



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

INCA MEMORY:
UN VIDEO JUEGO PARA ANIMALES NO HUMANOS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

ERICK EDUARDO LEMUS FUENTES

PROFESOR GUÍA:
JÉRÉMY BARBAY

MIEMBROS DE LA COMISIÓN:
ALEXANDRE BERGEL
DANIEL CALDERÓN SAAVEDRA

SANTIAGO DE CHILE
2022

Resumen

Animal Computer Interaction (abreviado ACI) es un campo de investigación centrado en la interacción entre los animales no humanos y la computación (tecnología). También se encarga de estudiar como se puede mejorar el bienestar de animales en estado de cautiverio y por otro lado, avanzar en un camino hacia la comunicación entre animales de distintas especies o animales con humanos. Dentro de este contexto, se estableció comunicación con Jennifer Cunha, investigadora en ACI, para levantar los principales requisitos de una aplicación o juego orientado a animales no humanos. Posteriormente, se desarrolló un juego para dispositivos móviles, que permite a animales no humanos jugar (ayudando a reducir niveles de estrés) mientras que, al mismo tiempo le permite a guardianes e investigadores estudiar la capacidad de memoria y relacionamiento de objetos de estos animales. El juego incluye un sistema de parametrización, tanto de niveles de juego como de configuraciones generales, un mecanismo para poder subir imágenes, y por último, un mecanismo para poder exportar datos generados. Finalmente, Jennifer Cunha validó positivamente la aplicación.

A mi madre y mi padre.

Agradecimientos

Quiero agradecer a cada persona que me acompañó en este proceso. A mi Madre, Padre y Hermano que siempre me apoyaron y me entregaron palabras de aliento para seguir con mis estudios e hicieron lo posible para poder pagar mi carrera y no endeudarme. A mis amigos de plan común (Cata, Cami, Nacho, Camilo, Max, Luccas, Javier y Ariel) y mis amigos del DCC (Pablo, Joaco, Vale Valdés, Max, Vale Fuentes, Analia y Diego) que hicieron las tardes de estudio mucho más agradables y con quienes he compartido muchos momentos inolvidables. A aquellas personas (Luz, Pili y Pía) que me acompañaron incondicionalmente y me escucharon cuando me sentí frustrado. A Jennifer Cunha, que fue parte importante de este trabajo aportando con ideas y retroalimentación. Y finalmente, al profesor Jérémy que dedicó mucho tiempo en guiarme y hacerme un mejor profesional.

Tabla de Contenido

1. Introducción	1
1.1. Contexto y Problema a Abordar	1
1.2. Objetivos de la Memoria	2
1.3. Resultados	2
1.4. Estructura del Documento	3
2. Estado del Arte	4
2.1. Investigaciones en ACI	4
2.2. Juegos	5
2.3. Limitaciones de aplicaciones existentes	6
2.4. Frameworks de desarrollo	7
2.5. Literatura	8
2.6. Ética	8
3. Diseño de la Solución	10
3.1. Tipos de Usuarios	10
3.2. Principales Requisitos	10
3.3. Principales Procesos	11
4. Implementación de la Solución	16
4.1. Software	16
4.2. Vistas	16

4.3. Funcionalidades	23
5. Evaluación de la Solución	30
5.1. Participantes	30
5.2. Descripción del proceso de evaluación	30
5.3. Instrumento de evaluación de Usabilidad percibida	30
5.4. Resultados obtenidos	31
6. Conclusión y Trabajo a Futuro	33
6.1. Contribuciones	33
6.2. Lecciones Aprendidas	33
6.3. Trabajo futuro	34
Bibliografía	37
Anexo A. Vistas de la aplicación	38
Anexo B. Cuestionario de la evaluación final	44
Anexo C. Respuesta original a pregunta abierta	45

Índice de Ilustraciones

3.1.	Flujo para jugar. Siguiendo el diagrama, se debe ir a la sección “Start Game”, luego interactuar para completar 1 o más niveles hasta que el guardián intervenga o no aparezcan más niveles de juego (Juego Terminado).	12
3.2.	Flujo para configurar parámetros de juego. Desde el inicio, se debe dirigir a sección “Settings” para ir a las configuraciones donde se presentan configuraciones generales o específicas para cada juego. Una vez que se modifican las configuraciones, se deben guardar para finalizar el proceso.	14
3.3.	Flujo para agregar un mazo y sus cartas. El flujo consiste principalmente en crear un mazo vacío y asignarle un nombre para posteriormente iniciar un subproceso cíclico agregando fotos con sus respectivos nombres.	15
3.4.	Flujo para exportar datos y enviar por correo. Inicialmente se debe ir a la sección Logs y gatillar acción para generar archivo con los registros. Una vez que se genera el archivo se deben ingresar los datos solicitados para realizar el envío por correo.	15
4.1.	Vista de bienvenida de InCA-Memory. Se despliega un menú vertical con cuatro botones que redirigen a distintas vistas. Logo elaborado y proporcionado por Jérémy Barbay.	17
4.2.	Vista de juego. En la parte superior se ubica botón para finalizar juego y secciones con información referente a estado y número de juego. En la parte central se disponen las cartas.	18
4.3.	Vista de menú una vez finalizado el juego. En la parte central se muestra un botón grande con un ícono para repetir el tipo de juego, y también se colocan 3 botones para cambiar a otro tipo de juego.	19
4.4.	Vista de configuraciones generales. Se muestra la primera pestaña seleccionada para modificar las configuraciones generales, por ejemplo, nombre de guardián, nombre de animal, tipo de juego, cantidad de juegos, tipo de mazo, pausa inicial y pausa entre toque de cartas.	20

4.5.	Continuación de la vista de configuraciones generales. Desde arriba hacia abajo, pausa entre toque de cartas, seleccionar ayuda, seleccionar reproducción de sonido, pausa en respuesta correcta, pausa en respuesta incorrecta, sonido en respuesta correcta, sonido en respuesta incorrecta, sonido en puntaje excelente, sonido en puntaje promedio, sonido en puntaje deficiente.	21
4.6.	Vista de logs, acciones realizadas por animal no humano. Las columnas de cada log son: hora y fecha de acción, juego, animal, guardián, acción, anotación, número de juego, número de cartas, mazo, tamaño de grilla, cartas en pantalla, ayuda, sonido, pausa acción correcta, pausa acción incorrecta, pausa entre toque de cartas.	22
4.7.	Vista del listado de mazos de cartas. Se incorpora botón en el costado derecho para crear nuevo mazo. Cada elemento de la lista corresponde a un conjunto de cartas.	23
4.8.	Vista del detalle de mazo de cartas determinado. En la parte superior se muestra un campo de texto para insertar nombre del mazo. Inmediatamente después aparecen dos botones, uno para abrir la cámara y otro para abrir la galería. Debajo de los botones se muestra el listado de cartas, cada una con un botón para eliminar la carta del mazo.	24
4.9.	Estructura de los registros almacenados. Cada registro se guarda como un objeto, donde cada campo guarda un tipo de información, por ejemplo: hora y fecha, tipo de juego, cartas utilizadas, entre otros.	25
4.10.	Pseudocódigo que permite generar el archivo de logs. Se itera sobre el arreglo de logs para transformarlo en una cadena de texto uniendo todos los elementos de un mismo registro mediante comas.	26
4.11.	Configuraciones generales. Se detalla cada campo o parámetro de las configuraciones que se aplican de forma general al juego.	27
4.12.	Configuraciones específicas de cada juego. Se detallan los campos o parámetros de las configuraciones que se aplican a cada tipo de juego.	28
A.1.	Configuraciones específicas para juego Memory. Se muestra la segunda pestaña (Memory) seleccionada. Las configuraciones de arriba hacia abajo son, tamaño de grilla, mostrar grilla, máxima cantidad de cartas en pantalla.	38
A.2.	Configuraciones específicas para juego Matching. Se muestra la tercera pestaña (Matching) seleccionada. Las configuraciones de arriba hacia abajo son, tamaño de grilla, mostrar grilla, máxima cantidad de cartas en pantalla.	39
A.3.	Configuraciones específicas para juego Guess. Se muestra la cuarta pestaña (Guess) seleccionada. La configuración posible es máxima cantidad de cartas en pantalla.	39

A.4. Modal de confirmación para agregar carta. En el centro de la pantalla se muestra la imagen seleccionada y debajo se presenta un campo de texto para ingresar el nombre. Finalmente, se colocan dos botones, uno para cancelar la creación y otro para confirmar los datos de la carta.	40
A.5. Vista de inicio de juego Memory. En la parte superior se ubica un botón para volver al menú principal, y también se muestra información general del juego. En la parte central se muestran las distintas cartas, mientras que en el costado derecho se presenta el temporizador para iniciar el nivel.	40
A.6. Vista de juego Memory con la ayuda activada. En la parte central se muestran las distintas cartas escondidas (representadas con color verde). La ayuda se representa con el borde rojo, que es activada una vez que se revela una carta.	41
A.7. Vista de inicio de juego Matching. Se muestran las cartas en la parte central de la pantalla, de forma similar al juego Memory.	41
A.8. Vista de juego Matching con la ayuda activada. En la parte central se muestran todas las cartas reveladas. La ayuda se representa con el borde rojo, y se activa una vez que se presiona una carta (carta con tonalidad azul).	42
A.9. Vista de inicio de juego Guess. En la parte superior se muestra botón para volver al menú principal, y también información general del juego. Posteriormente se muestra una pregunta para seleccionar una determinada carta.	42
A.10. Vista de juego guess con la ayuda activada. En la parte central se muestran las cartas cubiertas, y la ayuda se representa con el borde rojo.	43
A.11. Modal de envío de correo. Al momento de presionar botón “Generate file”, se genera archivo y se le pide al dispositivo que lo envíe a través de correo electrónico, por lo que el dispositivo abre la ventana para escoger la aplicación de mensajería.	43

Capítulo 1

Introducción

1.1. Contexto y Problema a Abordar

Animal Computer Interaction (ACI) es un campo de investigación que tiene por objetivos (1) entender la interacción entre animales y computadores, (2) influenciar el desarrollo de tecnología para mejorar la calidad de vida de animales principalmente en cautiverio y fomentar la relación entre humanos y animales, y (3) desarrollar un enfoque centrado en el usuario, considerando las necesidades y preferencias de los animales [24]. Iniciado por investigadores en Human Computer Interaction en el año 2011, es un entorno pluridisciplinario y emergente.

Muchos investigadores en ACI no pertenecen al área de Ciencias de la Computación, sino ingenieros en electrónica, psicólogos, veterinarios, neurocientíficos e incluso guardianes dedicados a enriquecer la vida de animales no humanos y comunicarse de mejor manera con ellos. Por ejemplo, Cunha, fundadora de una escuela para guardianes de loros [17], es doctora en leyes con conocimientos de psicología, y actualmente investigadora en ACI con aves de la especie cacatúas, enseñándoles a leer, escribir, pintar o dibujar [18, 19, 22].

Existe poco material y software diseñado específicamente para animales no humanos. Por ejemplo, Hunger [21] usó botones diseñados por humanos con problemas de elocución para entrenar su perro. Su éxito generó un movimiento general considerando otras especies como gatos, aves [12], al mismo tiempo que se mejoró el diseño de los botones haciendo que sean más apropiados para perros. Otro ejemplo es Cunha [22], que recomienda el uso de varias aplicaciones y juegos, diseñados para niños o adultos con discapacidades, para enriquecer la vida y mejorar las capacidades comunicacionales de loros [7].

Existe un fuerte potencial de mejora en desarrollar herramientas específicamente para animales no humanos. A lo largo de sus investigaciones, Cunha [22] logró identificar varias funcionalidades deseables en las aplicaciones que ella ha utilizado, por ejemplo, adaptar las transiciones entre distintas pantallas para no alterar a los animales, otorgar independencia al animal para comenzar el juego, y que la aplicación se adapte al ritmo de aprendizaje del animal no humano [7].

1.2. Objetivos de la Memoria

Objetivo General

Desarrollar un primer prototipo de aplicación optimizada para animales no humanos, principalmente cacatúas, enfocado en el desarrollo y estudio de habilidades cognitivas a través de un juego para relacionar elementos (Matching) y para entrenar la memoria a corto plazo a través del juego memorice (Memory).

Objetivos Específicos

1. Identificar funcionalidades esenciales de aplicaciones de tipo Memory y Matching para animales no humanos, y en particular, cacatúas.
2. Implementar un juego simple de tablet que permita apoyar en la investigación de habilidades cognitivas en cacatúas, considerando las limitaciones al momento de interactuar con un dispositivo electrónico y el bienestar del animal.
3. Validar con investigadora en ACI que se puede usar la aplicación para estudiar las hipótesis deseadas.

Cabe notar que, aunque el propósito de usar aplicación de tipo “Matching” o “Memory” con animales no humanos es validar hipótesis científicas sobre el enriquecimiento, los modelos de memoria de dichos animales y/o la habilidad de relacionar objetos, tal propósito no entra en los objetivos del presente trabajo de memoria, sino de producir un primer prototipo que se podría utilizar a futuro para validar tales hipótesis que requieren años de investigación, lo que impide ser consideradas dentro del alcance de un trabajo de memoria.

1.3. Resultados

El proceso de validación de la aplicación fue llevado a cabo por Jennifer Cunha, investigadora en ACI, quién tuvo acceso a la aplicación para poder realizar distintas pruebas junto a un ave de la especie cacatúa.

Una vez que la aplicación fue probada, se le realizó un cuestionario de 10 preguntas a Jennifer Cunha para poder determinar el nivel de usabilidad y utilidad del juego, donde cada pregunta entregó un puntaje siguiendo una escala Likert [23] de 5 puntos.

El promedio de puntajes fue de 4.6 de un máximo de 5, lo que es un resultado positivo para la aplicación, además de entregar observaciones positivas acerca del uso del juego.

1.4. Estructura del Documento

A continuación se presenta un listado con los capítulos contenidos en esta memoria y un breve resumen de cada uno.

- **Capítulo 1: “Introducción”**. En este capítulo se da a conocer la problemática a resolver, los objetivos propuestos y los principales resultados obtenidos.
- **Capítulo 2: “Estado del Arte”**. En este capítulo se plasma el estado en que se encuentra ACI, algunas aplicaciones utilizadas por animales no humanos y sus respectivas ventajas y desventajas.
- **Capítulo 3: “Diseño de la Solución”**. Este capítulo contiene información asociada al proceso de levantamiento de requisitos, así como la identificación de las partes principales y necesarias para obtener un software usable y útil.
- **Capítulo 4: “Implementación de la solución”**. Capítulo destinado a presentar las distintas interfaces, funcionamiento de cada componente y como interactúan entre sí el Back End y el Front End.
- **Capítulo 5: “Evaluación de la Solución”**. En este capítulo se da a conocer la forma en que se evaluó la utilidad y usabilidad de la plataforma.
- **Capítulo 6: “Conclusiones y Trabajo a futuro”**. Capítulo destinado a resumir el trabajo realizado, así como también contrastar los resultados obtenidos con los objetivos planteados. Además se incorpora una sección para sentar las bases de trabajos futuros relacionados a esta memoria.

