Mecabeat

Jessica Paola Murillo Lopez

Universidad del Valle
Sede Meléndez
Facultad de Artes Integradas
Departamento de Diseño
Diseño Gráfico
Santiago de cali
Junio 2024

Agradecimientos

Con gran emoción y gratitud, quiero dedicar estas palabras a mi familia, quienes han sido mi roca y mi inspiración durante estos cinco años de formación. Gracias por su amor incondicional, su paciencia y su constante apoyo. Cada logro es tan suyo como mío, porque han estado a mi lado en cada paso del camino.

También quiero expresar mi más profundo agradecimiento a mi director de grado, José Cuartas. Su orientación, sabiduría y apoyo han sido fundamentales para mi desarrollo académico y personal.

Abstract

This project focuses on the development of a rhythm game called Mecabeat, designed to test users' typing skills. The software GDevelop, a tool for 2D video game development, was used for its creation. Subsequently, a Cyberpunk aesthetic was implemented for the game. Finally, the results obtained from user testing sessions will be analyzed.

Resumen

Este proyecto se centra en el desarrollo de un juego de ritmo llamado Mecabeat, diseñado para poner a prueba las habilidades de mecanografía de los usuarios. Para su creación, se utilizó el software GDevelop, una herramienta para el desarrollo de videojuegos en 2D. Posteriormente, se trabajó en una estética de Cyberpunk para el juego. Por último, se analizarán los resultados obtenidos a partir de sesiones de prueba con usuarios.

Palabras clave:

Videojuegos, Mecanografía, Ritmo, Programación, Desarrollo, Experimentación.

Tabla de contenido

Introducción 5

Justificación 8

Planteamiento del problema 9

Pregunta de la investigación 9

Objetivo General 10

Objetivos específicos 10

Conceptos y bases teóricas 11

Mecanografía 11

Videojuegos de ritmo 11

GDevelop 11

Gamificación 12

Heurística de usabilidad 13

Metodología 16

Estado del arte 18

Planteamiento inicial 22

Método 23

Experimento 1 - prototipo de la mecánica con la fila guía 24

Movimiento de las teclas 25

Actualización de textos en pantalla: 25

Iluminación de los dedos de las manos: 25

Colisión con la ranura y sistema de precisión: 26

..... 27

Eliminación de teclas cuando se falla al pulsar en la ranura 28

Generación de las teclas/notas 28

Experimento 2: palabras aleatorias y las letras de una palabra 29

Palabra y letra aleatoria: 29

Las letras de una palabra aleatoria 30

Experimento 3 - Prototipo con todas las letras 30

Vidas: 31

Generación de teclas: 31

Navegación 32

Escena pausa: 32

Escena game over: 33

Escena resultados 33

Escena inicio 34

Estética 35

Pruebas del prototipo con usuarios 37

Correcciones 39

Conclusiones 41

Anexos 42

Referencias 52

Índice de Figuras

Figura 1 Captura de gameplay 18
Figura 2 Captura de gameplay 18
Figura 3 Captura de gameplay 18
Figura 4 Captura de gameplay 19
Figura 5 Pantalla Inicio NotITG 19
Figura 6 Captura de Lección..... 20
Figura 7 Captura de Lección..... 20
Figura 8 Captura de gameplay 21
Figura 9 Captura de gameplay 21
Figura 10 Mapa de navegación..... 23
Figura 11 Bocetación de las pantallas de Mecabeat..... 23
Figura 12 Captura del experimento 1 24
Figura 13 Captura de Evento GDevelop 24
Figura 14 Captura de Evento GDevelop 25
Figura 15 Captura de Evento GDevelop 25
Figura 16 Captura del experimento 1 26
Figura 17 Captura de Evento GDevelop 26
Figura 18 Captura sistema de precisión..... 26
Figura 19 Captura de Evento GDevelop 27
Figura 20 Captura máscara de colisión..... 27
Figura 21 Captura de Evento GDevelop 28
Figura 22 Captura de Evento GDevelop 28
Figura 23 Captura de Evento GDevelop 28
Figura 24 Captura de Evento GDevelop 29
Figura 25 Captura de Evento GDevelop 30
Figura 26 Captura de Evento GDevelop 30

Figura 27 Captura de Evento GDevelop 31
Figura 28 Captura de Evento GDevelop 31
Figura 29 Captura de Evento GDevelop 32
Figura 30 Captura experimento 3 32
Figura 31 Captura de Evento GDevelop 32
Figura 32 Captura de Evento GDevelop 33
Figura 33 Captura de Evento GDevelop 33
Figura 34 Captura de Evento GDevelop 33
Figura 35 Pantalla de resultados 33
Figura 36 Captura de Evento GDevelop 34
Figura 37 Captura de Evento GDevelop 34
Figura 38 Captura de Evento GDevelop 34
Figura 39 Captura de Evento GDevelop 34
Figura 40 Moodboard 35
Figura 41 Simulación de daltonismo..... 36
Figura 42 Paleta de color 36
Figura 43 Antes y después tecla Espacio 39
Figura 44 Posicionamiento de las manos guia 39
Figura 45 Nivel Difícil 40
Figura 46 Nivel Medio..... 40
Figura 47 Nivel fácil..... 40

Introducción

La mecanografía, a menudo subestimada en esta era digital, sigue siendo una habilidad esencial en el mundo moderno. Aunque la tecnología ha evolucionado hacia dispositivos electrónicos y pantallas táctiles, la capacidad de escribir con precisión y velocidad en un teclado sigue siendo crucial en muchos aspectos de la vida cotidiana y profesional.

Por otro lado, el mercado de los videojuegos está en constante crecimiento y ha quedado demostrado que pueden ser una excelente herramienta para el aprendizaje. Esto nos llevó a plantear la siguiente pregunta: ¿Qué sucedería al combinar mecanografía y un juego de ritmo? De esta curiosidad nació Mecabeat, un videojuego cuyo objetivo es poner a prueba la memoria muscular, enfocándose más en la sincronía que en la velocidad, ya que esta última será determinada por el ritmo de la música.

En este documento, describiremos el desarrollo de un prototipo siguiendo un enfoque experimental. También se realizará un análisis de los resultados obtenidos a partir de pruebas con usuarios para evaluar la eficacia y el potencial del juego como herramienta para practicar la mecanografía.

