



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

INCA TRAIN:  
APLICACIÓN PARA ENTRENAR ANIMALES NO HUMANOS  
Y ESTUDIAR SUS CAPACIDADES DE APRENDIZAJE

MEMORIA PARA OPTAR AL TÍTULO DE  
INGENIERO CIVIL EN COMPUTACIÓN

BRYAN NICOLÁS RIVEROS RIVEROS

PROFESOR GUÍA:  
JÉRÉMY BARBAY

MIEMBROS DE LA COMISIÓN:  
ALEJANDRO HEVIA ANGULO  
SANDRA DE LA FUENTE GONZÁLEZ

SANTIAGO DE CHILE  
2022

# Resumen

La repetición espaciada es una metodología de aprendizaje que ha sido estudiada extensamente en seres humanos, dando lugar a una gran cantidad de aplicaciones (Duolingo, Anki, Memrise, etc.) que aprovechan los resultados de estos estudios para mejorar el proceso de aprendizaje y memorización de una gran cantidad de temas. Sin embargo, este ha sido un tema poco estudiado en animales no humanos y actualmente no existen aplicaciones que permitan acompañar un estudio del mismo. A partir de esta problemática, se diseñó e implementó un primer prototipo de aplicación web, el cual permite crear, utilizar y compartir planes de entrenamiento para animales, con el fin de poder estudiar su aprendizaje y en particular, el uso de la repetición espaciada con ellos. Esta implementación tuvo una recepción mayormente positiva por parte de la Dra. Charlotte Robinson, quien participó como clienta del producto.

*A mi lela.*

# Agradecimientos

Quiero agradecer a todas las personas que me han apoyado en este largo viaje. A mi familia, quienes siempre estuvieron cuando los necesitaba para brindarme ánimo y ayudarme en todo lo que necesité. A mis amigos, los de toda la vida y los que hice en la universidad, hicieron que estos años fuesen mucho más llevaderos y gracias a ellos las risas nunca faltaron.

También agradecer a todos los profesores que tuve, si bien nunca fui el más participativo en clases puedo decir que todos fueron buenos y me entregaron una gran cantidad de conocimiento que espero poder usar el resto de mi vida. A Charlotte Robinson por su tiempo desinteresado, sin ella este trabajo sería totalmente distinto (para peor). Finalmente agradecer al profesor Jérémy, quien fue mi guía durante esta memoria e invirtió gran cantidad de tiempo en mi y en mis compañeros para que tuviésemos la mejor experiencia y un trabajo digno de un ingeniero.

# Tabla de Contenido

<b>1. Introducción</b>	<b>1</b>
<b>2. Estado del arte</b>	<b>3</b>
2.1. Repetición Espaciada . . . . .	3
2.2. Planes de entrenamiento . . . . .	4
2.3. Cuadro comparativo . . . . .	5
<b>3. Diseño de la solución</b>	<b>7</b>
3.1. Requisitos generales . . . . .	7
3.2. Planes de entrenamiento . . . . .	7
3.2.1. Nodos . . . . .	8
3.3. Sesiones de entrenamiento . . . . .	9
3.4. Repetición espaciada y sugerencias . . . . .	10
3.5. Modelo de datos . . . . .	11
<b>4. Implementación</b>	<b>14</b>
4.1. Tecnologías utilizadas . . . . .	14
4.2. Funciones de repetición espaciada . . . . .	14
4.3. Vistas . . . . .	15
4.3.1. Inicio, recomendaciones y navegación . . . . .	15
4.3.2. Aprendices . . . . .	16
4.3.3. Planes . . . . .	17

4.3.4. Sesión de entrenamiento . . . . .	20
4.3.5. Registros . . . . .	22
4.3.6. Configuraciones . . . . .	23
4.4. Almacenamiento . . . . .	23
<b>5. Validación</b>	<b>26</b>
5.1. Metodología de validación . . . . .	26
5.2. Evaluación . . . . .	26
5.3. Análisis . . . . .	28
<b>6. Conclusión y trabajo futuro</b>	<b>29</b>
6.1. Resumen de lo logrado . . . . .	29
6.2. Lecciones aprendidas . . . . .	29
6.3. Trabajo futuro . . . . .	30
<b>Bibliografía</b>	<b>31</b>
<b>Anexos</b>	<b>33</b>
<b>Anexo A. Vistas de la aplicación</b>	<b>33</b>
<b>Anexo B. Creación de planes</b>	<b>43</b>
<b>Anexo C. Evaluación</b>	<b>45</b>
C.1. Preguntas en inglés . . . . .	45
C.2. Resultados de la evaluación . . . . .	46
<b>Anexo D. Tablas</b>	<b>47</b>

# Índice de Tablas

2.1. Tabla comparativa de aplicaciones que utilizan de alguna forma planes de entrenamiento y/o algoritmos de repetición espaciada. Se observa que ninguna de las aplicaciones estudiadas cumplen con todas las características esperadas.	6
5.1. Preguntas utilizadas en la evaluación de la aplicación y respuestas de la Dra. Robinson. Se observa que en su mayoría sus respuestas fueron positivas. . . .	27
D.1. Ejemplo de archivo csv generado por la aplicación. Contiene los registros generados durante la ejecución de una sesión de entrenamiento. En este se ve a la aprendiz Sofi utilizando dos sesiones distintas del mismo plan y en la segunda salió de manera prematura. . . . .	47
D.2. Ejemplo de archivo csv generado por la aplicación. Contiene los registros generados durante la ejecución de una sesión de entrenamiento. En este se ve a la aprendiz Sofi utilizando dos sesiones distintas del mismo plan y en la segunda salió de manera prematura. (Continuación) . . . . .	48

# Índice de Ilustraciones

2.1. Caja de Leitner casera en la que cada fila corresponde a la frecuencia, con los ejercicios más frecuentes en la parte inferior. Imagen provista por Jérémy Barbay. . . . .	4
2.2. Ejemplo de plan de entrenamiento en aplicación Adidas Running de Android. Se observan las sesiones semanales y una dificultad creciente, además de una barra que marca el progreso del usuario. . . . .	5
3.1. Ejemplo de plan en Duolingo, con el cual se pretende aprender alemán. Los círculos representan sesiones de entrenamiento. A color se encuentran las que están disponibles para usar y en gris las que aún no se han desbloqueado. . .	8
3.2. Diagrama de flujo de una sesión de entrenamiento. . . . .	10
3.3. Modelo de datos relacional. Se observa que el diseño gira en torno a los aprendices, aunque los usuarios siguen siendo importantes. . . . .	11
4.1. Pantalla de inicio de InCA Train, con recomendaciones de planes a utilizar calculados con una función de repetición espaciada. Se ven dos planes: uno que ha sido utilizado previamente y otro que no. . . . .	16
4.2. Visualización de los detalles de una aprendiz, con sus campos de nombre, edad, especie y genero. Además, botones para ir a sus planes, editarla, eliminarla y crear un nuevo aprendiz. . . . .	17
4.3. Formulario de creación de un aprendiz, con todos los campos requeridos por el sistema. . . . .	17
4.4. Vista de los planes. Contiene un listado de estos, correspondientes al aprendiz seleccionado, junto a una barra de búsqueda y un filtro de ordenamiento. . .	18
4.5. Detalles de un plan. Contiene información sobre su creador y contenido, además de información de uso por parte del usuario . . . . .	19
4.6. Visualización de un plan. Se pueden ver los nodos que redirigen a sesiones de entrenamiento, y es posible observar sus tres posibles estados: Session 1 y 2 se encuentran completados, Session 3 disponible y Session 4 no disponible. . . .	20

4.7.	Ejemplo de sesión de entrenamiento. En la parte superior se muestra el nombre del plan, seguido por las instrucciones y un botón que indica que hay un video disponible. Más abajo se ve un mensaje con la cantidad de repeticiones a completar y un contador con forma de estrella. Finalmente, los botones de control en orden descendente de izquierda a derecha: Failed, Clicker, So-So, Undo, Back, Exit, Next/Finish. . . . .	21
4.8.	Vista de los registros. Muy similar a la vista de los planes (Figura 4.3.3), contiene un listado de estos y un filtro para buscarlos por su nombre. . . . .	22
A.1.	Pantalla de inicio de InCA Train cuando no se tiene ningún aprendiz. Muestra un acceso directo a la creación de aprendices y no existe el ícono en la zona superior derecha. . . . .	33
A.2.	Pantalla de inicio de InCA Train cuando el aprendiz seleccionado no tiene planes de entrenamiento. Muestra un acceso directo a la vista de planes. . . . .	34
A.3.	Pantalla de inicio de InCA Train cuando se han completado las recomendaciones para el día actual y el aprendiz seleccionado, o bien, no existen recomendaciones para el día. . . . .	35
A.4.	Menú de navegación de la aplicación, que aparece al presionar el botón de la zona superior izquierda. Permite dirigirse a cualquier vista de la misma. . . . .	36
A.5.	Menú de selección de aprendices. En lo más alto y coloreado en gris se muestra el aprendiz actualmente seleccionado, seguido por el resto de aprendices creados por el usuario. . . . .	37
A.6.	Vista al presionar Exit en una sesión de entrenamiento y luego de aceptar el primer mensaje de advertencia. Se muestra un campo para escribir las razones de la salida prematura y la posibilidad de penalizar (disminuir) el nivel actual del plan. . . . .	38
A.7.	Mensaje mostrado al finalizar la sesión, en el cual se ve la fecha propuesta para la siguiente sesión, calculada con los algoritmos de repetición espaciada. . . . .	39
A.8.	Vista de las configuraciones de la aplicación. Cada campo contiene un icono de ayuda, el cual muestra un mensaje con una descripción de lo que hace cada variable. . . . .	40
A.9.	Registros con lectura facilitada para los seres humanos. Posee botón de descarga, el cual genera un archivo csv con la información. . . . .	41
A.10.	Mensaje mostrado al tratar de subir un nuevo plan. Se ven dos botones para descargar un template (Apéndice B.1), descargar un plan de ejemplo (Apéndice B.2) y un campo para subir el plan. . . . .	42

# Capítulo 1

## Introducción

“Animal-Computer Interaction” (ACI) [23] es un campo de investigación para el diseño y uso de tecnología con, para y por animales no humanos (denominados como “animales” en el resto del trabajo). Ésta fue iniciada por investigadores de “Human-Computer Interaction” (HCI) al rededor del año 2011 con la publicación del “ACI Manifesto”, y desde el año 2016 cuenta con su propia conferencia (International Conference on Animal-Computer Interaction) [3]. En la actualidad se presenta como un área en crecimiento y con un enfoque multidisciplinario.

En el conjunto de personas que realizan investigación en ACI, es posible encontrar a profesionales de diversas áreas, tales como la ingeniería, psicología, biología, criadores y entrenadores de animales, todos trabajando en conjunto para buscar la manera de mejorar la vida, la forma en que nos comunicamos e incrementar el conocimiento que tenemos de los animales. Una de las áreas relevantes de ACI es el trabajo relacionado al aprendizaje de los animales, ya sea estudiando como estos aprenden o tratando de comprobar si es posible que estos aprendan a hacer ciertas cosas que para los humanos resultan naturales.

Entre las técnicas que normalmente se utilizan para el aprendizaje en seres humanos se encuentra la repetición espaciada [26]. Ésta es una metodología que consiste en hacer revisiones a elementos estudiados con anterioridad en intervalos de tiempo cada vez más largos, para así dar tiempo a que el cerebro procese lo que ha aprendido y posteriormente fortalecer estos enlaces mediante el esfuerzo de tener que recordar y aplicar este conocimiento.

