UNIVERSIDAD DE CHILE
FACULTAD DE MEDICINA
ESCUELA DE POSTGRADO

Lineamientos de Seguridad y Usabilidad en el Desarrollo de Software en Salud de la Red de Salud Digital de las Universidades del Estado (RSDUE)

RICHARD ALEXANDER VON MOLTKE NECOCHEA

AFE PARA OPTAR AL GRADO DE MAGISTER EN INFORMÁTICA MÉDICA

Director de AFE: Prof Dr Steffen Härtel
Co-Directora de AFE: MSc Macarena Molina

2025

Índice

Resumen

Este trabajo desarrolla una propuesta de instrumentos para enfrentar los desafíos de calidad en el desarrollo de software en salud, con énfasis en las dimensiones de seguridad y usabilidad. El estudio se contextualiza en los prototipos del Gestor de Casos Clínicos “GeCCos” y del Registro Clínico Electrónico “U-Salud”, diseñados en el marco de la Red de Salud Digital de las Universidades del Estado (RSDUE). Ante la ausencia de lineamientos prácticos adaptados a entornos académicos con recursos limitados, se diseñaron instrumentos aplicables y comprensibles por equipos sin especialización en ciberseguridad ni diseño centrado en el usuario.

La metodología incluyó una revisión sistemática de literatura técnica y normativa nacional e internacional, la construcción de lineamientos de seguridad en formato de *checklist* y de usabilidad como guía de buenas prácticas, el análisis comparativo de instrumentos de evaluación de usabilidad, y la validación de los instrumentos propuestos mediante la retroalimentación de expertos, desarrolladores y usuarios clave.

Los resultados evidenciaron una alta valoración del *checklist* de seguridad por parte de los equipos de desarrollo (76,9% lo consideró útil o muy útil), destacando su claridad, aplicabilidad y valor formativo. La guía de usabilidad, si bien fue comprendida, presentó menor impacto práctico, lo que señaló la necesidad de una mayor contextualización. Adicionalmente, se realizó una comparación estructural con sellos de calidad asociados a certificaciones nacionales, y se obtuvo una alta concordancia interpretativa entre desarrolladores en la comprensión del *checklist* (89%), validando su precisión terminológica y funcional.

Se concluye que los instrumentos desarrollados permiten traducir estándares complejos en herramientas accionables para contextos emergentes, fortaleciendo la incorporación de criterios de calidad desde etapas tempranas del desarrollo. Más allá de su valor evaluativo, estos lineamientos actúan como recursos pedagógicos y promotores de una cultura de calidad en el desarrollo de software en salud digital.

Abstract

This work presents a set of instruments designed to address quality challenges in health software development, with a focus on the dimensions of security and usability. The study is situated within the context of two prototypes, the Clinical Case Manager “GeCCos” and the Electronic Health Record System “U-Salud”, both developed by the Digital Health Network of State Universities in Chile (RSDUE). In response to the lack of practical guidelines tailored to academic environments with limited resources, the project proposes tools that are applicable and understandable by teams with no formal training in cybersecurity or user-centered design.

The methodology included a systematic review of relevant technical literature and national and international regulations; the construction of a security checklist and a usability guide based on good practices; a comparative analysis of usability evaluation instruments; and the validation of the proposed tools through feedback from experts, developers, and key users.

The results showed a high level of acceptance of the security checklist by development teams (76.9% found it useful or very useful), emphasizing its clarity, applicability, and educational value. While the usability guide was generally understood, it had less practical impact, indicating a need for greater contextual adaptation. Additionally, a structural comparison was made with national certification standards, and a high level of interpretative agreement among developers (89%) validated the checklist’s terminological and functional precision.

In conclusion, the instruments developed in this study enable the translation of complex standards into actionable tools for emerging contexts, supporting the integration of quality criteria from early development stages. Beyond their evaluative role, these guidelines serve as educational resources that promote a culture of quality in the development of digital health software.

1. Introducción

El avance sostenido de las Tecnologías de la Información y Comunicación (TIC) ha transformado múltiples sectores de la sociedad, siendo el área de la salud uno de los más beneficiados de los avances tecnológicos de las últimas décadas (Deloitte; 2015). Esta transformación digital ha dado lugar a una redefinición de procesos tanto clínicos como administrativos, consolidando una nueva era caracterizada por la informatización de los flujos de trabajo, la disponibilidad inmediata de información clínica de los pacientes y la capacidad de tomar decisiones basadas en datos o apoyadas en bases de conocimiento (World Bank, 2024). En este escenario, la informática médica se presenta como la disciplina que articula los conocimientos provenientes de la medicina y de las ingenierías para el diseño e implementación de soluciones tecnológicas orientadas a la mejora de la salud tanto individual como colectiva (Kulikowski et al, 2012)

El desarrollo de software en salud debe responder a exigencias complejas relacionadas con la seguridad del paciente, la integridad de los datos y la eficiencia de los procesos clínicos impactados por los sistemas. En este sentido, la calidad se define como el grado en que un producto satisface los requisitos funcionales y no funcionales establecidos, tal como lo plantea la norma ISO 9000 (ISO, 2015). Para poder evaluar esta calidad se han desarrollado numerosos modelos de evaluación de calidad de software (Ronchieri et al, 2023). Una de las herramientas más completas y reconocida corresponde a la norma ISO 25010:2023, la cual establece nueve características principales y diversas subcaracterísticas asociadas, lo que permite un análisis detallado y holístico del desempeño de un sistema de software (ISO, 2023) (*Tabla 1*).

Característica de calidad	Subcaracterísticas
Adecuación funcional	Completitud funcional, Corrección funcional, Pertinencia funcional

Característica de calidad	Subcaracterísticas
Eficiencia de desempeño	Comportamiento temporal, Utilización de recursos, Capacidad
Compatibilidad	Coexistencia, Interoperabilidad
Capacidad de interacción	Reconocibilidad de adecuación, Aprendizabilidad, Protección frente a errores de usuario, Involucración del usuario, Inclusividad, Asistencia al usuario, Auto-descriptividad
Fiabilidad	Ausencia de fallos, Disponibilidad, Tolerancia a fallos, Recuperabilidad
Seguridad	Confidencialidad, Integridad, No repudio, Responsabilidad, Autenticidad, Resistencia
Mantenibilidad	Modularidad, Reusabilidad, Analizabilidad, Capacidad de ser modificado, Capacidad de ser probado
Flexibilidad	Adaptabilidad, Escalabilidad, Instalabilidad, Reemplazabilidad
Protección	Restricción operativa, Identificación de riesgos, Protección ante fallos, Advertencia de peligro, Integración segura

Tabla 1: Características de la Norma ISO 25010:2023 para la evaluación de la calidad de software. El desglose en subcaracterísticas permite una evaluación exhaustiva del software en las diferentes dimensiones

Dentro de este modelo, las dimensiones de seguridad y usabilidad adquieren un rol protagónico en el ámbito de la salud (ISO, 2018). La primera, referida a la capacidad del software para proteger la información y los datos frente a accesos no autorizados, alteraciones o pérdidas, es esencial para resguardar la confidencialidad de los datos clínicos, dando cumplimiento a normativas de protección de datos personales y garantizar la confianza de los usuarios. La segunda, relativa al grado en que un sistema puede ser utilizado de manera eficaz, eficiente y satisfactoria por usuarios específicos en un contexto determinado, resulta crucial para asegurar la adopción de las tecnologías por parte del personal clínico y administrativo. Una baja usabilidad puede

traducirse en errores en el registro de la atención clínica, demoras innecesarias y rechazo a los sistemas, incluso cuando estos sean técnicamente correctos.

En Chile, el Ministerio de Salud (MINSAL) ha sido un actor central en este proceso de adopción tecnológica mediante la implementación de iniciativas como la Estrategia SIDRA (Sistemas de Información de la Red Asistencial), la cual, desde el año 2008, ha fomentado la adopción progresiva de soluciones tecnológicas en los distintos niveles del sistema público de salud (MINSAL, 2006/2015; SSMC, 2018). Esta estrategia, que considera la informatización de procesos clínicos y administrativos, se ha acompañado del despliegue de un modelo de arquitectura de información sectorial, la definición de estándares de interoperabilidad y la generación de condiciones para la integración de servicios a nivel nacional que se encuentran en constante mejora e implementación, lo cual se ha visto reforzado a través de la promulgación de la ley 21668 (MINSAL, 2024).

Apoyando el quehacer del Ministerio de Salud se encuentra el Centro Nacional en Sistemas de Información en Salud (CENS), una corporación sin fines de lucro formada por 5 universidades nacionales con financiamiento de la Agencia Nacional de Investigación y Desarrollo (ANID) con la finalidad de contribuir al fomento y adopción de tecnologías de información en salud (CENS, 2016). Esta institución ha desarrollado una propuesta de certificación de calidad de software en salud basada en sellos, mediante la cual busca promover el cumplimiento de buenas prácticas en desarrollos informáticos para salud incorporando criterios alineados con las normas ISO y consideran diferentes dimensiones de calidad según el objetivo del sello a aplicar. Entre las certificaciones disponibles se encuentran: el Sello de Registro Clínico Electrónico enfocado en la usabilidad y la seguridad, el Sello de Calidad de Software en Salud que evalúa las características y buenas prácticas técnicas para toda herramienta basada en tecnologías de la información en salud y el Sello de Calidad en Telemedicina que busca evaluar y medir de forma estandarizada los productos de software utilizados para la prestación de servicios de Telemedicina.

1.1 Contexto del desarrollo de la RSDUE

El Ministerio de Educación de Chile, a través del Plan de Fortalecimiento de las Universidades del Estado (PFUE), ha fomentado la consolidación de capacidades institucionales para abordar los desafíos de la salud digital, creando en 2021 la Red de Salud Digital de las Universidades del Estado (RSDUE) (PFUE, 2021). Esta red, conformada actualmente por 14 universidades estatales y con la colaboración de instituciones nacionales e internacionales como HL7 Chile[1], I-Dair[2], AtrysHealth[3] y REUNA[4], tiene como propósito promover la salud digital, integrando dimensiones como la salud mental y la implementación de tecnologías de información en los procesos formativos de los futuros profesionales de la salud y facilitar su aplicación en contextos de simulación y atención clínica a población estudiantil (RSDUE, 2021).

Figura 1: Estructura organizativa de la RSDUE. Se destacan 4 mesas de trabajo que componen la red, cada una con enfoque específico. Esto facilita la colaboración interdisciplinaria y la coordinación efectiva entre las universidades que la conforman. (Fuente: rsdue.cl)

La RSDUE se organiza en torno a 4 mesas de trabajo, orientándose cada una a uno de los 4 objetivos de la red (Figura 1), enfocándose a través del tercer objetivo al di-

[1] https://hl7chile.cl/
[2] https://www.i-dair.org/
[3] https://atrys.cl/
[4] https://www.reuna.cl/

seño, desarrollo e implementación de prototipos de sistemas de software para atención, formación e investigación, el cual se encuentra a cargo de la “Mesa de Sistemas” (RSDUE, 2021).

La mesa de sistemas de la RSDUE ha impulsado el diseño y pilotaje de dos soluciones de desarrollo propio: un Gestor de Casos Clínicos “GeCCos” y un Registro Clínico Electrónico “U-Salud”, soluciones concebidas para fortalecer el aprendizaje basado en simulación y facilitar el registro de las atenciones brindadas a beneficiarios de los centros dependientes de las casas de estudio respectivamente (Molina et al, 2023). Estos sistemas son fruto de un levantamiento de las necesidades de software de las diferentes facultades y escuelas de las ciencias de la salud que componen la red a nivel nacional, en la cual se detectó la ausencia de soluciones informáticas o la existencia de productos comerciales que no cubren los requerimientos específicos que requiere el área de salud en entornos educativos.

El Sistema Gestor de Casos Clínicos “GeCCOs” es una plataforma digital basada en la nube diseñada para optimizar la creación, administración y evaluación de casos clínicos en el ámbito académico y asistencial. Su desarrollo responde a la necesidad de digitalizar procesos educativos en las carreras de la salud, proporcionando un entorno estructurado y colaborativo para la elaboración de casos clínicos utilizados en la formación de estudiantes y profesionales. Durante el trabajo de levantamiento realizado por la RSDUE se identificó la necesidad de una herramienta informática que favoreciera la relación entre los docentes y estudiantes que consistiese en una plataforma sencilla para intercambiar casos clínicos adecuados para su uso por parte de los centros de simulación. Su enfoque es dirigido a los entornos de simulación clínica, área que se encuentra digitalizada solo en 23% de las instituciones. Esta herramienta busca ser un aporte al aprendizaje de los estudiantes de carreras del área de salud tales como Medicina, Enfermería, Kinesiología, Odontología, Nutrición, Matronería, entre otras, aportando al desarrollo de habilidades clínicas y fomentando el pensamiento crítico para la toma de decisiones (RSDUE, 2023). A nivel de producto, sus principales características son:

- Diseño colaborativo en tiempo real: Múltiples usuarios pueden trabajar simultáneamente en la creación y edición de un caso clínico, facilitando la interacción entre docentes, estudiantes y otros profesionales de la salud.
- Repositorio centralizado: Permite almacenar, categorizar y buscar casos clínicos históricos, promoviendo la reutilización de materiales educativos y el aprendizaje basado en evidencia.
- Herramientas de evaluación integradas: Incluye la posibilidad de desarrollar cuestionarios que facilitan la medición del desempeño de los estudiantes.
- Comunicación y retroalimentación: Dispone de un sistema de comentarios, foros y chat integrado, fomentando la discusión interdisciplinaria y la mejora continua de los casos clínicos.
- Exportación y compatibilidad: Los casos clínicos pueden exportarse en formatos estándar como PDF, asegurando su accesibilidad fuera del sistema.

El Registro Clínico Electrónico (RCE) “U-Salud” ha sido diseñado específicamente para los Centros de Salud Universitarios de las universidades pertenecientes a la RSDUE. Este sistema es un aplicativo web orientado a mejorar la gestión clínica, administrativa y la calidad de atención en contextos de atención ambulatoria a la población beneficiaria de los Centros de Salud Universitarios. Su diseño responde a los estándares de calidad en salud digital y permite una implementación escalable y adaptable a distintos entornos institucionales. Si bien se reconoce la existencia de soluciones comerciales que se ajustan a parte de las necesidades de estos centros, tales como RESERVO o MEDILINK, estas soluciones no cuentan con una cobertura de la totalidad de los procesos que conlleva la atención de un estudiante tales como: validación del beneficiario a nivel de matrícula, ausencia de cobros por servicios adicionales, reportería que permita detectar problemas de salud pública a nivel de escuelas/facultades, o generación de bases de datos secundarias para investigación bajo cumplimiento normativo. Por otro lado, se encontró que este proceso está digitalizado solo en un 44% de las universidades (RSDUE, 2023). Esta propuesta de la Mesa de Sistemas de la RSDUE corresponde a un desarrollo basado en código abierto y el estándar de interoperabilidad HL7 FHIR, el cual ha considerado desde un

comienzo el levantamiento de *stakeholders* y de requerimientos funcionales relacionados a las particularidades de la atención a estudiantes, con la finalidad de ser la herramienta apropiada para brindar atención dentro de los Centros de Salud Estudiantil de las Universidades.

Cabe destacar que este trabajo busca aportar herramientas prácticas que permitan a desarrolladores con experiencia limitada en ciberseguridad o diseño centrado en el usuario, especialmente en contextos como el de la RSDUE, incorporar buenas prácticas desde etapas tempranas del desarrollo de software.

1.2 Problema

El desarrollo de software en salud presenta desafíos inherentes de la ingeniería de software a los que se suma un conjunto de exigencias éticas, legales y funcionales propias del contexto clínico. En este entorno, los errores de diseño o implementación no solo representan fallas operacionales, sino que pueden tener consecuencias directas para el prestador de salud o el paciente. Esta realidad exige que los productos digitales orientados a la salud sean concebidos bajo criterios rigurosos de calidad desde sus etapas más tempranas.

Dos de las dimensiones de calidad más críticas en contextos clínicos son la seguridad y la usabilidad (CENS, 2022). Esto se debe a que el incumplimiento de estándares en seguridad puede derivar en la exposición de datos sensibles, vulnerabilidades ante ciberataques o alteraciones no autorizadas de información clínica, lo que representa un riesgo sanitario y legal significativo. Paralelamente, la ausencia de criterios de usabilidad adecuados genera barreras para la adopción del software por parte de los usuarios finales, ya sean profesionales clínicos, administrativos o docentes, dificultando su integración efectiva en los flujos de trabajo, incrementando los errores operativos y fomentando la resistencia al cambio tecnológico.

Reconociendo la existencia de modelos consolidados para abordar estas dimensiones, como las normas ISO 25010, ISO 27001, las recomendaciones del NIST y los

sellos de calidad del CENS, su implementación en desarrollos de pequeña o mediana escala, como aquellos que emergen en contextos académicos, resulta limitada. Esto obedece, entre otras razones, a la falta de personal especializado, recursos acotados, y ausencia de una cultura organizacional orientada a la gestión de calidad en el desarrollo de software.

En el caso particular de la RSDUE, tanto el Gestor de Casos Clínicos como el Registro Clínico Electrónico representan una oportunidad de responder a necesidades formativas y asistenciales específicas. Sin embargo, carecen de un marco metodológico estandarizado que asegure la incorporación de buenas prácticas de seguridad y usabilidad desde las fases iniciales del ciclo de vida del software. Esta situación dificulta su evolución hacia productos más robustos que puedan consolidarse como soluciones que cumplan requerimientos normativos, y que además logren niveles adecuados de adopción y aceptabilidad por parte de los usuarios finales.

En este contexto se hace necesario contar con un conjunto de lineamientos que permitan operacionalizar los principios de seguridad y usabilidad en desarrollos de software en salud, favoreciendo la mejora continua de los productos tecnológicos en salud generados por agrupaciones y programadores de menor experiencia que buscan ser un aporte al sistema de salud nacional.

2. Objetivos

2.1 Objetivo general

Abordar la evaluación de la calidad del software en salud en términos de seguridad y usabilidad, utilizando instrumentos de evaluación para mejorar la calidad de los prototipos del Gestor de Casos Clínicos y del Registro Clínico Electrónico desarrollados en el contexto de la Red de Salud Digital de las Universidades del Estado, de manera que cumplan con los requisitos necesarios para la obtención de una certificación nacional.

2.2 Objetivos específicos

1) Analizar fuentes relevantes de literatura académica y técnica acerca de las dimensiones de seguridad y usabilidad en software de salud, identificando prácticas recomendadas que permitan guiar el diseño y desarrollo de los prototipos del Gestor de Casos Clínicos y del Registro Clínico Electrónico.

2) Definir lineamientos para la identificación y el abordaje de los requisitos de seguridad en el desarrollo de productos de software en salud, alineados con certificaciones y buenas prácticas reconocidas, organizándolos en un *checklist* de requerimientos que sirva como guía desde las etapas iniciales del desarrollo de software.

3) Definir lineamientos para cumplir con los requisitos de un instrumento validado que evalúe la usabilidad del software en el ámbito de la salud, seleccionándolo según su adecuación al contexto y a los objetivos de los prototipos del Gestor de Casos Clínicos y del Registro Clínico Electrónico.

4) Validar los lineamientos definidos en relación con las dimensiones de seguridad y usabilidad, tomando como referencia los certificados de calidad emitidos por CENS, contrastándolos con los resultados de las evaluaciones para identificar oportunidades de mejora.

3. Metodología

La estrategia metodológica utilizada presenta un enfoque de diseño y selección de instrumentos basado en la evidencia que permiten abordar el desarrollo de software con métodos cualitativos y cuantitativos, y realizar la validación de propuesta de forma sistemática. Este proceso fue organizado en función de los cuatro objetivos específicos previamente definidos, estableciendo una secuencia lógica y dependencias entre las etapas. El énfasis fue colocado en la generación de instrumentos aplicables relacionados al contexto de los desarrollos y diseñados para ser utilizados por equipos de desarrollo con experiencia limitada en ciberseguridad o experiencia de usuario, como ocurre en el contexto académico de la RSDUE.

Durante el desarrollo del trabajo se participó directamente con el equipo de conectores de la mesa de sistemas de la RSDUE. Este equipo está conformado por 4 profesionales, dos ingenieros y dos clínicos, los que actúan como gestores de proyecto, administran los fondos y aseguran la coordinación entre *stakeholders* y equipos de desarrollo.

A nivel de equipos de desarrollo, nos encontramos con dos grupos independientes enfocados en un producto especifico de la RSDUE:

- Equipo de desarrollo "BEDA Software": Encargado del desarrollo del RCE “U-Salud”. Este grupo es un proveedor de software y servicios de desarrollo, el cual cuenta con Beda EMR, un RCE basado en la web con estructura orientada a HL7 FHIR y base de código abierto.
- Equipo de desarrollo "LIB U-Talca": Encargado del desarrollo del Gestor de Casos Clínicos “GeCCos”. El equipo del Laboratorio de Informática Biomédica de la Universidad de Talca.

En relación con la infraestructura tecnológica a utilizar para los proyectos, esta será suministrada por la Universidad de O’Higgins, lo que incluye la configuración y mantenimiento de servidores, monitoreo y aseguramiento de la continuidad operativa de

los sistemas. Cabe destacar que el equipo de dicha Universidad ha aportado al desarrollo del Registro Clínico Electrónico mediante modificaciones menores a la interfaz y funcionalidades del flujo administrativo.

3.1 Análisis y síntesis del marco teórico (OE1)

La primera etapa correspondió a un análisis sistemático de literatura científica, normativa técnica y documentos de referencia relevantes para el desarrollo de software en salud, con foco en las dimensiones de seguridad y usabilidad. La revisión incluyó fuentes académicas y normativas reconocidas, tanto a nivel internacional como nacional.

La búsqueda se realizó en bases de datos académicas indexadas como PubMed y ScienceDirect, utilizando combinaciones de términos como: health software security, usability in health information systems, ISO 25010, ISO 27001, software usability assessment, entre otros. Además, se revisaron documentos técnicos elaborados por el Ministerio de Salud (MINSAL), el Centro Nacional en Sistemas de Información en Salud (CENS) y la Biblioteca del Congreso Nacional de Chile.