Justificación

La idea de desarrollar un juego de ritmo para practicar mecanografía surgió de la curiosidad por combinar dos elementos aparentemente dispares: el entretenimiento de los videojuegos de ritmo y la práctica de la mecanografía. Los juegos de ritmo representan una forma emocionante de captar la atención de los usuarios (Yu, 2022) y fomentar el desarrollo de habilidades de mecanografía de manera divertida y motivadora.

Este proyecto busca abordar la falta de entusiasmo en el aprendizaje de la mecanografía a través de una experiencia interactiva e innovadora. Al incorporar música y ritmo, el juego puede ofrecer un entorno donde los usuarios practiquen la mecanografía sin sentir la presión del aprendizaje tradicional. Este enfoque tiene el potencial de reducir la monotonía asociada con la mecanografía, haciendo que los usuarios estén más dispuestos a practicar y mejorar sus habilidades.

Además, el desarrollo de este juego brinda la oportunidad de explorar conceptos de diseño gráfico y de experiencia de usuario (UX). En el ámbito del diseño gráfico, se aborda el desafío de crear una interfaz que sea visualmente atractiva y funcional al mismo tiempo, proporcionando una experiencia que guíe a los usuarios a través del juego de manera intuitiva. En cuanto a la experiencia del usuario, el proyecto investiga la creación de una experiencia atractiva y gratificante que motive a los usuarios a seguir jugando y mejorando sus habilidades.

Planteamiento del problema

En la era digital, la mecanografía sigue siendo una habilidad esencial para la productividad y el éxito profesional. (AgileFingers, s. f.)Sin embargo, muchas personas no dominan la mecanografía con la suficiente precisión y velocidad, lo que puede llevar a una disminución en la eficiencia y errores en entornos laborales y académicos. La enseñanza tradicional de la mecanografía, a menudo centrada en ejercicios repetitivos y poco atractivos, puede resultar monótona y desmotivante para los estudiantes y trabajadores.

Por otro lado, los videojuegos han demostrado ser una herramienta efectiva para el aprendizaje y la motivación, con un impacto positivo en el desarrollo de habilidades en diversos campos (Ortíz et al., 2017). Los juegos de ritmo, en particular, ofrecen una experiencia interactiva y envolvente, donde la música y el tiempo juegan un papel fundamental en la dinámica del juego. Esto plantea una oportunidad única para combinar la mecanografía y los juegos de ritmo para hacer que el aprendizaje de la mecanografía sea más atractivo y entretenido.

Para abordar este problema, se necesita una solución innovadora que fusione elementos de ritmo y mecanografía, creando una experiencia educativa y divertida. El desarrollo de un juego con estas características podría ofrecer a los usuarios una forma atractiva de practicar sus habilidades de mecanografía, al mismo tiempo que se divierten.

Al explorar esta solución, este proyecto busca aportar a la comunidad educativa y profesional una herramienta novedosa para el aprendizaje de la mecanografía, con la esperanza de aumentar la eficacia de las prácticas actuales y fomentar el interés en el desarrollo de esta habilidad.

Pregunta de la investigación

¿Qué resulta de combinar la mecanografía y los juegos de ritmo para practicar la mecanografía?

Objetivo General

Desarrollar un prototipo que combine la mecanografía y los juegos de ritmo

Objetivos específicos

1. Describir el proceso de desarrollo a partir de experimentos
2. Comprender la programación sin código a través del programa GDevelop
3. Realizar pruebas con usuarios
4. Analizar los resultados de las pruebas con usuarios y describir las posibilidades de uso del prototipo.

Conceptos y bases teóricas

A continuación, se presentan definiciones y conceptos importantes utilizados para la realización de este proyecto

Mecanografía

Según la Rae, la mecanografía es técnica de escribir a máquina, sin embargo, en este para este proyecto haremos referencia a una de sus técnicas la cual es la mecanografía de tacto:
La mecanografía de tacto, también conocida como mecanografía ciega, es la habilidad de escribir en un teclado utilizando todos los dedos de manera fluida y precisa, sin necesidad de mirar el teclado. Se basa en la memoria muscular y la familiarización con la disposición de las teclas en el teclado estándar QWERTY o en otros diseños como el Dvorak. (Open AI, 2024)

Videojuegos de ritmo

"El género de ritmo en los videojuegos o juegos musicales se define como aquel que se centra en la sincronización y coordinación del jugador con la música del juego, requiriendo habilidades rítmicas y de tiempo para avanzar y tener éxito en el juego." (OpenAI, 2024).
Además, estos juegos suelen presentar mecánicas de juego que requieren que el jugador presione botones o realice acciones específicas en respuesta al ritmo de la música. (Lacasa,2011) De esta forma, el objetivo principal es mantener la sincronización con la música y completar los niveles con la mayor precisión posible. (OpenAI, 2024).

GDevelop

GDevelop es una plataforma de desarrollo de videojuegos que permite a los usuarios crear juegos sin necesidad de tener experiencia en programación. Es una herramienta de código abierto que utiliza una interfaz visual y lógica para diseñar juegos en 2D y 3D, lo que la hace accesible para principiantes y desarrolladores experimentados por igual. Con GDevelop, los usuarios pueden

crear una variedad de juegos, desde simples juegos casuales hasta proyectos más complejos, utilizando una combinación de eventos, condiciones y acciones sin necesidad de escribir código tradicional. (Open Ai, 2024)

La programación sin código en GDevelop se basa en el uso de interfaces visuales y lógicas para crear juegos y aplicaciones sin necesidad de escribir código tradicional. Esto se logra a través de la creación de eventos y acciones mediante la interacción con elementos visuales en la plataforma.

En GDevelop, los usuarios pueden definir eventos, condiciones y acciones utilizando una interfaz intuitiva de arrastrar y soltar, lo que permite crear la lógica del juego sin necesidad de conocimientos de programación profundos.

Gamificación

La gamificación, también conocida como ludificación, implica emplear tácticas, estructuras, interacciones y componentes característicos de los juegos en ámbitos distintos, con el fin de comunicar un mensaje, transmitir contenido o influir en un comportamiento, mediante una experiencia recreativa que fomente la motivación, la participación y el entretenimiento.