Esta técnica o variantes de la misma han logrado buenos resultados en la retención del aprendizaje en seres humanos [5] y ha permitido el desarrollo de aplicaciones tales como Duolingo o Anki, que utilizan diferentes implementaciones de esta metodología en sus planes de enseñanza. Sin embargo, Duolingo no se puede utilizar con animales (pues no está diseñada para ellos) y Anki está limitada en el aspecto del diseño y creación de planes de entrenamiento, y en la recolección de datos de uso, lo que dificulta utilizarla como herramienta de acompañamiento en el estudio del aprendizaje animal.

En el caso de animales distintos a los humanos, la repetición espaciada no se ha estudiado en profundidad, en parte por lo costoso que es llevar a cabo este tipo de investigación y por otra que no es sencillo idear un plan que se adecue a las fortalezas y debilidades de

cada animal. En el aspecto de los costos, han surgido en el último tiempo conceptos como “crowdsourcing” y “citizen science”, en donde se espera que sea la ciudadanía quienes formen parte de las investigaciones científicas, ya sea aportando fondos, ideas o datos. Es el caso de los datos que resulta más relevante en esta situación, dado que en el punto de validar una hipótesis es necesario recopilar la mayor cantidad de información posible.

Dada la problemática expuesta, el objetivo general de este trabajo consiste en desarrollar un prototipo de aplicación que permita a investigadores del aprendizaje de animales crear sus propios planes de entrenamiento, aplicarlos en el tiempo siguiendo principios de repetición espaciada y generar registros de los resultados obtenidos al hacer uso de los mismos con distintos tipos de animales no humanos. Del mismo modo, los objetivos específicos corresponden a formalizar el formato de una secuencia de entrenamiento y los modos de implementar repetición espaciada, diseñar y construir el primer prototipo de una aplicación que permita crear y poner uso planes de entrenamiento, además de recolectar datos de estos y finalmente, validar la usabilidad de la aplicación creada con investigadores de ACI y entrenadores o guardianes de animales no humanos, potencialmente sin formación científica.

El trabajo de esta memoria consistió en el diseño e implementación de un primer prototipo de aplicación web, que funcionase como acompañamiento en el entrenamiento de un animal, permitiendo crear planes de entrenamiento, hacer uso de estos y recopilar información de las acciones tanto del animal como del entrenador.

Este documento comienza describiendo la situación actual en cuanto a ACI, la repetición espaciada y las aplicaciones disponibles en el mercado que podrían dar solución a la problemática expuesta (Capítulo 2). Luego se describe la solución diseñada (Capítulo 3) y la implementación de una aplicación con este diseño (Capítulo 4). A continuación se describe el proceso de validación de la aplicación y su evaluación por parte de una experta en el área de ACI (Capítulo 5) y finaliza con un resumen de lo logrado, junto a un análisis de lo aprendido y una descripción del camino que debería seguir este proyecto en el futuro (Capítulo 6).

# Capítulo 2

## Estado del arte

El área de ACI es relativamente nueva, y no se cuenta con gran cantidad de software dedicado a solucionar problemáticas particulares. Aún así, es posible tomar inspiración de técnicas y aplicaciones dedicadas a humanos, o para animales que no cumplen con todo lo que se desearía tener y al mismo tiempo ver que es lo que se les puede agregar o modificar para lograr una aplicación que pueda ser de utilidad para el estudio de la cognición animal.

En este capítulo, se describen varios puntos relacionados relevantes a este trabajo, desde algoritmos de repetición espaciada y aplicaciones que los implementan (Sección 2.1), el uso y visualización de planes de entrenamiento (Sección 2.2) y una comparación entre aplicaciones que usan lo visto en los puntos anteriores (Sección 2.3).

### 2.1. Repetición Espaciada

La repetición espaciada es una técnica que consiste en hacer revisiones a sesiones o elementos estudiados en el pasado en distintos intervalos de tiempo (normalmente incrementales), dando tiempo al cerebro de procesar lo estudiado, para luego fortalecer estas ideas mediante el esfuerzo de recordarlas y aplicarlas. Esta metodología ha sido estudiada desde hace varias décadas, lo que ha permitido generar diferentes algoritmos a partir de sus principios:

- **Sistema Leitner [25]:** Es un sistema que consiste en el uso de cartas y cajas, cada caja representa un nivel, el cuál a su vez representa el intervalo de tiempo hasta la próxima repetición. El responder correctamente a la carta hará que esta suba de nivel bajando su frecuencia y de lo contrario vuelve a una frecuencia más alta bajando su nivel.
- **Algoritmos SuperMemo [21]:** En el algoritmo más simple, cada elemento a estudiar está asociado a una dificultad, la que es un factor numérico que se va modificando según se responda correcta o incorrectamente. Este factor es multiplicado al valor que representa la cantidad de días a esperar para la próxima repetición, el que luego es reemplazado por el valor obtenido.

- **Intervalos uniformes:** Consiste en repetir una acción o actividad cada cierta cantidad de días, sin que este intervalo cambie.



Figura 2.1: Caja de Leitner casera en la que cada fila corresponde a la frecuencia, con los ejercicios más frecuentes en la parte inferior. Imagen provista por Jérémy Barbay.

A partir de estos y otros algoritmos se han creado diferentes aplicaciones que permiten a seres humanos estudiar sobre distintos temas, entre las cuales podemos encontrar:

- **Anki [13]:** es una aplicación que busca mejorar la memorización mediante el uso de cartas o flashcards. Este permite crear y compartir mazos de cartas de diferentes temas.
- **Duolingo [7]:** es una aplicación orientada al aprendizaje de lenguajes mediante el uso de distintas actividades como unir palabras, completar oraciones, etc. Con cursos de trabajo otorgados por la misma aplicación.
- **Memrise [11]:** es una aplicación orientada al aprendizaje de lenguajes y vocabulario mediante el uso de flashcards. Existen cursos tanto de parte de la aplicación como de otros usuarios.

## 2.2. Planes de entrenamiento

En general, las aplicaciones de aprendizaje como las mencionadas anteriormente proveen sus propios “planes de entrenamiento o aprendizaje”, aunque en algunas no es fácil identificar

el progreso que se lleva en este, ya que simplemente se va agregando material mientras otro se va dejando cada vez más atrás.

En este sentido Duolingo [7] (Figura 3.1) y Khan Academy [2] presenta una buena representación del avance del usuario, mostrando el camino a recorrer y marcando con colores o ticks las actividades completadas. Otro ejemplo en el que podemos ver este manejo de los planes es en las aplicaciones o sitios orientados a los deportes (por ejemplo: Adidas Running [17] y C25K [1]), en las que se suelen proporcionar planes establecidos y muestran las condiciones requeridas para avanzar en el mismo.



Figura 2.2: Ejemplo de plan de entrenamiento en aplicación Adidas Running de Android. Se observan las sesiones semanales y una dificultad creciente, además de una barra que marca el progreso del usuario.

## 2.3. Cuadro comparativo

A continuación se comparan algunas de las aplicaciones mencionadas en la sección anterior, dadas ciertas características que se consideraban necesarias y que dieron origen a este

proyecto.

	Anki	Duolingo	Memrise	Puppr	Adidas Running
Permite crear planes de entrenamiento	Si, orientado a flashcards	No	Si, orientada a idiomas y flashcards	No	No
Permite seleccionar la técnica de repetición espaciada a utilizar	No	No	No	No	No
Genera información de una sesión de entrenamiento	Si	Si	Si	No	Si
Es posible visualizar la información o es solo de uso interno	Ambas	Solo uso interno con visualización de ciertos "logros"	Uso interno, visualización de estadísticas solo en versión pro	Uso interno, para ver que planes se están realizando o completado.	Uso interno, visualización de progreso
Es posible utilizar o compartir la información fuera de la aplicación	Si, es posible recuperar la base de datos desde el almacenamiento del smartphone	No	No	No	No
Es posible recopilar información del uso de otros usuarios de los planes creados	No	No	No	No	No

Tabla 2.1: Tabla comparativa de aplicaciones que utilizan de alguna forma planes de entrenamiento y/o algoritmos de repetición espaciada. Se observa que ninguna de las aplicaciones estudiadas cumplen con todas las características esperadas.

Del cuadro comparativo (Tabla 2.1) es posible observar que no existe una aplicación que cumpla con todos los puntos expuestos, en particular tres puntos importantes como lo son el crear y compartir planes de entrenamiento, el poder utilizar distintas técnicas de repetición espaciada y finalmente, poder recopilar información de su uso de manera simple para permitir realizar investigaciones. Sin embargo, es posible tomar algunas ideas presentes en estas aplicaciones (tanto de funcionalidades como de usabilidad) como inspiración para lograr los objetivos planteados.

# Capítulo 3

## Diseño de la solución

En este capítulo se presenta la solución ideada para el problema expuesto anteriormente. Aquí se presenta el diseño de un software que permite recolectar información de la aplicación de metodologías de repetición espaciada en el uso de planes de entrenamiento de animales no humanos. Se describen los requisitos del software (Sección 3.1), el diseño que se espera de los planes de entrenamiento (Sección 3.2), se expone el desarrollo general de una sesión de entrenamiento (Sección 3.3), el mecanismo para la implementación y uso de las metodologías de repetición espaciada para la generación de sugerencias de planes (Sección 3.4) y una estructura de bases de datos para el sistema (Sección 3.5)

### 3.1. Requisitos generales

El software a implementar debe permitir la creación de planes de entrenamiento, los que consisten de un conjunto de sesiones de entrenamiento. Al hacer uso de estos planes, se debe recopilar la información de lo que suceda durante la sesión. Además, se deben generar sugerencias de planes a llevar a cabo en un día determinado utilizando algún algoritmo de repetición espaciada. Finalmente, se necesita poder crear perfiles para uno o más animales (desde ahora en adelante también llamados “aprendices”), donde se tenga sus características básicas como nombre, especie, edad y género.

### 3.2. Planes de entrenamiento

Los planes de entrenamiento de animales tanto humanos como no humanos, consisten en un listado ordenado de sesiones de entrenamiento, las cuales deben ir en un orden progresivo de dificultad hasta lograr el aprendizaje de la habilidad deseada. Estas sesiones se deben disponibilizar a medida que se cumplan ciertas condiciones, por ejemplo, si un humano quiere entrenar y correr 10km, no puede (o no debiese) hacerlo antes de lograr correr al menos 5km, ya que arriesga lesiones o perder rápidamente la motivación, similar a lo que sucede cuando se entrena a un animal.

Esta noción de plan se juntó con lo que se puede ver en la aplicación Duolingo (Figura 3.1), en la cual es posible visualizarlos como un grafo dirigido donde cada nodo consiste en una sesión de entrenamiento (Sección 3.3) o práctica, las cuales al ser completadas dan acceso a nuevas sesiones.

Entonces, un plan de entrenamiento se definió para esta aplicación como una colección de nodos, que en su conjunto forman un grafo ordenado en orden creciente de dificultad, el cual al ser completado debiese cumplir con el aprendizaje de una habilidad por parte del aprendiz. Además, contiene datos extra que pueden servir a investigadores y usuarios, tales como un título, quien lo creó y una breve descripción de los objetivos del mismo.



Figura 3.1: Ejemplo de plan en Duolingo, con el cual se pretende aprender alemán. Los círculos representan sesiones de entrenamiento. A color se encuentran las que están disponibles para usar y en gris las que aún no se han desbloqueado.