Como criterios de inclusión, se priorizaron documentos publicados desde 2010 en adelante, aunque en el caso específico de la usabilidad se aceptaron fuentes anteriores debido a la alta validación y vigencia de instrumentos como la System Usability Scale (SUS) creada en 1986 y en uso para aplicaciones móviles de salud digital (Maramba et al, 2019), o las Heurísticas de Jackob Nielsen (Nielsen, 1994). Se excluyeron publicaciones con baja aplicabilidad al contexto nacional o que no abordaran explícitamente las dimensiones de calidad evaluadas.

Los documentos seleccionados fueron revisados, permitiendo categorizar buenas prácticas, modelos conceptuales, criterios técnicos y requisitos normativos. Esta sistematización constituyó el insumo base para la construcción de los instrumentos para las siguientes etapas.

3.2 Definición de lineamientos de seguridad (OE2)

A partir de la evidencia recolectada en la etapa anterior, se procedió a la elaboración de un conjunto de lineamientos prácticos orientados a guiar a los equipos de desarrollo en la incorporación de requisitos de seguridad desde las fases tempranas del ciclo de vida del software.

Para transmitir de mejor manera los lineamientos a los equipos de desarrollo, se decidió organizar los elementos en un *checklist* de forma tal que pudiera ser autoaplicable por equipos de desarrollo sin especialización en ciberseguridad. En este sentido la idea fue proporcionar un instrumento concreto, comprensible y contextualizado que facilitara la incorporación de buenas prácticas en entornos académicos como los de la RSDUE. Cada elemento incluido en este instrumento considera: descripción, criterios de aceptabilidad, y medios sugeridos de verificación. El formato de respuesta es binario (cumple/no cumple), con una opción adicional “no aplica”, diseñada para flexibilizar su uso según las características del software evaluado.

La aplicación del *checklist* de seguridad fue realizada en múltiples instancias durante el desarrollo de los productos de software de la RSDUE, buscando por un lado reconocer los elementos considerados cubiertos por parte de los desarrolladores y por otro identificar aquellos que puedan presentar desviaciones o no estar considerados. Para la primera aplicación de este instrumento, se realizaron inducciones a los desarrolladores a través de reuniones explicativas coordinadas en conjunto con el líder de proyecto. En estas sesiones se presentó el objetivo de los lineamientos y se entregaron los instrumentos de evaluación, detallando los criterios de aceptabilidad y los medios de verificación requeridos.

Durante el plazo de una semana, los desarrolladores completaron de forma autónoma el *checklist* de seguridad, adjuntando la evidencia correspondiente. Esta aplicación inicial tuvo lugar en fases tempranas del desarrollo, lo que permitió entregar una retroalimentación temprana al equipo de la RSDUE. A lo largo del proyecto, se mantuvo una comunicación continua con el equipo, canalizada principalmente a través

del líder de proyecto, quien actuó como interlocutor para centralizar las respuestas y coordinar con los responsables técnicos para la validación de elementos técnicos específicos, como por ejemplo, el mecanismo de cifrado en tránsito y cifrado en reposo. Esta modalidad de interacción permitió optimizar el proceso de recolección de información, evitando duplicaciones y mejorando la precisión de las respuestas.

3.3 Definición de lineamientos de usabilidad (OE3)

Para abordar la dimensión de usabilidad se elaboró un instrumento que sintetiza buenas prácticas de diseño centrado en el usuario en contextos de software para salud. La guía fue organizada bajo el formato “Haz y Evita”, estructurada en dos categorías principales:

- Experiencia usuaria y funcionalidades, que abarca aspectos como alineación con flujos clínicos reales, personalización por perfil, soporte a la toma de decisiones, entre otros.
- Elementos de interfaz usuaria, que incluye principios de visibilidad, accesibilidad, prevención de errores y diseño coherente.

Esta bajada de lineamientos fue utilizada para evaluar críticamente capturas de pantalla y versiones de desarrollo interactivas de los productos de la RSDUE, permitiendo el levantamiento de observaciones que fueron procesadas por el equipo para la generación de requerimientos de desarrollo que permitiesen lograr una experiencia usuaria deseada.

En paralelo a la elaboración del instrumento de buenas prácticas, se desarrolló un análisis comparativo de instrumentos de evaluación de usabilidad comúnmente utilizados en salud y tecnología, como SUS, TAM, PSSUQ, entre otros. Este análisis fue sistematizado en una matriz de comparación de criterios, considerando variables como profundidad de análisis, validación empírica, facilidad de aplicación, y pertinencia al contexto de los desarrollos de la RSDUE.

El enfoque de la comparación fue reconocer las fortalezas y características de cada instrumento para permitir la identificación del más adecuado según el contexto y estado del desarrollo del producto, considerando el acceso o no a usuarios finales y conocimientos requeridos para su aplicación como por ejemplo detalles de la institución que hará uso del software o requerir de cierto grado de expertiz para su aplicación. En este sentido se reconoce que, dado que los contextos de uso de los desarrollos pueden variar entre universidades, la aplicación del instrumento más adecuado dependerá del nivel de avance del proyecto, considerando si este se encuentra en etapa de desarrollo o de implementación.

Para la evaluación práctica de los productos de la red, se aplicó la selección del instrumento que mejor se adapte al nivel de desarrollo alcanzado, lo cual coincidió con la disponibilidad de versiones funcionales de prueba para ambos productos de software, sin acceso a los usuarios finales (universidades). En este sentido, se definió un grupo de prueba para cada producto levantado por la RSDUE, sobre el cual se aplicó el instrumento seleccionado correspondiente.

3.4 Validación de los lineamientos definidos (OE4)

La etapa final de la metodología se enfocó en validar los lineamientos generados desde dos perspectivas complementarias: la alineación de los lineamientos con el modelo de referencia establecido por CENS y la utilidad práctica de los lineamientos para los equipos de desarrollo el valor percibido de los instrumentos desde el punto de vista de los usuarios clave.

Alineación con modelo de referencia

Para evaluar los lineamientos propuestos se utilizaron dos métodos. El primero correspondió a comparación de los lineamientos propuestos con el modelo referencial del sello de calidad para software en salud de CENS. En relación con este se elaboró una matriz de correspondencia entre los ítems del *checklist* de seguridad y los requerimientos asociados a seguridad de CENS. Esta matriz buscó verificar la cobertura

conceptual de los instrumentos, identificando aquellos que coinciden plenamente con las exigencias del sello, así como posibles vacíos o redundancias.

Para cubrir aspectos más específicos, se incorporó un segundo método basado en una revisión por juicio de expertos con el objetivo de verificar la pertinencia técnica de los instrumentos y corroborar la alineación de los lineamientos de seguridad con los criterios utilizados en el sello de calidad de RCE al cual no se obtuvo acceso directo. En este contexto, se estableció una colaboración con el equipo de calidad del CENS, contactando con dos ingenieros del área. A lo largo del proceso se sostuvieron múltiples reuniones técnicas en las que se presentó el contenido del *checklist* de seguridad y se discutió su estructura, cobertura temática y formato de aplicación.

Validación de la utilidad, claridad y aplicabilidad

Con el objetivo de evaluar si los lineamientos generados resultaban comprensibles, útiles y efectivamente utilizables por los desarrolladores, se realizó una validación centrada en el usuario de los instrumentos de bajada de lineamientos.

Se aplicó un cuestionario a integrantes del equipo de desarrollo de los sistemas evaluados, con preguntas orientadas a medir:

- Claridad de los lineamientos
- Relevancia percibida de los contenidos
- Valor práctico como herramienta de mejora o guía de diseño
- Dificultades encontradas en su aplicación
- Y sugerencias de mejora

Este cuestionario fue diseñado con escala de Likert y preguntas abiertas para permitir su análisis. Para su interpretación se generaron los indicadores presentados en la tabla 2.

Indicador	Descripción	Fórmula
Claridad del *checklist* de seguridad	Evaluación subjetiva de comprensión mediante porcentaje de participantes que calificaron con 4 o 5 la claridad del *checklist*.	(N° encuestas con 4 o 5 ptos en la pregunta) / Total respuestas
Utilidad del *checklist* de seguridad	Percepción de utilidad práctica del *checklist* mediante porcentaje de participantes que valoraron como útil o muy útil el formato.	(N° encuestas con 4 o 5 ptos) / Total respuestas
Detección de aspectos nuevos de seguridad	Porcentaje que indicó que el *checklist* le ayudó a identificar aspectos previamente no considerados (valor formativo).	(N° encuestas con 4 o 5 ptos) / Total respuestas
Claridad de la guía de usabilidad	Comprensión de la guía de recomendaciones de usabilidad mediante porcentaje que la calificó como clara o muy clara.	(N° encuestas con 4 o 5 ptos) / Total respuestas
Utilidad percibida de la guía de usabilidad	Potencial de aplicación de la guía de usabilidad mediante porcentaje que la considera de utilidad.	(N° que respondió "Sí, serían muy útiles") / Total respuestas
Uso futuro de los instrumentos	Disposición de adoptar instrumentos en proyectos futuros medido como porcentaje de participantes que recomendarían usar estos instrumentos.	(N° que respondió "Sí") / Total respuestas

Tabla 2: Indicadores propuestos para evaluación de percepción de instrumentos propuestos para la bajada de lineamientos de seguridad y usabilidad.

Para complementar la utilidad de los instrumentos desarrollados, se realizó una validación de la comprensión de los instrumentos. Esta se realizó con la ayuda de dos profesionales con experiencia en desarrollo de software en salud, con roles distintos (*backend* y *fullstack*), quienes leyeron de forma independiente los instrumentos dedicados a seguridad y usabilidad.

Para su evaluación se le pidió a cada profesional que interprete libremente el significado de cada elemento, explicando en sus propias palabras lo que entiende que se requiere o recomienda. Posteriormente, sus interpretaciones intercambiadas entre los profesionales, asumiendo un rol de validador con lo cual revisaron si el otro profesional entendió lo mismo que ellos, permitiendo establecer si hay concordancia en la interpretación. Los indicadores correspondientes se encuentran desarrollados en la tabla 3.

Indicador	**Descripción**	**Fórmula**
Concordancia en Interpretación (Seguridad)	% de ítems del *checklist* en que ambos profesionales coincidieron en la interpretación clave	(N° ítems con interpretación coincidente) / Total ítems *checklist* seguridad
Concordancia en Interpretación (Usabilidad)	% de recomendaciones "Haz y Evita" en que hubo concordancia total o parcial	(N° recomendaciones coincidentes) / Total recomendaciones

Tabla 3: Indicadores propuestos a partir de evaluación cruzada de interpretación de lineamientos entre profesionales de desarrollo de software.

Para complementar la experiencia de uso de los instrumentos propuestos, se realizó también un *focus group* con integrantes de la Mesa de Sistemas de la RSDUE. Esta actividad tuvo un carácter cualitativo y participativo, adoptando la forma de un conversatorio orientado a la reflexión grupal. A diferencia de las encuestas individuales previamente aplicadas a desarrolladores, esta instancia buscó recoger percepciones transversales, identificar aprendizajes y explorar tanto los factores facilitadores como las barreras para la incorporación de lineamientos de calidad, con foco en seguridad y usabilidad, desde la perspectiva de los equipos de proyecto. El *focus group* fue grabado con autorización de los participantes, y la conversación fue posteriormente analizada con el propósito de enriquecer la interpretación de los resultados y aportar una mirada situada desde la experiencia práctica del equipo RSDUE.

4. Resultados

4.1 Literatura y normativa técnica (OE1)

La revisión de literatura especializada y normativas permitió reconocer los marcos conceptuales, regulatorios y técnicos que fundamentan la elaboración de los lineamientos de seguridad y usabilidad. Estos principios se abordan en los siguientes cinco apartados.

a) Estándares internacionales de calidad del software

La norma ISO 25010 define el modelo de calidad del producto de software, y se consolidó como la principal referencia estructural. Esta norma incluye las características de usabilidad y seguridad entre sus ocho dimensiones principales (ISO, 2023). En particular, se destacaron las siguientes características:

- Usabilidad: operabilidad, protección contra errores de usuario, estética, accesibilidad.
- Seguridad: confidencialidad, integridad, autenticación, autorización.

Estas características fueron extraídas y traducidas en componentes evaluables dentro de los instrumentos construidos, particularmente en el *checklist* de seguridad y la guía de usabilidad.

b) Certificaciones de calidad de software en salud nacionales

El Centro Nacional en Sistemas de Información en Salud (CENS) cuenta con modelos de certificación orientados a garantizar la calidad de productos de software en salud, siendo los más relevantes para este trabajo el Sello de Calidad de Software en Salud (CENS, 2022) y el Sello de Registro Clínico Electrónico (CENS, 2023 4. Ambos sellos comparten una lógica similar de aplicación (Figura 2), la que se realiza con el llenado inicial de un formulario y posterior presentación y entrega de entorno demo. Las principales características de cada uno son descritas a continuación y una breve comparativa se encuentra disponible en la Tabla 4:

- El Sello de Calidad de Software en Salud se basa en la normativa ISO 25010:2011, evaluando una selección de ocho dimensiones, entre las que se incluyen fiabilidad, seguridad y usabilidad. Su evaluación se realiza mediante la aplicación de un instrumento de desarrollo propio que consta de 107 preguntas específicas y 6 generales, la que se realiza mediante un formulario donde el interesado (por ejemplo, el equipo de proyecto) además de responder hace entrega evidencia y organiza sesiones demostrativas del software.
- El Sello de Registro Clínico Electrónico se focaliza en seguridad y usabilidad, aplicando no solo los criterios de la ISO 25010, sino que la expande bajo la mirada de la ISO 27001 en cuanto a seguridad y la NISTIR 7804 respecto a usabilidad. Para su evaluación se exige contar con un entorno de pruebas para uso por un comité de expertos, y la presentación del software mediante recorrido guiado en vivo.

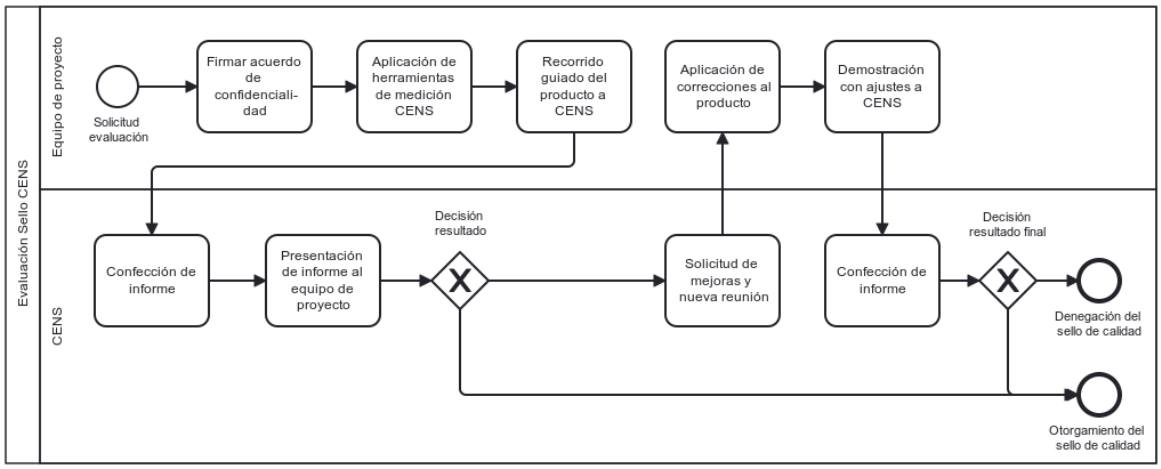

Figura 2: Flujo que describe el proceso para la evaluación y obtención de los Sellos de calidad ofrecidos por CENS, desde la realización de la solicitud hasta la obtención del sello.

Aspecto	Sello de Calidad CENS Software en Salud	Sello de Calidad CENS Registro Clínico Electrónico (RCE)
Normas en que se basó su desarrollo	ISO 25010:2011	NISTIR 7804, ISO 25010 e ISO 27001
Dimensiones de calidad que evalúa	- Seguridad - Portabilidad - Fiabilidad - Usabilidad - Compatibilidad - Eficiencia de desempeño - Adecuación funcional	- Usabilidad - Seguridad
Características diferenciadoras	*Permite evaluar la calidad del software en salud de manera general, integrando la adecuación funcional. *Es aplicable a todo desarrollo orientado al sector salud independiente del objetivo del software. *Consiste en un cuestionario extenso en base a carga de evidencia con 107 preguntas dirigidas a las dimensiones de calidad evaluadas y 6 preguntas generales.	*Permite evaluar la calidad en dos dimensiones críticas para las instituciones de salud, acotando la evaluación a aspectos esenciales. *Uso exclusivo para Registro Clínico Electrónico, especialmente en la dimensión de Usabilidad *El proceso de evaluación es llevado por un equipo de expertos del CENS, requiriendo de una demostración en vivo dirigida por el desarrollador, y acceso a un entorno de pruebas para revisiones específicas.

Tabla 4: Comparación entre los dos sellos desarrollados por CENS, denotando el alcance en dimensiones a evaluar y características diferenciadoras.

c) Seguridad de la información/ciberseguridad

La seguridad de la información es parte de la familia de normas ISO 27000, donde nos encontramos con algunas específicas como la ISO 27001 que se relaciona con establecer un sistema de gestión de seguridad de la información “SGSI” (ISO, 2022), la ISO 27002 que guía en la implementación de controles de seguridad y la ISO 27799 asociada a la aplicación de las medidas de confidencialidad, integridad y disponibilidad de información para el sector salud (ISO, 2016).

Siendo la ISO 27001 la que encabeza esta familia de normativas de seguridad, se debe reconocer que su aplicación busca la minimización, mas no la eliminación completa de los riesgos. Su enfoque va asociado a la identificación temprana de los riesgos y la generación de planes de acción, los que mediante controles y procedimientos aportan a robustecer la seguridad de la información. En este sentido se busca garantizar la confidencialidad de la información a través del acceso solo de quienes tengan autorización, la integridad a través de un procesamiento y almacenamiento adecuado, y la disponibilidad de los datos bajo demanda para quienes tengan el acceso autorizado. Esta mirada fue rescatada para la lectura crítica de la literatura buscando no solo elementos técnicos sino también prácticas que atingentes al desarrollo de los productos de la RSDUE.

En el ámbito normativo nacional, la modificación de la Ley N°19628 sobre Protección de la Vida (MINSEGPRES, 2023; MINSEGPRES, 2024), cuya entrada en vigencia plena está programada para diciembre de 2026, en conjunto con la Ley Marco de Ciberseguridad (Ministerio Interior, 2024) establecen una nueva arquitectura legal de protección de datos en Chile, alineada con leyes como GDPR en Europa (European Parlament, 2016) y HIPAA en Estados Unidos (HHS, 1996).

Para un cumplimiento adecuado del nuevo marco legislativo se debe considerar transparencia al usuario respecto del manejo de los datos y el uso de software que permita el cumplimiento de la protección de la información. Estas exigencias justifican la incorporación de mecanismos de cumplimiento normativo dentro del *checklist* de seguridad desarrollado.

A partir de la ley actualizada, en su artículo segundo, se rescatan las siguientes definiciones:

- Dato personal (correspondiente a las normativas internacionales al PII o "*Personally Identifiable Information*"): Se refiere a toda información que permite identificar directa o indirectamente a una persona natural. Esto incluye datos como el Rol Único Nacional (RUN), nombre completo, dirección domiciliaria,

número de teléfono, correo electrónico institucional, o incluso la combinación de múltiples atributos que, en conjunto, permiten identificar a un individuo. Por ejemplo, el registro “Estudiante de Medicina de la Universidad de Valparaíso” o “mujer de 23 años de Chillán” corresponden a esta categoría al permitir la identificación directa o indirecta e la persona en contextos determinados.

- Datos personales sensibles (correspondiente en normativas internacionales al PHI o “*Protected Health Information*”): Abarca toda información relacionada con las características físicas o morales, junto con registros relacionados a la vida privada, ideología, afiliaciones gremiales, datos de salud, biométricos y orientación sexual entre otros que puedan llevar a una discriminación o vulneración de su intimidad.

d) Normativa técnica nacional complementaria

Se revisaron los siguientes documentos relacionados a la política de seguridad del MINSAL y Gobierno Digital que orientan el desarrollo de software en el estado:

- Política General de Seguridad de la Información y Ciberseguridad v4.0 (MINSAL, 2023)
- Política de Desarrollo de Sistemas v3.0 (MINSAL, 2023)
- Política de Protección de Datos Personales (MINSAL, 2023)
- Ley 20584 de Derechos y Deberes de los Pacientes (MINSAL, 2012)
- Decreto 41, relacionado a la regulación de las fichas clínicas (MINSAL, 2012)
- Lineamientos de Gobierno Digital para desarrollo de software (Gobierno Digital, 2021)

De esta documentación se extrajeron elementos técnicos como lo es la versión mínima exigida para protocolos como TLS v1.3, el nivel de seguridad mínimo exigido a contraseñas y otras indicaciones relacionadas a buenas prácticas. Además, en el caso de las fichas clínicas se establece el tratamiento de su contenido como dato sensible y el almacenamiento por a lo menos 15 años en calidad de custodio a partir del último registro realizado en la ficha.

e) Fundamentos metodológicos para la evaluación de usabilidad

La evaluación de usabilidad fue fundamentada en tres referencias clave:

- ISO 9241-11:2018: Define la usabilidad como el grado en que un sistema puede ser utilizado por usuarios específicos para lograr objetivos concretos con efectividad, eficiencia y satisfacción, dentro de un contexto de uso determinado. Establece un marco metodológico para evaluar la calidad de uso de productos interactivos, considerando tanto las características del sistema como el entorno, las tareas y el perfil del usuario, guiando el diseño centrado en el usuario (ISO, 2018).
- Heurísticas de Usabilidad de Jackob Nielsen: Conjunto de 10 principios para el diseño de interfaces centradas en el usuario. Permiten, mediante la evaluación por expertos, detectar errores comunes de diseño que afectan la experiencia del usuario abordando aspectos como visibilidad del estado del sistema, prevención de errores, flexibilidad y control del usuario, entre otros. Estas heurísticas son especialmente útiles en entornos de desarrollo con recursos limitados, donde se requiere una evaluación rápida (Nielsen, J., 1994).
- NISTIR 7804: Establece un marco técnico para la evaluación, prueba y validación de la usabilidad en sistemas de registro clínico electrónico, proporcionando una metodología estructurada basada en principios de diseño centrado en el usuario, que incluye la definición del contexto de uso, el desarrollo de prototipos interactivos, la realización de pruebas con usuarios representativos y el análisis de métricas como tasa de errores, eficiencia de tareas y satisfacción percibida (Lowry, S. et al, 2012).