Gamificar implica transformar cualquier proceso en una experiencia similar a un juego. Los participantes se convierten en jugadores y, por ende, son el foco central del juego, debiendo sentirse comprometidos, tomar decisiones propias, percibir su progreso, enfrentar nuevos desafíos, interactuar en un entorno social, ser reconocidos por sus logros y recibir retroalimentación instantánea. En resumen, el objetivo de gamificar es que los participantes se diviertan mientras alcanzan los objetivos del proceso gamificado. (Gallego-Durán, F.J., Molina-Carmona, R., & Largo, F.L., 2014)

También, el proceso de gamificar el aprendizaje puede ser mejorado mediante diferentes enfoques: gamificación estructural, gamificación de contenido y gamificación de rendimiento. La gamificación estructural implica utilizar elementos propios de los juegos para dirigir al estudiante a través del material sin modificar el contenido en sí, sino ajustando únicamente la forma en que

este se presenta. Una forma común de implementación consiste en incorporar elementos de puntuación de los videojuegos, como puntos, niveles, insignias, tablas de clasificación y logros, dentro de un entorno educativo. (Karl Kapp, 2024, 4:35)

La gamificación de contenido implica utilizar elementos de juego, sus mecánicas y mentalidad para transformar el contenido y darle un enfoque más similar al de un juego. Una forma habitual de implementar esta gamificación sería incorporar elementos narrativos, desafíos, aspectos intrigantes, misterios y personajes al contenido para captar el interés del estudiante. (Karl Kapp, 2024, 6:22)

La gamificación del rendimiento implica transformar las actividades diarias de un empleado en una experiencia similar a un juego. El concepto es que el rendimiento del empleado en su labor cotidiana esté directamente vinculado con elementos de juego y su eficacia laboral. (Karl Kapp, 2024, 8:10)

Heurística de usabilidad

En 1990, Jakob Nielsen y Rolf Molich presentaron nueve principios para diseñar interfaces de usuario. Las heurísticas son pautas generales que se pueden aplicar para diseñar productos digitales que sean más accesibles y sencillos de usar:

1. Diálogo simple y natural:

   Los diálogos no deben contener información irrelevante o que raramente se necesite. Cada unidad de información superflua compite con las unidades relevantes, reduciendo su visibilidad relativa. La información debe presentarse en un orden natural y lógico.

2. Hablar el lenguaje del usuario:

   El diálogo debe expresarse claramente con palabras, frases y conceptos familiares para el usuario, en lugar de términos orientados al sistema.

3. Minimizar la carga de memoria del usuario:

La memoria a corto plazo del usuario es limitada. El usuario no debería tener que recordar información de una parte del diálogo a otra. Las instrucciones para el uso del sistema deben ser visibles o de fácil acceso cuando sea necesario. Las instrucciones complejas deberían simplificarse.

4. Ser coherente:

   Los usuarios no deberían tener que preguntarse si diferentes palabras, situaciones o acciones significan lo mismo. Una acción del sistema debería ser siempre alcanzable con una acción particular del usuario. La coherencia también implica coordinación entre subsistemas y entre sistemas independientes que comparten una base común de usuarios.

5. Proporcionar retroalimentación:

   El sistema siempre debe mantener al usuario informado sobre lo que está sucediendo, proporcionando la retroalimentación apropiada en un tiempo razonable.

6. Proporcionar salidas claramente marcadas:

   Un sistema nunca debe atrapar a los usuarios en situaciones sin una salida visible. Los usuarios a menudo eligen funciones del sistema por error y necesitarán una "salida de emergencia" claramente marcada para salir del estado no deseado sin tener que pasar por un diálogo extenso.

7. Proporcionar atajos:

   Las características que hacen que un sistema sea fácil de aprender—como diálogos detallados y pocos campos de entrada en cada pantalla—pueden ser tediosas para el usuario experimentado. Los atajos inteligentes, invisibles para el usuario principiante, pueden incluirse en un sistema para satisfacer tanto a usuarios inexpertos como experimentados.

8. Proporcionar buenos mensajes de error:

   Los buenos mensajes de error son defensivos, precisos y constructivos. Los mensajes de error defensivos atribuyen el problema a deficiencias del sistema y nunca culpan al usuario. Los mensajes de error precisos proporcionan información exacta sobre la causa del problema. Los mensajes de error constructivos ofrecen sugerencias útiles al usuario sobre qué hacer a continuación.

9. Prevención de errores:

   Incluso mejor que buenos mensajes de error es un diseño cuidadoso que evite que ocurra un problema desde el principio.

Experiencia de usuario

La Experiencia de Usuario (UX) engloba la impresión que experimenta una persona al relacionarse con un producto, servicio o plataforma digital. Esta experiencia incluye todos los aspectos de la interacción, desde la forma de navegar y el diseño visual, hasta la facilidad de uso y la satisfacción que obtiene el usuario. En resumen, se trata de la percepción que tiene un usuario al utilizar un sitio web, una aplicación móvil u otro producto digital. (Purita, 2023)

Metodología

La metodología de este proyecto consistió de las siguientes etapas:

1. ***Estado de arte:***

En esta etapa se investigará con el fin de realizar un estado del arte que de luces del alcance y posibilidades del proyecto

2. ***Planteamiento inicial:***

En esta etapa se hará un bosquejo del videojuego Mecabeat y teniendo en cuenta lo propuesto por Jesse Schell, en el libro The Art of Game Design estos son los cuatro elementos básicos del videojuego:

- Mecánica: Hace referencia a las reglas que rigen el juego y determinan cuál es el objetivo o la meta que los jugadores deben alcanzar.
- Historia: Consiste en la serie de eventos y situaciones que forman el relato principal que guía las acciones del personaje en el juego.
- Estética: Corresponde al estilo visual y auditivo del juego, lo cual contribuye a darle su identidad y debe estar alineado con la narrativa.
- Tecnología: Comprende todos los requerimientos técnicos necesarios para ejecutar el juego.

3. ***Experimentación:***

En esta etapa se procede a experimentar la programación sin código del programa GDevelop. Para ello se plantean objetivos a alcanzar y a través de la prueba y error se buscará el resultado óptimo.

4. ***Prueba con usuarios:***

En esta etapa se realizaron pruebas del prototipo con posibles usuarios, con el fin de analizar el nivel de diversión que provoca la mecánica, la percepción sobre la estética y buscar posibles errores o bugs. Además, se tendrá en cuenta el conocimiento sobre mecanografía de los usuarios.

