



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

**REINGENIERÍA DE PLATAFORMA DE FICHA MÉDICA DIGITAL PARA
CENTRO KINTUN**

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

VÍCTOR MANUEL GARRIDO URZÚA

PROFESOR GUÍA:
MAURICIO CERDA VILLABLANCA

MIEMBROS DE LA COMISIÓN:
IVÁN SIPIRAN MENDOZA
JOSÉ PIQUER GARDNER

SANTIAGO DE CHILE
2021

Resumen

En las últimas décadas, y a medida que la pirámide demográfica de Chile tiende al envejecimiento, ha aumentado nuestra preocupación como sociedad por los adultos mayores. Con esto, se han impulsado políticas públicas que promuevan la educación y la prevención en torno a enfermedades asociadas a la vejez, como lo son las demencias. En este contexto surge el Centro Comunitario para Personas con Demencias y sus Familias y Entorno Cercano Kintun (2012), con el patrocinio del Servicio Nacional del Adulto Mayor y la Municipalidad de Peñalolén, como también la primera Clínica de Memoria y Neuropsiquiatría, o CMYN (2018).

Un aspecto fundamental de la atención que entregan estas instituciones proviene de la capacidad de manejar a cabalidad la información de sus usuarios. Debido a que son múltiples las disciplinas que se desempeñan para entregar asistencia a los pacientes y sus cuidadores, es de suma importancia tener la capacidad de almacenar y transmitir la información relacionada a las prestaciones entregadas.

Es con esta directriz que, en el año 2019, es desarrollada por alumnos del DCC una ficha médica digital para ser utilizada como sistema de información de los pacientes de CMYN. Esta es creada en conjunto con el personal de la clínica, y demuestra gran utilidad como plataforma tras su despliegue, en especial en tiempos de pandemia donde el trabajo se torna en buena parte remoto. Con esta experiencia de uso, la ficha es una excelente candidata para desempeñarse como sistema de información en otros contextos similares, como lo es el centro Kintun.

Este trabajo de título consiste en realizar una labor de reingeniería de la ficha CMYN, para lograr generalizarla y adaptarla para ser usada por el centro Kintun. Esto plantea el desafío de desacoplar diversas funcionalidades que estaban directamente relacionadas a su ambiente original, como también desarrollar funcionalidades que son críticas para que la ficha sea efectiva en su ambiente nuevo.

Durante este desarrollo se realizó un estrecho trabajo con los integrantes del centro Kintun para entender las similitudes y diferencias en sus necesidades con respecto a CMYN. Es así como se identificaron los puntos más importantes a adaptar y a desarrollar: cambiar la nomenclatura de diversas partes de la plataforma, adaptar un proceso llamado *evaluación integral* a la ficha e integrar facultades de reportería. Estos puntos fueron llevados a cabo en constante validación con sus usuarios mediante reuniones recurrentes en las cuales se exponían y discutían las funcionalidades desarrolladas. Finalmente, se validó el uso de esta plataforma por los usuarios con una prueba de usabilidad utilizando la escala System Usability Scale, la cual reportó un resultado de “Excelente” al reflejarse en una escala adjetiva.

*Es un mundo mágico, Hobbes, viejo amigo...
¡Vamos a explorar!*
– *Calvin y Hobbes, 31 de diciembre, 1995.*

Agradecimientos

Este trabajo no podría haberse realizado sin la inmensa disposición de los integrantes del centro Kintun, que demostraron siempre estar dispuestos a ayudar para sacar todo adelante. Gracias a Michel, Binti, Trinidad y Nubia por prestar su tiempo y esfuerzo todas las semanas para encontrarnos y así encontrar nuestro terreno en común. Gracias a todos los profesionales y miembros que se reunieron conmigo para enseñarme los detalles necesarios y así plasmarlos en este trabajo: Valentina Vega, Valentina Urrutia, Tiare y Margarita. Y aún si no nos vemos, agradezco a todos los integrantes de Kintun por hacer una gran labor en este mundo, asistiendo de forma humana a quienes fuertemente lo necesitan.

Agradezco a Mauricio por entregarme las herramientas y la dirección para realizar este trabajo de título, y tu ayuda en los momentos que las cosas se pusieron más difíciles. En especial, te agradezco por tu excelente actitud en todo momento.

Gracias a Luis Ampuero, por tu trabajo en la ficha, por tus correcciones y comentarios en los PR, y por ayudarme a resolver problemas que parecían irresolubles.

Le doy las gracias a Mati, monito de mi corazón, que estuvo ahí en cada momento, apoyándome cada vez que todo parecía imposible, levantándome cada vez que estaba abajo, y sacándome a abrazos las penas. Sin tenerte a mi lado nada de esto sería real. Gracias por el amor y apoyo que me has dado durante todos estos años, y por todas las aventuras que hemos vivido juntos.

Gracias a mi familia: a mi mamá y a mi papá por su amor incondicional, a mi hermana Natalia por su apoyo lejano en lo geográfico pero muy cercano al corazón, a mis hermanos Miguel y Benjamín, por toda su energía, cariño y gracia, a mi abuela Oby por cada momento que hemos pasado juntos y a mi hermana Sofía por permitirme conocerla cada día un poco más.

A mis amigos. A Vicho por darme la fuerza para terminar este trabajo. A la Gabi por su cariño y locura. A Pedro, Javier, Javiera, Nico, Claudio, Nano, Juanpi, Fede, Lucas, Pipe y Bruno, por acompañarme en esta vida y siempre estar ahí cuando los necesito.

Finalmente, a la Hera, el Polo y el Chidi, que si bien son perros y no pueden leer esto, me han dado inmensa alegría y energía.

Tabla de Contenido

1. Introducción	1
1.1. Antecedentes	1
1.1.1. Contexto	1
1.1.2. Centro Kintun y CMYN	1
1.1.3. Ficha Médica Digital	2
1.1.4. Estructura y procesos de Kintun	4
1.2. Situación Inicial	5
1.3. Objetivos	6
1.3.1. Objetivo General	6
1.3.2. Objetivos Específicos	6
1.4. Metodología	6
1.4.1. Identificación y levantamiento	7
1.4.1.1. Consideraciones durante crisis sanitaria	7
1.4.1.2. Reuniones semanales	7
1.4.1.3. Kanban	7
1.4.2. Refactorización y desarrollo	8
1.4.2.1. Desarrollo iterativo incremental	8
1.4.2.2. Control de versiones	9
1.4.3. Despliegue de la plataforma	9
1.4.3.1. Periodo de prueba	9
1.4.3.2. Capacitación de uso	9
2. Identificación y levantamiento de requerimientos	11
2.1. Necesidades Centro Kintun	11
2.1.1. Plataforma digital	11
2.1.2. Exportación de reportes	12
2.1.3. Aplicación de Evaluación Integral	13
2.1.4. Aplicación de instrumentos	13
2.2. Adaptaciones a la ficha original	13
2.2.1. Generalización para futuro	14
2.2.2. Mejora de integración REDCap	14
2.2.3. Testing	14
3. Refactorización e implementación	16
3.1. Refactorización necesaria	16
3.1.1. Constantes	16
3.1.2. Terminología e Iconografía	18

3.1.3. Captura de información	18
3.2. Módulo Tailor	19
3.3. Nuevas funcionalidades	21
3.3.1. Evaluación Integral	21
3.3.2. Reportería	22
3.4. Optimizaciones ficha original	22
3.4.1. REDCap	22
3.4.2. Testing	23
3.5. Habilitación de ambiente productivo	25
3.6. Creación y habilitación de proyecto en REDCap	25
3.7. Cobertura de requerimientos	26
4. Validación	28
4.1. Reuniones de validación	28
4.2. Encuesta de usabilidad SUS	28
5. Conclusiones	31
Bibliografía	33
Anexo A. Módulo Tailor	34
A.1. Implementación	34
Anexo B. Escala SUS	36
B.1. Resultados completos	36

Índice de Tablas

3.1.	Cobertura obtenida para código nuevo desarrollado para este proyecto, y para código heredado.	24
3.2.	Cobertura de los requerimientos funcionales planteados.	26
3.3.	Cobertura de los requerimientos no funcionales planteados.	27
4.1.	Resumen resultados escala SUS en Kintun	29
B.1.	Resultados particulares por enunciado para los cuatro usuarios encuestados, siendo el puntaje 1: Muy en desacuerdo y 5: Muy de acuerdo.	36

Índice de Ilustraciones

1.1.	Vista principal de un paciente en ficha CMYN, con información sensible oculta.	3
1.2.	Barra lateral que muestra los módulos del sistema.	3
1.3.	Esquema del proceso por el que pasa un usuario que ingresa a Kintun.	5
1.4.	Esquema de componentes ficha CMYN.	6
1.5.	Kanban del proyecto Kintun, con diversos tipos de tareas y estados de avance.	8
1.6.	Flujograma de desarrollo iterativo incremental.	8
1.7.	Ejemplo de captura utilizada para guiar las acciones de los usuarios. En particular, esta captura detalla la manera de entrar a la funcionalidad de creación de nuevos usuarios.	10
2.1.	Formato de la ficha de prestaciones mensuales utilizado por los integrantes del centro Kintun.	12
3.1.	Ejemplo de los estados de los pacientes/usuarios en las plataformas CMYN y Kintun respectivamente	17
3.2.	Diferencias en los logotipos en la pantalla de inicio de sesión.	18
3.3.	Registro de una evolución clínica/prestación, con diferencias en los campos registrados en CMYN y Kintun	19
3.4.	Pantalla de búsqueda de plataforma CMYN, proveniente del Código 3.1. . . .	20
3.5.	Pantalla de búsqueda de plataforma Kintun, proveniente del Código 3.1. Notar la diferencia en la nomenclatura entre Paciente y Usuario.	20
3.6.	Modelo de la entidad abstracta <code>evaluation</code> , con dos modelos que heredan de ella.	21

Capítulo 1

Introducción

1.1. Antecedentes

1.1.1. Contexto

En las últimas décadas, latinoamérica, y en particular Chile, ha experimentado una rápida transición demográfica y epidemiológica. En particular, se registra un acelerado envejecimiento de la población y un aumento de la frecuencia de enfermedades crónicas y degenerativas. La población de tercera edad se ha más que duplicado desde el año 1990 en el país [1], siendo este el segmento más afectado por patologías relacionadas a trastornos de la memoria.

Con el aumento de la esperanza de vida que se registra en Chile, la demografía del país cambia paulatinamente a una población más envejecida. Con esto, se presenta la preocupación por la calidad de vida de las personas mayores para la comunidad médica, científica, y para el Ministerio de Salud del país, en especial respecto al aumento en la incidencia y prevalencia de la demencia, un problema típico de los países desarrollados. Esta preocupación se extiende al ámbito social, pues levanta el desafío de cómo nos hacemos cargo como sociedad de este grupo etario y sus patologías, y qué políticas públicas se decretan para esto.

En este contexto, múltiples iniciativas han sido levantadas en los últimos años por parte del gobierno y los sistemas de salud, tales como el Servicio Nacional del Adulto Mayor (2003) y el Plan Nacional de Demencia (2017). Es así cómo surge el Centro Comunitario para Personas con Demencias y sus Familias y Entorno Cercano Kintun (2012), con el patrocinio del Servicio Nacional del Adulto Mayor y la Municipalidad de Peñalolén [2], como también la primera Clínica de Memoria y Neuropsiquiatría, o CMYN (2018), alojada en el Campus Oriente de la Facultad de Medicina vinculado al Hospital del Salvador.

1.1.2. Centro Kintun y CMYN

El centro Kintun y CMYN ofrecen apoyo interdisciplinario, con enfoque transdisciplinario, a personas con demencia y a sus familias, las cuales son derivadas desde la atención primaria en casos que indiquen síntomas de demencia. El diagnóstico de demencia de una persona por lo general afecta el estado de todo su núcleo familiar, pues requiere cuidados constantes y atención al entorno en el cual se desenvuelve la persona. Sumado a esto, son condiciones de

salud degenerativas que conforme avanza su desarrollo, aumenta la cantidad de síntomas y la carga del cuidador y disminuye la funcionalidad de la persona, por lo que su tratamiento evoluciona a medida avanza la condición. Para poder abordar a las personas con demencia y sus familias, es necesario un equipo de múltiples disciplinas trabajando en conjunto, con integrantes tales como asistente administrativo, nutricionista, geriatra, auxiliares de enfermería, terapeuta ocupacional, kinesiólogo, psicólogo y trabajador social [3].