### 3.2.1. Nodos

Los nodos son la unidad básica del plan, cada uno de estos contiene información relevante tanto para su visualización como para el funcionamiento del plan. Se decidió que estos puedan estar en tres estados distintos, donde cada uno difiere tanto en funcionalidad como visualización:

1. **No disponible:** indica que no se han completado los requisitos para acceder a la sesión

(no se han completado los nodos anteriores).

2. **Disponible:** Es posible acceder a la sesión de entrenamiento.
3. **Completado:** Se ha completado la sesión correspondiente.

Además, al ser el plan un grafo dirigido, cada nodo tiene un listado de nodos a los que dirigirse una vez completada la sesión. Con esto es posible hacer que un nodo dependa de dos o más nodos distintos, o que al completar la sesión éste habilite dos o más nodos sin problema.

Finalmente, cada nodo debe redirigir a una sesión de entrenamiento, lo que permite ejecutar y avanzar en el plan al completar cada sesión. También se decidió que era una buena idea que los nodos puedan redirigir a otros planes, con lo que se podría poner un plan como pre-condición para hacer uso del resto del plan principal, sin embargo, para este prototipo esta idea no suma mucho valor, por lo que se dejó de lado por el momento.

### 3.3. Sesiones de entrenamiento

En la investigación realizada sobre lo que ocurre durante una sesión de entrenamiento, se logró identificar que éstas suelen estar compuestas de uno o más “bloques”, los que a su vez contienen una o más instrucciones a seguir. Estas instrucciones pueden o no repetirse antes de permitir avanzar al siguiente bloque, y esto se repite hasta completar todos los bloques.

Es también durante la ejecución de la sesión, el momento ideal de realizar la recopilación de información del uso de la aplicación. Durante las reuniones que se tuvieron con la Dra. Charlotte Robinson [22] (persona con experiencia trabajando en ACI y entrenando animales) quien participó como “cliente” durante las iteraciones del producto, se llegó a la conclusión de que durante la sesión de entrenamiento debía existir una forma de indicar el desempeño del aprendiz, ya sea si cumplió completamente las instrucciones, de manera parcial o si falló. Además, indicó que sería buena idea tener una forma de salir de la sesión de manera prematura para los casos en que el animal deje de participar por algún motivo. Todos estos elementos fueron incluidos en el diseño en forma de botones en la interfaz de entrenamiento.

También es durante estas sesiones el momento ideal para registrar información y se decidió que las siguientes acciones generaran una entrada en el registro: iniciar la sesión, los indicadores de desempeño del aprendiz (cada botón una entrada distinta), la salida prematura (junto a un mensaje que indique que es lo que sucedió) y finalizar la sesión. Además, se optó por incluir un botón para deshacer la última acción en caso de un error por parte del usuario, sin embargo, este no borra el registro generado, ya que puede ser útil para saber que acciones se suelen hacer por error y mejorar la interfaz de manera acorde.

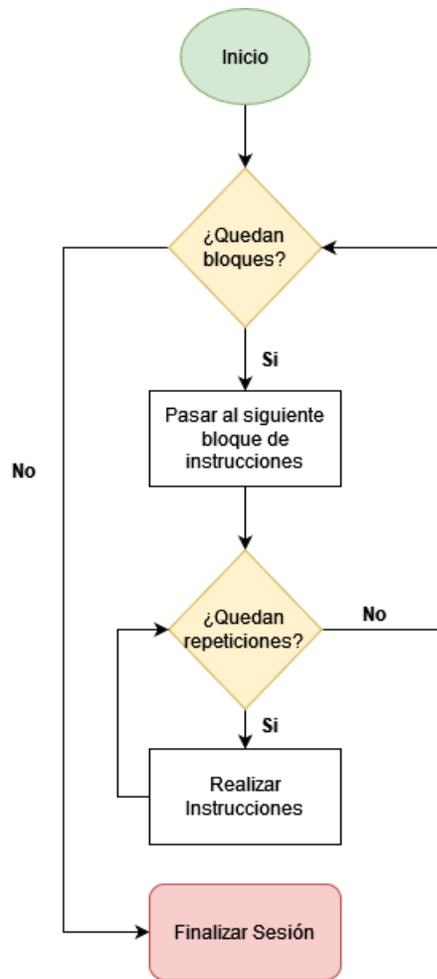


Figura 3.2: Diagrama de flujo de una sesión de entrenamiento.

### 3.4. Repetición espaciada y sugerencias

La repetición espaciada incluye una gran variedad de algoritmos, los que requieren distintos tipos de variables para funcionar. Para esta primera versión se optó por solo incorporar dos de estos, uno basado en el **sistema Leitner** y otro en los **intervalos uniformes**. Para el primer caso, es necesario incluir un campo “nivel” en los planes, mientras que para la segunda función no se necesita nada extra.

En el caso del sistema Leitner y de los niveles, se decidió dejar por defecto que estos comiencen a incrementar una vez se ha completado el plan de entrenamientos, dado que normalmente se suele entrenar más seguido cuando recién se esta aprendiendo, hasta que se logra cierto nivel de conocimiento. Sin embargo, como el fin de esta aplicación es estudiar modelos de aprendizaje, se dejó la opción de que estos incrementen cada vez que se completa una sesión sin importar que no sea la última.

El uso de estos algoritmos no es directamente por parte del usuario, si no que son utilizados en la generación de sugerencias. Para esto es posible usar alguna de las funciones descritas

previamente, junto a la fecha de la última ejecución de algún plan de entrenamiento (y a su nivel en caso de utilizar el sistema Leitner). Así se generan recomendaciones para el usuario de manera programática y utilizando su propia información de uso.

Finalmente, debe existir la opción de poder seleccionar la función de repetición espaciada por el usuario y poder editar de cierta forma los parámetros utilizados por ellas, para así flexibilizar la aplicación y aumentar su valor como herramienta de investigación.

### 3.5. Modelo de datos

A continuación se enseña y describe el modelo de datos diseñado para la aplicación en un formato SQL para hacer más sencillo su entendimiento. Sin embargo, al no existir al momento de entregar este trabajo un servidor centralizado para la aplicación, este modelo queda libre para implementarse de otras formas, por ejemplo, con sistemas NoSQL, similar al que se usa en la implementación de esta aplicación (Sección 4.4).

El detalle más importante a notar, es que en realidad los usuarios principales de la aplicación son los aprendices, puesto que toda la información (planes y registros) dependen de ellos. Además, este diseño permite que un aprendiz pertenezca a uno o más usuarios distintos, con lo que se cubre el caso en que hayan más personas participando de su entrenamiento.

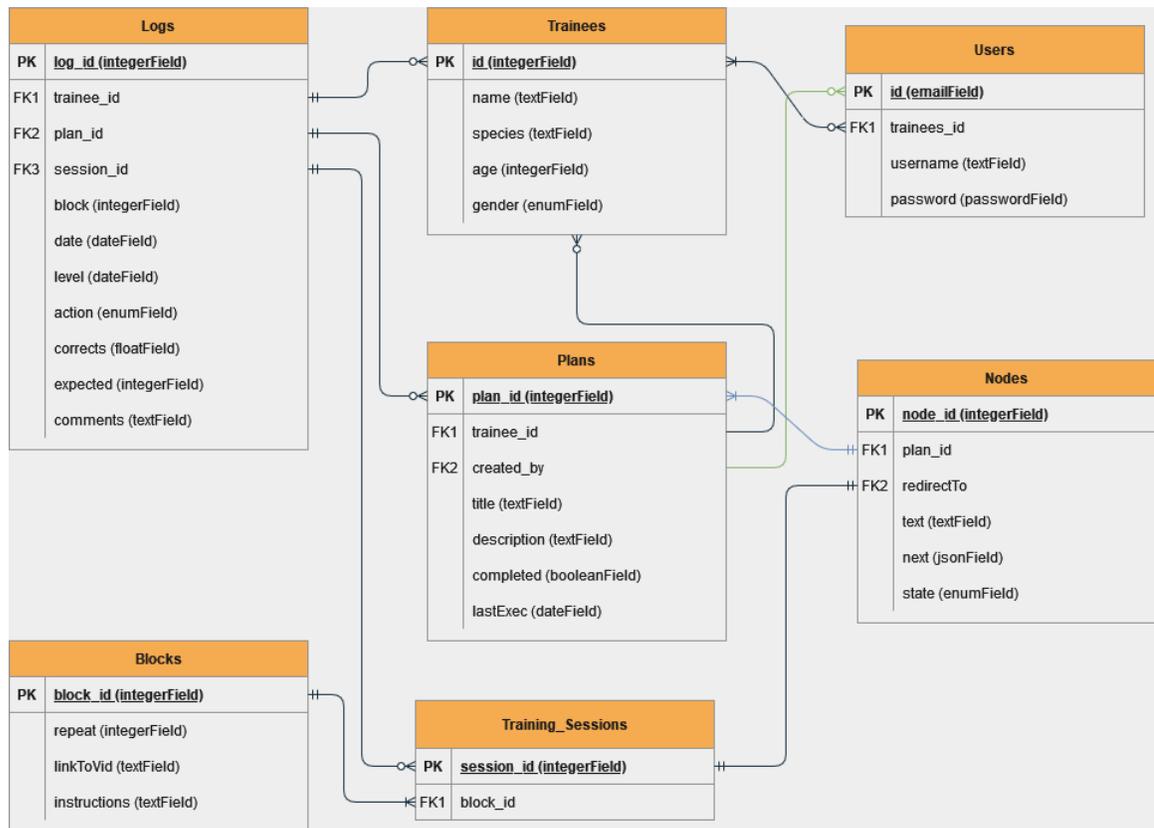


Figura 3.3: Modelo de datos relacional. Se observa que el diseño gira en torno a los aprendices, aunque los usuarios siguen siendo importantes.

El modelo de datos diseñado (Figura 3.3) contiene las siguientes tablas:

- Tabla **Trainees**: Tabla que contiene a los aprendices. Puede pertenecer a uno o más entrenadores.
  - *id*: Identificador del aprendiz. Llave primaria.
  - *name*: Nombre del aprendiz.
  - *species*: Especie del aprendiz.
  - *age*: Edad del aprendiz.
  - *gender*: Género del aprendiz.
- Tabla **Users**: Tabla que contiene a los usuarios de la aplicación. Pueden tener cero o más aprendices.
  - *id*: Email del usuario. Llave primaria.
  - *trainee\_id*: Id de los aprendices. Llave foránea.
  - *username*: Nombre de usuario.
  - *password*: Contraseña del usuario.
- Tabla **Plans**: Tabla que contiene los planes. Los planes tienen referencias a sus creadores (*users*) y a quien lo está utilizando (*trainees*), esto último dado que los planes contienen información con respecto a su uso.
  - *plan\_id*: Identificador del plan. Llave primaria.
  - *trainee\_id*: Aprendiz al que pertenece. Llave Foránea.
  - *created\_by*: Quién creó el plan. Llave Foránea.
  - *title*: Título del plan.
  - *description*: Descripción del plan.
  - *completed*: Estado del plan (completado o no completado).
  - *lastExec*: Fecha de la última ejecución.
- Tabla **Nodes**: Tabla que contiene los nodos de un plan.
  - *node\_id*: Identificador del nodo. Llave primaria.
  - *plan\_id*: Plan al que pertenece. Llave foránea.
  - *redirectTo*: A que lugar debe redireccionar (por ahora solo sesiones de entrenamiento). Llave foránea.
  - *text*: Texto que muestra el nodo al visualizar el plan.
  - *next*: Listado de *node\_id*. Corresponde a los nodos de los que es pre-requisito.
  - *state*: Estado del nodo. Puede estar no disponible, disponible o completado.
- Tabla **Blocks**: Tabla que contiene los bloques de instrucciones de las sesiones de entrenamiento.