5. ***Correcciones:***

En esta etapa, a partir de los resultados de las pruebas con usuarios, se realizarán los cambios pertinentes y se solucionarán errores en caso de que el prototipo los presente. esta etapa nos devuelve a la etapa de experimentación.

Estado del arte

Slash it 2

Es un colorido juego arcade 2D en el que debes presionar letras, números y escribir palabras en el momento y dirección correctos. El juego ofrece seis modos de juego, tablas de puntuaciones y clasificación. Para jugar, se requiere cierto nivel de mecanografía y habilidad con el teclado. Las teclas aparecen al ritmo de la música, y el jugador debe presionarlas para evitar quedarse sin tiempo o permitir que se acumulen demasiadas en la pantalla. Sin embargo, el juego no requiere que sigas el ritmo de la música al presionar las teclas.Osu!taiko

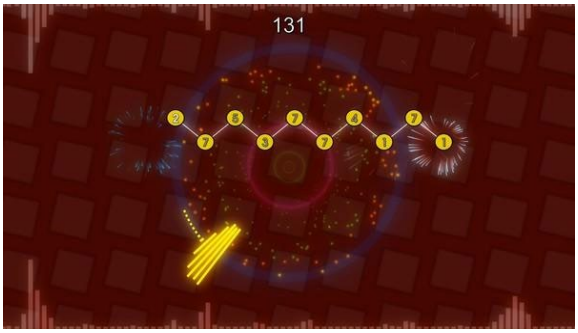


Figura 1 Captura de gameplay

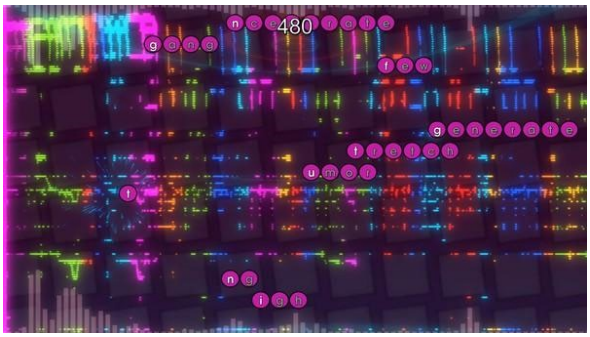


Figura 2 Captura de gameplay

Osu!taiko

osu!taiko es un modo de juego en Osu! que incorpora algunos elementos del juego de ritmo japonés Taiko no Tatsujin.Destacó la forma en la que los elementos aparecen en pantalla (de derecha a izquierda) y el sistema de vidas donde necesitas al menos el 50% para pasar el nivel.



Figura 3 Captura de gameplay

NotITG

NotITG es un juego de ritmo con visuales extravagantes, diseñado para facilitar a los creadores de archivos mod implementar sus ideas. Su objetivo es mantener la compatibilidad con todos los archivos mod existentes de StepMania 3.95 y In The Groove, y ser el entorno definitivo para crear y disfrutar de ese contenido. Destacó su función de editar canciones ya que permite crear canciones totalmente personalizadas.



Figura 5 Pantalla Inicio NotITG

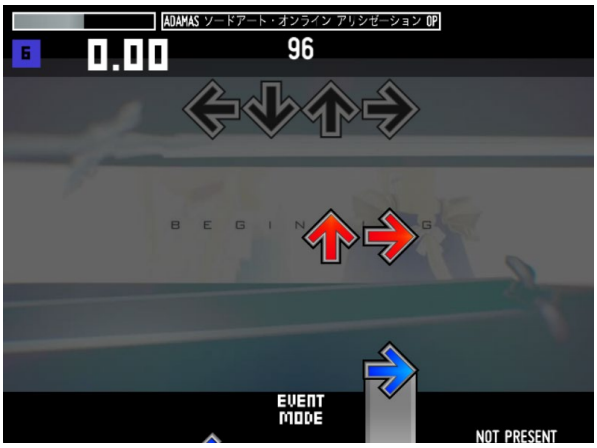


Figura 4 Captura de gameplay

TypingClub

Es una página web que ofrece cursos gratuitos (algunas lecciones y juegos de pago) para aprender y practicar mecanografía.

Lo que destaco son las lecciones en general, el orden y la forma en la que te van enseñando cada tecla, por filas (Guía, superior e inferior), los caracteres, números, patrones y palabras más comunes y desafiantes. Asimismo, destaco la forma de señalizar la posición de las manos, es un buen apoyo para el aprendizaje y cuando se domina es fácil de ignorar.

Este curso será la principal fuente de información respecto a mecanografía para proyecto.

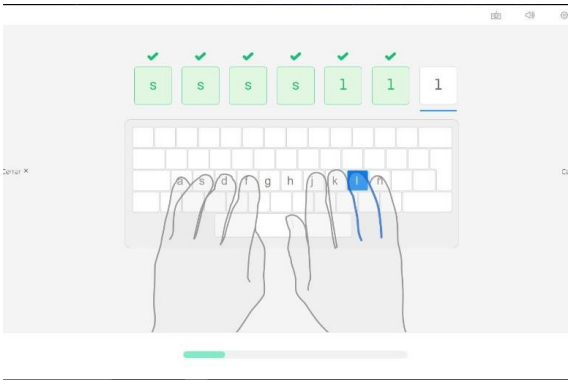


Figura 6 Captura de Lección

Curso Mecanet

Es un programa que ofrece un curso totalmente gratuito de mecanografía, posee 20 lecciones con sus exámenes. Además posee minijuegos para practicar la mecanografía. En este programa destacó la forma que está diseñada cada lección con palabras que solo usan las letras de cada lección.



Figura 7 Captura de Lección

Digitopolis 2

En este libro escrito por Jose David Cuartas,podemos encontrar un basica guia de como utilizar el programa GDevelop. Este libro servirá como material de apoyo a la hora de plantear y desarrollar los experimentos

Game Example Rhythm

En este juego de ejemplo desarrollado por GDevelop de como crear un Dance Dance Revolution o Friday Night Funkin.