Es en este contexto que los centros como Kintun y CMYN presentan un desafío tecnológico importante. La característica de transdisciplinariedad de su atención exige que los profesionales mantengan un alto grado de coordinación en sus actividades, tanto en sus tareas de atención, diagnóstico y seguimiento de personas con demencia, como en las tareas administrativas y de contacto con las personas y sus familias. Es entonces como surge la necesidad de trabajar con una plataforma que mantenga un registro de estas actividades asociadas a los personas, con todas las facilidades que nos puede brindar los actuales sistemas de información. Algunas de las características son la inmediatez, facilidad de acceso, capacidad de automatizar procesos e informes, y flexibilidad para registrar y obtener datos [4]. Por otra parte, para situaciones de emergencias, se habilitan otras modalidades de trabajo más flexibles, como lo es el trabajo remoto de los profesionales mediante seguimiento telefónico de los usuarios de Kintun.

1.1.3. Ficha Médica Digital

Debido al gran aporte que puede entregar un sistema que apoye las prestaciones de los centros de memoria, es que en agosto de 2019 se genera un proyecto entre el CMYN y alumnos de Ingeniería Civil en Computación de la Universidad de Chile cursando el ramo “Proyecto de Software”, enfocado en generar una mejora de los sistemas de fichas médicas utilizadas. A partir de este proyecto se crea la ficha CMYN, plataforma web que tiene como objetivo unificar las distintas fuentes de información y manejar los datos de los pacientes y de las atenciones realizadas por los profesionales del centro. La ficha fue puesta en funcionamiento en diciembre de 2019, y según lo reportado por los integrantes de CMYN, ha resultado muy útil como herramienta de información y gestión, en especial en el contexto de trabajo remoto impuesto por la pandemia causada por el COVID-19. Como referencia, en el mes de abril de 2020, trabajando con una cantidad reducida de personal de forma presencial, se registraron aproximadamente 22,5 atenciones diarias promedio por día hábil, la misma cantidad que se registró en el mes de febrero de 2020, en el cual todavía no se iniciaba con el teletrabajo. Con esto se puede evidenciar que las atenciones no han disminuido a pesar de disminuir el personal presencial en CMYN, pues se habilitó la oportunidad de visualizar la información del paciente de forma remota y con esto realizar atenciones telefónicas y registrarlas en la ficha, permitiendo a todos los profesionales acceder a esta información paralelamente.

La vista principal de la ficha de una persona se puede observar en la Figura 1.1, en la cual se muestra las distintas dimensiones de información que la caracterizan. Además de esto, en la Figura 1.2 se muestran los distintos módulos incorporados al sistema de los que hacen uso los usuarios.

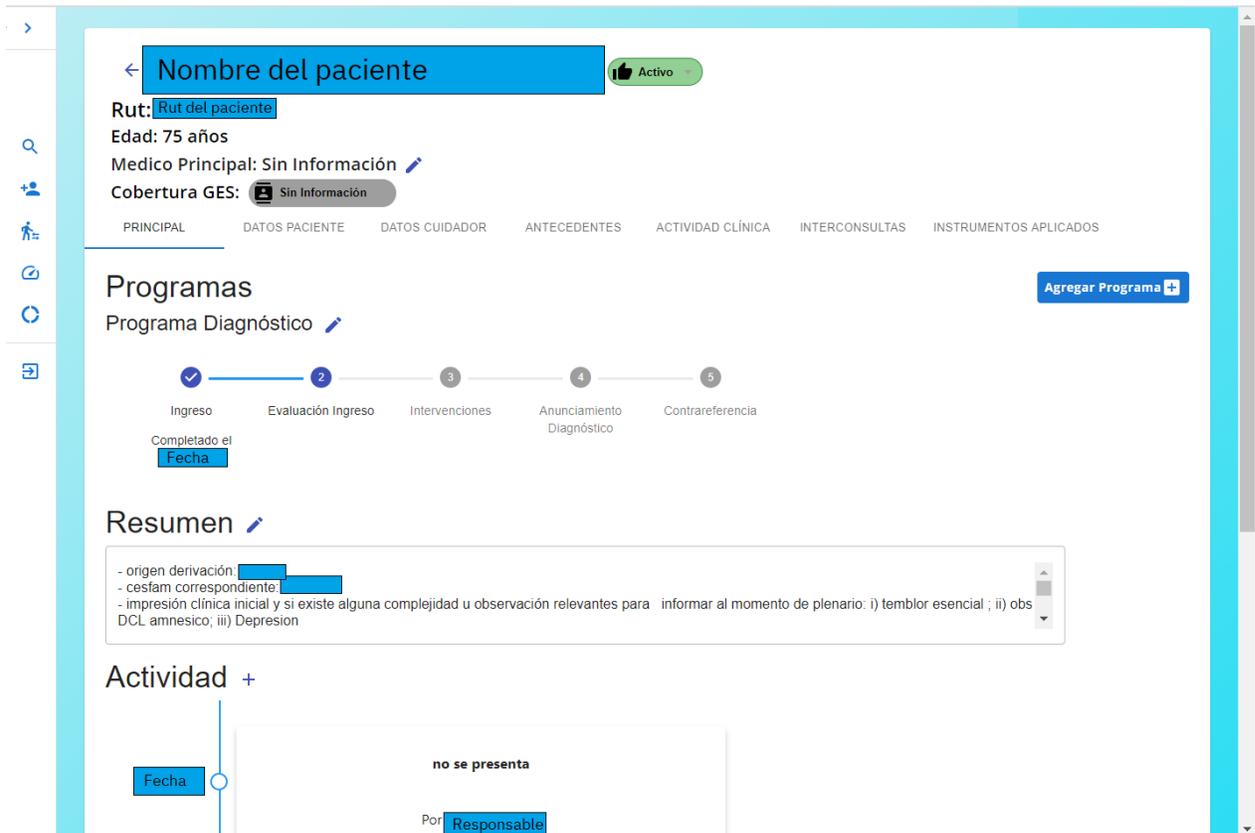


Figura 1.1: Vista principal de un paciente en ficha CMYN, con información sensible oculta.

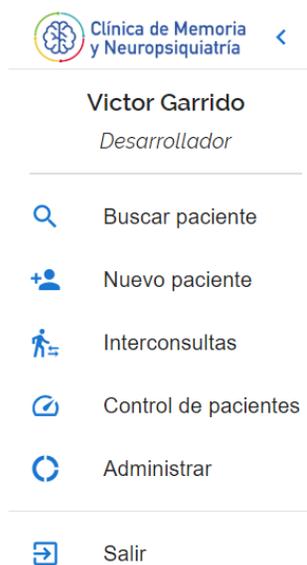


Figura 1.2: Barra lateral que muestra los módulos del sistema.

1.1.4. Estructura y procesos de Kintun

Kintun está actualmente compuesto por 17 integrantes, distribuidos entre asistente administrativo, personal auxiliar, personal técnico, personal profesional no médico, y personal profesional médico. Cada uno de estos cumpliendo labores específicas:

- **Asistente Administrativo:** Se encarga de agendar las horas médicas, gestionar las compras, pagos y gastos del centro, como también mantener y reportar estadísticas mensuales.
- **Personal Auxiliar:** Ayuda en los talleres de Kintun, transporta a los usuarios y prepara materiales.
- **Personal Técnico:** Además de realizar las tareas que realiza el personal auxiliar, se encargan de los medicamentos y de realizar seguimiento sobre ellos, y realizan labores de apoyo al personal profesional.
- **Personal Profesional No Médico:** Atienden a los usuarios de Kintun de forma presencial y remota. Se encargan de gestionar los casos de cada usuario. Realizan las evaluaciones a los usuarios. Entre sus especialidades, se encuentran la psicología, nutrición, kinesiología, terapia ocupacional y asistencia social.
- **Personal Profesional Médico:** Corresponde al personal médico que atiende a los usuarios. Entre sus especialidades, se encuentra la geriatría, neurología y psiquiatría.

Kintun trabaja en un modelo de gestión de casos basado en la persona, donde un profesional se hace cargo de cada usuario. Cuando ingresa un usuario hay un proceso en el cual se hace una evaluación denominada *Evaluación Integral*. Esta evaluación se realiza desde cuatro perspectivas distintas: social, funcional, cognitiva y biomédica, y tiene como objetivo conocer a la persona en todas sus dimensiones. A partir de esta evaluación se determina el ingreso de la persona al centro Kintun; de ser así, los objetivos y las tareas respectivas y la posibilidad de participar en los talleres grupales. El resultado concreto de la evaluación integral es un plan de intervención de tratamiento integral (PITI), el cual es particular para cada persona según sus necesidades, y va siendo modificado según las necesidades que sienten los usuarios del Kintun.

En los próximos 6 meses el usuario recibe atenciones alineadas con lo que se determina en la evaluación integral de ingreso. Además se realizan atenciones multidisciplinarias que intentan dar respuesta a las necesidades sentidas por los usuarios, ya sean talleres grupales como consejerías de salud. En la mitad de este periodo, existe una evaluación realizada por el referente de la persona, donde en algunos casos se da de alta del centro. Transcurridos los 6 meses, se realiza la misma evaluación integral pero de egreso. En algunos casos especiales el usuario continúa recibiendo más intervenciones hasta un plazo máximo de 3 meses después de esta evaluación, pero el plazo máximo de estadía en el centro es de 9 meses.

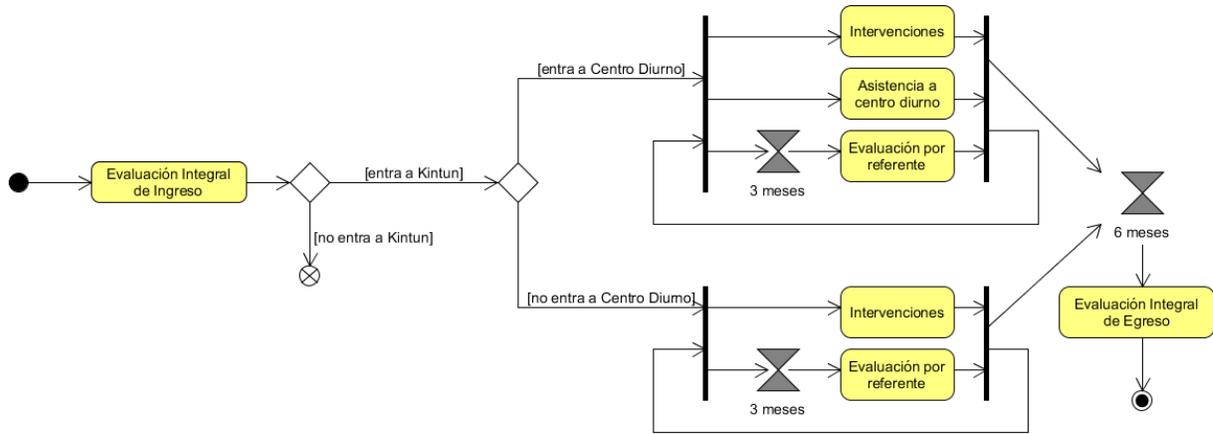


Figura 1.3: Esquema del proceso por el que pasa un usuario que ingresa a Kintun.

Cabe notar que esta evaluación incorpora el trabajo de distintos profesionales para dar una mirada polifacética del usuario de Kintun, la cual incluye la realización de múltiples pruebas, llamadas *instrumentos*, que se utilizan como herramientas de caracterización. Los instrumentos son actualmente registrados de forma física en papel.

1.2. Situación Inicial

En el centro Kintun no se tiene un sistema formal de ficha electrónica de sus usuarios. El manejo de material digital en el centro se realiza a partir de carpetas compartidas con diversos archivos como documentos y hojas de cálculo. Es de gran necesidad la utilización de un sistema más establecido para el manejo de esta información y procesos.

