



UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO

---

FACULTAD DE CIENCIAS

ACOLI VIDEOJUEGO SERIO PARA REHABILITACIÓN  
DEL SÍNDROME DE HOMBRO CONGELADO

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Licenciado en Ciencias de la Computación

PRESENTA:

Fernando Almazán Cantoral

TUTOR

M. en I. Carlos Ricardo Cruz Mendoza





Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

1. Datos del alumno

Almazán  
Cantoral  
Fernando  
56 77 75 18  
Universidad Nacional Autónoma de México  
Facultad de Ciencias  
Ciencias de la computación  
308012505

2. Datos del tutor

M. en I.  
Carlos Ricardo  
Cruz  
Mendoza

3. Datos del sinodal 1

Mat.  
María Concepción Ana Luisa Solís  
González  
Cosío

4. Datos del sinodal 2

Dr.  
Gustavo  
De la Cruz  
Martínez

3. Datos del sinodal 3

Dra.  
Hanna  
Jadwiga  
Oktaba

6. Datos del sinodal 4

M. en C.  
Edgar Omar  
Cebolledo  
Gutiérrez

7. Datos del trabajo escrito

Acoli videojuego serio para rehabilitación del síndrome de hombro congelado  
79 p  
2018

### **Agradecimientos**

Agradezco a mis padres por haberme traído a este mundo y darme todo lo necesario para poder cumplir este gran logro de mi vida; a mi madre por enseñarme que no hay nada más valioso que el conocimiento, ya que es lo único que nunca nadie podrá quitarnos; a mi padre por todo el apoyo que siempre me brindó y por enseñarme, a través de su ejemplo, la importancia de ayudar a otros; a mis hermanos por ser guías en mi formación, tanto personal como académica; a mis amigos que han sido como mi segunda familia durante toda mi vida; y a mi novia que ha sido un gran apoyo para mí este último año. De igual forma, quiero agradecer a mi tutor, el M. en I. Carlos Ricardo Cruz Mendoza, por la paciencia y dedicación que puso en este proyecto, así como todos los conocimientos que me transmitió a lo largo de este proyecto.

Del mismo modo, expreso mi agradecimiento a los sinodales por ser parte del jurado del presente proyecto, por sus observaciones y comentarios que ayudaron a mejorar esta tesis, y por todo el conocimiento que me brindaron cuando fui su alumno.

### **Dedicatoria**

Dedico este trabajo a mi padre Marco Antonio Almazán Aponte, que donde quiera que esté, espero se sienta orgulloso de mí.



# Índice general

<b>1. Introducción</b>	<b>7</b>
1.1. Antecedentes tecnológicos	8
1.2. Antecedentes médicos	9
1.3. Solución propuesta	9
<b>2. Juegos serios</b>	<b>11</b>
2.1. Definición y características	11
2.2. Usos	12
2.3. Aplicación en rehabilitación	12
2.4. Ventajas	16
2.5. Proceso de desarrollo	17
2.5.1. Proceso de desarrollo de un Juego Serio	17
2.5.2. Proceso de desarrollo utilizado en IHC	18
2.5.3. IHC y Desarrollos tradicionales	20
<b>3. Teoría de Flujo</b>	<b>23</b>
3.1. Definición y características	23
3.1.1. ¿Qué es el flujo?	23
3.1.2. Historia de la Teoría del Flujo	24
3.1.3. Fomento de un estado de flujo	25
3.1.4. Medición del flujo	26
3.2. Usos y aplicaciones	27
3.3. El Flujo en los juegos	27
<b>4. Caso de Estudio: Macoli y el videojuego serio Acoli</b>	<b>29</b>
4.1. Análisis	29
4.1.1. Needfinding	29
4.1.2. Análisis de requerimientos	31
4.2. Diseño	33
4.2.1. Diagramas UML	33
4.2.2. Temática del juego	35
4.2.3. Prototipado	37
4.3. Implementación	47
4.3.1. Dispositivo de control	47
4.4. Experimento y Pruebas	49
4.4.1. Proceso de evaluación	49
4.4.2. Resultados	50
<b>Conclusiones y trabajo a futuro</b>	<b>55</b>
<b>Anexo A</b>	<b>56</b>
<b>Anexo B</b>	<b>70</b>
Antecedentes técnicos	71

<b>Anexo C</b>	<b>73</b>
<b>Referencias</b>	<b>75</b>
<b>Bibliografía</b>	<b>78</b>

# Capítulo 1

## Introducción

En esta tesis se presenta el desarrollo de un videojuego serio, llamado Acoli<sup>1</sup>, el cual brinde apoyo en las terapias de rehabilitación que realizan pacientes que sufren de síndrome de **Hombro congelado**<sup>2</sup>. El desarrollo surge debido a una problemática existente en las terapias de rehabilitación que se realizan actualmente en el Instituto Nacional de Rehabilitación (INR), la cual consiste en que la mayoría de los ejercicios que se realizan durante la terapia son repetitivos, lo que provoca que las terapias lleguen a ser tediosas y desmotivacionales, de esta manera, los pacientes después de múltiples sesiones optan por no asistir, dejando su tratamiento inconcluso y no consiguiendo una adecuada recuperación.

El propósito de este proyecto es el de brindarle a los pacientes una alternativa en el modo de realizar sus terapias, buscando que éstas sean más entretenidas y logren captar la atención de los pacientes, con el fin de disminuir la deserción por parte de los mismos.

Es importante mencionar que Acoli no funciona de manera independiente, ya que forma parte de un proyecto más grande llamado Macoli<sup>3</sup>, el cual es una plataforma tecnológica para el seguimiento y tratamiento de pacientes en una terapia física/mecanoterapia, que mediante objetivos lúdicos y refuerzos objetivos mejora la motivación del paciente y por lo tanto la motivación en la realización de ejercicios de terapia física/mecanoterapia. Acoli juega un papel importante dentro de Macoli, puesto que es la herramienta que se encarga de brindar el aspecto lúdico a las terapias. Macoli, por otra parte, permite a los terapeutas llevar un control de sus pacientes y de las sesiones de rehabilitación que cada paciente realiza, además de permitir visualizar los avances que se van teniendo conforme las sesiones avanzan. Es importante resaltar, que para este proyecto no profundizaremos en el desarrollo ni los componentes de Macoli, ya que esa información forma parte de otro proyecto.

Acoli cuenta con dos componentes, un componente físico (hardware) y otro no físico (software). El componente físico consta de una escalera de aluminio, similar a la escalera de pared, de siete escalones, que se usa en el INR [Figura 1.1], con la diferencia de que ésta se sostiene sobre una base de aluminio que permite su portabilidad. Al igual que en la [Figura 1.1] se cuenta con un tubo, en nuestro caso hecho



Figura 1.1: Escalera de pared

<sup>1</sup>Hombro en náhuatl

<sup>2</sup>También llamado capsulitis adhesiva

<sup>3</sup>Brazo en náhuatl

---

de PVC, similar en tamaño y peso al que se usa en las sesiones de terapia ocupacional del INR. Por otra parte, el componente de software consta de un conjunto de programas, escritos en lenguaje C#, que corren sobre el motor de videojuegos multiplataforma Unity3D.

Cabe mencionar que con Acoli no se pretende que se reemplacen las terapias convencionales, sino ser un complemento de las mismas.

Esta tesis presenta cada uno de los pasos en el desarrollo de Acoli, así como los aspectos teóricos y médicos en los que está fundado, para poder brindar una solución a la problemática mencionada. Con el fin de alcanzar tal objetivo fue necesario, por un lado, hacer uso de la teoría de flujo para fomentar una mejor experiencia en el usuario, y por el otro utilizar un enfoque basado en Interacción Humano-Computadora (IHC), para lo correspondiente al diseño y desarrollo; todo esto haciendo uso de metodologías tradicionales de software.

## 1.1. Antecedentes tecnológicos

A lo largo de los años se han desarrollado múltiples sistemas de rehabilitación, que han sido de ayuda en el ámbito terapéutico. Entre los más destacados están: el sistema Rutgers ARM II y el sistema Armeo, ambos enfocados en terapias de pacientes con problemas en extremidades superiores, que además combinan estructuras físicas de soporte con entornos de realidad virtual, como las que se muestran en la [Figura 1.2], que permiten realizar ejercicios terapéuticos.



(a)

Figura 1.2: Estructuras físicas de soporte (ARMEO, 2009)

Ambos sistemas hacen uso de videojuegos para llevar a cabo los diferentes ejercicios seleccionados, dependiendo de la extremidad que se quiera rehabilitar. Para la realización de los ejercicios, se realizan tareas específicas o juegos que se muestran a través de la pantalla, con el propósito de que la terapia resulte más divertida. Los dos sistemas cuentan con múltiples juegos en los que apoyan sus terapias, los cuales se describen con mayor detalle en el siguiente capítulo.

---

## 1.2. Antecedentes médicos

El hombro es un complejo arreglo de huesos, articulaciones, nervios y músculos que permiten la flexibilidad o rango de movimiento (ROM, por sus siglas en inglés) de la extremidad superior o brazo. Es de gran importancia ya que permite la realización de múltiples actividades de la vida cotidiana, como el trabajo, la educación, el ocio, la realización de diversos deportes, entre otras.

El hombro congelado o capsulitis adhesiva *“es un trastorno de la articulación caracterizado por dolor y limitación de la movilidad, que ocurre generalmente como consecuencia de un traumatismo o una inmovilización prolongada”* (HOMBRO CONGELADO, p. 1); las personas más propensas a sufrir este padecimiento son aquellas con edades entre 40 y 60 años, personas con diabetes, parkinson o personas que sufrieron una fractura o traumatismo de la extremidad superior.

Los terapeutas ocupacionales utilizan diversos tipos de tratamientos para los distintos diagnósticos en el hombro. Estos tratamientos van desde actividades para mejorar el rango de movimiento articular, hasta terapias enfocadas en objetivos y roles específicos para el usuario (HEYDE,2011), pasando por métodos de calor, fármacos, electroterapia, etc. Todos estos enfoques han demostrado su eficacia en el tratamiento de capsulitis adhesiva (HEYDE,2011), pero el uso de nuevas tecnologías beneficiaría sustancialmente la forma de tratar este padecimiento, como puede verse en los resultados obtenidos en en (SVEISTRUP,2003), quien hace uso de la realidad virtual como parte de las terapias del síndrome de hombro congelado. La incorporación de nuevas tecnologías no solo fortalece el campo de la terapia ocupacional, sino también podría contribuir en su longevidad y fuerza.

Una sesión de terapia tradicional, consta en ejercicios de movilidad pasiva, ejercicios terapéuticos estándar (movimientos pendulares de Codman, ejercicios de polea, ejercicios active ROM usando un palo de pvc, estiramientos del hombro contra la pared, entre otros) y ejercicios de flexión. Con la inclusión de Acoli, se reemplazaron algunos de los ejercicios de flexión tradicionales, necesarios en un programa de rehabilitación tradicional, por el ejercicio seleccionado y que se plasmará en el videojuego.

Actualmente son pocos los sistemas que se enfocan en el ámbito terapéutico, además de tener precios muy elevados, lo cual dificulta su adquisición. De igual forma, no existe ningún sistema que se enfoque en el tratamiento de la capsulitis adhesiva, por lo que, contar con un desarrollo tecnológico como Acoli, que garantice una sana y entretenida recuperación en los pacientes y que además tenga un impacto positivo en la forma, calidad y tiempos de atención, y por lo tanto, en la calidad de vida del paciente en rehabilitación, sería de gran ayuda.

## 1.3. Solución propuesta

Con el fin de alcanzar el propósito de esta tesis se realizó una investigación sobre las terapias de rehabilitación para personas con lesiones en el hombro, que se aplican actualmente en el INR, además de un estudio de dispositivos que ya hacen uso de herramientas tecnológicas y de videojuegos como apoyo a la rehabilitación, y una investigación acerca de la experiencia de usuario óptima y el flujo aplicado a juegos serios, para abordar el problema del tedio y aburrimiento en las terapias convencionales. Con base en estos estudios se pensó en la dinámica y diseño del juego serio, buscando reproducir los ejercicios que se realizan actualmente en las terapias, en un ambiente virtual de forma segura para los pacientes.

Una vez desarrollada la dinámica del juego, se realizó un módulo de calibración, que registre los rangos de movilidad del hombro que tiene el paciente, para adaptar los escenarios y dinámicas del juego a estos datos; evitando así que el paciente exceda sus límites y llegue a sentir dolor. Todos los datos obtenidos en la fase de calibración y durante el uso del videojuego, son almacenados en una base de datos, con el fin de brindarle al terapeuta información histórica acerca de los avances del paciente en cada sesión.

Otro aspecto considerado, es el modo en que se enseña al paciente a utilizar los controles para poder jugar, para lo cual se creó una fase de entrenamiento, con la cual el paciente se familiarice con los controles.

---

Por último, al final de cada sesión se registran las puntuaciones obtenidas y se guardan en una base de datos, permitiendo al paciente ver sus avances obtenidos en cada sesión, con el fin de alentarlos a querer seguir realizando sus terapias.

La organización de esta tesis es la siguiente: en el capítulo 2 se introduce el concepto de juego serio, el padecimiento del hombro congelado, y los diferentes procesos de desarrollo de un juego serio. En el capítulo 3 se contempla toda la parte de la Teoría de Flujo; qué es, cómo surgió dicha teoría y como se entra en un estado de flujo, además de sus usos y aplicaciones, y su utilidad en el desarrollo de Acoli. En el capítulo 4 se explica la implementación del juego serio, desde el análisis de requerimientos hasta la construcción del mismo. Por último en el capítulo 5 se mostrarán las conclusiones obtenidas y el trabajo a futuro del proyecto.

## Capítulo 2

# Juegos serios

En este capítulo se introduce el concepto de juegos serios, su definición, sus características principales, algunos de sus usos, los beneficios que se tienen al usarlos y algunos ejemplos, con base en los sistemas Rutgers ARM II y Armeo.

### 2.1. Definición y características

Entender la naturaleza de lo que es un juego serio puede llegar a ser complicado, ya que comúnmente asociamos la palabra 'juego' con realizar una actividad que nos genere entretenimiento y diversión, en ocasiones en escenarios totalmente fantásticos. Por otro lado, el término 'serio' se asocia con algo que se caracteriza por ser importante, formal o de gravedad, que no da cabida a divertir o entretener, y que generalmente se asocia con cuestiones de la vida real. Por lo que llega a ser difícil imaginar un juego que cumpla con estas características.

Según (CHIPIA,2011) un juego serio, se puede definir como: "*un ejercicio recreativo basado en escenarios reales, donde se asume un rol en un mundo real o virtual, para obtener un aprendizaje, dirigido a una gran variedad de público*"(p.5).

Por otra parte (CHIPIA,2011), define un juego serio como: "*una prueba mental, llevada a cabo frente a una computadora de acuerdo con un conjunto de reglas específicas, que usa la diversión como modo de formación, con objetivos en el ámbito de la educación, sanidad, política pública y comunicación estratégica*" (p.4). Analizando ambas definiciones, podemos encontrar algunas de las características principales que debe contener un juego serio, algunas de estas características son las siguientes:

- **El contenido debe ir más allá del entretenimiento**, es decir, que el objetivo principal del juego no es simplemente entretener al jugador, sino el de proporcionar un contexto en el cual se logre un auto fortalecimiento, mediante el cual se motiva, educa y entrena al jugador.
- **El juego debe mantener una asociación con el mundo real**, de modo que los ambientes en los cuales se desarrolle el juego, de preferencia, deben situarse en escenarios reales. Esto favorece al jugador para identificarse, con mayor facilidad, con el área que se está representando en el ambiente virtual.
- **Debe tener fines formativos o de aprendizaje**, los juegos serios están destinados en el entrenamiento de procesos en aptitudes determinadas y la comprensión de procesos complejos, en diferentes ámbitos.

Estas características dan lugar a que el jugador realice con seriedad el papel que desempeñan al ingresar al juego, es decir, que aunque el usuario esté jugando nunca debe desviarse del objetivo principal del juego serio, lo cual genera motivación en los jugadores y hace posible un aprendizaje entretenido, que en conclusión harán que una experiencia sea significativa.

---

## 2.2. Usos

Conforme a lo que dice (MARCANO,2008), existen diversas áreas de uso, en las que se ocupan los juegos serios, algunas de éstas son:

- **Militar:** esta área hace uso principalmente de simuladores, los cuales ayudan en el entrenamiento de tácticas, toma de decisiones y uso de armamento. El uso de juegos ayuda, entre otras cosas, a que los usuarios tengan una mayor velocidad de reacción, además de que se evitan sucesos lamentables, como pérdidas humanas y ahorra gastos en equipos costosos.
- **Política:** el uso de juegos en esta área han sido fomentados, principalmente por organismos gubernamentales y organismos no gubernamentales, con el fin de informar, entrenar y persuadir a la población.
- **Empresariales:** esta área es una de las que mayor uso ha dado a los juegos, principalmente para el entrenamiento de empleados y en publicidad de productos y servicios. En el ámbito de entrenamiento, se usa para entrenar habilidades en el manejo de herramientas, o en algunas ocasiones, para el manejo de relaciones interpersonales, de comunicación o de estrategias. Una de las ventajas que se tienen en este ámbito, es que permiten a los usuarios tener una visión a futuro y generan una capacidad en la toma de decisiones y reducción de riesgos.
- **Educación:** en esta área es importante no confundir los juegos serios con *edutainment*, que se refiere al software donde se muestra información de un modo divertido, apoyándose de animaciones y elementos multimedia; por lo que se le da una mayor carga a la diversión que al contenido. Mientras que con un juego serio, sucede al revés, se le da una mayor prioridad a la parte educativa.
- **Religión:** los juegos de esta área son creados principalmente con el propósito de difundir enseñanzas y contenido bíblico. Los juegos han tenido su mayor auge en Estados Unidos, y consisten en recrear escenas y personajes de diversas historias bíblicas, como David y Goliat y la Torre de Babel, entre otros.
- **Artes:** en esta área se han desarrollado juegos para estimular la creatividad de los artistas, permitiéndoles simular posibles diseños. De igual forma en la música, brindando juegos que permiten la creación de melodías musicales.
- **Salud:** el uso de juegos para esta área se puede dividir de dos formas, ya que se tienen juegos, que al igual que en el uso militar, se usan como simuladores para el entrenamiento de estudiantes de medicina y de personal paramédico. El otro uso que se tiene en esta área es el terapéutico, la función que ejercen los juegos, como ya se dijo anteriormente, es la de recuperar habilidades motoras de pacientes que se encuentran en rehabilitación; en otros casos, también se usan como distractores de pacientes que sufren de dolor crónico y en la rehabilitación de habilidades cognitivas a través de ejercicios mentales de preguntas y respuestas. Será necesario analizar con mayor rigor el ámbito terapéutico, debido a la naturaleza del proyecto.

## 2.3. Aplicación en rehabilitación

El uso de juegos serios ha tenido mucho auge en el ámbito terapéutico, debido a los beneficios que éstos aportan y por brindar otra forma de hacer terapia. Por esta razón se analizaron a detalle diversos juegos de este ramo, con el fin de encontrar las características que tienen en común y las diferencias que tienen con los juegos utilizados en otras áreas, de modo que esta información sirva como base para el diseño y creación de Acoli.

En el capítulo I, se habló de los dos sistemas que han tenido mayor trascendencia en el ámbito terapéutico, el Rutgers ARM II y el sistema ARMEO, ambos utilizados en terapias de extremidades superiores.

El sistema Rutgers Arm II ejercita el control motor del brazo-hombro, su fortalecimiento, la velocidad de movimiento, resistencia y fortaleza. El sistema combina componentes de software y de hardware. En cuanto al hardware, el sistema cuenta con los siguientes componentes: una mesa de baja fricción

inclinada, un soporte del antebrazo con marcadores y un transmisor inalámbrico, una correa colocada sobre el hombro para detectar la inclinación, seguimiento de visión infrarroja, una pantalla grande y una PC. Por el lado del software, el sistema tiene los siguientes componentes: un dispositivo de rastreo, usado para la calibración del dispositivo, y un conjunto de juegos de rehabilitación física, con los cuales se busca, entre otras cosas, mejorar el tiempo de respuesta, la coordinación mano/ojo, el control motor, el rango de movimiento, la fuerza del hombro, la resistencia, la fuerza de agarre y coordinación de manos (BURDEA,2010).

Consta de 4 juegos, tres de ellos fueron diseños a la medida, es decir, que no son jugables por cualquier usuario fuera de la plataforma, ya que solo buscan apoyar la función clínica, o sea la rehabilitación de los pacientes. Esto permite tener un mejor control de los niveles de dificultad y la adaptabilidad de los usuarios, lo cual permite que el juego no sea imposible de ganar, esto genera un resultado favorable en la moral y la determinación, manteniendo al usuario interesado en aplicar el máximo esfuerzo.

Estos juegos están dirigidos a mejorar el tiempo de respuesta, la coordinación mano-ojo, el control motor de Extremidad Superior<sup>1</sup>, el Rango de Movimiento<sup>2</sup>, la fuerza del hombro, la resistencia, la fuerza y el agarre, además de la coordinación de las manos y el alcance (BURDEA,2010).



Figura 2.1: Breakout 3D (BURDEA, 2010)

Esta versión del juego varía de las demás, ya que con cada rebote de la pelota solo se puede destruir un bloque, lo cual está diseñado para maximizar los movimientos requeridos para destruir todos los bloques. También se manejan diferentes grados de dificultad, para evitar el aburrimiento de los usuarios que empiezan a dominar ciertos niveles del juego y fomentar así una autosuperación; la dificultad va a depender de la velocidad con la que se mueve la pelota y del tamaño de la superficie donde rebota. El último aspecto importante que cabe mencionar de este juego, es el *Conocimiento de resultados* por parte del usuario, el cual se proporciona por el porcentaje de cubos que se van destruyendo a lo largo del juego.

**Pick-and-Place**[Figura 2.2], este juego consiste en seguir un camino predefinido, buscando colocar una pelota en un área rectangular designada. El movimiento del brazo se ve reflejado en un brazo virtual que aparece en la pantalla. En un principio se requiere sujetar la pelota del juego, para lograrlo se cuenta con un dispositivo de presión, que además permite medir la fuerza que se emite. Una vez que se tiene sujeta la pelota hay que llevarla al área rectangular marcada, a través del camino trazado. Al realizar los movimientos necesarios para seguir la ruta, se llevan a cabo en el brazo movimientos de abducción/aducción, flexión/extensión o combinaciones de ambos. Por último, el *Conocimiento de resultados* se ve reflejado en la trayectoria que se va marcando en la pantalla, de los movimientos que realiza el usuario, de esta forma, el usuario podrá tener noción de la certeza que está teniendo en el juego.

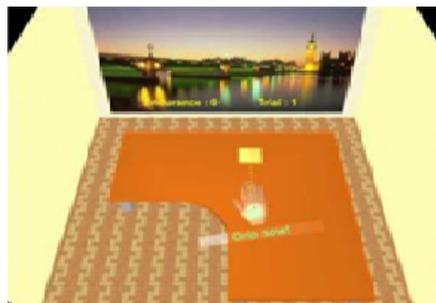


Figura 2.2: Pick and Place (BURDEA, 2010)

<sup>1</sup>UE, por sus siglas en inglés

<sup>2</sup>ROM, por sus siglas en inglés

**Treasure Hunt** [Figura 2.3], la dinámica del juego consiste en ir desenterrando tesoros que se encuentran ocultos en la arena, por medio de una pala virtual que es controlada con el movimiento del brazo. Al igual que en *Pick and Place*, se hace uso del dispositivo de presión, para realizar la acción de escavar, una vez que se ha situado la pala con el movimiento del brazo. El objetivo es desenterrar la mayor cantidad de tesoros posibles en un tiempo dado. Existen además, algunos aspectos que dan una mayor dificultad al juego, como las tormentas de arena que aparecen cada cierto lapso de tiempo, y que cubren los tesoros que ya se encontraban descubiertos, lo cual motiva al usuario a moverse con rapidez; también existen un tipo de tesoros especiales, que brindan una mayor cantidad de puntos al usuario, ya que éstos se encuentran en zonas donde el movimiento del brazo logra su máximo alcance. El *Conocimiento de resultados* lo proporcionan dos factores, uno auditivo y el otro visual; el auditivo lo hace por medio del sonido que se genera cuando desentieras un tesoro; el visual, por medio del número de tesoros que faltan por desenterrar, que se muestra en pantalla, y por los tesoros encontrados que se muestran en el escenario.