Capítulo 2

Estado del Arte

En el campo de Animal Computer Interaction (ACI) convergen distintas áreas: cognición animal, juegos e interfaces. En este capítulo se profundiza en cada uno de los tópicos mencionados. Se comienza con Investigaciones en ACI (Sección 2.1) donde se muestran investigaciones realizadas por la investigadora Cunha, relacionadas al estudio de habilidades animales asociadas a pintar, dibujar o escribir. Luego, en la Sección 2.2 se abarcan ejemplos de juegos que son de interés y motivación, pertinentes a las habilidades que pueden ser investigadas. Posteriormente, en la Sección 2.3 se establecen las limitaciones que presentan las interfaces de distintos juegos, las que fueron consideradas dentro del proyecto desarrollado, para luego continuar con la Sección 2.4 presentando las posibles herramientas para desarrollar una aplicación, y finalmente, concluir en la Sección 2.5 con algunos estudios que avalan el uso ético de aplicaciones en el bienestar de animales no humanos en cautiverio.

2.1. Investigaciones en ACI

A la fecha del 3 de mayo de 2021, se pudo encontrar aplicaciones creadas para humanos en la tienda Google Play, que pueden ser útiles a no humanos. Dentro de las distintas categorías de aplicaciones existen algunas creadas con fines pedagógicos, enfocándose en ayudar a desarrollar habilidades cognitivas en personas, tales como, memoria, comunicación, idiomas, lógica, entre otras.

Cunha ha realizado investigaciones y experimentos con aves de la especie cacatúas [19, 22]. El principal método que ha usado es con reforzamientos positivos [26] luego de que las aves completan una secuencia de tareas de manera satisfactoria. Las habilidades investigadas por Cunha son variadas, por ejemplo, reconocimiento de objetos, memoria, escritura y lectura [18]. Entre otros resultados demostró que las cacatúas pueden tener la capacidad de procesar significados y la correspondencia grafema-fonema [22].

Las aves con las que trabaja Cunha han entrenado sus habilidades a través de aplicaciones para dispositivos electrónicos como tablets. Existen muchos juegos diseñados principalmente para humanos enfocados en emparejar dos elementos, o de recordar pares de imágenes y

descubrir las, sin embargo, son difíciles para los animales, y no les otorga autonomía para poder disfrutar y aprender, es más, requiere de la participación del guardián del animal para iniciar y preparar el juego, e incluso para desbloquear configuraciones que son necesarias para que un animal pueda ocupar la aplicación.

Cada aplicación posee características favorables para realizar estudios, y funcionalidades que impiden buenos resultados. Por esta razón, Cunha ha usado muchos juegos, lo que le ha permitido detectar las principales características positivas y negativas, que son presentadas a continuación:

- Ventajas
 - Entrenamiento de habilidad para hacer touch en la pantalla.
 - Se enfocan en habilidades como Emparejamiento y Memoria.
- Desventajas
 - Requieren de guardián para comenzar el juego (resolver parte del juego para configurar la dificultad al comenzar).
 - Obtener información de sesiones de juego es largo y tedioso, porque se debe grabar la sesión de juego y procesar el video producido.

Es importante destacar las capacidades de memoria que poseen algunas aves. Como se menciona en “Memory in food-hoarding birds“ [29], algunas especies de aves poseen la habilidad de recordar miles de escondites donde almacenan sus semillas, incluso durante varios meses, lo que demuestra que poseen una excelente memoria espacial.

2.2. Juegos

Tipo Memory

Se consideran como juegos de tipo Memory a los juegos que permitan entrenar la memoria a corto o largo plazo. Es decir, juegos en que los usuarios deban retener información por una cantidad limitada de tiempo, para posteriormente utilizar la información para lograr un objetivo determinado.

Algunos ejemplos de juegos de memoria son:

1. Memorice: Se presentan un conjunto de pares de cartas desordenadas, que son mostradas al jugador por un tiempo determinado. Posteriormente el jugador debe voltear de a dos cartas, si estas coinciden deben quedar descubiertas, en caso contrario se deben voltear. El juego continúa hasta que todas las cartas quedan descubiertas.
2. Memorizar imágenes: El jugador debe observar una imagen por una cantidad de tiempo determinada y posteriormente responder una serie de preguntas acerca de la imagen mostrada.

3. Encadenar palabras: Juego en que se deben decir palabras en orden de tal forma que la siguiente palabra debe comenzar con la última letra de la palabra anterior. De esta forma, cada jugador debe recordar la última palabra mencionada.
4. Formar Palabras: Juego en que se presentan un conjunto de letras, y el participante debe formar todas las palabras posibles que tengan un significado válido.
5. Simón Dice: Juego en que una persona dicta una secuencia de movimientos, y el resto de participantes debe ejecutar las acciones. El juego es incremental, es decir, en cada turno la cantidad de acciones se incrementa, por lo que es esencial memorizar los movimientos.

Tipo Matching

Por el contrario de los juegos de tipo Memory, los de tipo Matching entrenan la habilidad de relacionar información. Es decir, juegos en los que se debe establecer una conexión entre distintos objetos.

Algunos ejemplos de este tipo de juego son:

1. Learn Numbers 123: Juego en que se muestran fichas con distintos números (cada número está presente dos veces en el tablero). El objetivo es seleccionar pares de números que son idénticos hasta que no queden fichas en el tablero.
2. Mahjong: Juego de mesa chino que consiste en seleccionar pares de fichas idénticas que se encuentren libres según las reglas del juego.
3. Binary Choice: La esencia del juego es poder relacionar un concepto con una imagen u objeto, por lo que se puede considerar implícitamente como un juego de tipo Matching.
4. Candy Crush: Este juego consiste en agrupar conjuntos de al menos tres piezas idénticas con solo un movimiento para hacer desaparecer las fichas del tablero.

2.3. Limitaciones de aplicaciones existentes

Existen demasiadas aplicaciones para entrenar memoria (Memory), sin embargo, las que se han utilizado con fines investigativos por Cunha no estaban muy adaptadas para animales no humanos [7], lo que complicó poder realizar estudios con ellos. Las limitaciones que presentaban este tipo de aplicaciones son las siguientes:

1. Requieren una intervención larga (resolver parte del juego) de un humano para poder configurar niveles.
2. Transiciones o sonidos indeseables del juego pueden alterar al animal no humano.
3. Cantidad de elementos en pantalla no es siempre el adecuado para el animal no humano.
4. Dificultad de juego no se adapta al ritmo de aprendizaje.

También existen aplicaciones para entrenar la habilidad de emparejamiento (Matching), pero al igual que el caso de Memory, este tipo de aplicaciones estaban diseñada para humanos, por lo que presentaban limitaciones al momento de ser usadas por animales no humanos. A continuación se mencionan las restricciones habituales en aplicaciones de Matching:

1. Requieren de intervención humana para comenzar y configurar juego y niveles de dificultad.
2. Juego no se adapta al ritmo de aprendizaje de un animal no humano.
3. En muchos juegos, para hacer emparejamiento de elementos, se necesita que el usuario arrastre un elemento hacia el otro, que es una acción compleja para animales no humanos.
4. Sonidos, contrastes de colores y transiciones de pantallas pueden alterar al animal no humano.

Una limitación adicional a las aplicaciones se encontraba al momento de hacer la entrega de premios cuando se concluye una tarea de forma exitosa. Una forma de resolver esta limitante, es crear dispositivos mecánicos que respondan a señales bluetooth enviadas por una aplicación, o también enviadas a dispositivos que estén en la misma red de internet. Sin embargo, esto no está considerando dentro del trabajo de memoria.

Al ser ACI una disciplina emergente requiere de herramientas que se adapten más a sus necesidades, por lo que se consideró la opción de proyectos de código abierto, para que una vez finalizado el trabajo, este quede disponible para la comunidad como una base de trabajos futuros en el mismo contexto de Animal Computer Interaction. Además, los juegos desarrollados son bastante simples como para haber adaptado una versión existente de código abierto.

2.4. Frameworks de desarrollo

Parte importante para desarrollar el proyecto, es poder definir las herramientas que serán utilizadas. En este contexto, existen distintos frameworks para crear aplicaciones móviles, en distintos lenguajes de programación, entre los que destacan los siguientes:

- **Flutter [4]:** Framework desarrollado por Google, que permite construir y compilar aplicaciones nativas para dispositivos Android, IOS o escritorio a partir de una misma base de código. Es una herramienta open source utilizando el lenguaje de programación Dart.
- **React Native [9]:** Framework desarrollado por Facebook. Es una herramienta open source que permite construir aplicaciones para Android, IOS o web a partir de una misma base de código. Esta construida con Javascript para luego ser compilada a código nativo.

- **Expo [3]:** Herramienta open source que permite crear, compilar y distribuir aplicaciones de React Native, con el uso limitado de algunas librerías.
- **Ionic [5]:** Framework de código abierto, que permite desarrollar aplicaciones cross-platform a partir de una misma base de código en Javascript. Ionic ofrece una librería de componentes optimizados.
- **Svelte Native [11]:** Framework que permite construir aplicaciones nativas de Android y IOS utilizando Svelte [10] y NativeScript [6]. Svelte es un framework para desarrollar páginas web utilizando Javascript, mientras que NativeScript permite transpilar el código Javascript a código nativo de los dispositivos Android e IOS.
- **Cordova [2]:** Herramienta open source que permite construir aplicaciones móviles utilizando HTML, CSS y Javascript.
- **Xamarin [13]:** Framework que permite crear aplicaciones para múltiples plataformas utilizando .NET y lenguaje C#. Xamarin fue desarrollado por Microsoft.

Considerando los frameworks mencionados, se opta por utilizar Expo, que además de otorgar una misma base de código tanto para Android como IOS, también permite compilar la aplicación y se encarga de posibles errores de compatibilidad entre distintas versiones de Android.

2.5. Literatura

El bienestar de los animales es una componente principal de este proyecto, es decir, se debe considerar en todo momento que el animal no humano no sufra alteraciones y no se estrese a causa del juego. En relación a esto, existe un estudio que avala el uso de juegos computarizados (y otros factores) como un impacto positivo en el bienestar de animales en estado de cautiverio [32].

Otro factor importante de las aves, es la capacidad de recordar cosas, en específico cientos o miles de escondites donde guardan semillas. Esto se ha reflejado en una investigación de Sara Shettleworth [29], donde ha identificado una especie de ave (Cascanueces Americano) con excelente memoria espacial. Por lo que esta aplicación podría permitir estudiar la capacidad de memoria de otros tipos de aves, logrando así aumentar el conocimiento en este campo de investigación.

2.6. Ética

En muchas ocasiones, los animales no humanos son utilizados como sujetos de investigación siendo sometidos a estudios que en ocasiones pueden provocar daños en los animales, debido al estrés y frustración condicionados por una interacción hostil con el entorno.

Este proyecto se realiza bajo dos principios: permitir a investigadores realizar estudios, y principalmente, influir positivamente en el bienestar de los animales no humanos. Para lograr influir en el bienestar de los animales es necesario que la afinidad con la tecnología y el consentimiento por parte del usuario sean una prioridad, tal como menciona Mancini en [25]. Debido a esto, se pone como prioridad que la interacción del animal no humano sea lo menos estresante posible, y evitar escenarios de frustración durante el transcurso del juego. Esto último se puede lograr probando la aplicación exhaustivamente en humanos antes de que sea probada por animales no humanos.

Capítulo 3

Diseño de la Solución

Una parte muy fundamental en el desarrollo de software es planificar y diseñar antes de implementar, por lo que en este capítulo se detalla la información necesaria para desarrollar un software que sea útil y usable. Primero, se describen los perfiles de usuarios que usarán esta aplicación (Sección 3.1), posteriormente, se enuncian los principales requisitos que se deben satisfacer, tales como poder configurar dificultad de juego, o poder añadir más elementos, o bien generar las estadísticas y que se puedan almacenar para que el guardián pueda acceder a ellas de manera sencilla (Sección 3.2). Y finalmente, se detallan los principales procesos de la aplicación (Sección 3.3).

3.1. Tipos de Usuarios

Para el uso de esta aplicación se identifican tres tipos de usuarios: “Investigador”, “Guardián” y “Jugador”

- **Jugador:** Usuario (animal no humano) que interactuará con la sección de juego de la aplicación, por lo que se debe implementar una interfaz que sea amigable y sencilla de utilizar.
- **Guardián:** Usuario que se encargará de configurar el tipo de juego para el usuario de tipo Jugador.
- **Investigador:** Usuario que se encargará de procesar los datos generados, por lo que es necesario recopilar datos que sean valiosos para que los Investigadores puedan realizar estudios de manera más conveniente que con las aplicaciones tradicionales.

3.2. Principales Requisitos

Los requisitos principales se establecieron en conjunto con Jennifer Cunha, quién ha probado distintos tipos de juegos y ha reunido información suficiente para pedir requisitos básicos

en una aplicación.

La lista de requisitos principales es la siguiente:

1. Mostrar una o más parejas de cartas en pantalla, que deben ser presionables por el usuario.
2. Debe tener múltiples niveles de juego.
3. Aumento progresivo en la dificultad de los niveles.
4. Juego se debe adaptar al ritmo de aprendizaje del animal no humano.
5. Guardián debe poder agregar más diseños de cartas.
6. Guardián debe tener la opción de ver estadísticas de juego.
7. Juego debe otorgar retroalimentación al animal no humano.

3.3. Principales Procesos

En la aplicación existen procesos fundamentales por los que deben pasar los usuarios para poder familiarizarse con la aplicación. Los procesos, que serán explicados a continuación, son los siguientes:

1. **Jugar**, que está enfocado para el usuario de tipo Jugador.
2. **Parametrizar nivel de juego**, que está centrado para el usuario de tipo Guardián.
3. **Crear conjunto de cartas**, que está diseñado para el usuario de tipo Guardián.
4. **Exportar datos**, que está enfocado para los usuarios de tipo Guardián e Investigador.

Jugar

El proceso más importante de la aplicación corresponde a la interacción directa del animal no humano con el juego. Y además, se debe evitar que en este flujo el animal no humano deba tomar decisiones innecesariamente, o explicado con otras palabras, en este flujo el animal no humano debe limitarse solamente a jugar, por lo que debe ser un flujo sencillo.

En la Figura 3.1 se observa el proceso para jugar, iniciando con el ingreso a la sección de "Start Game", posteriormente, se debe completar el nivel de juego. Una vez que se finaliza un nivel, se debe comprobar si se puede jugar otra vez, esto quiere decir, si la cantidad de partidas jugadas es menor al límite seleccionado por el guardián. Una vez que el animal no humano completa todos los niveles de la sesión el juego termina y junto con ello finaliza el proceso. También debe existir la opción de que el guardián pueda interrumpir la sesión de juego en caso de que el animal no humano se canse de jugar.

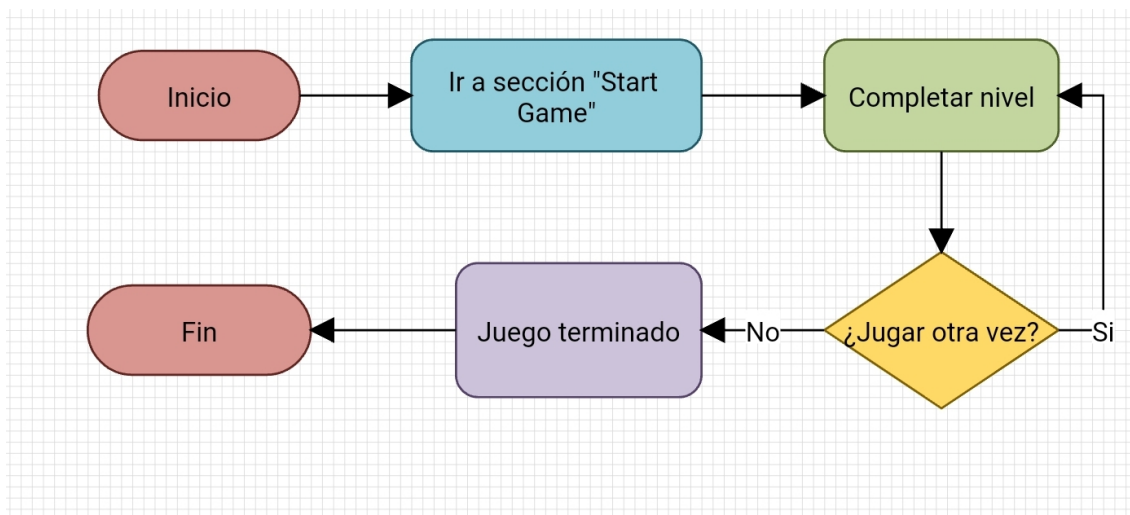


Figura 3.1: Flujo para jugar. Siguiendo el diagrama, se debe ir a la sección “Start Game”, luego interactuar para completar 1 o más niveles hasta que el guardián intervenga o no aparezcan más niveles de juego (Juego Terminado).