- *block\_id*: Identificador del bloque. Llave primaria.
- *repeat*: Número de veces que se deben repetir las instrucciones del bloque.
- *linkToVid*: Enlace al video (si es que existe alguno).
- *instructions*: Instrucciones a seguir.
- Tabla **Training\_Sessions**: Tabla que contiene las sesiones de entrenamiento. Estas corresponden a un conjunto de *blocks*.
  - *session\_id*: Identificador de la sesión. Llave primaria.
  - *block\_id*: Bloques pertenecientes a la sesión. Llave foránea.
- Tabla **Logs**: Tabla que contiene los registros. Estos dependen de los *trainees*, *plans* y *training\_sessions*
  - *log\_id*: Identificador del registro. Llave primaria.
  - *trainee\_id*: Aprendiz al que pertenece. Llave foránea.
  - *plan\_id*: Plan al que pertenece. Llave foránea.
  - *session\_id*: Sesión a la que pertenece.
  - *block*: El número del bloque en el que se realizó la acción. Corresponde a su posición dentro de la sesión de entrenamiento.
  - *date*: Fecha del registro.
  - *level*: Nivel del plan al momento del registro.
  - *action*: Acción que realizó el registro (Sección 3.3).
  - *corrects*: Número de repeticiones correctas al momento del registro.
  - *expected*: Número de repeticiones correctas que se deben completar.
  - *comments*: Comentarios del registro (utilizado para dar razones de una salida prematura).

# Capítulo 4

## Implementación

La implementación del diseño mencionado en el Capítulo 3 consistió en una aplicación web. Este capítulo comienza con una descripción de las tecnologías utilizadas (Sección 4.1), luego se da paso a la implementación de las funciones de repetición espaciada (Sección 4.2) y continúa con las vistas creadas para el funcionamiento de la aplicación (Sección 4.3), tales como: la pantalla de inicio, la visualización de planes, sesiones de entrenamiento, entre otras. Finalmente, se incluye la implementación de un sistema de almacenamiento utilizando la API de “LocalStorage” (Sección 4.4).

### 4.1. Tecnologías utilizadas

Una de las principales ideas de la implementación de esta aplicación era que estuviese disponible en la mayor cantidad de dispositivos posibles. Para lograrlo, se decidió crear una “aplicación web progresiva” (PWA) utilizando el “framework” “Vue.js” [28] en su tercera versión (Vue3). Esto permitió que estuviese disponible para computadoras y celulares utilizando el mismo código. Además, se utilizó la librería “Quasar” [15] para facilitar el diseño del “frontend” y la librería “Mermaid.js” [12], en particular un paquete diseñado para Vue3 (vue3-mermaid [8]) para dibujar los grafos de los planes (Sección 4.3.3).

El almacenamiento de la aplicación se hizo utilizando la API “LocalStorage” y el despliegue de la aplicación se realizó en un servidor del DCC con la siguiente dirección web al momento de la entrega de este documento: <https://buho.dcc.uchile.cl/~inca-train/>.

### 4.2. Funciones de repetición espaciada

Como se mencionó en la Sección 3.4, para este prototipo se decidió implementar solo dos técnicas de repetición espaciada. La implementación de ambas es muy similar y corresponde a las siguientes funciones:

- **Sistema Leitner:**  $f(plan) = nivel\ del\ plan * constante\ de\ intervalo$
- **Intervalos constantes:**  $f(plan) = constante\ de\ intervalo$

Donde la constante de intervalo es un valor que indica el tiempo mínimo que debiese de haber entre sesiones en el caso de que el plan sea de nivel 1. Éste valor es posible modificarlo en las configuraciones (Sección 4.3.6) y por defecto tiene un valor de 1, es decir, para el caso del sistema Leitner, para un plan con niveles 1, 2 o 3 se tendría un intervalo creciente del mismo valor en días, mientras que si esta constante fuese 2, los intervalos cambiarían a 2, 4, 6 para los mismos niveles.

Luego, estas funciones son utilizadas para calcular las recomendaciones que se muestran en la pantalla de inicio de la aplicación (Sección 4.3.1). Un plan es recomendado si cumple con la siguiente condición:

$$f(plan) \geq fecha\ actual - última\ ejecución\ plan$$

Es decir, un plan es recomendado al usuario si el intervalo calculado por las funciones de repetición espaciada es mayor o igual a la diferencia entre los días transcurridos entre la última ejecución del plan y el día actual. Igualmente, un plan puede ser utilizado sin ser recomendado yendo a la vista de planes (Sección 3.2), aunque esto requiere que el usuario lo busque de manera manual.

## 4.3. Vistas

A continuación, se muestran las vistas creadas para la aplicación.

### 4.3.1. Inicio, recomendaciones y navegación

La pantalla de inicio contiene un mensaje de bienvenida junto a un listado de las recomendaciones generadas a partir de la función de repetición espaciada seleccionada a partir de los planes ejecutados por el aprendiz, junto a los planes que se han añadido pero no han sido utilizados. Además, contiene mensajes y accesos directos en caso de no tener aprendices (Apéndice A.1), en caso de que el aprendiz no tenga planes de entrenamiento (Apéndice A.2) y si ha utilizado todas las recomendaciones o no tiene ninguna para el día actual (Apéndice A.3).

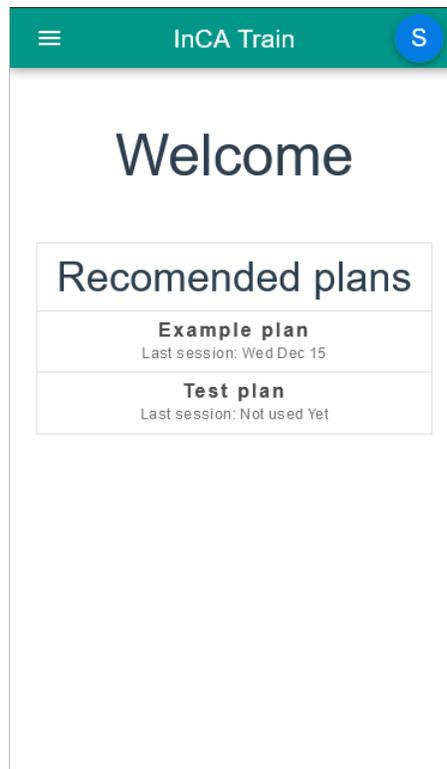


Figura 4.1: Pantalla de inicio de InCA Train, con recomendaciones de planes a utilizar calculados con una función de repetición espaciada. Se ven dos planes: uno que ha sido utilizado previamente y otro que no.

Por otra parte, en la zona superior izquierda contiene un menú de navegación, el que permite dirigirse a las otras vistas de la aplicación (Apéndice A.4) y en la zona superior derecha posee un icono con una funcionalidad similar a la que se puede ver en otras aplicaciones (por ej: Gmail), muestra el aprendiz actual y permite cambiar rápidamente a otro (Apéndice A.5).

### 4.3.2. Aprendices

Los aprendices como se mencionó en la parte de diseño, son los animales que el usuario espera entrenar. En esta vista es posible visualizar los detalles de un aprendiz (Figura 4.2) además de editarlo y eliminarlo, lo que elimina todo lo relacionado a él (planes, sesiones y registros). En caso de no tener un aprendiz o querer crear uno nuevo, cuenta con un botón que redirige a un formulario de creación (Figura 4.3). Finalmente, se cuenta con un acceso rápido a los planes del aprendiz seleccionado.

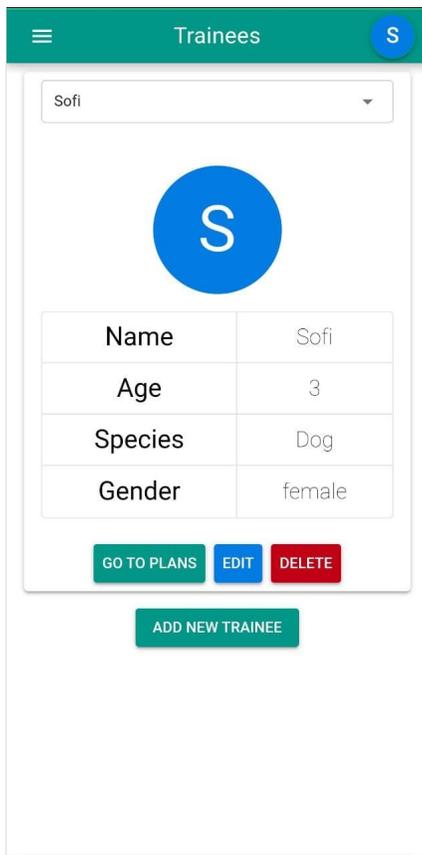


Figura 4.2: Visualización de los detalles de una aprendiz, con sus campos de nombre, edad, especie y genero. Además, botones para ir a sus planes, editarla, eliminarla y crear un nuevo aprendiz.

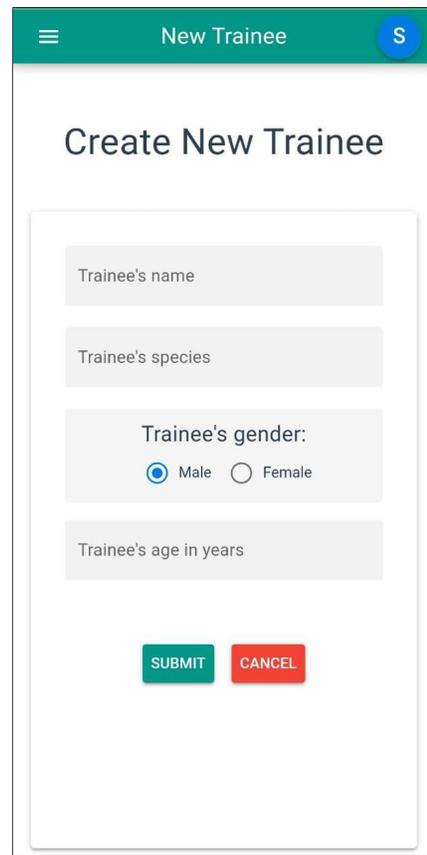


Figura 4.3: Formulario de creación de un aprendiz, con todos los campos requeridos por el sistema.

### 4.3.3. Planes

En la vista de planes se muestra el listado de planes pertenecientes al aprendiz seleccionado. También cuenta con una barra de búsqueda según su nombre y un filtro de ordenamiento, el cual posee cuatro posibles opciones: ejecución más reciente, ejecución menos reciente, nivel descendente y nivel ascendente, además de la opción por defecto, la cual muestra los planes por orden de inclusión.

