, Apr 2022. [Online]. Available: <http://dx.doi.org/10.3390/su14084840>

- [51] R. De Michele and M. Furini, "Iot healthcare: Benefits, issues and challenges," in *Proceedings of the 5th EAI International Conference on Smart Objects and Technologies for Social Good*, ser. GoodTechs '19. Valencia, Spain: Association for Computing Machinery, 2019, p. 160–164. [Online]. Available: <https://doi.org/10.1145/3342428.3342693>
- [52] —, "Iot healthcare: Benefits, issues and challenges," in *Proceedings of the 5th EAI International Conference on Smart Objects and Technologies for Social Good*, ser. GoodTechs '19. Valencia, Spain: Association for Computing Machinery, 2019, p. 160–164.
- [53] V. Kashyap, A. Kumar, A. Kumar, and Y.-C. Hu, "A systematic survey on fog and iot driven healthcare: Open challenges and research issues," *Electronics*, vol. 11, no. 17, 2022.
- [54] S. Selvaraj and S. Sundaravaradhan, "Challenges and opportunities in iot healthcare systems: a systematic review," *SN Applied Sciences*, vol. 2, no. 1, p. 139, Dec 2019.
- [55] B. Pradhan, S. Bhattacharyya, and K. Pal, "Iot-based applications in healthcare devices," *Journal of Healthcare Engineering*, vol. 2021, p. 6632599, Mar 2021.
- [56] Syagnik (Sy) Banerjee, T. Hemphill, and P. Longstreet, "Wearable devices and healthcare: Data sharing and privacy," *The Information Society*, vol. 34, no. 1, pp. 49–57, 2018.
- [57] K. Riad, R. Hamza, and H. Yan, "Sensitive and energetic iot access control for managing cloud electronic health records," *IEEE Access*, vol. 7, pp. 86 384–86 393, 2019.
- [58] Y. Suryandari, "Survei iot healthcare device," *Jurnal Sistem Cerdas*, vol. 3, no. 2, pp. 153–164, Aug. 2020.
- [59] H. Yu and Z. Zhou, "Optimization of iot-based artificial intelligence assisted telemedicine health analysis system," *IEEE Access*, vol. 9, pp. 85 034–85 048, 2021.
- [60] F. Lemoine, T. Aubonnet, and N. Simoni, "Iot composition based on self-controlled services," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 5167–5186, Nov 2020.

- [61] M. Wiberg, "Addressing iot: Towards material-centered interaction design," in *Human-Computer Interaction. Theories, Methods, and Human Issues*, M. Kurosu, Ed. Cham: Springer International Publishing, 2018, pp. 198–207.
- [62] E. Sari and A. Tedjasaputra, "Designing valuable products with design sprint," in *Human-Computer Interaction – INTERACT 2017*, R. Bernhaupt, G. Dalvi, A. Joshi, D. K. Balkrishan, J. O'Neill, and M. Winckler, Eds. Cham: Springer International Publishing, 2017, pp. 391–394.
- [63] J. Knapp, J. Zeratsky, and B. Kowitz, *Sprint: How to solve big problems and test new ideas in just five days*. Simon and Schuster, 2016.
- [64] W. J. W. Keijzer-Broers and M. de Reuver, "Applying agile design sprint methods in action design research: Prototyping a health and wellbeing platform," in *Tackling Society's Grand Challenges with Design Science*, J. Parsons, T. Tuunanen, J. Venable, B. Donnellan, M. Helfert, and J. Kenneally, Eds. Cham: Springer International Publishing, 2016, pp. 68–80.
- [65] H. Southall, M. Marmion, and A. Davies, *Adapting Jake Knapp's Design Sprint Approach for AR/VR Applications in Digital Heritage*. Cham: Springer International Publishing, 2019, pp. 59–70. [Online]. Available: https://doi.org/10.1007/978-3-030-06246-0_5
- [66] M. Cohn, *User stories applied: For agile software development*. Addison-Wesley Professional, 2004.
- [67] C. Gutwin and S. Greenberg, "The mechanics of collaboration: developing low cost usability evaluation methods for shared workspaces," in *Proceedings IEEE 9th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE 2000)*, Gaithersburg, MD, USA, 2000, pp. 98–103.
- [68] I. Campos-Nonato, M. Ramírez-Villalobos, A. Flores-Coria, A. Valdez, and E. Monterrubio-Flores, "Prevalence of previously diagnosed diabetes and glycemic control strategies in mexican adults: Ensanut-2016," *PLOS ONE*, vol. 15, no. 4, pp. 1–11, 04 2020. [Online]. Available: <https://doi.org/10.1371/journal.pone.0230752>
- [69] I. G. Cohen and M. M. Mello, "HIPAA and Protecting Health Information in the 21st Century," *JAMA*, vol. 320, no. 3, pp. 231–232, 07 2018.