En la agenda digital de Chile se propone como meta que a través de la estrategia SIDRA (Sistemas de Información de la Red Asistencial) se cuente con una ficha clínica electrónica para el 80% de los pacientes del sistema público de salud para el año 2020 [5]. Existen en el mercado distintas soluciones de fichas electrónicas entregadas como servicio, tales como Rayen y OMIap, ambas utilizadas en el sistema de salud primario del país. Sin embargo, estas soluciones no están adaptadas para un centro de atención especializada como lo es Kintun. En general no están enfocadas de forma principal para la utilización regular por parte de personal que no sea médico, los cuales en este caso caracterizan buena parte de los integrantes del centro Kintun. Además de esto, se requiere que la ficha electrónica habilite la transdisciplinariedad de las atenciones, permitiendo integraciones horizontales, mientras que por lo general las fichas médicas digitales en el mercado están enfocadas a atenciones médicas separadas, caracterizadas por silos verticales. Por último, tramitar y costear el uso de una solución comercial como estas es una dificultad administrativa para el centro Kintun, que no se ha podido superar en el pasado.

La ficha CMYN, si bien joven y de funcionalidades sencillas, se diseña para responder a necesidades muy similares a las que tiene Kintun. En cuanto a parámetros técnicos, es

una plataforma web alojada en el data center de la Universidad de Chile¹. Se tiene un servidor encargado de entregar las funcionalidades backend de la plataforma, programado en Python con el framework Django, integrado a una base de datos PostgreSQL. El frontend está programado en Javascript usando el framework web React, que consume las API REST expuestas por el backend, como se muestra en la Figura 1.4. Se tiene acceso al código fuente y al conocimiento interno de su funcionamiento debido a la experiencia anterior de trabajo en la ficha CMYN.

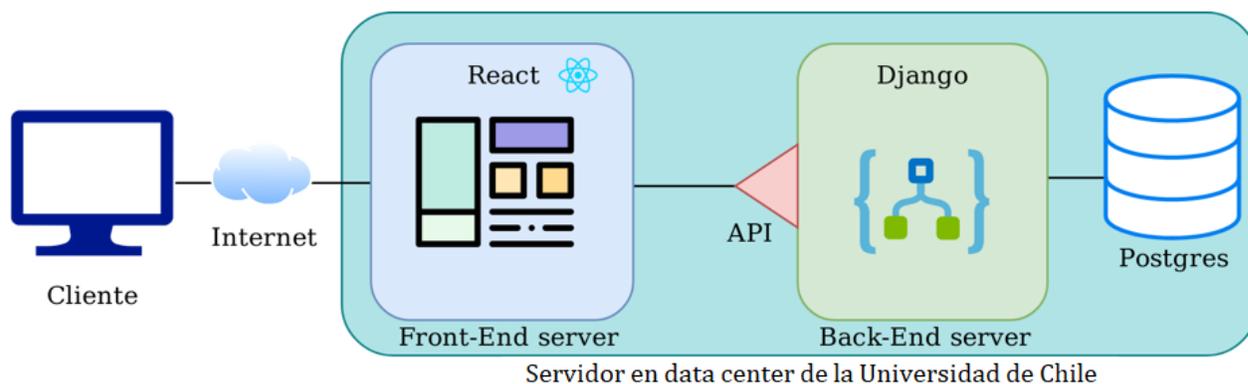


Figura 1.4: Esquema de componentes ficha CMYN.

1.3. Objetivos

1.3.1. Objetivo General

Adaptar las funcionalidades de la ficha médica electrónica CMYN para el funcionamiento en otros centros de memoria, en particular en el centro Kintun, incorporando en el proceso la identificación y desarrollo de nuevos requerimientos que sean críticos para el funcionamiento en este lugar.

1.3.2. Objetivos Específicos

1. Identificación de funcionalidades de la plataforma, y levantamiento de requerimientos críticos para su funcionamiento.
2. Refactorización y generalización de la plataforma para su funcionamiento en otros centros.
3. Despliegue de la plataforma en centro Kintun, considerando un proceso integral de adopción y prueba de la ficha por parte de los usuarios.

1.4. Metodología

A continuación, se detalla la metodología utilizada para responder a los objetivos específicos desarrollados durante este proyecto.

¹ SASIBA, Data Center <http://redec.med.uchile.cl/sasiba>

1.4.1. Identificación y levantamiento

Para identificar las funcionalidades que tiene la plataforma, cómo estas responden a las necesidades del centro Kintun, y qué desarrollo es necesario hacer para cerrar la brecha entre lo que se tiene y lo que se necesita, se trabaja directamente con los integrantes del centro Kintun, quienes serán los futuros usuarios de la plataforma.

En este ámbito se opta por utilizar prácticas de la metodología Agile [6] de forma de visibilizar de mejor forma el proyecto a medida que este va avanzando, como también priorizar las funcionalidades y cambios que los integrantes del centro Kintun estimen más convenientes.

1.4.1.1. Consideraciones durante crisis sanitaria

Debido a que el desarrollo de este proyecto se ve inmerso en el contexto de la crisis sanitaria causada por el virus COVID-19, hay que tomar en cuenta diversos ajustes a la forma de trabajar que debieron ser realizados para lidiar con esta situación. Se debe considerar que todo el trabajo, tanto de levantamiento como de desarrollo, fue realizado de forma remota. Esto tiene especial relevancia en las reuniones con el equipo Kintun, pues se realizaron todas en formato de videollamada. Afortunadamente el equipo del centro Kintun demuestra una excelente disposición a la hora de realizar estas reuniones, como también al disponibilizar documentos relevantes al proyecto.

1.4.1.2. Reuniones semanales

Para realizar la validación continua de las funcionalidades desarrolladas, como también para levantar información atinente al proyecto y priorizar las tareas pendientes, se realizaron reuniones semanales con los integrantes de Kintun. La estructura e impacto de estas reuniones se discuten en la sección 4.1.

1.4.1.3. Kanban

Kanban es un sistema de seguimiento de tareas que de forma sencilla permite conocer el estado de estas, haciendo seguimiento desde su origen, su etapa de desarrollo, pruebas y estado final de aceptación. Este seguimiento es una herramienta útil tanto para el desarrollador como para las demás personas involucradas en el proyecto, pues permite comprender a simple vista el estado general del proyecto, las tareas que quedan pendientes y los avances obtenidos en una última ventana de tiempo.

En este proyecto, de forma más concreta, se utilizó la aplicación Trello² para generar una versión digital de un Kanban, que permitió trabajar de forma online el tablero con sus tareas. Adicionalmente, esta aplicación habilitó la capacidad de agregar información a las tareas, tales como etiquetas, descripción, lista de tareas y archivos adjuntos. En la siguiente figura, se muestra el tablero con sus columnas emblemáticas, en un estado parcial del desarrollo.

² Trello helps teams move work forward, <https://trello.com/home>

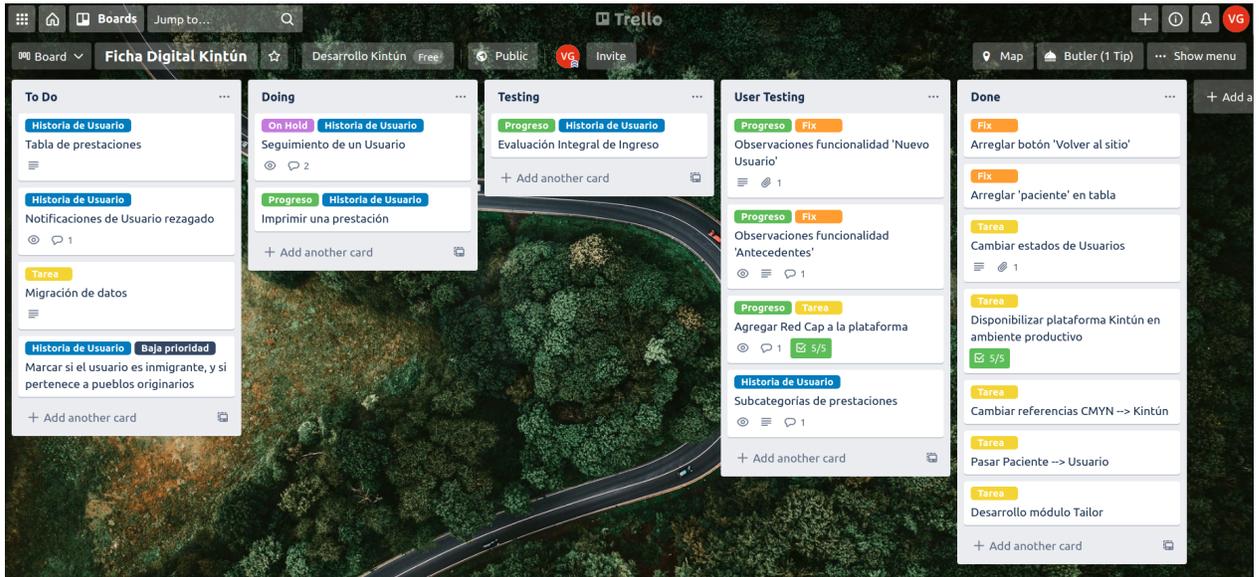


Figura 1.5: Kanban del proyecto Kintun, con diversos tipos de tareas y estados de avance.

1.4.2. Refactorización y desarrollo

1.4.2.1. Desarrollo iterativo incremental

Como metodología general para llevar a cabo el proyecto se utilizó el desarrollo iterativo incremental [7], pues este calza de forma beneficiosa con el modelo de identificación y levantamiento de necesidades.

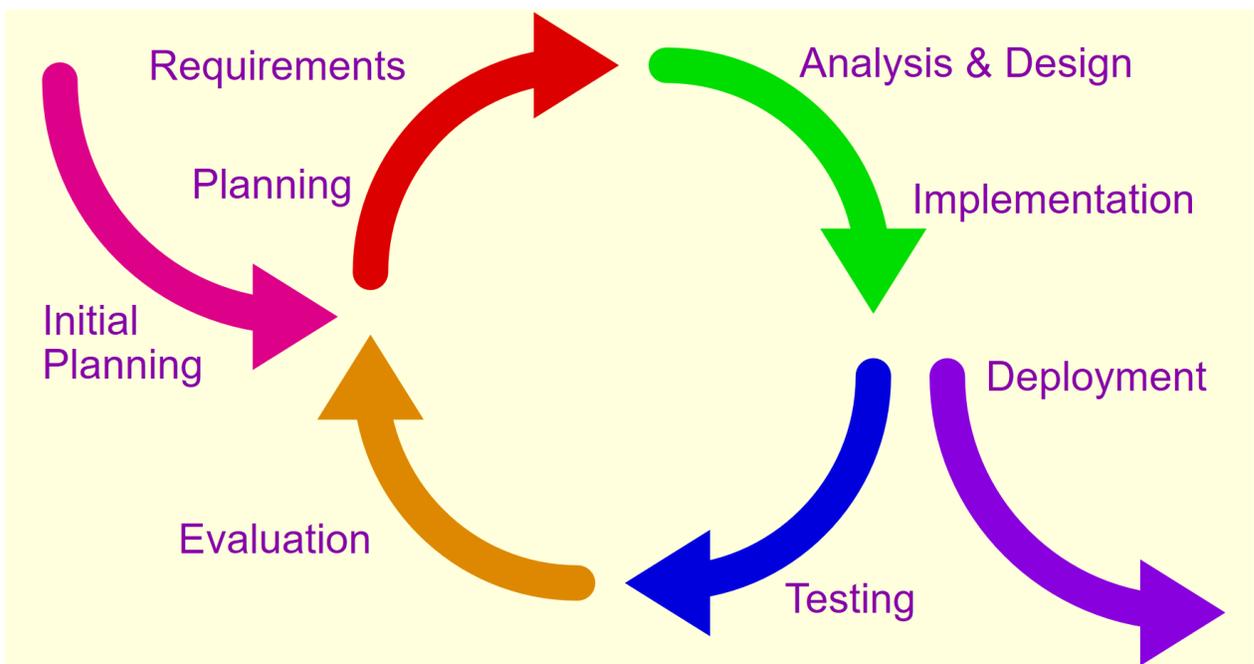


Figura 1.6: Flujo de desarrollo iterativo incremental.