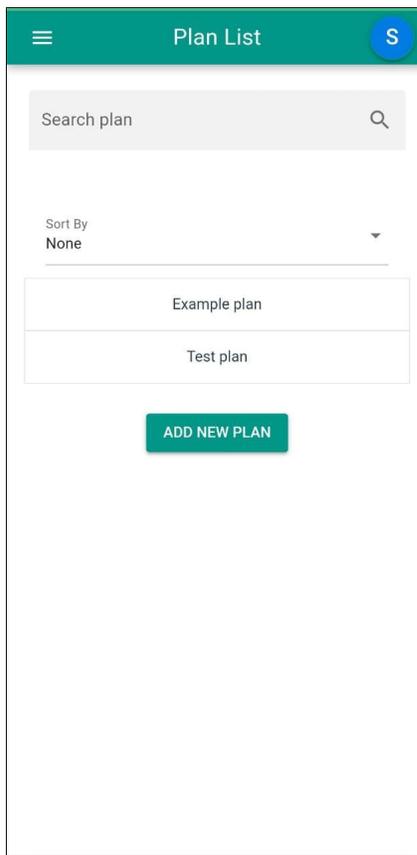


Figura 4.4: Vista de los planes. Contiene un listado de estos, correspondientes al aprendizaje seleccionado, junto a una barra de búsqueda y un filtro de ordenamiento.

Esta vista también cuenta con la opción de subir planes (Apéndice A.10). Dado que en este prototipo no se incluyó una vista para crear planes directamente, se da la posibilidad de subir un plan en formato “JSON”, junto a la descarga de un “template” (Apéndice B.1) y un ejemplo de como éste debería ser rellenado (Apéndice B.2).

## Detalles del plan

Al hacer clic en alguno de los planes de la lista o en la vista de inicio, se accede a los detalles del mismo, en el cual es posible ver su nombre, quien lo creó, una breve descripción, su nivel actual, la fecha de la última ejecución y la fecha esperada para la próxima ejecución generada con los algoritmos de repetición espaciada. Además, es posible dirigirse a la visualización del grafo generado para el plan y también es posible eliminarlo, lo que a su vez elimina los registros y las sesiones de entrenamiento.

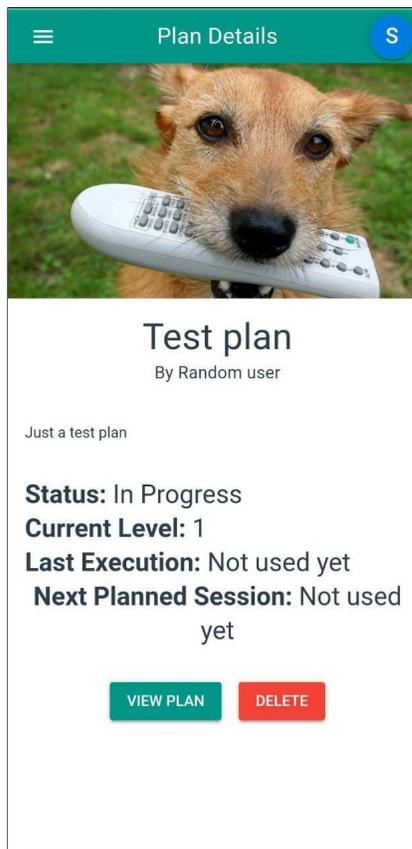


Figura 4.5: Detalles de un plan. Contiene información sobre su creador y contenido, además de información de uso por parte del usuario

### Visualización del plan

Aquí es posible visualizar el grafo del plan junto a su avance. Este grafo es generado con la librería Mermaid.js [12] lo que permitió realizarlo rápidamente aunque con el costo de tener que incluir una estructura específica para los nodos. Los colores de los nodos indican lo siguiente:

- **Verde:** Sesión completada.
- **Amarillo:** Sesión disponible.
- **Gris:** Sesión no disponible.

Al hacer clic en un nodo, se redirige al usuario a la vista de su sesión de entrenamiento correspondiente. Si el nodo presionado era de color amarillo y la sesión fue completada con éxito, éste cambia de color a verde y abre paso al siguiente nodo (en caso de que sea el único nodo requerido o que fuese el último requerimiento necesario) cambiando el color de este último a amarillo.

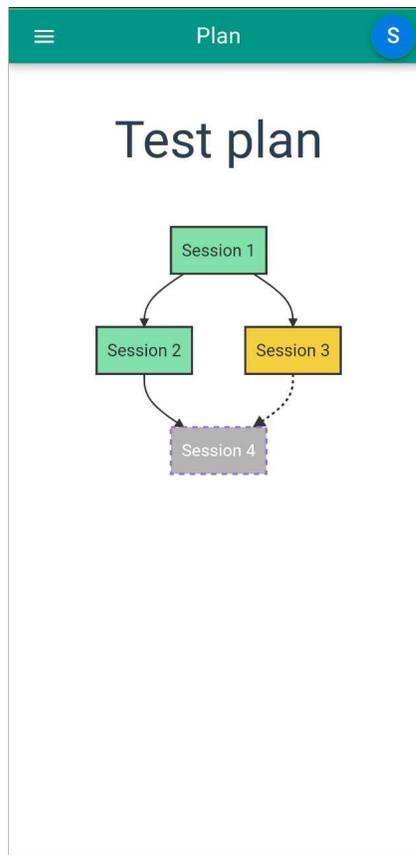


Figura 4.6: Visualización de un plan. Se pueden ver los nodos que redirigen a sesiones de entrenamiento, y es posible observar sus tres posibles estados: Session 1 y 2 se encuentran completados, Session 3 disponible y Session 4 no disponible.

#### 4.3.4. Sesión de entrenamiento

La sesión de entrenamiento es el punto principal de la aplicación, en esta vista es posible ver las instrucciones a seguir del bloque actual, un mensaje con la cantidad de repeticiones completadas y las que se espera que se realicen, junto a los siguientes botones:

1. **Clicker:** Este botón indica que el aprendiz realizó correctamente las instrucciones señaladas. Tiene la forma y emite un sonido similar al de un clicker real (herramienta muy utilizada en el entrenamiento de mascotas). Además incrementa el contador de repeticiones en 1.
2. **Failed:** Indica que el aprendiz falló al realizar las instrucciones. Emite un sonido de vibración para informar que fue presionado, pero es más leve para no incomodar al aprendiz. Este botón no incrementa el contador de repeticiones.
3. **So-So:** Indica una mediana completación de las instrucciones. Este botón es el más complejo pues requiere de un alto entendimiento por parte del usuario de las acciones del aprendiz. Incrementa el contador de repeticiones en 0.5.

4. **Undo:** Deshace los cambios del último botón presionado, disminuyendo el contador en caso de ser necesario.
5. **Exit:** Proporciona una salida prematura a la sesión. Es posible incluir un mensaje de porque se salió y es posible disminuir el nivel del plan a modo de penalización (Apéndice A.6)
6. **Back y Next:** Botones para avanzar y retroceder entre bloques de instrucciones. El botón “Next” es habilitado una vez se completa el numero de repeticiones necesarias para el bloque actual. En caso de ser el último bloque, “Next” se transforma en “Finish”.
7. **Video:** Como se indicó en el diseño del modelo de datos (Sección 3.5), existe un campo opcional de enlace a un video. De existir, se muestra este botón, el cual abre un reproductor web. En caso contrario, este botón queda oculto.

Todos los botones mencionados anteriormente con excepción de “Video”, “Back” y “Next”; pero incluyendo “Finish”, generan un registro.

Una vez completada la sesión, se muestra un mensaje en el cual aparece la fecha calculada con la función de repetición espaciada para la siguiente sesión (Apéndice A.7).

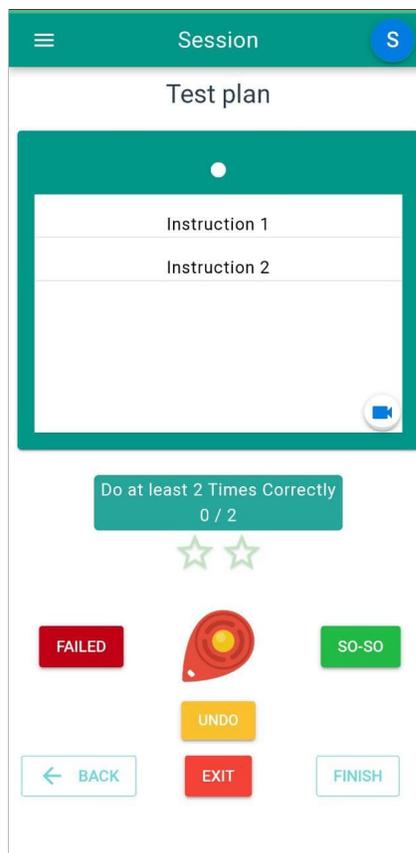


Figura 4.7: Ejemplo de sesión de entrenamiento. En la parte superior se muestra el nombre del plan, seguido por las instrucciones y un botón que indica que hay un video disponible. Más abajo se ve un mensaje con la cantidad de repeticiones a completar y un contador con forma de estrella. Finalmente, los botones de control en orden descendente de izquierda a derecha: Failed, Clicker, So-So, Undo, Back, Exit, Next/Finish.

### 4.3.5. Registros

La vista de registros es similar a la de planes (Sección 4.3.3), dado que es posible ver un listado de estos junto a un filtro por nombre. Al presionar en alguno de los planes es posible visualizar el registro en un formato amigable para el ser humano (Apéndice A.9), el cual contiene más palabras y conectores similar a como sería una oración.

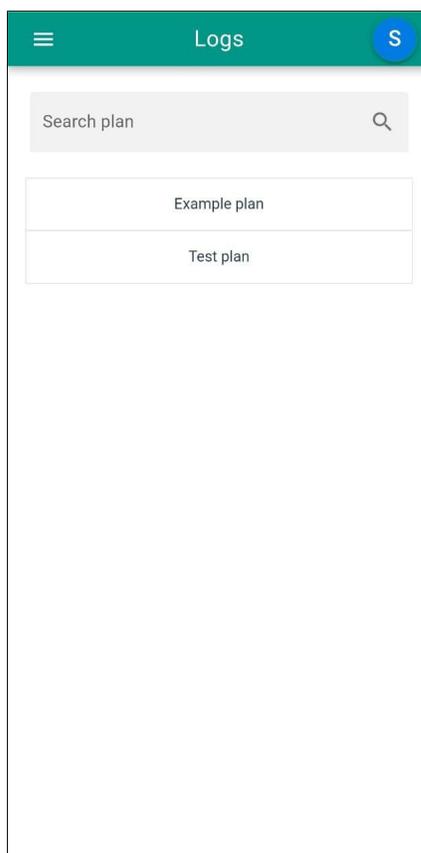


Figura 4.8: Vista de los registros. Muy similar a la vista de los planes (Figura 4.3.3), contiene un listado de estos y un filtro para buscarlos por su nombre.

La aplicación permite descargar el registro en formato “CSV”, para facilitar su procesamiento por parte de investigadores y usuarios que estén interesados. Estos registros contienen los siguientes campos:

- **date:** La fecha en que se realizó el registro.
- **traineeName:** El nombre del aprendiz.
- **planId:** El id del Plan.
- **level:** El nivel del Plan al momento de registrar.
- **sessionId:** El id de la sesión de entrenamiento.
- **block:** El numero del bloque de instrucciones.

- **action:** La acción que gatilló el registro. Puede ser el inicio de una sesión o alguno de los botones de la misma.
- **corrects:** La cantidad de repeticiones completadas correctamente.
- **expected:** La cantidad de repeticiones esperadas para el bloque.
- **comments:** Posibles comentarios de la acción. Por ahora solo para la salida prematura de la sesión.

Es posible ver un ejemplo en los Apéndices D.1-D.2

### 4.3.6. Configuraciones

En esta vista es posible editar las configuraciones por defecto de la aplicación, al lado de cada una se aprecia un icono de información, el cual al ser presionado muestra una pequeña descripción de lo que implica el campo (Apéndice A.8).