- [70] ardalis, “Common web application architectures - .NET,” Mar. 2023. [Online]. Available: <https://learn.microsoft.com/en-us/dotnet/architecture/modern-web-apps-azure/common-web-application-architectures>
- [71] “Modern Web App Architecture: Types, Tips and Diagrams,” May 2023, section: Digital Strategy. [Online]. Available: <https://www.digiteum.com/web-application-architecture/>
- [72] “Stack Overflow Developer Survey 2023.” [Online]. Available: https://survey.stackoverflow.co/2023/?utm_source=social-share&utm_medium=social&utm_campaign=dev-survey-2023
- [73] David, “Web Application Architecture in 2024: Best Practices & Trends - GraphicSprings,” Mar. 2014, section: General. [Online]. Available: <https://graphicsprings.com/blog/view/web-application-architecture-in-2024/>
- [74] P. Di Francesco, P. Lago, and I. Malavolta, “Architecting with microservices: A systematic mapping study,” *Journal of Systems and Software*, vol. 150, pp. 77–97, Apr. 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0164121219300019>
- [75] “Microservices Architecture.” [Online]. Available: <https://publications.opengroup.org/w169>
- [76] M. Lamothe, Y.-G. Guéhéneuc, and W. Shang, “A systematic review of api evolution literature,” *ACM Computing Surveys (CSUR)*, vol. 54, no. 8, pp. 1–36, 2021.
- [77] R. T. Fielding, “Architectural styles and the design of network-based software architectures, university of california, irvine, 2000,” *Google Scholar Google Scholar Digital Library Digital Library*, 2000.
- [78] H. B. Christensen, “Teaching Distributed Programming – Revisiting the Broker Pattern,” in *Proceedings of the 5th European Conference on Software Engineering Education*, ser. ECSEE ’23. New York, NY, USA: Association for Computing Machinery, Jun. 2023, pp. 162–168. [Online]. Available: <https://doi.org/10.1145/3593663.3593674>
- [79] T. Schopf, K. Arabi, and F. Matthes, “Exploring the landscape of natural language processing research,” *arXiv preprint arXiv:2307.10652*, 2023.
- [80] G. Hohpe and B. Woolf, “Enterprise integration patterns,” in *9th conference on pattern language of programs*. Citeseer, 2002, pp. 1–9.