Figura 2.3: Treasure Hunt (BURDEA, 2010)

**The Baseline exercise** [Figura 2.4], este juego consiste en limpiar una área específica que representa una mesa, con el movimiento del brazo. El *baseline* es el área a la que el cliente puede llegar con el brazo afectado, estando sentado sin generar ningún tipo de inclinación del tronco. Por otra parte, también se mide la fuerza de presión, usando el dispositivo mencionado anteriormente; durante la ejecución del juego se le pide al usuario que presione el dispositivo tres veces, se saca un promedio de las tres presiones y se muestra el resultado en una barra medidora, que se encuentra en el extremo derecho de la pantalla, este porcentaje ayuda, entre otras cosas, a medir la fatiga del usuario.

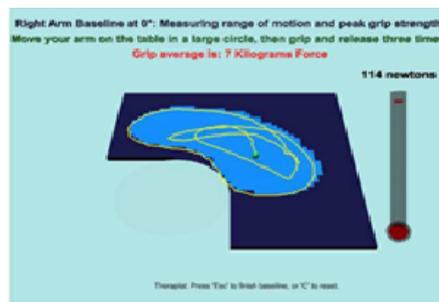


Figura 2.4: Baseline (BURDEA, 2010)

Por otro lado, tenemos el sistema Armeo, cuya terapia consta de tres productos: Armeo Boom, Armeo Spring y Armeo Power. El uso de cada producto va a depender de la gravedad de la lesión y la función motora. El software Armeo es la base de un sistema compartido de gestión de usuarios y una base de datos, proporcionando planes de terapia, modalidades de evaluación, ejercicios y juegos a través de los tres productos ARMEO.

Los tres productos corren sobre una misma plataforma de software, la plataforma brinda retroalimentación sobre el 5 desempeño, lo cual motiva el deseo de los pacientes para lograr un mayor número de repeticiones, ya que al darle conocimiento al usuario sobre sus resultados éste buscará mejorar en cada ocasión, lo que conduce a obtener mejores resultados en menos tiempo y mejorar los resultados a largo plazo.

A diferencia del sistema Rutgers Arm II, el sistema Armeo cuenta con dispositivos más sofisticados; se cuenta con dos exoesqueletos para las versiones Spring y Power, y con un sistema de poleas para la versión Boom (ARMEO,2009).

El sistema Armeo por sus diversos dispositivos, posibilita la realización de una gran gama de movimientos, que permiten la formación de extensos espacios 3D, por esta misma razón cuenta con múltiples juegos, a continuación menciono algunos de ellos:

**Diving** [Figura 2.5, a)], la dinámica de este juego, consiste en explorar un espacio 3D subacuático, con el fin de ir recopilando monedas a lo largo del recorrido. Se tiene un personaje humano que es el que se manipula con el movimiento del brazo, para ir recogiendo las monedas. Este ejercicio entrena el rango de movimiento y la coordinación de movimientos.

**Supermarket** [Figura 2.5, b)], este juego se desarrolla en una tienda en 3D, el objetivo es ir comprando

artículos de la tienda que se describen en una lista de compras, la cual se muestra en el extremo derecho de la pantalla. El movimiento del brazo mueve una pequeña mano, modelada en 3D, que simula la mano del usuario, los artículos de la tienda se toman colocando la mano sobre de ellos y ejerciendo fuerza de agarre con la mano sobre el dispositivo. Los artículos una vez agarrados, se dejan en una canasta que se muestra en pantalla, y se modifican los valores en la lista de compras, lo que brinda al usuario un conocimiento de resultados a lo largo del juego. Este ejercicio entrena el rango de movimiento, la fuerza de agarre de la mano y funciones cognitivas.

**Treasure Island** [Figura 2.5, c)], el juego consiste en ir guiando una llave a través de un escenario, construido con bloques que forman puentes, para ir desbloqueando tesoros. La llave se guía con el movimiento del brazo, también se muestra un reloj en la esquina superior derecha, para mostrar al usuario el tiempo restante que tiene para seguir desbloqueando tesoros. Mientras que en la esquina superior izquierda, se muestra el total de tesoros desbloqueados, con el fin de brindar un conocimiento de resultados al usuario. Este ejercicio entrena principalmente la coordinación de movimiento.

**Balloons** [Figura 2.5, d)], la dinámica consiste en mover un personaje e ir reventando un conjunto de globos que se muestran en pantalla, mientras se evita hacer contacto con las bombas. El movimiento del personaje, es el resultado del movimiento que se genera con el brazo al actuar sobre el dispositivo (exoesqueleto). Con el ejercicio se ejercita el rango de movimiento en el plano frontal.

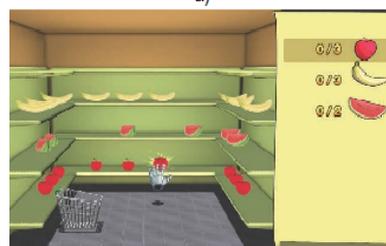
**Fishing** [Figura 2.6, f)], este juego consiste en ir atrapando peces en movimiento, que van pasando por la pantalla, en un escenario submarino. Para atrapar los peces se cuenta con una red, que se manipula con el movimiento del brazo. La realización del ejercicio ayuda a mejorar el rango de movimiento en el plano frontal, así como la coordinación de movimientos.

Los juegos serios de los sistemas Rutgers y Armeo, así como los otros analizados tienen en común las siguientes características:

- **Labores de la vida cotidiana**, en los juegos *Pick and Place* y *Supermarket* se realizan actividades que las personas realizan comúnmente, como seleccionar artículos de los anaqueles de un supermercado o mover objetos que se encuentran en una mesa de un lugar a otro. El objetivo de crear este tipo de juegos, es el de motivar a los pacientes a rehabilitarse, mostrándoles sus avances realizando actividades cotidianas y buscando así su reincorporación en las mismas.
- **Conocimiento de resultados**, en la mayoría de los juegos se tiene este elemento, que consiste en mostrar, a lo largo del juego, los resultados que se van obteniendo, ya sea con elementos visuales, como marcadores y barras de progreso, o con audio que indique que se ha tenido un logro. El hecho de incluir dichos elementos, ayuda a mantener al usuario motivado y consciente de sus avances a lo largo del juego.



a)



b)



c)



d)



f)

Figura 2.5: Juegos Armeo (ARMEO, 2009)

- 
- **Actividades reales**, como parte de las características de los juegos serios que se mencionaron anteriormente, se habla de la realización de actividades existentes en el mundo real, excluyendo todas aquellas ficticias. Es el caso de los juegos, *Supermarket*, *Diving*, *Fishing*, *Treasure Hunt* y *Pick and Place*, este acercamiento con la realidad favorece la identificación del jugador con el área de la realidad que se está representando en el ambiente virtual.
  - **Fase de calibración**, esta característica es muy específica de los juegos utilizados en rehabilitación o juegos que utilicen kinect, ya que en esta fase se miden los rangos de movimiento del paciente, dependiendo de la extremidad o extremidades involucradas en el juego, lo cual ayuda a establecer las dimensiones del área de juego y dificultad del mismo. El objetivo es que el usuario pueda realizar la actividad sin riesgo de esforzarse más allá de sus capacidades. Además, conocer de antemano los límites de movilidad del paciente, permite saber qué tanto es posible estimular al paciente a hacer pequeños esfuerzos, para obtener una recuperación más pronta.
  - **Fase de entrenamiento**, ésta es una característica de suma importancia, ya que permite a los usuarios aprender la dinámica del juego antes de iniciar con los ejercicios de terapia real. Esta propiedad evita la frustración de aquellos usuarios que no están familiarizados con los videojuegos, además de encaminarlos en este tipo de actividades y modo de hacer terapia.
  - **Dificultades diversas**, en varios de los juegos se menciona el uso de diversos niveles de dificultad. Esta característica tiene múltiples ventajas que favorecen al paciente. Por un lado, evita que se genere un aburrimiento en el paciente, ya que con el manejo de diversas dificultades se le estimula a querer superarse. Por el otro lado, también ayuda en el tiempo de recuperación, debido a que cada nivel de dificultad se asigna según el progreso del paciente, lo cual también sirve como incentivo en los pacientes, que pueden ver su progreso dependiendo del nivel en el que se encuentre.
  - **Diversos elementos gráficos**, esta característica se refiere a aquellos elementos visuales que tiene el juego, y que brindan una experiencia más agradable al momento de jugar, algunos de estos elementos son:
    - **Ambientes y personajes 3D**, en juegos como *Diving*, *Ballons*, *Fishing*, entre otros, se puede ver que a pesar de que la dinámica del juego se desarrolla en 2D, los elementos como los peces, los globos y los mismos avatares, están diseñados en 3D, al igual que los escenarios. Estos elementos 3D, como ya se mencionó, ayudan a los usuarios a pasar un momento placentero, participativo e interactivo.
    - **Tiempo restante/transcurrido**, este elemento visual se muestra en casi todos los juegos que tienen un límite de tiempo, en el cual se busca cumplir un objetivo o generar un cierto número de puntos. El hecho de mostrar al usuario, en todo momento el tiempo restante, fomenta el conocimiento de resultados, que como se mencionó en un principio, ayuda a mantener al usuario motivado y consciente de sus avances a lo largo del juego.
    - **Menú de opciones**, el contar con este elemento, ayuda a evitar frustración en los usuarios, ya que un menú bien diseñado, promueve la fácil navegación del usuario, permitiendo así encontrar cualquier elemento del juego sin ninguna dificultad.
    - **Marcadores o barras de progreso**, por último tenemos este elemento visual, cuya tarea es, al igual que mostrar el tiempo restante, generar el conocimiento de resultados en el usuario, ya que pretende mostrar los avances que se tienen a lo largo del juego, ya sea con un marcador de puntaje o con algún tipo de barra de progreso.

## 2.4. Ventajas

Se tienen diversas ventajas al hacer uso de juegos serios, algunas de éstas, según (MARGULIS,2010) son:

- El juego no aborda un problema de frente. Las soluciones más innovadoras a los problemas labo-

---

rales y personales surgen de un enfoque lúdico.

- Brinda una oportunidad para aprender sin riesgo de ser juzgados, debido a que todos estamos en igualdad de condiciones. Dado que la participación puede ser simultánea, entonces nadie queda expuesto.
- Cuando jugamos podemos tomar otros roles. Ponernos en el lugar de los otros para comprender sus puntos de vista.
- El juego ayuda a los participantes a disfrutar de un momento placentero, participativo, interactivo y desafiante en entornos presenciales y virtuales.
- Promueve la experimentación y nos enfrenta a problemas simulados antes que a los reales, nos permite ver la realidad desde un lugar y un modo diferentes. Se pueden cometer errores y aprender de ellos sin consecuencias directas sobre la realidad.

Además de las ventajas ya mencionadas, (CHIPIA,2011) menciona otras ventajas que aplican principalmente a estudiantes, estas son:

- Permiten la estructuración del lenguaje y el pensamiento
- Permiten aprendizajes de fuerte significación
- Invita a la participación activa del jugador
- Desarrolla creatividad, competencia intelectual, fortaleza emocional y estabilidad personal.

Además se desarrolla la coordinación ojo-mano, una mayor agudeza visual, rapidez de reacción, capacidad de reacción a múltiples estímulos, facilidad para relacionarse con otros, una alta motivación al logro, mayor tolerancia a la frustración, capacidad para tomar riesgos, resolver problemas y tomar decisiones.

## 2.5. Proceso de desarrollo

Para el desarrollo de Acoli, se siguieron dos procesos distintos; el primero enfocado en el desarrollo de juegos serios, y el segundo utilizado en Interacción Humano Computadora, el cual va enfocado en las necesidades del usuario. Combinando ambos procesos se pretende desarrollar, de forma correcta, un juego serio que se acople a las necesidades de los usuarios.

### 2.5.1. Proceso de desarrollo de un Juego Serio

Según las buenas prácticas para el desarrollo de un juego serio de (SÁNCHEZ,2007), el proceso consta de 3 pasos:

- Análisis del contexto
- Diseño de la información
- Evaluación de la funcionalidad

A continuación se describe a detalle, en que consiste y que actividades se realizan en cada paso.

**Análisis del contexto**, en esta fase se pretende realizar una planeación que nos permita tener objetivos claramente expresados, es decir, tener bien delimitados los alcances del juego que se quiere desarrollar. Igualmente, en esta fase se define el público al cual irá dirigido el juego, ya que dependiendo del público, se establecerán las variables, es decir aquellos valores que nos facilitarán el trabajo en la etapa de diseño y desarrollo, como son el género, la edad, el nivel social, las motivaciones y los intereses de los jugadores.

Por último, se debe dejar claramente acotado el *plan de diseño*, en el cual se definen los recursos y el presupuesto con el que se cuenta para la elaboración del juego. Elaborar un buen *plan de diseño* nos

---

va a permitir que las tareas se realicen de forma más ágil, que se disminuyan imprevistos, se mejore la calidad de los resultados y se pueda hacer una mejor predicción de las consecuencias y las limitaciones que se puedan tener posteriormente.

**Diseño de la información**, esta fase implica definir el tema y el estilo visual, establecer un diseño de pantallas, crear los elementos estructurales y los elementos de control<sup>3</sup>, agregar imágenes y texto que llevarán las pantallas. Una vez teniendo todos estos elementos se integran y se generan prototipos de pantallas, sin dejar de tomar en cuenta los antecedentes de la fase anterior.

Para lograr un buen diseño, en el cual exista un adecuado balance entre lo lúdico y el aprendizaje, es importante no perder de vista aspectos como la claridad, la sencillez y la facilidad de uso en combinación con aquello que se quiere enseñar. Es decir, el objetivo debe ser lo más fácil e intuitivo posible, ya que si al usuario se le dificulta demasiado el juego, éste terminará por frustrarse y por abandonarlo, de igual forma, si el juego es demasiado complicado, esto provocará que se pierda la diversión y la motivación relacionada al juego. Una vez alcanzado el objetivo lúdico será más sencillo agregar los aspectos de aprendizaje sin que el jugador lo note.

El último aspecto a considerar en el diseño es la creación de personajes y ambientes del juego, que ayuden a que los usuarios sientan una mayor motivación mientras juegan. En el caso de Acoli, no se usarán personajes ni ambientes reales, debido a que una de las motivaciones para crear Acoli fue que las terapias llegaban a ser tediosas. Por esta razón, reproducirlas tal cual en un ambiente virtual no resuelve esta problemática, es por eso que se decidió ambientar Acoli en un contexto completamente distinto al terapéutico, buscando así captar la atención de los usuarios.

**Evaluación de la funcionalidad**, en esta última etapa se lleva a cabo una prueba real del juego, con el propósito de encontrar errores, ya sea de funcionalidad o de diseño, para los cuales se busca una solución o una manera de minimizarlos. De igual forma, en esta etapa se prueban diferentes opciones de presentación y controles del juego, desde los menús iniciales hasta la ejecución, buscando la que ofrezca un ambiente más lúdico y fácil de entender.

Es recomendable hacer una prueba piloto con voluntarios, para observar las fortalezas y debilidades del juego, y realizar las modificaciones necesarias, con base en la retroalimentación obtenida, además de poder medir el flujo en los usuarios, es decir, el nivel de concentración que éstos pueden alcanzar mientras hacen uso de Acoli. Para medir el flujo, se aplicarán diversos cuestionarios a cada uno de los usuarios una vez terminada su partida.

En nuestro caso, las pruebas del juego se realizarán en personas sanas, es decir, que no padecen de hombro congelado, debido a que la realización de pruebas con pacientes reales tardaba poco más de un año para poder ser aprobada por el INR.

## 2.5.2. Proceso de desarrollo utilizado en IHC

Como en todo proceso de desarrollo entender las necesidades del usuario muchas veces es complicado, debido a que éstas cambian constantemente o se interpretan de forma errónea. Por lo que, usar este proceso es de gran ayuda, ya que mejora la utilidad, la eficiencia e incrementa la usabilidad<sup>4</sup> del producto a desarrollar. Existen dos pasos o subprocesos que son de gran importancia en el proceso de desarrollo.

El primer subproceso es la **Búsqueda de necesidad o Needfinding**, el cual nos proporciona un mejor entendimiento de las necesidades latentes de los usuarios y consta de cinco principios según (MIT,2012):

1. Concentrarse en las necesidades y no en las soluciones, ya que esto podría limitar las oportunidades.
2. Ir a los ambientes de los usuarios, esto ayuda a hacer las necesidades más específicas.

---

<sup>3</sup>Fondo, ventanas, botones, enlaces, etc.

<sup>4</sup>Desarrollar sistemas fáciles de aprender y fáciles de usar (FLORÍA, 2000)

- 
3. Ir más allá del problema obvio.
  4. Reunir diferentes tipos de datos como notas, fotos, video y audio.
  5. Iterar cuantas veces sea necesario.

Además de estos principios del Needfinding, (MIT,2012) menciona cuatro pasos importantes que hay que seguir para un correcto desarrollo:

1. **Identificar un grupo de usuarios y tareas de estudio**, establecer los límites a la hora de seleccionar el grupo de usuarios y la combinación de la actividad; es importante considerar un enfoque no muy amplio, ya que impide y limita aquellas oportunidades específicas.
2. **Crear hipótesis sobre la actividad**, crear hipótesis ya sea de causalidad o de correlación entre las diferentes variables, que puedan ser probadas a través de observaciones y entrevistas.
3. **Reunir información acerca de la actividad**
  - a) **Investigación de escritorio**, antes de realizar investigaciones de campo, es importante realizar investigaciones de escritorio para encontrar soluciones existentes. Esto puede incluir evaluaciones comparativas, investigación en fuentes físicas o investigación en línea.
  - b) **Investigación de campo**, los estudios de campo son los datos que se recogen de primera mano de los usuarios. Esto incluye hallazgos y puntos de vista de las observaciones, entrevistas y encuestas. Elementos visuales como fotos y videos son muy útiles en esta etapa. Se pueden extraer posibles soluciones en este punto.
4. **Identificar oportunidades del producto**, analizar aquellos aspectos que puedan mejorarse o incluirse, este paso se debe procurar llevar a cabo todo el tiempo.

El siguiente subproceso, es el de **Prototyping** (o prototipado), que consiste en poner nuestras ideas en un modelo interactivo preliminar. En el desarrollo de software se busca desplegar la funcionalidad del producto en fase de desarrollo, aunque puede o no contener la lógica del software.

Realizar el prototipado, antes del proceso de desarrollo, tiene diversas ventajas. Por ejemplo, ayuda en la estimación de esfuerzo. Permite que los usuarios evalúen propuestas de los desarrolladores y probarlos antes de su implementación. También ayuda a entender los requerimientos que son específicos del usuario y que no fueron considerados durante el diseño, además de ser un método de bajo costo.

Existen diversos pasos que hay que seguir, según (SDLC,2006) para lograr un prototipado exitoso:

- **Identificación de requerimientos básicos**, es decir, aquellos requerimientos que se encuentran en términos de la interfaz de usuario.
- **Desarrollo del prototipo inicial**, en él se muestran los requisitos básicos y se proporcionan interfaces de usuario, las cuales pueden tener o no funcionalidad.
- **Revisión del prototipo**, se muestra el prototipo al usuario. Además, se colecta, organiza y usa la retroalimentación obtenida de los usuarios, para hacer mejoras.
- **Revisar y mejorar el prototipo**, en este paso se discute la retroalimentación y los comentarios del paso anterior; se hacen negociaciones con el usuario, basándose en los siguientes factores: el tiempo, el presupuesto y la viabilidad de la aplicación efectiva. Los cambios aceptados son incorporados en un nuevo prototipo.

El tercer y cuarto paso se iteran varias veces, hasta cumplir con las expectativas de los usuarios.

Además (SDLC,2006) menciona diversos tipos y modos de hacer prototipado. Por ejemplo, tenemos el modo de **Prototipado Horizontal** y el **Prototipado Vertical**. El primero es en el que se muestran las interfaces de usuario del producto y brinda una visión más amplia del sistema entero. Por otro lado, el segundo detalla solo una función específica o un subsistema del producto.

Entre los tipos de prototipado que menciona (SDLC,2006), están los siguientes:

- **Prototipado rápido/Throwaway**: se requiere poco esfuerzo, ya que en cuanto se entienden las necesidades reales, el prototipo se desecha y se inicia con el desarrollo.

- 
- **Prototipado evolutivo:** inicia con un mínimo de funcionalidades, el cual será el núcleo de los siguientes prototipos, alrededor del cual el sistema será construido.
  - **Prototipado incremental:** se construyen diversos prototipos funcionales de varios subsistemas para después integrarlos todos para formar el sistema completo.
  - **Prototipado extremo:** se compone de tres fases secuenciales. Primero, se hace un prototipo básico con todas las páginas existentes, el cual se presenta en formato html. A continuación, el procesamiento de datos se simula mediante una capa de servicios del prototipo. Por último, los servicios se implementan e integran para el prototipo final.

### 2.5.3. IHC y Desarrollos tradicionales

En esta sección se lleva a cabo una comparativa entre los procesos de desarrollo utilizados en el presente proyecto, contra las metodologías tradicionales de desarrollo de software. Con el fin de encontrar similitudes entre ambas metodologías y sustentar el proceso de desarrollo utilizado, además de localizar aquellas diferencias que puedan ser de utilidad para el desarrollo.

#### Proceso unificado

Un proceso de desarrollo de software se define como "*el conjunto de actividades necesarias para transformar requisitos de un usuario en un sistema de software*" (JACOBSON, 1999). El proceso unificado o UP<sup>5</sup>, es un proceso de desarrollo, capaz de ser adaptado a proyectos específicos. El primer libro sobre el proceso unificado se denominó, en su versión española, El Proceso Unificado de Desarrollo de Software (ISBN 84-7829-036-2) y fue publicado en 1999 por Ivar Jacobson, Grady Booch y James Rumbaugh, desarrolladores del Lenguaje Unificado de Modelado<sup>6</sup>

Dicho proceso se caracteriza por ser Iterativo e Incremental y por estar dirigido a Casos de Uso. Con esto se refiere a que los contenidos de cada iteración se definen según un determinado número de casos o escenarios, en donde, por cada iteración estos casos se diseñan, implementan y prueban de manera incremental. Tal proceso, se divide en cuatro iteraciones, que constituyen el ciclo de vida de un proyecto (JACOBSON, 1999):

1. **Inicio**, en esta fase se realizan actividades tales como justificar el proyecto, plantear los casos de uso y requerimientos, y todo lo referente a la planeación del proyecto. Es una fase corta, pero de suma importancia, debido a que las fases siguientes van a depender de lo planteado aquí.
2. **Elaboración**, durante esta fase se continua con la recolección de requerimientos, se realiza una identificación de riesgos, además de establecer la arquitectura del sistema. Esta fase ayuda a mostrar, de manera adecuada, la conducta, la escalabilidad y el coste del proyecto, lo cual será de gran ayuda para la fase de construcción. De manera adicional, en esta fase se pueden realizar diversos diagramas UML, que sean de ayuda para representar, de manera simple, los requerimientos.
3. **Construcción**, se caracteriza por ser la fase más larga del proyecto, lo cual se debe a que se realiza toda la construcción con base en lo especificado en la fase de Elaboración. Esta fase, puede dividirse en pequeñas iteraciones, en donde por cada iteración se entrega una versión ejecutable de software al usuario.
4. **Transición**, en esta fase se muestra el sistema a los usuarios finales, con el fin de que se genere una retroalimentación y se incorporen mejoras en iteraciones futuras. De igual forma, se incluye el entrenamiento de los usuarios para la utilización del sistema.

Analizando las fases del proceso UP, se encuentran varias similitudes con los procesos de IHC y el proceso de desarrollo de un juego serio. Específicamente, en la fase de **Inicio** y **Elaboración**, en las cuales se planifica gran parte del proyecto, con base en los requerimientos obtenidos, lo cual se llevó a cabo con el proceso de needfinding y la fase de análisis del proceso de desarrollo de un juego

---

<sup>5</sup>Por sus siglas en inglés.