Estas referencias permitieron desarrollar una guía operativa y comprensible, adaptable al contexto de uso de los prototipos RSDUE, y compatible con procesos de validación participativa con usuarios clínicos.

4.2 Lineamientos de seguridad (OE2)

A continuación, se presentan los resultados obtenidos en relación con el desarrollo y la aplicación del instrumento diseñado para orientar la incorporación de criterios de seguridad.

a) Estructura del instrumento

El *checklist* desarrollado en torno a la dimensión de seguridad incluye una serie de elementos formulados como requerimientos técnicos concretos, operativizables y verificables, con el fin de guiar la integración de criterios de seguridad desde las etapas iniciales del ciclo de desarrollo. Cada componente está acompañado de criterios de evaluación explícitos y medios de verificación sugeridos, lo que permite tanto orientar al desarrollador como facilitar la validación externa. Esta estructura permite que el *checklist* funcione no solo como una pauta técnica, sino también como una herramienta de mejora continua. Su diseño contempla 18 elementos agrupados en cinco bloques temáticos:

- Control de acceso y autenticación: requerimientos relacionados a la segmentación de perfiles, uso de credenciales y protección contra ataques de fuerza bruta.
- Protección de Datos Sensibles: Enfocado en la seguridad de los datos a través de cifrado tanto en tránsito como en reposo, consentimiento de almacenamiento y políticas de manejo de datos.
- Monitoreo, Integridad y Auditoría: mecanismos para registro de eventos críticos e interacción con registros, integridad de datos y gestión de incidentes relacionados a los datos.
- Resistencia y Recuperación: Evaluación de pruebas de seguridad, existencia de respaldos y continuidad operativa.
- Normativas y Buenas Prácticas: Recomendaciones de seguridad a usuarios e institución que hará uso del software.

Cada elemento considerado en el *checklist* se encuentra descrito mediante un enunciado que consiste en el número de elemento y breve descripción, la redacción del requerimiento propiamente tal y los criterios de aceptabilidad. A continuación, se presentan tres columnas para indicar el nivel de cumplimiento, y finalmente un campo de texto abierto donde se encuentra una breve descripción de medios de verificación

sugeridos para evidenciar el cumplimiento, lo que se puede apreciar en el extracto presentado en la tabla 5.

Protección de Datos Sensibles				
Elemento	**Cumple (1)**	**No cumple (0)**	**No aplica (N/A)**	**Comentario / Evidencia**
5. Cifrado en Tránsito: Implementación de TLS 1.3 o equivalente. Cifrar toda información confidencial que se envía entre el producto y los servidores. Criterios aceptabilidad: Toda la información confidencial, como contraseñas, datos financieros y datos personales, debe ser cifrada antes de transmitirse por la red. Se debe utilizar un algoritmo de cifrado estándar y probado, como AES o RSA. MINSAL sugiere cifrado de tránsito TLS 1.3. Las claves de cifrado deben almacenarse de forma segura y deben ser solo accesibles para el administrador del sistema.				Documentación del protocolo de cifrado utilizado con captura del navegador donde se muestre la versión en uso.

Tabla 5: Extracto del *checklist* propuesto para representación de columnas presentes.

El instrumento puede ser aplicado a modo resumido en tres momentos clave del ciclo de vida del software: desarrollo inicial, etapa de pruebas e implementación/explotación.

b) Aplicación del instrumento

El *checklist* fue implementado como parte del proceso de desarrollo de los dos productos de software de la RSDUE. En el Gestor de Casos Clínicos, el instrumento acompañó el desarrollo hasta la versión *beta*, mientras que en el Registro Clínico Electrónico fue hasta una versión *alpha*. Los principales hallazgos derivados de la aplicación del instrumento fueron:

- **Gestor de Casos Clínicos**: Dado que el sistema no maneja datos de personas reales, se observaron varios elementos marcados como “no aplica” dentro del bloque de “Protección de Datos Sensibles”. No obstante, se detectaron mejoras necesarias en cuanto a la diferenciación de perfiles de usuario, mecanismos de trazabilidad de acciones y, el elemento más débil: la documentación técnica del sistema.
- **Registro Clínico Electrónico:** Se identificaron deficiencias en la implementación del control de accesos por perfil, baja seguridad de las credenciales de acceso y carencia de cifrado en base de datos. Estas observaciones permitieron al equipo priorizar ajustes en la arquitectura del sistema, definir el orden de desarrollo de los nuevos requerimientos a solicitar y orientar los esfuerzos hacia una versión inicial robusta, alineada con los estándares normativos establecidos. La retroalimentación entregada sirvió también como insumo para realizar mejoras en cuanto a la documentación técnica exigida por el sello de calidad para RCE del CENS.

c) Evolución del instrumento

Posterior a la primera aplicación del *checklist* de seguridad, se identificaron dificultades en la interpretación de algunas categorías de respuesta por parte de los desarrolladores. En particular con una opción intermedia enfocada en reconocer aquellos elementos que no se encuentran completados, pero si considerados o en desarrollo. Esto generó confusión sobre su relación con el cumplimiento efectivo de los requisitos, dificultando la evaluación comparativa entre versiones del producto.

Como respuesta a estas observaciones, se realizó una revisión estructural del instrumento, con énfasis en mejorar la claridad de uso y reducir la ambigüedad en la recolección de evidencia. Las principales modificaciones realizadas fueron:

- Cambio de las opciones de respuesta eliminando la categoría “pendiente de desarrollo” y manteniendo únicamente las respuestas “Cumple”, “No cumple” y “No aplica”, con el objetivo de favorecer decisiones claras y facilitar el análisis evolutivo.
- Clarificación de criterios de aceptabilidad y medios de verificación para cada ítem, incorporando ejemplos más concretos y mejor alineados con el elemento evaluado.

Estas mejoras fueron bien recibidas en las aplicaciones subsecuentes del *checklist*, donde se observó una mayor consistencia en las respuestas entregadas y una reducción en la cantidad de observaciones referidas a interpretación de los elementos.

4.3 Lineamientos de usabilidad (OE3)

La dimensión de usabilidad fue abordada mediante dos estrategias: por un lado, la elaboración de una guía de recomendaciones basada en buenas prácticas que pueda apoyar al equipo en la revisión del diseño de la interfaz, y por otro la recomendación de instrumentos estandarizados que permitieran una evaluación estructurada de la usabilidad de los sistemas en desarrollo.

a) Elaboración de una guía de buenas prácticas

La guía propuesta fue construida a partir de literatura especializada y marcos normativos internacionales, incluyendo principios del NISTIR 7804, y criterios derivados de heurísticas de diseño centradas en el usuario. La guía contiene recomendaciones positivas ("Haz") y alertas negativas ("Evita") orientadas a facilitar decisiones de diseño aplicables en entornos de desarrollo ágil. Este enfoque para la socialización de buenas prácticas, utilizado por empresas tecnológicas como Apple para recomendaciones de interfaces de usuario (Apple, 2017), y Google en relación a *Material Design 3* (Google, 2021) en documentación orientada a diseñadores y desarrolladores *front-*

end, dado que permite presentar la información en pares antagónicos que delimitan explícitamente comportamientos deseados y errores comunes. No se trata de una guía de estilo estricta, sino de un conjunto de orientaciones prácticas suficientemente abiertas para permitir adaptación contextual, pero que establecen umbrales mínimos de aceptabilidad para evitar errores críticos, entregando libertad para su interpretación y siendo un apoyo incluso para equipos de desarrollo con experiencia limitada en diseño centrado en el usuario.

El documento fue organizado en dos secciones para reducir su extensión, estableciendo enfoques complementarios:

- Experiencia usuaria y funcionalidades, con énfasis en la alineación con flujos clínicos reales, personalización de funciones por perfil, soporte a la toma de decisiones y prevención de errores.
- Elementos de interfaz, incluyendo principios como visibilidad del estado del sistema, consistencia, legibilidad, retroalimentación inmediata y diseño accesible.

Esta guía fue aplicada junto con el equipo de la RSDUE en la evaluación de los productos Gestor de Casos Clínicos en su versión *beta* y Registro Clínico Electrónico en su versión *alpha* guiando una evaluación a juicio del equipo respecto de la experiencia entregada en dichas versiones funcionales para la elaboración de requerimientos que permitieran propiciar una mejor experiencia usuaria.

b) Aplicación de la guía de buenas prácticas durante el desarrollo

La guía de buenas prácticas propuesta fue utilizada como insumo en la revisión crítica de las versiones *beta* del Gestor de Casos Clínicos y *alpha* del Registro Clínico Electrónico. La aplicación consistió en sesiones de análisis y navegación por la interfaz de los sistemas en conjunto con el equipo de proyecto en las que se identificaron elementos positivos y áreas de mejora en las interfaces.

En el caso del Gestor de Casos Clínicos, se identificaron aciertos como el uso de etiquetas comprensibles en los formularios y la retroalimentación inmediata al ingresar respuestas incorrectas en cuestionarios. No obstante, se observaron deficiencias en la navegación entre secciones del caso clínico, uso inconsistente de términos, la falta de una página de inicio quedando un fondo blanco junto al menú una vez iniciada la sesión y un menú de navegación que dificultaba el acceso a botones como el de cierre de sesión. Estas observaciones fueron derivadas al equipo de desarrollo como nuevos requerimientos o bugs según correspondiese para su abordaje.

En el caso del Registro Clínico Electrónico, la revisión reveló un mayor número de oportunidades de mejora, las que se pueden agrupar de la siguiente manera:

- Navegación y estructura de la interfaz: Observaciones relacionadas a simplificar la secuencia de pasos, reduciendo clics innecesarios y mejorando la orientación del usuario dentro del sistema.
- Consistencia visual y uso de componentes: Recomendación de estandarizar textos asociados a botones y etiquetas. Se destacó el uso consistente de colores evitando confusiones derivadas de variabilidad visual.
- Errores y retroalimentación: Se realizaron sugerencias para mejorar el mensaje entregado en algunos eventos de error, junto con potenciar validaciones de ingreso de datos en formularios y modales de confirmación de acciones críticas.
- Flujos funcionales y lógica de uso: Se identificación redundancias o ambigüedades en flujos clínicos simulados, proponiendo alternativas más coherentes con una atención clínica real permitiendo además la reutilización de datos relevantes.

Ambas evaluaciones guiadas por las recomendaciones propuestas permitieron fortalecer la comprensión del equipo sobre principios de diseño centrado en el usuario, y generaron una propuesta de mejoras para ajustar la interfaz de los productos.

c) Revisión y comparación de modelos de evaluación de usabilidad

En paralelo se desarrolló un análisis comparativo de instrumentos de evaluación de usabilidad, considerando tanto escalas estandarizadas como marcos teóricos de aceptación tecnológica. Se revisaron modelos como System Usability Scale (SUS), Post-Study System Usability Questionnaire (PSSUQ), Technology Acceptance Model (TAM) y DeLone & McLean, evaluando su aplicabilidad, profundidad diagnóstica, la posibilidad de acceso a usuarios finales e instituciones y la adecuación al contexto de uso del software. Esta revisión fue sistematizada en una tabla comparativa, en la que se destacó la simplicidad del SUS como ventaja operativa, pero también su limitación diagnóstica frente a necesidades de mejora concreta.

Este análisis buscó identificar el modelo más adecuado para ser aplicado sobre los prototipos desarrollados por la RSDUE, considerando características como la profundidad analítica, facilidad de implementación, y adaptabilidad al entorno clínico. Los instrumentos comparados se detallan en la tabla 6 presente a continuación:

Modelo	Qué evalúa	Facilidad de uso	Contexto de uso sugerido	Fortalezas	Debilidades
SUS[5]	Usabilidad general	Muy alta	Evaluaciones rápidas desde productos mínimos	Fácil de aplicar, resultados cuantificables	Superficial, no aborda dimensiones complejas
PSSUQ[6]	Satisfacción post-uso	Alta	Evaluación tras prueba funcionales exitosas	Profundiza en utilidad percibida	Requiere explicación al usuario y prueba del sistema
Modelo D&M[7]	Éxito de sistemas de información a nivel organizacional	Media	Evaluación global del impacto del sistema en organizaciones de salud	Integra calidad de sistema, de información y satisfacción del usuario	Requiere medición compleja, menos enfocado en la interfaz y la UX

[5] Brooke, J. “SUS: A ‘quick and dirty’ usability scale”. In: Jordan, P.; Thomas, B.; Weerdmeester, B.; McClelland, I., editors. Usability Evaluation in Industry. London: Taylor & Francis; 1995. p. 189–194.
[6] Lewis, J.R. “Psychometric evaluation of the post-study system usability questionnaire: The PSSUQ”. Proc Hum Factors Soc Annu Meet. 1992;36(16):1259–1263.
[7] DeLone, W.H.; McLean, E.R. “The DeLone and McLean model of information systems success: A ten-year update”. J Manag Inf Syst. 2003;19(4):9–30.

Modelo	Qué evalúa	Facilidad de uso	Contexto de uso sugerido	Fortalezas	Debilidades
TAM[8]	Aceptación tecnológica	Alta	Evaluaciones de disposición de usuarios finales	Predice intención de uso a través de la aceptación	No mide directamente experiencia real
HOT-Fit[9]	Factores humano-organizativos	Media	Evaluaciones holísticas en entornos de salud considerando características de la institución	Integra tecnología, organización y usuario	Complejo, difícil de aplicar sin expertos
ISO 25010	Calidad formal del software	Baja	Certificaciones formales, auditorías	Normativo, amplio y estandarizado	No tiene cuestionarios predefinidos
AttrakDiff[10]	Calidad emocional y estética	Media	Evaluación de percepción subjetiva	Diferencia entre lo hedónico y lo funcional	Menor aplicabilidad en salud

Tabla 6: Comparación entre instrumentos de usabilidad, denotando fortalezas y debilidades y sugiriendo contexto de uso.

Dado que al momento de tener que realizar la evaluación de los sistemas no se contaba con acceso a usuarios finales, se consideró la selección y aplicación del instrumento en versiones de desarrollo frente a grupos representativos de los potenciales usuarios del sistema. En este sentido, a partir del análisis previo, se pudo establecer la siguiente estrategia de evaluación:

c) PSSUQ (Post-Study System Usability Questionnaire)

Se sugirió el uso del instrumento PSSUQ para la evaluación de la usabilidad en el Gestor de Casos Clínicos, dado su potencial para capturar de manera detallada percepciones de los usuarios tras la interacción con un sistema en estado funcional. Es-

[8] Ma, Q.; Liu, L. “The technology acceptance model”. In: Dwivedi, Y.K. et al., editors. Handbook of Research on Contemporary Theoretical Models in Information Systems. IGI Global; 2005. p. 17–36.
[9] Yusof, M.M.; Kuljis, J.; Papazafeiropoulou, A.; Stergioulas, L.K. “An evaluation framework for health information systems: Human, organization and technology-fit factors (HOT-fit)”. Int J Med Inform. 2008;77(6):386–398.
[10] Hassenzahl, M.; Burmester, M.; Koller, F. “AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität”. In: Szwillus, G.; Ziegler, J. Mensch & Computer 2003. Vieweg+Teubner Verlag; 2003. p. 187–196.

te cuestionario, compuesto por 16 elementos evaluados en una escala de Likert de 7 puntos, permite analizar dimensiones clave como la utilidad percibida del sistema, la calidad de la información entregada, la interfaz de usuario y la satisfacción general.

La elección de este instrumento se basó en su capacidad diagnóstica para identificar áreas de mejora en productos digitales que han alcanzado un nivel avanzado de desarrollo, y en su amplio uso en contextos de evaluación post-interacción. Además, su estructura validada y enfoque en la experiencia subjetiva del usuario lo convierten en una herramienta relevante para reforzar decisiones de rediseño centradas en la experiencia.

Los resultados recolectados de su aplicación en relación con el Gestor de Casos Clínicos permitieron observar patrones útiles para el análisis. Estos incluyeron una satisfacción general media-alta, con buena valoración en la facilidad de uso, pero también la detección de dificultades en aspectos como la navegación inicial y la claridad de los menús.

d) SUS (System Usability Scale)

Se propuso el uso del cuestionario SUS para la evaluación de usabilidad en el RCE, considerando su breve extensión, alta validación internacional y su idoneidad para sistemas en fases tempranas de desarrollo. Este instrumento, compuesto por 10 elementos en escala de Likert, permite obtener una métrica general de usabilidad con bajo requerimiento técnico y sin necesidad de un entorno plenamente operativo, lo cual lo hace especialmente útil en etapas iniciales de diseño funcional.

La selección de SUS respondió a la necesidad de contar con una medición basal de usabilidad que pudiera ser utilizada como referencia para evaluar el progreso del sistema a lo largo de futuras iteraciones. En este sentido, se planificó su aplicación a una cohorte de usuarios provenientes de los centros de salud universitarios y de la Red de Salud Digital Universitaria del Estado (RSDUE), haciendo uso de su versión validada al español (Sevilla et al., 2020). No obstante, debido al nivel de avance al-

canzado en el desarrollo del RCE durante el periodo de realización del presente trabajo, no fue posible contar con los resultados de la aplicación del instrumento.

4.4 Validación de los lineamientos (OE4)

Para confirmar la utilidad en entornos de desarrollo de los lineamientos propuestos se realizó una validación en dos niveles complementarios. Por una parte, se realizó una validación técnica basada en la alineación con el modelo de referencia establecido por CENS, y por otra la evaluación de su utilidad práctica y aplicabilidad por parte de los equipos de desarrollo que corresponden a los usuarios de los instrumentos desarrollados. Este doble enfoque permite asegurar que los instrumentos desarrollados no sean solo pertinentes sino también viables de aplicar en entornos reales de desarrollo con recursos y conocimientos limitados.

4.4.1 Alineación técnica con el modelo de calidad nacional

El primer paso de la validación consistió en evaluar la correspondencia de los elementos incluidos en el *checklist* de seguridad respecto a los criterios de evaluación del Sello de Calidad de Software en Salud de CENS, el cual está basado en la norma ISO 25010. Esta comparación se llevó a cabo en una matriz donde se enlazaron los distintos elementos propuestos con su contraparte del Sello permitiendo evaluar cuántos puntos de evaluación de CENS se encuentran cubiertos, a la vez de permitir detectar puntos en que el *checklist* tuviera una cobertura mayor a lo requerido. A través de esta revisión se objetivó que todos los elementos requeridos por el sello de calidad de software de CENS se encontraba cubierto en alguna medida por parte del *checklist* de seguridad, no obstante, también se identificó que la cobertura del *checklist* abarcaba elementos adicionales relacionados principalmente a componentes requeridos por la ley marco de ciberseguridad en relación a protocolos de seguridad de la información. Cabe destacar que esta metodología no fue utilizada con el Sello de Calidad para Registro Clínico Electrónico debido a que solo se obtuvo acceso parcial a la parte de usabilidad mediante un acuerdo de confidencialidad.

Por parte de la usabilidad, cabe destacar que el abordaje realizado por CENS tiene dos variantes según el sello a aplicar. En el caso de Calidad de Software se requiere presentar evidencia formal que incluye pruebas de usuario, análisis heurísticos, capturas de las maquetas de la interfaz para evaluar su evolución a lo largo del proyecto y todo otro elemento que permita demostrar un diseño centrado en el usuario. Mientras que para el Registro Clínico Electrónico se basa en la aplicación de una extensa pauta derivada de la NISTIR 7804 con un enfoque bastante granular a entornos hospitalarios que limitan su aporte a otras áreas de salud. Dadas estas singularidades de los instrumentos asociados al modelo de calidad CENS es que los lineamientos correspondientes a usabilidad no pudieron someterse a esta evaluación.

Para complementar esta validación, se realizaron múltiples reuniones con los integrantes del equipo de calidad de CENS que permitieran revisar la cobertura de los criterios de evaluación, con énfasis en las características del Sello de Calidad para Registro Clínico Electrónico basado en la ISO 27001. A través de esta retroalimentación se pudo ajustar los medios de verificación solicitados en 4 elementos buscando que puedan servir de evidencia para un eventual enfrentamiento del sello, además de recibir sugerencia que permitiesen aumentar la seguridad en la operación de los productos como lo son la ubicación de las estaciones de trabajo y el control de dispositivos externos a la institución. Cabe destacar que los expertos valoraron el potencial del instrumento para incorporar procesos de aseguramiento de calidad en equipos de desarrollo menos experimentados.

4.4.2 Validación por parte de usuarios

En relación con la otra arista de la validación se buscó la retroalimentación de los equipos de desarrollo y de proyecto relacionados a los productos de la red junto a la participación voluntaria de ingenieros y tecnólogos en informática médica. Este levantamiento buscó identificar si los lineamientos eran comprensibles, aplicables y útiles para quienes debiesen utilizarlos como herramienta para apoyar el desarrollo de software.

Esta encuesta fue aplicada a 26 profesionales voluntarios con perfiles diversos, tanto con cómo sin experiencia en el área de salud, recabando información estructurada que consideró un levantamiento de expertiz subjetiva para evaluar la preparación de los profesionales en relación con el diseño centrado en el usuario y la seguridad por diseño, denotándose que existe una mayor inclusión de cursos formales relacionados a seguridad en el pregrado, lo cual no sucede en cuanto a usabilidad (Figura 3). A nivel de perfiles participantes, la muestra fue variada con diversos niveles de experiencia y llegó a perfiles desde la gestión de proyectos, pasando por *frontend* y *backend*, hasta desarrolladores *fullstack*.