- [81] D. S. Choi, J. Park, M. Loeser, and K. Seo, "Improving counseling effectiveness with virtual counselors through nonverbal compassion involving eye contact, facial mimicry, and head-nodding," *Scientific Reports*, vol. 14, no. 1, p. 506, Jan 2024.
- [82] M. Hassenzahl, "The effect of perceived hedonic quality on product appealiness," *International Journal of Human-Computer Interaction*, vol. 13, no. 4, pp. 481–499, 2001.
- [83] E. L.-C. Law, V. Roto, M. Hassenzahl, A. P. Vermeeren, and J. Kort, "Understanding, scoping and defining user experience: a survey approach," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '09. Boston, MA, USA: Association for Computing Machinery, 2009, p. 719–728.
- [84] Y. Lukic, C. Vogt, O. Dürr, and T. Stadelmann, "Speaker identification and clustering using convolutional neural networks," in *2016 IEEE 26th international workshop on machine learning for signal processing (MLSP)*. IEEE, 2016, pp. 1–6.
- [85] I. G. Valdespin Garcia, "Sistema de reconocimiento de voz y del habla multi-fuente orientado a compañeros digitales," Master's thesis, Centro de Investigación y de Estudios Avanzados del IPN, 2023. [Online]. Available: <https://www.cs.cinvestav.mx/tesisgraduados/2023/resumenIvanValdesp%C3%ACn.html>
- [86] A. Carruthers, *Role-Based Access Control (RBAC)*. Berkeley, CA: Apress, 2022, pp. 123–149. [Online]. Available: https://doi.org/10.1007/978-1-4842-8593-0_5
- [87] K. Vijayalakshmi and V. Jayalakshmi, "A study on current research and challenges in attribute-based access control model," in *Intelligent Data Communication Technologies and Internet of Things*, D. J. Hemanth, D. Pelusi, and C. Vuppapapati, Eds. Singapore: Springer Nature Singapore, 2022, pp. 17–31.
- [88] A. K. Y. S. Mohamed, D. Auer, D. Hofer, and J. Küng, "A systematic literature review for authorization and access control: definitions, strategies and models," *International Journal of Web Information Systems*, Jan 2022. [Online]. Available: <https://doi.org/10.1108/IJWIS-04-2022-0077>
- [89] S. M. Chandran and J. B. D. Joshi, "Lot-rbac: A location and time-based rbac model," in *Web Information Systems Engineering – WISE 2005*, A. H. H. Ngu, M. Kitsuregawa, E. J. Neuhold, J.-Y. Chung, and Q. Z. Sheng, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 361–375.

- [90] F. Cuppens and A. Mieke, "Modelling contexts in the or-bac model," in *Proceedings of the 19th Annual Computer Security Applications Conference*, 2003, pp. 416–425.
- [91] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, "Proposed nist standard for role-based access control," *ACM Trans. Inf. Syst. Secur.*, vol. 4, no. 3, p. 224–274, aug 2001. [Online]. Available: <https://doi.org/10.1145/501978.501980>
- [92] A. Masoumzadeh, H. van der Laan, and A. Dercksen, "Bluesky: Physical access control: Characteristics, challenges, and research opportunities," in *Proceedings of the 27th ACM on Symposium on Access Control Models and Technologies*, ser. SACMAT '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 163–172. [Online]. Available: <https://doi.org/10.1145/3532105.3535019>
- [93] J. Kim and N. Park, "Role-based access control video surveillance mechanism modeling in smart contract environment," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 4, p. e4227, 2022, e4227 ETT-20-0572.R2. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.4227>
- [94] J. van der Laan, "Incremental verification of physical access control systems," January 2021. [Online]. Available: <http://essay.utwente.nl/85634/>
- [95] Y. Cao, Z. Huang, Y. Yu, C. Ke, and Z. Wang, "A topology and risk-aware access control framework for cyber-physical space," *Frontiers of Computer Science*, vol. 14, 2020.
- [96] L. Pasquale, C. Ghezzi, E. Pasi, C. Tsigkanos, M. Boubekeur, B. Florentino-Liano, T. Hadzic, and B. Nuseibeh, "Topology-aware access control of smart spaces," *Computer*, vol. 50, no. 7, pp. 54–63, 2017.
- [97] A. Ben Fadhel, D. Bianculli, L. Briand, and B. Hourte, "A model-driven approach to representing and checking rbac contextual policies," in *Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy*, ser. CODASPY '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 243–253. [Online]. Available: <https://doi.org/10.1145/2857705.2857709>
- [98] Y. K. R. Sánchez, S. A. Demurjian, J. C. Conover, T. Agresta, X. Shao, and M. Diamond, "Role-based access control for mobile computing and applications," *Information Diffusion Management and Knowledge Sharing*, 2020.