Entre las opciones a modificar se encuentran:

- **Spaced Repetition Function:** La función de repetición espaciada a utilizar. Por defecto está seleccionada la función incremental (sistema Leitner).
- **Repetition Interval:** Es el intervalo entre sesiones, cuyo valor es utilizado por las funciones de repetición espaciada para calcular la próxima sesión.
- **Max Plan Level:** El nivel máximo que puede alcanzar un plan.
- **Default Exit Penalty:** El valor por defecto utilizado al hacer una salida prematura de una sesión de entrenamiento cuando se desea disminuir el nivel del plan.
- **Increase Plan Level After:** Indica si se desea incrementar el nivel del plan tras cada sesión o solo una vez se ha completado el mismo.

Estas configuraciones están pensadas principalmente para investigadores, dado que entrega un mayor control sobre la aplicación, pero, también se debe entender que es lo que conlleva hacer el cambio de alguna de estas.

Finalmente, se encuentra un botón para “resetear” la aplicación, lo que implica el borrado de toda la información del usuario, tales como: los aprendices, los planes, los registros y las configuraciones.

## 4.4. Almacenamiento

En un comienzo se esperaba implementar una base de datos en un servidor, pero como solo es el primer prototipo y aún necesita otras funcionalidades se optó por dejar esta opción

para las próximas versiones de la aplicación. Otro factor que influyó en este cambio fue el deseo de implementar un “backend” común entre otras aplicaciones desarrolladas en InCA Labs, por lo que se decidió que no valdría la pena invertir tiempo en algo que lo más probable es que tuviese una vida muy corta.

Es por lo anterior que el almacenamiento de la aplicación consistió en el uso de la API HTML5 llamada “LocalStorage”. Ésta permite guardar información en un “hashmap”, pero cada cosa guardada debe ser un “String”, por lo que es necesario un proceso de conversión entre objetos de “Javascript” y “Strings” cada vez que se desea leer o guardar información. LocalStorage tiene un máximo establecido de 5MB (aunque puede aumentar entre navegadores), lo que si bien no es mucho, para el alcance esperado de este prototipo es más que suficiente.

Dada esta implementación, se optó por hacer un diseño de almacenamiento más simple, en el cual se guardan los planes junto a sus nodos en un lugar y las sesiones de entrenamiento en otro, al igual que los detalles de los aprendices y los registros (Listing 4.1). Cada cosa esta separada a su vez por el id del aprendiz al que pertenecen, lo que si es similar al diseño realizado en la Sección 3.5.

Listing 4.1: Estructuras de datos utilizadas en la aplicación en como almacenamiento junto a la API LocalStorage.

```
1 trainee = {
2     id: Number,
3     name: String,
4     age: Number,
5     species: String,
6     gender: String
7 }
8
9 node = {
10    id: String,
11    text: String,
12    next: [String],
13    style: String,
14    link: String,
15    redirectTo: Number
16 }
17
18 plan = {
19    id: Number,
20    title: String,
21    completed: Boolean,
22    level: Number,
23    lastExec: Date,
24    nodes: [node]
25 }
26
27 plans = {
```

```

28     trainee_id: {
29         plan_id: plan
30     }
31 }
32
33 block = {
34     repeat: Number,
35     linkToVid: String,
36     instructions: [String]
37 }
38
39 trainingSessions = {
40     trainee_id: {
41         plan_id: {
42             session_id: [block]
43         }
44     }
45 }
46 log = {
47     date: Date,
48     traineeName: String,
49     planId: Number,
50     level: Number,
51     session: Number,
52     action: String,
53     corrects: Number,
54     expected: Number,
55     comments: String,
56     block: Number
57 }
58 logs = {
59     trainee_id: {
60         plan_id: [log]
61     }
62 }
63 settings = {
64     repInterval: Number,
65     exitPenalty: Number,
66     maxPlanLevel: Number,
67     spacedRepetitionFunction: {value: Number, label: String},
68     levelAfter: {value: Number, label: String}
69 }

```

# Capítulo 5

## Validación

En este capítulo se comenta la metodología utilizada para validar la aplicación durante el semestre de trabajo (Sección 5.1), luego se describen y muestran los resultados de la evaluación final (Sección 5.2) y se concluye con un pequeño análisis de los mismos (Sección 5.3).

### 5.1. Metodología de validación

La validación se hizo durante el semestre de trabajo mediante reuniones mensuales con la Dra. Robinson, en las cuales, una semana antes de la reunión se le daba acceso a la aplicación con lo que se dejaba que ella explorara las funcionalidades de la misma sin darle pistas de lo que se había implementado hasta el momento. De esta forma, se esperaba que ella sin ninguna guía las pudiera encontrar y examinar, dando así una opinión más personal en cuanto a estas características, la usabilidad y su experiencia en general como usuaria.

Estas reuniones influyeron tanto en el diseño de la aplicación como en su implementación, dado que con su experiencia como entrenadora e investigadora, se logró aterrizar ideas desarrolladas durante la propuesta de trabajo de título e incluir características que ella consideraba necesarias para una aplicación que pretende estudiar el aprendizaje animal (Sección 3.3).

### 5.2. Evaluación

Durante la última reunión se realizó una evaluación general de la aplicación mediante una entrevista. Ésta consistía de una serie de preguntas relacionadas tanto a las funcionalidades provistas por la aplicación, como la usabilidad y utilidad que percibía sobre el posible uso de esta. Las posibles respuestas corresponden a una “escala de Likert” [27] y debían ser un número entre 1 y 5, cuyos valores representaban: (1) Totalmente en desacuerdo, (2) En desacuerdo, (3) Ni de acuerdo ni en desacuerdo, (4) De acuerdo, (5) Totalmente de acuerdo.

Las preguntas (Apéndice C.1) y los resultados (Apéndices C.2) se realizaron en inglés, pero a continuación se muestran en español (y completados con sus comentarios) para facilitar su entendimiento.

N°	Pregunta	Respuesta
1	¿La aplicación fue fácil de usar?	4
2	¿La aplicación fue simple y fácil de usar?	3.5
3	Me gustaría usar esta aplicación como usuario general (entrenador).	2.5
4	Me gustaría utilizar la aplicación para fines de investigación.	5
5	Los registros serán útiles para algunos fines de investigación.	5
6	Me gustaría utilizar la aplicación para fines de investigación (incluso si no tiene un registro).	3.5
7	La aplicación tiene el potencial de enriquecer la vida de los animales.	5
8	La aplicación permite la creación de planes.	5
9	La aplicación permite el uso de planes y sesiones de entrenamiento.	5
10	La aplicación provee recomendaciones de planes a usar en un día.	5
11	La aplicación permite ver registros.	5
12	La aplicación permite exportar registros	5

Tabla 5.1: Preguntas utilizadas en la evaluación de la aplicación y respuestas de la Dra. Robinson. Se observa que en su mayoría sus respuestas fueron positivas.

Las preguntas con valores intermedios, como la segunda y tercera, se deben a que ella no estaba del todo segura de su respuesta, y al ser la única persona que participaría de la evaluación, se le permitió optar por esos valores. Además, en el caso de la tercera pregunta, complementó diciendo que ella probablemente no la usaría para entrenar dado que ya tiene su propia metodología para hacerlo y la aplicación no se adaptaba a esta. Sin embargo, remarcó que otros usuarios sí podrían encontrarla útil.

Finalmente, se incluyó la pregunta abierta: **¿Qué piensas que la aplicación necesita o qué te gustaría ver en ella?** A la cual respondió con lo siguiente:

1. Poder mapear configuraciones a sujetos y planes individualmente sería importante para hacer investigaciones.
2. Validar la interfaz del clicker digital (botones Clicker/So-So/Failed) para potencialmente mejorarla.
3. Un Back end para recopilar toda la información.

Estas ideas fueron tomadas en consideración y se analizó su posible complejidad/costo en el Capítulo 6

### **5.3. Análisis**

En general, se puede observar una buena evaluación de la aplicación por parte de la Dra. Robinson, encontró que puede ser útil en posibles investigaciones con animales y que podría ayudar a enriquecer la relación humano-animal. Además, se validó que es posible utilizar las principales funcionalidades pensadas e implementadas en este prototipo y finalmente, se logró recopilar requisitos para una próxima versión de esta aplicación.

# Capítulo 6

## Conclusión y trabajo futuro

Para finalizar este informe, se realiza un resumen de lo logrado (Sección 6.1), se discute sobre lo que se podría haber hecho mejor (Sección 6.2) y se termina haciendo un análisis de los siguientes pasos que debería seguir esta aplicación (Sección 6.3).

### 6.1. Resumen de lo logrado

Se logró crear un primer prototipo de la aplicación InCA Train, la cual permite a investigadores, criadores y entrenadores, crear, usar y compartir planes de entrenamiento para cualquier tipo de animal no humano. Mediante su uso es posible generar y descargar registros de las acciones realizadas durante las sesiones de entrenamiento. También es posible configurar (aunque de manera limitada) el uso de la metodología de repetición espaciada, mediante la selección de funciones de la misma y edición de ciertos parámetros. Todo esto habilita a la aplicación para ser utilizada en el estudio del aprendizaje de animales no humanos.

Se ha provisto de un diseño para la base de datos de un futuro backend para guardar los datos generados. Por otra parte, el código generado para este prototipo, es de código abierto y se encuentra disponible en gitlab (<https://gitlab.com/incalab/inca-train>). Para concluir, se ha obtenido y compartido conocimiento sobre las tecnologías y metodologías utilizadas, lo que se espera pueda ayudar a otros estudiantes interesados en el área de ACI.

### 6.2. Lecciones aprendidas

En general, la principal lección aprendida fue el manejo de las expectativas respecto a las funcionalidades que se deseaban implementar. Se tuvo que priorizar y dejar de lado ciertas cosas que se habían propuesto previamente para este prototipo, como lo era un backend y la posibilidad de crear planes directamente dentro de la aplicación.

También, en cuanto a la selección de tecnologías, se aprendió a no ligarse tan rápido a

algunas de estas, dado que aunque en un principio se veían muy útiles y simples, trajeron más de algún problema. Como en el caso de la librería Quasar para el frontend, que en muchos casos los ejemplos provistos en su página [15] no funcionaban del todo y el paquete vue3-mermaid [8], el que si bien resultó de gran utilidad, también era bastante limitado, lo que dificultó algunas tareas.

### 6.3. Trabajo futuro

Aunque el prototipo cumplió con la implementación de la mayoría de las características que se esperaban, aún quedan algunas cosas que hacer antes de que esta aplicación se pueda utilizar de manera masiva. Entre las principales cosas que se necesitan se encuentran la implementación de un backend que permita el manejo de usuarios y de los registros generados. Otra cosa a realizar es habilitar la creación y edición de planes dentro de la misma aplicación, facilitando este proceso y permitiendo así junto al backend, la búsqueda de planes que otras personas hayan creado.

Además, se hace necesario revisar las vistas creadas, tanto la interfaz de usuario como su usabilidad, puesto que en este prototipo se priorizó la funcionalidad en la mayoría de los casos y se evaluó con una usuaria con experiencia investigando, entrenando y testeando aplicaciones. Por esto, se hace necesario validar InCA Train con usuarios con menos experiencia. Una de las ideas que se tenía relacionado a este ámbito, era tener una interfaz de entrenamiento separada para personas experimentadas o investigadores y usuarios que solo quieren entrenar a sus mascotas o que lo hacen por primera vez.