<sup>6</sup>UML, por sus siglas en inglés

---

serio. Continuando con la parte de planificación, utilizaremos algunos diagramas UML del proceso UP, para representar de forma clara, los requerimientos previamente obtenidos y reforzar la planeación del proyecto.

Los elementos mencionados en este capítulo junto con un proceso de desarrollo basado en IHC (Needfinding y Prototipado) nos proporcionaron una guía para la implementación y desarrollo de la interfaz de Acoli.

Con respecto al aspecto emocional, en el siguiente capítulo hablaremos a detalle de la teoría de flujo en la cual se apoya Acoli para reforzar aspectos de motivación intrínseca y extrínseca por parte de los usuarios, lo cual es un diferenciador y aspecto clave de esta tesis respecto a otros videojuegos serios realizados hasta la fecha.

---

## Capítulo 3

# Teoría de Flujo

En este capítulo se profundiza en un marco teórico de IHC, llamado Experiencia de Usuario Óptima o Teoría de flujo, empezando por su historia, definición, así como de las aplicaciones en el campo computacional, y como puede ser de utilidad en el desarrollo de juegos serios y en particular de Acoli.

### 3.1. Definición y características

#### 3.1.1. ¿Qué es el flujo?

La Teoría de la Experiencia de Usuario Óptima o Flujo surge en el área de la psicología para explicar un estado de absorción e implicación en una actividad. Su creador el psicólogo Mihaly Csikszentmihalyi en 1975, la define de la siguiente manera: *"Estar totalmente absorto por la actividad que se realiza. El ego desaparece. El tiempo vuela. Cada acción, movimiento y pensamiento sigue inevitablemente al anterior, como si se tocara jazz. Todo tu ser está implicado, y estás utilizando tus habilidades al máximo"*(NAKAMURA,2002).

Dicha teoría intenta explicar lo que ocurre cuando la realización de una actividad provoca en una persona una sensación tan placentera que el individuo la realizará por el puro placer de hacerla, aunque la actividad sea muy difícil. Un aspecto que cabe resaltar para alcanzar dicho estado, es el balance que debe existir entre las habilidades que tiene uno mismo y la actividad que se realiza, Csikszentmihalyi logro hacer un primer mapeo de este balance en un diagrama, en términos de desafíos percibidos y habilidades, identificando tres regiones de experiencia (NAKAMURA,2002):

- Un canal de flujo en el cual los desafíos y las habilidades empatan.
- Una región de aburrimiento, cuando las habilidades sobre la acción superan a la misma.
- Una región de la ansiedad, cuando los retos superan gradualmente la capacidad de adaptación.

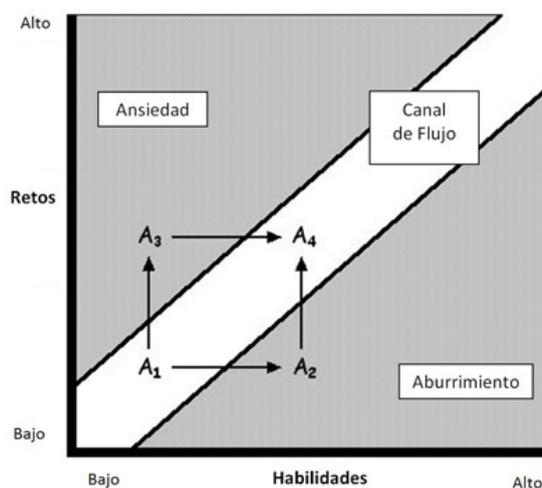


Figura 3.1: Modelo original del estado de flujo (NAKAMURA, 2002)

Este modelo se basa en los reportes originales del flujo (véase la Figura 3.1).

Posteriormente, el grupo Milan mostró otro mapeo en el cual se divide el área en ocho canales experimentales (apatía, aburrimiento, relajación, control, preocupación, ansiedad, excitación y flujo) en lugar de cuatro cuadrantes (véase la Figura 3.2). En donde, la condición de la experiencia se intensifica dentro de un canal o cuadrante como retos y habilidades se alejan de los niveles promedio de una persona. Es decir, que dividen el espacio de desafío/destreza en una serie de anillos concéntricos, asociados con el aumento de la intensidad de la experiencia.

El nombre del concepto surgió durante las entrevistas que realizaba Csikszentmihalyi en 1975, cuando las personas describían sus experiencias al encontrarse en dicho estado, éstos usaban la metáfora de una corriente que los llevaba hacia adelante, y así fue como se le dio el nombre de flujo.

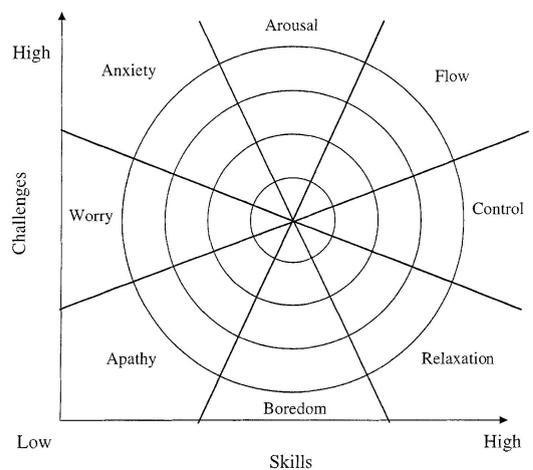


Figura 3.2: Modelo actual del estado de flujo (NA-KAMURA, 2002)

### 3.1.2. Historia de la Teoría del Flujo

Csikszentmihalyi inició sus estudios sobre lo que hoy en día llamamos *Flujo*, mientras estudiaba a un grupo de artistas, que cuando pintaban o esculpían, se mostraban en una completa concentración, pasaban hora tras hora en la actividad que realizaban, al grado de olvidarse de comer, de la fatiga y el malestar. Sin embargo, Csikszentmihalyi se percató de que la mayoría de los artistas perdían rápidamente interés, una vez que concluían su obra. De igual forma, notó que la motivación de los artistas por realizar su obra, no venía de factores externos como el dinero, el reconocimiento o la fama. Este fenómeno, después conocido como motivación intrínseca<sup>1</sup>, fue lo que llevó a Csikszentmihalyi a investigar con mayor profundidad y dar origen a la teoría de flujo (WALTER, 2011).

En 1968, el investigador Abraham Maslow llegó a la conclusión de que personas, como los artistas que había investigado Csikszentmihalyi, encontraban la recompensa por su trabajo al realizar el trabajo mismo, razón por la cual hacían su actividad sin esperar ninguna otra recompensa. Maslow nombró a este tipo de motivación como deseo de "*self-actualization*" (WALTER, 2011).

A Csikszentmihalyi no le convenció del todo la investigación de Maslow, debido a que aún quedaba la interrogante de saber si existen otras actividades intrínsecamente gratificantes, y si todas las personas eran capaces de entrar en un estado de flujo. Fue entonces, que el investigador Richard DeCharms, uno de los primeros investigadores enfocados en la motivación intrínseca, descubrió que las diferencias existían según el nivel de control que tuvieran las personas sobre sus vidas. Es decir, que aquellos intrínsecamente motivados, disfrutaban hacer su trabajo sin importar los factores externos. Mientras que aquellas personas que decían no sentir un control sobre sus vidas, sentían que sus acciones eran controladas por factores externos (WALTER, 2011).

Por otro lado, el investigador de la Universidad de Rochester, Edward Deci descubrió que las personas que reciben una recompensa monetaria por realizar actividades que disfrutaban, perdían interés en realizar la actividad más rápido que aquellas que no recibían ningún tipo de recompensa externa (WALTER, 2011).

Aún con todas las investigaciones realizadas en aquel tiempo, Csikszentmihalyi seguía sin encontrar respuesta a la pregunta, de si era posible reproducir el estado de los artistas, haciendo otras actividades de la vida cotidiana. Esto lo llevó a investigar personas que ocupaban mucho tiempo en

<sup>1</sup>Que es propio o característico de la cosa que se expresa por sí misma y no depende de las circunstancias

---

realizar actividades agotadoras, por las cuales no recibían ningún tipo de recompensa. Fue entonces que empezó a investigar a atletas aficionados, maestros de ajedrez, escaladores de montaña, bailarines, basquetbolistas de colegios y compositores de música. Él quería centrarse en cómo las personas describían la actividad cuando iba bien. Esta investigación concluyó que existía un autotélico<sup>2</sup> común, o gratificante en y por sí mismo, experimentado por todos los participantes, el cual fue nombrado como flujo (WALTER,2011).

### 3.1.3. Fomento de un estado de flujo

Según Csikszentmihalyi en (CSIKSZENTMIHALYI,1975), existen cuatro factores que deben estar presentes para alcanzar un estado de flujo. Aunque, no es necesario en todos los casos que se cumplan todos los factores.

1. **Equilibrio entre habilidad y desafío.-** Debes encontrar una actividad coherente con tus habilidades, gustos y nivel de dificultad. El Flujo se produce cuando las habilidades de la persona están en equilibrio con los retos y desafíos que la actividad le plantea. En estas circunstancias, la atención se concentra en la obtención de una meta realista, la retroalimentación que se obtiene sobre el nivel de ejecución es inmediata y uno llega a olvidarse de sí mismo. Además, es más sencillo entrar en un estado de flujo cuando tus talentos y fortalezas internas, ya sea habilidades, conocimientos y actitudes se ponen en juego para hacer una actividad que te lleva a un objetivo al que emocionalmente estás vinculado positivamente, o que te gusta.
2. **Concentración y enfoque.-** Se debe buscar la máxima concentración evitando todo tipo de interrupciones. Es decir, se debe construir un entorno adecuado, libre de distracciones. Cuando se quiere entrar en un estado de flujo, siempre se debe empezar con este hábito, pues si tienes distracciones, interrupciones, tu mente está dispersa, o estás con la tentación de caer en la multitarea, será imposible llegar a un estado de flujo. Para conseguirlo, necesitas una ausencia de distracciones internas: pensamientos, preocupaciones, ideas, etc. Así mismo, una ausencia de distracciones externas: mensajes, llamadas, alertas, radio, televisión, entre otras.
3. **Objetivos claros.-** Se debe tener un objetivo claro, antes de empezar a realizar la actividad, se tiene que conocer claramente lo que se quiere lograr. Debes visualizar el objetivo y el beneficio que se logrará al terminar la actividad que está por comenzar. Si no se sabe a dónde se quiere llegar, da lo mismo lo que se haga. Si se quiere estar presente en lo que se hace, hay que saber para que se está haciendo. Cuando se sabe lo que se quiere, es más fácil alcanzar el estado de flujo.
4. **Retroalimentación directa e inmediata.-** Es importante que quien realiza la actividad conozca en todo momento de su ejecución, sus logros y errores obtenidos. Ser consciente de esta información estimula a quien realiza la actividad. Por una parte, a sentir que está haciendo bien las cosas y motivarlo a continuar con la actividad. Por otra parte, el conocimiento de errores, ayuda a estimular a quien realiza la actividad a esforzarse más, para poder continuar realizando la actividad, buscando que se supere en todo momento.

Además de estos cuatro factores, en 2013 el investigador Owen Schaffer, propuso nuevos factores requeridos para entrar en flujo. Debido a que surgieron diversas discusiones sobre el primer modelo propuesto. Una de ellas fue, que no aseguraba un equilibrio entre la percepción de los desafíos y las habilidades. Ya que, en otras investigaciones notaron que los individuos con un nivel promedio bajo de habilidades y un nivel promedio alto de desafíos (o al revés), no necesariamente experimentaban un ajuste entre las capacidades y desafíos cuando ambos están por encima de su promedio individual. Además, otro estudio descubrió que en situaciones con pocos desafíos que fueron superados por la habilidad, se asociaron con el disfrute, la relajación y la felicidad, que según el modelo inicial es contrario a la teoría de flujo. Los nuevos factores, propuestos por Schaffer son:

- Alta percepción de los retos implicados en la actividad
- Alta percepción de las habilidades

---

<sup>2</sup>Se refiere a cualquier tipo de acción que tiene en sí misma el justificativo de su propio fin

- Conocimiento de qué hacer
- Conocimiento de cómo hacerlo
- Conocimiento de lo bien que se hace
- Conocimiento de dónde ir (si la navegación está involucrada)
- Liberación de las distracciones

Una vez que se esta en flujo, en (CSIKSZENTMIHALYI,1975) se menciona que existen diversas características resultantes, que se ven reflejadas en la persona que realiza la actividad, estas son las siguientes:

- Pérdida del autoconocimiento.
- Alteración del sentido del tiempo.
- Sentimiento de control sobre la actividad que se realiza y el entorno.
- Se asimila uno con la actividad, es decir, la conciencia se reduce a la actividad misma. Acción y conciencia se hacen uno.

Cabe resaltar que dichos factores juegan un papel importante para el desarrollo de Acolí, debido a que se busca retomarlos e implementarlos en los diversos componentes del juego, con el fin de fomentar una experiencia óptima de usuario. En la última sección de este capítulo se especifican en que componentes del juego se incorporan los elementos del flujo.

### 3.1.4. Medición del flujo

Por último, en esta sección se explica el procedimiento para poder medir el flujo en los usuarios, es decir, medir cada uno de los factores requeridos para entrar en un estado de flujo.

Para medir el flujo nos basamos en *La escala de estado de flujo* desarrollada por (JACKSON,1996), la cual consiste en un cuestionario de 36 preguntas, las cuales representan los factores requeridos para entrar en flujo, donde cada factor es medido por 4 preguntas, la correspondencia de cada factor con las preguntas se muestra en la siguiente tabla:

FACTOR	PREGUNTAS
EQUILIBRIO ENTRE HABILIDAD Y DESAFÍO	1, 10, 19 y 28
CONCENTRACIÓN Y ENFOQUE	2, 11, 20 y 29
METAS CLARAS	3, 12, 21 y 30
RETROALIMENTACIÓN DIRECTA E INMEDIATA	4, 13, 22 y 31
CONCENTRACIÓN EN LA TAREA EN CUESTIÓN	5, 14, 23 y 32
PARADOJA DE CONTROL	6, 15, 24 y 33
PERDIDA DEL AUTOCONOCIMIENTO	7, 16, 25 y 34
ALTERACIÓN DEL SENTIDO DEL TIEMPO	8, 17, 26 y 35
EXPERIENCIA AUTOTELICA	9, 18, 27 y 36

Dicha escala es del tipo Likert, o sea, que las respuestas son puntuadas en un rango de valores como el que se muestra a continuación:

- 1 - Totalmente en desacuerdo
- 2 - En desacuerdo
- 3 - Ni de acuerdo ni en desacuerdo
- 4 - De acuerdo
- 5 - Totalmente de acuerdo

---

Una vez contestadas las 36 preguntas se realiza una suma de todas las respuestas, el resultado obtenido es el valor que nos indica el nivel de flujo que el usuario alcanzó. Para obtener un resultado con valores válidos, es necesario tomar una muestra de un mínimo de 20 personas que realicen la actividad y posteriormente contesten el cuestionario. Las 36 preguntas que conforman el cuestionario se encuentran en el ANEXO C del presente trabajo.

## 3.2. Usos y aplicaciones

Csikszentmihalyi menciona algunas aplicaciones extrínsecas del concepto de flujo, tales como métodos para el diseño de espacios de juegos que promuevan la obtención de la experiencia de flujo. Otros investigadores se enfocan en aplicaciones intrínsecas, tales como espiritualidad, la mejora del rendimiento o autoayuda. En algunos casos se ocupa el proceso de flujo para aumentar el rendimiento en áreas de negocios, improvisación al tocar piano, psicología del deporte y comedia en vivo.

**En educación**, existe el término de "sobre-aprendizaje", el cual influye en la capacidad de los estudiantes para entrar en flujo. Según (DRISKELL, 1992), "*Se refiere al exceso de entrenamiento deliberado de una tarea más allá de un criterio de conjunto*". Este "sobre-aprendizaje", permite alcanzar una concentración mental, visualizando el rendimiento deseado como algo singular, una acción integrada en vez de un conjunto de acciones. Un ejemplo que notó Csikszentmihalyi, fue el método Montessori, en el cual según el estudio realizado en alumnos de nivel medio superior, bajo el régimen Montessori, se reportó un mayor efecto en cuanto a la potencia, motivación intrínseca, experiencia de flujo y la participación indivisa en el ejercicio de las actividades académicas, en comparación con los alumnos del régimen tradicional (RATHUNDE, 2005).

**En deportes**, muchos atletas mencionan el término "estar en la zona", según (COX, 2007) se dice que un deportista "está en la zona", *Cuando el cuerpo se encuentra en una situación extrema, y la mente está completamente centrada en eso, incluso sin tener conciencia de lo que hacemos, podemos lograr lo más extraordinario*. Dicho concepto encaja perfectamente con la definición dada por Csikszentmihalyi de la experiencia de flujo. Actualmente existen múltiples investigaciones sobre el flujo, que giran alrededor de cómo utilizarlo para optimizar el desempeño de los atletas y obtener mejores resultados.

**En música**, los casos de flujo se han presentado principalmente en solistas, que han experimentado un estado similar mientras tocan sus instrumentos. Donde el rendimiento del músico va acompañado de un estado de experiencia óptima. Investigaciones anteriores han demostrado una correlación positiva entre el flujo y la calidad de los resultados y sugiere que el flujo puede funcionar como una señal de recompensa que promueve la práctica. Fue entonces, que tocar el piano se utilizó como medio para inducir el flujo, con el propósito de analizar la relación entre los informes subjetivos de flujo y medidas psicofisiológicas (MANZANO,2010).

**En juegos**, pasa algo interesante, ya que muchos de los elementos necesarios para entrar en estado de flujo, se pueden encontrar en los juegos. Por ejemplo, los objetivos y metas claras; en la mayoría de los juegos antes de iniciar a jugar, se dan a conocer al jugador los objetivos, ya sea de forma explícita o implícita. En los juegos también tenemos una retroalimentación inmediata, que se presenta en forma de puntuaciones, premios o logros que se obtienen ya sea al final o durante el juego, también se puede presentar por medio de audio o elementos visuales que den señal de si se tuvo un logro o un error. Por último, el balance entre la habilidad y el desafío; este elemento tan importante para alcanzar un estado de flujo, se da en juegos que contemplan diversos niveles de dificultad, esto permite que los jugadores elijan la dificultad que se acople a sus habilidades, y que puedan aumentarla conforme crecen las mismas.

## 3.3. El Flujo en los juegos

El flujo desempeña un papel importante en los juegos, ya que nos brinda parámetros para llevar a cabo el diseño de los mismos, buscando siempre que el juego cumpla las condiciones para entrar en flujo.

---

Además, contribuyó a crear una técnica de aprendizaje, conocida como **Gamificación (o ludificación)**. La Gamificación “*es una técnica de aprendizaje que traslada la mecánica de los juegos al ámbito educativo-profesional con el fin de conseguir mejores resultados, ya sea para absorber mejor algunos conocimientos, mejorar alguna habilidad, o bien recompensar acciones concretas, entre otros muchos objetivos*”(GAITÁN,2013). En (GALLEGO,2014) definen la Gamificación como “*el uso de estrategias, modelos, dinámicas, mecánicas y elementos propios de los juegos en contextos ajenos a éstos, con el propósito de transmitir un mensaje o contenidos o de cambiar un comportamiento, a través de una experiencia lúdica que propicie la motivación, la implicación y la diversión.*”

La gamificación ha ganado gran terreno en el campo de la educación, ya que por su carácter lúdico facilita la obtención de conocimientos de una forma más divertida. Además de que con el juego se motiva a las personas, desarrollando un mayor compromiso, e incentiva el ánimo de superación. Por ejemplo, una técnica utilizada para generar motivación en los usuarios, es la forma de recompensarlos en función de los objetivos alcanzados, según (GAITÁN,2013), éstas son las técnicas más utilizadas:

- **Acumulación de puntos:** se asigna un valor por determinadas acciones, que se van acumulando a medida que se realizan.
- **Escalado de niveles:** se definen una serie de niveles que el usuario debe ir superando para llegar al siguiente.
- **Obtención de premios:** a medida que se obtienen diferentes objetivos se van entregando premios coleccionables.
- **Regalos:** bienes que se dan al jugador o jugadores de forma gratuita al conseguir un objetivo.
- **Clasificaciones:** clasificar a los usuarios en función a los puntos u objetivos logrados, destacando a los mejores en un ranking.
- **Desafíos:** competiciones entre los usuarios, donde el mejor obtiene los puntos o el premio.
- **Misiones o retos:** conseguir o resolver el objetivo planteado, ya sea solo o en equipo.

Otro beneficio que brinda la gamificación, es el poder de experimentar, es decir, que tenemos la oportunidad de cometer errores, donde la mayor consecuencia es volver a empezar a jugar. Con los métodos de enseñanza convencionales, casi no se les permite a los alumnos equivocarse y volver a empezar, esto provoca que los alumnos tengan miedo a equivocarse. Es por esta razón, que la gamificación muestra ser una buena opción no solo en el ámbito educativo, sino también en otras áreas, como en el ámbito militar y la medicina.

Sin embargo, la utilización de la gamificación también tiene ciertos riesgos sino la ocupamos adecuadamente. Por esta razón, en el momento de implementar un sistema gamificado hay aspectos que no debemos pasar por alto, por ejemplo (ZALDÍVAR):

- Reducir la gamificación a la mera obtención de premios. Es decir, el riesgo puede darse por una rápida y automática realización de las tareas por parte del alumno que impida la correcta asimilación del aprendizaje, y pueda dejar en segundo término el objetivo planteado. Existe otro problema al respecto, ya que el obtener un premio muchas veces es intrascendente. Se ha demostrado, que el hecho de ganar u obtener algún tipo de premio inmaterial no perdura por mucho tiempo.
- Otro aspecto complicado y laborioso, es encontrar el término medio en un juego para asegurar que, dentro de un entorno interesante, se pueda realizar un aprendizaje efectivo.

Una vez que se tomaron en cuenta los elementos del flujo, y los beneficios que aportan en el desarrollo de juegos. Se retomaron dichos elementos para el desarrollo de Acoli, buscando así que los usuarios alcancen un estado de flujo durante la realización de su terapia y logren, al estar completamente inmersos en la actividad, una mejor y más pronta recuperación.

En el siguiente capítulo se hablará sobre todo lo referente al prototipo de Acoli y el proceso de desarrollo que se llevó a cabo, tomando en cuenta los elementos del proceso de desarrollo de un juego serio, el prototipado y el needfinding, mencionados en el capítulo 2, y los elementos de la teoría de flujo aplicados a videojuegos mencionados en este capítulo.

## Capítulo 4

# Caso de Estudio: Macoli y el videojuego serio Acoli

En este capítulo se describe el proceso de desarrollo del videojuego serio Acoli. El proceso está dividido en tres etapas: análisis, diseño e implementación, incluyendo los subprocesos de IHC para la creación de sistemas digitales, needfinding y prototipado mencionados en el capítulo 2. De igual forma, dentro de la etapa de diseño se incluyen los elementos tomados de la teoría de flujo, descritos en el capítulo anterior, además de la comparación del proceso de desarrollo basado en IHC y desarrollos tradicionales.

### 4.1. Análisis

En esta primer etapa se buscó definir claramente el público al que irá dirigido el juego serio, para esto fue necesario realizar una investigación acerca de los padecimientos más frecuentes que se atienden en el INR. De igual forma, se empleó el proceso de needfinding, para obtener un mejor entendimiento de las necesidades de los usuarios, buscando enfocar el desarrollo en dichas necesidades.

#### 4.1.1. Needfinding

Con el fin de tener un mejor entendimiento de las necesidades de los usuarios de Acoli. A continuación, se siguen los pasos del proceso de needfinding aplicados en Acoli.

Se **identificó un grupo de usuarios y tareas de estudio**, para esto se analizaron diversos censos que se han realizado en México. Por ejemplo, el censo de (INEGI,2004), nos muestra que el tipo de discapacidad más frecuente es la motriz, con un 45.3%, la cual agrupa discapacidades músculo-esqueléticas y neurológicas. Además, se encontró que son los adultos y adultos mayores, los que en su mayoría presentan este tipo de discapacidad, siendo las principales causas la enfermedad y los accidentes.