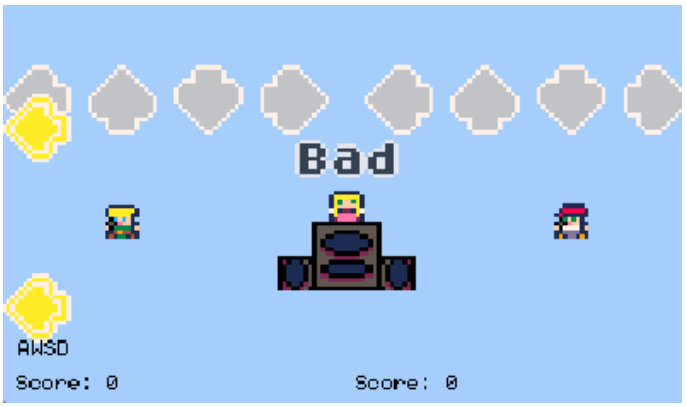


Figura 8 Captura de gameplay

Game Example Keyboard practice

En este juego de ejemplo desarrollado por GDevelop el jugador debe pulsar las teclas lo más rápido posible en un determinado tiempo. Destacó que este ejemplo muestra cómo se pueden usar las variables directamente en expresiones como cambio de animaciones.



Figura 9 Captura de gameplay

Planteamiento inicial

Mecánica

El jugador deberá pulsar la tecla que aparezca en pantalla sobre la pista cuando esta esté sobre la ranura siguiendo el ritmo de la música. Por otra parte, en pantalla habrá una guía visual que indicará con qué dedo se debe pulsar cada tecla y la pista indicará en que línea el teclado pertenece la tecla. Además, el jugador contará con 5 vidas las cuales se perderán cuando este no pulse a tiempo las teclas en pantalla.

Historia

No posee una historia como tal, sino que una serie de niveles donde la dificultad aumentará gradualmente.

Estética

Para este Juego de ritmo se pensó utilizar música electrónica ya que la composición de este género en particular es más repetitiva y con un ritmo claramente marcado. De esta forma se escogieron canciones de NoCopyrightSounds porque es música libre de regalías. Asimismo en base a la estética de este sello discográfico se pensó en una estética de Cyberpunk.

Tecnología

Este videojuego se desarrolla para la plataforma de PC donde se requiere de un teclado QWERTY. Por otra parte, para este proyecto se utilizaron tres programas: Adobe Illustrator para la bocetación y generación de gráficos, Audacity para la edición de sonido y GDevelop para el montaje y programación del prototipo. Además se utilizó la página web jsfxr para realizar los efectos de sonido del juego.

Método

Primero se creó un diagrama de flujo de las pantallas que contaría el juego.

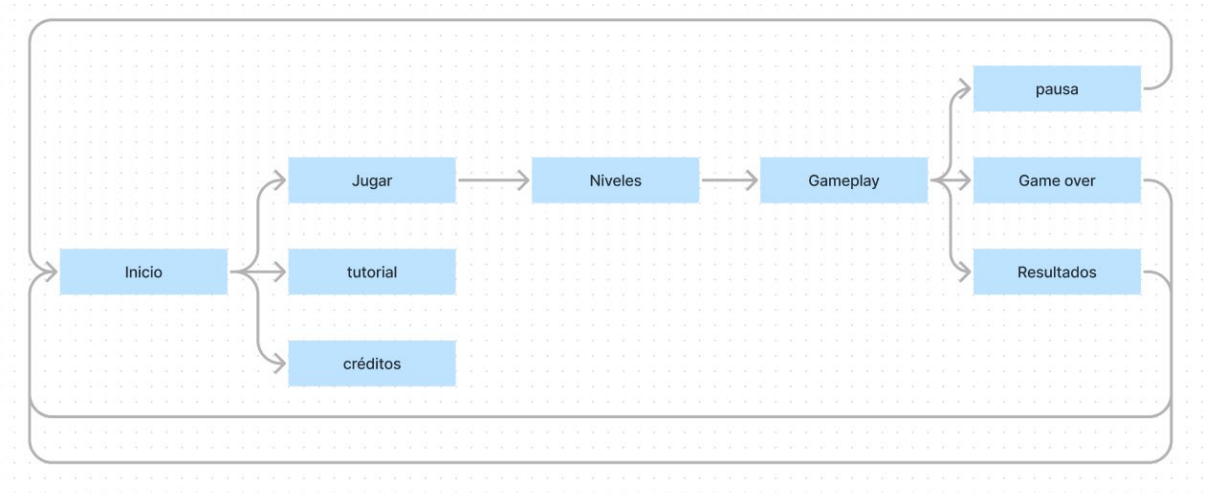


Figura 10 Mapa de navegación

Después, en illustrator, se crearon las diferentes pantallas y objetos. Cabe mencionar que en la primera etapa de desarrollo no se había propuesto una estética en específico, es así que, como se menciona en las heurísticas de usabilidad, se generaron gráficos que fueran altamente reconocibles y minimalistas.



Figura 11 Bocetación de las pantallas de Mecabeat

Experimento 1 - prototipo de la mecánica con la fila guía

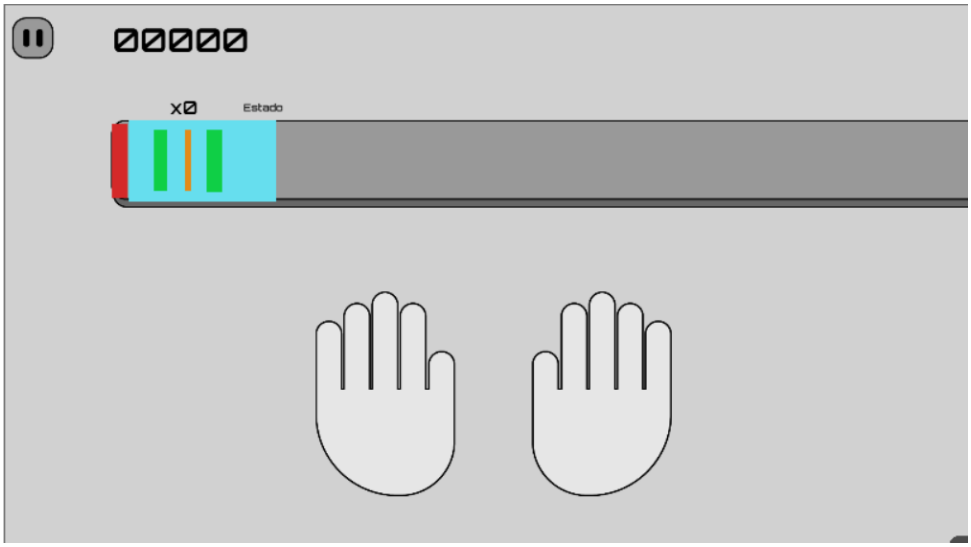


Figura 12 Captura del experimento 1

El fin de este primer experimento era probar la mecánica del juego y para ello solo se utilizó las teclas de la fila guía.