El desarrollo iterativo en etapas de dos semanas, con revisión en todas las semanas, permite

ir agregando valor de forma constante y tangible a la plataforma.

1.4.2.2. Control de versiones

El código original del proyecto está almacenado utilizando el sistema de versionamiento Git, en la plataforma Github. Para continuar el trabajo en la ficha, se continua realizando desarrollo sobre el mismo repositorio, de forma que las funcionalidades puedan ser aplicadas también en CMYN, o a futuro, en cualquier otro centro que busque disponibilizar esta ficha.

Para esto se utiliza el modelo de Git Feature Branches³, con el cual se trabajan en distintas ramas asociadas a funcionalidades o avances, que se van probando y luego agregando a la rama principal. Como el trabajo para este proyecto fue desarrollado por una sola persona, este modelo de trabajo se utiliza principalmente para preservar el orden y tener una visión clara de la evolución del proyecto. La rama principal, *master* en el caso de este proyecto, sólo contiene código probado y listo para producción, tanto en el centro Kintun como en CMYN, por lo tanto debe estar probado para ambos ambientes.

1.4.3. Despliegue de la plataforma

Una vez realizado el despliegue de la plataforma desde un punto de vista técnico, se consideran los siguientes elementos para lograr el objetivo de realizar un proceso integral de adopción de la ficha.

1.4.3.1. Periodo de prueba

Para que los usuarios del centro Kintun puedan probar y validar la plataforma, se utiliza un despliegue temprano que les permita entrar y familiarizarse con su uso. Durante el desarrollo, se permite a los usuarios utilizar todas las funcionalidades de la plataforma a medida se avanza en las iteraciones de desarrollo. Esto lo pueden realizar sin perjuicio de su futuro uso en producción, debido a que finalizado este periodo las bases de datos se vacían para comenzar con el uso productivo de la plataforma.

1.4.3.2. Capacitación de uso

Para capacitar a los usuarios en el uso de la plataforma, se considera utilizar varios métodos. El primero es que, durante las reuniones semanales con los usuarios, se realice el uso guiado de la plataforma, enseñando de esta manera su utilización y despejando dudas a medida que surgen de forma natural. El segundo es material instructivo entregado a los usuarios, de forma de que estos puedan realizar el uso de la plataforma por su cuenta. Este material instructivo se acompaña de capturas de la interfaz, dejando claro la relación entre instrucciones y la interfaz de la plataforma.

³ Git Feature Branch Workflow, <https://www.atlassian.com/git/tutorials/comparing-workflows/feature-branch-workflow>

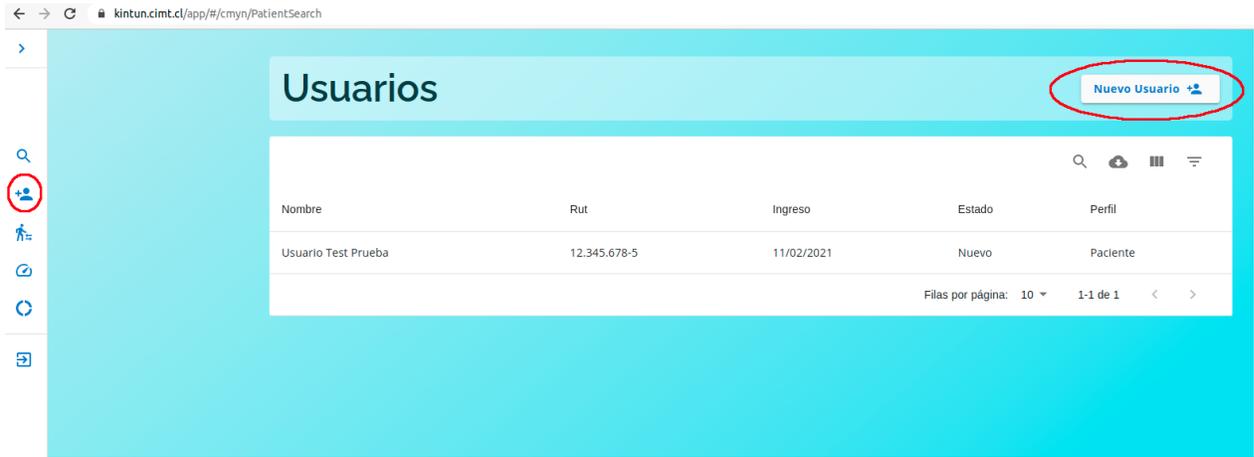


Figura 1.7: Ejemplo de captura utilizada para guiar las acciones de los usuarios. En particular, esta captura detalla la manera de entrar a la funcionalidad de creación de nuevos usuarios.

Capítulo 2

Identificación y levantamiento de requerimientos

Este capítulo detalla el trabajo realizado durante la fase de levantamiento de las capacidades de la ficha original CMYN, como también las necesidades y requerimientos planteados por Kintun. A partir de esta información se expresan requerimientos funcionales [RF] y no funcionales [RN] que guiarán el desarrollo del proyecto.

2.1. Necesidades Centro Kintun

2.1.1. Plataforma digital

Como se plantea anteriormente en este informe, Kintun actualmente no posee un sistema digital para sus fichas médicas. Esto se validó en las reuniones realizadas. Es importante entender además las restricciones obligatorias que tienen en Kintun en cuanto a permanencia de la información. Debido a los requisitos legales del sistema de salud, existe una obligación de poder presentar un respaldo de las prestaciones realizadas de los últimos 15 años. Es por tanto que las fichas físicas de sus usuarios son mantenidas con todas las intervenciones realizadas.

Si bien existe esta necesidad de mantener la ficha médica de papel, es de suma necesidad para los integrantes de Kintun tener alguna manera de acceder a la información de forma remota. En el centro utilizan carpetas compartidas en una red local donde se guardan documentos y tablas para cada usuario del centro. Estas carpetas no se pueden acceder fuera de la red local, lo que se ha vuelto una necesidad imperante en el contexto de la pandemia de COVID-19. Es por esto que han recurrido a utilizar carpetas compartidas en Google Drive.

El equipo de Kintun plantea la importancia de tener una plataforma dedicada accesible de forma online por sobre las demás necesidades y funcionalidades. Además de esto, es necesario que además de ser accesible, la plataforma entregue la información de forma clara y concisa, permitiendo así a los profesionales utilizar esta información para entregar una atención más completa a sus usuarios.

Requerimientos:

- [RN-01] La plataforma debe ser accesible desde cualquier dispositivo con conexión a internet.
- [RN-02] La información dentro de la plataforma debe ser entregada de forma clara y concisa.
- [RN-03] Los tiempos de respuesta al interactuar con la aplicación deben ser cortos. Ninguna interacción debiese hacer esperar al usuario más de 1 segundo de forma de no interrumpir su flujo de trabajo [8].

2.1.2. Exportación de reportes

Uno de los dolores que presentan los futuros usuarios de la ficha es que se utiliza una buena porción del tiempo en generar archivos y documentos informativos sobre los usuarios y las prestaciones entregadas. Se plantea entonces que esta información puede ser rápidamente recopilada y exportada por la ficha, tomando en cuenta que ya fue ingresada anteriormente en los diversos procesos del día a día. De esta forma se puede reducir el tiempo en que los usuarios trabajan en tareas administrativas, y así permitirles priorizar otras tareas, aumentando las horas de trabajo disponibles efectivas. En particular, uno de los documentos que más se necesita automatizar es el denominado *Registro de prestaciones mensuales*, tomando las prestaciones ingresadas en el sistema como fuente de verdad.

Nombre usuario/a Registro de Prestaciones 				
<i>mes</i>				
Fecha	Prestación	Resultados/Observaciones	Entrevistado/a	Responsable (Profesional o interno/a)
XX/XX/2020	EJ: VDI, VDS, Seguimiento telefónico, CSM, actualización de medicamentos	✓		
		✓		
		✓		
		✓		
		✓		
		✓		
		✓		

Figura 2.1: Formato de la ficha de prestaciones mensuales utilizado por los integrantes del centro Kintun.

Requerimientos:

- [RF-01] La plataforma debe permitir exportar las atenciones brindadas por un integrante del centro Kintun durante cierto intervalo de tiempo, en particular, durante el

último mes.

- [RF-02] La plataforma debe permitir exportar las atenciones brindadas a un usuario durante cierto intervalo de tiempo, en particular, durante el último mes.

2.1.3. Aplicación de Evaluación Integral

Una parte importante del flujo por el que pasa un usuario en Kintun es la Evaluación Integral, el cual es un hito al entrar al Kintun, antes de egresar, y posiblemente dentro de las evaluaciones que se hacen rutinariamente. Es por tanto que un sistema digital como lo es esta ficha, debe tener como pieza esta evaluación.

Requerimientos:

- [RF-03] La plataforma debe permitir aplicar una evaluación integral a un usuario.
- [RF-04] La evaluación integral debe contar con autoguardado.

2.1.4. Aplicación de instrumentos

La utilización de la ficha digital también habilita la oportunidad de trabajar con la plataforma REDCap, que permite la creación y aplicación de instrumentos. Esta robusta herramienta contiene una amplia gama de funcionalidades que concede la capacidad de configurar y aplicar instrumentos con múltiples cualidades, tanto a la hora de aplicarlos como de extraer información de sus resultados. Además de esto, la ficha digital está integrada con REDCap, permitiendo visualizar desde la ficha misma información de los instrumentos aplicados a cada usuario de Kintun.

Como segundo beneficio en este ámbito, se tiene la oportunidad de importar los instrumentos utilizados en CMYN, los cuales comparten dominio con los de Kintun. Es así como se pueden aplicar instrumentos relevantes desde el primer día, sin necesidad (pero con la capacidad) de configurar y crear instrumentos nuevos para ser aplicados.

Requerimientos:

- [RF-05] La plataforma debe integrar con REDCap permitiendo la creación y edición de instrumentos.
- [RF-06] La integración de REDCap debe permitir llenar un instrumento para un usuario, y visualizarse en la ficha.

2.2. Adaptaciones a la ficha original

El desarrollo realizado en este proyecto no solo busca mejorar los flujos de trabajo del centro Kintun, sino que también responde a mejorar esta ficha digital para convertirla en una mejor plataforma, independiente del usuario de esta. Esto se logra agregando nuevas funcionalidades a la ficha, como también mejorando la calidad del código del repositorio original.

2.2.1. Generalización para futuro

Como este proyecto conlleva llevar la ficha a un ambiente distinto al original en el que fue planteada, esto significa que se debe encontrar una manera de generalizar la ficha, y esta manera debe ser utilizada en el contexto del centro. El beneficio de esto es que en un futuro la manera de llevar la ficha a un nuevo ambiente ya estará instaurada, y por lo tanto será más sencilla su extrapolación.

Requerimientos:

- [RN-04] Las diferencias que tenga la ficha según el ambiente en el que se utilice deben tener una implementación limpia, que sea fácil de utilizar en el futuro.

2.2.2. Mejora de integración REDCap

La ficha en su estado actual posee una integración con la plataforma REDCap. Sin embargo, esta muestra falencias en la velocidad que tiene a la hora de consultar la data a la API de REDCap. Debido a la cantidad de instrumentos que se manejan, y a la estructura en la que se devuelven los datos, para un usuario que ya tiene N instrumentos llenados del total de M instrumentos posibles, el tamaño de la respuesta a la consulta para ver sus datos tiene una cota superior asintótica de:

$$O(N \cdot M) \tag{2.1}$$

Es de esperarse que la respuesta entregada no escale con la cantidad de instrumentos posibles, idealmente consiguiendo una cota de:

$$O(N) \tag{2.2}$$

Esto resulta en consultas que pueden demorar desde varios segundos a minutos en retornar. Considerando que las personas tienen hasta un máximo de 10 segundos de atención antes de perder la concentración y cambiar a otra tarea [9], es importante lograr optimizar esta funcionalidad para lograr mejores tiempos de respuesta.