Parametrización

En la Figura 3.2 se muestra el proceso para poder configurar los parámetros de la aplicación, es decir, tanto aspectos generales del juego, como configuraciones aplicables a un tipo de juego en particular. El primer paso para este proceso es dirigirse a la opción de Configuraciones/Settings. Posteriormente, se debe escoger el tipo de configuración a modificar, es decir, configuraciones generales o específicas para cada juego. En el caso de escoger modificar los parámetros para un juego en particular, sólo se puede cambiar la cantidad de cartas a mostrar y el tamaño de la grilla en que se ordenan las cartas.

Por otro lado, las configuraciones generales corresponde a un conjunto más grande de parámetros, los que se aplican a todos los juegos. En este caso, lo primero que se debe poder modificar es el nombre de los participantes en el juego, posteriormente, ingresar el tipo de juego inicial, la cantidad máxima de juegos para una sesión y las cartas que se utilizarán. Después, se puede modificar el tiempo de espera entre acciones determinadas y también se debe poder escoger si el juego será con ayuda, o reproducción de sonidos (lo que también habilita la opción para ingresar los textos que serán reproducidos en sonido).

Para finalizar, se realiza el guardado de los parámetros y el proceso de configuración queda finalizado.

Conjunto de cartas

Tal como se observó en la Figura 3.2, existe un paso que consiste en seleccionar las cartas a utilizar. Es por esto que también debe existir un proceso que permita modificar los conjuntos de cartas (agregar, modificar o eliminar).

El proceso para poder modificar los mazos de cartas, consiste en primero dirigirse a la

sección “Desk” y luego seleccionar el mazo a modificar (o bien dirigirse a la sección “Add desk”), luego se debe ingresar el nombre que tendrá el conjunto de cartas. Finalmente, sigue un subproceso cíclico en que se deben ir agregando cartas, ya sean a través de la cámara o de la galería del dispositivo, para luego ingresar el nombre de la carta y confirmar si se debe guardar o no.

Exportar datos

El proceso más importante para las personas que se dedican a la investigación, es poder exportar los datos recopilados en las sesiones de entrenamiento.

El proceso debe ser simple, tal como se muestra en la Figura 3.4, donde primero se debe dirigir a la sección destinada a mostrar los registros (Logs), posteriormente gatillar la acción para generar el archivo que debe ser enviado por correo y finalmente completar la información necesaria para enviar los datos por correo electrónico.

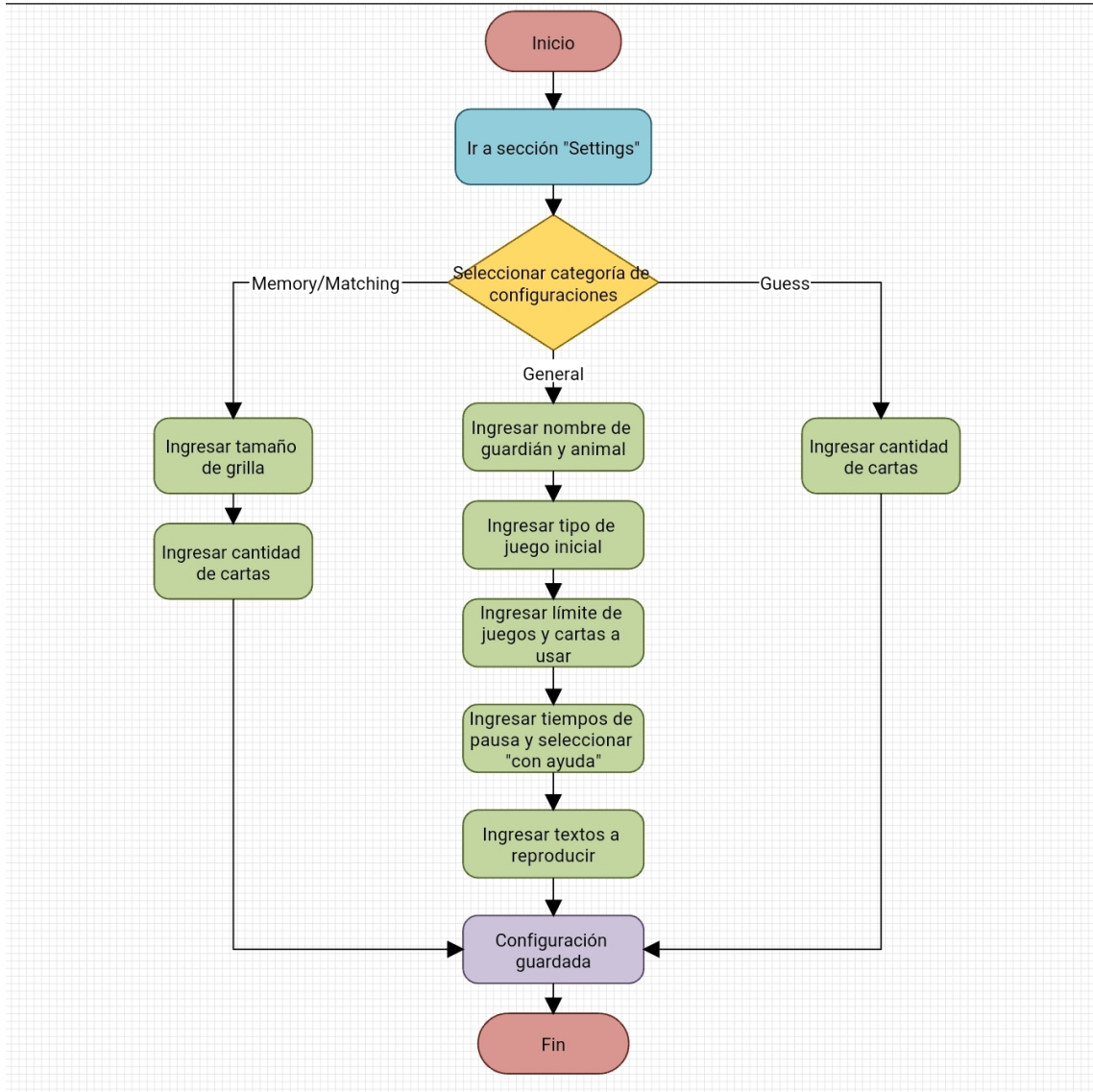


Figura 3.2: Flujo para configurar parámetros de juego. Desde el inicio, se debe dirigir a sección “Settings” para ir a las configuraciones donde se presentan configuraciones generales o específicas para cada juego. Una vez que se modifican las configuraciones, se deben guardar para finalizar el proceso.

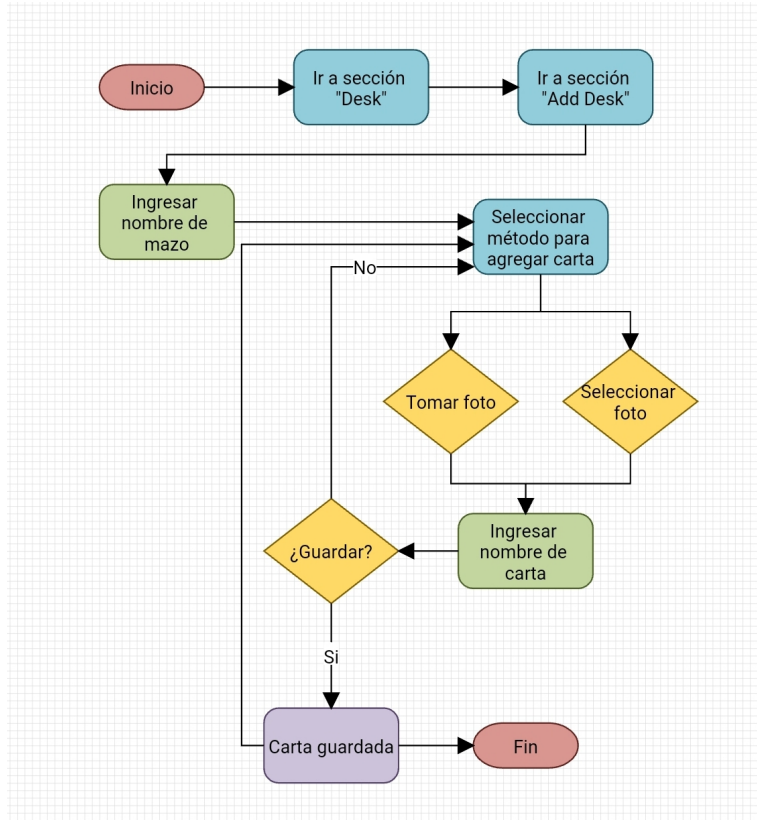


Figura 3.3: Flujo para agregar un mazo y sus cartas. El flujo consiste principalmente en crear un mazo vacío y asignarle un nombre para posteriormente iniciar un subproceso cíclico agregando fotos con sus respectivos nombres.

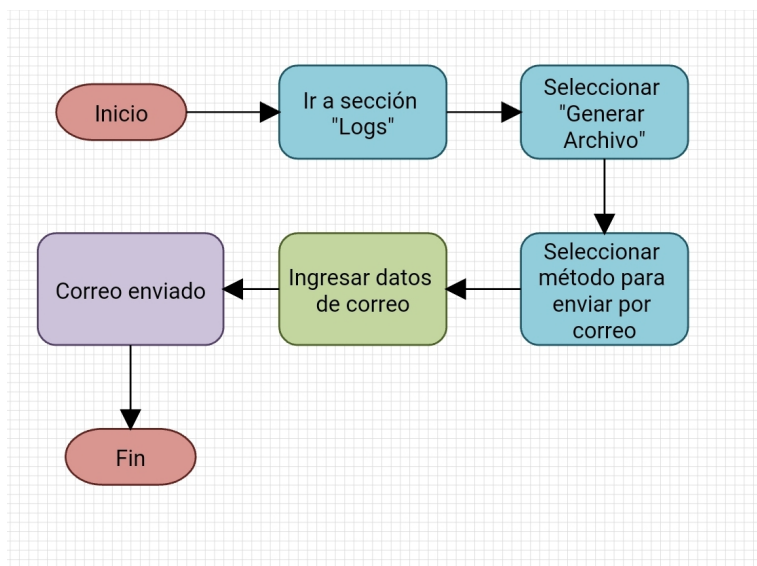


Figura 3.4: Flujo para exportar datos y enviar por correo. Inicialmente se debe ir a la sección Logs y gatillar acción para generar archivo con los registros. Una vez que se genera el archivo se deben ingresar los datos solicitados para realizar el envío por correo.

Capítulo 4

Implementación de la Solución

El proceso de implementación considera aspectos tales como vistas incorporadas (Juego, Configuración, Logs), funcionalidades añadidas (usuarios, múltiples tipos de juegos, configuración por parte del guardián, juego infinito, etc), y estructura de la información que se guarda. Las vistas diseñadas y añadidas fueron probadas por personas voluntarias, quienes entregaron retroalimentación para poder mejorar la experiencia de usuario y entregar más valor al prototipo final. Este capítulo plasma lo mencionado recientemente, así como también las opciones que se contemplaron y decisiones que se tomaron al momento de desarrollar las distintas vistas y funcionalidades de la aplicación.

4.1. Software

Inicialmente, se decidió elaborar el juego bajo una licencia Open Source, lo que permite que esta aplicación sea de código abierto y quede disponible para la comunidad de ACI.

Posterior a considerar los distintos frameworks de desarrollo mencionados en el Capítulo 2.4, se decidió utilizar Expo debido a que se posee experiencia previa utilizando este framework, lo que permitió reducir tiempos de investigación sobre tecnologías. Además, la utilización de Expo permitió establecer una misma base de código para distintos sistemas operativos como IOS y Android, aunque principalmente se quiso otorgar más prioridad a que la aplicación funcionara en dispositivo con Android. Por otra parte, Expo ofrece una librería que permite conectarse con distintas API's que interactúan con los componentes nativos del dispositivo.

4.2. Vistas

El prototipo de juego enfocado en habilidad de memoria y relacionamiento, está separada en distintas vistas, cada una encargada de realizar una función dentro del juego, las que son nombradas a continuación:

- **Home:** Vista encargada de dar la bienvenida al juego y ofrecer acceso a las otras vistas de la aplicación.
- **Game:** Vista encargada de mostrar cada uno de los subjuegos de la aplicación, así como también, encargada de mostrar información relacionada al estado del juego.
- **Settings:** Vista destinada a mostrar las configuraciones para cada juego, donde el guardián o investigador puede configurar el nivel de juego.
- **Desk:** Vista encargada de mostrar el listado de conjuntos de cartas disponibles para jugar, además de ofrecer acceso a crear un nuevo conjunto.
- **Logs:** Vista destinada a mostrar las acciones realizadas al momento de que el animal no humano esté jugando.

Home Screen

La vista más sencilla de la aplicación está plasmada en la Figura 4.1, donde se muestran cuatro botones que direccionan al resto de vistas principales de la aplicación.

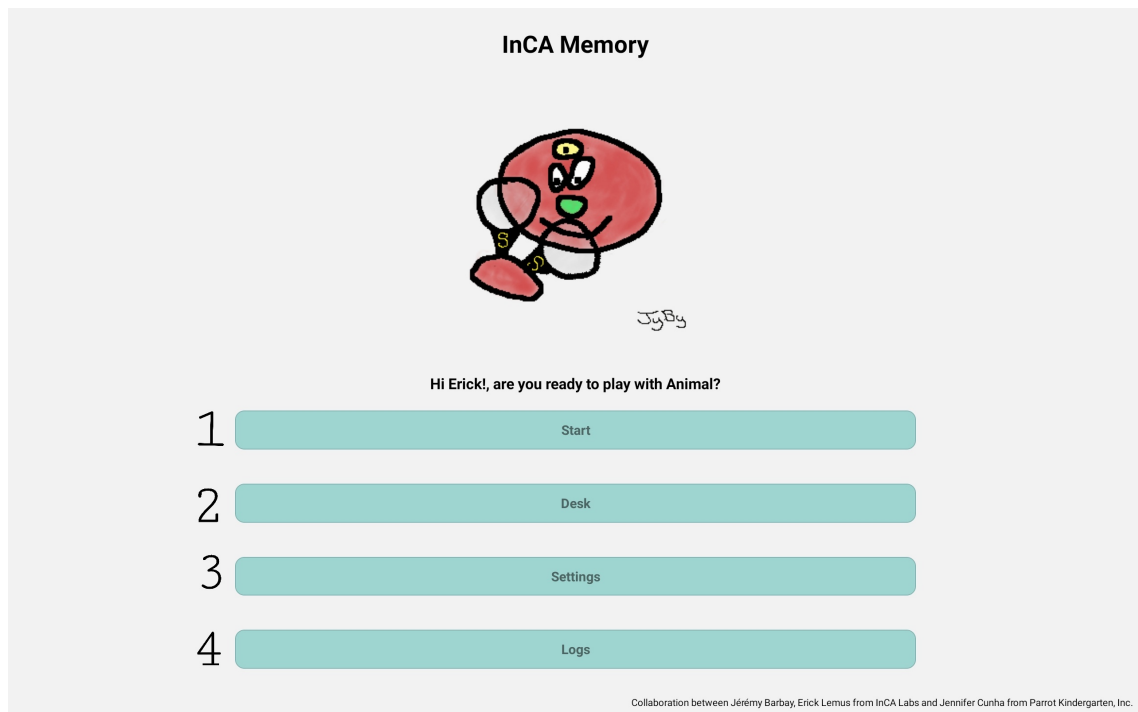


Figura 4.1: Vista de bienvenida de InCA-Memory. Se despliega un menú vertical con cuatro botones que redirigen a distintas vistas. Logo elaborado y proporcionado por Jérémy Barbay.