Por otra parte, una investigación realizada en el INR, revela que de un total de 2271 pacientes, el 30.8% (699) sufren de trastornos de tejidos blandos<sup>1</sup> y de la sinovia, de los cuales, según sub grupos (ver Tabla 4.1), la capsulitis adhesiva del hombro ocupa el quinto lugar. Estas personas son comúnmente canalizados con terapistas ocupacionales.

---

<sup>1</sup>Todos los tejidos corporales no óseos, como los músculos, la grasa, el tejido fibroso, los vasos sanguíneos o cualquier otro tejido conjuntivo del cuerpo (GREEN,2016)

Tabla 4.1: TRASTORNOS DE LOS TEJIDOS BLANDOS SEGÚN SUB GRUPOS, INR, 2011.

TRASTORNOS DE LOS TENDONES Y DE LA SINOVIA		
DIAGNOSTICOS	FRECUENCIA	PORCENTAJE
SÍNDROME DEL MANGUITO ROTATORIO	317	13.96 %
TENOSINOVITIS	90	3.96 %
SÍNDROME DE ABDUCCIÓN	88	3.87 %
RUPTURA ESPONTÁNEA DE LA SINOVIA Y DEL TENDÓN	79	3.48 %
CAPSULITIS ADHESIVA DEL HOMBRO	61	2.69 %
TENDINITIS DEL BICEPS	46	2.03 %
SINOVITIS	14	0.62 %
OTROS TRASTORNOS DE LA SINOVIA Y DEL TENDÓN	4	0.18 %

Fuente: Sistema de Informática INR 2011

Elabora: Unidad de Vigilancia Epidemiológica Hospitalaria

Con base en estos censos se reduce al grupo de usuarios a aquellos pacientes del INR que sufren de capsulitis adhesiva. Como parte del grupo de usuarios, se incluirán algunos pacientes sanos, es decir, pacientes que no sufren de capsulitis adhesiva, con fines de evaluación del videojuego.

Por otro lado, la principal tarea de estudio es en si el ejercicio seleccionado, de los muchos que se realizan durante las sesiones de terapia ocupacional del INR, que en este caso, se seleccionó el ejercicio de escalera de pared, que por su sencillez nos permite crear la temática del juego. Es importante plasmar en su mayoría los movimientos involucrados del ejercicio, para así obtener los resultados médicos esperados, sin causar un daño al paciente que se encuentre realizando la terapia. Como parte de este estudio, se requiere analizar a detalle la siguiente información:

- La correcta ejecución del ejercicio, desde el inicio hasta el final del mismo.
- Las variaciones en el tiempo de ejecución del ejercicio.
- La medición del rango de movimiento de los pacientes, previa al inicio del ejercicio.

El siguiente paso es **crear hipótesis sobre la actividad**, en este caso, para el ejercicio de escalera de pared. Algunas de las hipótesis que se crearon fueron las siguientes:

- El ejercicio de escalera de pared se combina con otros ejercicios, durante una sesión de terapia.
- La escalera de pared se puede subir o bajar, según sean las capacidades de cada paciente.
- La duración del ejercicio va a depender del criterio del terapeuta, según las capacidades de cada paciente.
- Antes de iniciar la terapia, el terapeuta mide el rango de movimiento del hombro que tiene el paciente.
- En el ejercicio de escalera de pared, el salto entre escalones se hace de uno en uno.

Para corroborar dichas hipótesis, se tuvo que **reunir información acerca de la actividad** (tercer paso en el proceso de needfinding). En primera instancia se revisaron algunos artículos médicos referentes a terapias de extremidades exteriores (TARANG, 2014), (WILLIAM), etc., para conocer los diferentes ejercicios que se realizan y los padecimientos que tratan. De igual forma, se revisaron diversos video internos del INR, del ejercicio de escalera de pared, con el fin de visualizar la forma correcta de realizar el ejercicio y algunos detalles del mismo.

Además, con el propósito de fortalecer dicha información, se realizaron múltiples visitas a la área de terapia ocupacional del INR, en las cuales se pudo visualizar la ejecución del ejercicio de forma presencial. De este estudio se recabó la siguiente información:

- La duración de una sesión de terapia es de aproximadamente 45 minutos.
- Durante una sesión se combinan diferentes tipos de ejercicios.

- Antes de iniciar el ejercicio, se mide que tanto puedo mover el paciente el hombro sin sentir dolor.
- En el ejercicio de escalera de pared, se utiliza la medición del rango de movimiento del paciente, para fijar el escalón máximo al que el paciente puede subir.
- El ejercicio de escalera de pared, consiste en subir y bajar por los siete escalones de la escalera, sosteniendo con ambas manos un bastón de madera o PVC.
- En el ejercicio de escalera de pared, el salto entre escalones puede darse entre todos los escalones posibles (p. ej. un salto válido puede darse del primer al cuarto escalón), siempre y cuando no se supere el escalón máximo.
- Para el ejercicio de escalera de pared, la escalera está fija en una posición en la pared, por lo que habrá situaciones en las que personas, por sus características, no puedan realizar el ejercicio.
- Conforme avanzan las terapias, se estimula al paciente a realizar un pequeño esfuerzo para superar su rango de movimiento.

Por último, se **identificaron oportunidades del producto**, es decir, en que puntos se podía mejorar la actividad, en este caso, el ejercicio de escalera de pared. A continuación, menciono aquellas posibles mejoras que se encontraron para el desarrollo de Acoli y de Macoli:

- El desarrollo de Acoli, ya se considera como una mejora a la actividad, por el hecho de volverla más lúdica, al realizarla por medio de un videojuego.
- La creación de una estructura favorece a que un público mayor tenga acceso a la realización de la actividad, por el hecho de que ésta se puede ajustar según las características de cada paciente. Además de permitir a personas que se encuentran en silla de ruedas, también puedan participar en la actividad.
- De igual forma, la inclusión de ruedas en la parte inferior de la estructura facilita su portabilidad.
- El sistema Macoli, en comunicación con Acoli, permite almacenar toda la información registrada de una sesión dentro de una base de datos, con el fin de generar datos históricos que sean de ayuda en las sesiones posteriores.
- El sistema Macoli permite a los terapeutas llevar registro de todos los pacientes que se encuentran a su cargo, además de permitir crear su propia cuenta a cada terapeuta.

Una vez realizada toda la investigación referente a la actividad, el siguiente paso fue hacer un análisis de la misma, con el fin de retomar aquellas características de la actividad e incluir aquellas que se consideren de mayor importancia para el desarrollo de Acoli. De igual forma, se definió la temática del juego.

#### 4.1.2. Análisis de requerimientos

Los requerimientos son determinados por la teoría de la experiencia de usuario óptima, por los procesos de needfinding y prototipado, además de algunas de las características de otros juegos serios.

A continuación se numeran los requerimientos obtenidos para el desarrollo de Acoli, según el estándar internacional de Especificación de Requerimientos IEEE830 (MONTEFERRER,2001).

##### *Ambiente físico*

- Acoli funciona como una aplicación de Windows, para lo cual solo se requiere una laptop o una PC de escritorio con conexión bluetooth.
- Para el control de Acoli se requiere del dispositivo, que se asemeja a la escalera de pared de siete escalones del INR, y una conexión eléctrica cerca para alimentarlo.

##### *Usuarios y factores humanos*

- Los usuarios de Acoli son principalmente los pacientes del INR que sufren de hombro congelado<sup>2</sup>.

<sup>2</sup>El uso de Acoli, por parte de personas sanas, será solo con fines de evaluación del juego.

- 
- Acoli debe contar con una fase de calibración previa, que se adapte a las características físicas de cada usuario.
  - En la fase de calibración se ajusta la altura de la escalera de acuerdo a la estatura del paciente. De igual forma, se ajusta el nivel o número de escalón al cual el paciente es capaz de alcanzar sin sentir dolor.

#### ***Funcionalidad***

- La dinámica de juego de Acoli debe ajustarse al ejercicio de escalera de pared que se realiza en el INR, reproduciendo todos los movimientos, el número de repeticiones y los tiempos de ejecución.
- Acoli debe contar con tres diferentes tiempos de juego (1 min, 2 min y 3 min).
- Acoli debe contar con tres diferentes dificultades (Fácil, Medio y Difícil).
- Acoli debe contar con siete diferentes niveles, donde cada nivel corresponde a cada uno de los escalones de la escalera, es decir, en el nivel 1 solo se ocupa el primer escalón, en el nivel 2 se ocupan el primer y segundo escalón, así sucesivamente.
- La pantalla de Configuración tendrá valores por defecto; en caso de que el usuario no cambie nada y pase a la siguiente pantalla, se tomarán los valores predeterminados.
- Acoli debe contar con una fase de entrenamiento, previa al inicio del juego.
- En la fase de entrenamiento, no se debe permitir al usuario continuar con el siguiente paso, hasta haber concluido con éxito el paso anterior.

#### ***Recursos***

- Para el uso de Acoli es necesario contar con los siguientes recursos:
  - El dispositivo de escalera de pared y el bastón de PVC.
  - Una computadora, ya sea de escritorio o laptop, que tenga como sistema operativo Windows 8.0 o superior, y conexión a bluetooth.
  - De igual forma, debe contar con la aplicación de Macoli instalada en la computadora.
  - Un televisor, el cual deberá ir conectado a la computadora, para comodidad de los usuarios.
- Además, se requiere de una conexión eléctrica cerca, para conectar los recursos anteriores.

#### ***Datos***

- Los valores obtenidos en la calibración los obtiene Macoli en una fase previa a la ejecución de Acoli, los cuales se envían como datos de entrada al momento de ejecutar el videojuego.
- Al presionar el botón de volver a jugar o el de salir al menú inicial se debe guardar toda la información del juego en la base de datos.
- La información que se debe guardar en la base de datos es la siguiente:
  - El id de la sesión.
  - El id de la partida.
  - El puntaje obtenido durante una partida.
  - El tiempo que duró la partida.
  - El nivel en el que se jugó la partida.
  - La dificultad con la que se jugó la partida.
- Si se juegan múltiples partidas, se debe guardar cada partida en un registro diferente de la base de datos.

Con base en los requerimientos antes descritos, se da inicio con el desarrollo formal de Acoli. En las siguientes secciones, se describe todo el proceso de desarrollo Acoli, iniciando con el diseño

---

de la base de datos, de las clases que modelaran los objetos del videojuego, así como la temática del videojuego y todas las pantallas que se incluirán en él.

## 4.2. Diseño

En esta fase del proceso de desarrollo se describen los temas, estilos visuales, y diseño de las pantallas, para después integrarlos y generar los prototipos de pantallas, tomando en cuenta en todo momento, la temática del juego descrita a continuación.

### 4.2.1. Diagramas UML

Para el presente proyecto, se decidió solo incluir dos diagramas UML, del Proceso Unificado, el Diagrama de clases y el Diagrama de bases de datos, ya que por la orientación a agilizar el proceso de desarrollo son de ayuda para la fase de construcción.

#### Diagrama de clases

El primer diagrama que utilizaremos será el de clases [Figura 4.14], éste nos ayudará a visualizar las relaciones entre las clases que involucran el sistema, y entender de manera breve como funciona el videojuego. En el caso de Unity, casi todos los componentes 3D que se muestran en pantalla se manejan como una clase, las cuales no tiene caso incluirlas en el presente diagrama, por esta razón, solo se van a incluir, aquellas clases de mayor importancia así como atributos y métodos relevantes, es decir, aquellos que se encargan del manejo y funcionamiento del juego, así como aquellos que se encargan de la comunicación con el dispositivo de control y con la base de datos. Cabe mencionar, que algunos de los scripts hacen lo mismo, pero para diferentes componentes del juego, por lo que no se incluyen en el diagrama de clases, debido a que su comportamiento es el mismo.

#### Diagrama de base de datos

El segundo diagrama que utilizaremos, es un diagrama de base de datos, este diagrama se puede modelar haciendo uso de un diagrama de clases UML. En donde las clases y los atributos hacen referencia a las entidades lógicas y sus atributos. Este diagrama nos va a ayudar a visualizar el modo de almacenar los datos que son de importancia para el usuario. Para el presente proyecto, por parte de Acoli solo vamos a almacenar datos obtenidos del juego, como los aciertos que se tuvieron, el tiempo, nivel, etc. Además de estos datos, también se guarda información que es de importancia para Macoli, es decir, toda la información de los pacientes, con el fin de que se pueda generar un histórico de cada paciente, y ver el progreso que se tiene conforme las sesiones avanzan.

En la [Figura 4.15], se muestra el diagrama Entidad Relación de Macoli, en donde la entidad llamada *juego* es la que corresponde a la tabla de Acoli, es decir, que las columnas de la tabla, se van a llenar con la información que se mande del juego, la información es la siguiente:

- Id\_ Juego,
- Id\_ Sesion,
- Duracion,
- Dificultad,
- Escalon,
- Estado,
- Cancion,

- Puntuacion,
- MaxPuntuacion,
- Errores

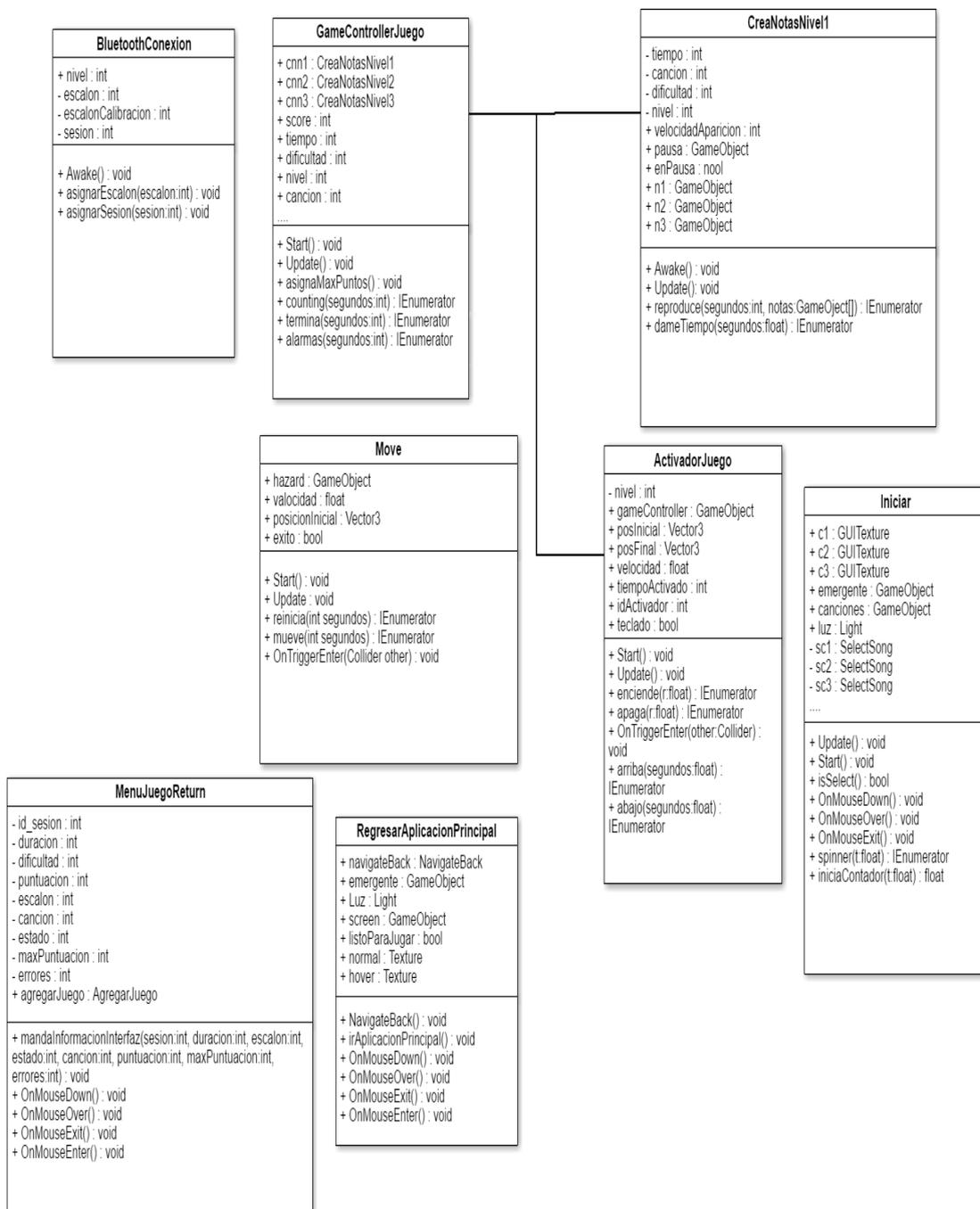


Figura 4.1: Diagrama de clases Acoli

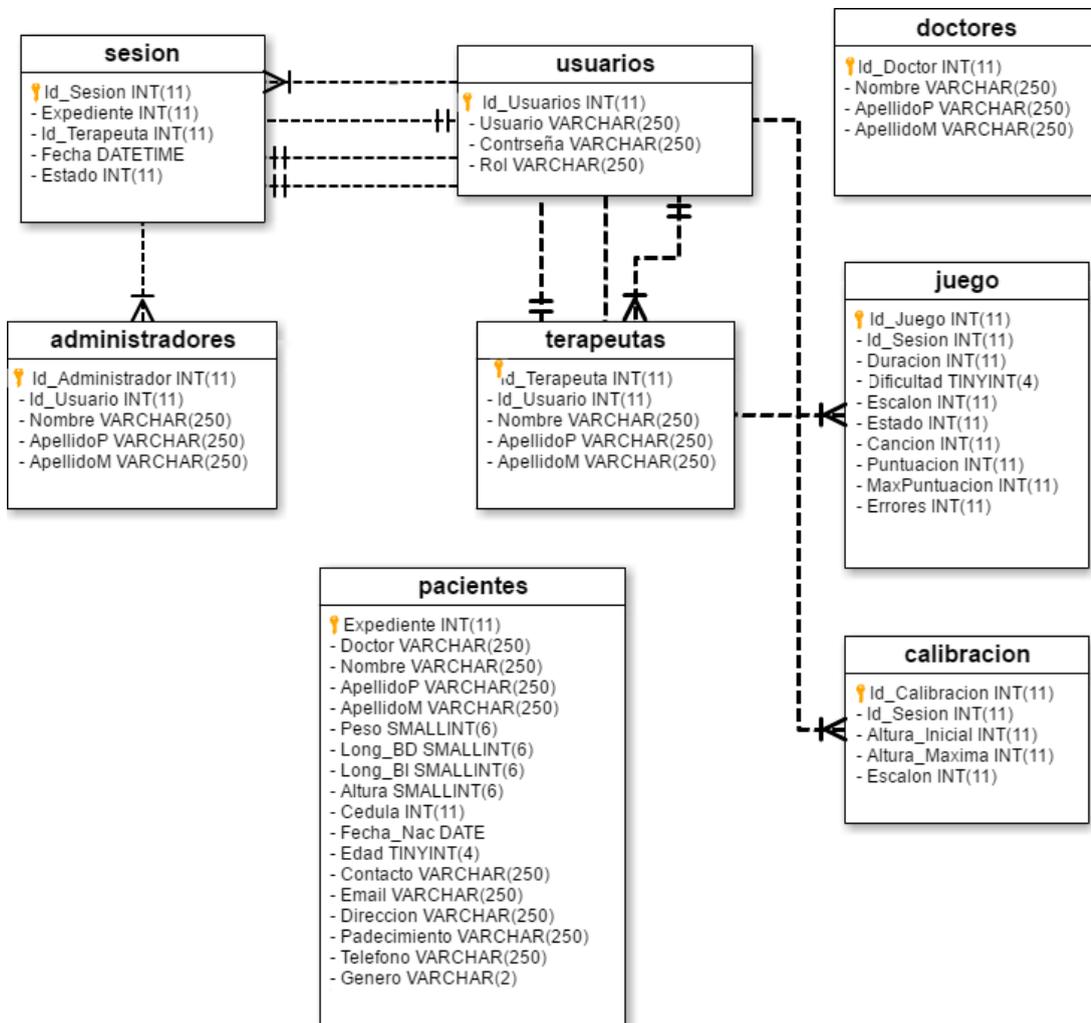


Figura 4.2: Diagrama de Base de Datos de Macoli

#### 4.2.2. Temática del juego

Con base en la información obtenida sobre la actividad, se pensó de qué manera reproducirla como parte de la temática del videojuego, tomando en cuenta la información del tipo de usuarios que harán uso de Acoli, además de los aspectos de la teoría de flujo y de los juegos serios, vistos en capítulos anteriores. A continuación, se da una explicación de dicha temática y los aspectos que se tomaron en cuenta para su creación.

Primero se tomó en cuenta el grupo de usuarios, en este caso, pacientes que sufren de hombro congelado, que en su mayoría son mujeres de 55 a 59 años y que se dedican a actividades del hogar. En un principio, se pensó en crear una temática relacionada con actividades del hogar, ya que según las características de un juego serio, la temática debe estar asociada con actividades del mundo real. El problema con dicha situación fue la dificultad de reproducir el ejercicio en una actividad cotidiana, sin que ésta se tornara aburrida. Por lo que se dio mayor importancia a buscar una actividad que permitiera realizar la actividad de forma

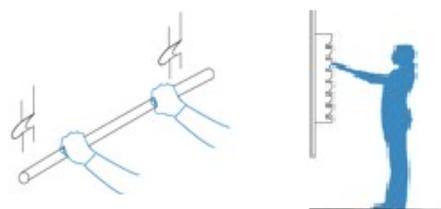


Figura 4.3: Posición correcta de posicionamiento en escalón

---

adecuada, sin dejar de tomar en cuenta el aspecto de entretenimiento.

De este modo, fue que se pensó en desarrollar un juego musical, ya que la música es del agrado de casi toda la población, sin importar la edad, y nos permite desarrollar un juego que se adapte perfectamente al ejercicio seleccionado, en donde, por cada escalón de la escalera se pinta una línea del pentagrama musical. Al final de cada línea se encuentra un “activador”, cuya función será ir atrapano notas musicales, que irán saliendo poco a poco del lado izquierdo de la pantalla. Cada uno de los activadores, se encenderá cuando el bastón de PVC, se coloque correctamente sobre el escalón correspondiente, es decir, cuando ambos extremos del bastón estén bien colocados sobre el escalón [Figura 4.3]. Una nota se considera atrapada si el activador está encendido en el momento en que la nota pasa sobre él, de ser así, se escuchará el sonido correspondiente a la nota. Por cada nota atrapada se otorgará un puntaje al usuario, de igual forma, por cada nota que no sea atrapada se afectarán los puntos de vida. El objetivo del juego será atrapar el mayor número de notas en el tiempo establecido.

En una etapa previa al prototipado, se enlistan los requerimientos encontrados con lo que respecta a las interfaces de Acoli, con base en la temática del juego y los requerimientos anteriormente mencionados.

### ***Interfaces***

- Acoli, en su pantalla de inicio, debe contar con un menú de opciones, que permita navegar entre las diversas pantallas
- Acoli debe contar con una pantalla de configuración, que permita modificar el tiempo, el nivel y la dificultad.
- En la pantalla de configuración, el usuario podrá regresar al menú inicial, o pasar a la pantalla de selección de canciones.
- En la pantalla de configuración, el ajuste del nivel solo permitirá subir a lo más un nivel extra, del máximo definido en la calibración.
- En la pantalla de selección de canciones, el usuario podrá escoger entre las canciones existentes, Para Elisa( Beethoven), Otoño(fragmento de las 4 estaciones de Vivaldi) y Jingle Bells(James Pierpont).<sup>3</sup>.
- En la pantalla de selección de canciones, el usuario podrá regresar a la pantalla de configuración o pasar a la pantalla de juego.
- No se debe permitir el acceso a la pantalla de juego, hasta que se seleccione una canción.
- En la pantalla de la fase de entrenamiento, se debe mostrar de forma clara, y de preferencia haciendo uso de apoyos visuales, la forma de jugar del videojuego.
- En la pantalla de fase de entrenamiento, se debe indicar al usuario cuando se ha concluido una fase de forma exitosa, de no finalizar de forma exitosa, no se permitirá pasar a la siguiente fase hasta haber finalizado con éxito.
- Antes de dar inicio el juego, se debe dar un banderazo de salida, que indique al usuario que está por empezar a jugar.
- Debe ser posible pausar el juego, al hacerlo se detendrá el juego y se mostrará un menú que permita salir del juego, reiniciar el juego o continuar jugando.
- Si se escoge la opción de seguir jugando, se reanudará el juego, dando un breve espacio de tiempo, para que el usuario se prepare.
- En la pantalla de juego, debe mostrarse lo siguiente:
  - El tiempo transcurrido.
  - La puntuación acumulada.
  - El nivel en el que se está jugando.