Requerimientos:

- [RN-05] Los instrumentos llenados en REDCap deben visualizarse dentro de la ficha Kintun en un tiempo acotado, menor a 10 segundos.

2.2.3. Testing

Un aspecto importante del desarrollo es garantizar la calidad y robustez del software que se está desarrollando. Una manera de hacer esto es generar tests unitarios para validar el comportamiento de las partes construidas, como también prevenir que cambios a futuro impacten en una funcionalidad ya desarrollada. En el desarrollo de software, los tests se pueden escribir de forma de que sean ejecutados de forma ligera cada vez que se necesite validar el estado del desarrollo. La ficha original fue desarrollada con nulo testing, por lo tanto mejorar este ámbito va a mejorar la calidad del código de la ficha.

Requerimientos:

- **[RN-06]** Las nuevas funcionalidades implementadas en la plataforma deben tener test unitarios.

Capítulo 3

Refactorización e implementación

A continuación, se detalla cuales son los puntos que se desarrollan como solución a las problemáticas y a las necesidades del centro Kintun.

3.1. Refactorización necesaria

Una de las principales debilidades que presenta la ficha CMYN para el centro Kintun es que fue diseñada y desarrollada de forma estrechamente ligada a los procesos de CMYN. Es por tanto que en el proceso de implementar la ficha para Kintun, e idealmente disponibilizarla para que también se pueda utilizar en otros centros, se debe encontrar la manera de desacoplar las funcionalidades de la ficha de los procesos originales según cuales se implementaron.

Es así como se identifican distintos grupos de temáticas que se deben abordar para lograr esta refactorización y permitir generalizar el uso de la ficha. Estos grupos se exponen a continuación.

3.1.1. Constantes

Hay definiciones de datos que están dentro del código fuente del proyecto, y que corresponden a información concreta que se asocia al funcionamiento del día a día de la ficha. Esta información comúnmente se encuentra en menús de selección dentro de la plataforma. Algunas de las constantes que se encuentran son:

1. Nombre de los centros asociados. En CMYN se registran múltiples CESFAM asociados por toda la zona oriente de Santiago (24 en total), mientras que en Kintun son distintos sectores de CESFAM específicos (16).
2. Nombre de exámenes indicados a realizar. En CMYN utilizan una lista exámenes como Hemograma o Perfil Lipídico (23), mientras que Kintun es un subconjunto de estos (11)
3. Prestaciones entregadas. En CMYN se entregan 16 prestaciones, mientras que en Kintun se utilizan 42 prestaciones distribuidas en 11 categorías distintas.
4. Tipos de conclusiones asociadas a las prestaciones. En CMYN se categorizan mientras que en Kintun se registran de forma libre.

5. Categorías de antecedentes. Nuevamente en CMYN se categorizan mientras que en Kintun se registran de forma libre.
6. Estados de los pacientes/usuarios. El estado en el que pueden estar los pacientes/usuarios difiere en ambos centro. Ver el ejemplo de la figura 3.1.

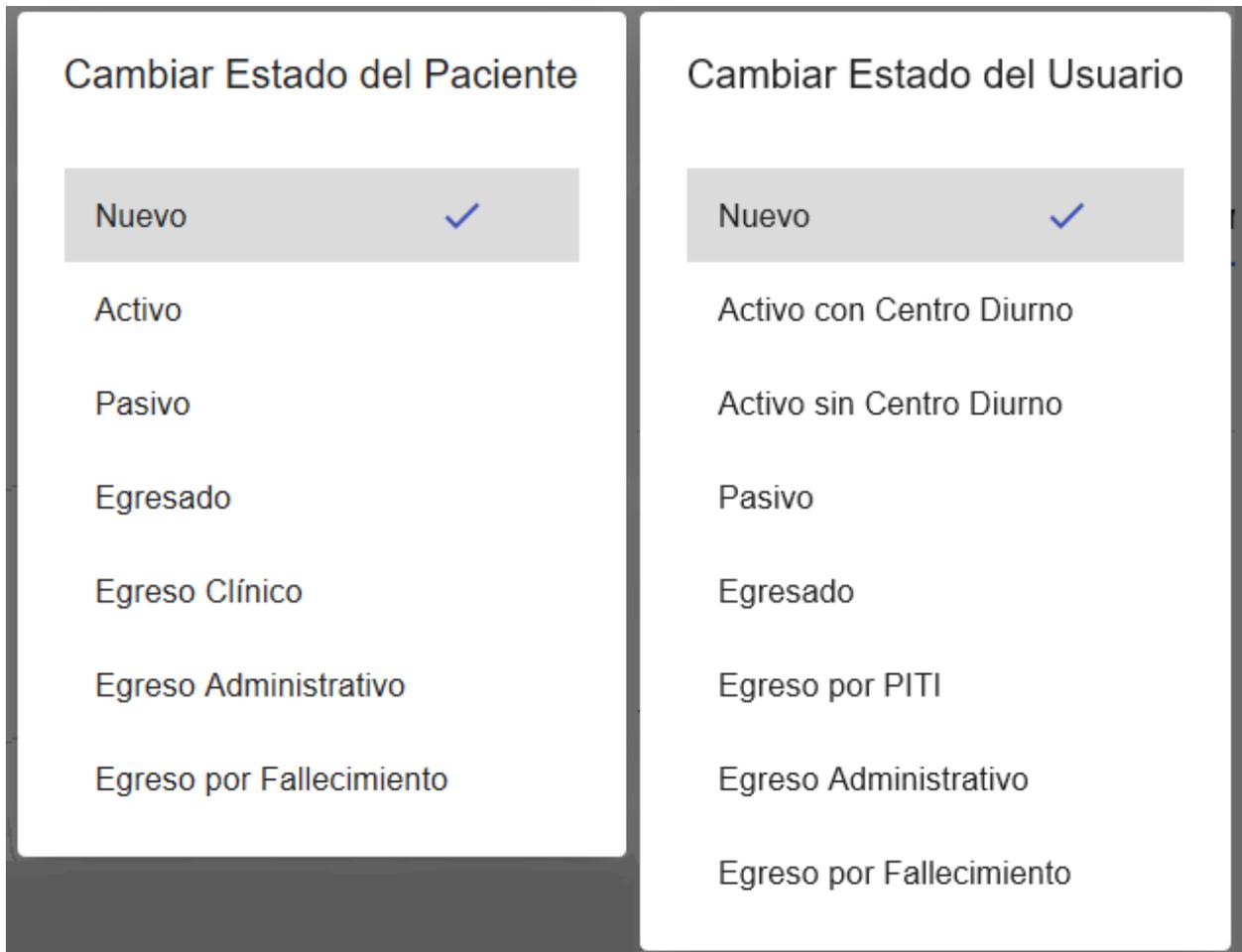


Figura 3.1: Ejemplo de los estados de los pacientes/usuarios en las plataformas CMYN y Kintun respectivamente

Para refactorizar estos valores se puede hacer que se carguen de lugares distintos dependiendo del centro en el que la ficha está configurado. Eso puede ser tanto en archivos de configuración separados que se utilizan en el momento de construcción¹, como también almacenando los valores en base de datos en vez del código fuente. Notar que almacenarlo en la base de datos trae como consecuencia que esta debe ser poblada para que la ficha funcione correctamente, como también que cada vez que se accede a esta información se incurre en una transacción a la base de datos con el tiempo y el procesamiento que esto conlleva.

¹ En este contexto *Build time*, o momento de construcción, es cuando se enlazan los archivos que componen el código fuente del proyecto para generar el programa que termina siendo ejecutado

3.1.2. Terminología e Iconografía

Además de los valores utilizados como constantes en la ficha, hay otro cambio que es necesario hacer para que esta esté completamente adaptada al lugar en el que se esté utilizando, y este es que se debe adoptar la terminología y la iconografía acorde al caso. Un ejemplo concreto, es que en Kintun se desfavorece utilizar el término *paciente*, prefiriendo el término más activo *usuario*, que denota un individuo más involucrado por el proceso que está pasando. Este cambio en particular resulta ser trabajoso, pues este término se encuentra fuertemente acoplado a la ficha original, y requirió el cambio de 43 interfaces distintas.

Por la parte más gráfica, el logotipo de CMYN se presenta en varios lugares de la ficha, que quedan fuera de lugar al utilizarse en Kintun. Para realizar este cambio se crea un logotipo sencillo para Kintun, validado por el equipo, ya que el centro no tenía uno en utilización. Este se puede evidenciar en la figura 3.2

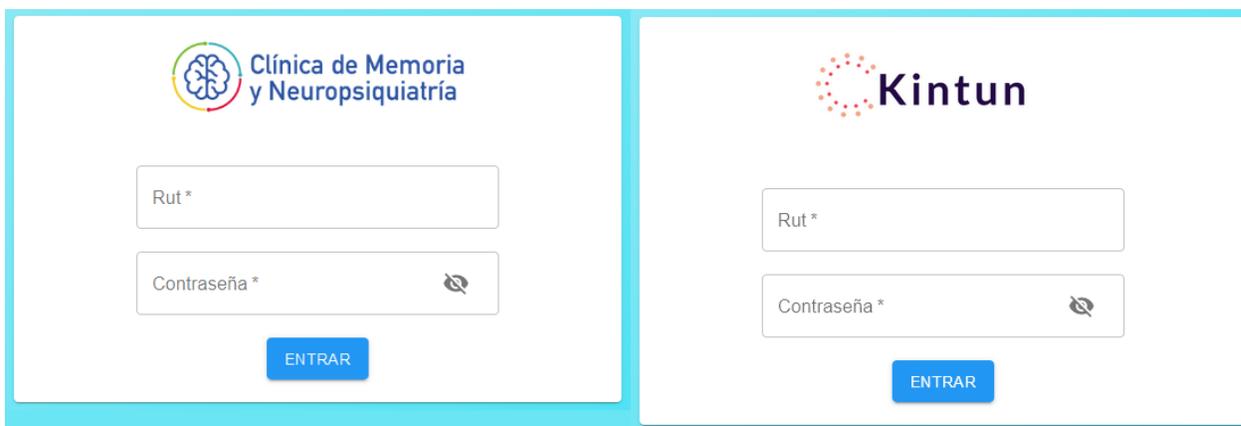


Figura 3.2: Diferencias en los logotipos en la pantalla de inicio de sesión.

3.1.3. Captura de información

En algunos casos, las funcionalidades de la ficha debieron adaptarse levemente para capturar información distinta dependiendo del contexto en el que se encuentre. Para esto, la información que se captura cambia dependiendo del contexto en el que se encuentre. Podemos ver un ejemplo de esto en la figura 3.3, en los cuales la cantidad, nomenclatura y opciones de los campos varían según el ambiente en que se utiliza.

The image shows two screenshots of web forms. The top screenshot is titled 'Nueva Evolución Clínica' and contains the following fields: 'Fecha de Evaluación:' (31/07/2020), 'Responsable' (Luis Ampuero), 'Prestación' (Consulta de psiquiatría a cuidador), 'Aplicado a' (Paciente), 'Evolución' (Conclusiones informe fonoaudiología (*)), 'Conclusiones', and 'Observaciones'. The bottom screenshot is titled 'Nueva Prestación' and contains: 'Fecha de Evaluación:' (11/01/2021), 'Responsable' (Victor Garrido Urzúa), 'Prestación' (Seguimiento), 'Tipo', 'Modalidad' (Paciente, Presencial), and 'Resultados/Observaciones'. Both forms have navigation buttons for 'ANTERIOR' and 'SIGUIENTE'.

Figura 3.3: Registro de una evolución clínica/prestación, con diferencias en los campos registrados en CMYN y Kintun

3.2. Módulo Tailor

Como se ha descrito en las secciones anteriores, existe una clara necesidad de generalizar la ficha digital para que sea efectiva y utilizables en contextos que sean distintos de su contexto original. El trabajo de reestructuración en este sentido tiene que ser efectivo tanto en las interfaces que visualiza el usuario, como en los procesos que manejan la lógica de la ficha.