Los números de la Figura 4.1 indican cada uno de los botones que direccionan a las vistas, siguiendo el orden, los botones son:

1. **Start:** Botón que dirige a la vista de juegos.
2. **Desk:** Botón para mostrar la vista de conjuntos de cartas.

3. **Settings:** Botón que redirecciona a la vista para editar las configuraciones generales de juego y las configuraciones particulares de cada juego.
4. **Logs:** Botón que permite mostrar la vista de logs o registros de la aplicación, es decir, cada acción realizada al momento de jugar.

Game Screen

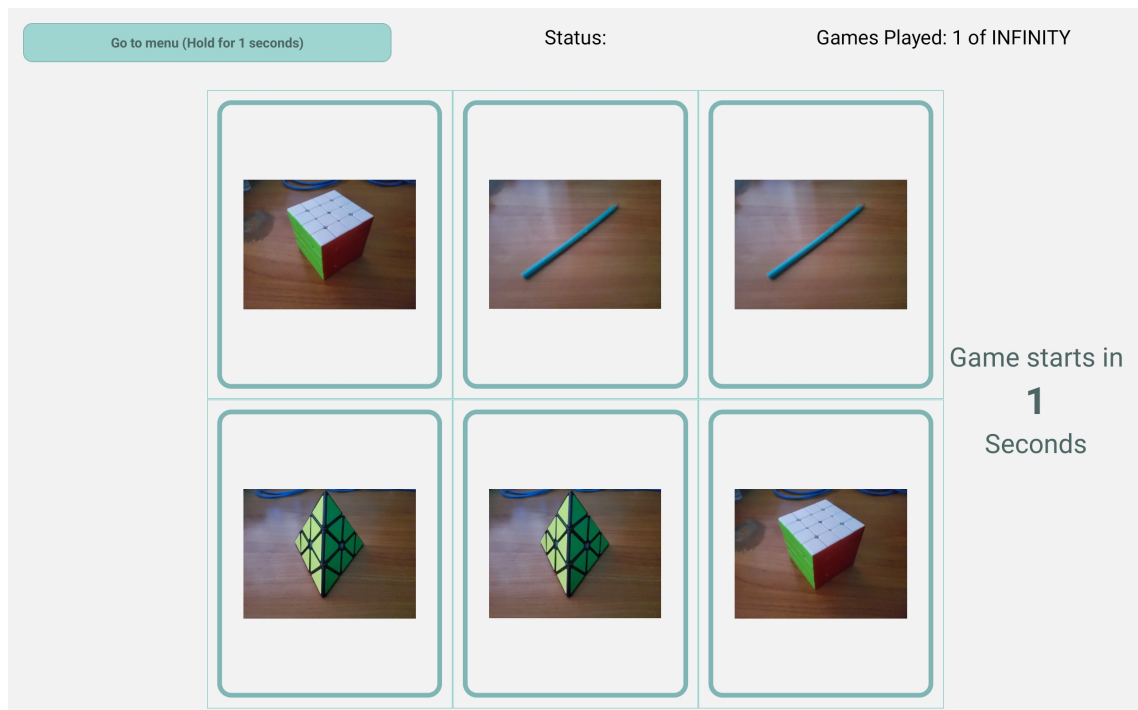


Figura 4.2: Vista de juego. En la parte superior se ubica botón para finalizar juego y secciones con información referente a estado y número de juego. En la parte central se disponen las cartas.

La vista principal de la aplicación corresponde a la vista de juego, donde destacan tres tipos de juegos:

- **Memory:** Juego donde se presenta un conjunto de pares de cartas que son cubiertas luego de un tiempo determinado, el juego consiste en descubrir de a dos cartas con la finalidad de que estas coincidan.
- **Matching:** Juego similar a Memory, con la principal diferencia que las cartas no son cubiertas, es decir, el jugador está observando todo el momento las cartas y el objetivo es discernir cuales son los pares de cartas idénticas entre sí.
- **Guess:** Juego en el que se muestra un conjunto de 1 a 5 cartas y se realiza una pregunta del estilo ¿Dónde está la carta con el contenido X?. De forma similar a Memory, las cartas son mostradas por un tiempo determinado y posteriormente son volteadas para que el jugador intente recordar su posición.

En la Figura 4.2, se dispuso en la parte superior tres elementos, de izquierda a derecha:

- Un botón para volver al menú principal, el que se debe mantener presionado por un segundo para evitar que el sujeto lo presione accidentalmente y se interrumpa el nivel de juego.
- Un indicador de acción correcta (Status: Correct), o incorrecta (Status: Wrong).
- Contador de juegos realizados.

Por otra parte, la sección inferior se dividió en tres columnas, dejando la columna central con mayor espacio para colocar uno de los tres juegos, mientras que la columna izquierda se deja vacía, y la sección derecha se utiliza para mostrar el tiempo restante para comenzar el nivel de juego.

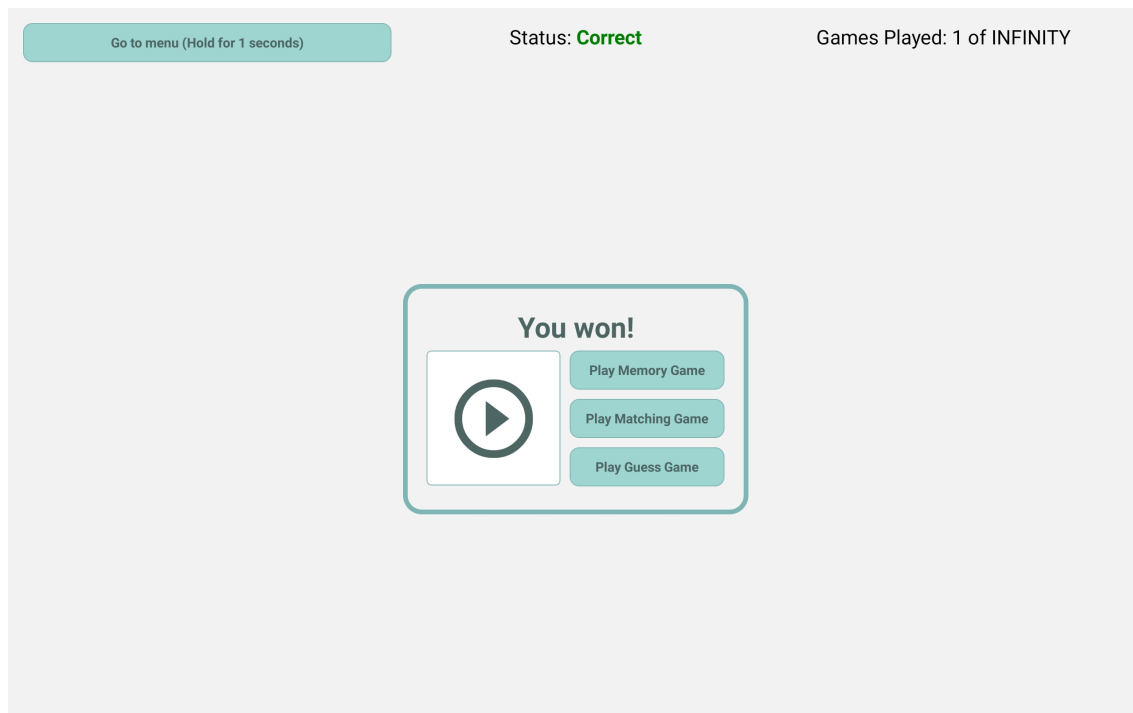


Figura 4.3: Vista de menú una vez finalizado el juego. En la parte central se muestra un botón grande con un ícono para repetir el tipo de juego, y también se colocan 3 botones para cambiar a otro tipo de juego.

Además, como se observa en la imagen 4.3 se agregó un menú que se despliega una vez finalizado el nivel de juego para escoger otro tipo de juego, poniendo énfasis en mantener el tipo seleccionado (Memory, Matching o Guess).

Settings Screen

Como se plasmó recientemente, se implementaron tres tipos de juegos enfocados en las habilidades de memoria y relacionamiento de objetos. El cliente tenía como necesidad poder

parametrizar los niveles de juego, así como también la cantidad de cartas que se mostraran en pantalla, la distribución y la configuración de la retroalimentación.

Figura 4.4: Vista de configuraciones generales. Se muestra la primera pestaña seleccionada para modificar las configuraciones generales, por ejemplo, nombre de guardián, nombre de animal, tipo de juego, cantidad de juegos, tipo de mazo, pausa inicial y pausa entre toque de cartas.

En la parte superior de la Figura 4.4 se colocaron dos botones, uno para volver al menú principal (esquina superior izquierda) y otro para dirigirse a la vista de juegos (esquina superior derecha). Inmediatamente debajo de los botones, se puso un menú para separar los distintos tipos de configuraciones para cada juego, al igual que las configuraciones que se aplican de forma general.

Las primeras configuraciones generales que se aplicaron, sirven para poder registrar tanto el nombre del guardián, así como del animal no humano que está jugando, para lo que se dispuso entradas de tipo texto.

Otro tipo de configuraciones que se añadieron fueron:

1. Tipo de juego inicial (Figura 4.4).
2. Cantidad de juegos por sesión, incluyendo juegos sin límite (Figura 4.4).
3. Conjunto de cartas a utilizar en el juego (Figura 4.4).
4. Tiempo de espera para comenzar o preparar el juego (Figura 4.4).
5. Tiempo de pausa entre cada acción realizada por animal no humano (Figura 4.4).
6. Entrada de tipo Switch (booleano) para activar/desactivar ayuda (Figura 4.5).

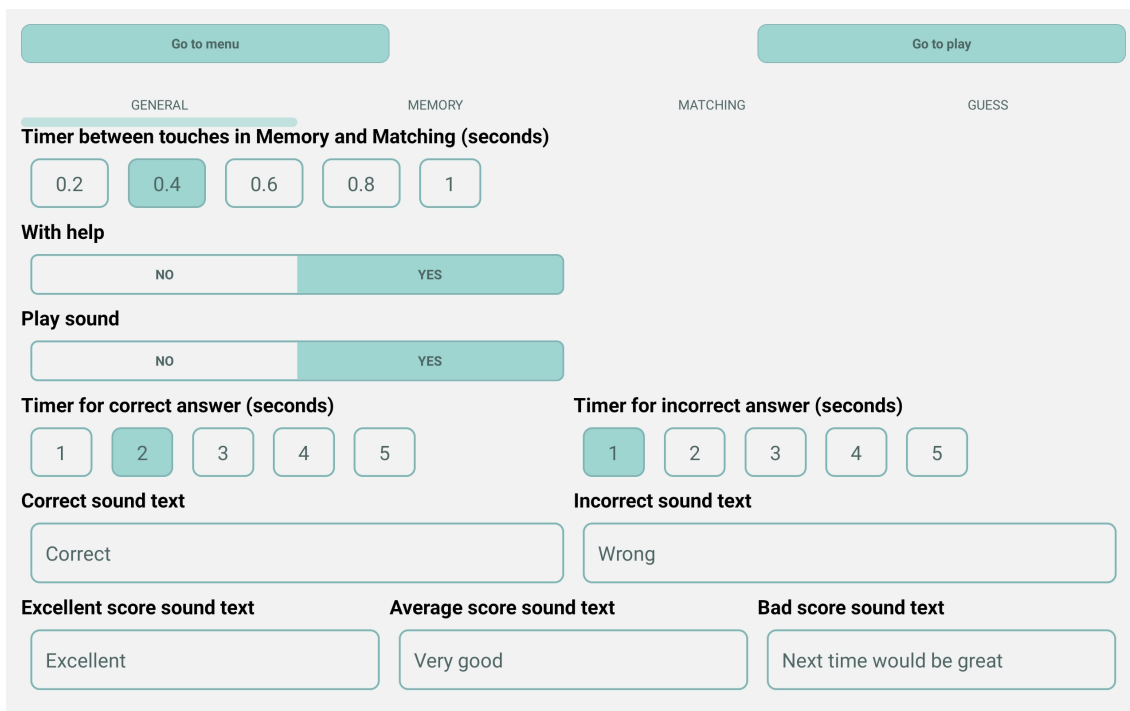


Figura 4.5: Continuación de la vista de configuraciones generales. Desde arriba hacia abajo, pausa entre toque de cartas, seleccionar ayuda, seleccionar reproducción de sonido, pausa en respuesta correcta, pausa en respuesta incorrecta, sonido en respuesta correcta, sonido en respuesta incorrecta, sonido en puntaje excelente, sonido en puntaje promedio, sonido en puntaje deficiente.

7. Entrada de tipo Switch para activar/desactivar sonido de retroalimentación (Figura 4.5).
8. Tiempo de pausa para cuando se realiza acción correcta o incorrecta (Figura 4.5).
9. Entradas de tipo texto cuando se realiza acción correcta o incorrecta, o bien según el desempeño al completar un nivel.

Además, como se puede observar en la Figura (Anexo B) se dispusieron parametrizaciones para cada uno de los juegos, por ejemplo, para elegir el tamaño de la grilla donde se colocan las cartas.

Logs Screen

Durante el transcurso del juego el animal no humano realiza acciones tales como, presionar una carta, realizar una selección correcta o incorrecta, o bien, completar un nivel de juego. Estos eventos son registrados y almacenados localmente para posteriormente ser mostrados en la vista de la Figura 4.6.

Para mantener consistencia a lo largo de las vistas, se ubica un botón para volver al menú principal en la parte superior de la página (indicado con el número 1 en la Figura 4.6.