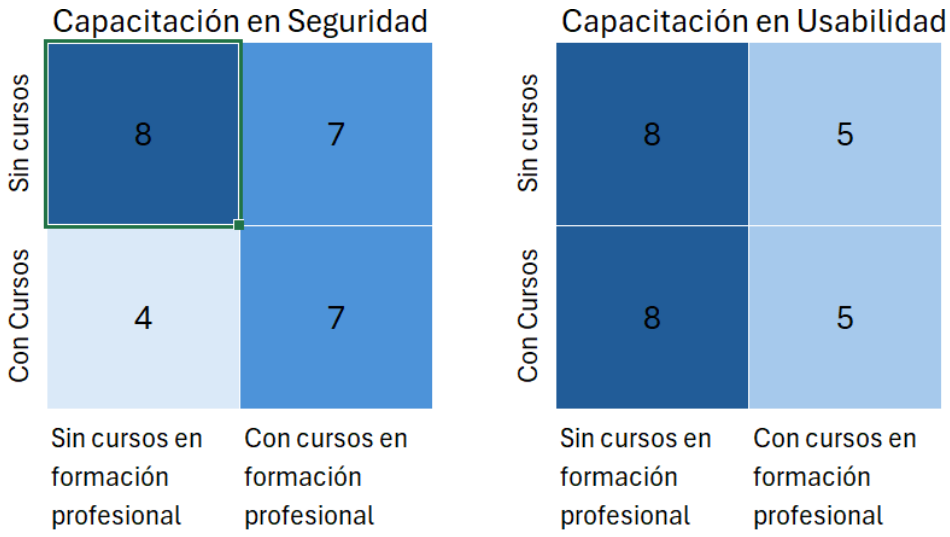

Figura 3: Mapas de calor de capacitación en seguridad (izquierda) y usabilidad (derecha). Cada gráfico muestra la distribución de participantes según su formación profesional y la realización de cursos adicionales, organizados en cuadrantes. El eje horizontal distingue si declararon haber recibido formación profesional en la dimensión respectiva, y el eje vertical si realizaron algún tipo de curso posterior. En seguridad, se observa una mayor proporción de participantes con capacitación en el área. En cambio, en usabilidad se evidencia una menor presencia de formación estructurada y un número considerable de personas que solo manejan nociones generales.

A partir de las respuestas se calcularon los indicadores planteados obteniendo los siguientes resultados, que se encuentran graficados en las figuras 4 a 6:

- Claridad del *checklist* de seguridad: El 65,4% de los participantes (17 de 26) consideró que los elementos del *checklist* eran claros o muy claros (puntaje 4 o 5 en escala Likert).

- Utilidad del *checklist* de seguridad: El 76,9% (20 de 26) indicó que el formato tipo *checklist* fue útil o muy útil para aplicar en sus contextos reales.
- Detección de aspectos nuevos de seguridad: Un 76,9% (20 de 26) i afirmó que el *checklist* les ayudó a identificar aspectos que no habían considerado previamente.
- Claridad de la guía de usabilidad: El 65,4% (17 de 26) calificó la guía como clara o muy clara.
- Utilidad percibida de la guía de usabilidad: Solo el 38,5% la consideró muy útil para mejorar desarrollos actuales o futuros, aunque solo 11.5 (3 de 26) la consideraron como que no sería útil.
- Uso futuro de los instrumentos: El 73,1% (19 de 26) indicó que recomendaría utilizar ambos instrumentos en proyectos futuros.

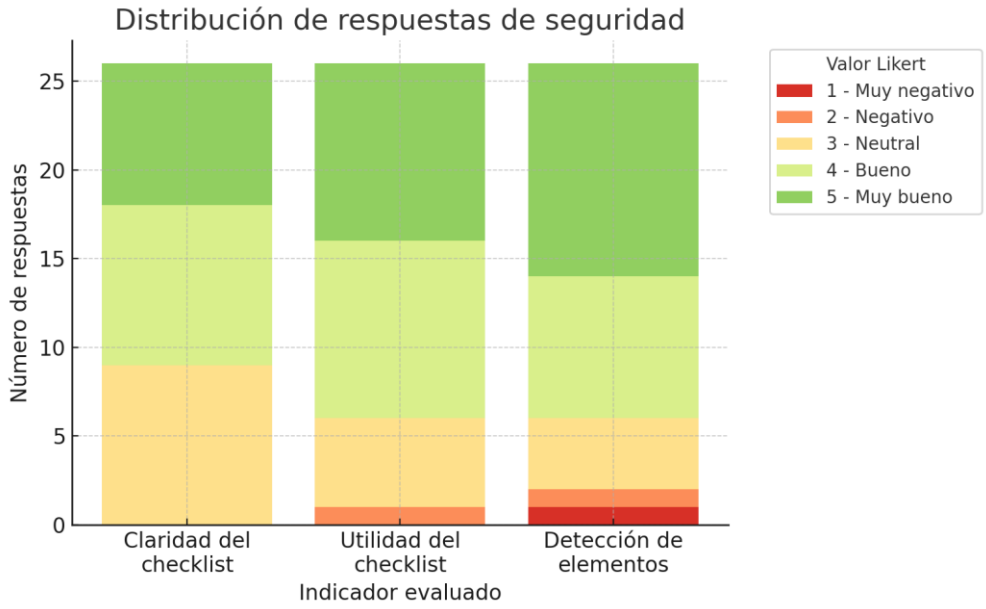

Figura 4: Gráfico de barras apiladas que muestra la distribución de respuestas para tres aspectos clave del checklist de seguridad: claridad, utilidad percibida y detección de aspectos nuevos. Cada color representa un valor de la escala Likert (1 a 5), desde rojo intenso (valor 1 – muy negativo) hasta verde intenso (valor 5 – muy bueno). Se observa una alta concentración de respuestas positivas (valores 4 y 5), especialmente en utilidad y detección, lo que evidencia una buena recepción del instrumento por parte de los participantes.

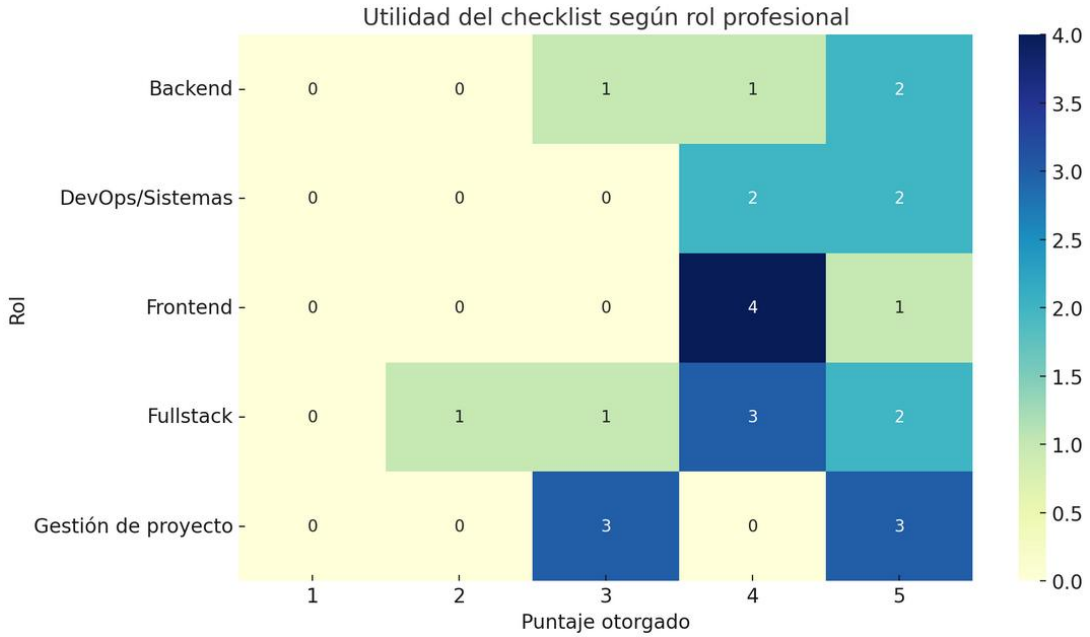

Figura 5: Mapa de calor que muestra cómo distintos perfiles valoraron la utilidad del *checklist* de seguridad. Cada celda representa la cantidad de participantes de un perfil profesional que asignó un puntaje específico (escala Likert de 1 a 5) al instrumento. Se aprecia que los perfiles concentran sus valoraciones en los niveles más altos (4 y 5) reflejando percepciones positivas sobre su utilidad práctica.

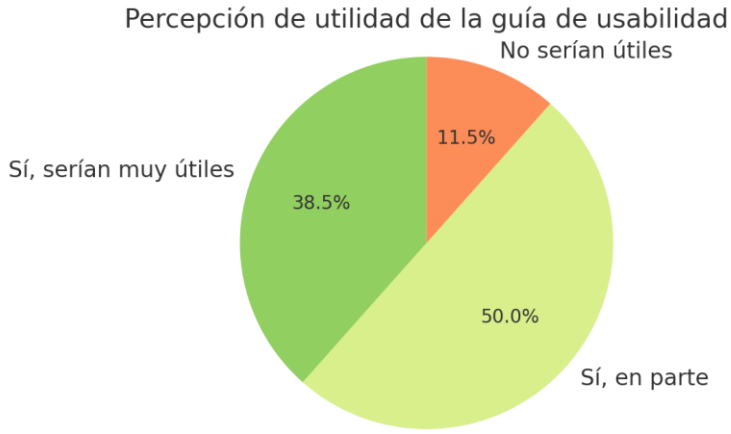

Figura 6: Gráfico de torta que representa la evaluación de los participantes respecto a la utilidad práctica de la guía de usabilidad. El 38,5% consideró que sería “muy útil”, mientras que el 50% la evaluó como “útil en parte”. Si bien la recepción general es positiva, estos resultados sugieren que aún existe margen de mejora para lograr una mayor aplicabilidad.

Como se puede apreciar, los resultados obtenidos evidencian una valoración positiva general respecto a los lineamientos propuestos, especialmente respecto del instrumento relacionado a seguridad. A nivel de retroalimentación, los encuestados desta-

caron el aporte de los instrumentos para generar sistemas robustos y útiles para los usuarios, absteniéndose en su mayoría de entregar sugerencias de mejora.

Adicionalmente, el análisis cualitativo de las respuestas abiertas de la encuesta permitió identificar un valor formativo asociado al *checklist* de seguridad. Varias respuestas señalaron que el instrumento evidenciando “aspectos que no se habían considerado antes”. Este hallazgo refuerza la utilidad del instrumento no solo como herramienta evaluativa, sino también como recurso pedagógico para equipos con experiencia limitada en ciberseguridad.

Como complemento a la encuesta, se realizó un ejercicio cualitativo de interpretación del *checklist* de seguridad y la guía de usabilidad. Para ello, dos profesionales evaluadores dedicados al desarrollo de software analizaron cada ítem e interpretaron su significado sin consultar entre ellos. Posterior a esta interpretación, se intercambiaron sus respuestas y actuaron de evaluador de la interpretación del otro profesional. Esta actividad buscó determinar la replicabilidad conceptual, es decir, si distintos desarrolladores entienden lo mismo a partir de los mismos lineamientos.

Los resultados mostraron un alto nivel de concordancia en los elementos del *checklist* de seguridad (89%), confirmando que el *checklist* utiliza un lenguaje técnico comprensible, basado en normas conocidas por el perfil profesional consultado. En cambio, la guía de usabilidad mostró menor claridad con una mayor dispersión interpretativa (68% de aceptación de la interpretación cruzada), lo que evidencia áreas de ambigüedad, especialmente en recomendaciones generales sin contexto específico.

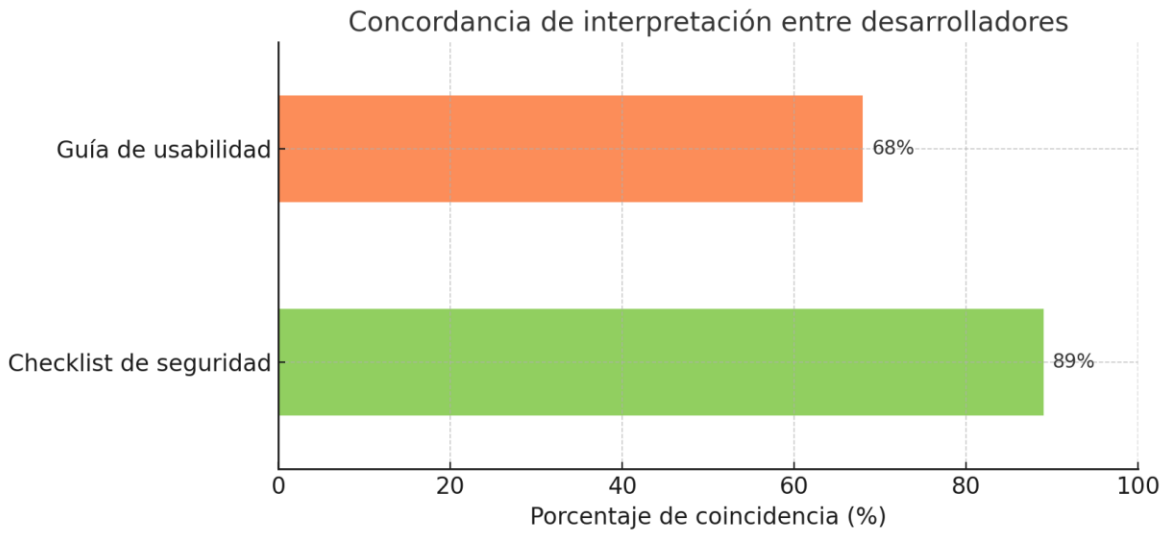

Figura 7: Gráfico de barras horizontales que muestra el porcentaje de coincidencia entre pares de evaluadores al interpretar y aplicar los instrumentos de calidad. El *checklist* de seguridad alcanzó una concordancia del 89%, evidenciando una redacción clara y compartida entre los evaluadores. En contraste, la guía de usabilidad obtuvo un 68% de coincidencia, que si bien es superior a la mitad sugiere un mayor grado ambigüedad en los lineamientos presentados. Esta diferencia refuerza la idea de que en usabilidad se requiere una mayor especificidad para presentar un instrumento de utilidad a los desarrolladores.

El *focus group*, complementario a la validación técnica de los lineamientos de calidad, se realizó con tres integrantes del equipo de proyecto de la RSDUE: los responsables del Gestor de Casos Clínicos y del Registro Clínico Electrónico, junto a una colaboradora transversal. Esta instancia permitió observar una evolución en la comprensión de las dimensiones de seguridad y usabilidad por parte del equipo.

Dentro de la conversación se transmitió que, al inicio del proyecto, la seguridad se concebía como un aspecto externo vinculado más a la explotación que al diseño del sistema. La usabilidad, por otro lado, se abordaba de manera intuitiva, sin una estructura formal de evaluación ni lineamientos definidos. A medida que avanzó el desarrollo, y especialmente al considerar enfrentarse al proceso de certificación, el equipo reconoció la necesidad de integrar estos aspectos desde etapas tempranas, adoptando prácticas como la documentación de pruebas de seguridad o la revisión crítica de la experiencia de usuario.

Un aspecto destacado del diálogo fue la identificación de tensiones prácticas entre seguridad y usabilidad. Por ejemplo, se discutió cómo la autenticación en dos pasos,

aunque altamente recomendada, puede resultar poco viable en entornos donde los usuarios priorizan la agilidad por sobre la robustez técnica. Este tipo de decisiones operativas puso en evidencia la necesidad de lineamientos que permitan evaluar alternativas viables en función del contexto.

Además, surgió una autocrítica respecto al levantamiento de requerimientos. Se reveló que, en las etapas iniciales, los aspectos no funcionales como la seguridad o la usabilidad fueron subvalorados o considerados implícitos, lo que llevó a ajustes posteriores. Esta experiencia reforzó la importancia de incorporar estas dimensiones como parte estructural del diseño, y no solo como exigencias normativas.

Finalmente, se acordó la necesidad de contar con ejemplos prácticos que orienten la aplicación de los lineamientos. Otra de las observaciones del equipo de la RSDUE fue que tuvieron que dedicar tiempo a interpretar los requerimientos y las posibles soluciones, por lo que se valoró la idea de construir repositorios de buenas prácticas o casos de uso que permitan guiar decisiones frente a dilemas comunes del desarrollo, especialmente cuando se debe equilibrar calidad, contexto y capacidades del equipo.

5. Discusión

El desarrollo de software en salud implica desafíos complejos que van más allá de su aplicación técnica. Las dimensiones de seguridad y usabilidad, reconocidas internacionalmente como pilares de la calidad del software, adquieren una relevancia aún mayor en contextos clínicos, donde los errores pueden tener consecuencias directas en la atención a pacientes. A continuación, se procede a analizar los resultados obtenidos a lo largo de este proyecto.

5.1 Relación entre normativas y realidades operativas locales

La implementación de lineamientos de calidad basados en normas reconocidas internacionalmente, como ISO 25010:2023 para calidad de software e ISO 27001 para seguridad de la información, representa una estrategia ampliamente respaldada por la literatura. Estas normativas entregan un marco robusto para la evaluación de aspectos funcionales y no funcionales del software, permitiendo establecer criterios objetivos para su desarrollo, implementación y certificación. Autores como Ronchieri et al. (2023) y Al-Kilidar et al. (2005) han demostrado que la adopción de estos modelos en fases tempranas de desarrollo mejora la calidad, reduce costos de retrabajo y facilita la interoperabilidad y escalabilidad de los sistemas.

No obstante, la aplicación práctica de estos modelos en entornos de desarrollo con restricciones de recursos, como los observados en este estudio, evidencia una desconexión entre el ideal normativo y la capacidad operativa de los equipos técnicos. Este hallazgo coincide con la experiencia documentada por Braa et al. (2007), donde la aplicación de normativas internacionales requiere adaptaciones significativas y una estrategia incremental basada en las capacidades existentes.

En el caso del equipo de la RSDUE, si bien manifestaron apertura a incorporar los lineamientos propuestos, su adopción efectiva resultó posterior al inicio de los desarrollos requiriendo iteraciones adicionales para alcanzar el nivel de madurez requerido y la obtención de evidencia de su cumplimiento. Esto sugiere que los modelos de

calidad deben ser contextualizados y acompañados de estrategias de capacitación a los equipos de desarrollo y no solo exigibles a nivel de explotación de software.

5.2 Seguridad por diseño

El principio de seguridad por diseño (*security by design*) plantea que las consideraciones de protección de la información no deben incorporarse como capas accesorias, sino como un eje estructurante desde el inicio del desarrollo del software (Fauzi et al., 2023; CISA, 2024). Esta aproximación ha sido promovida activamente por normativas internacionales como la ISO/IEC 27001 y por organismos como la CISA de Estados Unidos, en respuesta a los crecientes riesgos de ciberseguridad en el sector salud.

Sin embargo, los hallazgos de este trabajo revelan que, en contextos como el de la RSDUE, la seguridad sigue siendo percibida como una función técnica separada, más asociada al despliegue o al soporte que al diseño del sistema. Por ejemplo, en las entrevistas con los equipos de desarrollo, fue común encontrar una visión limitada de la seguridad, enfocada en aspectos como el uso de contraseñas robustas, firewalls o cifrado en tránsito, pero sin una comprensión profunda de conceptos como trazabilidad, control de accesos granular o cifrado en reposo.

La aplicación del *checklist* de seguridad permitió visibilizar estas brechas y operar como un instrumento formativo. Más allá de evaluar, sirvió para introducir buenas prácticas, documentar ausencias críticas, como la falta de pruebas de seguridad o protocolos para abordaje de incidentes, y priorizar mejoras con los equipos de desarrollo. La experiencia confirma lo descrito por Hayam (1994) quien describe que la auditoría formal no es suficiente si no va acompañada de procesos internos que permitan prevenir vulnerabilidades desde el diseño.

Asimismo, es relevante señalar que no todas las sugerencias planteadas por los expertos del área de calidad del CENS fueron incorporadas en los instrumentos desarrollados. Si bien se reconoció su pertinencia técnica y alineación con estándares de

seguridad, muchas de estas recomendaciones resultaban excesivamente específicas o exigentes para el nivel de madurez de los desarrollos y las capacidades de los equipos involucrados. Si estas son incorporarlas sin adaptación habría significado elevar innecesariamente la barrera de entrada para equipos de desarrollo nuevos, contradiciendo el fomento de la innovación desde entornos académicos y abiertos, a través de la incorporación de nuevos actores que puedan pensar “fuera de la caja”.

En este sentido, se optó por una estrategia de lineamientos basada en el principio 80/20 en que se priorizan aquellas prácticas de seguridad que representan un alto impacto en la protección de datos y la integridad del sistema, sin exigir una cobertura absoluta o de difícil implementación. Este enfoque busca ser un aporte a los equipos de desarrollo, entregando claridad, orientación y mejora continua, pero sin convertirse en una carga que inhiba la experimentación o la evolución progresiva del software. Como ha sido señalado en literatura reciente, los marcos de calidad que no se adaptan a los contextos reales pueden actuar como freno a la innovación, especialmente cuando se trata de desarrollos emergentes y orientados al bien público (Braa et al., 2007; Gallagher, 2013).

5.3 Diseño centrado en la utilidad clínica

Uno de los aprendizajes más relevantes de este trabajo fue constatar que la usabilidad no puede ser entendida exclusivamente desde métricas o principios normativos; su eficacia real se determina por la manera en que el sistema se alinea con los flujos de trabajo de sus usuarios. En el caso de los desarrollos acompañados, las pruebas internas del Gestor de Casos Clínicos mostraron que la plataforma permitía reconocer de forma casi intuitiva el uso del sistema, facilitando los flujos de creación y uso de los casos clínicos. En cambio, a nivel del RCE, que se encuentra todavía en desarrollo, la versión explorada no permitía concretar flujos de atención y carecía opciones o visualizaciones de datos familiares a los usuarios, por lo cual fue posible reconocer de forma casi inmediata si el sistema respondía a lo esperado inconscientemente por el usuario de pruebas.

Esta paradoja ha sido ampliamente descrita en la literatura, bajo la reconocida frase de que los usuarios no siempre saben especificar lo que necesitan, pero detectan con rapidez aquello que interrumpe o facilita su trabajo (Zahabi et al., 2015; Kaipio et al., 2017). Por ello, un diseño centrado en el usuario no debe basarse solo en requerimientos explícitos, sino en la observación, acompañamiento y prototipado en colaboración con ellos. Esto demanda pasar de un modelo consultivo a uno verdaderamente participativo, donde los usuarios buscan mejoras en sus flujos de trabajo, además de la estética o navegación del sistema, volviéndose estos una condición para la adopción y efectividad del sistema.