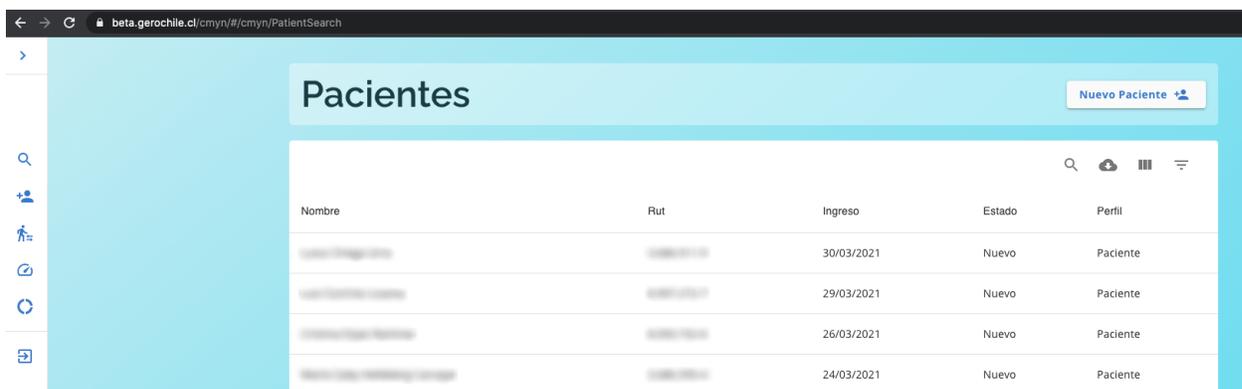
Para responder a estas necesidades es que se propone generar un módulo para frontend y para backend que permita al código ser agnóstico a los específicos de cada contexto en el que se utilice. El objetivo es que se pueda con una sola llamada el código sea capaz de extraer de forma generalizada los específicos de cada contexto. El nombre elegido para este módulo, **Tailor**, hace alusión a un sastre que de un mismo traje, puede ajustarlo a la medida de sus clientes.

El módulo está implementado en Javascript y Python, para ser usado en frontend y backend respectivamente. La forma en la que funciona es que este módulo exporta un objeto, llamado **tailor**, que puede ser usada para obtener objetos generalizados desde otras partes del código. Un ejemplo concreto es el hecho de que la gente atendida en CMYN es llamada

paciente, pero en Kintun es denominada *usuario*. De esta forma, el código que define cierta interfaz es único, pero los resultados son diferentes, como se muestra a continuación en la pantalla de búsqueda.

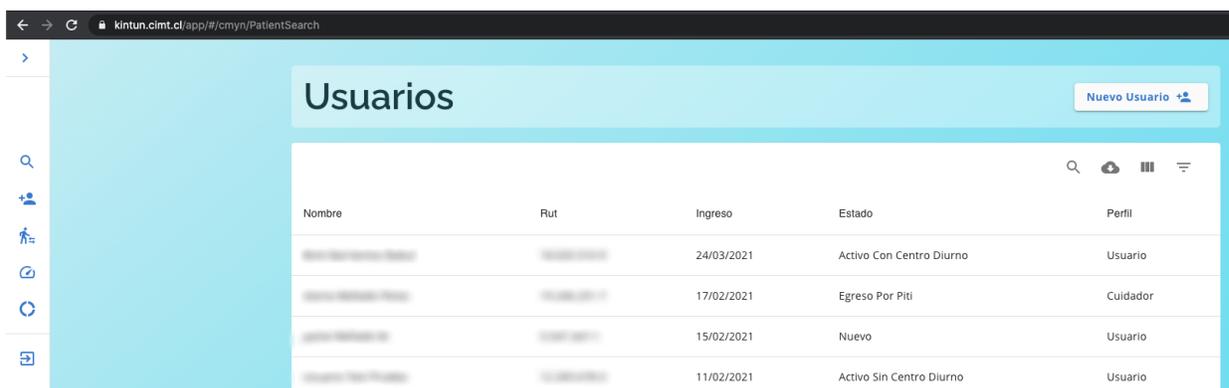
Código 3.1: Ejemplo de un uso del módulo Tailor.

```
1 import tailor, { pluralize } from '../././config/tailor'
2
3 ...
4
5 <Grid item>
6   <Typography variant={'h3'} className={styles['title']}>
7     tailor.patientName
8   </Typography>
9 </Grid>
10
```



Nombre	Rut	Ingreso	Estado	Perfil
[Redacted]	[Redacted]	30/03/2021	Nuevo	Paciente
[Redacted]	[Redacted]	29/03/2021	Nuevo	Paciente
[Redacted]	[Redacted]	26/03/2021	Nuevo	Paciente
[Redacted]	[Redacted]	24/03/2021	Nuevo	Paciente

Figura 3.4: Pantalla de búsqueda de plataforma CMYN, proveniente del Código 3.1.



Nombre	Rut	Ingreso	Estado	Perfil
[Redacted]	[Redacted]	24/03/2021	Activo Con Centro Diurno	Usuario
[Redacted]	[Redacted]	17/02/2021	Egreso Por Piti	Cuidador
[Redacted]	[Redacted]	15/02/2021	Nuevo	Usuario
[Redacted]	[Redacted]	11/02/2021	Activo Sin Centro Diurno	Usuario

Figura 3.5: Pantalla de búsqueda de plataforma Kintun, proveniente del Código 3.1. Notar la diferencia en la nomenclatura entre Paciente y Usuario.

La manera en que se elige cuál contexto utilizar es a partir del valor de una variable de ambiente específica en el servidor donde la plataforma está desplegada. La implementación

específica de los módulos, tanto en Javascript como en Python, se pueden ver en el Anexo A.

El módulo Tailor puede ser importado desde virtualmente cualquier parte del código, por lo que sirve para definir y cambiar detalles tales como el nombre que se muestra en la pestaña, imágenes (al permitir usar otra ruta), y hasta el esquema de colores puede ser redefinido. En cuanto al backend y a la lógica implementada, se pueden definir comportamientos distintos, tal como la redefinición de valores, o incluso llevar el flujo de ejecución por un camino diferente.

3.3. Nuevas funcionalidades

3.3.1. Evaluación Integral

Para la implementación de la evaluación integral, se toma como base el uso de la evaluación médica dentro de la ficha original, pero se reconfigura para conseguir capturar la información requerida por Kintun. Esto se realiza así aprovechando el modelo ya existente en la base de datos del proyecto, en el cual las evaluaciones heredan de la entidad abstracta **evaluation**, como se muestra en la Figura 3.6.

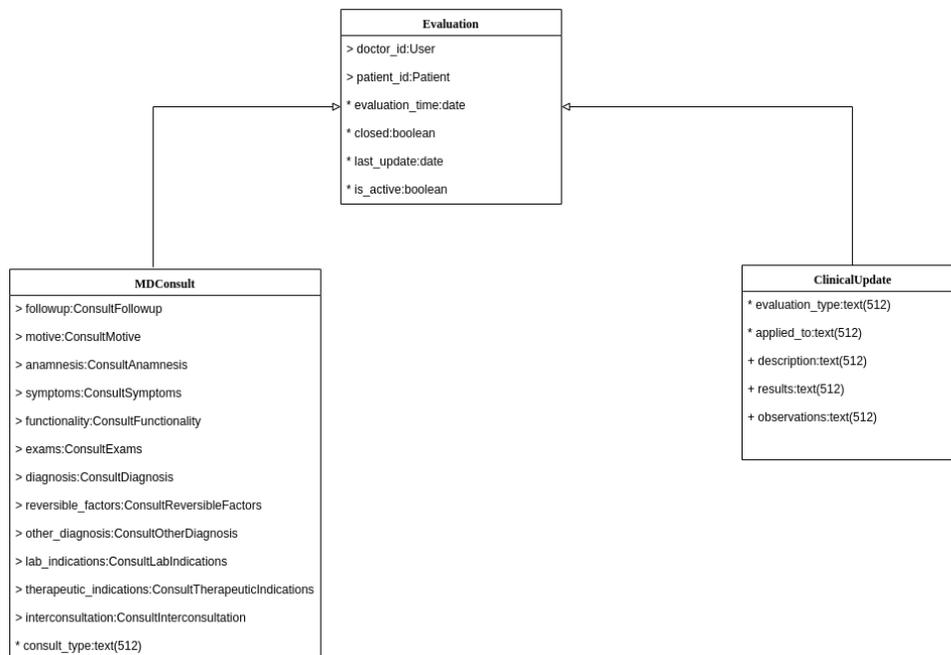


Figura 3.6: Modelo de la entidad abstracta **evaluation**, con dos modelos que heredan de ella.

De esta manera, con agregar otro modelo que hereda de **evaluation**, y realizar los campos respectivos en las interfaces, se agrega la evaluación integral a la ficha, permitiendo aprovechar características ya desarrolladas para esta como el autoguardado.

3.3.2. Reportería

Dentro de la ficha CMYN ya existe un módulo que permite obtener ciertas estadísticas que son de utilidad para Kintun. En particular, se puede sacar estadísticas de la cantidad de usuarios por su estado actual, y el detalle de las prestaciones globales realizadas en cierto intervalo de tiempo. Esto es de utilidad para Kintun, pero se necesita tener más datos para poder automatizar tareas que se realizan periódicamente. El principal foco es tener las prestaciones entregadas a un usuario en particular en cierto intervalo de tiempo, con especial interés en las entregadas en el último mes calendario, y asimismo tener las prestaciones entregadas por cierto profesional en cierto intervalo de tiempo.

Para esto, la solución que se toma es exportar esta información en forma de tablas, utilizando una librería para exportar la data en formato CSV. En particular, se utiliza la librería `react-csv`², que convierte la información enviada por el backend a un archivo que puede ser utilizado por el personal de Kintun.

3.4. Optimizaciones ficha original

3.4.1. REDCap

Como se explicó en la sección 2.2.2, actualmente la integración con REDCap posee un problema de escalamiento de la solución que causa que las consultas hechas a esta plataforma sean muy lenta. Para optimizar estas consultas se planea explorar los endpoints de la API de REDCap para ya sea encontrar un endpoint que funcione mejor que el actual, o configurar el endpoint utilizado de forma de hacer la consulta de forma más eficiente.

La API de REDCap está debidamente documentada dentro del proyecto, lo cual ayuda a encontrar alternativas a la forma actual en la que están ocurriendo las requests. Además de esto, existe un sandbox para probar request mediante API, lo cual es una buena herramienta para encontrar nuevas maneras de pedir la data de forma de hacer el proceso más eficiente.

Para explicar en mayor detalle el problema que se tiene con REDCap, hay que entender el detalle de implementación que hace que las consultas hayan sido tan lentas. Cuando a REDCap se consultaba por todos los registros de un usuario, lo que corresponde a los formularios que han sido aplicados a este usuario, se exportaba en formato JSON un diccionario con todos los campos de todos los instrumentos. Este diccionario contiene los nombres de las variables y sus valores, independiente de si estas se han llenado o no. Considerando que en la ficha CMYN esto corresponde a 3463 campos. En sí no es una consulta muy pesada, pero esto se dificulta al considerar cómo REDCap maneja los datos cuando un instrumento está configurado para ser repetible. En ese caso se vuelve a enviar el mismo diccionario JSON, con todos los nombres de los campos de todos los instrumentos, pero sus valores vacíos excepto para el instrumento llenado. Es decir que con 10 instrumentos llenados, se envían en total 34630 campos, la gran mayoría con valores vacíos. Esta respuesta, que puede tener varios megabytes de tamaño, es muy dispersa, y por tanto puede ser optimizada.

² Generate a CSV file from given data. <https://www.npmjs.com/package/react-csv>

La solución encontrada fue configurar la llamada a la API, de forma que la respuesta no venga en formato **flat** (el antes descrito), sino que en formato **eav**, o *entidad-atributo-valor*, el cual está diseñado para contenido disperso. De esta forma, cada campo sólo está presente si es que este tiene un valor. Esto trae como consecuencia un poco más de *overhead* por cada request, pero al eliminar mucha data vacía, resulta en respuestas mucho más livianas y manejables. Notar se comparan las dos respuestas en los códigos 3.2 y 3.3, el formato flat trae muchos datos vacíos, y si bien el formato eav trae más datos por cada campo, resulta muchísimo más eficiente su uso.