1 Go to menu

2 Generate file

3 Clear Logs

Datetime	Game	Learner	Guardian	Action	Annotat...	GameN...	Number...	Desk	GridSize	Cards	Help	Sound	PauseC...	PauseIn...	PauseB...
15-12-20 21 16:31: 16.939	Matching	Lorenzo	Erick	Selected card on position (2;1) with value 'cube'		2 of INFINITY	6	Fotos	(2;3)	[[book;pyramid;book];[cube;cube;pyramid]]	Yes	Yes	2	1	0.4
15-12-20 21 16:31: 16.941	Matching	Lorenzo	Erick		Correct pair of cards	2 of INFINITY	6	Fotos	(2;3)	[[book;pyramid;book];[cube;cube;pyramid]]	Yes	Yes	2	1	0.4
15-12-20 21 16:31: 18.952	Matching	Lorenzo	Erick		You won with 6 attempts	2 of INFINITY	6	Fotos	(2;3)	[[book;pyramid;book];[cube;cube;pyramid]]	Yes	Yes	2	1	0.4
15-12-20 21 16:31: 31.731	Guess	Lorenzo	Erick	Selected card on position (1;2) with value 'book'		3 of INFINITY	3	Fotos	(1;5)	[[book;cube;book];[pencil]]	Yes	Yes	2	1	0.4
15-12-20 21 16:31: 31.732	Guess	Lorenzo	Erick		Correct answer to question 'Where is book'	3 of INFINITY	3	Fotos	(1;5)	[[book;cube;book];[pencil]]	Yes	Yes	2	1	0.4
15-12-20 21 16:31:	Guess	Lorenzo	Erick		You won with 1	3 of INFINITY	3	Fotos	(1;5)	[[book;cube;book];[pencil]]	Yes	Yes	2	1	0.4

4

Figura 4.6: Vista de logs, acciones realizadas por animal no humano. Las columnas de cada log son: hora y fecha de acción, juego, animal, guardián, acción, anotación, número de juego, número de cartas, mazo, tamaño de grilla, cartas en pantalla, ayuda, sonido, pausa acción correcta, pausa acción incorrecta, pausa entre toque de cartas.

Posteriormente debajo se sitúa el botón indicado con el número 2 con el texto "Generate file" que acciona la funcionalidad de crear un archivo con extensión csv con los registros. Y en el lado derecho (número 3) se encuentra un botón para limpiar el listado de logs.

Finalmente, en la sección indicada con el número 4, se muestra una tabla donde cada fila representa una acción con información que puede ser utilizada por investigadores, tales como parámetros utilizados para el juego, las cartas seleccionadas y la distribución de cartas en la pantalla.

Desk Screen

La última vista implementada corresponde a la pantalla donde se modifican los mazos de cartas, con lo que se pueden crear nuevos conjuntos para poder acostumbrar y/o enseñar al animal no humano nuevos objetos.

En la Figura 4.7 se agregó un botón para volver al menú inicial (al igual que en las otras vistas) y un botón para agregar un nuevo mazo de cartas. También se despliega el listado de mazos de cartas disponibles, donde cada elemento de la lista es presionable y redirige a la vista mostrada en la Figura 4.8.

En la Figura 4.8 se implementó en la esquina superior izquierda un botón para volver a la vista anterior (Figura 4.7). Además, se implementó una entrada de texto para modificar

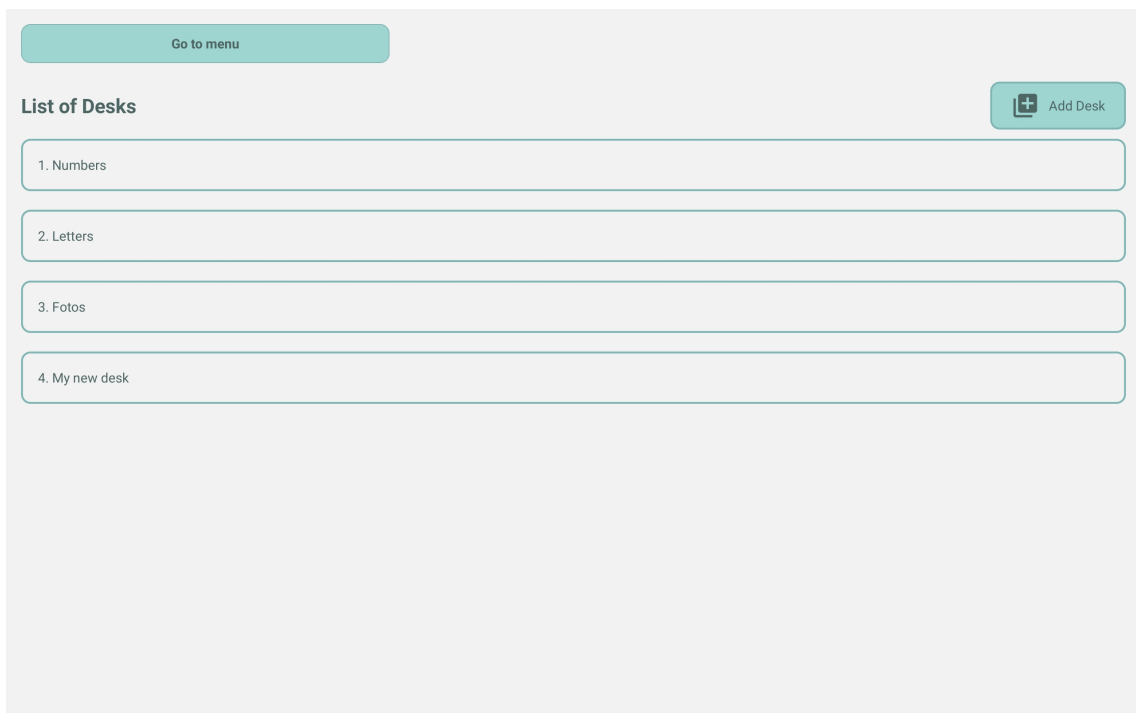


Figura 4.7: Vista del listado de mazos de cartas. Se incorpora botón en el costado derecho para crear nuevo mazo. Cada elemento de la lista corresponde a un conjunto de cartas.

el nombre del conjunto de cartas, y también se agregaron dos botones para poder agregar cartas. Las opciones que se ofrecen son:

- Tomar foto: Al presionar botón se despliega la cámara del dispositivo para realizar una captura de la imagen.
- Seleccionar foto: Botón que al ser presionado abre la galería de imágenes y permite seleccionar una foto que será agregada al conjunto de cartas.

Después de escoger la foto, se despliega un modal (Figura A.4) para modificar el nombre con que será guardada la imagen.

4.3. Funcionalidades

Durante el desarrollo del prototipo se fueron incorporando distintas funcionalidades que agregaron valor al software y mejoraban la experiencia de usuario. Algunas funcionalidades, mencionadas a continuación, estaban contempladas desde un inicio, mientras que otras surgieron a raíz de retroalimentación otorgada por personas voluntarias.

1. Almacenamiento de Logs.
2. Envío de Logs por correo.

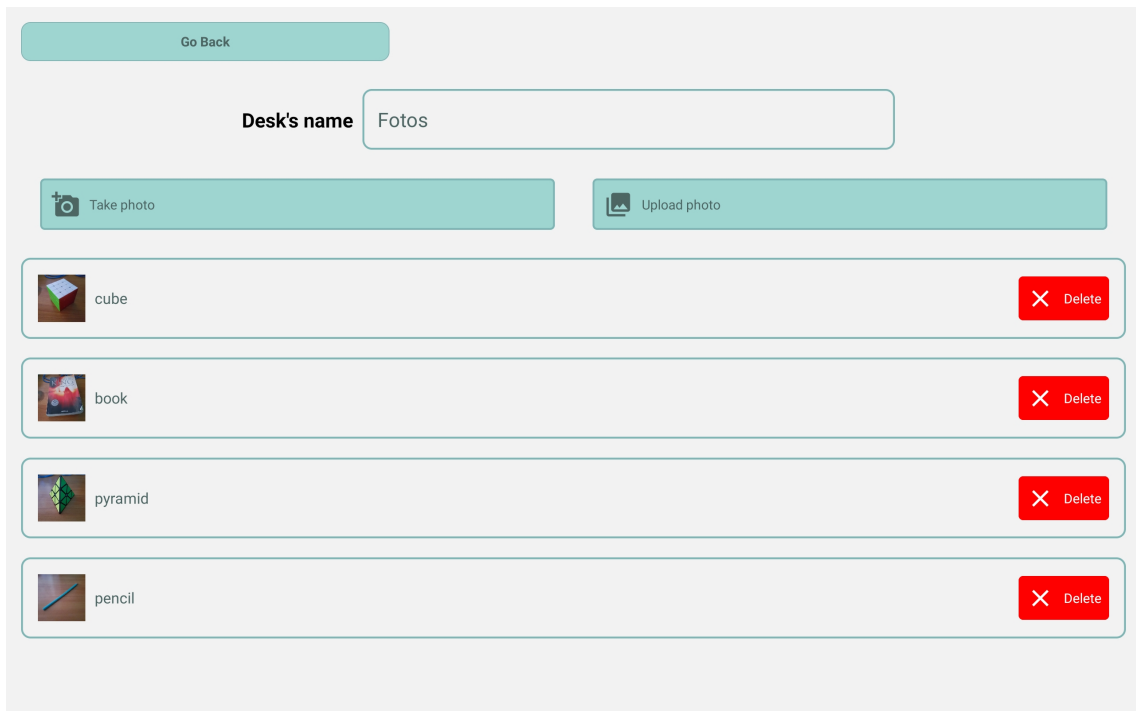


Figura 4.8: Vista del detalle de mazo de cartas determinado. En la parte superior se muestra un campo de texto para insertar nombre del mazo. Inmediatamente después aparecen dos botones, uno para abrir la cámara y otro para abrir la galería. Debajo de los botones se muestra el listado de cartas, cada una con un botón para eliminar la carta del mazo.

3. Agregar fotos para cartas.
4. Parametrización.
5. Incorporación de sonidos.
6. Pausa entre acciones.

Almacenamiento de Logs

El almacenamiento de Logs corresponde a la funcionalidad de mayor utilidad para las personas que se dedican a la investigación, porque esto les permite obtener datos y poder estudiar, extraer información o validar/rechazar hipótesis. Para poder almacenar los registros de cada acción, se pensó en mantener la información en un arreglo donde cada elemento representa una acción. Por otra parte, los juegos implementados requieren de una función `onUpdate`, que es invocada cada vez que el animal no humano interactúa con el juego. Esta función se encarga de organizar la información que será añadida al listado de registros. La información guardada contiene la estructura descrita en la Figura 4.9.

```

{
  datetime: Hora en que se produce un evento (selección de carta).
  game: Tipo de juego seleccionado.
  learner: Nombre del animal no humano.
  guardian: Nombre del guardián.
  action: Carta seleccionada.
  annotations: Información adicional.
  gameNumber: Número de juego actual y límite de juegos.
  numberOfCards: Número de cartas que se despliegan en pantalla.
  desk: Nombre del conjunto de cartas utilizado.
  gridSize: Tamaño de la grilla en que se colocan las cartas.
  cards: Matriz de cartas mostradas en pantalla.
  help: Si está habilitada la opción con ayuda.
  sound: Si está habilitada la opción de retroalimentación auditiva.
  pauseCorrect: Tiempo de pausa cuando una jugada es correcta.
  pauseIncorrect: Tiempo de pausa cuando una jugada es incorrecta.
  pauseBetweenTouches: Tiempo de pausa entre toques de cartas.
}

```

Figura 4.9: Estructura de los registros almacenados. Cada registro se guarda como un objeto, donde cada campo guarda un tipo de información, por ejemplo: hora y fecha, tipo de juego, cartas utilizadas, entre otros.

Envío de Logs por correo

Además de poder recopilar datos de la aplicación, también fue necesaria una funcionalidad que permitiera exportar la información. Para poder generar un archivo y almacenarlo se manejaron dos opciones, la primera consistió en crear un archivo y descargarlo directamente en el dispositivo, y la segunda opción fue crear el archivo y enviarlo por correo.

Para ambas opciones se debe generar primero un archivo con extensión .csv para poder tener la información estructurada. Sin embargo, Expo posee una API que permite enviar correos, otorgando la ventaja de compartir la información con más investigadores y tener acceso desde distintos dispositivos. De esta forma, la opción escogida corresponde al envío de correos.

Para poder generar el archivo se utiliza el pseudódigo mostrado en la Figura 4.10.

Agregar fotos para cartas

Una funcionalidad que permite tener acceso a más tipos de imágenes consiste en poder tomar fotos o subir directamente desde la galería.

Para poder realizar esto, se utiliza una librería de Expo que interactúa con los componentes nativos del dispositivo, tales como, galería y cámara.

```

fields <- Arreglo de nombre de campos de logs.
file <- ""
for log in logs:
    newline <- ""
    for f in fields:
        fixedData <- log[f].replace(",", ";")
        newline <- newline + fixedData + ","
    file <- file + newline + "\n"

```

Figura 4.10: Pseudocódigo que permite generar el archivo de logs. Se itera sobre el arreglo de logs para transformarlo en una cadena de texto uniendo todos los elementos de un mismo registro mediante comas.

Una vez que se selecciona la foto a utilizar, se pide ingresar un nombre para la carta y posteriormente confirmar o cancelar la creación.

Parametrización

La parametrización de la aplicación permite al guardián configurar los niveles de juego y otras componentes de la aplicación. La forma de implementar esta funcionalidad corresponde a un objeto global que contiene todas las variables parametrizables y que se puede acceder desde cualquier componente de la aplicación. Para poder modificar estos valores es necesario ejecutar una función que se encarga de cambiar los valores y actualizar el estado global, con la finalidad de que no ocurran inconsistencias.

La parametrización se divide en cuatro categorías principales:

1. General
2. Memory
3. Matching
4. Guess

Las parámetros configurables en la categoría **General** corresponden a los valores que afectan el funcionamiento general del juego, tales como habilitar ayuda o sonido. En la Figura 4.11, se presenta el detalle de las configuraciones generales.

También existen las configuraciones que se aplican exclusivamente a cada tipo de juego, tal como se muestra en la Figura 4.12.

```

{
  guardian: Nombre del guardián del animal.
  learner: Nombre del animal no humano.
  gameType: Tipo de juego seleccionado.
  gamesPerSession: Cantidad de juegos para una sesión (1 a 5, o infinito)
  desk: Conjunto de cartas a utilizar.
  initialTimer: Temporizador para analizar las cartas antes de comenzar.
  betweenTimer: Tiempo de pausa entre selección de cartas.
  withHelp: Opción para habilitar ayuda.
  withSound: Opción para habilitar retroalimentación auditiva.
  pauseCorrect: Tiempo de pausa cuando se selecciona correctamente.
  pauseIncorrect: Tiempo de pausa cuando se selecciona equivocadamente.
  correctSoundText: Sonido cuando una selección es correcta.
  wrongSoundText: Sonido cuando una selección es incorrecta.
  excellentSoundText: Sonido de nivel completado con excelente rendimiento.
  averageSoundText: Sonido de nivel completado con rendimiento aceptable.
  badSoundText: Sonido de nivel completado con bajo rendimiento.
}

```

Figura 4.11: Configuraciones generales. Se detalla cada campo o parámetro de las configuraciones que se aplican de forma general al juego.

Incorporación de sonidos

Para poder entregar retroalimentación auditiva al animal, es necesario incorporar sonidos a la aplicación. En principio, se pensó en incorporar archivos con extensión .mp4 y reproducirlos cuando ocurren determinados eventos, sin embargo, ante la falta de un servidor para almacenar archivos, se optó por guardar textos que son transformados a sonidos mediante una librería que posee Expo.

De esta forma, existen principalmente cinco categorías de sonidos, que se pueden modificar:

1. Correcto
2. Incorrecto
3. Excelente puntaje
4. Puntaje promedio
5. Mal puntaje

Los primeros dos sonidos (correcto e incorrecto) son llamados cuando se selecciona un par de cartas idénticas, en el caso de Memory y Matching, o una sola carta en el juego Guess. Mientras que los otros tres sonidos se emiten cuando se completa un nivel según el puntaje que se obtiene.