Primero se añadieron los diferentes objetos como teclas, la pista, las manos como sprites, y los textos como objetos de texto y dentro de las opciones se añadió la tipografía. Las diferentes barras de colores rojo, azul, verde, naranja son mosaicos y estos elementos se ocultan al inicio de la escena. Más adelante se profundizará en su funcionamiento. asimismo al inicio de la escena comienza la música.

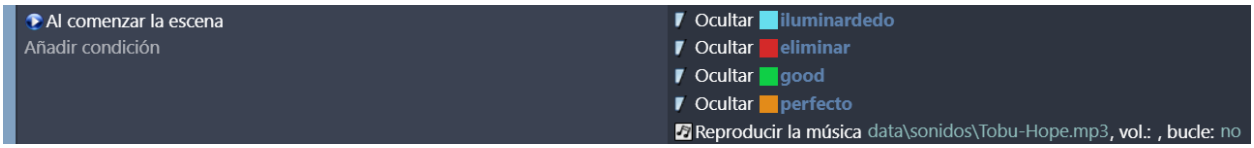


Figura 13 Captura de Evento GDevelop

Movimiento de las teclas

Para el movimiento de las teclas simplemente se añadió una acción para cada tecla de cambiar su posición en el eje X mediante una sustracción esto hace que la letra se mueva de derecha a izquierda desde una posición inicial.

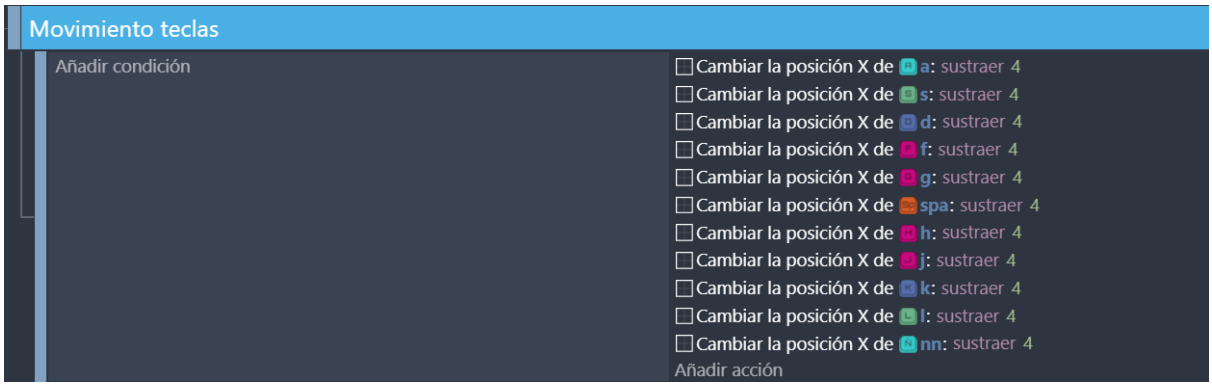


Figura 14 Captura de Evento GDevelop

Actualización de textos en pantalla:

Para que funcionaran los textos de puntuación, combo y estado se creó una acción de "cambiar el texto" enlazando cada texto con una variable. En el caso de combo fue necesario utilizar el signo "+" para combinar un texto fijo como la "x" y el valor de la variable.

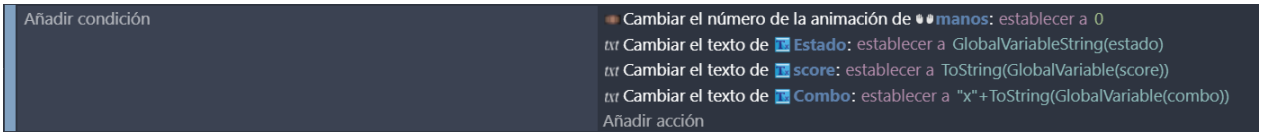


Figura 15 Captura de Evento GDevelop

Iluminación de los dedos de las manos:

Cada vez que una tecla este cerca de la ranura, la idea es que las manos se muestren con qué dedo se debe pulsar cada tecla. Para lograrlo, primero se crearon distintos sprites de las manos con cada dedo iluminado. Hay que mencionar que ese es por qué las teclas tienen distintos colores para diferenciar cada dedo. Luego se creó un objeto tipo mosaico llamado "iluminardedo" y se situó sobre la ranura y un poco antes.

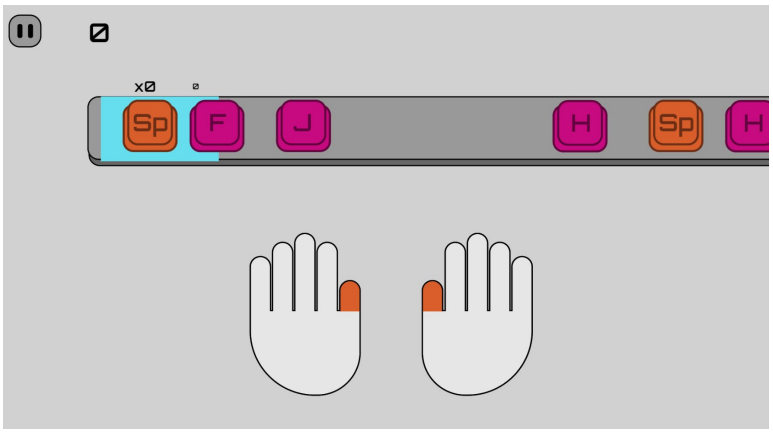


Figura 16 Captura del experimento 1

Luego se creó un evento donde la condición era que el objeto "iluminardedo" estuviera en colisión con una tecla, en este caso la "a", y en la acción es cambiar el número de animación de las manos al correspondiente con el dedo meñique y así se hizo con cada una de las teclas.