Código 3.2: Ejemplo de respuesta en formato flat. Solo muestra los primeros y últimos campos de los 3463 totales.

```

1 [{"record_id":"1","redcap_repeat_instrument":"tadlq","redcap_repeat_instance":1,"
   ↪ responsable_t_adlq":"","fecha_t_adlq":"","missing_tadlq":"","otromiss_tadlq":"","
   ↪ comer_t_adlq":"","vestirse_t_adlq":"","ba_arse_t_adlq":"","ir_al_ba_o_t_adlq":"","
   ↪ "medicamentos_t_adlq":"","apariencia_t_adlq":"","cocinar_t_adlq":"","mesa_t_adlq
   ↪ ":"","aseo_hogar_t_adlq":"","mant_casa_t_adlq":"","rep_hogar_t_adlq":"","
   ↪ lavado_ropa_t_adlq":"","trabajo_t_adlq":"","
2 ...
3 "evolucion_anamnesis":"","antecedentes_anamnesis":"","parece_el_paciente_concien":"","
   ↪ observaciones_anamnesis":"","podr_a_el_paciente_entrar":"","anamnesis_complete":
   ↪ ""}]

```

Código 3.3: Ejemplo de respuesta en formato eav.

```

1 [{"record":"1","redcap_repeat_instrument":"tadlq","redcap_repeat_instance":1,"field_name":
   ↪ "record_id","value":"1"},{"record":"1","redcap_repeat_instrument":"test_del_reloj","
   ↪ redcap_repeat_instance":1,"field_name":"test_del_reloj_complete","value":"1"},{"
   ↪ record":"1","redcap_repeat_instrument":"tadlq","redcap_repeat_instance":"2","
   ↪ field_name":"record_id","value":"1"}]
2

```

Una de las dificultades que presenta el formato eav es que es requerido parsear la respuesta y asignar cada campo a un objeto que codifica un instrumento para poder mostrarlo en la ficha. Esto se resuelve iterando por la respuesta y construyendo un diccionario con el nombre y el número de instrumento como llave, y una lista de pares de valores como valor.

Con estos cambios, el tiempo de carga de los instrumentos desde la plataforma disminuyó al rango de $500ms$ a $750ms$, lo cual está dentro del rango de aceptación del requisito planteado.

3.4.2. Testing

Una buena forma de integrar tests a un proyecto, tanto por el lado de backend utilizando Python, como por el frontend utilizando Javascript, es mediante librerías que permitan escribir y correr tests de forma sencilla y estandarizada. Los tests se ubican en su propio directorio dentro del proyecto, generando un archivo análogo a cada archivo de código que se quiera probar. De esta manera, se pueden organizar los tests para ser ejecutados de forma automatizada mediante se desarrolla y así asegurar un piso mínimo de calidad de código en

el proyecto.

Durante el transcurso del proyecto se incorporan tests a las nuevas funcionalidades y optimizaciones desarrolladas con el objetivo de que toda nueva funcionalidad esté probada. De similar manera, se intenta agregar test a la mayor cantidad de funcionalidades preexistentes posibles.

Es así cómo se introduce testing unitario al backend y al frontend. En ambos casos, se toma apoyo en los frameworks utilizados para escribir y correr los tests: Django³ y React⁴. Esto tiene el beneficio de otorgar funciones auxiliares para definir los tests de forma más sencilla y legible. En el código 3.4 se puede observar el uso de estas funciones para permitir definir tests, tales como la función `it`, `expect`, o el método `toEqual`.

Código 3.4: Ejemplo de un test para el módulo Tailor.

```
1 import cmynTailor from '../config/tailorConfig/cmyn'
2 import kintunTailor from '../config/tailorConfig/kintun'
3
4 it('All tailors should have the same keys', () => {
5   const cmynKeys = Object.keys(cmynTailor).sort()
6   const kintunKeys = Object.keys(kintunTailor).sort()
7   expect(cmynKeys).toEqual(kintunKeys)
8 })
9
```

Estos tests unitarios son ejecutados desde la terminal, obteniendo una lista de todos los tests que pasan correctamente, y en caso de que alguno falle, cuál era el dato esperado para cumplir el test, y cuál fue el resultado obtenido. Adicionalmente a esto, es posible obtener el porcentaje de cobertura de código utilizando estas mismas herramientas.

Código 3.5: Comandos para correr tests unitarios en Django y en React.

```
1 ./manage.py test
2 npm test -- --coverage
```

Finalmente, utilizando estos comandos, se puede visualizar en la Tabla 3.1 el nivel de cobertura obtenido por los tests. Se distingue en esta tabla la cobertura del código desarrollado en específico para este proyecto de adaptación, del código anterior de la ficha. Notar que el nivel de cobertura del código anterior era de 0% al iniciar este proyecto.

Tabla 3.1: Cobertura obtenida para código nuevo desarrollado para este proyecto, y para código heredado.

Componente	Cobertura código desarrollado	Cobertura código anterior
Frontend	92 %	5 %
Backend	81 %	7 %

³ Writing and running tests in django, <https://docs.djangoproject.com/en/3.1/topics/testing/overview/>

⁴ Running tests, <https://create-react-app.dev/docs/running-tests/>

3.5. Habilitación de ambiente productivo

Para que la ficha digital pudiese ser probada y subsecuentemente utilizada por los integrantes de Kintun, fue necesario pasar por un proceso de puesta en marcha en un ambiente productivo. Un servidor fue habilitado en condiciones muy similares al de CMYN, también en el data center de la Universidad de Chile.

En este servidor se configuraron las dependencias necesarias para levantar tanto el backend como el frontend de la plataforma. Las dependencias necesarias para levantar la ficha se separan en los siguientes ámbitos, todos necesarios ser disponibilizados:

1. Lenguajes Python 3.7 y NodeJS 15.12 para correr el frontend y el backend respectivamente.
2. Postgresql como motor gestor de base de datos, que alberga los datos de la aplicación de Django.
3. Yarn como gestor de paquetes y dependencias del frontend, entre ellas el framework React 16.12 como bloque elemental.
4. Pip como gestor de dependencias del backend, entre ellas el framework Django 3.1 para construir la aplicación web.
5. Apache HTTP Server 2 como servidor web de la aplicación.

Además de la instalación y habilitación de las dependencias, se configuró la base de datos para ser compatible con el proyecto y se definieron las variables de ambiente pertinentes para así poder cargar las configuraciones del módulo Tailor. Adicional a esto, fue necesario la habilitación de un proxy en el servidor de la Universidad de Chile para poder servir el servidor web al internet.

3.6. Creación y habilitación de proyecto en REDCap

Para poder habilitar las funcionalidades de la plataforma REDCap en la ficha digital, fue necesario pedir la habilitación de un proyecto a la administración de REDCap de la universidad. Este proyecto inicialmente fue levantado en blanco, es decir, sin ningún usuario, instrumento configuración adicional. El CMYN accedió a exportar la lista de instrumentos utilizados, los cuales se lograron importar a la plataforma, pero requirieron de correcciones manuales para funcionar adecuadamente, las cuales fueron realizadas para que los instrumentos quedaran completamente funcionales. Queda como responsabilidad de los integrantes del centro Kintun adaptar la definición de estos instrumentos en caso de ser necesario aplicarlos de forma distinta que CMYN.

Finalmente, la optimización de las consultas a REDCap requirió que el proyecto estuviese configurado de una manera en particular para que la API respondiera de forma eficiente a las consultas. De forma más concreta, se configuró que los instrumentos se pudieran repetir en su aplicación al mismo usuario, y se deshabilitó asociar el registro de instrumentos a eventos.

3.7. Cobertura de requerimientos

En esta sección, se detalla cómo se mapean y se cubren los requerimientos planteados en el capítulo 2 con lo descrito respecto a la implementación en este capítulo.

Tabla 3.2: Cobertura de los requerimientos funcionales planteados.

Requerimiento	Cubierto por	Notas
[RF-01] La plataforma debe permitir exportar las atenciones brindadas por un integrante del centro Kintun durante cierto intervalo de tiempo, en particular, durante el último mes.	Módulo de reporería.	
[RF-02] La plataforma debe permitir exportar las atenciones brindadas a un usuario durante cierto intervalo de tiempo, en particular, durante el último mes.	Módulo de reporería.	
[RF-03] La plataforma debe permitir aplicar una evaluación integral a un usuario.	Módulo de evaluación integral.	
[RF-04] La evaluación integral debe contar con autoguardado.	Módulo de evaluación integral.	Esta funcionalidad se obtiene al heredar desde la evaluación médica de la ficha CMYN.
[RF-05] La plataforma debe integrar con REDCap permitiendo la creación y edición de instrumentos.	Módulo de integración con REDCap.	
[RF-06] La integración de REDCap debe permitir llenar un instrumento para un usuario, y visualizarse en la ficha.	Módulo de integración con REDCap.	La visualización fue mejorada al permitir que cargue en un tiempo razonable, como se plantea en el requerimiento [RN-05] .

Tabla 3.3: Cobertura de los requerimientos no funcionales planteados.

Requerimiento	Cubierto por	Notas
[RN-01] La plataforma debe ser accesible desde cualquier dispositivo con conexión a internet.	Despliegue en producción.	
[RN-02] La información dentro de la plataforma debe ser entregada de forma clara y concisa.	General.	Se puede notar su cobertura al mirar los resultados en la sección 4.2
[RN-03] Los tiempos de respuesta al interactuar con la aplicación deben ser cortos. Ninguna interacción debiese hacer esperar al usuario más de 1 segundo de forma de no interrumpir su flujo de trabajo.	General.	Los tiempos de respuesta de la aplicación, para el usuario, varían entre $150ms$ a $700ms$.
[RN-04] Las diferencias que tenga la ficha según el ambiente en el que se utilice deben tener una implementación limpia, que sea fácil de utilizar en el futuro.	General.	Ayuda a cubrir este requerimiento el uso de test y comentarios extensivos dentro del código, documentando el uso del módulo Tailor.
[RN-05] Los instrumentos llenados en REDCap deben visualizarse dentro de la ficha Kintun en un tiempo acotado, menor a 10 segundos.	Módulo de integración con Redcap	El tiempo de respuesta de la visualización no pasa de los $700ms$
[RN-06] Las nuevas funcionalidades implementadas en la plataforma deben tener test unitarios.	Testing.	Ver sección 3.4.2

Capítulo 4

Validación

4.1. Reuniones de validación

Durante el desarrollo de este trabajo, se realizaron reuniones semanales con personal de Kintun para validar el trabajo realizado y así acercar el estado del desarrollo a los objetivos planteados. Las reuniones tienen una duración de aproximadamente una hora, las cuales se desarrollan según el siguiente itinerario:

1. Se presenta brevemente los puntos a tratar en la reunión.
2. Se revisa el estado actual del Kanban, poniendo énfasis en tareas que cambiaron de estado.
3. Se muestran en vivo las nuevas funcionalidades implementadas en la ficha.
4. Se discute sobre la funcionalidad que probó el personal de Kintun durante la semana, y se levanta cualquier cambio que sea imperativo realizar. Por lo general esta parte de la reunión toma la mayor parte del tiempo, así que se guía la discusión para ser lo más concreta posible en pasos a seguir.
5. Se acuerda por una nueva funcionalidad a probar por el personal de Kintun para la semana entrante.