El puntaje se define en base al número de cartas que presiona el jugador (animal no humano), estableciendo rangos para diferenciar un juego óptimo, juego promedio o juego deficiente, considerando que en un nivel se muestran N cartas.

```

Memory: {
    gridSize: Tamaño de grilla para juego Memory.
    numberOfCards: Cantidad de cartas para mostrar en pantalla.
    showGrid: Opción para mostrar separadores de cartas.
}
Matching: {
    gridSize: Tamaño de grilla para juego Matching.
    numberOfCards: Cantidad de cartas para mostrar en pantalla.
    showGrid: Opción para mostrar separadores de cartas.
}
Guess: {
    numberOfCards: Cantidad de opciones posibles.
}

```

Figura 4.12: Configuraciones específicas de cada juego. Se detallan los campos o parámetros de las configuraciones que se aplican a cada tipo de juego.

En el caso del juego Memory y Matching, se establecen los siguientes rangos:

- Juego óptimo es cuando la cantidad de cartas presionadas está en el rango $[N, 1.5 * N]$, considerando que en un inicio observa las N cartas, por lo que se podría determinar la posición de cada par en un solo intento, o bien, permitir que olvide la posición de una de las cartas de cada par, por lo que tiene que revelar tres cartas por cada par.
- Juego promedio es cuando la cantidad de cartas presionadas está en el rango $[1.5 * N, 2.5 * N]$, permitiendo que en promedio se revisen a lo más cinco cartas por cada par que se muestre en pantalla, lo que equivale a equivocarse dos veces por cada par de cartas.
- Juego deficiente es cuando la cantidad de cartas presionadas es mayor a $2.5 * N$, lo que indica que el animal no humano tiene muchos problemas para identificar los pares de cartas.

En el caso del juego Guess, se determinó los siguientes rangos:

- Juego óptimo es cuando el animal no humano identifica la respuesta correcta en el primer intento (dentro de a lo más cinco opciones).
- Juego promedio es cuando el jugador presiona a lo más la mitad de cartas mostradas en el tablero, es decir $N / 2$.
- Juego deficiente es cuando la cantidad de cartas presionadas es mayor a $N / 2$.

Pausa entre acciones

Una funcionalidad que influye directamente en la experiencia del animal no humano es la pausa entre acciones (toques de cartas). Hay ocasiones en que el sujeto realiza toques involuntarios al momento de desplazarse frente a la pantalla de juego, por ejemplo, presionar dos cartas contiguas. Para evitar estas situaciones, después de presionar una carta se agrega un temporizador (configurable por guardián) para que no se pueda presionar ninguna otra carta hasta que el tiempo termine.

Capítulo 5

Evaluación de la Solución

El proceso de evaluación requirió de personas que entrenasen con animales no humanos (aves), por lo que se le pidió a Jennifer Cunha que validara el software implementado, y que respondiera una encuesta con el fin de cuantificar aspectos importantes de la aplicación, tales como la facilidad de uso o la ayuda que le otorga en sus labores.

5.1. Participantes

La evaluación del sistema estuvo principalmente a cargo de la investigadora Jennifer Cunha en conjunto a Ellie, ave de la especie cacatúa, quién se encargó de realizar distintas pruebas y requerir mejoras para una mejor experiencia de usuario.

5.2. Descripción del proceso de evaluación

Para poder evaluar el prototipo, es necesario que el participante y sus animales se familiaricen con el juego, por lo que antes de realizar las preguntas, se entregó acceso al juego una semana antes, entregando una breve descripción de las funcionalidades implementadas.

5.3. Instrumento de evaluación de Usabilidad percibida

En palabras de John Brook [15, 14], la usabilidad de una aplicación depende del contexto en que se utilice, por lo que bajo este escenario Brook desarrolló System Usability Scale (o SUS abreviado), que corresponde a una encuesta de diez preguntas, cada una en escala de Likert de 5 puntos.

Basándose en la escala mencionada recientemente, se elaboraron diez preguntas con puntuación (1) a (5), y una pregunta de desarrollo que permitieran conocer la usabilidad del

sistema. Las preguntas [1-5], fueron pensadas para reflejar la afinidad de cada tipo de usuario con la aplicación, mientras que las preguntas [6-10] fueron hechas para reflejar si las funcionalidades principales se pueden realizar. Las preguntas (traducidas al español, originales en el Apéndice B) son las siguientes:

1. La aplicación es sencilla de usar para los guardianes.
2. La aplicación es sencillas de utilizar para los animales.
3. La aplicación ayuda a enriquecer la vida de los animales.
4. La aplicación es sencilla de usar para los investigadores.
5. Usaré la aplicación con propósitos investigativos.
6. La aplicación permite que los animales jueguen.
7. La aplicación permite que el guardián configure niveles de juego.
8. La aplicación permite que el guardián suba nuevos conjuntos de cartas.
9. La aplicación permite que los investigadores vean los registros.
10. La aplicación permite que los investigadores puedan exportar los registros.
11. (Pregunta abierta) Comentarios, sugerencias o funcionalidades que desearías haber visto.

Cada una de estas preguntas (excepto la última) se debe responder según la escala de Likert para obtener un puntaje asociado:

- **1 puntos:** Totalmente en desacuerdo.
- **2 puntos:** En desacuerdo.
- **3 puntos:** Indiferente.
- **4 puntos:** De acuerdo.
- **5 puntos:** Totalmente de acuerdo.

5.4. Resultados obtenidos

Respecto a la encuesta realizada, los puntajes otorgados por Jennifer Cunha son los siguientes:

1. La aplicación es sencilla de usar para los guardianes. [5]
2. La aplicación es sencillas de utilizar para los animales. [5]

3. La aplicación ayuda a enriquecer la vida de los animales. [4]
4. La aplicación es sencilla de usar para los investigadores. [4]
5. Usaré la aplicación con propósitos investigativos. [3]
6. La aplicación permite que los animales jueguen. [5]
7. La aplicación permite que el guardián configure niveles de juego. [5]
8. La aplicación permite que el guardián suba nuevos conjuntos de cartas. [5]
9. La aplicación permite que los investigadores vean los registros. [5]
10. La aplicación permite que los investigadores puedan exportar los registros. [5]

Lo que entrega una puntuación promedio de **4.6**, considerando que el puntaje más bajo fue de **3** en la pregunta 5.

Mientras que la respuesta (original en el Apéndice C) a la pregunta abierta fue la siguiente: **Creo que los sonidos orientados a los niños (un pop para respuestas correctas, niños animando) pueden aumentar la participación. También me doy cuenta de que puedo configurarlos ahora gracias a sus opciones de personalización. El único motivo por el que no lo usaré para investigación, es porque no es mi área, aunque si considero que está en buena forma para otorgar información valiosa a investigadores.**

Por otra parte, el día 29 de diciembre de 2021 [8], Jennifer Cunha mencionó aspectos positivos relacionados a la experiencia por parte de Ellie (cacatúa):

- Buen juego, 17 rondas de juego.
- Único juego que Ellie juega por sí misma de manera consistente.
- Ellie pidió el juego.
- Juego muy orientado para aves.

Además, Jennifer Cunha propuso hipótesis de investigación que se pueden realizar con la aplicación InCA-Memory. Tales hipótesis pueden ser:

1. Medida de motivación para jugar un video juego, comparando registros de sesiones hechas por humanos, o comparando con y sin recompensas.
2. Medida de lo buena que es como enriquecimiento.
3. Medida de la rapidez con la que el ave es precisa.
4. Diferencias entre animales no humanos y niños.
5. Identificar los elementos que mejoran la experiencia de usuario: ¿por qué Ellie le gusta más que otras aplicaciones? (Identificar qué comportamiento permite medir la tendencia a jugar)

Capítulo 6

Conclusión y Trabajo a Futuro

Para concluir el trabajo hecho, se elaboró un resumen de lo que se logró (Sección 6.1), se realizó una discusión de las cosas que se pudieron realizar de mejor manera (Sección 6.2) y finalmente, se elaboró un listado de tareas que permitan guiar trabajos futuros (Sección 6.3).

6.1. Contribuciones

Animal Computer Interaction es un área emergente en el ámbito de la computación, por lo que es necesaria la creación de herramientas que permitan realizar más investigaciones. En este contexto, se desarrolló un prototipo de juego que permite explorar la capacidad de la memoria de animales no humanos, específicamente aves cacatúas. Tal juego tiene por objetivo permitir que el ave se pueda entretener al mismo tiempo que le permite al guardián o investigador poder configurar niveles de juego o personalizar algunas características de la aplicación con tal de investigar de mejor manera.

Las principales funcionalidades que se implementaron corresponden a la posibilidad de almacenar las acciones realizadas por el animal no humano, poder enviar los datos por correo electrónico, poder crear mazos de cartas y también añadir nuevas cartas utilizando la cámara o galería del dispositivo, y finalmente, añadir sonidos para entregar retroalimentación al animal.

6.2. Lecciones Aprendidas

Tanto a nivel de propuesta y trabajo de memoria, existieron dificultades y hubo puntos débiles al momento de desarrollar.

En un principio, se pensó en desarrollar dos juegos similares, sin embargo, eso se traduce en aumentar al doble el tiempo del desarrollo para funcionalidades que son muy similares, por lo tanto, después se optó por juntar ambas aplicaciones en una sola. Esto se pudo haber

evitado con una mejor planificación de la solución para el problema planteado en un inicio.

La principal dificultad, fue la desorganización al momento de desarrollar, principalmente el hecho de no implementar funcionalidades en periodos prolongados o no seguir el cronograma propuesto inicialmente.

A pesar de tener buenos resultados en la evaluación de la aplicación, hubiese sido mucho mejor contar con más personas o guardianes que validarán la aplicación. De esa forma, se evita el sesgo de un sólo usuario al momento de desarrollar el juego. También se pudo realizar un proceso de evaluación que permitiera medir el tiempo de interacción del usuario con la aplicación, para detectar las partes en que se presentan mayores problemas, o bien, los elementos que son de más ayuda para el animal no humano.

6.3. Trabajo futuro

Aún cuando se implementó la aplicación en casi su totalidad, quedaron algunas correcciones por realizar, mejoras por incorporar y funcionalidades por implementar. En esta línea, se enuncia el listado de tareas por realizar ordenadas por prioridad o importancia junto a su descripción y estimación de tiempo para desarrollar.

La primera tarea a desarrollar corresponde a la implementación de un Backend para la aplicación, que permita almacenar la información de los registros (logs) de forma centralizada, con la debida autorización de los guardianes para que los datos puedan ser compartidos y utilizados por investigadores. Esta tarea también permite que un usuario pueda recuperar sus datos al momento de cambiar dispositivo mediante un mecanismo de inicio de sesión. La implementación de un Backend debería tener un tiempo de desarrollo aproximado de ocho a diez semanas, considerando que se debe desarrollar una API que le permita a la aplicación conectarse al servidor, además de crear una base de datos para almacenar la información y archivos de sonido que pueden ser utilizados al momento de realizar acciones correctas o incorrectas (también se puede considerar la opción de mostrar videos).

Una segunda tarea, corresponde a la creación de un nuevo tipo de juego que permita extender las habilidades de memoria y relacionamiento de objetos. Por ejemplo, un juego en que se deban unir dos elementos distintos pero que representen un mismo objeto. Esto debe tener una duración estimada de cuatro a seis semanas, considerando que se debe pensar en como mostrar al jugador y sea intuitivo para jugar, además de modificar la información de cada foto, ya que dos imágenes distintas deben tener un valor en común para poder relacionarlas.

Una tarea menor consiste en mejorar la forma en que se muestran las parametrizaciones al usuario, y cómo se almacenan los Logs, es decir, analizar si la información está bien estructurada o se deben realizar cambios para que los datos sean más valiosos.

Una funcionalidad adicional, corresponde a poder compartir los conjuntos de cartas utilizados para que otros guardianes o usuarios de la aplicación puedan buscarlos y utilizarlos. Esta funcionalidad se puede desarrollar en un período de dos semanas en caso de que sea

implementado previamente un Backend que permita almacenar imágenes.

Otra mejora que es importante para la aplicación, consiste en parametrizar los límites del puntaje de cada juego, es decir, añadir un parámetro que le permita al guardián definir los rangos para considerar un juego como bueno, regular o deficiente.

Finalmente, una tarea más extensa consiste en incorporar la conexión de la aplicación con un dispositivo externo para realizar la entrega de alimentos al animal no humano automáticamente, es decir, cada vez que el ave realiza una acción correcta, la aplicación debe emitir una señal (vía Bluetooth o Wifi) hacia un dispositivo hecho en Arduino, por ejemplo.

Bibliografía

- [1] *2021 Developer Survey*. <https://insights.stackoverflow.com/survey/2021>, visitado el 2021-11-07.
- [2] *Cordova*. <https://cordova.apache.org/>, visitado el 2022-01-02.
- [3] *Expo*. <https://docs.expo.dev/>, visitado el 2022-01-02.
- [4] *Flutter*. <https://flutter.dev/>, visitado el 2022-01-02.
- [5] *Ionic*. <https://ionicframework.com/>, visitado el 2022-01-02.
- [6] *Native Script*. <https://nativescript.org/>, visitado el 2022-01-02.
- [7] *Personal Communication*. Meeting 26-03-2021.
- [8] *Personal Communication*. Meeting 29-12-2021.
- [9] *React Native*. <https://reactnative.dev/>, visitado el 2022-01-02.
- [10] *Svelte*. <https://svelte.dev/>, visitado el 2022-01-02.
- [11] *Svelte Native*. <https://svelte-native.technology/>, visitado el 2022-01-02.
- [12] *They Can Talk*. <https://www.theycantalk.org/home>, visitado el 2021-04-28.
- [13] *Xamarin*. <https://dotnet.microsoft.com/en-us/apps/xamarin>, visitado el 2022-01-02.
- [14] Brooke, John: *SUS: a retrospective*. *Journal of usability studies*, 8(2):29–40, 2013.
- [15] Brooke, John y cols.: *SUS-A quick and dirty usability scale*. *Usability evaluation in industry*, 189(194):4–7, 1996.
- [16] Corless, Robert M., David J. Jeffrey y Donald E. Knuth: *A Sequence of Series for the Lambert W Function*. En *International Symposium on Symbolic and Algebraic Computation, ISSAC '97, Maui, Hawaii, USA, July 21–23, 1997*, páginas 197–204, 1997.
- [17] Cunha, Jennifer: *My Reading Pets*. <https://myreadingpets.com/>, visitado el 2021-04-28.
- [18] Cunha, Jennifer: *Enrichment Through Learning*. En *The International Association of Avian Trainers and Educators (IAATE)*, 2020.