Para terminar, hay características a mejorar dentro de la aplicación, como lo son agregar más funciones de repetición espaciada o permitir crear nuevas funciones a los usuarios. Y por supuesto, incluir lo mencionado por la dra. Robinson en la entrevista, relacionado a permitir que cada plan cuente con su propia configuración. Estas dos ideas pueden incrementar de gran manera el valor de esta aplicación como herramienta de acompañamiento en investigaciones.

# Bibliografía

- [1] 5 Km, Couch to. [http://www.c25k.com/c25k\\_spanish.htm](http://www.c25k.com/c25k_spanish.htm), [Online; visitada 14-Julio-2021].
- [2] Academy, Khan. <https://es.khanacademy.org/>, [Online; visitada 14-Julio-2021].
- [3] ACIconf. <https://www.aciconf.org/>, [Online; visitada 07-Julio-2021].
- [4] Anderson, Teoti: *The ultimate guide to Dog TRAINING: Puppy training to advanced techniques plus 25 problem Behaviors solved!* Baker & Taylor, 2014.
- [5] Ausubel, David P. y Mohamed Youssef: *The Effect of Spaced Repetition on Meaningful Retention*. The Journal of General Psychology, 73(1):147–150, 1965. <https://doi.org/10.1080/00221309.1965.9711263>, PMID: 14316956.
- [6] Cohn, Jeffrey P: *Citizen science: Can volunteers do real research?* BioScience, 58(3):192–197, 2008.
- [7] Duolingo, 2011. <https://www.duolingo.com>, [Online; visitada 01-Mayo-2021].
- [8] Inc, Sango Technologies. <https://npm.io/package/vue3-mermaid>, [Online; visitada 29-Diciembre-2021].
- [9] Kaminski, Juliane, Josep Call y Julia Fischer: *Word learning in a domestic dog: evidence for "fast mapping"*. Science, 304(5677):1682–1683, 2004.
- [10] Mancini, Clara: *Animal-computer interaction: a manifesto*. interactions, 18(4):69–73, 2011.
- [11] Memrise. <https://www.memrise.com/es/>, [Online; visitada 06-Junio-2021].
- [12] Mermaid. <https://mermaid-js.github.io/mermaid/#/>, [Online; visitada 06-Junio-2021].
- [13] P., Damien: *Anki*, 2006. <https://apps.ankiweb.net/>, [Online; visitada 01-Mayo-2021].
- [14] Pets, The Spruce. <https://www.thesprucepets.com/six-weeks-to-a-well-trained-dog-1118224>, [Online; visitada 13-Julio-2021].
- [15] PULSARDEV. <https://quasar.dev/>, [Online; visitada 29-Diciembre-2021].
- [16] Puppr. <https://www.puppr.app/>, [Online; visitada 13-Julio-2021].

- [17] Runtastic, Adidas. <https://www.runtastic.com/es/>, [Online; visitada 14-Julio-2021].
- [18] Scistarter. <https://scistarter.org/citizen-science>, [Online; visitada 01-Mayo-2021].
- [19] Scistarter. <https://scistarter.org/finder>, [Online; visitada 01-Mayo-2021].
- [20] Shettleworth, Sara J: *Memory in food-hoarding birds*. Scientific American, 248(3):102–111, 1983.
- [21] SuperMemo. <https://www.supermemo.com/en/archives1990-2015/english/ol/sm2>, [Online; visitada 07-Julio-2021].
- [22] Sussex, University of. <https://profiles.sussex.ac.uk/p415999-charlotte-robinson/publications>, [Online; visitada 24-Diciembre-2021].
- [23] Wikipedia: *Animal-computer interaction* — *Wikipedia, The Free Encyclopedia*. <http://en.wikipedia.org/w/index.php?title=Animal-computer%20interaction&oldid=1015613723>, 2021. [Online; visitada 02-May-2021].
- [24] Wikipedia: *Clicker training* — *Wikipedia, The Free Encyclopedia*. <http://en.wikipedia.org/w/index.php?title=Clicker%20training&oldid=1013830455>, 2021. [Online; visitada 23-Julio-2021].
- [25] Wikipedia: *Leitner system* — *Wikipedia, The Free Encyclopedia*. <http://en.wikipedia.org/w/index.php?title=Leitner%20system&oldid=986552121>, 2021. [Online; visitada 13-Julio-2021].
- [26] Wikipedia: *Spaced repetition* — *Wikipedia, The Free Encyclopedia*. <http://en.wikipedia.org/w/index.php?title=Spaced%20repetition&oldid=1027670533>, 2021. [Online; visitada 08-Julio-2021].
- [27] Wikipedia: *Escala Likert* — *Wikipedia, The Free Encyclopedia*. <http://es.wikipedia.org/w/index.php?title=Escala%20Likert&oldid=141858990>, 2022. [Online; visitada 10-Febrero-2022].
- [28] You, Evan. <https://v3.vuejs.org/>, [Online; visitada 29-Diciembre-2021].

# ANEXOS

## Anexo A

### Vistas de la aplicación

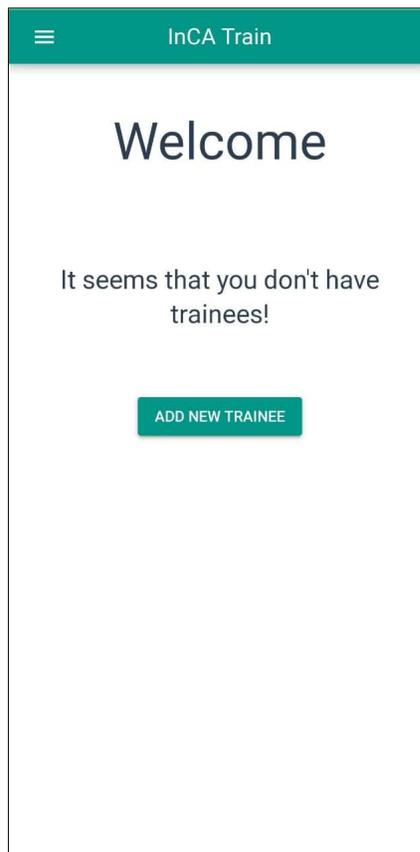


Figura A.1: Pantalla de inicio de InCA Train cuando no se tiene ningún aprendiz. Muestra un acceso directo a la creación de aprendices y no existe el ícono en la zona superior derecha.

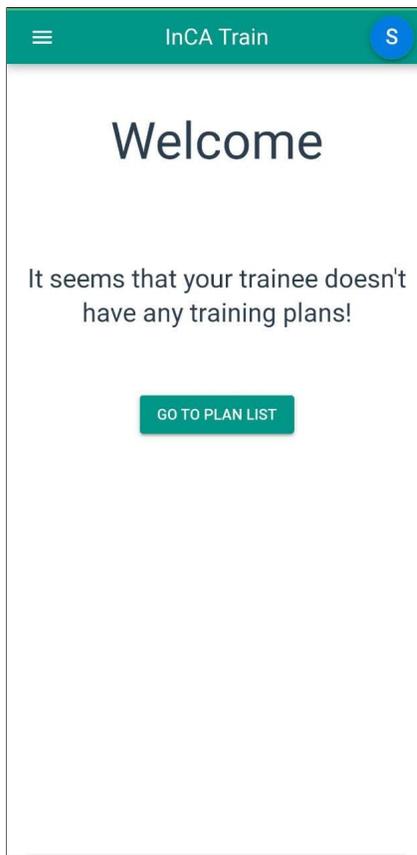


Figura A.2: Pantalla de inicio de InCA Train cuando el aprendiz seleccionado no tiene planes de entrenamiento. Muestra un acceso directo a la vista de planes.

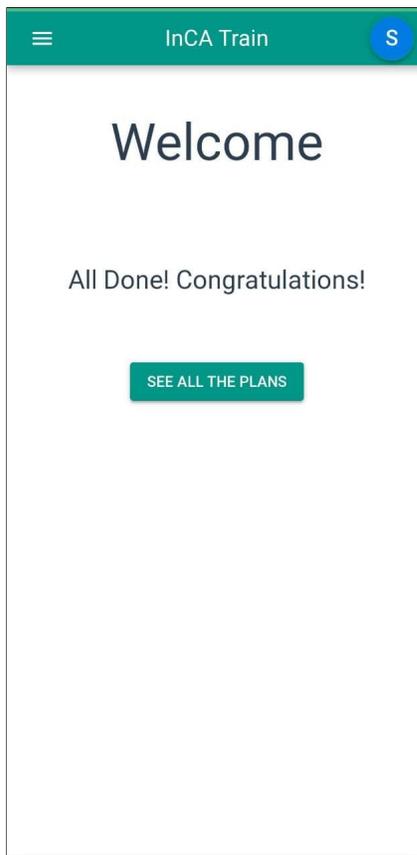


Figura A.3: Pantalla de inicio de InCA Train cuando se han completado las recomendaciones para el día actual y el aprendiz seleccionado, o bien, no existen recomendaciones para el día.

	Home
	Trainees
	Plans
	Logs
	Settings
	About

Figura A.4: Menú de navegación de la aplicación, que aparece al presionar el botón de la zona superior izquierda. Permite dirigirse a cualquier vista de la misma.

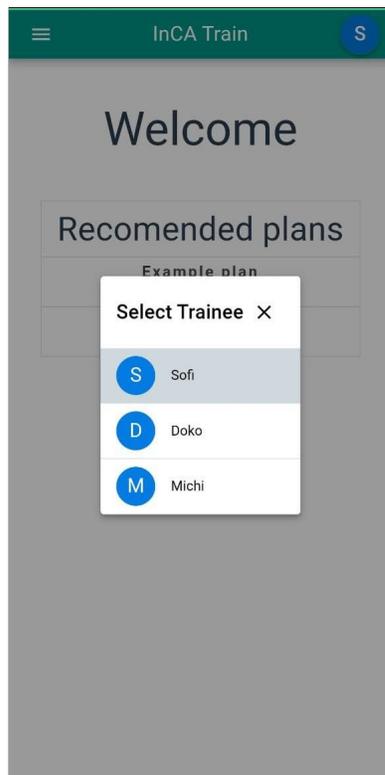


Figura A.5: Menú de selección de aprendices. En lo más alto y coloreado en gris se muestra el aprendiz actualmente seleccionado, seguido por el resto de aprendices creados por el usuario.

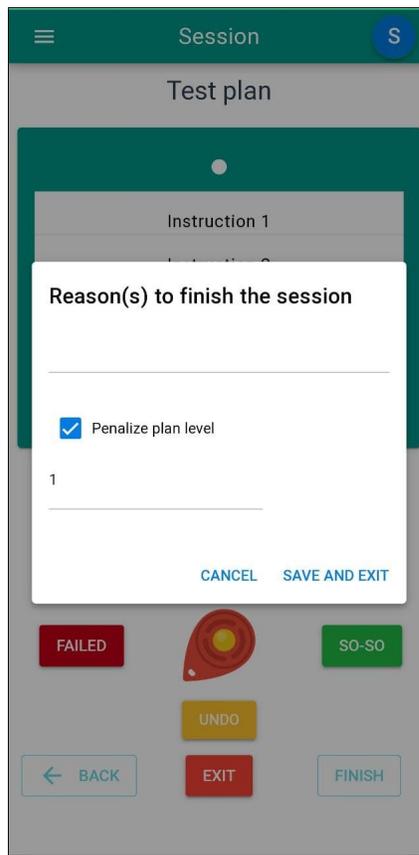


Figura A.6: Vista al presionar Exit en una sesión de entrenamiento y luego de aceptar el primer mensaje de advertencia. Se muestra un campo para escribir las razones de la salida prematura y la posibilidad de penalizar (disminuir) el nivel actual del plan.