- [99] E. Bertino, P. A. Bonatti, and E. Ferrari, "Trbac: A temporal role-based access control model," *ACM Trans. Inf. Syst. Secur.*, vol. 4, no. 3, p. 191–233, aug 2001. [Online]. Available: <https://doi.org/10.1145/501978.501979>
- [100] Y. Wang, Y. Yang, B. Wang, Q. Ran, and X. Ju, "Research on improved access control model based on t-rbac," *Journal of Physics: Conference Series*, vol. 1453, no. 1, p. 012011, jan 2020. [Online]. Available: <https://dx.doi.org/10.1088/1742-6596/1453/1/012011>
- [101] J. Joshi, E. Bertino, U. Latif, and A. Ghafoor, "A generalized temporal role-based access control model," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 1, pp. 4–23, 2005.
- [102] F. Hansen and V. Oleshchuk, "Srbac: A spatial role-based access control model for mobile systems," in *Proceedings of the 7th Nordic Workshop on Secure IT Systems (NORDSEC'03)*, 2003, pp. 129–141.
- [103] Z. Zou, C. Chen, S. Ju, and J. Chen, "The research for spatial role-based access control model," in *Computational Science and Its Applications – ICCSA 2010*, D. Taniar, O. Gervasi, B. Murgante, E. Pardede, and B. O. Apduhan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 296–308.
- [104] M. L. Damiani, E. Bertino, B. Catania, and P. Perlasca, "Geo-rbac: A spatially aware rbac," *ACM Trans. Inf. Syst. Secur.*, vol. 10, no. 1, p. 2–es, feb 2007. [Online]. Available: <https://doi.org/10.1145/1210263.1210265>
- [105] C. Bertolissi and M. Fernandez, "Time and location based services with access control," in *2008 New Technologies, Mobility and Security*, 2008, pp. 1–6.
- [106] K. Guesmia and N. Boustia, "Orbac from access control model to access usage model," *Applied Intelligence*, vol. 48, no. 8, p. 1996–2016, aug 2018. [Online]. Available: <https://doi.org/10.1007/s10489-017-1064-3>
- [107] M. Dekker, J. Crampton, and S. Etalle, "Rbac administration in distributed systems," in *Proceedings of the 13th ACM Symposium on Access Control Models and Technologies*, ser. SACMAT '08. Estes Park, CO, USA: Association for Computing Machinery, 2008, p. 93–102. [Online]. Available: <https://doi.org/10.1145/1377836.1377852>
- [108] E. Freudenthal, T. Pesin, L. Port, E. Keenan, and V. Karamcheti, "drbac: distributed role-based access control for dynamic coalition environments," in *Proceedings*