- [19] Cunha, Jennifer y Carlie Rhoads: *Use of a Tablet-Based Communication Board and Subsequent Choice and Behavioral Correspondences in a Goffin's Cockatoo (Cacatua Goffiana)*. Association for Computing Machinery, New York, NY, USA, 2020, ISBN 9781450375740. <https://doi.org/10.1145/3446002.3446063>.
- [20] Gupfinger, Reinhard y Martin Kaltenbrunner: *Animal-Centred Sonic Interaction Design: Musical Instruments and Interfaces for Grey Parrots*. En *Proceedings of the Sixth International Conference on Animal-Computer Interaction*, páginas 1–11, 2019.
- [21] Hunger, Christina: *Hunger for words*. <https://www.hungerforwords.com/>, visitado el 2021-04-28.
- [22] Jennifer Cunha, Susan Clubb: *Advancing Communication with Birds: Can They Learn to Read?* En *Proceedings of the Conference "Animal Computer Interaction" (ACI)*, 2018.
- [23] Joshi, Ankur, Saket Kale, Satish Chandel y D Kumar Pal: *Likert scale: Explored and explained*. British journal of applied science & technology, 7(4):396, 2015.
- [24] Mancini, Clara: *Animal-Computer Interaction (ACI): a manifesto*. Interactions, 18(4):69–73, 2011.
- [25] Mancini, Clara: *Towards an animal-centred ethics for Animal-Computer Interaction*. International Journal of Human-Computer Studies, 98:221–233, 2017.
- [26] McLeod, Saul: *Skinner-operant conditioning*. 2007.
- [27] Newman, M. H. A. y Alan M. Turing: *A Formal Theorem in Church's Theory of Types*. J. Symb. Log., 7(1):28–33, 1942.
- [28] Overflow, Stack: *Stack Overflow Developer Survey 2020*. <https://insights.stackoverflow.com/survey/2020#technology-programming-scripting-and-markup-languages-professional-developers>, visitado el 2021-06-02.
- [29] Shettleworth, Sara J: *Memory in food-hoarding birds*. Scientific American, 248(3):102–111, 1983.
- [30] Tedore, Cynthia y Dan Eric Nilsson: *Avian UV vision enhances leaf surface contrasts in forest environments*. Nature communications, 10(1):1–12, 2019.
- [31] Turing, Alan M.: *Systems of Logic Based on Ordinals*. Tesis de Doctorado, Princeton University, NJ, USA, 1938.
- [32] Washburn, DA: *The four Cs of psychological wellbeing: lessons from three decades of computer-based environmental enrichment*. Animal Behavior and Cognition, 2(3):218–232, 2015.

ANEXOS

Anexo A

Vistas de la aplicación



Figura A.1: Configuraciones específicas para juego Memory. Se muestra la segunda pestaña (Memory) seleccionada. Las configuraciones de arriba hacia abajo son, tamaño de grilla, mostrar grilla, máxima cantidad de cartas en pantalla.

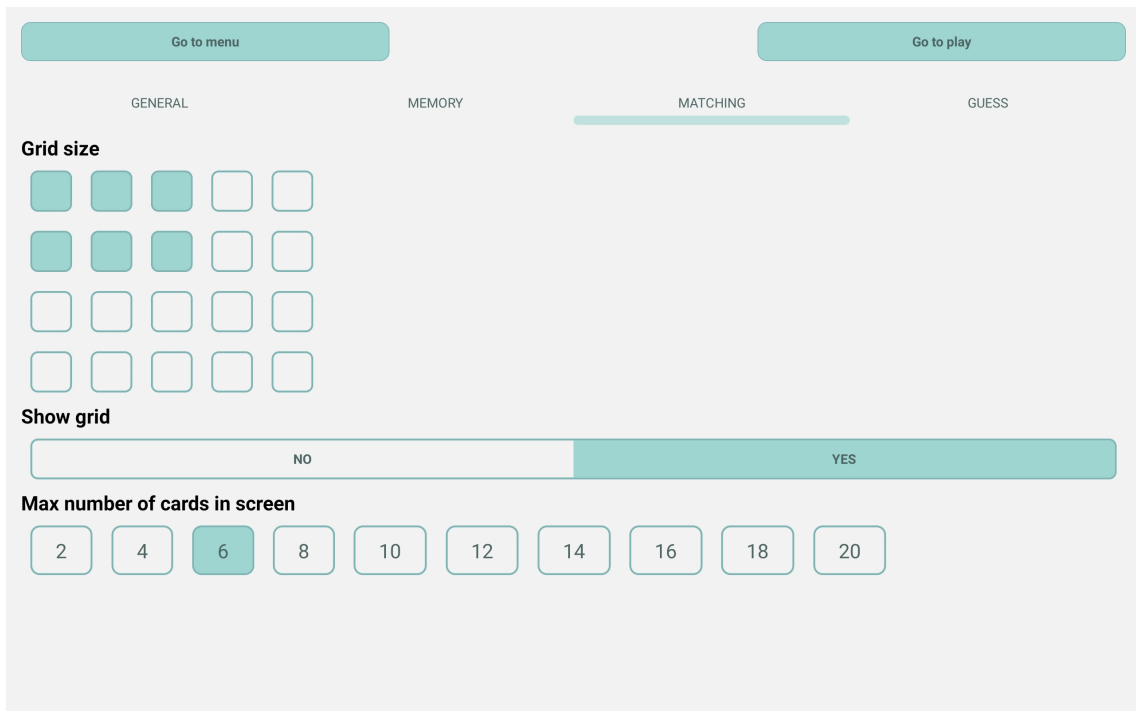


Figura A.2: Configuraciones específicas para juego Matching. Se muestra la tercera pestaña (Matching) seleccionada. Las configuraciones de arriba hacia abajo son, tamaño de grilla, mostrar grilla, máxima cantidad de cartas en pantalla.

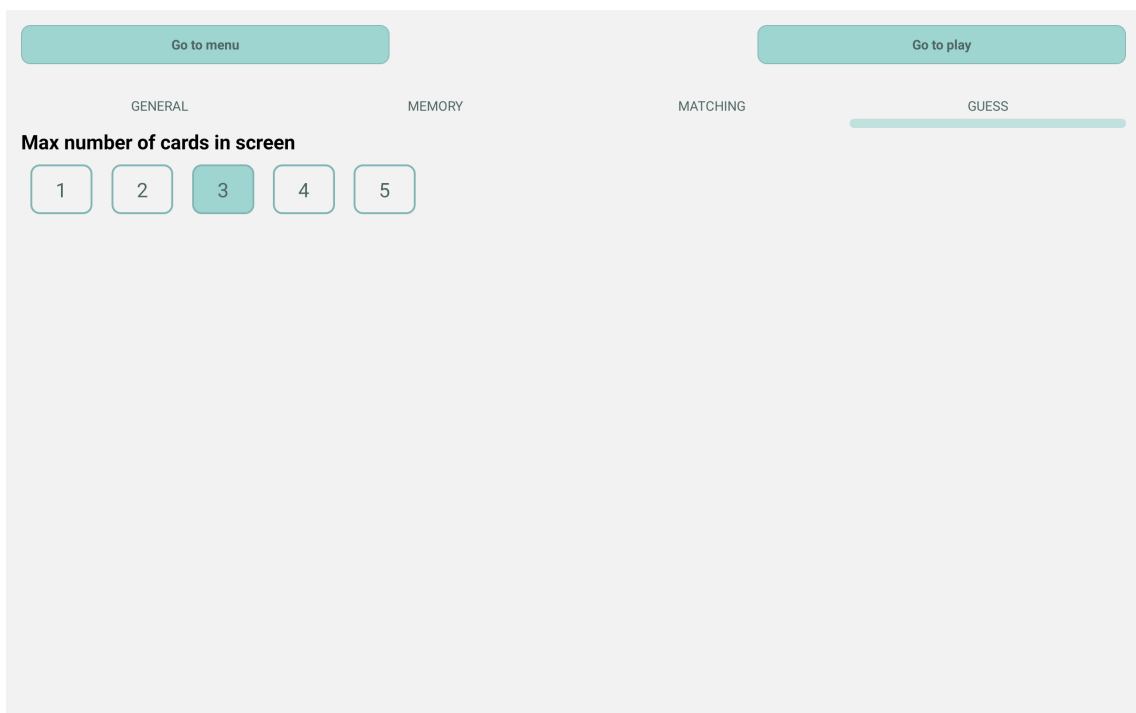


Figura A.3: Configuraciones específicas para juego Guess. Se muestra la cuarta pestaña (Guess) seleccionada. La configuración posible es máxima cantidad de cartas en pantalla.

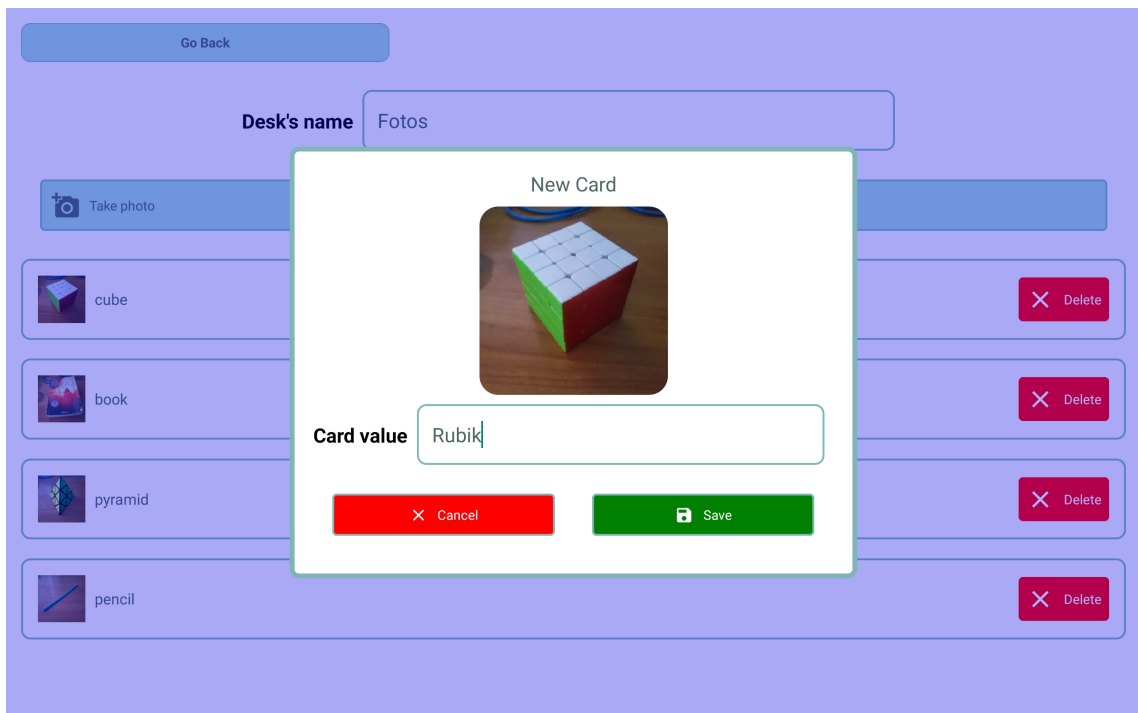


Figura A.4: Modal de confirmación para agregar carta. En el centro de la pantalla se muestra la imagen seleccionada y debajo se presenta un campo de texto para ingresar el nombre. Finalmente, se colocan dos botones, uno para cancelar la creación y otro para confirmar los datos de la carta.

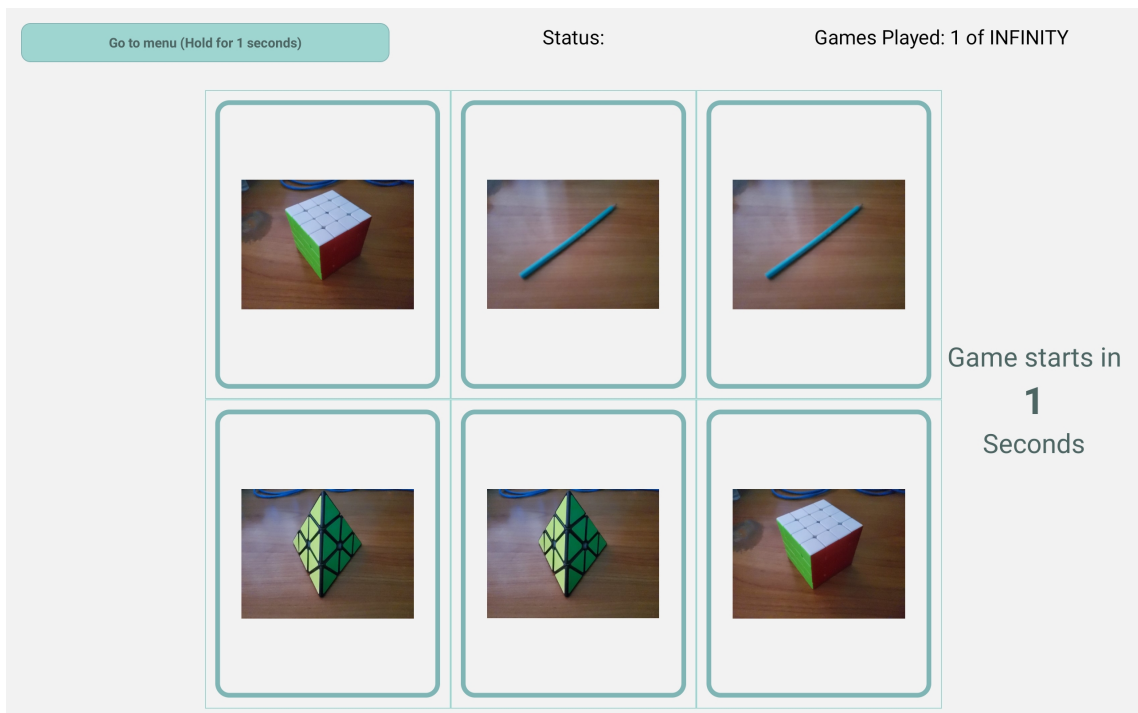


Figura A.5: Vista de inicio de juego Memory. En la parte superior se ubica un botón para volver al menú principal, y también se muestra información general del juego. En la parte central se muestran las distintas cartas, mientras que en el costado derecho se presenta el temporizador para iniciar el nivel.

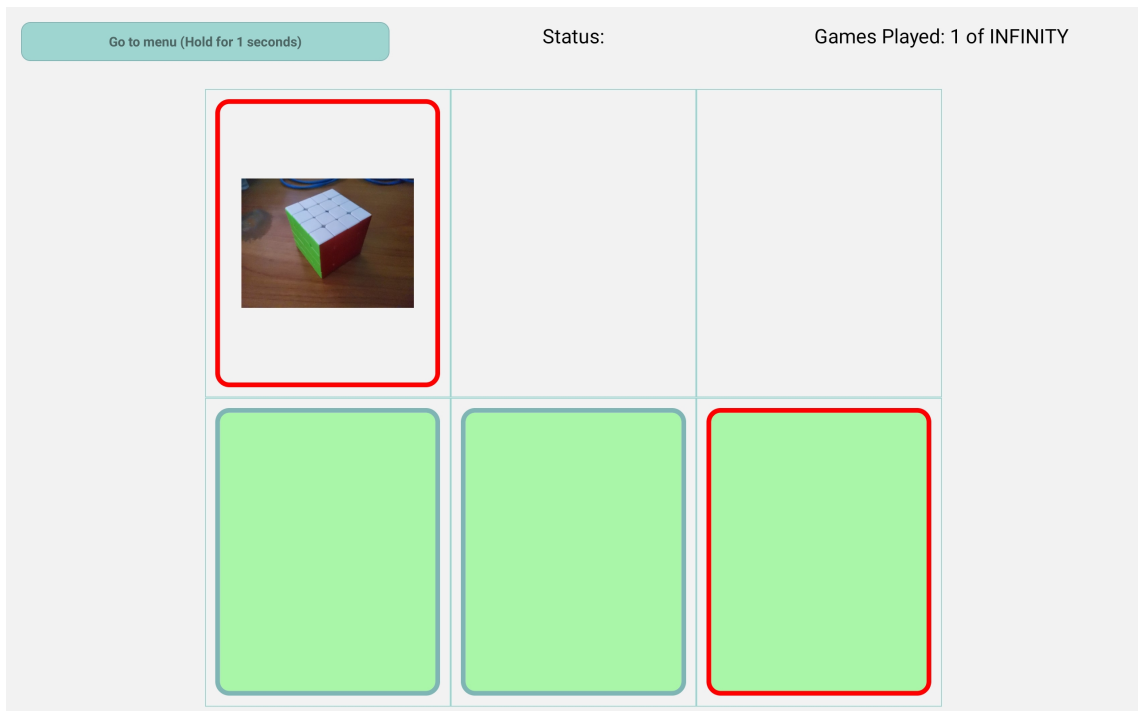


Figura A.6: Vista de juego Memory con la ayuda activada. En la parte central se muestran las distintas cartas escondidas (representadas con color verde). La ayuda se representa con el borde rojo, que es activada una vez que se revela una carta.