Esta lógica se alinea con los marcos de diseño centrado en el usuario del NIST (Lowry et al., 2012) y con estudios que vinculan la usabilidad con la seguridad del paciente y la eficiencia clínica (Ratwani et al., 2015). En este sentido, se reafirma que un sistema usable no es el que cumple con todos los principios, sino el que se adapta a la lógica del usuario sin que este tenga que adaptarse a él.

Por tanto, la inclusión temprana de usuarios clínicos no es un aporte metodológico, sino una estrategia para asegurar pertinencia funcional, minimizar el rechazo al cambio y maximizar la adopción. Esta participación, sin embargo, debe institucionalizarse más allá de voluntades individuales, convirtiéndose en un componente estructural de los ciclos de desarrollo de software en salud.

5.3 Evaluación para la mejora

Otro aspecto relevante fue comprender la aplicación de los instrumentos no como un acto puntual ni como un proceso de fiscalización, sino como una herramienta de aprendizaje colectivo. La evaluación del software con el *checklist* relacionado a seguridad y la guía de buenas prácticas para usabilidad, complementada con la aplicación de instrumentos como SUS y PSSUQ, sirvió no solo para medir el cumplimiento, sino para activar procesos de mejora iterativa dentro de los equipos (Maramba et al., 2019; Farzandipour et al., 2022). Este enfoque se alinea con lo propuesto por Green-

halgh et al. (2017) quienes sugieren que la implementación de tecnología en salud debe ser adaptativa, centrada en los actores y abierta al rediseño permanente.

En este sentido, la validación final aplicada a los instrumentos propuestos fue diseñada con esa lógica, evaluando no solo la pertinencia de la herramienta respecto de los modelos de calidad, sino también su capacidad de aportar al equipo de desarrollo. Este, probablemente, es uno de los aportes más valiosos del presente trabajo.

5.4 Evaluación de los instrumentos desarrollados

En relación con los instrumentos propuestos para apoyo a los desarrolladores, su evaluación por parte de profesionales del área de desarrollo evidencia una recepción favorable especialmente para la dimensión de seguridad. El *checklist* de seguridad alcanzó niveles destacables de utilidad (76,9%) y capacidad de aportar con nuevos conocimientos, lo que confirma su aplicabilidad práctica. Además, se obtuvo una concordancia en su interpretación del 89%, lo que refuerza su utilidad dentro de los equipos de desarrollo sentando un piso de requerimientos para el desarrollo de software en salud. Esto es coherente con lo planteado por Glinz (2007) y Wieringa (2014), quienes destacan que los instrumentos son más fácilmente adoptados cuando presentan estructura clara, consistencia terminológica y alineación con estándares conocidos, favoreciendo la integración en procesos reales de desarrollo. En este sentido, el hecho de que el *checklist* se encuentre alineado con marcos normativos como las normativas ISO 25010 y 27001 facilita su comprensión y aplicabilidad al brindar un lenguaje común ya familiar para los desarrolladores.

En cambio, la guía de usabilidad, a pesar de ser considerada clara por un 65,4%, fue valorada como “muy útil” solo por el 38,5% de los encuestados. Esta brecha sugiere que el contenido, aunque comprensible, no logra transformar fácilmente la teoría en práctica efectiva. Este problema de “ambigüedad semántica” afecta particularmente en lineamientos generales que no se contextualizan según especialidad clínica, perfil de usuario o etapa del flujo de trabajo. La falta de segmentación específica en la guía “Haz y Evita” puede haber dificultado su conexión con la práctica cotidiana de los

desarrolladores, no obstante, esto plantea un desafío mayor ya que la generación de un instrumento detallado con ejemplos para diferentes casos se podría traducir en un manual de consulta respecto de recomendaciones de usabilidad, cuya extensión podría hacerlo poco atractivo para los equipos de desarrollo. Realizando una revisión de la utilidad de los lineamientos nos encontramos con Zhang & Walji (2011), quienes a través de su marco de trabajo “TURF” enfocado en usabilidad de RCE señalan que la usabilidad debe incluir una dimensión de utilidad proveniente de las funcionalidades asociadas, reconociendo la importancia de estas para el usuario final. En este sentido, para contar con un instrumento de apoyo a usabilidad se debiera recoger además la diversidad de contextos clínicos y perfiles de usuario, facilitando no solo el diseño de la interfaz a través de la traducción de principios abstractos en acciones concretas, sino también potenciando el levantamiento de requerimientos funcionales claves para el sistema para que estos cubran las necesidades de los usuarios como base para la usabilidad.

La diferencia en la percepción entre ambos instrumentos refuerza la importancia de no solo diseñar herramientas técnicas, sino que estas deben ser validadas con quienes las utilizarán, debiendo ser mejoradas en base a su uso en entornos reales tal como proponen Kushniruk y Borycki (2015). Si bien los instrumentos fueron bien acogidos por la RSDUE como base de trabajo, se identifica la necesidad de refinar la guía de usabilidad, incorporando criterios accionables y ejemplos contextualizados para distintas realidades de uso.

Finalmente, el hecho de que el 73,1% de los participantes recomendaría el uso de ambos instrumentos en futuros proyectos, indica una disposición positiva a su adopción, lo que valida el enfoque multidimensional planteado. Esta integración entre seguridad y usabilidad como dimensiones de calidad relevantes consistente con lo establecido en normativas como la ISO 25010, y a su vez son un aporte para conseguir una adecuada implementación de RCE en hospitales como parte del enfoque de planteado por Boonstra et al. (2014).

Por su parte, la evaluación cualitativa mediante interpretación cruzada entre profesionales demostró ser una herramienta útil para validar la claridad semántica de los instrumentos. Esta aproximación puede considerarse un aporte metodológico, en línea con propuestas de validación de contenido desde la ingeniería de requisitos, las que deben realizarse para una correcta transferencia tecnológica desde la academia (Gorschek et al., 2006).

5.5 Aprendizajes desde la experiencia del equipo RSDUE

El focus group realizado con integrantes del equipo de la RSDUE aportó una visión desde la experiencia práctica de integrar seguridad y usabilidad en desarrollos de software en salud. Uno de los principales hallazgos fue que los lineamientos propuestos, aunque valorados como pertinentes, exigieron una inversión de tiempo relevante para ser comprendidos e interpretados adecuadamente por el equipo. No se trató de aplicar mecánicamente un *checklist*, sino de internalizar el propósito de cada elemento, comprender sus implicancias técnicas y metodológicas, y traducirlos a decisiones concretas de desarrollo. Esta experiencia genera una proyección para este trabajo: la necesidad de contar, en el futuro, con ejemplos prácticos de implementación que sirvan como referentes orientadores. No se busca establecer una receta universal, lo cual sería improcedente dadas las diferencias entre tipos de software, contextos de uso y perfiles de usuario, sino mas bien de visibilizar enfoques diversos adoptados en otros desarrollos. Es decir, mostrar cómo distintas soluciones abordaron una misma necesidad desde puntos de vista diferentes como por ejemplo el reforzar las contraseñas en un sistema educativo en que los stakeholders no consideran adecuada la implementación de la autenticación en dos factores, mientras que este si es un elemento relevante en entornos clínicos para reforzar la seguridad a nivel de acceso; o priorizando flujos de navegación más directos frente a procedimientos estandarizados más restrictivos. Este tipo de repositorio de experiencias permitiría a nuevos equipos comprender que la calidad no es una meta única, sino un equilibrio dinámico entre exigencias técnicas, capacidades del equipo y necesidades reales del usuario final, dando a su vez una base sobre la cual discutir alternativas de abordaje.

Otro aspecto relevante que surgió de la conversación fue la autocrítica respecto al proceso de levantamiento de requerimientos. El equipo reconoció que, en las etapas iniciales del proyecto, se dio prioridad casi exclusiva a los requerimientos funcionales, con una lógica centrada en “qué debe hacer el sistema”. Las dimensiones no funcionales, como lo son la seguridad, la usabilidad o el rendimiento, fueron tratadas de forma implícita, muchas veces asumidas como evidentes o postergadas para fases avanzadas del desarrollo. Esta visión aunque comprensible en entornos con tiempos o recursos limitados, tuvo consecuencias concretas: la necesidad de volver a realizar algunos requerimientos, enfrentar observaciones en procesos de certificación, o dificultades para cumplir con expectativas de uso real. A partir de esta experiencia, el equipo identificó como aprendizaje clave la necesidad de integrar los requerimientos no funcionales desde el inicio del ciclo de vida del software, y no como anexos o “extras” del producto. Además, se visibilizó el valor de contar con instrumentos que orienten la incorporación de las dimensiones de calidad como parte estructural del diseño, y no solo como normativas externas, denotando la necesidad de promover una cultura de calidad desde el diseño en el desarrollo de software en salud.

5.6 Desafío de incorporar la calidad

Pese a los avances logrados, persisten desafíos estructurales importantes en la incorporación efectiva de criterios de calidad en el desarrollo de software en salud. En particular, se observa que los lineamientos propuestos deben integrarse como parte del trabajo cotidiano de los equipos técnicos. Esta dificultad se vincula tanto a la ausencia de roles institucionalizados, como responsables de calidad o perfiles especializados en experiencia usuaria, como a la falta de mecanismos formales, simples y accesibles de evaluación continua.

En el ámbito de la ciberseguridad, algunos de estos vacíos están comenzando a abordarse parcialmente mediante la figura del Encargado de Seguridad de la Información “CISO”, requerida por la reciente Ley Marco de Ciberseguridad (Ministerio

Interior, 2024). No obstante, esta solución normativa no se traduce automáticamente en capacidades instaladas en entornos como universidades o centros de salud, donde los desarrollos suelen realizarse con recursos limitados y alta rotación de personal. Esta situación refleja lo señalado por Gallagher (2013), reconociendo que las políticas de aseguramiento de calidad pueden fracasar si no se acompañan de estrategias formativas y operativas que faciliten su adopción efectiva.

En este contexto, la academia tiene un rol crítico y aún subutilizado respecto de formar profesionales con una cultura de calidad incorporada desde la base, que no vean la seguridad y la usabilidad como exigencias externas o burocráticas, sino como parte esencial del diseño de software. Para ello, es necesario integrar en las mallas curriculares contenidos relacionados a normativas como la ISO 25010, ISO 27001, junto con experiencias prácticas de evaluación de usabilidad y seguridad de software.

Como se ha podido apreciar, los resultados de este trabajo confirman que los instrumentos diseñados, especialmente el *checklist* de seguridad, tienen un potencial transformador en entornos de desarrollo incipientes, al traducir marcos normativos complejos en herramientas prácticas, comprensibles y accionables. Esta capacidad de "puente" entre el estándar formal y la práctica cotidiana es clave para cerrar la brecha de adopción de calidad en sistemas de información en salud, particularmente en contextos académicos. En este sentido, los lineamientos desarrollados no deben entenderse únicamente como herramientas técnicas de evaluación, sino como insumos habilitantes para procesos de aprendizaje, mejora continua e instalación de una cultura de calidad. Tal como plantea Greenhalgh et al. (2017), la implementación exitosa de tecnologías en salud requiere marcos adaptativos que consideren la interacción entre tecnologías, actores y contextos organizacionales, más allá de la adopción normativa. En este sentido, los instrumentos generados en este trabajo permiten visibilizar brechas, activar reflexiones dentro de los equipos de desarrollo y ofrecer una vía progresiva hacia estándares de mayor exigencia, sin desincentivar la innovación en entornos con capacidades limitadas.

6. Conclusión

El presente trabajo permitió avanzar en la integración práctica de las dimensiones de seguridad y usabilidad en el desarrollo de software en salud, particularmente en el contexto académico y público representado por la RSDUE. A partir del análisis normativo y técnico, y mediante un enfoque centrado en la realidad operativa de los equipos de desarrollo, se diseñaron e implementaron instrumentos que contribuyen a traducir estándares complejos en herramientas comprensibles y aplicables.

El *checklist* de seguridad se consolidó como una herramienta especialmente útil, no solo para verificar cumplimiento normativo, sino para facilitar el reconocimiento progresivo de los elementos que componen un sistema seguro. Este instrumento permitió a los equipos de proyecto identificar riesgos, fortalecer prácticas técnicas y comprender por qué ciertos requisitos, como el cifrado, el control de accesos o la trazabilidad, son fundamentales cuando se manejan datos personales y sensibles en entornos clínicos. Su estructura clara y autoaplicable facilitó su adopción en contextos con experiencia limitada, convirtiéndolo en una herramienta formativa tanto como evaluativa.

En contraste, la dimensión de usabilidad presentó desafíos importantes en términos de validación. Si bien se elaboró una guía basada en buenas prácticas y se realizó la recomendación de aplicar instrumentos como SUS y PSSUQ, la valoración por parte de los desarrolladores fue menor en comparación con la herramienta de seguridad, y se observó una menor claridad en su interpretación. Esta situación evidencia que, a diferencia del *checklist* de seguridad, la guía de usabilidad no logró la misma eficacia para transformarse en una herramienta práctica, probablemente debido a su carácter más genérico y la dificultad de adaptarse a contextos clínicos diversos sin ejemplos concretos. Por ello, este instrumento debe entenderse como una versión inicial, susceptible de mejora, y cuya evolución futura requerirá mayor segmentación por tipo de usuario, especialidad y entorno clínico, así como una validación más robusta con usuarios finales.

En base a lo observado, la calidad del software no puede depender exclusivamente de marcos normativos o certificaciones externas. Debe construirse desde una cultura técnica e institucional que valore la seguridad por diseño y el diseño centrado en el usuario como principios esenciales. Para ello, resulta urgente fortalecer la formación en estas materias. Si bien algunos participantes del estudio reportaron haber cursado instancias de capacitación, estas fueron mayoritariamente teóricas u online, con escasas oportunidades de aplicación práctica. En áreas como la usabilidad, donde la comprensión requiere exploración, prueba y error, el aprendizaje teórico no basta: se necesita experiencia directa y acompañamiento metodológico.

Finalmente, este trabajo deja instalada una base concreta sobre la cual continuar iterando y mejorando. Los instrumentos desarrollados no son soluciones cerradas, sino herramientas abiertas que pueden ser adaptadas, ampliadas y refinadas. Más importante aún, aportan un marco para fomentar el desarrollo de software que no solo sea técnicamente correcto, sino también útil, usable y alineado con las necesidades reales de quienes lo utilizan.

En este sentido, se espera que los lineamientos aquí propuestos sirvan como punto de partida para una cultura de calidad en salud digital, especialmente en contextos académicos y públicos, donde los recursos pueden ser limitados. A partir de la propuesta realizada, futuras iniciativas podrían enfocarse en documentar y compartir enfoques de implementación, generando repositorios de buenas prácticas contextualizadas que orienten a nuevos equipos frente a desafíos similares. De esta forma, este trabajo no solo busca mejorar software, sino también contribuir a una salud digital más confiable, segura y amigable.

6.1 Síntesis de hallazgos y protecciones

El presente trabajo permitió avanzar en la integración práctica de las dimensiones de **seguridad** y **usabilidad** en el desarrollo de software en salud, con especial énfasis en el contexto académico y público representado por la Red de Salud Digital de las

Universidades del Estado (RSDUE). A continuación, se presentan las principales conclusiones:

1. **Traducción operativa de estándares complejos:**
   A partir del análisis normativo y técnico, se logró diseñar e implementar instrumentos que facilitan la aplicación de estándares de seguridad y usabilidad en entornos de desarrollo reales, transformándolos en herramientas comprensibles, aplicables y alineadas con las capacidades de los equipos técnicos, facilitando su adopción efectiva.
2. **Valor del *checklist* de seguridad como herramienta formativa y evaluativa:**
   El *checklist* de seguridad demostró ser especialmente útil para apoyar a los equipos en la identificación de riesgos, la mejora de prácticas técnicas y la comprensión de principios fundamentales como el cifrado, el control de accesos y la trazabilidad. Su formato claro y autoaplicable facilitó su adopción incluso en equipos con experiencia limitada, consolidándose como una herramienta tanto pedagógica como de control de cumplimiento.
3. **Limitaciones en la adopción de la guía de usabilidad:**
   A pesar de contar con fundamentos sólidos y basarse en buenas prácticas, la guía de usabilidad no alcanzó el mismo nivel de apropiación ni claridad que el *checklist* de seguridad. La menor valoración por parte de los desarrolladores sugiere que se requieren versiones más específicas, adaptadas a distintos perfiles de usuario y contextos clínicos, con ejemplos concretos y validación directa con usuarios finales.
4. **Importancia de una cultura de calidad más allá del cumplimiento normativo:**
   La calidad del software en salud no puede depender únicamente de certificaciones externas o normativas. Se necesita consolidar una **cultura técnica e institucional** que promueva la **seguridad por diseño** y el **diseño centrado en el usuario** como principios centrales del desarrollo. Para ello, es clave fortalecer la formación práctica en estas áreas. Esta cultura debe ser promovida tanto desde las instituciones como desde la formación profesional

5. **Déficit en la formación aplicada en seguridad y usabilidad:**
   Si bien algunos participantes reportaron haber recibido capacitación, esta fue en su mayoría teórica u online, con escasas instancias prácticas. En temas como la usabilidad, donde la comprensión requiere ensayo, error y experiencia directa, el aprendizaje exclusivamente teórico resulta insuficiente, por lo que se requiere fomentar experiencias formativas más aplicadas, centradas en el desarrollo y evaluación de sistemas reales.
6. **Aportes y proyección de los instrumentos desarrollados:**
   Los instrumentos propuestos no deben entenderse como soluciones finales, sino como **bases funcionales** que pueden ser **adaptadas, ampliadas y refinadas**. Representan un punto de partida para fortalecer el desarrollo de software clínico que sea no solo técnicamente robusto, sino también seguro, útil y centrado en las necesidades reales de los usuarios.
7. **Hacia una cultura de calidad en salud digital pública:**
   Este trabajo busca contribuir al desarrollo de una **salud digital más confiable, segura y amigable**, particularmente en contextos académicos y públicos con recursos limitados. Se espera que los lineamientos propuestos sirvan como base para futuras iniciativas orientadas a documentar, compartir y contextualizar buenas prácticas, generando conocimiento reutilizable por otros equipos frente a desafíos similares, y promoviendo el diseño centrado en el usuario y la seguridad como pilares de desarrollo de software en salud.

Bibliografía

Al-Kilidar, H.; Parkin, P.; Aurum, A.; Jeffery, R. “Evaluation of effects of pair work on quality of designs”. Proc Aust Softw Eng Conf. 2005:78–87. 10.1109/ASWEC.2005.24

Apple Inc. “UI design dos and don’ts”. Apple Developer; 2017. Disponible en: https://developer.apple.com/design/tips/

Boonstra, A.; Versluis, A.; Vos, J.F. “Implementing electronic health records in hospitals: A systematic literature review”. BMC Health Serv Res. 2014;14:370. 10.1186/1472-6963-14-370

Braa, J.; Monteiro, E.; Sahay, S. “Networks of action: Sustainable health information systems across developing countries”. MIS Q. 2007;31(2):337–362. 10.2307/25148643

Centro Nacional en Sistemas de Información en Salud (CENS). “Sobre CENS”. 2016. Disponible en: https://cens.cl/sobre-cens-2/

Centro Nacional en Sistemas de Información en Salud. (2022). Sello Calidad: Software en salud. https://cens.cl/sello-calidad-software-en-salud

Centro Nacional en Sistemas de Información en Salud. (2022). Convocatoria Programa de Pre-sello de Calidad Registro Clínico Electrónico: Usabilidad y Seguridad. https://cens.cl/wp-content/uploads/2022/04/Programa-Pre-Sello-Calidad-Registro-Clinico-Electronico_-Usabilidad-y-Seguridad.pdf

Centro Nacional en Sistemas de Información en Salud. (2023). Sello de Registro Clínico Electrónico. https://cens.cl/sello-calidad-registro-clinico-electronico

Cybersecurity & Infrastructure Security Agency (CISA). “CISA Secure by Design Pledge”. 2024. Disponible en: www.cisa.gov/resources-tools/resources/cisa-secure-design-pledge

Deloitte Centre for Health Solutions. (2015) How digital technology is transforming health and social care. www2.deloitte.com/uk/en/pages/life-sciences-and-healthcare/articles/connected-health.html.