---

<sup>3</sup>se utilizaron canciones de música clásica, para evitar problemas de derechos de autor

- 
- Los puntos de vida total y restante.
  - El botón de pausa.
  - En la pantalla de juego, se le debe indicar al usuario cuando cometa un error o un acierto.
  - Al terminar el juego se debe mostrar un menú que permita volver a jugar o salir al menú inicial.

### 4.2.3. Prototipado

Como parte del proceso de diseño de IHC, en esta sección se lleva a cabo todo lo referente al prototipado; siguiendo los pasos mencionados en el capítulo II, y utilizando el modo de prototipado rápido.

El primer paso del prototipado, fue hacer **identificación de requerimientos básicos** con lo que respecta a la interfaz, para lo cual, bastó con retomar los requerimientos de interfaces obtenidos en la sección anterior, y seleccionar aquellos posibles para mostrar en el prototipo. A continuación, se mencionan el total de pantallas que se definieron e incluyeron posteriormente en Acoli:

1. Pantalla de menú inicial
2. Pantalla de fase de entrenamiento
3. Pantalla de configuración
4. Pantalla de selección de canción
5. Pantalla de juego

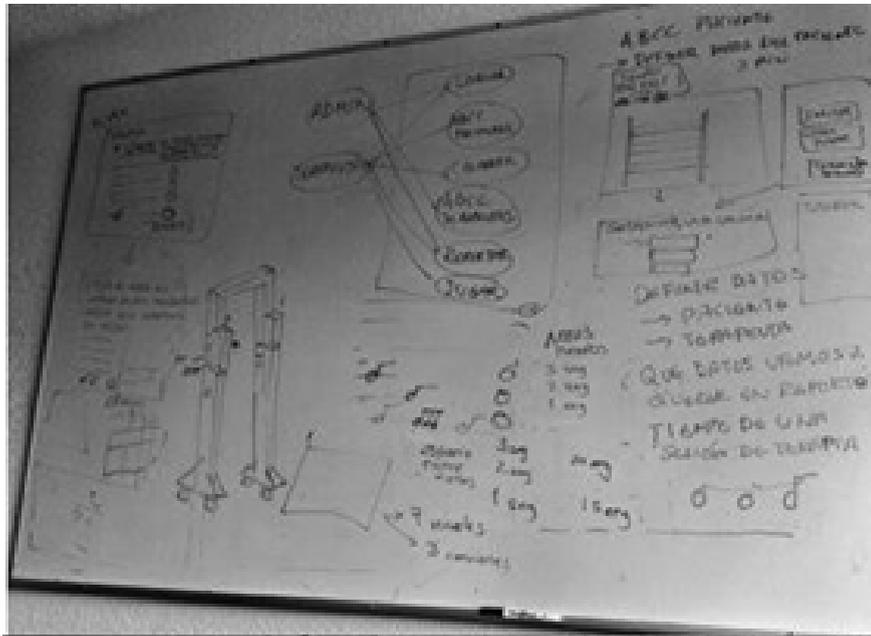
Conociendo el número total de pantallas se plasmaron los requerimientos de interfaz obtenidos en su pantalla correspondiente, los cuales se enlistan a continuación:

- En la pantalla de Menú inicial, el menú permite escoger de entre las siguientes tres opciones:
  1. Ir a la pantalla de Configuración.
  2. Ir a la pantalla de Fase de entrenamiento.
  3. Salir de Acoli.
- En la pantalla de Configuración debe mostrarse, de forma clara, las opciones para aumentar y disminuir el nivel, la dificultad y el tiempo de juego, además de las opciones para regresar a la pantalla de Menú inicial y avanzar a la pantalla de Selección de canción.
- Los valores predeterminados que se deben mostrar en la pantalla de Configuración son los siguientes:
  1. Tiempo - 1 minuto.
  2. Dificultad - Fácil.
  3. Nivel - El máximo establecido en la fase de calibración.
- En la pantalla de Fase de Entrenamiento se debe explicar en cada fase, de manera detallada, la forma de jugar, haciendo uso de elementos visuales y texto.
- Una vez concluida la Fase de Entrenamiento, se debe mostrar al usuario un menú en donde pueda volver a realizar el entrenamiento, o salir al menú inicial.
- En cada fase de entrenamiento, se debe mostrar la opción para salir al Menú inicial o regresar a la fase anterior.
- En la pantalla de Selección de canción, se debe mostrar al usuario, la lista de canciones disponibles, así como las opciones para regresar a la pantalla de Configuración o avanzar a la pantalla de Juego.
- Si el usuario no ha seleccionado una canción, e intenta avanzar a la pantalla de Juego, se debe

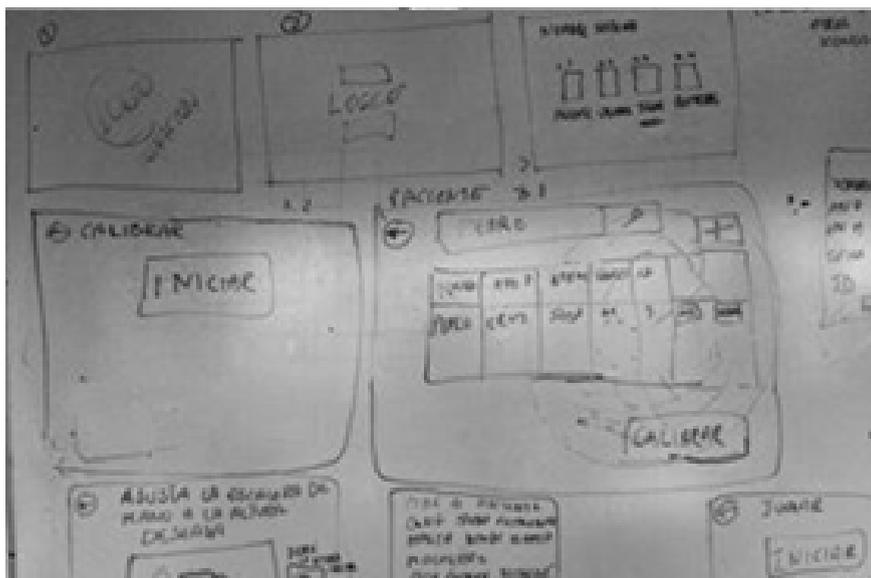
mostrar un mensaje, en donde se especifique que debe seleccionar una canción.

- Una vez que inicia la partida, el usuario podrá pausar el juego, lo que mostrará un menú que permita reanudar la partida, reiniciar la partida o salir a la pantalla de Menú inicial.
- Al finalizar el juego, se debe mostrar al usuario un mensaje, con el puntaje obtenido, además de las opciones para volver a jugar y la opción para salir a la pantalla de Menú inicial.

Una vez que se han identificado los requerimientos de interfaz, se inició con el **desarrollo del prototipo inicial**, que por ser un prototipado rápido, se llevó a cabo en un pizarrón mediante una lluvia de ideas, para después realizar la maquetación (ver Fig. 4.2).



(a)



(b)

Figura 4.4: Prototipado en pizarrón

---

Posteriormente, teniendo una idea más clara de los requisitos de la interfaz, se maquetaron los prototipos de cada pantalla, respetando los requerimientos definidos en la sección anterior. A continuación, se muestran cada una de las pantallas:

En la [Figura 4.3], se muestra la pantalla de Menú inicial, con el logo de Acoli, y tres botones que permiten navegar entre las diferentes pantallas.

1. Botón INICIAR (ver Número 1 de la Figura 4.3), al presionarlo nos envía a la pantalla de Configuración.
2. Botón AYUDA (Número 2 de la Figura 4.3), al presionarlo nos envía a la pantalla de Fase de entrenamiento.
3. Botón SALIR (Número 3 de la Figura 4.3), al presionarlo se muestra un mensaje donde pregunta al usuario, si está seguro de querer salir de Acoli; si se presiona SI, termina la ejecución de Acoli y regresa al sistema Macoli, si se presiona NO, cierra el mensaje y continúa en la pantalla de Menú inicial.

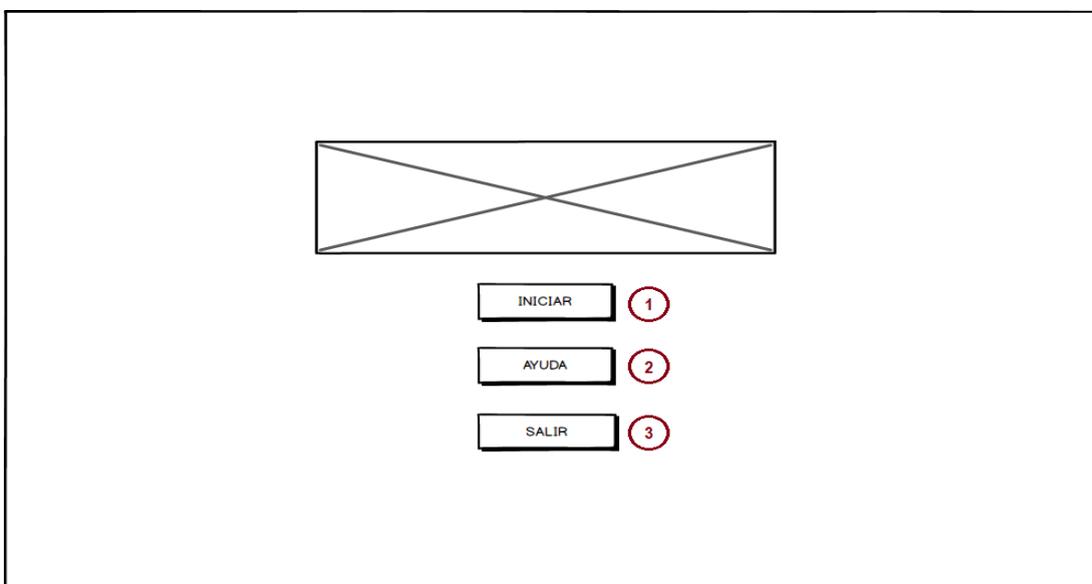


Figura 4.5: Prototipo de pantalla de Menú inicial

La siguiente pantalla, es la de Mensaje en Fase de entrenamiento<sup>4</sup> [Figura 4.4], la cual consta de tres elementos, que se describen a continuación:

1. Mensaje de explicación, en el cual se detalla el modo de juego de la fase de entrenamiento correspondiente. .
2. Botón JUGAR (ver Número 2 de la Figura 4.4), al presionarlo se oculta el mensaje de explicación y el botón de regreso al menú inicial, y da inicio a la fase de entrenamiento [Figura 4.5].
3. Botón REGRESO (ver Número 3 de la Figura 4.4), al presionarlo sale de la pantalla de Entrenamiento y regresa a la pantalla de Menú inicial.

---

<sup>4</sup>Cabe mencionar que, esta pantalla es similar en cada una de las fases de entrenamiento, por lo que basta con que se muestre una vez.

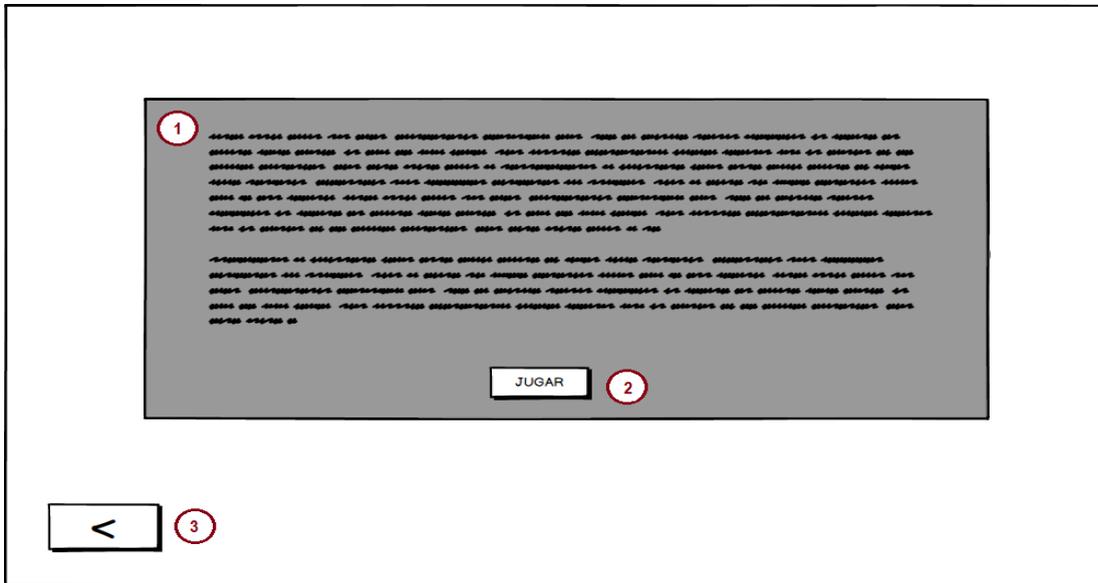


Figura 4.6: Prototipo de pantalla de Mensaje en Fase de entrenamiento

En la [Figura 4.5], se muestra la pantalla de la Fase I de entrenamiento, es decir, en esta fase solo se muestra una línea del pentagrama (ver Número 1 Figura 4.5) y un activador (ver Número 3 de la Figura 4.5). Una vez que inicia el juego, aparece una nota (ver Número 2 de la Figura 4.5), que llamaremos "nota sencilla", que empieza a avanzar de izquierda a derecha, hasta salir de la pantalla o colisione con el activador, la nota solo podrá colisionar con el activador cuando éste se encuentre activado; en el caso donde la nota sale de la pantalla, ésta volverá a aparecer al inicio de la línea del pentagrama y reiniciará su trayecto, hasta que logre colisionar con el activador.

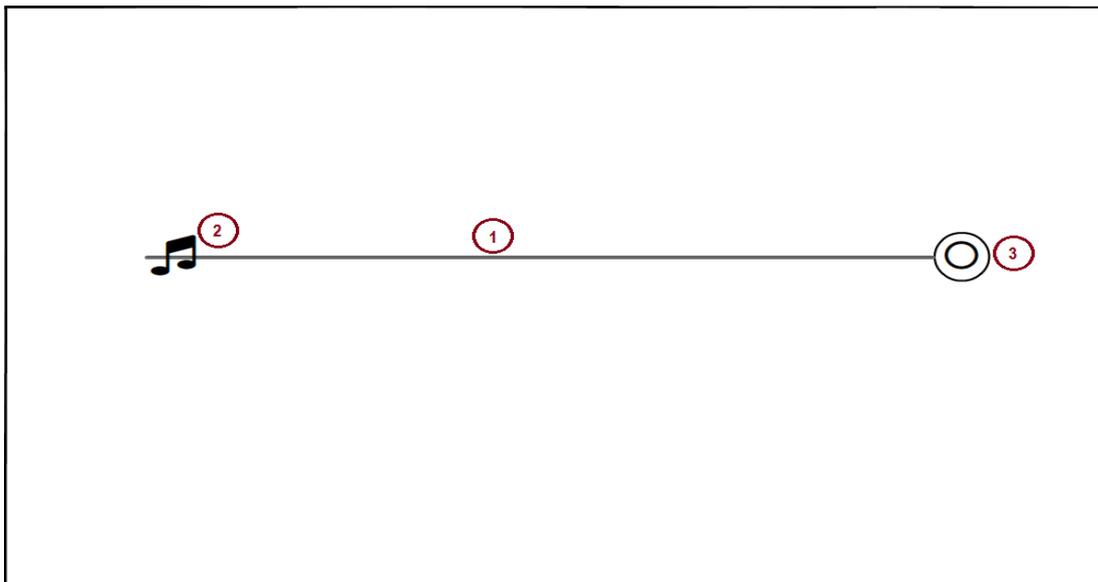


Figura 4.7: Prototipo de pantalla de Fase I de entrenamiento

La Fase I de entrenamiento finaliza en el momento en que la nota logra colisionar con el activador, que a su vez para poder ser activado requiere que el usuario coloque, de forma correcta, el bastón de PVC sobre el primer escalón de la escalera. De ser así, se muestra un mensaje de felicitación (ver Número 1 de la Figura 4.6), indicando al usuario que el entrenamiento se realizó de forma correcta. Además se

muestran dos botones:

1. Botón REPETIR (ver Número 2 de la Figura 4.6), al presionarlo vuelve a iniciar la fase I de entrenamiento.
2. Botón SIGUIENTE (ver Número 3 de la Figura 4.6), al presionarlo nos envía a la pantalla de Fase II de entrenamiento [Figura 4.7].

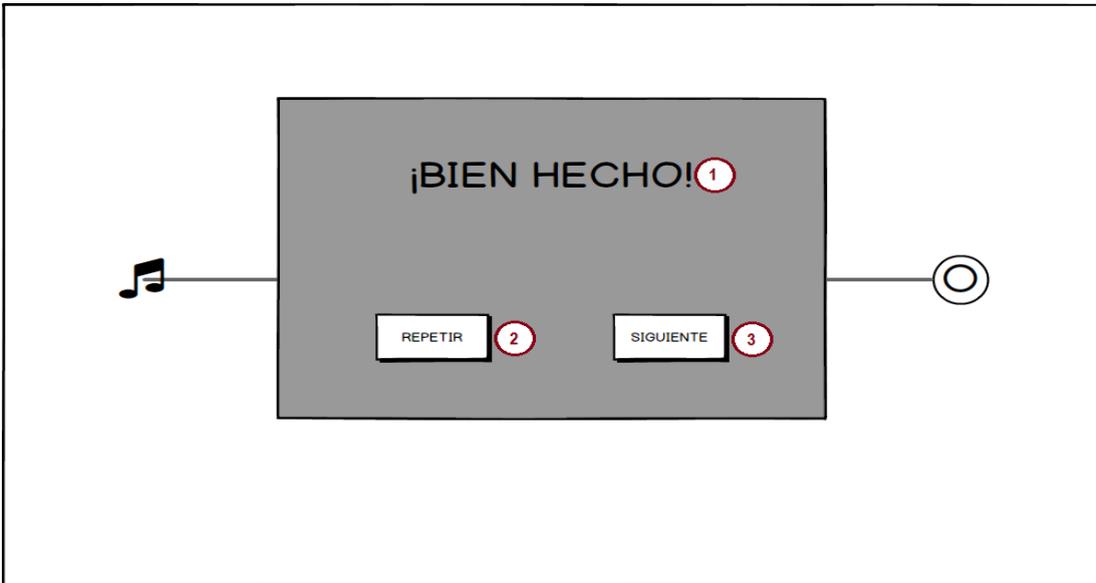


Figura 4.8: Prototipo de pantalla de Mensaje de éxito Fase I de entrenamiento

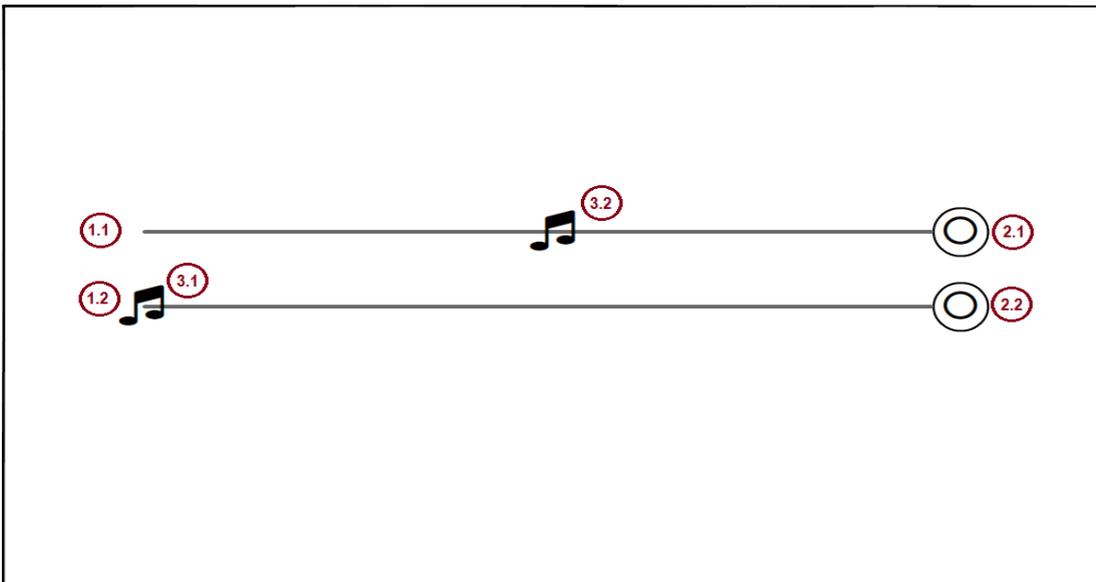


Figura 4.9: Prototipo de pantalla de Fase II de entrenamiento

Una vez concluida la Fase I de entrenamiento, y se ha presionado el botón siguiente (ver Número 3 de la Figura 4.6), se despliega la pantalla de Mensaje de Fase II de entrenamiento, con su mensaje de instrucciones correspondiente. La fase II consiste en atrapar dos notas de manera secuencial, de modo que el usuario, pueda practicar el cambio entre escalones, por lo que, en esta pantalla se cuenta con lo

---

siguiente:

1. Dos líneas del pentagrama (ver Números 1.1 y 1.2 de la Figura 4.7).
2. Dos activadores, uno por cada línea (ver Números 2.1 y 2.2 de la Figura 4.7).
3. Dos notas, una por cada línea del pentagrama (ver Números 3.1 y 3.2 de la Figura 4.7), que aparecen de manera secuencial dado un intervalo de tiempo determinado.

En el caso donde solo se atrape una de las dos notas, la nota que no haya sido atrapada, repetirá su recorrido, hasta que sea capturada por el usuario. Una vez que el usuario logre atrapar ambas notas, se mostrará un mensaje de felicitación, parecido al de la [Figura 4.6]. Donde, el botón REPETIR (ver Número 2 de la Figura 4.6) reiniciará la Fase II del entrenamiento, y el botón SIGUIENTE (ver Número 3 de la Figura 4.6) nos mandará a la Fase III del entrenamiento [Figura 4.8].