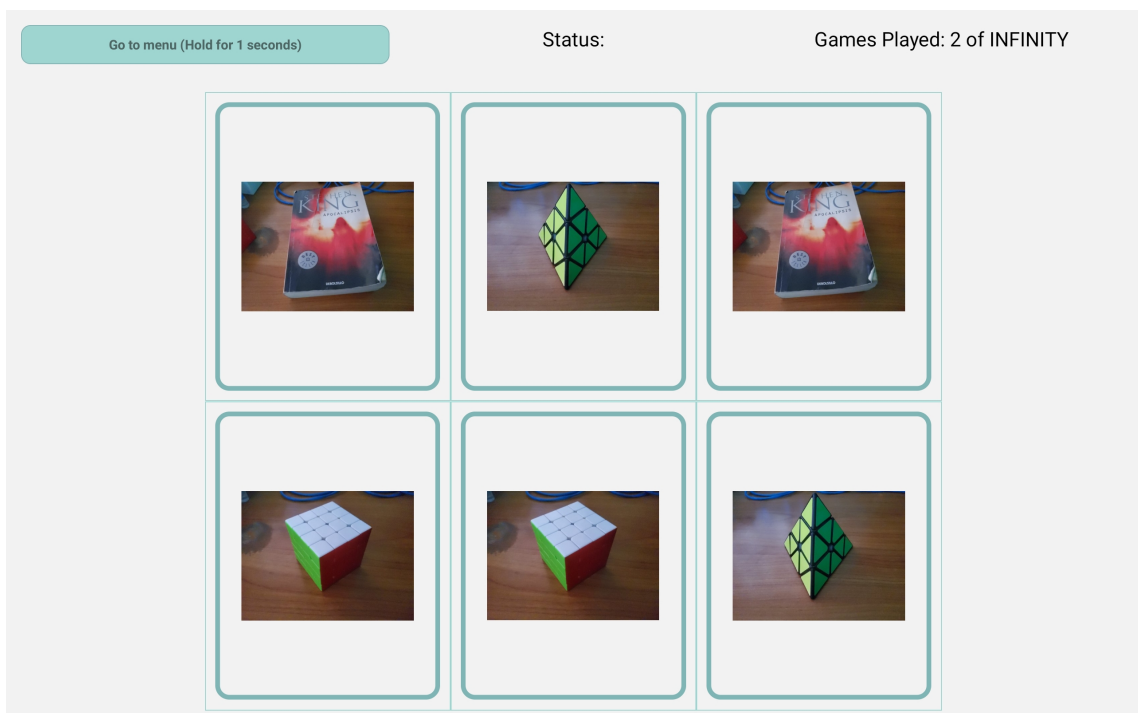


Figura A.7: Vista de inicio de juego Matching. Se muestran las cartas en la parte central de la pantalla, de forma similar al juego Memory.

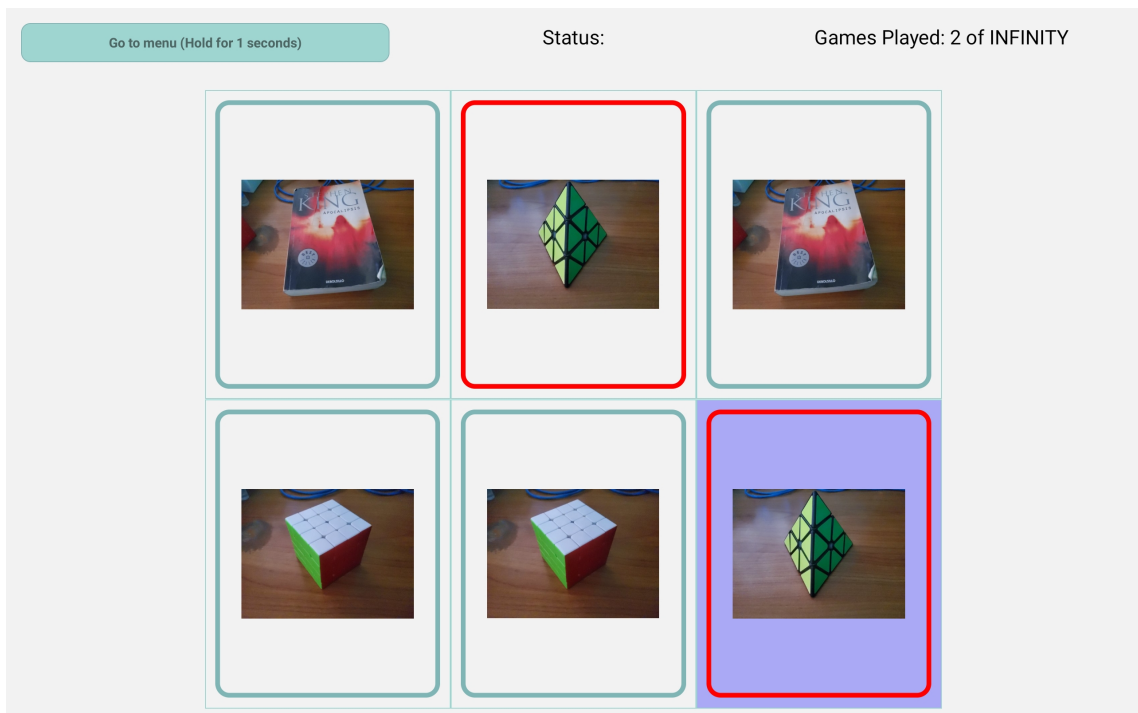


Figura A.8: Vista de juego Matching con la ayuda activada. En la parte central se muestran todas las cartas reveladas. La ayuda se representa con el borde rojo, y se activa una vez que se presiona una carta (carta con tonalidad azul).

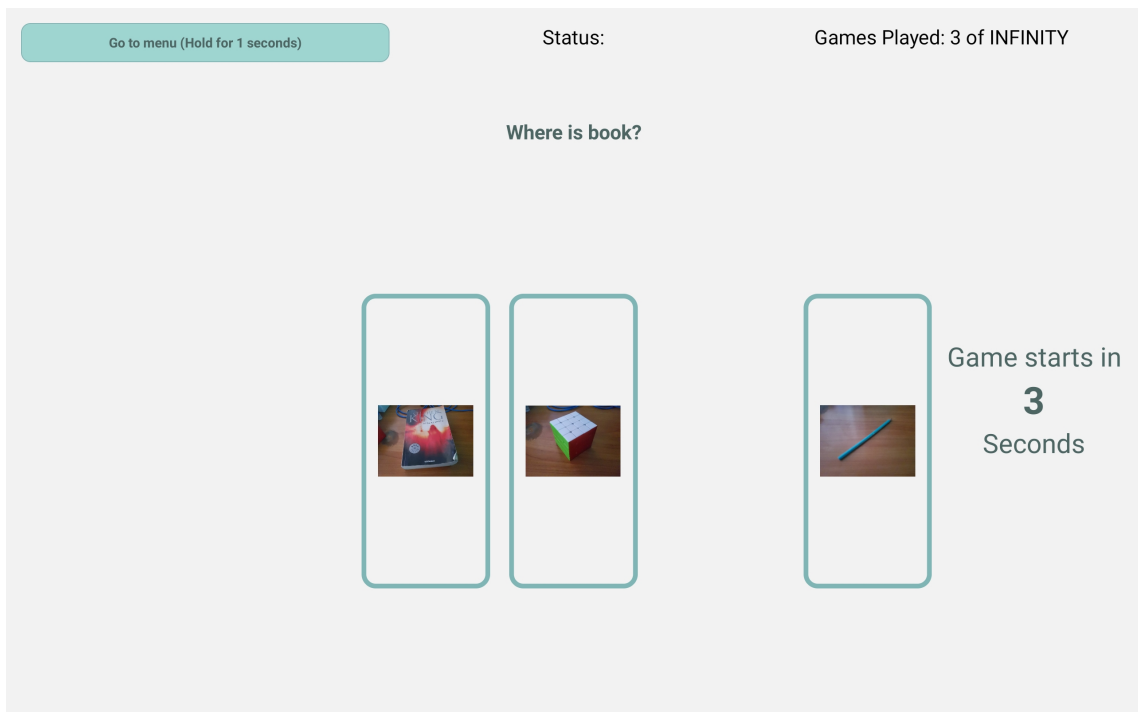


Figura A.9: Vista de inicio de juego Guess. En la parte superior se muestra botón para volver al menú principal, y también información general del juego. Posteriormente se muestra una pregunta para seleccionar una determinada carta.

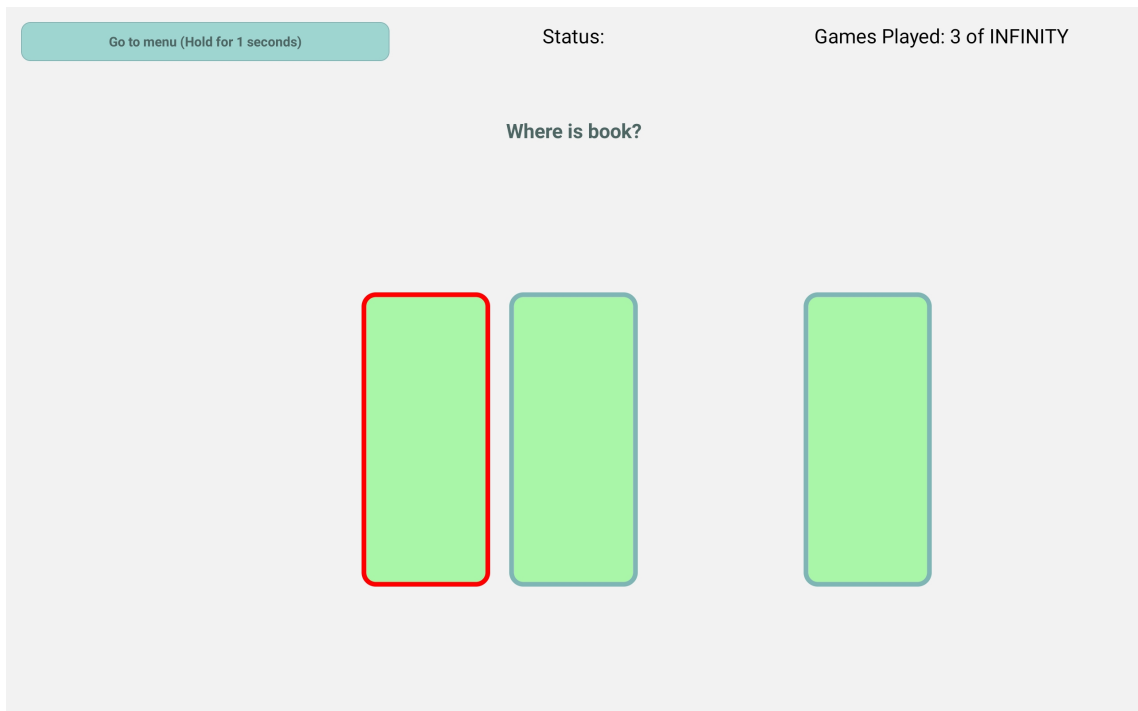


Figura A.10: Vista de juego guess con la ayuda activada. En la parte central se muestran las cartas cubiertas, y la ayuda se representa con el borde rojo.

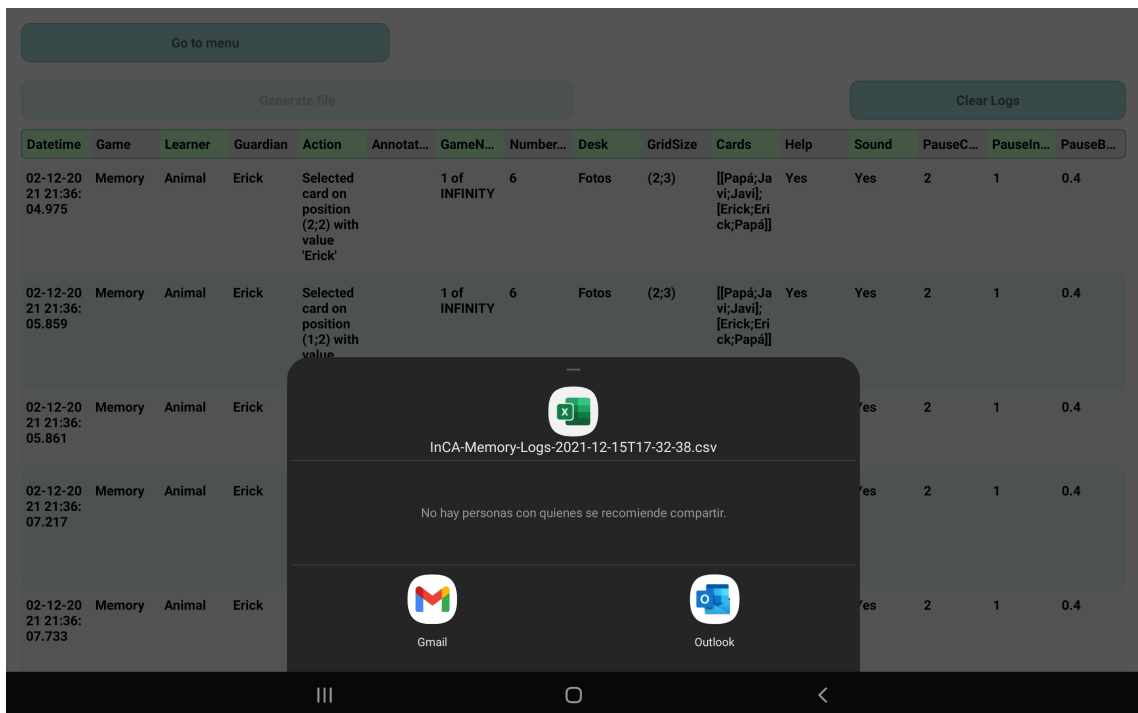


Figura A.11: Modal de envío de correo. Al momento de presionar botón “Generate file”, se genera archivo y se le pide al dispositivo que lo envíe a través de correo electrónico, por lo que el dispositivo abre la ventana para escoger la aplicación de mensajería.

Anexo B

Cuestionario de la evaluación final

1. The application was easy to use for the guardian.
2. The application was easy to use for the animals.
3. The application helps the life enrichment of animals.
4. The application was easy to use for the investigators.
5. I will use the application for investigation purposes
6. The application allows the animal to play.
7. The application allows the guardian to set up game levels.
8. The application allows the guardian to upload new card desks.
9. The application allows the investigator to see logs.
10. The application allows the investigator to export logs.

Anexo C

Respuesta original a pregunta abierta

–Future directions - I think kid-oriented sounds (a pop for correct answers, children cheering) may increase engagement. I also realize I can set these now thanks to your customization options!

–The only reason it's not going to be used for research by me is because that's not my area of research. I do think it's in good shape for providing valuable information for a researcher