Gobierno Digital. (2021). Lineamientos para el desarrollo de software en instituciones públicas. Secretaría General de la República de Chile. https://digital.gob.cl/transformacion-digital/estandares-y-guias/guia-desarrollo-software

European Parlament (2016) General Data Protection Regulation GDPR. Official Journal of the European Union https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679

Farzandipour, M.; Nabovati, E.; Sadeqi Jabali, M. “Comparison of usability evaluation methods for a health information system: Heuristic evaluation versus cognitive walkthrough method”. BMC Med Inform Decis Mak. 2022;22(1):157. 10.1186/s12911-022-01905-7

Fauzi, M.F.; Mohan, V.R.; Qi, Y.; Chandrasegar, C.; Muzafar, S. “Secure software development: Best practices”. Int J Emerg Multidiscip Comput Sci Artif Intell. 2023;2(1). 10.54938/ijemdcsai.2023.02.1.256

Gallagher, M.E. “Improving software sustainability: Lessons learned from Profiles in Science”. IS&T Archiving Conf Proc. 2013:74–79. https://pubmed.ncbi.nlm.nih.gov/25267934/

Glinz, M. “On non-functional requirements”. IEEE Int Req Eng Conf Proc. 2007:21–26. 10.1109/RE.2007.45.
Google Inc. “Material Design 3”. 2021. Disponible en: https://m3.material.io/
Gorschek, T.; Garre, P.; Larsson, S.; Wohlin, C. “A model for technology transfer in practice”. IEEE Softw. 2006;23(6):88–95. 10.1109/MS.2006.147.
Greenhalgh, T.; Wherton, J.; Papoutsi, C.; Lynch, J.; Hughes, G. “Beyond adoption: A new framework for theorizing and evaluating nonadoption, abandonment, and challenges to the scale-up, spread, and sustainability of health and care technologies”. J Med Internet Res. 2017;19(11):e367. 10.2196/jmir.8775
Hayam, A. “Security audit center: A suggested model for effective audit strategies in health care informatics”. Int J Biomed Comput. 1994;35(Suppl):115–127.
International Organization for Standardization. ISO 9000:2015 – Quality management systems – Fundamentals and vocabulary. ISO; 2015.
International Organization for Standardization. ISO 27799:2016 – Health informatics – Information security management in health using ISO/IEC 27002. ISO; 2016.
International Organization for Standardization. ISO 9241-11:2018 – Ergonomics of human-system interaction – Usability: Definitions and concepts. ISO; 2018.
International Organization for Standardization. ISO/IEC 27001:2022 – Information security, cybersecurity and privacy protection – Information security management systems – Requirements. ISO; 2022.
International Organization for Standardization. ISO/IEC 25010:2023 – Systems and software quality models. ISO; 2023.
Kaipio, J.; Lääveri, T.; Hyppönen, H.; Vainiomäki, S.; Reponen, J.; Kushniruk, A.; Vänskä, J. “Usability problems do not heal by themselves: National survey on physicians’ experiences with EHRs in Finland”. Int J Med Inform. 2017;97:266–281. 10.1016/j.ijmedinf.2016.10.010
Kulikowski, C.A.; Shortliffe, E.H.; Currie, L.M.; Elkin, P.L.; Hunter, L.E.; Johnson, T.R. et al. “AMIA Board white paper: Definition of biomedical informatics and specification of core competencies for graduate education in the discipline”. J Am Med Inform Assoc. 2012;19(6):931–938. 10.1136/amiajnl-2012-001053.
Kushniruk, A.W.; Borycki, E.M. “Integrating low-cost rapid usability testing into agile system development of healthcare IT: A methodological perspective”. Stud Health Technol Inform. 2015;210:200–204.
Lowry, S.Z.; Quinn, M.T.; Ramaiah, M.; Schumacher, R.M.; Patterson, E.S.; North, R.; et al. “Technical evaluation, testing and validation of the usability of electronic health records”. NIST Interagency/Internal Report (NISTIR 7804); 2012. 10.6028/NIST.IR.7804
Maramba, I.; Chatterjee, A.; Newman, C. “Methods of usability testing in the development of eHealth applications: A scoping review”. Int J Med Inform. 2019;126:95–104. 10.1016/j.ijmedinf.2019.03.018
Ministerio de Salud de Chile. (2006). El libro azul: agenda digital del Ministerio de Salud. Departamento de Agenda Digital en Salud del Ministerio de Salud, Gobierno de Chile. https://ws.studylib.es/doc/4921722/libro-azul.p65---observatorio-digital
Ministerio de Salud de Chile. (2012, diciembre). Decreto 41 - APRUEBA REGLAMENTO SOBRE FICHAS CLÍNICAS. Biblioteca del Congreso Nacional. www.bcn.cl/leychile/navegar?i=1046753

Ministerio de Salud de Chile. (2012, abril). Ley 20584 - REGULA LOS DERECHOS Y DEBERES QUE TIENEN LAS PERSONAS EN RELACIÓN CON ACCIONES VINCULADAS A SU ATENCIÓN EN SALUD. Biblioteca del Congreso Nacional. www.bcn.cl/leychile/navegar?idNorma=1039348

Ministerio de Salud de Chile. (2015). SIDRA - Sistemas Información de Red Asistencial. Ministerio de Salud. www.minsal.cl/SIDRA

Ministerio de Salud de Chile. (2020). POLÍTICA PROTECCIÓN DE DATOS PERSONALES V.3. www.minsal.cl/wp-content/uploads/2020/08/POLÍTICA-PROTECCIÓN-DE-DATOS-PERSONALESV.3.pdf

Ministerio de Salud de Chile. (2023). POLÍTICA DESARROLLO DE SISTEMAS V3.0. Ministerio de Salud. www.minsal.cl/wp-content/uploads/2015/08/POLITICA-DESARROLLO-DE-SISTEMAS-PS-NC-002-V.3.pdf

Ministerio de Salud de Chile. (2023). POLÍTICA GENERAL DE SEGURIDAD DE LA INFORMACIÓN Y CIBERSEGURIDAD V4.0. Ministerio de Salud. www.minsal.cl/wp-content/uploads/2015/08/POLITICA-GENERAL-DE-SEGURIDAD-DE-LA-INFORMACION-Y-CIBERSEGURIDAD-V.4.pdf

Ministerio de Salud de Chile. (2024). Ley 21668 MODIFICA LA LEY Nº20584 CON EL OBJETO DE ESTABLECER LA INTEROPERABILIDAD DE LAS FICHAS CLÍNICAS. Biblioteca del Congreso Nacional. www.bcn.cl/leychile/navegar?idNorma=1203827

Ministerio del Interior y de Seguridad Pública. (2024). Ley 21663 LEY MARCO DE CIBERSEGURIDAD. Biblioteca del Congreso Nacional. www.bcn.cl/leychile/navegar?i=1202434

Ministerio Secretaría General de la Presidencia. (2023). Ley 19628 SOBRE PROTECCIÓN DE LA VIDA PRIVADA. Biblioteca del Congreso Nacional. www.bcn.cl/leychile/navegar?idNorma=141599

Ministerio Secretaría General de la Presidencia “MINSEGPRES”. (2024). Ley 21719 REGULA LA PROTECCIÓN Y EL TRATAMIENTO DE LOS DATOS PERSONALES Y CREA LA AGENCIA DE PROTECCIÓN DE DATOS PERSONALES. Biblioteca del Congreso Nacional. www.bcn.cl/leychile/navegar?idNorma=1209272

Molina, M.; Campos, R.; Carvajal, A.; Salinas, A.; Bravo, F.; Monzones, M.; et al. Prototipos de sistemas de información en salud, Red de Salud Digital de las Universidades del Estado RSDUE – CUECh 21994. Santiago, Chile; 2023.

Nielsen, J. “10 usability heuristics for user interface design”. Nielsen Norman Group; 1994. Disponible en: https://www.nngroup.com/articles/ten-usability-heuristics/

Plan de Fortalecimiento de las Universidades del Estado (PFUE). (2021). Proyecto Red Temática: Plan de Fortalecimiento Universidades Estatales Año 2021.

Ratwani, R.M.; Benda, N.C.; Hettinger, A.Z.; Fairbanks, R.J. “Electronic health record vendor adherence to usability certification requirements and testing standards”. JAMA. 2015;314(10):1070–1071. 10.1001/jama.2015.8372

Red de Salud Digital de las Universidades del Estado. (2021). RSDUE: Salud Digital. www.rsdue.cl/salud-digital/

Red de Salud de las Universidades del Estado: RSDUE (2023). Términos de Referencia (TdR) Contratación para el Desarrollo de un Prototipo del Sistema GCC. (Santiago, Chile)

Red de Salud de las Universidades del Estado: (2023). Términos de Referencia (TdR) Contratación para el Desarrollo de un Prototipo del Sistema RCE. (Santiago, Chile)
Ronchieri, E.; Canaparo, M. “Assessing the impact of software quality models in healthcare software systems”. Health Syst. 2023;12(1):85–97. 10.1080/20476965.2022.2162445
Servicio de Salud Metropolitano Central (SSMC). Proyecto SIDRA. 2018. Disponible en: https://ssmc.redsalud.gob.cl/
Sevilla-González, M.D.R.; Moreno Loaeza, L.; Lázaro-Carrera, L.S.; Bourguet Ramírez, B.; Vázquez Rodríguez, A.; et al. “Spanish version of the System Usability Scale for the assessment of electronic tools: Development and validation”. JMIR Hum Factors. 2020;7(4):e21161. 10.2196/21161
Health and Human Services, United States. (1996) Summary of the HIPAA Privacy Rule www.hhs.gov/hipaa/for-professionals/privacy/laws-regulations/index.html
Wieringa, Roel. (2014). Design Science Methodology for Information Systems and Software Engineering. 10.1007/978-3-662-43839-8.
World Bank. (2024, 5 marzo). Acelerado por la COVID y la inteligencia artificial, el panorama digital en el mundo sigue siendo dispar. Grupo Banco Mundial. www.bancomundial.org/es/news/press-release/2024/03/05/accelerated-by-covid-and-ai-global-digital-landscape-remains-uneven
Zahabi, M., Kaber, D. B., & Swangnetr, M. (2015). Usability and Safety in Electronic Medical Records Interface Design: A Review of Recent Literature and Guideline Formulation. Human Factors, 57(5), 805-834. https://doi.org/10.1177/0018720815576827 (Original work published 2015)
Zhang, J., & Walji, M. F. (2011). TURF: toward a unified framework of EHR usability. Journal of biomedical informatics, 44(6), 1056–1067. https://doi.org/10.1016/j.jbi.2011.08.005

Anexos

Anexo N°1: Checklist de Requerimientos de Seguridad

CHECKLIST DE REQUERIMIENTOS
DIMENSIÓN DE SEGURIDAD - CALIDAD EN SOFTWARE EN SALUD

Este documento es una pauta de evaluación para verificar el cumplimiento de criterios de seguridad referentes a la calidad del software para el ámbito de la salud. Se basa en los requisitos definidos por el Centro Nacional en Sistemas de Salud (CENS), contemplando normativas internacionales como ISO 25010 e ISO 27001, y regulaciones nacionales en Chile.

El objetivo de esta pauta es proporcionar una herramienta que permita evaluar el alineamiento del desarrollo con los estándares de seguridad exigidos. Los desarrolladores deben completar la evaluación marcando el nivel de cumplimiento y proporcionando la evidencia correspondiente, la cual permita a un tercero verificar el cumplimiento. Para facilitar su lectura, los 18 elementos a evaluar han sido agrupados en 5 categorías.

DATOS GENERALES DEL DESARROLLO	
NOMBRE DEL PROYECTO	PROYECTO
EQUIPO DE PROYECTO	**RSDUE**
EQUIPO DE DESARROLLO	EQUIPO DESARROLLO
FECHA DE EVALUACIÓN	DD/MM/AAAA
PUNTAJE OBTENIDO / TOTAL	___/___ **pts**

DESCRIPCIÓN	**PUNTAJE**
Cumple: El requerimiento está implementado correctamente y se posee la evidencia requerida.	1
No cumple: El requerimiento no está implementado o no cumple con los criterios de aceptación establecidos.	0
No aplica: El requerimiento no es relevante para el tipo de software evaluado.	N/A

Tabla de evaluación del nivel de cumplimiento

Nota: Para el puntaje total NO debe considerarse aquellos elementos que sean evaluados “N/A

Control de Acceso y Autenticación				
Elemento	**Cumple (1)**	**No cumple (0)**	**No aplica (N/A)**	**Medios de verificación**
1. Control de Acceso: Restricción de acceso mediante roles y permisos. Contar con un sistema de control de acceso, el cual permita restringir el acceso mediante roles a las funcionalidades y datos almacenados según nivel de autorización. Criterios aceptabilidad: El sistema debe garantizar la autenticación de usuarios y la asignación de roles con privilegios definidos. Todo usuario debe tener un rol asignado o su acceso será desactivado. Si hay datos de pacientes, solo podrán acceder quienes los traten o tengan autorización para auditoría. Debe haber usuarios responsables de gestionar y asignar roles.				Documento técnico con definición de roles y permisos; logs de acceso.
2. Autenticación de Dos Factores (2FA): Implementación de al menos dos métodos de autenticación. Implementar un mecanismo de autenticación de dos factores (2FA) que requiera dos métodos de verificación independientes para confirmar la identidad de un usuario. Criterios aceptabilidad: Debe ofrecer al menos dos métodos de autenticación, como contraseñas, tokens físicos o biometría.				Capturas de pantalla del flujo de autenticación.
3. Protección contra Ataques de Fuerza Bruta: Bloqueo temporal tras intentos fallidos. Implementar un mecanismo de limitación de ataques de fuerza bruta al sistema de autenticación de usuarios. Criterios aceptabilidad: MINSAL requiere implementar un mecanismo de bloqueo temporal tras un número determinado de intentos fallidos, aplicable a la IP de origen o a la cuenta de usuario afectada.				Configuración del sistema de bloqueo; logs de acceso bloqueado.

4. Requisitos de Contraseñas: Políticas de robustez y renovación forzada. Implementar restricciones a las contraseñas de usuario para garantizar contraseñas que cumplan con requerimientos específicos, y forzar su actualización periódica. Criterios aceptabilidad: Las contraseñas deben ser renovadas por el usuario con una frecuencia determinada, lo cual debe ser gatillado por el sistema. Se debe limitar el uso de contraseñas débiles (por ejemplo, solo numéricas). MINSAL mínimo requiere: Al menos 8 caracteres, combinación letras números, símbolos, mayúsculas y minúsculas. No debe contener palabras del diccionario o datos personales del usuario.				Documentación de la gestión de contraseñas, comprobable a través de instancia demo.
Protección de Datos Sensibles				
Elemento	**Cumple (1)**	**No cumple (0)**	**No aplica (N/A)**	**Comentario / Evidencia**
5. Cifrado en Tránsito: Implementación de TLS 1.3 o equivalente. Cifrar toda información confidencial que se envía entre el producto y los servidores. Criterios aceptabilidad: Toda la información confidencial, como contraseñas, datos financieros y datos personales, debe ser cifrada antes de transmitirse por la red. Se debe utilizar un algoritmo de cifrado estándar y probado, como AES o RSA. MINSAL sugiere cifrado de tránsito TLS 1.3. Las claves de cifrado deben almacenarse de forma segura y deben ser solo accesibles para el administrador del sistema.				Documentación del protocolo de cifrado utilizado con captura del navegador donde se muestre la versión en uso.
6. Cifrado en Reposo: Almacenamiento seguro de datos sensibles. Almacenar bajo cifrado toda la información sensible Criterios aceptabilidad: Toda la información sensible (incluyendo contraseñas y datos personales) debe almacenarse cifrada en un entorno seguro. Se debe utilizar un algoritmo de cifrado estándar y probado, como AES o				Especificación de algoritmos utilizados y almacenamiento de claves.

RSA. MINSAL sugiere mínimo aplicar hash como bcrypt, pbkdf2 o Argon2, y evitar usar SHA-1, SHA-2 o MD5. Las claves de cifrado deben almacenarse de forma segura y deben ser sólo accesibles para el administrador del sistema.				
7. Consentimiento Explícito: Registro seguro de la aceptación de términos por parte de los usuarios. Obtener el consentimiento explícito de los usuarios antes de almacenar cualquier información sensible. Criterios aceptabilidad: Los usuarios deben quedar en conocimiento de qué se almacenará y con qué propósito, junto con poder aceptar o rechazar el almacenamiento de su información sensible.				Captura del formulario de consentimiento y registro en base de datos o documentación de flujo de obtención.
8. Eliminación Segura de Datos: Proceso documentado de eliminación irrecuperable. Implementar medidas para eliminar de forma segura los datos sensibles cuando ya no sean necesarios, dejando registro del acto. Criterios aceptabilidad: Los datos sensibles deben poder eliminarse de forma segura una vez que ya no sean necesarios o hayan alcanzado el final de su ciclo de vida útil. Se debe utilizar un método de eliminación de datos seguro que haga que los datos sean irrecuperables. Debe existir un registro de auditoría de los datos eliminados o de las solicitudes de eliminación de datos.				Logs de eliminación de datos y metodología utilizada.
9. Manejo de Datos Sensibles: Políticas de almacenamiento y auditoría. Describir el manejo de datos sensibles en la documentación del producto. Criterios aceptabilidad: Describe cómo se almacenan y transmiten los datos de fichas clínicas, los tiempos de almacenamiento, los procesos de auditoría de control de acceso y modificaciones, responsables de la gestión de la base de datos, todo en cumplimiento con normativa MINSAL				Documento con gestión de datos clínicos.

Monitoreo, Integridad y Auditoría				
Elemento	**Cumple (1)**	**No cumple (0)**	**No aplica (N/A)**	**Comentario / Evidencia**
10. Integridad de Datos: Manejo y corrección de errores. Existencia de gestión de integridad de datos Criterios aceptabilidad: Se debe automatizar el manejo de errores de datos, incluyendo su corrección y monitoreo de actividad inusual. En caso de existir datos rechazados se debe permitir su ingreso/corrección manual. En caso de datos huérfanos, estos deben poder identificarse y gestionar su eliminación adecuada para evitar datos innecesarios u obsoletos en el sistema.				Documentación sobre validaciones y registros de datos rechazados.
11. Auditoría de Actividades: Registro de acciones realizadas en el sistema. Sistema de auditoría de actividades dentro del software Criterios aceptabilidad: Se debe registrar fecha/hora, acción y quien realiza (con usuario identificado o identificador en caso de acceso externo) que permita trazabilidad de acceso y modificaciones de información dentro del sistema.				Logs de auditoría con identificación de usuarios y eventos.
12. Gestión y Recuperación ante Fallos Implementar un proceso integral de gestión de fallos que incluya su detección, registro, mitigación y recuperación, asegurando la continuidad operativa del sistema. Criterios aceptabilidad: El sistema debe contar con un mecanismo de monitoreo y registro de errores técnicos, que incluya fallos de software, problemas de rendimiento y caídas inesperadas. Se deben implementar medidas preventivas para reducir la probabilidad de				Documentación del funcionamiento del sistema de monitoreo y registro de fallos. Registro de eventos. Ejemplo de reporte de fallos en entorno de prueba, incluyendo medidas de contingencia activadas, tiempo requerido para su recuperación e impacto en la conti-

fallas críticas, como redundancia de sistemas, balanceo de carga y detección temprana de errores. Cada incidente debe ser documentado con información clave: tipo de error, fecha/hora, severidad, componente afectado y acciones correctivas tomadas. Se debe definir un tiempo máximo esperado de recuperación ante distintos tipos de fallos, asegurando que el sistema pueda restaurarse en un porcentaje aceptable de escenarios. Se debe establecer un protocolo para analizar las causas de los fallos y aplicar mejoras continuas para fortalecer la resiliencia del sistema.				nuidad operativa. Análisis del porcentaje de incidentes en los que el sistema logró recuperarse dentro del tiempo esperado.
13. Gestión de Incidentes de Seguridad: Procedimiento documentado para respuesta ante incidentes. Describir en la documentación del producto un procedimiento de gestión de incidentes de seguridad, estableciendo cómo se detectan, notifican, analizan y responden los incidentes dentro del sistema. Criterios aceptabilidad: El sistema debe contar con un procedimiento documentado que detalle la detección, notificación, análisis y respuesta ante incidentes de seguridad, especificando los roles y responsabilidades del equipo encargado. Los incidentes deben ser registrados y documentados, incluyendo su resolución y medidas de mitigación aplicadas.				Documento de gestión de incidentes; formato de registros de incidentes pasados.
Resistencia y Recuperación				
Elemento	**Cumple (1)**	**No cumple (0)**	**No aplica (N/A)**	**Comentario / Evidencia**
14. Pruebas de Seguridad: Ejecución de pruebas de penetración. Incluir pruebas de verificación de seguridad contra ataques informáticos realizadas al software.				Reportes de pruebas de seguridad. Los resultados de las pruebas deben incluir las vulnerabilidades encontradas, su severidad y las acciones correctivas implementadas. Si se realizaron correcciones, deben incluirse

Criterios aceptabilidad: Pruebas de penetración contra diferentes componentes del software (APIs, BBDD, Frontend, etc.), incluyendo escenarios de ataques comunes como inyección SQL, cross-site scripting (XSS), elevación de privilegios y fuerza bruta, y para los usuarios (por ejemplo, de phishing) Los informes deben documentar las vulnerabilidades encontradas y acciones pertinentes.				pruebas de seguimiento
15. Plan de Respaldos: Estrategia documentada de backup y restauración. Describir el plan de respaldo sugerido en la documentación del producto. Criterios aceptabilidad: El plan de respaldo debe documentar el proceso de creación, restauración y prueba de las copias de seguridad. Las copias de seguridad deben realizarse con regularidad y almacenarse en un lugar seguro fuera del sitio. El plan de pruebas diario debe verificar que las copias de seguridad se puedan restaurar correctamente.				Registro de pruebas de recuperación de respaldos.
16. Modo Offline y Continuidad Operativa: Protocolo para registrar datos sin conexión. Cuando el sistema forma parte de procesos críticos de una institución, este debe considerar un modo de funcionamiento fuera de línea o contar con un protocolo que permita la continuidad operativa en caso de caída del sistema, junto a su posterior sincronización/carga al sistema. Criterios aceptabilidad: Se debe establecer un protocolo de registro de datos fuera de línea, ya sea a través de una versión reducida del aplicativo de uso local, una plantilla a papel, Word u otro instrumento, que permita la continuidad operativa e identificación de los registros, y el proceso de carga de dichos datos, ante potenciales caídas por pérdida de conexión o suministro eléctrico u otro evento crítico que pueda perjudicar la continuidad operativa de la institución.				Documento con plan de operación offline.

Normativas y Buenas Prácticas				
Elemento	**Cumple (1)**	**No cumple (0)**	**No aplica (N/A)**	**Comentario / Evidencia**
17. Políticas de Seguridad en Redes: Recomendaciones de seguridad para la institución usuaria. Describir sugerencias de políticas y prácticas de seguridad en redes potencialmente implementables para prevenir, detectar y supervisar el acceso no autorizado, la modificación, el uso incorrecto o la denegación de una red y sus correspondientes recursos tanto en redes internas como externas. Criterios aceptabilidad: La institución tiene que implementar políticas de seguridad en redes para el uso del producto, las que deben estar sugeridas definiendo los permisos de acceso a la red, el uso de los recursos de red y las medidas de seguridad a tomar.				Documento con directrices de seguridad en redes.
18. Recomendaciones de Seguridad del Usuario y del Entorno Físico Incluir en la documentación del producto directrices para la seguridad del usuario y del entorno de trabajo, minimizando riesgos de acceso no autorizado y ataques dirigidos. Criterios aceptabilidad: Se deben incluir recomendaciones sobre almacenamiento seguro de credenciales (evitar texto plano, uso de gestores), buenas prácticas ante phishing e ingeniería social, y políticas para el uso de dispositivos (como restringir USB no verificados o bloquear sesiones por inactividad). También deben establecerse medidas de control físico de acceso a servidores y estaciones críticas, y protocolos internos para minimizar riesgos (por ejemplo, control de acceso a zonas sensibles y verificación de personal externo)				Documentación en el manual del usuario con directrices de seguridad y políticas de acceso a hardware.

Una vez completado el *checklist*, se debe calcular el puntaje obtenido y compararlo con el puntaje máximo posible, excluyendo los requerimientos marcados como "No Aplica". Esto permitirá una evaluación cuantitativa del nivel de seguridad del software en evaluación.