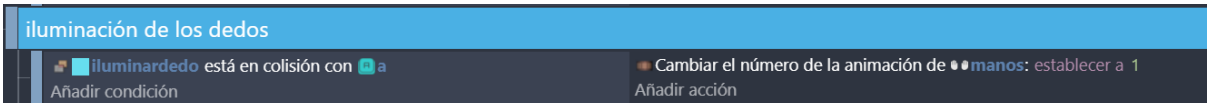


Figura 17 Captura de Evento GDevelop

Colisión con la ranura y sistema de precisión:

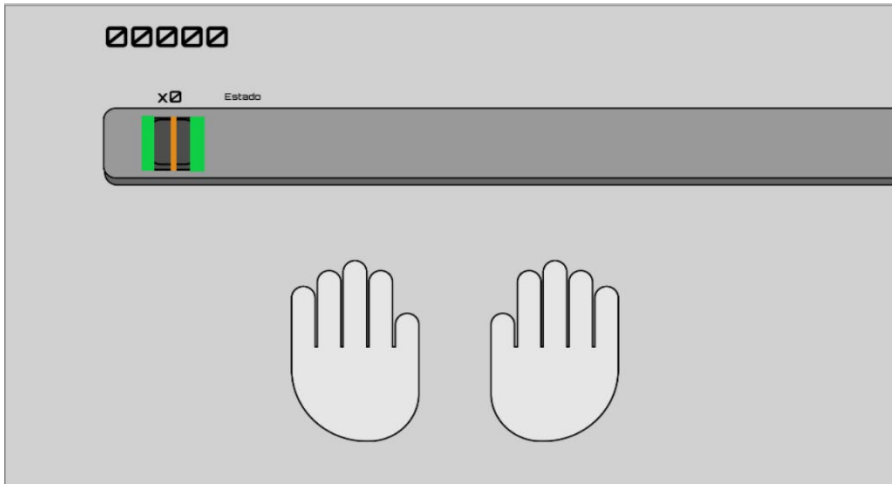


Figura 18 Captura sistema de precisión

Cuando una tecla llega a la ranura y el jugador pulse la tecla correspondiente, la tecla en pantalla debe eliminarse y sumar puntos, además de sumar a la variable “combo” y se añadió un sonido de una tecla siendo presionada.



Figura 19 Captura de Evento GDevelop

En cuanto al sistema de precisión fue dividido en “perfecto” y “good” (bueno). Este funciona con los objetos naranja y verde. Similar al sistema de colisión de la pista cuando una tecla este en colisión con “good”, se sumará un puntaje de 100 a la variable “score” y que en pantalla apareciera un sprite llamado “estado” el cual muestra la palabra “bueno” y tiene una animación llamada “quitar” que reduce su opacidad a 0 y destruye dicho objeto. Además, en el caso de perfecto, el cual es cuando la tecla está totalmente alineada con la ranura, se agregó la condición que no estuviera en colisión con “good”.

Por otro lado, fue necesario cambiar el mapa de colisión de las teclas.

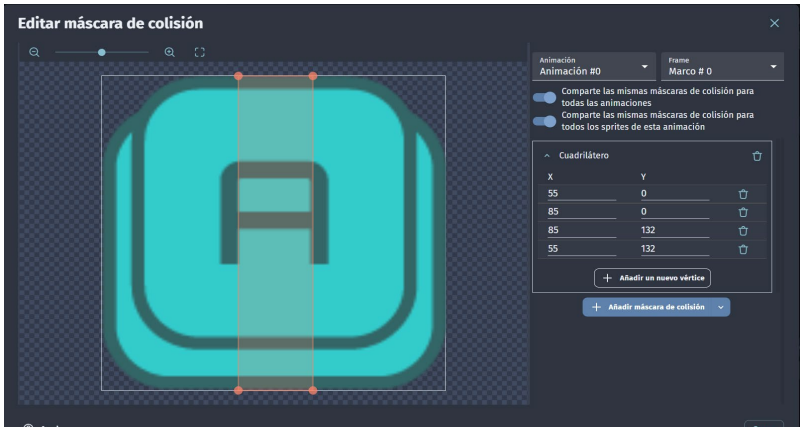


Figura 20 Captura máscara de colisión

Eliminación de teclas cuando se falla al pulsar en la ranura

Al borde de la pista hay una barra roja la cual se encarga de eliminar las teclas cuando el jugador no las pulsa, para ello en la condición se puso que cuando una tecla este en colisión con esta barra, en las acciones se elimina dicha tecla, se emite un sonido, se resta uno de la variable vida y se restablece el combo.

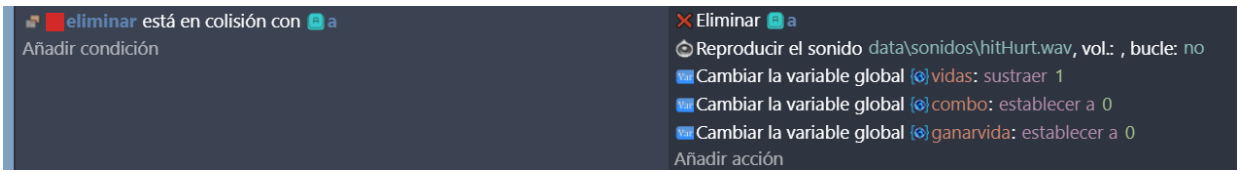


Figura 21 Captura de Evento GDevelop

Generación de las teclas/notas

Como estas se deben generar en un momento justo de la canción, primero se calculó cuántos segundos tarda una tecla desde el borde de la pantalla a la ranura, para ello se creó un objeto tipo texto llamado tiempo y, al inicio de la escena, un cronómetro llamado tiempo. Luego se creó un evento siendo la condición cuando una tecla es presionada, el objeto texto "tiempo" mostrará el valor del cronómetro. En nuestro caso dio 6.946 segundos.



Figura 22 Captura de Evento GDevelop

Luego con la ayuda del juego NotITG en su sección de editar, se marcó todas las notas de la canción y este programa nos muestra el segundo exacto de cuando ocurren. Entonces solo era cuestión de restar este último valor con los 6.946 de desfase para saber en que segundo se debe generar la tecla para que al llegar a la ranura vaya acorde con la música.