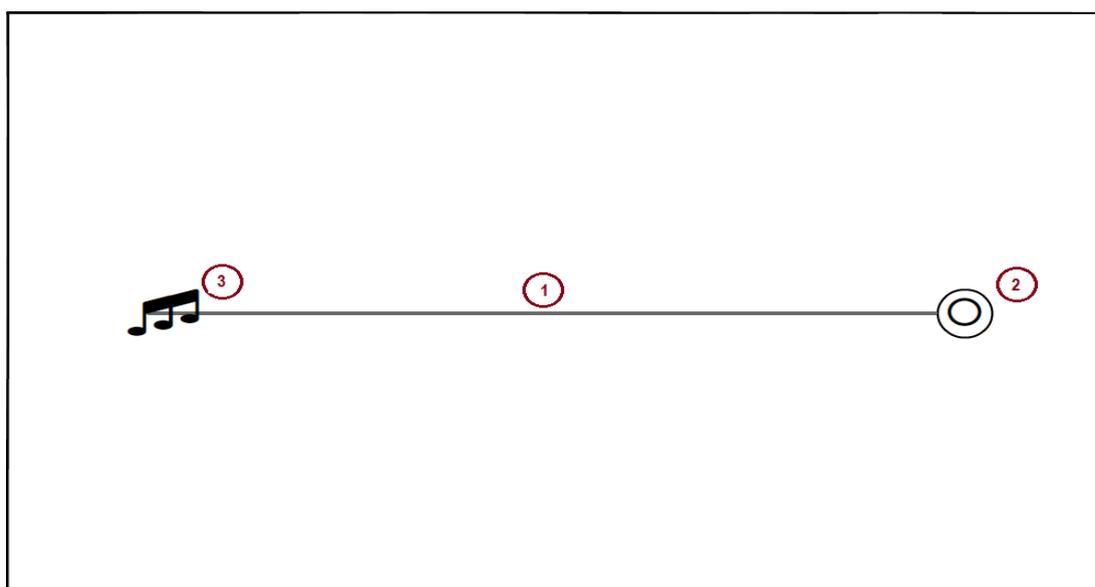


Figura 4.10: Prototipo de pantalla de Fase III de entrenamiento

En la fase III del entrenamiento, se incluye otro tipo de notas, llamadas "notas largas", que a diferencia de las notas sencillas, éstas tienen la característica de tener un sonido más prolongado, además de que su puntaje es mayor, siempre y cuando el bastón se mantenga sobre el escalón el tiempo necesario para que la nota pase por completo, de no dejarse el tiempo completo solo se contará la mitad de los puntos. El propósito de esta fase, es que el usuario se familiarice con este tipo de notas. Por lo tanto, la pantalla de Fase III de entrenamiento, solo se incluye lo siguiente:

1. Una línea del pentagrama (ver Número 1 de la Figura 4.8).
2. Un activador (Número 2 de la Figura 4.8).
3. Una nota larga (Número 3 de la Figura 4.8).

De igual forma que en las fases anteriores, la nota aparecerá en el inicio de la línea del pentagrama, y realizará un recorrido de izquierda a derecha, hasta que colisione con el activador, en cuyo caso se mostrará un mensaje de felicitación, similar al de las fases anteriores (ver Número 1 de la Figura 4.9), con la diferencia de que ahora el botón SIGUIENTE, será sustituido por el botón SALIR (Número 3 de la Figura 4.9) el cual nos regresará a la pantalla de Menú inicial, de igual forma, existe un botón REPETIR (ver Número 2 de la Figura 4.9), que nos permite volver a realizar la fase III; en el caso donde no colisione con el activador, reiniciará su recorrido.

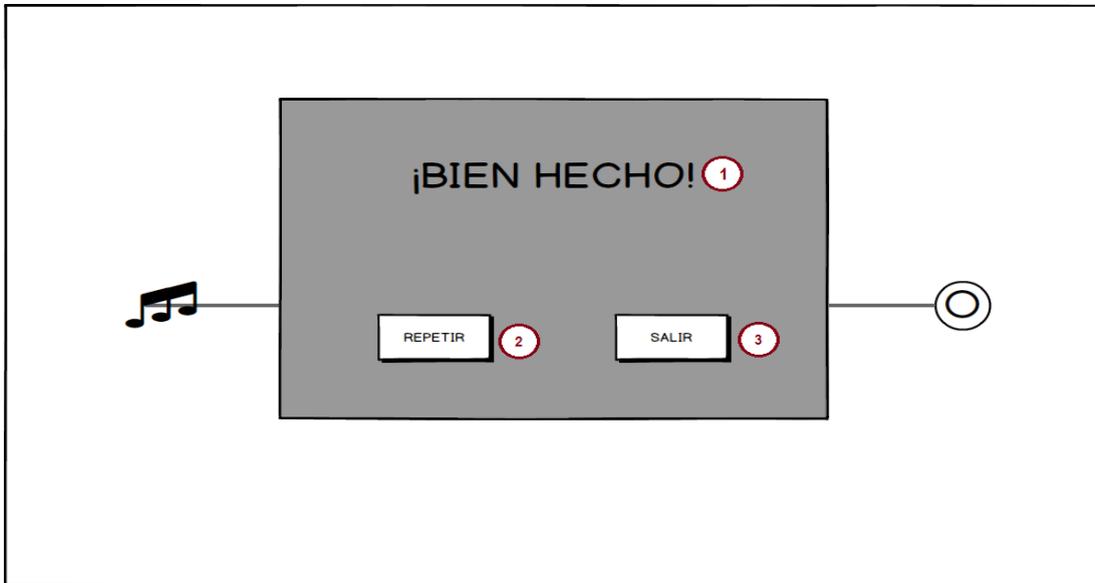


Figura 4.11: Prototipo de pantalla de Mensaje de Fase III de entrenamiento

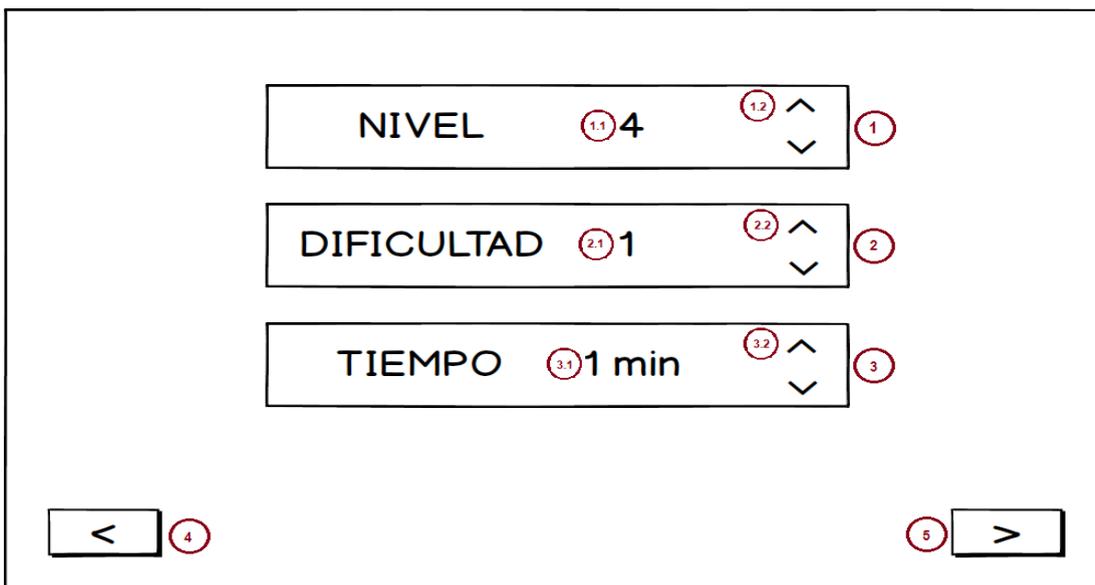


Figura 4.12: Prototipo de pantalla de Configuración

Una vez concluidas las tres fases de entrenamiento, se regresará al usuario a la pantalla Menú inicial, en donde podrá empezar a jugar, dando click en el botón INICIAR (ver Número 1 de la Figura 4.3), al hacer esto, se abrirá la pantalla de Configuración [Figura 4.10], la cual contiene los siguientes elementos:

1. Panel NIVEL (ver Número 1 de la Figura 4.10):
  - 1.1. Mensaje de texto, que indica el nivel seleccionado.
  - 1.2. Botones para subir y bajar el nivel, en el caso donde se ha seleccionado el nivel máximo, el botón SUBIR se deshabilita. De igual forma, si se llega al nivel mínimo, el botón BAJAR se deshabilita.

2. Panel DIFICULTAD (ver Número 2 de la Figura 4.10):
  - 2.1. Mensaje de texto, que indica la dificultad seleccionada.
  - 2.2. Botones para subir y bajar la dificultad, en el caso donde se ha seleccionado la dificultad máxima, el botón SUBIR se deshabilita. De igual forma, si se llega a la dificultad mínima, el botón BAJAR se deshabilita.
3. Panel TIEMPO (ver Número 3 de la Figura 4.10):
  - 3.1. Mensaje de texto, que indica el tiempo seleccionado.
  - 3.2. Botones para subir y bajar el tiempo, en el caso donde se ha seleccionado el tiempo máximo, el botón SUBIR se deshabilita. De igual forma, si se llega al tiempo mínimo, el botón BAJAR se deshabilita.
4. Botón SIGUIENTE (ver Número 4 de la Figura 4.10), al dar click, nos manda a la pantalla de Selección de Canción [Figura 4.11].
5. Botón REGRESA (ver Número 5 de la Figura 4.10), al dar click, nos regresa a la pantalla de Menú Inicial.

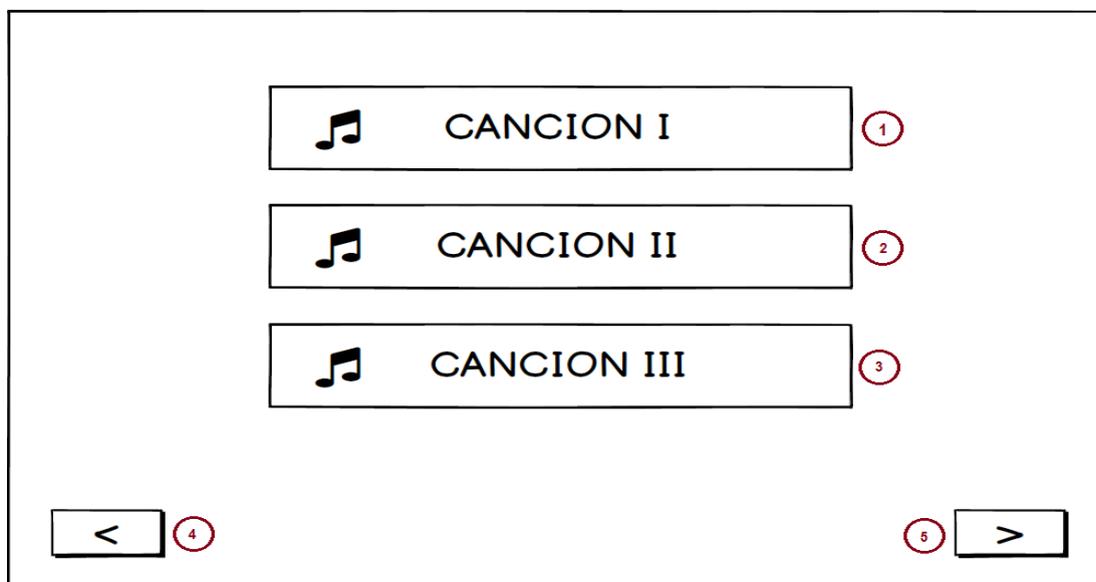


Figura 4.13: Prototipo de pantalla de Selección de Canción

En la pantalla de Selección de Canción [Figura 4.11], se muestra el repertorio de canciones<sup>5</sup>, de entre las cuales, el usuario solo podrá seleccionar una. Por lo que, en pantalla se muestra lo siguiente:

1. Canción I, primera canción del repertorio.
2. Canción II, segunda canción del repertorio.
3. Canción III, tercera canción del repertorio.
4. Botón ANTERIOR, al presionarlo nos regresa a la pantalla de Configuración [Figura 4.11].
5. Botón SIGUIENTE, éste permanecerá deshabilitado hasta que el usuario seleccione una canción del repertorio. Una vez habilitado, nos mandará a la pantalla de Juego [Figura 4.12] al hacer click sobre él.

<sup>5</sup>Para el presente proyecto, solo se contemplaron tres canciones

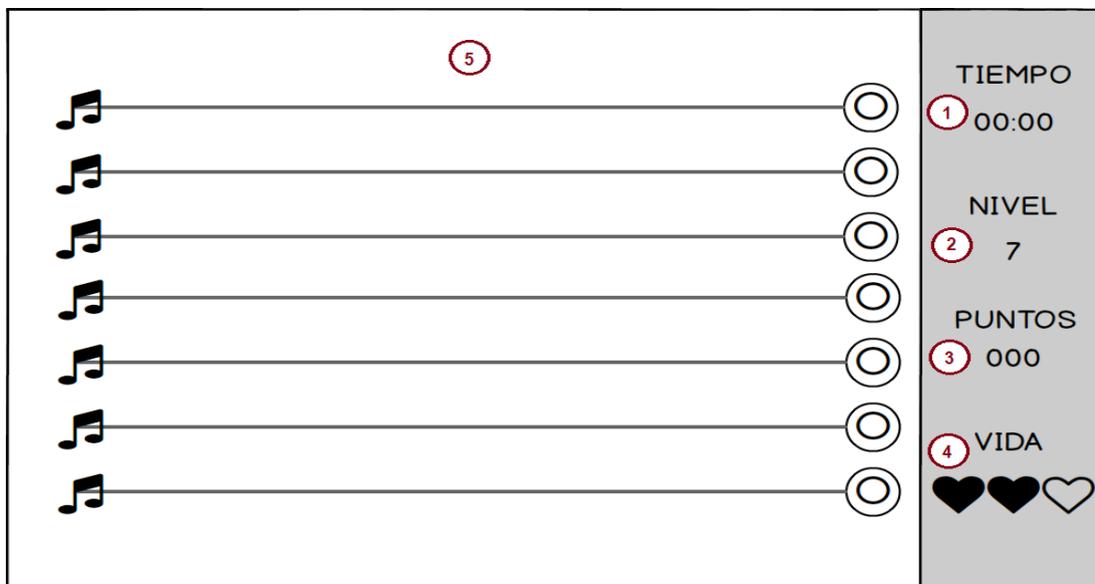


Figura 4.14: Prototipo de pantalla de Juego

En la pantalla de Juego [Figura 4.12], se lleva a cabo como tal la dinámica de Acoli, por lo que se muestran los siguientes: elementos, tomando en cuenta los requisitos descritos en los requerimientos:

1. Tiempo, se muestra de lado izquierdo el mensaje "TIEMPO", y abajo se muestra el tiempo con notación MINUTOS:SEGUNDOS, según lo seleccionado en la pantalla de Configuración, una vez que inicia el juego el tiempo empieza a correr de manera regresiva.
2. Nivel, se muestra de lado izquierdo el mensaje "NIVEL", y abajo se muestra el nivel seleccionado en la pantalla de Configuración.
3. Puntos, se muestra de lado izquierdo el mensaje "PUNTOS", y abajo se muestra un contador que inicia en cero, cada que se colisione una nota, el puntaje aumentará.
4. Vida, se muestra de lado izquierdo el mensaje "VIDA", y abajo se muestran tres corazones, cada que se pierda una nota se perderá la mitad de un corazón, es decir, que haya aparecido y concluya su recorrido sin haber sido colisionada con un activador. En el caso donde se pierdan todos los corazones, el juego habrá terminado.
5. Área de juego, se muestra de lado derecho todos los componentes del juego, es decir, las notas, las líneas de pentagrama y los activadores. El número de componentes va a depender del nivel seleccionado en la pantalla de Configuración.

Una vez terminado el tiempo, y el jugador aún tiene vidas, se da por terminada la partida, y se mostrará un mensaje como el que se muestra en la [Figura 4.13], dicho mensaje contiene la siguiente información:

1. Mensaje de felicitación, correspondiente al desempeño del usuario.
2. Puntuación obtenida contra puntuación total, es decir, los puntos que el jugador obtuvo, según su desempeño, en comparación contra el puntaje máximo a obtener.
3. Escala de desempeño visual, se muestra con estrellas el desempeño del jugador.
4. Botón REPETIR, al dar click sobre él, se guarda la información de la partida en la base de datos, y se inicia una partida nueva, con los mismos datos de configuración.
5. Botón SALIR, al dar click sobre él, se guarda la información de la partida en la base de datos, y sale a la pantalla de Menú Inicial.



Figura 4.15: Prototipo de pantalla Mensaje Termino de Juego

Si en una sesión, se juegan más de una vez, se guardará en la base de datos, la información de cada partida jugada. Una vez que se cierre Acoli, y se regrese a Macoli, se guardará en la base de datos, toda la información de la sesión.

En la siguiente sección, se lleva a cabo una comparación de las similitudes entre el proceso de desarrollo utilizado en esta tesis y las metodologías tradicionales de desarrollo de software.

### 4.3. Implementación

Todo lo referente a la implementación de Acoli, es decir, las partes de código que fueron necesarias para dar funcionamiento al videojuego y crear la comunicación entre el hardware y el software se incluyen en el Anexo A de este trabajo, de igual forma, en el Anexo B se mencionan las herramientas tecnológicas utilizadas, con una breve explicación de para que sirven y porque fueron de importancia para el desarrollo de Acoli.

#### 4.3.1. Dispositivo de control

En esta sección, se detalla el funcionamiento del dispositivo de control y el proceso de construcción que se llevó a cabo para lograr una buena sincronía entre el hardware y el software.

##### Comunicación hardware - software

Por el lado del hardware, se utilizó el siguiente material para la construcción del dispositivo de control:

1. Una placa Arduino Mega
2. 14 sensores de proximidad QRD1114
3. 14 piezas de acrílico
4. 14 circuitos impresos [Figura 4.16, a)]
5. Un dispositivo bluetooth HC-06
6. 14 resistencias de  $200\ \Omega$
7. 14 resistencias de  $5.6\text{k}\ \Omega$
8. Switch interruptor
9. Cable de alimentación de corriente

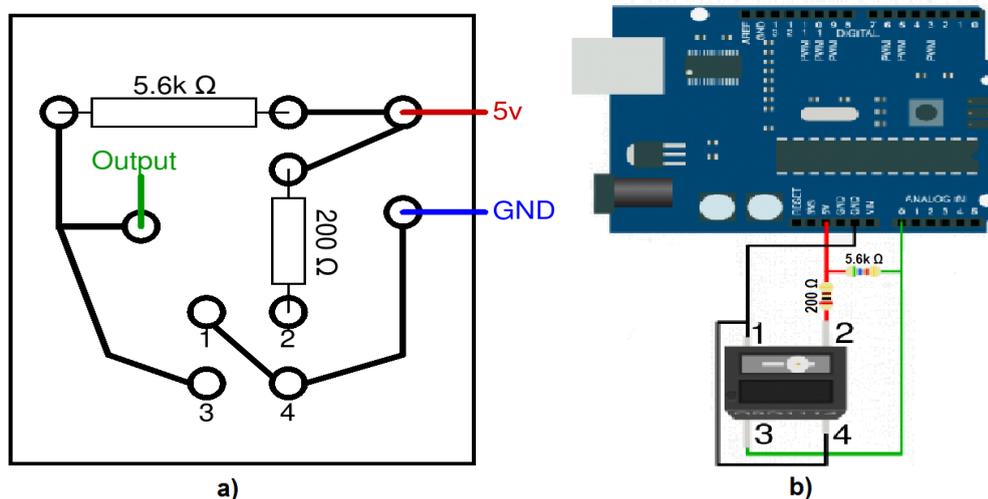


Figura 4.16: Circuito impreso y conexión con Arduino

La conexión del material se realizó de la siguiente manera, primero cada sensor de proximidad se

conectó sobre un circuito impreso, el cual se diseñó en ARES a modo de reproducir la conexión que se muestra en la [Figura 4.16, b)]. Donde la entrada de voltaje y de tierra las proporciona la placa arduino, de igual forma, se utiliza una entrada analógica para mandar la información obtenida por el sensor, es decir, que se utilizaron un total de 14 entradas analógicas para recibir la información de los 14 sensores, razón por la cual fue necesario usar una placa Arduino Mega. Una vez hecha esta conexión, para cada uno de los catorce circuitos impresos, éstos se distribuyeron sobre cada una de las piezas de acrílico, las cuales posteriormente se conectaron en serie y se colocaron sobre los escalones de la estructura de aluminio, es decir, dos sensores por cada nivel de la estructura.

Posteriormente, se realizó la conexión del dispositivo bluetooth de manera directa a la placa Arduino, como se muestra en la [Figura 4.17], al igual que la conexión del switch interruptor con el cable de alimentación, el cual brinda de energía a la placa Arduino para poder funcionar.

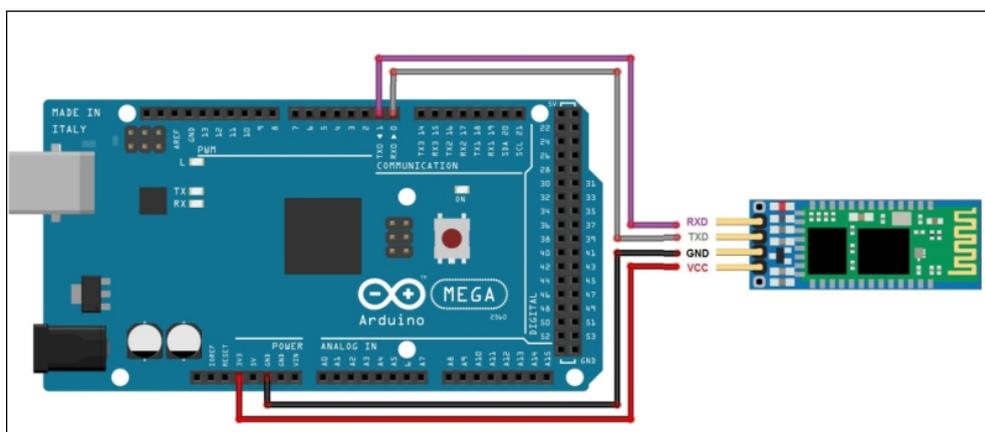


Figura 4.17: Conexión dispositivo bluetooth Arduino

El dispositivo bluetooth HC-06, por su parte, tuvo que configurarse para asignarle un nombre de dispositivo<sup>6</sup> y código de configuración, por medio de comandos AT<sup>7</sup>. Por su parte, para cambiar el nombre se ingresaron los siguientes comandos, por medio de una consola que brinda el IDE de Arduino:

**Enviar:** AT+NAME<Nombre>

**Respuesta:** OKsetname

De igual forma, para cambiar el código de configuración, se ingresaron los siguientes parámetros:

**Enviar:** AT+PIN<Pin>

**Respuesta:** OKsetPIN

Hecho esto, es posible vincular el dispositivo de control con cualquier PC o LAPTOP que cuente con conexión bluetooth. A la par que se ensamblaban los componentes del hardware del dispositivo de control, se fue desarrollando el software del dispositivo y de Acoli, el cual se encargará de lograr la correcta comunicación entre todos los componentes del hardware mencionados con Acoli. Dicho código se muestra en el Anexo A, del presente trabajo.

<sup>6</sup>Por defecto el nombre del dispositivo es "HC-06"

<sup>7</sup>Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un terminal modem

---

## 4.4. Experimento y Pruebas

Realizamos un experimento para determinar en que nivel Macoli afecta la experiencia de usuario en pacientes sanos, de esta manera la experiencia de usuario y la rehabilitación pueden ser estudiadas de forma separada.

### 4.4.1. Proceso de evaluación

Para la realización de las pruebas se invitó a un grupo de 40 personas, hombres y mujeres de entre 17 hasta 66 años. De dicha muestra, se formaron dos grupos de 20 usuarios, los que utilizaron Acoli, para hacer el ejercicio, y los que no lo usaron. Los pasos para la realización de la prueba, se enlistan a continuación:

1. Selección de forma aleatoria del tipo de ejercicio (CON o SIN Acoli).
2. Se da una introducción del ejercicio a realizar (Dependiendo del tipo de ejercicio).
3. Ejecución del ejercicio.
4. Entrega de cuestionario.

Una vez obtenido los resultados, se realizó el análisis estadístico. Para esto, se llevo a cabo una prueba T de Student para dos muestras independientes, utilizando el programa SPSS. Dicha prueba, consta de 4 pasos:

#### Paso 1 - Redacción de Hipótesis

**Hipótesis:** El total obtenido en *La escala de estado de flujo* de los usuarios que hicieron el ejercicio CON Acoli es mayor que los usuarios que hicieron el ejercicio SIN Acoli.

Posteriormente, se definen la hipótesis alterna ( $H_1$ ) y la hipótesis nula ( $H_0$ ).

$H_1$  = Existe una diferencia significativa entre la media de totales obtenidos del grupo CON y la media de totales obtenidos del grupo SIN.

$H_0$  = No existe una diferencia significativa entre la media de totales obtenidos del grupo CON y la media de totales obtenidos del grupo SIN.