Con estas reuniones, se fue corroborando de forma semi formal y continua que las funcionalidades desarrolladas fueran de valor para Kintun, y que el producto final al que se estaba llegando fuese útil. De la misma manera, al tener input constante de los usuarios, se permitió priorizar y entregar las funcionalidades que se consideraban más importantes primero, disminuyendo la incertidumbre del resultado final. Tomando el principio del Manifiesto por el Desarrollo Ágil de Software: “Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor”[6]

4.2. Encuesta de usabilidad SUS

Para validar la usabilidad de la plataforma entre los usuarios, se aplica la escala SUS (System Usability Scale) [10] [11]. Esta escala permite estimar el valor de usabilidad de la plataforma para los usuarios aplicando un breve cuestionario. En el contexto de este proyecto, se aplicó un cuestionario de forma remota, después de haber pedido a los usuarios

que probaran algunos de los flujos de la plataforma, como la creación de un usuario del centro Kintun e ingresar una nueva prestación para este. El cuestionario pidió a los usuarios que evaluaran de 1 a 5, siendo 1 “Muy en desacuerdo” y 5 “Muy de acuerdo”, su posición frente a los siguientes enunciados:

1. Creo que me gustará visitar con frecuencia esta plataforma
2. Encontré la plataforma innecesariamente compleja
3. Pensé que era fácil utilizar la plataforma
4. Creo que necesitaría del apoyo de un experto para recorrer la plataforma
5. Encontré las diversas funcionalidades de la plataforma bastante bien integradas
6. Pensé que había demasiada inconsistencia en la plataforma
7. Imagino que la mayoría de las personas aprenderían muy rápidamente a utilizar la plataforma
8. Encontré la plataforma muy incómoda al recorrerla
9. Me sentí muy seguro usando la plataforma
10. Necesito aprender muchas cosas antes de manejarme en la plataforma

A partir de esta escala, cada respuesta puede sumar entre 0 y 10 puntos, obteniendo así para cada usuario un puntaje entre 0 y 100 puntos. En la siguiente tabla, se muestra los puntajes obtenidos por los cuatro usuarios de la plataforma que realizaron la encuesta, en conjunto con sus perfiles.

Tabla 4.1: Resumen resultados escala SUS en Kintun

Perfil	Puntaje Final
Directora Kintun	85
Kinesiólogo	90
Terapeuta Ocupacional	95
Personal Administrativo	82.5
Puntaje Final Promedio	88.1

El puntaje final promedio de 88,1 indica que la plataforma posee una muy buena usabilidad, al considerar que el puntaje promedio para las interfaces web es de 68,2 y que al mapear la escala SUS a una escala adjetiva, un puntaje como este es considerado “Excelente”[12]. Sin embargo, no hay que perder de vista que, si bien hay una buena variedad de perfiles en quienes respondieron el cuestionario, el espacio muestral $n = 4$ es bastante pequeño, así que debe ser sólo considerado como una primera aproximación a la usabilidad de la plataforma.

En un análisis más en detalle por pregunta, el cual puede ser observado en el anexo B.1, vemos que algunos enunciados tienen gran concordancia. El enunciado “Creo que me gustará visitar con frecuencia esta plataforma” y “Encontré la plataforma muy incómoda al recorrerla” obtienen el puntaje perfecto. Por otro lado, el enunciado “Necesito aprender

muchas cosas antes de manejarme en la plataforma” tiene más varianza y el puntaje más negativo entre las preguntas. Esto puede significar que la plataforma tiene cierta curva de aprendizaje en su utilización y no es simple llegar y usar.

El puntaje otorgado por el Personal Administrativo es el más bajo de los recibidos. Esto puede estar relacionado a que este usuario tuvo menos tiempo utilizando la plataforma, de manera que comparativamente puede percibir que la plataforma es menos amigables que aquellos usuarios que tuvieron mayor tiempo de utilización. De todas formas, en la sección de comentarios adicionales libres, este usuario detalla que “Es una plataforma fácil de utilizar, muy ordenada.”, por lo que no es muy preocupante que haya otorgado un puntaje menor.

Capítulo 5

Conclusiones

En el presente trabajo se describe cómo se realizó la adaptación de una plataforma web de forma de generalizar su uso como ficha médica digital para distintos contextos. Si bien en este trabajo se pudo llevar tan solo a un segundo ambiente, se realizó el trabajo fundacional que permitirá tomar el mismo software y llevarlo a otros espacios.

Para expandir en este punto, el uso del módulo Tailor ya generaliza varias de las funcionalidades y nomenclaturas de la ficha, de forma que de ser necesario utilizarla en otro centro, simplemente basta configurar los valores para el nuevo contexto en el que se encuentra. Dentro de estas generalizaciones, se encuentran la iconografía utilizada, las URL de redirección, el nombre de múltiples campos y los valores que estos pueden tomar, y la posibilidad de optar usar funcionalidades como la exportación de reportes o aplicación de una Evaluación Integral. Si por otro lado, la funcionalidad que se busca cambiar no está aún generalizada por el módulo, se puede seguir el mismo patrón de diseño que se utilizó en este trabajo para extraer esta nomenclatura o funcionalidad y que se comporte de una manera distinta en otro ámbito.

Se destaca el trabajo realizado en conjunto con el equipo Kintun, que mediante el recurrente contacto se levantó en conjunto las necesidades a desarrollar para este proyecto. Haber tenido la capacidad de semanalmente mostrar, revisar, discutir y probar las nuevas funcionalidades desarrolladas permitió validar y dar aceptación a las necesidades resueltas, como también priorizar las funcionalidades que resultaron ser más importantes para Kintun. La modalidad de trabajo y la metodología escogida para llevar a cabo el desarrollo resultó muy adecuada para este trabajo.

En cuanto al trabajo de optimización a las consultas a REDCap, se destaca el gran impacto que se generó al mejorar los tiempos de respuesta, convirtiendo una parte de la ficha a un componente muchísimo más utilizable y amigable para el usuario. Desde un punto de vista técnico, se destaca la importancia de seleccionar cuidadosamente la estructura de datos correcta, o en este caso, el endpoint y la configuración de la API a la cual se integra la ficha de forma de obtener la estructura de datos más eficiente posible.

A partir de las pruebas realizadas con los usuarios, se puede evidenciar que la plataforma causa una buena sensación de usabilidad en los usuarios. Esto refleja que es prometedora en su calidad de apoyo para las labores realizadas en el centro Kintun. De complementarse esta

solidez en su usabilidad con un funcionamiento robusto en el largo plazo, esta herramienta será de gran utilidad para los integrantes de Kintun.

Dado el trabajo realizado en el levantamiento de las necesidades de Kintun, la refactorización y generalización de la plataforma implementada, y subsecuentemente su despliegue, testing y validación, se pueden considerar cumplidos los objetivos específicos de este trabajo.

Finalmente, desde un punto de vista personal, se destaca la oportunidad de trabajar tan de cerca con usuarios de la plataforma y tener la capacidad de captar sus necesidades en una construcción que les permita mejorar su trabajo y su día a día. Realizar trabajo que es utilizado y agradecido, es una experiencia sumamente gratificante. Espero que este trabajo sea un inicio en mi carrera para diseñar e implementar software que impacte positivamente en la vida de las personas.

Bibliografía

- [1] P. Fuentes and C. Albala, “An update on aging and dementia in Chile,” *Dementia & Neuropsychologia*, vol. 8, pp. 317 – 322, 2014.
- [2] J. Gajardo, J. Aravena, T. Callejas, J. Rodríguez, and D. Tello, “Terapia ocupacional en centro kintun: De la estimulación a la participación,” *Revista ContextO*, vol. 3, pp. 69 – 105, 2016.
- [3] J. Gajardo, J. Aravena, M. Budinich, A. Larraín, P. Fuentes, and L. N. Gitlin, “The kintun program for families with dementia: From novel experiment to national policy (innovative practice),” *Dementia*, vol. 19, no. 2, pp. 488 – 495, 2017.
- [4] I. Leroi, K. Watanabe, N. Hird, and T. Sugihara, ““psychogeritechnology” in japan: Exemplars from a super-aged society,” *International Journal of Geriatric Psychiatry*, vol. 33, no. 12, pp. 1533 – 1540, 2018.
- [5] G. Digital, “Sistema de informatización de la red asistencial: Ficha médica electrónica,” 2016 (recuperado el 9 de Mayo de 2020).
- [6] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, “Manifesto for agile software development,” 2001.
- [7] C. Larman and V. R. Basili, “Iterative and incremental developments. a brief history,” *Computer*, vol. 36, no. 6, pp. 47–56, 2003.
- [8] J. Nielsen, *Usability Engineering*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.
- [9] R. B. Miller, “Response time in man-computer conversational transactions,” in *Managing Requirements Knowledge, International Workshop on*, vol. 1, (Los Alamitos, CA, USA), p. 267, IEEE Computer Society, dec 1968.
- [10] J. Brooke, “*SUS-A quick and dirty usability scale.*” *Usability evaluation in industry*. CRC Press, 1996. ISBN: 9780748404605.
- [11] J. Sauro and J. R. Lewis, *Quantifying the User Experience: Practical Statistics for User Research*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1st ed., 2012.
- [12] A. B. P. K. J. Miller, “Determining what individual sus scores mean: Adding an adjective rating scale,” *Journal of Usability Studies*, vol. 4, no. 3, pp. 114–123, 2009.

Anexo A

Módulo Tailor

A.1. Implementación

Código A.1: Definición del módulo Tailor en Javascript

```
1 import kintunTailor from './tailorConfig/kintun'
2 import cmyntailor from './tailorConfig/cmyntailor'
3
4 /**
5  * The tailor module is designed to provide flexibility to the application,
6  * in order to be used by different centers.
7  * If certain value or feature flag has to be adjusted for a particular center,
8  * it should be defined inside it's tailorConfig file.
9  * As CMYN was the initial center used for this app, it is the default tailor used.
10 */
11
12 /**
13  * This variable defines the skeleton of the tailor object, in order to create
14  * a new tailor and to aid in code autocompletion.
15 */
16 let tailor = {
17   appName: '',
18   appNameVerbose: '',
19   backendURL: '',
20   clinicalEvolutionCollapseToOneField: false,
21   clinicalEvolutionName: '',
22   patientName: '',
23   patientStates: [],
24   useInstrumentsV2: false,
25 }
26
27 /**
28  * A tailor is chosen according to an environment variable
29  */
30 switch (process.env.REACT_APP_TAILOR_CONFIG) {
31   case 'kintun':
32     tailor = kintunTailor;
33     break;
```

```

34 case 'cmyn':
35     tailor = cmynTailor
36     break;
37 default:
38     tailor = cmynTailor
39 }
40
41 export const capitalize = word => word.charAt(0).toUpperCase() + word.slice(1);
42 export const pluralize = word => word + 's';
43
44 export default tailor
45

```

Código A.2: Definición del módulo Tailor en Python

```

1 import os
2
3 get = {
4     'backend_path': '',
5     'site_url': '',
6 }
7
8 cmyn_tailor = {
9     'backend_path': 'cmyn-back',
10    'site_url': 'https://beta.gerochile.cl/cmyn/',
11 }
12
13 kintun_tailor = {
14     'backend_path': 'kintun-back',
15     'site_url': 'https://kintun.cimt.cl/app/',
16 }
17
18 tailor_for = os.getenv('DJANGO_APP_TAILOR_CONFIG', default='cmyn')
19
20 if tailor_for == 'cmyn':
21     get = cmyn_tailor
22 elif tailor_for == 'kintun':
23     get = kintun_tailor
24 else:
25     get = cmyn_tailor
26

```

Anexo B

Escala SUS

B.1. Resultados completos

Tabla B.1: Resultados particulares por enunciado para los cuatro usuarios encuestados, siendo el puntaje 1: Muy en desacuerdo y 5: Muy de acuerdo.

	Usuario			
	1	2	3	4
Enunciado	Puntaje asignado			
Creo que me gustará visitar con frecuencia esta plataforma	5	5	5	5
Encontré la plataforma innecesariamente compleja	1	2	1	2
Pensé que era fácil utilizar la plataforma	5	4	4	2
Creo que necesitaría del apoyo de un experto para recorrer la plataforma	1	3	1	1
Encontré las diversas funcionalidades de la plataforma bastante bien integradas	5	5	5	5
Pensé que había demasiada inconsistencia en la plataforma	1	2	1	1
Imagino que la mayoría de las personas aprenderían muy rápidamente a utilizar la plataforma	5	5	5	4
Encontré la plataforma muy incómoda al recorrerla	1	1	1	1
Me sentí muy seguro usando la plataforma	4	5	5	5
Necesito aprender muchas cosas antes de manejarla en la plataforma	4	2	2	3