Anexo N°2: Guía de buenas prácticas de Usabilidad

RECOMENDACIONES PARA LA USABILIDAD EN SOFTWARE DE SALUD

A continuación, se presentan recomendaciones para guiar el diseño de software en salud en relación a la dimensión usabilidad de la calidad de software. Las recomendaciones están divididas en dos categorías: Experiencia usuaria y funcionalidades, y Elementos de la interfaz usuaria. En la parte final se presentan sugerencias de instrumentos para la evaluación de usabilidad, de los cuales se sugiere seleccionar el más adecuado al contexto y objetivo de la evaluación.

EXPERIENCIA USUARIA Y FUNCIONALIDADES

HAZ ✅

1. Involucra a usuarios clínicos en todas las etapas del diseño.
2. Realiza pruebas de usabilidad iterativas con usuarios reales.
3. Alinea los flujos digitales con los flujos clínicos reales.
4. Diseña funcionalidades alineadas a objetivos clínicos.
5. Optimiza tareas frecuentes para requerir mínimos pasos.
6. Personaliza según rol y especialidad clínica.
7. Muestra confirmaciones en acciones irreversibles.
8. Habilita registros rápidos en urgencias.
9. Ofrece soporte y ayudas integradas accesibles.
10. Asegura interoperabilidad con otras plataformas clínicas.
11. Implementa capacitaciones breves y prácticas.
12. Define métricas de uso, satisfacción y UX.
13. Facilita la colaboración entre profesionales.
14. Guía al usuario frente a errores.
15. Realiza pilotos antes del despliegue completo.
16. Involucra pacientes o cuidadores cuando corresponda.
17. Integra alertas proactivas de seguridad del paciente.
18. Prioriza estabilidad sobre novedades.
19. Genera resúmenes clínicos claros y sintéticos.
20. Apoya la toma de decisiones sin reemplazar el juicio clínico.
21. Considera diferencias generacionales.
22. Asegura disponibilidad y respuesta del sistema.
23. Diseña procesos de registro eficientes y seguros.
24. Valida datos para evitar errores de registro.
25. Usa plantillas para entradas repetitivas.

EVITA 🚫

1. Multiplicar pasos para tareas simples.
2. Asumir experiencia técnica avanzada de los usuarios.
3. Introducir cambios sin evaluar impacto ni dar aviso.
4. Mostrar información irrelevante para la actividad en curso.
5. Permitir que fallas interrumpan el flujo clínico.
6. Sobrecargar con validaciones innecesarias.
7. Exigir permisos administrativos para funciones críticas.
8. Liberar funciones experimentales en producción.
9. Imponer flujos rígidos y universales.
10. Ignorar limitaciones de hardware.
11. Depender solo de reuniones para recolectar feedback.
12. Confiar exclusivamente en métricas técnicas para evaluar UX.
13. Excluir a los usuarios del diseño y prueba.
14. Asumir que lo que funciona en un centro sirve para todos.
15. Alterar procesos clínicos solo para adaptarse al software.

ELEMENTOS DE LA INTERFAZ USUARIA

HAZ ✅

1. Usa colores con significado clínico consistente.
2. Diseña botones grandes, claros y accesibles.
3. Mantén alto contraste entre texto, botones y fondo.
4. Ubica elementos comunes de forma consistente.
5. Usa íconos intuitivos, estandarizados y sencillos con etiquetas.
6. Adapta el diseño a distintos dispositivos.
7. Aplica jerarquía visual clara en alertas.
8. Distingue acciones críticas de las secundarias.
9. Haz formularios breves y claros, marcando campos obligatorios.
10. Cumple estándares de accesibilidad (WCAG, ADA).
11. Provee feedback visual inmediato.
12. Muestra progreso en tareas largas.
13. Organiza menús de forma lógica.
14. Usa mensajes de confirmación claros.
15. Garantiza navegación intuitiva.
16. Diferencia claramente elementos interactivos.
17. Permite deshacer acciones.
18. Prueba en pantallas de baja resolución.
19. Prioriza visualmente la información clínica.
20. Usa textos claros y concisos.
21. Señala claramente estados del sistema.
22. Aplica filtros visuales útiles.
23. Revisa diseños con expertos y usuarios.
24. Separa visualmente las secciones.
25. Alinea el diseño a la marca institucional según corresponda.

EVITA 🚫

1. Usar paletas de colores difíciles de distinguir.
2. Botones demasiado pequeños y juntos.
3. Ubicación inconsistente de elementos.
4. Jerarquías visuales confusas.
5. Exceso de colores brillantes en alertas.
6. Formularios extensos y desordenados.
7. Ocultar errores sin señalización.
8. Menús sobrecargados.
9. Exceso de animaciones.
10. Imágenes de baja calidad.
11. No permitir ajuste del tamaño de la fuente.
12. Mensajes ambiguos o confusos.
13. Sobrecargar interfaces con elementos innecesarios.
14. No considerar errores humanos en el diseño.
15. Usar jerga técnica incomprensible.

EVALUACIÓN DEL RESULTADO

Comparación entre instrumentos de usabilidad, denotando fortalezas y debilidades y sugiriendo contexto de uso.

Modelo	Cobertura	Facilidad de uso	Contexto de uso sugerido	Fortalezas principales	Debilidades
SUS	Usabilidad general	Muy alta	Evaluaciones rápidas desde productos mínimos	Fácil de aplicar, resultados cuantificables	Superficial, no aborda dimensiones complejas
PSSUQ	Satisfacción post-uso	Alta	Evaluación tras prueba funcionales exitosas	Profundiza en utilidad percibida	Requiere explicación al usuario y prueba del sistema
TAM	Aceptación tecnológica	Alta	Evaluaciones de disposición de usuarios finales	Predice intención de uso a través de la aceptación	No mide directamente experiencia real
HOT-Fit	Factores humanos y organizacionales	Media	Evaluaciones holísticas en entornos de salud considerando características de la institución	Integra tecnología, organización y usuario	Complejo, difícil de aplicar sin expertos
ISO 25010	Calidad formal del software	Baja	Certificaciones formales, auditorías	Normativo, amplio y estandarizado	No tiene cuestionarios predefinidos
AttrakDiff	Calidad emocional y estética	Media	Evaluación de percepción subjetiva	Diferencia entre lo hedónico y lo funcional	Menor aplicabilidad en salud

Anexo N°3: Matriz de mapeo contra modelo de calidad del CENS

Elemento del *Checklist* de Seguridad	Elemento del Sello Calidad relacionado
1. Control de Acceso: Restricción de acceso mediante roles y permisos.	7.2.v
2. Autenticación de Dos Factores (2FA): Implementación de al menos dos métodos de autenticación.	7.2.v
3. Protección contra Ataques de Fuerza Bruta: Bloqueo temporal tras intentos fallidos.	7.2.v
4. Requisitos de Contraseñas: Políticas de robustez y renovación forzada.	Sin contraparte directa
5. Cifrado en Tránsito: Implementación de TLS 1.3 o equivalente.	7.1.iii 7.1.iv
6. Cifrado en Reposo: Almacenamiento seguro de datos sensibles.	7.1.vi
7. Consentimiento Explícito: Registro seguro de la aceptación de términos por parte de los usuarios.	7.1.viii 7.1.ix
8. Eliminación Segura de Datos: Proceso documentado de eliminación irrecuperable.	7.1.xii
9. Manejo de Datos Sensibles: Políticas de almacenamiento y auditoría.	7.1.v 7.1.x 7.1.xi
10. Integridad de Datos: Manejo y corrección de errores.	Sin contraparte directa
11. Auditoría de Actividades: Registro de acciones realizadas en el sistema.	Sin contraparte directa
12. Gestión y Recuperación ante Fallos	6.1.x
13. Gestión de Incidentes de Seguridad: Procedimiento documentado para respuesta ante incidentes.	Sin contraparte directa
14. Pruebas de Seguridad: Ejecución de pruebas de penetración.	7.1.i 7.1.iii 7.2.i 7.2.ii
15. Plan de Respaldos: Estrategia documentada de backup y restauración.	6.2.x
16. Modo Offline y Continuidad Operativa: Protocolo para registrar datos sin conexión.	Sin contraparte directa
17. Políticas de Seguridad en Redes: Recomendaciones de seguridad para la institución usuaria.	Sin contraparte directa
18. Recomendaciones de Seguridad del Usuario y del Entorno Físico	7.2.iii

Nota: Relación realizada en base a estructura del instrumento de evaluación del sello de Calidad de Software en Salud realizado por CENS.

Anexo N°4: Formato encuesta a desarrolladores

ENCUESTA A DESARROLLADORES
EVALUACIÓN DE LINEAMIENTOS DE CALIDAD

El objetivo de esta encuesta es evaluar la utilidad, aplicabilidad y comprensión de los lineamientos de seguridad y usabilidad planteados por el estudiante de magíster durante el desarrollo de los sistemas de salud propuestos por la Red de Salud de las Universidades del Estado (RSDUE).

SECCIÓN A. Perfil del participante

1. Rol principal en el equipo de desarrollo (selecciona uno):
☐ Backend (Desarrolla la lógica del servidor, integración de base de datos, autenticación, API, etc.)
☐ Frontend (Desarrolla la interfaz con la que interactúan los usuarios, usando tecnologías como HTML/CSS/JS)
☐ Fullstack (Realiza tanto desarrollo frontend como backend)
☐ Gestión de proyecto (Coordina, organiza y supervisa actividades del equipo técnico)
☐ DevOps/Sistemas (Administra la infraestructura tecnológica, despliegue, servidores)
☐ Otro (especificar): _____

2. Título profesional/técnico del participante:
_____

3. Años de experiencia en desarrollo de software:
☐ 0–2 años ☐ 3–5 años ☐ 6–10 años ☐ Más de 10 años

4. ¿Habías trabajado anteriormente en desarrollo de software para salud?
☐ Sí ☐ No

SECCIÓN B. Formación en seguridad y usabilidad

5. ¿Recibiste formación sobre seguridad informática durante tu formación profesional?
☐ Sí, de manera formal (asignatura o módulo)
☐ Solo nociones generales
☐ No

6. ¿Has realizado cursos posteriores sobre seguridad en desarrollo de software?
☐ Sí (especificar): _____
☐ No

7. ¿Recibiste formación sobre usabilidad en tu formación profesional?
☐ Sí, de manera formal (asignatura o módulo)
☐ Solo nociones generales
☐ No

8. ¿Has realizado cursos posteriores sobre usabilidad?
☐ Sí (especificar): _____
☐ No

SECCIÓN C. Lineamientos de seguridad

9. ¿Qué tan claros te resultaron los elementos del checklist de seguridad? (1 Nada claro – 5 Muy claro)

☐ 1 - Nada claro ☐ 2 - Poco claro ☐ 3 - Neutral ☐ 4 - Claro ☐ 5 - Muy claro

10. ¿El formato tipo checklist te facilitó su aplicación? (1 Nada útil – 5 Muy útil)

☐ 1 - Nada útil ☐ 2 - Poco útil ☐ 3 - Neutral ☐ 4 - Útil ☐ 5 - Muy útil

11. ¿El checklist ayudó a detectar aspectos de seguridad que no habías considerado previamente? (1 Para nada – 5 Muchísimo)

☐ 1 - Para nada ☐ 2 - Poco ☐ 3 - Neutral ☐ 4 - Bastante ☐ 5 – Muchísimo

12. ¿El *checklist* cuenta con elementos difíciles de interpretar?

☐ Sí (especificar cuáles y por qué): _____

☐ No

13. ¿El *checklist* cuenta con elementos difíciles de aplicar?

☐ Sí (especificar cuáles y por qué): _____

☐ No

SECCIÓN D. Lineamientos de usabilidad

13. ¿La guía “Haz y Evita” fue clara y comprensible? (1 Nada clara – 5 Muy clara)

☐ 1 – Nada clara ☐ 2 - Poco clara ☐ 3 - Neutral ☐ 4 - Clara ☐ 5 - Muy clara

14. ¿Consideras que las recomendaciones de la guía podrían ser útiles para mejorar la usabilidad en desarrollos actuales o futuros?

☐ Sí, serían muy útiles

☐ Sí, en parte

☐ No serían útiles

15. ¿Qué aspecto te pareció más valioso o práctico dentro de la guía?

_____

SECCIÓN E. Valoración general y sugerencias

16. ¿Recomendarías usar estos instrumentos en futuros proyectos?

☐ Sí

☐ No

17. ¿Qué sugerencias harías para mejorar los instrumentos entregados?

_____

Anexo N°5: Tabla de resultados de encuesta a desarrolladores

Fila	Rol	Título	Experiencia	Experiencia en salud	Formación en seguridad	Cursos seguridad
1	Fullstack	Ingeniero en administración de redes	3-5 años	No	Si, de manera formal	No
2	Backend	Analista Programador	Más de 10 años	Sí	Sí, de manera formal	No
3	Frontend	Ingeniero en Informática	Más de 10 años	No	Solo nociones generales	No
4	DevOps/Sistemas	Ingeniero en Informática	3–5 años	Sí	No	Sí
5	DevOps/Sistemas	Ingeniero en Informática	6–10 años	No	No	No
6	Backend	Ingeniero en Informática	3–5 años	Sí	Sí, de manera formal	No
7	Gestión de proyecto	Ingeniero en Informática	0–2 años	No	Solo nociones generales	No
8	Fullstack	Ingeniero en Informática	6–10 años	No	Solo nociones generales	No
9	Fullstack	Ingeniero en Informática	3–5 años	Sí	Sí, de manera formal	Sí
10	Fullstack	Analista Programador	3–5 años	Sí	No	No
11	Frontend	Ingeniero en Informática	0–2 años	No	Sí, de manera formal	Sí
12	Backend	Analista Programador	3–5 años	No	Sí, de manera formal	Sí
13	Gestión de proyecto	Ingeniero en Informática	3–5 años	Sí	Solo nociones generales	Sí
14	Gestión de proyecto	Ingeniero en Informática	Más de 10 años	No	Solo nociones generales	Sí
15	Frontend	Ingeniero en Redes	6–10 años	No	No	No
16	Frontend	Técnico en Informática	Más de 10 años	No	Solo nociones generales	No
17	Fullstack	Ingeniero en Redes	0–2 años	No	Sí, de manera formal	Sí
18	DevOps/Sistemas	Ingeniero en Redes	3–5 años	Sí	Solo nociones generales	Sí
19	Backend	Analista Programador	3–5 años	No	No	No
20	Fullstack	Ingeniero en Informática	Más de 10 años	Sí	Si, de manera formal	No
21	Gestión de proyecto	Ingeniero Civil en Computación	0–2 años	No	Sí, de manera formal	Sí
22	Frontend	Analista Programador	6–10 años	No	Sí, de manera formal	Sí
23	Fullstack	Ingeniero Civil en Computación	0–2 años	No	Si, de manera formal	No
24	Gestión de proyecto	Ingeniero en Informática	0–2 años	No	Sí, de manera formal	No
25	Gestión de proyecto	Ingeniero en Informática	3–5 años	Sí	Sí, de manera formal	No
26	DevOps/Sistemas	Ingeniero Civil en Computación	0–2 años	Sí	Si, de manera formal	Sí

Fila	Formación en usabilidad	Cursos usabilidad	Claridad checklist	Utilidad checklist	Detección checklist	Elementos difíciles de interpretar	Si es que sí, cuáles y por qué (interpretar)
1	No	No	5	5	5	No	
2	No	No	5	3	5	No	
3	No	No	4	4	4	No	
4	No	No	3	4	5	No	
5	Sí, de manera formal	No	4	5	5	Sí	Algunos términos técnicos pueden confundir a nuevos desarrolladores.
6	Sí, de manera formal	No	3	4	3	No	
7	Sí, de manera formal	No	5	3	5	No	
8	Sí, de manera formal	No	4	5	4	No	
9	Sí, de manera formal	No	3	4	5	No	
10	Solo nociones generales	No	3	2	2	Sí	Elementos como 'modo offline' no siempre están en nuestros flujos.
11	Solo nociones generales	No	4	4	5	No	
12	Solo nociones generales	No	5	5	4	No	
13	Solo nociones generales	No	4	3	1	No	
14	Solo nociones generales	Sí (diplomado)	5	5	4	No	
15	No	Sí (por internet)	3	5	5	No	
16	No	Sí (por internet)	5	4	4	No	
17	No	Sí (por internet)	3	4	5	No	
18	No	Sí (por internet)	5	4	3	No	
19	Sí, de manera formal	Sí (por internet)	3	5	5	No	
20	Sí, de manera formal	Sí (por internet)	5	4	3	No	
21	Sí, de manera formal	Sí (por internet)	4	5	4	No	
22	Sí, de manera formal	Sí (por internet)	4	4	5	No	
23	Solo nociones generales	Sí (por internet)	4	3	4	Sí	Hay elementos que requieren conocimientos previos para un correcto abordaje.
24	Solo nociones generales	Sí (por internet)	4	3	4	No	
25	Solo nociones generales	Sí (por internet)	3	5	3	No	
26	Sí, de manera formal	Sí (talleres de UX)	3	5	5	No	

Fila	Elementos difí-ciles de aplicar	Si es que sí, cuáles y por qué (aplicar)	Claridad guía	Utilidad guía
1	No		5	Sí, serían muy útiles
2	No		4	Sí, serían muy útiles
3	Sí	Los requisitos de eliminación segura son difíciles en sistemas heredados.	5	Sí, serían muy útiles
4	No		3	Sí, serían muy útiles
5	No		3	No serían útiles
6	No		5	Sí, serían muy útiles
7	No		4	No serían útiles
8	No		3	Sí, en parte
9	No		5	Sí, en parte
10	No		5	Sí, en parte
11	No		5	Sí, en parte
12	No		4	Sí, en parte
13	No		3	Sí, serían muy útiles
14	Sí	No tenemos suficientes personas en el equipo.	4	Sí, en parte
15	No		3	Sí, en parte
16	Sí	Extiende el tiempo requerido para los proyectos, por lo que aumenta sus cos-tos.	4	Sí, en parte
17	No		3	Sí, serían muy útiles
18	No		3	Sí, en parte
19	No		5	Sí, en parte
20	No		4	Sí, en parte
21	No		4	Sí, serían muy útiles
22	No		3	No serían útiles
23	No		3	Sí, serían muy útiles
24	No		4	Sí, serían muy útiles
25	No		5	Sí, en parte
26	No		5	Sí, en parte

Fila	Aspecto más valioso	Recomendaría instrumentos	Sugerencias de mejora
1	La pauta de evaluación.	Sí	En blanco, no entrega.
2	Trabajar con los usuarios	Sí	En blanco, no entrega.
3	La claridad sobre uso de colores con significado clínico estandarizado.	Sí	En blanco, no entrega.
4	La inclusión de funcionalidades offline para continuidad operativa.	No	En blanco, no entrega.
5	Mostrar la relevancia de un desarrollo metódico.	No	En blanco, no entrega.
6	El hacer pilotos de pruebas.	Sí	En blanco, no entrega.
7	Mantener la concordancia en los sistemas	Sí	Agregar ejemplos visuales podría facilitar la comprensión.
8	Recomendación de instrumentos de evaluación.	Sí	En blanco, no entrega.
9	Probar funciones en ambientes cerrados para ajustarlas.	Sí	En blanco, no entrega.
10	La guía de diseño de formularios breves y claros.	Sí	En blanco, no entrega.
11	Recoger feedback de los usuarios.	Sí	En blanco, no entrega.
12	Considerar la experiencia del paciente	Sí	En blanco, no entrega.
13	Sugerencias sobre accesibilidad y contraste de colores.	No	En blanco, no entrega.
14	El tener que incluir métricas de usabilidad.	No	Incluir glosario para nuevos desarrolladores.
15	Que existen muchas herramientas para evaluar los sistemas.	No	Separar mejor los criterios técnicos y funcionales.
16	Añadir plantillas para registros repetidos.	Sí	En blanco, no entrega.
17	Existencia de metodologías para evaluar la usabilidad.	Sí	Describir ejemplos.
18	El enfoque en guiar al usuario ante errores sin interrumpir el flujo clínico.	Sí	En blanco, no entrega.
19	La guía sobre confirmaciones en acciones irreversibles.	Sí	En blanco, no entrega.
20	Que permite entregar un sistema robusto.	Sí	Limitando lo lograble en corto plazo v/s mediano y largo plazo.
21	La recomendación de alinear flujos clínicos con digitales.	Sí	En blanco, no entrega.
22	El enfoque de interoperabilidad, ahora estamos implementando FHIR.	No	En blanco, no entrega.
23	Sugerencia de probar en pantallas de baja resolución, estamos llenos de equipos antiguos.	Sí	En blanco, no entrega.
24	Sugerencias sobre personalización según rol clínico, muy útil para sistemas complejos.	Sí	En blanco, no entrega.
25	El sistema tiene que ser útil para el usuario.	Sí	En blanco, no entrega.
26	Diseñar el sistema en conjunto y no solos dentro de la oficina.	No	En blanco, no entrega.