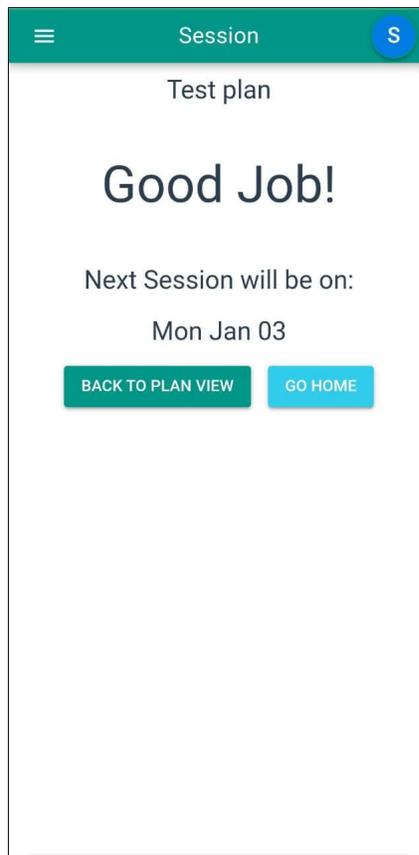


Figura A.7: Mensaje mostrado al finalizar la sesión, en el cual se ve la fecha propuesta para la siguiente sesión, calculada con los algoritmos de repetición espaciada.

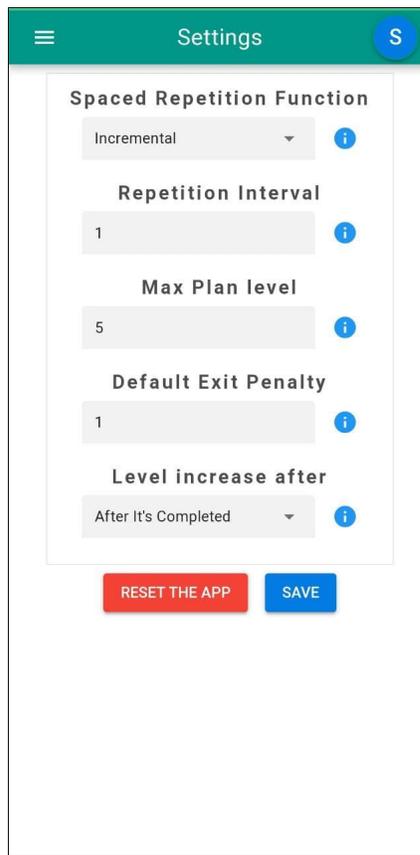


Figura A.8: Vista de las configuraciones de la aplicación. Cada campo contiene un icono de ayuda, el cual muestra un mensaje con una descripción de lo que hace cada variable.

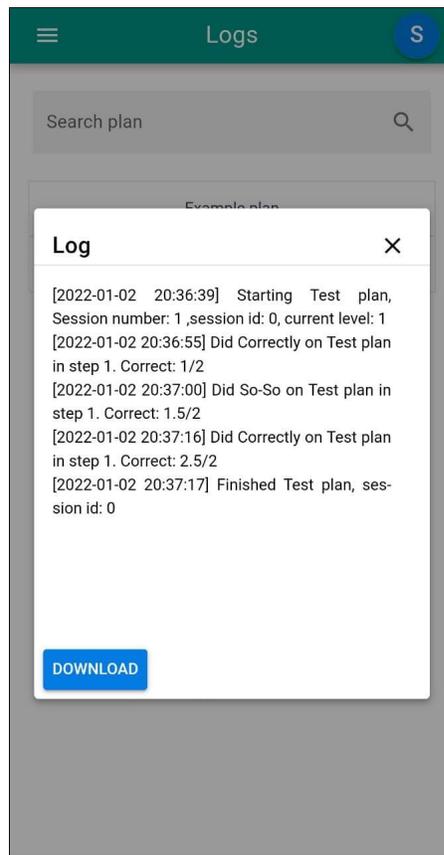


Figura A.9: Registros con lectura facilitada para los seres humanos. Posee botón de descarga, el cual genera un archivo csv con la información.

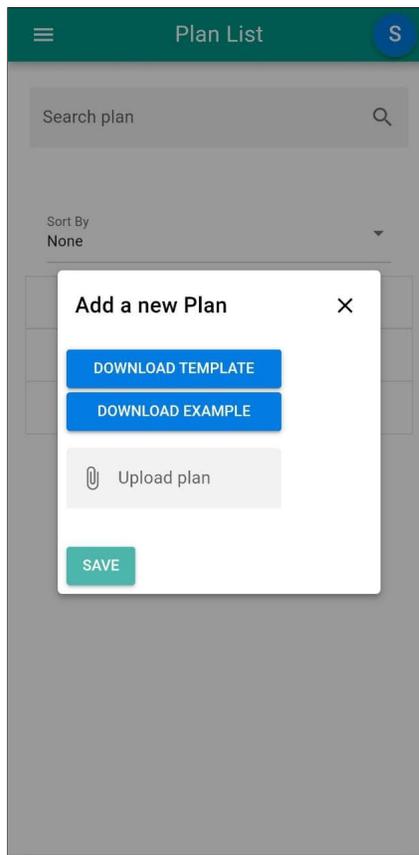


Figura A.10: Mensaje mostrado al tratar de subir un nuevo plan. Se ven dos botones para descargar un template (Apéndice B.1), descargar un plan de ejemplo (Apéndice B.2) y un campo para subir el plan.

# Anexo B

## Creación de planes

Listing B.1: Template de creación de planes. Contiene la estructura esperada y comentarios sobre lo que debería ir en la mayoría de los campos.

```
1  \\ You need to remove the comments before upload
2  {"title" : "title",
3   "madeBy": "user",
4   "description": "desc",
5  \\nodes are what you see in the graph (the squares with arrows)
6  "nodes": [
7     {
8       "id": ,
9       \\ The text that is shown in the graph
10      "text": "",
11      \\ A list with the id of the next node(s)
12      "next": [],
13      \\ which training session should the node redirect
14      (id)
15      "redirectTo": },
16  ],
17  "trainingSessions": [
18    {"id": 0,
19     \\ list of steps to follow in the training session,
20     each object has the amount of
21     \\ times it should be done (repeat) the
22     instructions of the step and
23     \\ an optional link to a youtube vid, it needs to
24     be a link to embeded video
25     \\ share button -> embed
26     "blocks": [
27       {"repeat": 0,
28        "instructions": ["do this", "then this"],
29        "linkToVid": ""},
30     ]},
31  ],
```

```
27     ]
28 }
```

Listing B.2: Ejemplo de uso del template. Contiene un plan simple cuya idea es que sea subido por el usuario y reconozca las partes del plan y las sesiones de entrenamiento.

```
1  {"title" : "Example plan",
2  "madeBy": "User",
3  "description": "plan description",
4  "nodes": [
5      {
6          "id": 0,
7          "text": "Training Session 0",
8          "next": [1],
9          "redirectTo": 0},
10     {"id": 1,
11      "text": "Training Session",
12      "next": [],
13      "redirectTo": 1}
14 ],
15 "trainingSessions": [
16     {"id": 0,
17      "blocks": [
18         {"repeat": 1,
19          "instructions": ["Do this", "Then this"],
20          "linkToVid": ""},
21         {"repeat": 2,
22          "instructions": ["Only this"],
23          "linkToVid": ""}
24     ]},
25     {"id": 1,
26      "blocks": [
27         {"repeat": 1,
28          "instructions": ["Do this", "Then this"],
29          "linkToVid": ""},
30         {"repeat": 2,
31          "instructions": ["Only this"],
32          "linkToVid": ""}
33     ]}
34 ]
35 }
```

# Anexo C

## Evaluación

### C.1. Preguntas en inglés

Preguntas en inglés utilizadas en la evaluación final de la aplicación con la Dra. Charlotte Robinson. Las primeras doce preguntas debían ser respondidas en una escala de 1 a 5 (totalmente desacuerdo a totalmente de acuerdo) y la última pregunta era abierta para ser respondida con palabras.

1. The application was easy to use.
2. The application was simple and straightforward to use.
3. I would like to use this application as a general user (trainer).
4. I would like to use the application for investigation purpose.
5. The logs will be useful for some research purpose.
6. I would like to use the application for investigation purpose (even if it did not have a log).
7. The application has the potential to enrich the life of animals.
8. The application allows the creation of plans.
9. The application allows the use of plans and training sessions.
10. The application provides recommendations of plans to use in a day.
11. The application allows to see logs.
12. The application allows to export logs.
13. What do you think the app needs or what would you like to see in it.

## C.2. Resultados de la evaluación

Respuestas de la Dra. Charlotte Robinson durante la evaluación final de la aplicación.

1. The application was easy to use. [4]
2. The application was simple and straightforward to use. [3.5]
3. I would like to use this application as a general user (trainer). [2.5]
4. I would like to use the application for investigation purpose. [5]
5. The logs will be useful for some research purpose. [5]
6. I would like to use the application for investigation purpose (even if it did not have a log). [3.5]
7. The application has the potential to enrich the life of animals. [5]
8. The application allows the creation of plans. [5]
9. The application allows the use of plans and training sessions. [5]
10. The application provides recomendations of plans to use in a day. [5]
11. The application allows to see logs. [5]
12. The application allows to export logs. [5]
13. What do you think the app needs or what would you like to see in it:
  - (a) Mapping the settings to subjects and plans would be important for doing research
  - (b) Validate and improve the digital clicker interface Clicker/SoSo/Fail to potentially improve it
  - (c) Back end to gather all the data

# Anexo D

## Tablas

date	traineeName	planId	level	sessionId	block	action
2022-01-03 15:24:57	Sofi	2	1	0	1	Start
2022-01-03 15:24:58	Sofi	2	1	0	1	Did Correctly
2022-01-03 15:25:01	Sofi	2	1	0	1	Did So-So
2022-01-03 15:25:03	Sofi	2	1	0	1	Undid
2022-01-03 15:25:05	Sofi	2	1	0	1	Did Correctly
2022-01-03 15:25:06	Sofi	2	1	0	1	Finished
2022-01-03 15:25:08	Sofi	2	1	1	1	Start
2022-01-03 15:25:09	Sofi	2	1	1	1	Failed
2022-01-03 15:25:10	Sofi	2	1	1	1	Failed
2022-01-03 15:25:36	Sofi	2	1	1	1	Exit

Tabla D.1: Ejemplo de archivo csv generado por la aplicación. Contiene los registros generados durante la ejecución de una sesión de entrenamiento. En este se ve a la aprendiz Sofi utilizando dos sesiones distintas del mismo plan y en la segunda salió de manera prematura.

action	corrects	expected	comments
Start	0	2	
Did Correctly	1	2	
Did So-So	1.5	2	
Undid	1	2	
Did Correctly	2	2	
Finished	2	2	
Start	0	2	
Failed	0	2	
Failed	0	2	
Exit	0	2	She was too distracted

Tabla D.2: Ejemplo de archivo csv generado por la aplicación. Contiene los registros generados durante la ejecución de una sesión de entrenamiento. En este se ve a la aprendiz Sofi utilizando dos sesiones distintas del mismo plan y en la segunda salió de manera prematura. (Continuación)