#### Paso 2 - Determinar $\alpha$

El nivel alfa es el porcentaje de error que estamos dispuestos a correr en nuestra prueba estadística, el cual lo definiremos así:  $\alpha = 0.05$

#### Paso 3 - Lectura del P-Valor

Para poder asegurar que existe una diferencia significativa es necesario revisar lo siguiente:

- **Normalidad:** se debe corroborar que la variable aleatoria en ambos grupos se distribuye de manera normal. Para esto se realiza la prueba de Kolmogorov-Smirnov. El criterio para determinar si la variable aleatoria se distribuye normalmente es:

*Si el P-Valor  $\geq \alpha$  Se acepta  $H_0$  . Los datos provienen de una distribución normal.*

*Si el P-Valor  $< \alpha$  Se acepta  $H_1$  . Los datos NO provienen de una distribución normal.*

- **Igualdad de varianzas:** Se debe corroborar la igualdad de varianzas entre los grupos. Para ello se realiza una prueba de Levene, en la cual se valida si:

*Si el P-Valor  $\geq \alpha$  Se acepta  $H_0$  . Las varianzas son iguales.*

*Si el P-Valor  $< \alpha$  Se acepta  $H_1$  . Existe una diferencia significativa entre las varianzas.*

- **Calcular P-Valor de la Prueba:** T de Student para muestras independientes, para calcular este valor es necesario haber cumplido las dos condiciones anteriores y el criterio de aceptación es el siguiente.

*Si el P – Valor  $\leq \alpha$  Se acepta  $H_1$ .*

*Si el P – Valor  $> \alpha$  Se acepta  $H_0$ .*

#### 4.4.2. Resultados

Los resultados estadísticos obtenidos, se pueden visualizar en la [Figura 4.18] y [Figura 4.19].

TIPO			Estadístico	Error estándar		
TOTAL	CON	Media	147.4500	2.69549		
		95% de intervalo de confianza para la media	Límite inferior	141.8083		
			Límite superior	153.0917		
		Media recortada al 5%	148.0556			
		Mediana	149.0000			
		Varianza	145.313			
		Desviación estándar	12.05459			
		Mínimo	117.00			
		Máximo	167.00			
		Rango	50.00			
		Rango intercuartil	12.25			
		Asimetría	-1.026	.512		
		Curtosis	1.332	.992		
		SIN	SIN	Media	133.0500	4.97757
				95% de intervalo de confianza para la media	Límite inferior	122.6318
Límite superior	143.4682					
Media recortada al 5%	133.3333					
Mediana	136.5000					
Varianza	495.524					
Desviación estándar	22.26036					
Mínimo	96.00					
Máximo	165.00					
Rango	69.00					
Rango intercuartil	41.50					
Asimetría	-.231			.512		
Curtosis	-1.304			.992		

Figura 4.18: Resultados que muestran las medias y varianzas CON y SIN Acoli.

Pruebas de normalidad							
TIPO	Kolmogorov-Smirnov <sup>b</sup>			Shapiro-Wilk			
	Estadístico	gl	Sig.	Estadístico	gl	Sig.	
TOTAL	CON	.202	20	.032	.927	20	.136
	SIN	.128	20	.200*	.934	20	.184

Figura 4.19: Resultados prueba de NORMALIDAD.

Como se puede observar en la [Figura 4.18], el resultado de las medias obtenidas muestra que la media del grupo **CON** es mayor a la media del grupo **SIN**. Sin embargo, es necesario corroborar las condiciones de normalidad y la igualdad de varianzas [Figura 4.19]. El análisis de los resultados de la prueba NORMALIDAD como se pueden ver en la [Tabla 4.2], muestra que el grupo CON tiene un P-Valor igual a 0.032 menor que valor de  $\alpha$ , lo que nos dice que dicho grupo no proviene de una distribución normal, por lo que no es posible continuar con la prueba T de Student.

Tabla 4.2: Resultados prueba NORMALIDAD

NORMALIDAD		
P-Valor (CON) = 0.032	<	$\alpha = 0.05$
P-Valor (SIN) = 0.200	>	$\alpha = 0.05$
CONCLUSIÓN: Aceptamos $H_1 =$  Los datos para el grupo CON no provienen de una distribución normal.		

Dado que no se cumplió con el requisito de NORMALIDAD, se aplicó una prueba U de Mann Whitney, la cual es una versión no paramétrica de la prueba T de Student. Por lo que nuestras hipótesis alterna ( $H_1$ ) y nula ( $H_0$ ) cambian de la siguiente manera:

$$H_0 = \text{No hay diferencias entre las medianas de las variables.}$$

$$H_1 = \text{Hay diferencias entre las medianas de las variables.}$$

En donde si se cumple que:

$$\text{Si } P\text{-Valor} < \alpha \text{ Rechazamos } H_0$$

$$\text{Si } P\text{-Valor} \geq \alpha \text{ Aceptamos } H_1$$

En la [Figura 4.20] podemos visualizar los resultados estadísticos arrojados por SPSS, en donde el P-Valor corresponde a la "Significancia asintótica (bilateral)",.

### Estadísticos de prueba

	TOTAL
U de Mann-Whitney	125.500
W de Wilcoxon	335.500
Z	-2.016
Sig. asintótica (bilateral)	.044
Significación exacta [2*(sig. unilateral)]	.043 <sup>b</sup>

Figura 4.20: Resultados prueba de U de Mann Whitney

En la [Tabla 4.3] podemos ver el análisis de los resultados obtenidos, donde tenemos que el P-Valor es igual a 0.044, menor a nuestro valor  $\alpha$ , por lo tanto rechazamos  $H_0$ .

Tabla 4.3: Resultados prueba U de Mann Whitney

U de Mann Whitney		
P-Valor = 0.044	<	$\alpha = 0.05$
CONCLUSIÓN:		
Se rechaza $H_0$ = La mediana del total de flujo es similar en ambos grupos.		
En consecuencia se acepta que: “Existe una diferencia estadísticamente significativa entre el total de flujo y el tipo de actividad.		

Una vez obtenida la diferencia estadística entre ambos grupos, lo siguiente fue ver si los resultados obtenidos en los cuestionarios, con lo que respecta a la parte lúdica, son favorables para poder asegurar que el juego cumple con el objetivo de realizar el ejercicio de manera entretenida. Para tal propósito, solo se tomaron en cuenta los resultados de las siguientes dimensiones del flujo del grupo CON, se tomaron en cuenta estas dimensiones, dado que son las que se presentan comunmente en cualquier videojuego:

- Concentración y enfoque
- Perdida del autoconocimiento
- Alteración del sentido del tiempo
- Experiencia autotélica

Como se mencionó anteriormente, cada cuatro preguntas del cuestionario miden una dimensión del flujo, y dado que se utiliza una escala Likert del 1 al 5, el valor máximo posible que un usuario puede obtener, por cada dimensión, es de 20. De igual forma, como solo se va a tomar el grupo de usuarios CON, el máximo valor posible que el grupo puede obtener por cada dimensión es de 400. En la [Tabla 4.4] se muestran los resultados obtenidos del grupo CON en el cuestionario.

Tabla 4.4: Resultados del cuestionario del grupo CON

No Usuario	Edad	Sexo	Grupo	C.E.	P.A.	A.S.T.	E.A.
16	44	M	CON	15	15	7	17
20	39	F	CON	20	20	12	20
21	39	M	CON	14	18	4	19
4	35	M	CON	12	18	13	19
29	32	M	CON	18	19	15	19
5	31	M	CON	15	20	16	16
34	31	F	CON	16	16	10	20
10	29	M	CON	8	8	8	16
23	26	M	CON	12	15	16	19
36	26	M	CON	16	17	15	17
12	25	M	CON	16	19	10	13
14	25	M	CON	19	18	15	15
26	25	M	CON	12	15	17	19
32	25	M	CON	16	19	13	14
39	25	M	CON	16	14	16	15
9	24	F	CON	17	20	12	20
15	20	M	CON	17	19	16	16
24	19	M	CON	17	18	17	17
1	18	M	CON	11	10	4	17
2	17	M	CON	8	16	11	20
<b>TOTALES*</b>				295 = 73.75 %	334 = 83.5 %	247 = 61.75 %	348 = 87 %
* Porcentajes obtenidos considerando el resultado máximo posible							

C.E. = Concentración y enfoque  
P.A. = Pérdida del autoconocimiento  
A.S.T. = Alteración del sentido del tiempo  
E.A. = Experiencia Autotelica

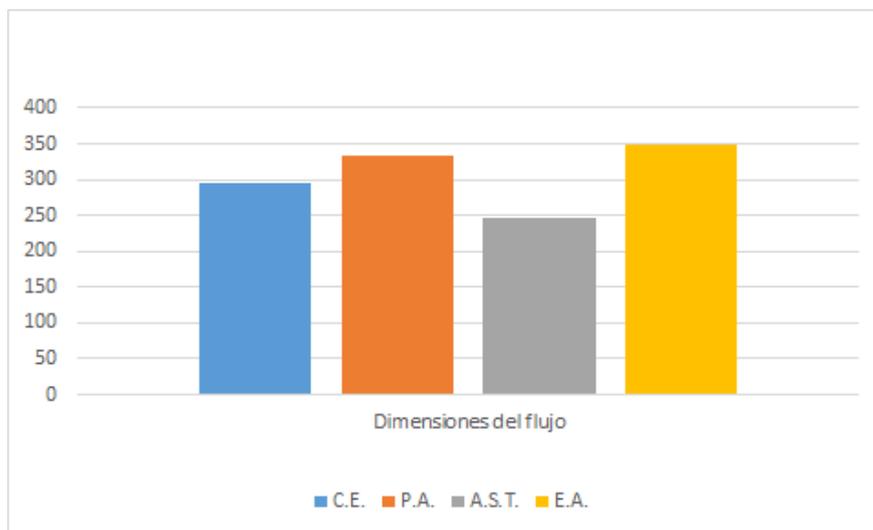


Figura 4.21: Gráfica de barras - Dimensiones del flujo

Por último, analizando los totales de la [Tabla 4.4], y como se puede observar en la [Figura 4.21] los resultados obtenidos para las cuatro dimensiones están por encima de la mitad del máximo posible. Por

---

lo que, es posible afirmar que los usuarios del grupo CON, tuvieron una experiencia lúdica al momento de realizar la actividad, por lo tanto Acoli cumple con el propósito de ser una alternativa en el modo de hacer terapias.

# Conclusiones y trabajo a futuro

## Conclusiones

En conclusión, el trabajo realizado cumple con las expectativas esperadas ya que ofrece una alternativa para el INR en el modo de hacer terapia, ya que sistemas como Acoli tienen el potencial para fomentar un mejor estado de flujo. Por lo visto en las terapias tradicionales que se realizan actualmente en el INR, el problema de deserción es muy común, por lo que de llegar a trascender este proyecto, será una herramienta, bien fundamentada, que será de gran ayuda para evitar dicha problemática.

## Trabajo a futuro

Una de las limitantes del presente proyecto fue el no poder hacer pruebas con pacientes reales, por lo que es difícil poder comparar el flujo cuando se usa y cuando no se usa Acoli, por tal razón, hacer pruebas con este tipo de pacientes podría ser de gran utilidad en un futuro.

Otra de las limitantes de Acoli desde un principio fue, que se pensó para un padecimiento en específico, en este caso el síndrome de hombro congelado, por lo que es necesario desarrollar o crear nuevos dispositivos para abarcar un mayor número de padecimientos. Así como, definir un mecanismo para estandarizar la creación de estos sistemas. De igual forma, sería de gran importancia evaluar Macoli en lo que respecta al rendimiento de los pacientes.

De tener una buena aceptación por parte de los pacientes del INR, se buscará ampliar la gama de juegos, buscando desarrollar nuevas dinámicas que cubran más padecimientos de extremidades superiores, sin dejar de aprovechar la conexión con Macoli y el dispositivo de control. De igual forma, se buscará ampliar el número de canciones de Acoli y mejorar algunos aspectos que puedan ser requeridos, como lo son interfaces de usuario, o la jugabilidad de Acoli, o aquellos que se reciban como sugerencias por parte de los pacientes del INR, con el fin de que Acoli se mantenga actualizado y no caiga en desuso por tales razones. Para tal propósito, es clara la necesidad de realizar pruebas con pacientes reales, con el fin de obtener una retroalimentación precisa.

---

# Anexo A

La primer clase, y una de las más importantes para el funcionamiento de Acoli es la clase **GameControllerJuego**, esta clase se encarga de mantener comunicación con las clases **ActivadorJuego** y **CreaNotasNivel1**, con el fin de inicializar valores de variables como el nivel, tiempo, dificultad y la canción seleccionada, los cuales se seleccionan en la pantalla de Ajustes.

```
public class GameControllerJuego : MonoBehaviour
{
    private CreaNotasNivel1 cnn1;
    private CreaNotasNivel2 cnn2;
    private CreaNotasNivel3 cnn3;
    public GameObject controladorCancion1;
    public GameObject controladorCancion2;
    public GameObject controladorCancion3;
    public int nScore;
    private int tiempo;
    private int dificultad;
    private int nivel;
    private int cancion;
    ...

    void Start ()
    {
        PlayerPrefs.SetInt ("Estado_Juego",0);
        texto_final_GUI = texto_final.GetComponent<
            GUIText> ();
        GUI.SetActive (false);
        suena = false;
        juegoTerminado = false;
        nScore = 0;
        pausa.SetActive (false);
        emergentePausa.SetActive (false);
        emergenteExito.SetActive (false);
        screen.SetActive (true);
        numeros.SetActive (true);
        enPausa = false;
        number = -3;
        tiempo = PlayerPrefs.GetInt ("Tiempo");
        nivel = PlayerPrefs.GetInt ("Nivel");
        dificultad = PlayerPrefs.GetInt ("Dificultad");
        cancion = PlayerPrefs.GetInt ("Cancion");

        couroutineStarted = false;

        tiempoSegundos = tiempo * 60;
    }
}
```

---

```

    tiempoText = tiempoObject.GetComponent<GUIText>()
        ;
    notasTerminadas = false;
    cnn1 = controladorCancion1.GetComponent<
        CreaNotasNivel1> ();
    cnn2 = controladorCancion2.GetComponent<
        CreaNotasNivel1C2> ();
    cnn3 = controladorCancion3.GetComponent<
        CreaNotasC4> ();

    asignaMaxPuntos ();
}

...

IEnumerator counting(int segundos)
{
    couroutineStarted = true;
    yield return new WaitForSeconds(segundos);
    number++;
    if(number > 0 && tiempoSegundos > 0)
        tiempoSegundos--;
    tiempoText.text = tiempoSegundos.ToString();
    couroutineStarted = false;
}

void asignaMaxPuntos()
{
    switch (cancion)
    {
        case 1://Cancion 1
            /*if(tiempo == 1 && dificultad ==
                1)
                maxPuntos = 64;*/
            if(tiempo == 1 && dificultad ==
                1)
                maxPuntos = 39;
            else if(tiempo == 1 && dificultad
                == 2)
                maxPuntos = 64;//
                maxPuntos = 82;
            else if(tiempo == 1 && dificultad
                == 3)
                maxPuntos = 64;//
                maxPuntos = 98;
            else if(tiempo == 2 && dificultad
                == 1)
                maxPuntos = 81;//
                maxPuntos = 130;
            else if(tiempo == 2 && dificultad
                == 2)
                maxPuntos = 98;//
                maxPuntos = 162;
            else if(tiempo == 2 && dificultad
                == 3)

```

---

```

        maxPuntos = 113;//
        maxPuntos = 226;
    else if(tiempo == 3 && dificultad
    == 1)
        maxPuntos = 113;//
        maxPuntos = 194;
    else if(tiempo == 3 && dificultad
    == 2)
        maxPuntos = 129;//
        maxPuntos = 226;
    else if(tiempo == 3 && dificultad
    == 3)
        maxPuntos = 162;//
        maxPuntos = 290;
    break;
    ...
}
PlayerPrefs.SetInt ("MaxPuntos",maxPuntos);
}
}

```

La siguiente clase de importancia es **Bluetooth conexion**, esta clase como su nombre lo dice, se encarga de mantener una conexión constante entre los sensores de proximidad del dispositivo de control y Acoli. En su método *asignarEscalon(int \_escalon)*, se recibe el escalón sobre el cual está posicionado el paciente, dicho valor se asigna en una variable global, haciendo uso de PlayerPrefs, la cual será de utilidad para la clase **ActivadorJuego**.

```

public class BluetoothConexion : MonoBehaviour
{
    public int nivel;
    void Awake ()
    {
        escalondelaCalibracion (nivel);
        DontDestroyOnLoad (transform.gameObject);
    }

    private int escalon =0;
    private int escalonCalibracion = 0;
    private int sesion = 0;

    public void asignarEscalon(int _escalon)
    {
        escalon = _escalon;
        PlayerPrefs.SetInt ("escalon",_escalon);
    }

    public void escalondelaCalibracion(int
    _escalonCalibracion)
    {
        escalonCalibracion = _escalonCalibracion;
        PlayerPrefs.SetInt ("MaxNivel",
        _escalonCalibracion);
    }
}

```

---

```

public void asignarSesion(int _sesion)
{
    sesion = _sesion;
    PlayerPrefs.SetInt ("Sesion",_sesion);
}
}

```

Otra clase de suma importancia es la clase **MenuJuegoReturn**, ésta se encarga de obtener la información obtenida del juego que se acaba de jugar, es decir, la sesión, el tiempo, la dificultad, el nivel, la canción, los puntos obtenidos así como el máximo de puntos (esto nos permite calcular los errores cometidos), además del estado del juego, esto se refiere a si se completó el tiempo de juego o no. Una vez obtenida la información del juego en cuestión, se envía a Macoli para que posteriormente sea almacenada en la base de datos, pero sin salir de Acoli, lo que permite que se pueda jugar más de una vez.

```

public class MenuJuegoReturn : MonoBehaviour
{
    public Texture normal;
    public Texture hover;

    private int id_sesion;
    private int duracion;
    private int dificultad;
    private int puntuacion;
    private int escalon;
    private int cancion;
    private int estado;
    private int maxPuntuacion;
    private int errores;

    public delegate void AgregarJuego(int sesion, int
        duracion, int dificultad, int escalon, int estado, int
        cancion, int puntuacion, int maxpuntuacion, int
        errores);
    public static AgregarJuego agregarJuego;

    void mandaInformacionInterfaz(int sesion, int duracion,
        int dificultad, int escalon, int estado, int cancion,
        int puntuacion, int maxpuntuacion, int errores)
    {
        UnityEngine.WSA.Application.InvokeOnUIThread(()
            =>
            {
                agregarJuego(sesion, duracion, dificultad
                    , escalon, estado, cancion, puntuacion
                    , maxpuntuacion, errores);
            }, false);
    }

    void OnMouseDown()
    {
        id_sesion = PlayerPrefs.GetInt ("Sesion");
        duracion = PlayerPrefs.GetInt ("Tiempo");
        dificultad = PlayerPrefs.GetInt ("Dificultad");
        escalon = PlayerPrefs.GetInt ("Nivel");
    }
}

```

---

```

        estado = PlayerPrefs.GetInt ("Estado_Juego");
        cancion = PlayerPrefs.GetInt ("Cancion");
        puntuacion = PlayerPrefs.GetInt ("Puntos");
        maxPuntuacion = PlayerPrefs.GetInt ("MaxPuntos");
        errores = maxPuntuacion - puntuacion;

        mandaInformacionInterfaz (id_sesion,duracion,
            dificultad,escalon,estado, cancion,puntuacion,
            maxPuntuacion,errores);
        Application.LoadLevel ("Menu");
    }

    void OnMouseOver()
    {
        GetComponent<GUITexture>().texture = hover;
    }

    void OnMouseExit()
    {
        GetComponent<GUITexture>().texture = normal;
    }
    void OnMouseEnter()
    {
        this.GetComponent<AudioSource>().Play ();
    }
}

```

Otra clase que es importante mencionar es la clase **Move**, ya que ésta se replica en cada una de las notas del juego, y es la encargada de hacer que las notas avancen a lo largo de pantalla.

```

public class Move : MonoBehaviour
{
    public GameObject hazard;
    public float velocidad;
    public Vector3 posicionInicial;
    public bool exito;

    void Start ()
    {
        posicionInicial = hazard.transform.position;
        exito = false;
    }

    void Update ()
    {
        if (!exito)
        {
            if (this.transform.position.x >= 14.2f)//
                Si no colisiono y sale del rango de la
                camara
            {
                StartCoroutine (reinicia (1));
            }
            else

```

---

```

        {
            StartCoroutine (mueve (1));
        }
    }

IEnumerator reinicia(int segundos)
{
    yield return new WaitForSeconds (segundos);
    gameObject.transform.position = posicionInicial;
}

IEnumerator mueve(int segundos)
{
    yield return new WaitForSeconds (segundos);
    this.transform.Translate (Vector3.left *
        velocidad);
}

void OnTriggerEnter(Collider other)
{
    if (other.gameObject.tag == Tags.activador)
    {
        this.GetComponent<AudioSource>().Play();
        exito= true;
    }
}
}

```

A diferencia de la clase **MenuJuegoReturn** la clase **RegresarAplicacionPrincipal**, como su nombre lo dice, se encarga de salir de Acoli y regresar a Macoli, es por medio del método *navigateBack()*, que a su vez se encuentra dentro del método *irAplicacionPrincipal()*, que esto se logra.

```

public class RegresarAplicacionPrincipal : MonoBehaviour
{

    public delegate void NavigateBack();
    public static NavigateBack navigateBack;

    void irAplicacionPrincipal()
    {
        UnityEngine.WSA.Application.InvokeOnUIThread(()
            =>
                {
                    navigateBack();
                }, false);
    }

    public Texture normal;
    public Texture hover;
    public GameObject emergente;
    public Light luz;
    public GameObject screen;
    public bool listoParaJugar;
}

```

---

```

void OnMouseDown ()
{
    GetComponent<GUITexture>().texture = normal;
    luz.intensity = 0.8f;
    listoParaJugar = true;
    screen.SetActive (true);
    emergente.SetActive(false);
    irAplicacionPrincipal();
}

void OnMouseOver ()
{
    GetComponent<GUITexture>().texture = hover;
}

void OnMouseExit ()
{
    GetComponent<GUITexture>().texture = normal;
}
void OnMouseEnter ()
{
    this.GetComponent<AudioSource>().Play ();
}
}

```

Una de las clases más importantes del juego, es la clase **ActivadorJuego**, ya que dicha clase está asignada a cada uno de los activadores del juego, y se encarga de mover los activadores según los valores que recibe el sensor de proximidad dependiendo del escalón en el que se esté posicionado. De igual forma, esta clase en su método *OnTriggerEnter*, se encarga de validar cuando una nota colisiona con el activador, y de asignar los puntos correspondientes dependiendo del tipo de nota que se halla atrapado.

```

public class ActivadorJuego : MonoBehaviour
{
    public GameObject gameController;
    private int nivel;
    private Vector3 posinicial;
    private Vector3 posfinal;
    public float velocidad;
    public float tiempoActivado;
    public int idActivador;
    public bool teclado; //0-version bluetooth, 1-version
        teclado
    ...

    void Start ()
    {
        nivel = PlayerPrefs.GetInt ("Nivel");
        posinicial = this.transform.position;
        posfinal = new Vector3 (posinicial.x,posinicial.y
            ,posinicial.z-0.4f);
        contador = 0;
        val = -1;
        enProceso = false;
        colisionLarga = false;
    }
}

```

---

```

}

void Update ()
{
    //Esto se hace para que el activador se quede en
    //su lugar y no huya de la pantalla
    if(this.transform.position.z > 0.1f)
    {
        this.transform.position = posinicial;
    }
    if (!enProceso)
    {
        if(idActivador == 1)
        {
            if(!teclado)
            {
                float recuperacion = 0;

                int vali = PlayerPrefs.
                    GetInt("escalon");
                Debug.Log (vali);
                if(contador > 45)//Si
                    lleva mucho tiempo
                    presionado lo bajamos
                {
                    vali = 0;
                    recuperacion =
                        0.5f;
                    StartCoroutine(
                        apaga(
                            recuperacion))
                    ;
                }
                else if(vali == 1)
                {
                    contador ++;
                    enProceso = true;
                    StartCoroutine(
                        enciende(
                            recuperacion))
                    ;
                }
                else
                {
                    StartCoroutine(
                        apaga(
                            recuperacion))
                    ;
                }
            }
            ...
        }
    }
}