- 22nd International Conference on Distributed Computing Systems, 2002, pp. 411–420.
- [109] E. Bertino, E. Ferrari, and V. Atluri, “The specification and enforcement of authorization constraints in workflow management systems,” *ACM Trans. Inf. Syst. Secur.*, vol. 2, no. 1, p. 65–104, feb 1999. [Online]. Available: <https://doi.org/10.1145/300830.300837>
 - [110] S. Oh and S. Park, “Task-role based access control (t-rbac): An improved access control model for enterprise environment,” in *Database and Expert Systems Applications*, M. Ibrahim, J. Küng, and N. Revell, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 264–273.
 - [111] J. Wainer, P. Barthelmeß, and A. Kumar, “W-rbac — a workflow security model incorporating controlled overriding of constraints,” *International Journal of Cooperative Information Systems*, vol. 12, no. 04, pp. 455–485, 2003. [Online]. Available: <https://doi.org/10.1142/S0218843003000814>
 - [112] Y. Sun, X. Meng, S. Liu, and P. Pan, “Flexible workflow incorporated with rbac,” in *Computer Supported Cooperative Work in Design II*, W.-m. Shen, K.-M. Chao, Z. Lin, J.-P. A. Barthès, and A. James, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 525–534.
 - [113] M. Leitner, S. Rinderle-Ma, and J. Mangler, “Aw-rbac: Access control in adaptive workflow systems,” in *2011 Sixth International Conference on Availability, Reliability and Security*, 2011, pp. 27–34.
 - [114] A. Maurino and S. Modafferi, “Workflow management in mobile environments,” in *Ubiquitous Mobile Information and Collaboration Systems*, L. Baresi, S. Dustdar, H. C. Gall, and M. Matera, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 83–95.
 - [115] J. Han, Y. Cho, E. Kim, and J. Choi, “A ubiquitous workflow service framework,” in *Computational Science and Its Applications - ICCSA 2006*, M. L. Gavrilova, O. Gervasi, V. Kumar, C. J. K. Tan, D. Taniar, A. Laganá, Y. Mun, and H. Choo, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 30–39.
 - [116] Y. Cao, Y. Ping, S. Tao, Y. Chen, and Y. Zhu, “Specification and adaptive verification of access control policy for cyber-physical-social spaces,” *Computers & Security*, vol. 114, p. 102579, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S016740482100403X>

- [117] Y. Cao, Z. Huang, Y. Yu, C. Ke, and Z. Wang, "A topology and risk-aware access control framework for cyber-physical space," *Frontiers of Computer Science*, vol. 14, no. 4, p. 144805, Jan 2020. [Online]. Available: <https://doi.org/10.1007/s11704-019-8454-0>
- [118] C. Tsigkanos, L. Pasquale, C. Ghezzi, and B. Nuseibeh, "Ariadne: Topology aware adaptive security for cyber-physical systems," in *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, vol. 2, 2015, pp. 729–732.
- [119] A. S. Rao and M. P. Georgeff, "Modeling rational agents within a bdi-architecture," in *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, ser. KR'91. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1991, p. 473–484.
- [120] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Human Mental Workload*, ser. Advances in Psychology, P. A. Hancock and N. Meshkati, Eds. North-Holland, 1988, vol. 52, pp. 139–183. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0166411508623869>
- [121] B. Shneiderman, "The future of interactive systems and the emergence of direct manipulation," *Behaviour & Information Technology*, vol. 1, no. 3, pp. 237–256, 1982.
- [122] W. E. Hick, "On the rate of gain of information," *Quarterly Journal of experimental psychology*, vol. 4, no. 1, pp. 11–26, 1952.
- [123] U. Saad, U. Afzal, A. El-Issawi, and M. Eid, "A model to measure QoE for virtual personal assistant," *Multimedia Tools and Applications*, vol. 76, no. 10, pp. 12 517–12 537, May 2017. [Online]. Available: <https://doi.org/10.1007/s11042-016-3650-5>
- [124] N. Carbonell, "Ambient multimodality: towards advancing computer accessibility and assisted living," *Universal Access in the Information Society*, vol. 5, no. 1, pp. 96–104, Jun. 2006. [Online]. Available: <https://doi.org/10.1007/s10209-006-0027-y>
- [125] S. M. Preum, S. Munir, M. Ma, M. S. Yasar, D. J. Stone, R. Williams, H. Alemzadeh, and J. A. Stankovic, "A review of cognitive assistants for healthcare: Trends, prospects, and future directions," *ACM Computing Surveys (CSUR)*, vol. 53, no. 6, pp. 1–37, 2021.
- [126] D. Norman, *The design of everyday things: Revised and expanded edition*. Basic books, 2013.

- [127] M. Weiser, "The computer for the 21st century," *Scientific american*, vol. 265, no. 3, pp. 94–104, 1991.