Anexo N°6: Evaluación cruzada de lineamientos de seguridad

Elemento del Checklist a interpretar	Interpretación Desarrollador A (Fullstack)	Interpretación Desarrollador B (Backend)	Apreciación de B respecto de interpretación de A	Apreciación de A respecto de la interpretación de B	Precisión de la discrepancia de Interpretación
1. Control de Acceso: Restricción de acceso mediante roles y permisos.	El sistema debe tener un mecanismo claro y bien documentado que permita asignar permisos de acceso a distintas funcionalidades, según el rol que cumpla cada usuario dentro de la aplicación. Estos permisos deben ser estrictos, especialmente para datos sensibles.	Cada usuario del sistema debería tener un perfil que determine sus privilegios, impidiendo que accedan a funciones que no les correspondan. Esto es clave para mantener la privacidad de los datos.	Sí	Sí	
2. Autenticación de Dos Factores (2FA): Implementación de al menos dos métodos de autenticación.	Debe implementarse una doble capa de autenticación, en donde el usuario primero se identifique con una credencial y luego valide con un método adicional, como una clave temporal enviada por SMS o una verificación biométrica.	Es necesario agregar una autenticación secundaria que confirme la identidad del usuario, usando un método distinto a la clave como puede ser un token o huella digital.	Sí	Sí	
3. Protección contra Ataques de Fuerza Bruta: Bloqueo temporal tras intentos fallidos.	El sistema tiene que protegerse contra intentos repetidos de acceso no autorizado, bloqueando temporalmente el acceso después de varios intentos fallidos, ya sea por IP o por cuenta afectada.	El sistema tiene que bloquear el acceso de manera automática cuando se detecten múltiples intentos fallidos de ingreso, como defensa ante ataques automatizados.	Sí	Sí	
4. Requisitos de Contraseñas: Polí-	Es fundamental que el sistema exija claves fuertes que com-	El sistema debería impedir el uso de contraseñas débiles y	Sí	Sí	

ticas de robustez y renovación forzada.	binen distintos tipos de caracteres, y además obligue a los usuarios a cambiarlas cada cierto tiempo, como parte de una política de seguridad.	asegurarse de que se actualicen cada cierto periodo, aumentando así la seguridad general.			
5. Cifrado en Tránsito: Implementación de TLS 1.3 o equivalente.	Todo intercambio de información sensible entre el cliente y el servidor debe ir encriptado con un protocolo como TLS 1.3 para evitar filtraciones de datos durante su transmisión.	Toda la información personal y sensible que viaja por internet debe ir cifrada con estándares actualizados como TLS 1.3, garantizando su protección en tránsito.	Sí	Sí	
6. Cifrado en Reposo: Almacenamiento seguro de datos sensibles.	La información privada de los usuarios no debe guardarse en texto plano, sino que debe estar cifrada con algoritmos robustos como AES o bcrypt, para evitar que sea legible en caso de acceso no autorizado.	Los datos críticos deben almacenarse en la base de datos usando algoritmos de cifrado o hash fuertes, asegurando que nadie pueda interpretarlos si accede sin permiso.	Sí	No	B no menciona explícitamente la necesidad de evitar algoritmos hash obsoletos como SHA-1 o MD5, lo cual puede interpretarse como una omisión relevante.
7. Consentimiento Explícito: Registro seguro de la aceptación de términos por parte de los usuarios.	Antes de almacenar datos personales, se debe obtener el consentimiento informado del usuario, quien debe aceptar explícitamente qué se hará con su información.	Debe quedar registrado que el usuario fue informado y consintió el uso de su información personal antes de que esta sea almacenada en el sistema.	Sí	Sí	
8. Eliminación Segura de Datos: Proceso documentado de eliminación irrecuperable.	Cuando los datos sensibles ya no sean necesarios, el sistema debe asegurarse de eliminarlos de forma que no puedan ser recuperados. Esto incluye técnicas como sobreescritura segura o destrucción criptográfica.	El sistema tiene que aplicar un proceso robusto para borrar datos personales de manera definitiva, sin dejar posibilidad de recuperación futura. Este proceso debería documentarse y generar un log o auditoría de cada eliminación.	Sí	Sí	
9. Manejo de	Debe quedar claramente es-	La política de datos sensibles	Sí	No	A hace énfasis en los

Datos Sensibles: Políticas de almacenamiento y auditoría.	pecificado en la documentación cómo se protegen los datos clínicos: cómo se almacenan, quién puede acceder, y cómo se audita cada cambio.	debe dejar claro dónde se guardan los datos clínicos, cuánto tiempo se conservan y cómo se controlan los accesos y modificaciones.			responsables de acceso y auditoría, mientras que B se limita a aspectos técnicos de almacenamiento.
10. Integridad de Datos: Manejo y corrección de errores.	El software debe tener procesos que aseguren que los errores en los datos se identifiquen automáticamente, permitiendo su corrección o eliminación sin dejar basura.	Deben existir validaciones automáticas para evitar errores en la base de datos, permitiendo corregir o eliminar registros inválidos sin intervención manual constante.	Sí	Sí	
11. Auditoría de Actividades: Registro de acciones realizadas en el sistema.	Cada acción importante realizada en el sistema debe quedar registrada con detalles como la hora, el usuario y el tipo de operación, para garantizar trazabilidad.	Debe haber un sistema de log que capture cada modificación o consulta hecha en el sistema, con identificación del usuario y fecha, para auditorías futuras.	Sí	Sí	
12. Gestión y Recuperación ante Fallos	El sistema debe contar con una estrategia completa para detectar errores, registrar fallas, activar respuestas automáticas y recuperarse en el menor tiempo posible.	Se deben registrar todos los fallos técnicos del sistema, activar protocolos de contingencia y asegurar que se pueda restaurar el servicio en plazos aceptables.	Sí	Sí	
13. Gestión de Incidentes de Seguridad: Procedimiento documentado para respuesta ante incidentes.	Se necesita un protocolo claro para manejar incidentes de seguridad, desde su detección hasta su análisis posterior, incluyendo acciones correctivas y responsables.	Frente a un evento de seguridad, el sistema tiene que contar con instrucciones precisas para manejarlo, desde el reporte inicial hasta el análisis post-incidente.	Sí	Sí	
14. Pruebas de Seguridad: Ejecución de pruebas	Es obligatorio someter el software a pruebas técnicas que simulen ataques comunes	Hay que hacer pruebas controladas que imiten ataques reales para evaluar qué tan vulnera-	Sí	Sí	

de penetración.	para detectar vulnerabilidades y aplicar mejoras antes de su despliegue.	ble es el sistema y cómo se comporta ante amenazas conocidas.			
15. Plan de Respaldos: Estrategia documentada de backup y restauración.	Debe existir un mecanismo periódico de respaldo de la información, con pruebas que demuestren que estos respaldos pueden recuperarse cuando sea necesario.	Debe existir una rutina de respaldo automático y un plan que asegure que los archivos respaldados pueden recuperarse de forma efectiva ante un incidente.	Sí	Sí	
16. Modo Offline y Continuidad Operativa: Protocolo para registrar datos sin conexión.	En caso de fallas o caídas, el sistema debe poder seguir funcionando de forma limitada y permitir que los datos se guarden localmente para sincronizar luego.	El sistema debe permitir trabajar sin conexión en casos de emergencia, y luego cargar la información acumulada en cuanto se recupere la conectividad.	Sí	Sí	
17. Políticas de Seguridad en Redes: Recomendaciones de seguridad para la institución usuaria.	La organización que utilice el software debe recibir recomendaciones sobre cómo proteger sus redes, incluyendo reglas de acceso, firewalls y monitoreo continuo.	Se deben incluir pautas de ciberseguridad para la red institucional, como reglas de acceso, segmentación y control de tráfico, que protejan al sistema.	Sí	Sí	
18. Recomendaciones de Seguridad del Usuario y del Entorno Físico	El manual debe enseñar al usuario cómo proteger su equipo y sus credenciales, prevenir ataques como el phishing y limitar accesos físicos no autorizados.	Deben entregarse consejos sobre cómo mantener el entorno físico seguro, restringir accesos no autorizados y evitar que los usuarios sean víctimas de ingeniería social.	Sí	Sí	

Anexo N°7: Evaluación cruzada de lineamientos de usabilidad

N°	Interpretación del enunciado por Desarrollador A (Fullstack)	Interpretación del enunciado por Desarrollador B (Backend)	Apreciación del Desarrollador B respecto de interpretación de A	Apreciación del Desarrollador A respecto de interpretación de B
Sección "Haz" de la lista de Experiencia Usuaria y Funcionalidades				
1	Es clave incluir a usuarios clínicos en todas las etapas de diseño para garantizar que el sistema responda a sus necesidades reales.	Debe garantizarse que el sistema sea capaz de involucrar a usuarios clínicos en todas las etapas del diseño.	Sí	Sí
2	Las pruebas iterativas con usuarios reales permiten mejorar la usabilidad de forma continua durante el desarrollo.	El sistema debe estar diseñado para permitir realizar pruebas de usabilidad frecuentes con usuarios reales.	Sí	Sí
3	Los flujos digitales deben reflejar los procesos clínicos reales para evitar interferencias en la práctica médica.	Debe garantizarse que el sistema sea capaz de alinear los flujos digitales con los flujos clínicos reales.	Sí	Sí
4	Las funcionalidades deben diseñarse con foco en los objetivos clínicos que apoyan la toma de decisiones.	La lógica del sistema debe contemplar el diseño de funcionalidades alineadas a objetivos clínicos.	No	No
5	Las tareas más frecuentes deben poder completarse con el mínimo de pasos posible para ahorrar tiempo.	El desarrollo debe considerar mecanismos que permitan optimizar tareas frecuentes para requerir mínimos pasos.	Sí	No
6	Es recomendable adaptar la interfaz según el rol o especialidad del usuario para facilitar su trabajo.	La lógica del sistema debe contemplar personalizaciones según rol y especialidad clínica.	Sí	Sí
7	Ante acciones irreversibles, el sistema debe mostrar mensajes de confirmación claros para evitar errores.	El sistema debe estar diseñado para mostrar confirmaciones en acciones irreversibles.	Sí	Sí
8	En contextos de urgencia, se debe permitir un registro rápido y eficiente de información crítica.	El sistema debe estar diseñado para permitir registros rápidos en urgencias.	Sí	Sí
9	El sistema debe contar con ayudas accesibles e integradas para apoyar a los usuarios en todo mo-	La lógica del sistema debe contemplar ofrecer soporte y ayudas integradas accesibles.	Sí	Sí

	mento.			
10	Es fundamental garantizar interoperabilidad con otras plataformas clínicas para continuidad de la atención.	El sistema debe estar diseñado para asegurar interoperabilidad con otras plataformas clínicas.	Sí	Sí
11	Las capacitaciones deben ser breves, prácticas y enfocadas en el uso real del sistema.	El sistema debe estar diseñado para permitir capacitaciones breves y prácticas.	Sí	Sí
12	Definir y monitorear métricas de uso, satisfacción y experiencia permite evaluar mejoras en usabilidad.	Se debe implementar funcionalidades que permitan definir métricas de uso, satisfacción y ux.	No	Sí
13	Facilitar la colaboración entre profesionales mejora la calidad del trabajo interdisciplinario.	El sistema debe estar diseñado para permitir facilitar la colaboración entre profesionales.	No	Sí
14	El sistema debe guiar al usuario de forma clara cuando se producen errores para que pueda resolverlos.	Debe garantizarse que el sistema sea capaz de guiar al usuario frente a errores.	No	No
15	Antes del despliegue general, es importante realizar pilotos en entornos reales de uso.	El sistema debe estar diseñado para permitir realizar pilotos antes del despliegue completo.	Sí	No
16	Cuando corresponde, se debe considerar a pacientes o cuidadores en el diseño de funcionalidades.	El desarrollo debe considerar mecanismos que permitan involucrar a pacientes o cuidadores cuando corresponda.	Sí	Sí
17	Integrar alertas proactivas ayuda a anticipar riesgos y proteger la seguridad del paciente.	El sistema debe estar diseñado para permitir integrar alertas proactivas de seguridad del paciente.	Sí	Sí
18	Se debe priorizar la estabilidad del sistema por sobre la incorporación de nuevas funciones.	La lógica del sistema debe priorizar estabilidad sobre novedades.	Sí	No
19	Resúmenes clínicos deben ser claros, concisos y contener lo esencial para la atención.	El desarrollo debe considerar mecanismos que permitan generar resúmenes clínicos claros y sintéticos.	Sí	Sí
20	Las herramientas del sistema deben apoyar al personal sin reemplazar su juicio clínico.	El sistema debe estar diseñado para permitir que apoyar la toma de decisiones sin reemplazar el juicio clínico.	Sí	Sí
21	Diseñar considerando diferencias generacionales mejora la inclusión y el uso por todo tipo de usuarios.	Se deben desarrollar las funcionalidades considerando la brecha de conocimiento entre diferentes rangos de edad.	Sí	Sí
22	El sistema debe estar disponible y responder de forma estable durante toda la jornada clínica.	Se debe implementar funcionalidad que permita que asegurar disponibilidad y respuesta del sistema.	Sí	Sí
23	Los procesos de registro deben ser rápidos, intuitivos y con validaciones apropiadas.	El sistema debe estar diseñado para permitir diseñar procesos de registro eficientes y seguros.	Sí	Sí
24	Validar los datos ingresados previene errores clíni-	Debe garantizarse que el sistema sea capaz de validar datos para	Sí	Sí

	cos y administrativos.	evitar errores de registro.		
25	Usar plantillas mejora la eficiencia en registros repetitivos y reduce el riesgo de omisiones.	Se debe implementar funcionalidades que permita que usar plantillas para entradas repetitivas.	Sí	Sí
Sección "Evita" de la lista de Experiencia Usuaria y Funcionalidades				
1	Debe evitarse que el sistema requiera pasos innecesarios para realizar tareas simples.	El sistema debe estar diseñado para permitir que las tareas sean simples.	Sí	Sí
2	No se debe asumir que los usuarios tienen conocimientos técnicos avanzados.	La lógica del sistema debe contemplar que la experiencia técnica de los usuarios.	Sí	No
3	Debe permitir introducir cambios sin impactar al usuario.	Se deben realizar cambios evaluando su impacto y dando aviso a los usuarios.	No	Sí
4	El sistema debe mostrar información irrelevante para la actividad en curso.	Se debe implementar funcionalidad que permita mostrar información relevante para la actividad en curso.	No	Sí
5	Es fundamental que el sistema no permita que fallas interrumpan el flujo clínico.	Se debe implementar funcionalidades que permitan avanzar en el flujo clínico.	Sí	No
6	Debe evitarse sobrecargar la experiencia con validaciones innecesarias.	Debe garantizarse que el sistema sea capaz de facilitar el trabajo reduciendo el número de validaciones redundantes.	Sí	Sí
7	No se debe exigir permisos administrativos para acceder a funciones críticas.	La lógica del sistema debe contemplar exigir permisos específicos para funciones críticas.	No	Sí
8	Es un riesgo liberar funciones experimentales directamente en entornos productivos.	El desarrollo debe considerar mecanismos que permitan ocultar funciones experimentales en producción.	Sí	Sí
9	No se deben imponer flujos rígidos que no consideren la diversidad de contextos clínicos.	Debe garantizarse que el sistema sea capaz de adaptarse a flujos haciéndolo universal.	Sí	Sí
10	Deben considerarse las limitaciones del hardware al diseñar el sistema.	Se debe reconocer las limitaciones de hardware del cliente.	No	Sí
11	No basta con reuniones; se deben usar diversos métodos para obtener retroalimentación.	La lógica del sistema debe contemplar más métodos y no depender solo de reuniones para recolectar feedback.	Sí	No
12	No se debe confiar únicamente en métricas técnicas para evaluar la experiencia de usuario.	Debe garantizarse que el sistema sea validado no solo con métricas técnicas para evaluar ux.	Sí	Sí
13	Excluir a los usuarios del diseño y pruebas afecta negativamente la usabilidad.	Nunca se debe excluir a los usuarios del diseño y prueba del sistema.	Sí	Sí
14	No se puede asumir que una solución funcionará	Se debe evitar asumir que lo que funciona en un centro sirve para	Sí	Sí

	igual en todos los centros clínicos.	todos.		
15	No debe modificarse un proceso clínico solo para que se ajuste al software.	La lógica del sistema debe contemplar que alterar procesos clínicos no es factible.	Sí	Sí
Sección "Haz" de la lista de Elementos de la Interfaz Usuaria				
1	Utilizar colores con significado clínico facilita la rápida interpretación de información.	La lógica del sistema debe contemplar usar colores con significado clínico consistente.	Sí	Sí
2	Los botones deben ser grandes, legibles y estar bien distribuidos para facilitar su uso.	El sistema debe estar diseñado para permitir botones grandes, claros y accesibles.	No	Sí
3	El contraste adecuado entre texto, botones y fondo es clave para accesibilidad visual.	El desarrollo debe considerar mecanismos que permitan mantener alto contraste entre texto, botones y fondo.	Sí	Sí
4	Ubicar elementos comunes en lugares consistentes mejora la navegación del sistema.	La lógica del sistema debe contemplar ubicar elementos comunes de forma consistente.	Sí	Sí
5	Los íconos deben ser sencillos, reconocibles y acompañados de etiquetas para mayor claridad.	El desarrollo debe considerar mecanismos que permitan usar íconos intuitivos, estandarizados y sencillos con etiquetas.	Sí	Sí
6	El diseño debe adaptarse a diferentes tamaños de pantalla y tipos de dispositivos.	Debe garantizarse que el sistema sea capaz de adaptar el diseño a distintos dispositivos.	Sí	No
7	Es importante jerarquizar visualmente las alertas según su nivel de criticidad.	Debe garantizarse que el sistema sea capaz de aplicar jerarquía visual clara en alertas.	Sí	Sí
8	Acciones críticas deben estar claramente diferenciadas de las secundarias para evitar errores.	La lógica del sistema debe contemplar distinguir acciones críticas de las secundarias.	Sí	Sí
9	Los formularios deben ser breves, claros y destacar los campos obligatorios.	El sistema debe estar diseñado para permitir hacer formularios breves y claros, marcando campos obligatorios.	Sí	Sí
10	Cumplir con estándares de accesibilidad como WCAG o ADA es esencial para inclusión.	Se debe implementar funcionalidades que cumplan estándares de accesibilidad.	Sí	Sí
11	El sistema debe entregar retroalimentación visual inmediata al usuario tras sus acciones.	El desarrollo debe considerar mecanismos que permitan proveer feedback visual inmediato.	Sí	Sí
12	En tareas extensas, es útil mostrar el avance para mantener la orientación del usuario.	Se debe implementar funcionalidad que permita que mostrar progreso en tareas largas.	Sí	Sí
13	Los menús deben organizarse de forma lógica y predecible para facilitar el acceso.	El desarrollo debe considerar mecanismos que permitan organizar menús de forma lógica.	Sí	Sí
14	Es clave mostrar mensajes de confirmación antes de ejecutar acciones importantes.	Se debe implementar funcionalidad que permita contar con mensajes de confirmación claros.	Sí	Sí
15	La navegación debe ser intuitiva y permitir mover-	Debe garantizarse que el sistema sea capaz de garantizar navega-	No	No

	se sin dificultad por todo el sistema.	ción intuitiva.		
16	Los elementos interactivos deben diferenciarse claramente del resto del contenido.	El sistema debe estar diseñado para permitir diferenciar claramente elementos interactivos.	Sí	Sí
17	Se debe permitir deshacer acciones en caso de errores o decisiones accidentales.	El sistema debe estar diseñado para permitir permitir deshacer acciones.	Sí	Sí
18	Es recomendable probar el diseño en pantallas con baja resolución para asegurar legibilidad.	El sistema debe estar diseñado para permitir usar pantallas de baja resolución.	Sí	Sí
19	La información clínica debe tener prioridad visual en la interfaz.	Debe garantizarse que el sistema sea capaz de priorizar visualmente la información clínica.	Sí	Sí
20	Los textos deben ser concisos, comprensibles y libres de ambigüedad.	El desarrollo debe considerar mecanismos que permitan usar textos claros y concisos.	Sí	Sí
21	El sistema debe mostrar claramente su estado (cargando, guardando, etc.) en todo momento.	Se debe implementar funcionalidad que permita que señalar claramente los estados del sistema.	Sí	Sí
22	Aplicar filtros visuales permite destacar lo relevante según contexto clínico.	El sistema debe estar diseñado para permitir que aplicar filtros visuales útiles.	Sí	Sí
23	Diseños revisados por expertos y usuarios mejoran la calidad de la interfaz.	Debe garantizarse que el sistema sea revisado en sus diseños con expertos y usuarios.	Sí	Sí
24	Separar las secciones de forma visual facilita la orientación dentro de la pantalla.	Se debe implementar funcionalidad que permita separar visualmente las secciones.	Sí	Sí
25	El diseño debe alinearse con la identidad visual de la institución, si corresponde.	Se debe implementar funcionalidades que permitan alinear el diseño a la marca institucional según corresponda.	Sí	No
Sección “Evita” de la lista de Elementos de la Interfaz Usuaria				
1	Evitar el uso de colores difíciles de distinguir para no afectar la interpretación clínica.	El sistema debe estar diseñado para usar paletas de colores fáciles de distinguir entre sí.	Sí	Sí
2	Los botones no deben ser pequeños ni estar demasiado juntos, para evitar errores.	Debe garantizarse que el sistema no tenga botones demasiado pequeños y juntos.	Sí	Sí
3	Ubicar elementos de forma inconsistente dificulta la navegación y la experiencia.	La lógica del sistema debe contemplar que ubicar elementos de forma consistente.	Sí	Sí
4	Jerarquías visuales confusas pueden obstaculizar la toma de decisiones.	Debe garantizarse que el sistema no haga uso de jerarquías visuales confusas.	No	Sí

5	Evitar el uso excesivo de colores brillantes en alertas para no saturar visualmente.	Debe garantizarse que el sistema sea capaz de no requerir uso de exceso de colores en alertas.	No	No
6	Formularios largos y desordenados dificultan el ingreso eficiente de datos.	El desarrollo debe considerar mecanismos que faciliten navegar por formularios extensos.	Sí	Sí
7	El sistema debe mostrar claramente los errores; no deben quedar ocultos.	Se debe implementar funcionalidades que permitan que ocultar errores no relevantes sin señalizarlos.	Sí	No
8	Evitar menús sobrecargados que dificulten encontrar funciones clave.	El desarrollo debe considerar mecanismos que eviten crear menús sobrecargados.	Sí	Sí
9	Las animaciones en exceso distraen y pueden impactar el rendimiento.	Se debe implementar elementos de la interfaz estáticos.	No	Sí
10	No se deben usar imágenes de baja calidad que afecten la claridad visual.	El sistema debe estar diseñado para usar íconos vectoriales.	Sí	Sí
11	El sistema debe permitir ajustar el tamaño de fuente para garantizar accesibilidad.	La lógica del sistema debe contemplar que la fuente usada sea siempre visible.	No	No
12	Mensajes ambiguos deben evitarse para no generar confusión en el usuario.	El desarrollo no tiene que tener mensajes ambiguos o confusos.	Sí	Sí
13	Evitar sobrecargar la interfaz con elementos innecesarios que entorpezcan el uso.	Se debe implementar las funcionalidades justas para que no se sobrecargue el sistema.	Sí	No
14	El diseño debe considerar los errores humanos como impracticables.	El desarrollo debe considerar mecanismos que eviten errores humanos.	No	No
15	No se debe usar jerga técnica que los usuarios no comprendan fácilmente.	Se debe implementar funciones que permitan que usar jerga técnica comprensible.	No	Sí