```

---

```

void OnTriggerEnter(Collider other)
{
    if (other.gameObject.tag == Tags.nota)
    {
        cortas.Play(true);
        gameController.GetComponent<
            GameControllerJuego>().nScore ++;
    }
    else if (other.gameObject.tag == Tags.notaDoble)
    {
        collisionRealD += Time.deltaTime;
        largas.Play(true);
        largas1.Play(true);
        gameController.GetComponent<
            GameControllerJuego>().nScore ++;
        gameController.GetComponent<
            GameControllerJuego>().nScore ++;
        collisionRealD = 0;
    }
    else if (other.gameObject.tag == Tags.notaTriple)
    {
        collisionRealT += Time.deltaTime;
        largas.Play(true);
        largas1.Play(true);
        gameController.GetComponent<
            GameControllerJuego>().nScore ++;
        gameController.GetComponent<
            GameControllerJuego>().nScore ++;
        gameController.GetComponent<
            GameControllerJuego>().nScore ++;
    }
    collisionRealD = 0;
    collisionRealT = 0;
}
}

```

Otra clase importante, es la clase **Iniciar**, ya que se encarga de pasar de la pantalla de Selección de Canción a la pantalla de Juego, dependiendo de la información seleccionada. Así como esta clase, existen otras clases que funcionan de manera similar, permitiendo la navegación entre las pantallas de Acoli. Para esto se usan métodos nativos de Unity, en este caso, es el método *Application.LoadLevelAsync("NombreNivel")* el que realiza tal acción.

```

public class Iniciar : MonoBehaviour
{
    public GUITexture c1;
    public GUITexture c2;
    public GUITexture c3;
    private SelectSong sc1;
    private SelectSong sc2;
    private SelectSong sc3;
    public GameObject emergente;
    public Light luz;
    public GameObject canciones;
    ...
}

```

---

```

void Update()
{
    Debug.Log (contador);
    if(iniciaConteo)
        StartCoroutine(iniciaContador(1));
    if(contador > 20)
        StartCoroutine(spinner(10));
}

void Start()
{
    spinner_o.SetActive (false);
    couroutineStarted = true;
    sc1 = c1.GetComponent<SelectSong> ();
    sc2 = c2.GetComponent<SelectSong> ();
    sc3 = c3.GetComponent<SelectSong> ();
}

void OnMouseDown()
{
    //Debug.Log (Time.deltaTime);
    GetComponent<GUITexture>().texture = normal;
    if (isSelect ())
    {
        iniciaConteo = true;
        if(PlayerPrefs.GetInt("Nivel") == 1)
        {
            Application.LoadLevelAsync ("
                Nivel1");
        }else if(PlayerPrefs.GetInt("Nivel") ==
            2)
        {
            Application.LoadLevelAsync ("
                Nivel2");
        }else if(PlayerPrefs.GetInt("Nivel") ==
            3)
        {
            Application.LoadLevelAsync ("
                Nivel3");
        }else if(PlayerPrefs.GetInt("Nivel") ==
            4)
            Application.LoadLevelAsync ("
                Nivel4");
        else if(PlayerPrefs.GetInt("Nivel") == 5)
            Application.LoadLevelAsync ("
                Nivel5");
        else if(PlayerPrefs.GetInt("Nivel") == 6)
            Application.LoadLevelAsync ("
                Nivel6");
        else if(PlayerPrefs.GetInt("Nivel") == 7)
            Application.LoadLevelAsync ("
                Nivel7");
    }
}

```

---

```

    }
    else
    {
        canciones.SetActive(false);
        //luz.intensity = 0.2f;
        emergente.SetActive(true);
    }
}
...
}

```

Por otro lado, tenemos la clase **CreaNotasNivel1**, esta clase es muy parecida para cada canción del juego (en este caso tres), con la diferencia de las notas que tiene como atributos, ya que éstas son diferentes para cada canción. Es de gran importancia, ya que su labor es la de crear, antes de iniciar el juego, las notas necesarias, según la dificultad, el nivel y el tiempo seleccionados previamente, e ir las mostrando según su tiempo de aparición. Es en el método *reproduce(float segundos, GameObject[] notas)*, que se toma el arreglo de notas, previamente construido, y por medio de un iterador global se van activando las notas contenidas en el arreglo.

```

public class CreaNotasNivel1 : MonoBehaviour
{
    private int tiempo;
    private int cancion;
    private int dificultad;
    private int nivel;

    public float velocidadAparicion;
    public GameObject gmN1;
    public GameObject pausa;
    public bool enPausa;

    //Cancion1
    public GameObject n1;
    public GameObject n2;
    public GameObject n3;
    public GameObject n4;
    ...

    void Awake ()
    {

        tiempo = PlayerPrefs.GetInt ("Tiempo");
        cancion = PlayerPrefs.GetInt ("Cancion");
        nivel = PlayerPrefs.GetInt ("Nivel");
        dificultad = PlayerPrefs.GetInt ("Dificultad");

        if (dificultad == 1)
            alentador = 3.3f;
        else if (dificultad == 2)
            alentador = 2.5f;
        else if(dificultad == 3)
            alentador = 2.0f;

        if(cancion == 1)
        {
            tam1c1 = 47;

```

---

```

tam2c1 = 98;
tam3c1 = 148;
enPausa = false;
finalCancion = false;
k = 0;
tiemposc1 = new float[11];
listo = gmN1.GetComponent<
    GameControllerJuego>().number;

ciclo = true;

tiemposc1 [0] = 0.5f; tiemposc1 [1] = 0.6f
    ; tiemposc1 [2] = 0.7f; tiemposc1 [3] =
    0.8f; tiemposc1 [4] = 0.83f;
tiemposc1 [5] = 0.85f; tiemposc1 [6] =
    0.87f; tiemposc1 [7] = 0.9f; tiemposc1
    [8] = 1.1f; tiemposc1 [9] = 1.2f; //
    8-1.6 9-1.7
tiemposc1 [10] = 1.4f; // 10 - 1.8

if(tiempo == 1)
{
    if(dificultad == 1)
    {
        tam1c1 = 27;
        aTocar = new GameObject[
            tam1c1];
        aTocar [0] = n1; aTocar
            [1] = n2; aTocar [2] =
            n3; aTocar [3] = n4;
            aTocar [4] = n5;
        aTocar [5] = n6; aTocar
            [6] = n7; aTocar [7] =
            n8; aTocar [8] = n9;
            aTocar [9] = n10;
        aTocar [10] = n11; aTocar
            [11] = n12; aTocar
            [12] = n13; aTocar
            [13] = n14; aTocar
            [14] = n15;
        aTocar [15] = n16; aTocar
            [16] = n17; aTocar
            [17] = n18; aTocar
            [18] = n19; aTocar
            [19] = n20;
        aTocar [20] = n21; aTocar
            [21] = n22; aTocar
            [22] = n23; aTocar
            [23] = n24; aTocar
            [24] = n25;
        aTocar [25] = n26; aTocar
            [26] = n27;
    }
    ...
}

```

---

```
        ...
    }

    IEnumerator reproduce(float segundos, GameObject[] notas)
    {
        ciclo = false;
        notas [k].SetActive (true);
        k++;
        yield return new WaitForSeconds (segundos);

        ciclo = true;
    }
    ...
}
```



# Anexo B

## Antecedentes técnicos

A continuación se presenta una breve descripción de las herramientas tecnológicas utilizadas para el desarrollo de Acoli, que fueron de gran ayuda para la realización del mismo. Cabe mencionar, que dichas herramientas cuentan con licencia de desarrollo libre, al menos para fines académicos, como el presente proyecto.

### MySQL

Es un sistema de gestión de bases de datos relaciones. Para el presente proyecto es de gran importancia, que toda la información que se genera a lo largo del juego como son: puntajes, el tiempo de juego, la dificultad y el nivel del juego, entre otros, se almacene de forma adecuada para su reutilización en las sesiones de terapia posteriores, así como para visualizar los avances de los pacientes.

### 3DS MAX

Es un software de modelado, renderizado y animación 3D, que forma parte de la familia de programas de la empresa Autodesk. Si bien, la dinámica de juego de Acoli se maneja en un ambiente 2D, casi todos los componentes del juego están modelados en 3D, con el fin de brindar una mejor experiencia visual al usuario. Para la creación de todos estos elementos 3D, el uso de 3DS MAX fue de gran ayuda.

### Arduino

Es una plataforma electrónica basada en hardware y software fácil de usar. Por el lado del hardware, Arduino cuenta con múltiples placas de circuito impreso, que se usan dependiendo de la naturaleza del proyecto en el que se ocupen. En el caso de Acoli, bastó con una placa Arduino Mega [Figura 4.16] para llevar a cabo toda la lógica del control de Acoli.

Por otra parte, el software de Arduino cuenta con su propio lenguaje de programación y Entorno de Desarrollo Integrado<sup>8</sup>. Ambos elementos se encargan de fijar las tareas en la placa Arduino para manipular la entrada y salida de valores, los cuales manda y recibe de Unity3D por medio de un dispositivo de bluetooth.

### PROTEUS - ARES

PROTEUS como tal, es un IDE, específico para el desarrollo de proyectos de electrónica. Mientras que ARES, es una de las herramientas que brinda PROTEUS, y que permite la elaboración de placas de

---

<sup>8</sup>IDE, por sus siglas en inglés

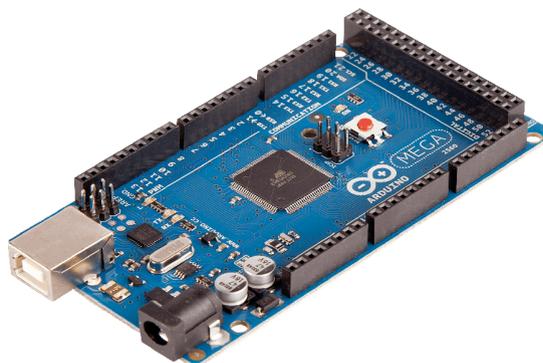


Figura 4.22: Arduino Mega 2560

circuito impreso, lo cual es de gran ayuda, ya que el trabajo duro de la realización de placas electrónicas y del diseño de pistas recae sobre el PC.

El uso de ARES, al igual que Arduino, sirvió de apoyo en la creación de la lógica del control de Acoli.

### Unity3D

Es un motor de juegos multiplataforma creado por la empresa Unity Technologies, que permite la creación de juegos para Windows, OS X, Linux, Xbox 360, PlayStation 3, PlayStation Vita, Wii, Wii U, iPad, iPhone, Android y Windows Phone.

Unity cuenta con múltiples herramientas, las cuales lo hacen un motor bastante completo para el desarrollo de videojuegos. Unity cuenta con diferentes motores gráficos dependiendo del Sistema Operativo<sup>9</sup> que se use, para Windows utiliza Direct3D, mientras que para Linux y Mac, utiliza OpenGL.

El scripting se lleva a cabo a través de MonoDevelop<sup>10</sup>. El script se basa en la implementación de código abierto de .NET Framework. Los programadores pueden utilizar UnityScript (lenguaje personalizado inspirado en la sintaxis ECMAScript), C# o Boo (con un sintaxis inspirada en Python). A partir de la versión 3.0 añade una versión personalizada de MonoDevelop para la depuración de scripts.

Otra de las tecnologías con las que cuenta Unity es Mecanim, la cual se encarga de la parte de animación de Unity. Dicha tecnología está diseñada para llevar el movimiento fluido y natural de los personajes con una interfaz eficiente. Mecanim además, incluye herramientas para la creación de máquinas de estados, árboles de mezcla (blend trees) y re-focalización automático de animaciones, desde el editor de Unity.

Además, está disponible el Unity Asset Store, tras el lanzamiento de la herramienta. En donde, muchos de estos archivos de animación de captura de movimiento son proporcionados sin costo por Unity Technologies. También existen otros proveedores del Asset Store que ofrecen animaciones, modelos 3D, paquetes de audio o juegos completos, entre otras cosas, ya sea de forma gratuita o de pago.

---

<sup>9</sup>SO, por sus siglas en español

<sup>10</sup>MonoDevelop es un IDE libre y gratuito, diseñado principalmente para C# y otros lenguajes .NET



---

		Totalmente en desacuerdo				Totalmente de acuerdo
19.	Sentí que era lo suficientemente competente como para satisfacer las altas exigencias de la situación	1	2	3	4	5
20.	Jugué de forma automáticamente	1	2	3	4	5
21.	Sabía lo que quería lograr	1	2	3	4	5
22.	Tenía una buena idea mientras estaba jugando sobre lo bien que lo estaba haciendo	1	2	3	4	5
23.	Tenía una concentración total	1	2	3	4	5
24.	Tenía una sensación de control total	1	2	3	4	5
25.	No estaba preocupado por cómo estaba presentandome	1	2	3	4	5
26.	Sentí que el tiempo se detenía mientras estaba jugando	1	2	3	4	5
27.	La experiencia me hizo sentirme muy bien	1	2	3	4	5
28.	El desafío y mis habilidades estaban en un nivel igualmente alto	1	2	3	4	5
29.	Hice cosas espontáneas y automáticamente sin tener que pensar	1	2	3	4	5
30.	Mis objetivos fueron claramente definidos	1	2	3	4	5
31.	Me di cuenta por la forma en que estaba jugando lo bien que lo estaba haciendo	1	2	3	4	5
32.	Estaba completamente enfocado en la tarea que tenía entre manos	1	2	3	4	5
33.	Me sentí en control total de mi cuerpo	1	2	3	4	5
34.	No estaba preocupado por lo que otros pudieran haber estado pensando en mí	1	2	3	4	5
35.	A veces, casi parecía que las cosas estaban pasando en cámara lenta	1	2	3	4	5
36.	Encontré la experiencia extremadamente gratificante	1	2	3	4	5

# Referencias

## Capítulo I

BURDEA, Grigore C. Burdea, Senior Member IEEE; Daniel Cioi, Member IEEE; Joseph Martin, Devin Fensterheim, and Maeve Holenski (2010). "The Rutgers Arm II Rehabilitation System — A Feasibility Study". IEEE Transactions on neural systems and rehabilitation engineering, VOL. 18, NO. 5, OCTOBER 2010 [online]. Disponible en: [http://www.ti.rutgers.edu/publications/papers/2010\\_VirtReh\\_RutgersArmII\\_Burdea.pdf](http://www.ti.rutgers.edu/publications/papers/2010_VirtReh_RutgersArmII_Burdea.pdf)

ARMEO, Hocoma.com (2009). "ARMEO Therapy concept". [online] Disponible en: [https://www.hocoma.com/fileadmin/user/Dokumente/Armeo/bro\\_Armeo\\_Therapy\\_Concept\\_140226\\_en.pdf](https://www.hocoma.com/fileadmin/user/Dokumente/Armeo/bro_Armeo_Therapy_Concept_140226_en.pdf)

HOMBRO CONGELADO, Tutraumatologo.com (2014). Hombro congelado. [online] Disponible en: <http://www.tutraumatologo.com/hombrocongelado.html>.

HEYDE, Heyde, R. (2011). "Occupational Therapy Interventions for Shoulder Conditions: A Systematic Review". American Journal of Occupational Therapy, 16-23.

SVEISTRUP, Sveistrup, H., Mccomas, J., Thornton, M., Marshall, S., Finestone, H., McCormick, A., Mayhew, A. (2003). "Experimental Studies of Virtual Reality-Delivered Compared to Conventional Exercise Programs for Rehabilitation". CyberPsychology & Behavior, 245-249.

IBARRA, Dr. Ibarra, Luis G. (2011). "Las enfermedades y traumatismos del sistema músculo esquelético. Un análisis del Instituto Nacional de Rehabilitación de México, como base para su clasificación y prevención". Centro colaborador de la OPS/OMS para la investigación y rehabilitación médica. <http://www.inr.gob.mx/Descargas/ops-oms/lasEnfermedadesTraumatismosSistemaMusculoEsqueletico.pdf>, 54-65.

GREEN, greenfacts.org (2016). <http://www.greenfacts.org/es/glosario/tuv/tejidos-blandos.htm>

INEGI, Instituto Nacional de Estadística y Geografía (2010). "Las personas con discapacidad en México : una visión al 2010 / Instituto Nacional de Estadística y Geografía".

INEGI, Instituto Nacional de Estadística y Geografía (2004). "Las personas con discapacidad en México una visión global, México" <http://www.libreacceso.org/nueva/wp-content/uploads/2014/05/inegidiscapacidad2004.pdf>, 25-34.

WALKER, Walker-Bone K., Palmer K., et al. (2004). Prevalence and impact of musculoskeletal disorders of the upper limb in the general population. Arthritis Care & Research vol. 51, no. 4, Aug 15, 642-651.

ÖSTÖR, Östör AJK, Richards CA, et al. (2005). Diagnosis and relation to general health of shoulder disorders presenting to primary care. Rheumatology; 44:800-805.

IMAI, amai.org (2016). <http://nse.amai.org/nseamai2/>

---

## Capítulo II

MARCANO, Beatriz (2008). Juegos serios y entrenamiento en la sociedad digital. En SÁNCHEZ i PERIS, Francesc J. (Coord.) Videojuegos: una herramienta educativa del “homo digitalis”. Revista Electrónica Teoría de la Educación: Educación y Cultura en la Sociedad de la Información. Vol. 9, nº 3. Universidad de Salamanca. [http://www.usal.es/~teoriaeducacion/rev\\_numero\\_09\\_03/n9\\_03\\_marcano.pdf](http://www.usal.es/~teoriaeducacion/rev_numero_09_03/n9_03_marcano.pdf)

ZYDA, Micheal, Zyda (2005). "From visual simulation to virtual reality to games". Published by the IEEE Computer Society. 25-32

SÁNCHEZ, Sánchez Gómez, María (2007). "Buenas prácticas en la creación de serious games (Objetos de aprendizaje reutilizables)". Universidad de Málaga. Facultad de Ciencias de la Comunicación. Campus de Teatinos. <http://ceur-ws.org/Vol-318/Sanchez.pdf>

FLORIA, Alejandro Floría Cortés (2000). Recopilación de Métodos de Usabilidad". Área de Ingeniería de Proyectos, Departamento de Ingeniería de Diseño y Fabricación, Centro Politécnico Superior, Universidad de Zaragoza. <https://www.sidar.org/recur/desdi/traduc/es/visitable/quees/usab.htm>.

MIT, appinventor.mit.edu (2012). "Needfinding". Teaching App Creation With MIT App Inventor. <http://appinventor.mit.edu/explore/sites/all/files/teachingappcreation/unit2/DesignUnit2.pdf>

SDLC, tutorialspoint.com (2016). "SDLC - Software Prototype Model". [http://www.tutorialspoint.com/sdlc/sdlc\\_software\\_prototyping.htm](http://www.tutorialspoint.com/sdlc/sdlc_software_prototyping.htm)

CHIPIA, Chipia Lobo, Joan Fernando (2011). "Juegos serios: alternativa innovadora". II Congreso en línea en conocimiento libre y educación CLED 2011. <http://erevistas.saber.ula.ve/index.php/cled/article/view/4862/4680>

BURDEA, Grigore C. Burdea, Senior Member IEEE; Daniel Cioi, Member IEEE; Joseph Martin, Devin Fensterheim, and Maeve Holenski (2010). "The Rutgers Arm II Rehabilitation System — A Feasibility Study". IEEE Transactions on neural systems and rehabilitation engineering, VOL. 18, NO. 5, OCTOBER 2010 [online]. Disponible en: [http://www.ti.rutgers.edu/publications/papers/2010\\_VirtReh\\_RutgersArmII\\_Burdea.pdf](http://www.ti.rutgers.edu/publications/papers/2010_VirtReh_RutgersArmII_Burdea.pdf)

## Capítulo III

NAKAMURA, Nakamura, J., & Csikszentmihalyi. The concept of flow. Handbook of positive psychology, 89-105. <http://eweaver.myweb.usf.edu/2002-Flow.pdf>

WALTER, Walter, John Carl III. Flow - A Theory of Optimal Experience: History and Critical Evaluation. <http://kreativproces.dk/wp-content/uploads/2011/02/flowpaper.pdf>

CSIKSZENTMIHÁLYI, Mihály Csikszentmihályi (1975). "Beyond boredom and anxiety". Jossey-Bass Publishers. 35-54.

DRISKELL, Driskell, James E.; Willis, R. P.; Copper, C. (1992). "Effect of Overlearning on Retention". Journal of Applied Psychology 77 (5): 615–622.

RATHUNDE, Rathunde, K. & Csikszentmihályi, M. (2005), "Middle school students' motivation and quality of experience: A comparison of Montessori and traditional school environments", American Journal of Education 111 (3): 341–371.

COX, Cox, Richard H. (2007), "Psicología del deporte. Conceptos y sus aplicaciones", 182-186.

MANZANO, Manzano, Örjan; Theorell, Töres; Harmat, László; Ullén, Fredrik (2010). "The psychophysiology of flow during piano playing". Emotion 10 (3): 301–311.

---

GAITÁN, Gaitán, Virginia (2013). "Gamificación: el aprendizaje divertido". <http://www.educativa.com/blog-articulos/gamificacion-el-aprendizaje-divertido/>

GALLEGO, Gallego, Francisco J.; Molina, Rafael; Llorens, Faraón (2014). "Gamificar una propuesta docente, Diseñando experiencias positivas de aprendizaje". XX Jornadas sobre la Enseñanza Universitaria de la Informática.

ZALDÍVAR, Solo de Zaldívar, Isabel Fernández; "Juego serio: gamificación y aprendizaje". Especial Gamificación. <https://www.centrocp.com/juego-serio-gamificacion-aprendizaje/>

JACKSON, Susan A.; Herbert W. Marsh (1996). "Development and Validation of a Scale to Measure Optimal Experience: The Flow State Scale". Journal of sport & exercise psychology. 1996. 18: 17-35.

QUESTION PRO (2018), "¿Qué es la escala de Likert y cómo utilizarla?" <https://www.questionpro.com/blog/es/que-es-la-escala-de-likert-y-como-utilizarla/>

## Capítulo IV

MONTEFERRER, Monteferrer Agut Raúl (2000-2001). "Especificación de Requisitos Software según el estándar de IEEE 830". E78. INGENIERÍA DEL SOFTWARE, 5º CURSO DE INGENIERÍA INFORMÁTICA. Departament d'Informàtica Universitat Jaume I.

JACOBSON, Ivar; BOOCH, Grady; RUMBAUGH, James. El Proceso Unificado de Desarrollo de Software. Pearson Addison-Wesley. Año 2000, <https://drive.google.com/file/d/0BwWwTdM7msEKMXVS0Hp0c05wNTg/view>



# Bibliografía

- [1] Tarang K. Jain and Neena K. Sharma. *The effectiveness of physiotherapeutic interventions in treatment of frozen shoulder/adhesive capsulitis: A systematic review*. Department of Physical Therapy and Rehabilitation Science, University of Kansas Medical Center, Kansas city, USA. 2014
- [2] William E. Morgan, Sarah Potthoff. *Managing the Frozen Shoulder*. Walter Reed National Military Medical Center.
- [3] Ivar Jacobson, Grady Booch y James Rumbaugh. *El proceso unificado de desarrollo de software*. Addison-Wesley, 1-102, 1999. <https://drive.google.com/file/d/0BwWwTdM7msEKMXVSOHp0c05wNTg/view>
- [4] Project Management Institute. *Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK)*. Quinta Edición, 38-46, 2013. [https://www.gob.mx/cms/uploads/attachment/file/79535/PMBOK\\_5ta\\_Edicion\\_Espanol\\_\\_1\\_.pdf](https://www.gob.mx/cms/uploads/attachment/file/79535/PMBOK_5ta_Edicion_Espanol__1_.pdf)
- [5] Mihaly Csikszentmihalyi. *Beyond boredom and anxiety*. Jossey-Bass Publishers, 35-54, 1975. [http://psy2.ucsd.edu/~nchristenfeld/Happiness\\_Readings\\_files/Class%207%20\protect\discretionary{\char\hyphenchar\font}{\}%20Csikszentmihalyi%201975.pdf](http://psy2.ucsd.edu/~nchristenfeld/Happiness_Readings_files/Class%207%20\protect\discretionary{\char\hyphenchar\font}{\}%20Csikszentmihalyi%201975.pdf)