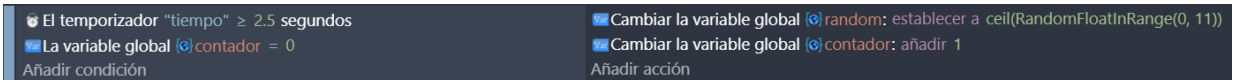


Figura 23 Captura de Evento GDevelop

Ahora como acciones del evento tenemos a cambiar el valor de la variable random esta variable indica qué tecla se va a generar, pero exactamente en esta parte debe cambiar su valor por un valor aleatorio y la variable contador es para evitar que la acción se ejecutará más de una vez.

Aquí podemos ver un fragmento de cómo funciona la variable “random”, cuando esta tiene un valor específico se genera una tecla.



Figura 24 Captura de Evento GDevelop

Experimento 2: palabras aleatorias y las letras de una palabra

Para este experimento se había planteado que en el juego en vez de teclas aleatorias aparecieran palabras, entonces era necesario averiguar cómo generar las teclas a partir de una palabra.

Palabra y letra aleatoria:

Para esta parte se creó una variable tipo matriz llamada palabras que contenía nueve palabras, además se creó la variable “palabraRandom” para seleccionar una palabra de forma aleatoria. Entonces en el evento cada vez que el temporizador llega a un valor de un segundo se selecciona una palabra aleatoriamente y en el texto “palabra” se actualiza automáticamente cada vez que ocurre ya que muestra el valor de la variable matriz palabra de esta forma: palabra[variable(palabraRandom)]. Por otra parte la letra aleatoriamente en vez de ser aleatoria cada segundo se suma uno a la variable “letraActual” y en el texto se muestra el valor de la matriz letra de esta forma: letra[variable(letraActual)].

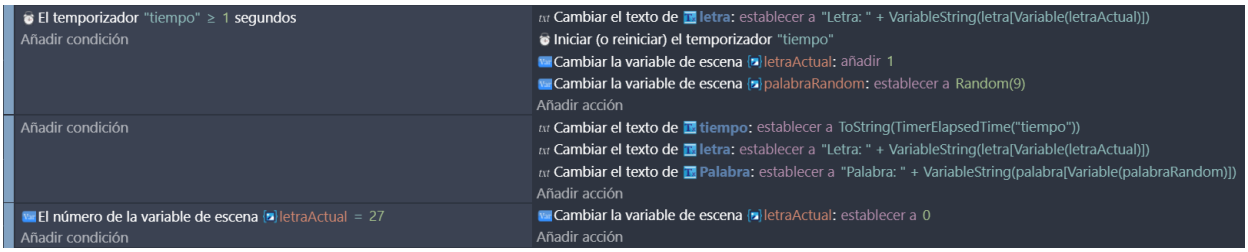


Figura 25 Captura de Evento GDevelop

Las letras de una palabra aleatoria

para esta parte la idea era que el programa diga una palabra, la cantidad de letras que tiene, cada segundo muestre las letras que lo conforma y cuando termine se seleccione aleatoriamente otra palabra.

para ello se creó una variable tipo matriz llamada “letraspalabra” y cada hijo sería otra matriz con las letras de una palabra luego una variable “palabraActual” la cual indicaria la palabra dentro de la matriz palabras y la variable “palabralength” que indica el número de letras que tiene la “palabraActual”, luego se creó una variable encargada de avanzar entre las letras de la matriz “letrapalabras” esta variable se llamó “restraletras” cada segundo se añade 1 y cuando su valor es igual a la variable “palabraLength” se selecciona otra palabra y el restaletras se reinicia a 0



Figura 26 Captura de Evento GDevelop

Experimento 3 - Prototipo con todas las letras

Para este experimento se duplicó el experimento 1 pero se añadió el sistema de vidas y, basándonos en los resultados del experimento 2, se cambió la forma en la que se generaban las teclas.

Vidas:

la idea es que el jugador tenga cinco vidas las cuales perderá si se equivoca, sin embargo tiene la posibilidad de recuperarlas cada vez que alcance cinco de combo. Para ello se creo una variable llamada “ganarvida” y cada vez que el jugador pulse correctamente una tecla se sumara 1 a esta variable cuando se llega a un valor de cinco se le suma uno a la variable “vidas”, además se programo las animaciones para el indicador de las vidas dependiendo de su valor, por otra parte para evitar errores cuando la variable vidas llegue a un valor de seis, se cambia su valor a cinco.



Figura 27 Captura de Evento GDevelop

Generación de teclas:

primero se creó una matriz llamada “palabras” y cada uno de sus hijos es una matriz con las letras de una palabra, luego tenemos la variable “palabraLength” la cual cuenta el número de hijos, en nuestro caso letras, de una palabra en la matriz “palabras”. Luego la variable “letraActual” muestra la letra que se está seleccionando gracias a la variable “restaletras”, la cual cada vez que se genera una nueva tecla aumenta su valor por uno.

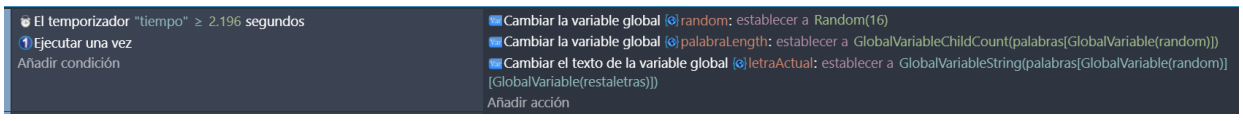


Figura 28 Captura de Evento GDevelop

luego cuando el valor de la variable “restaletras” es igual al de la variable “palabralength” se

selecciona una nueva palabra a través de la variable “random” y se reinicia el valor de la variable “restaletras” a 0.

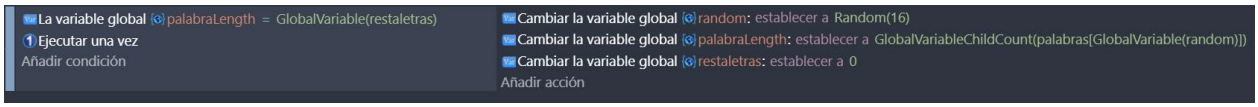


Figura 29 Captura de Evento GDevelop

De esta forma ahora podemos ver cómo se generan palabras en la pista del juego

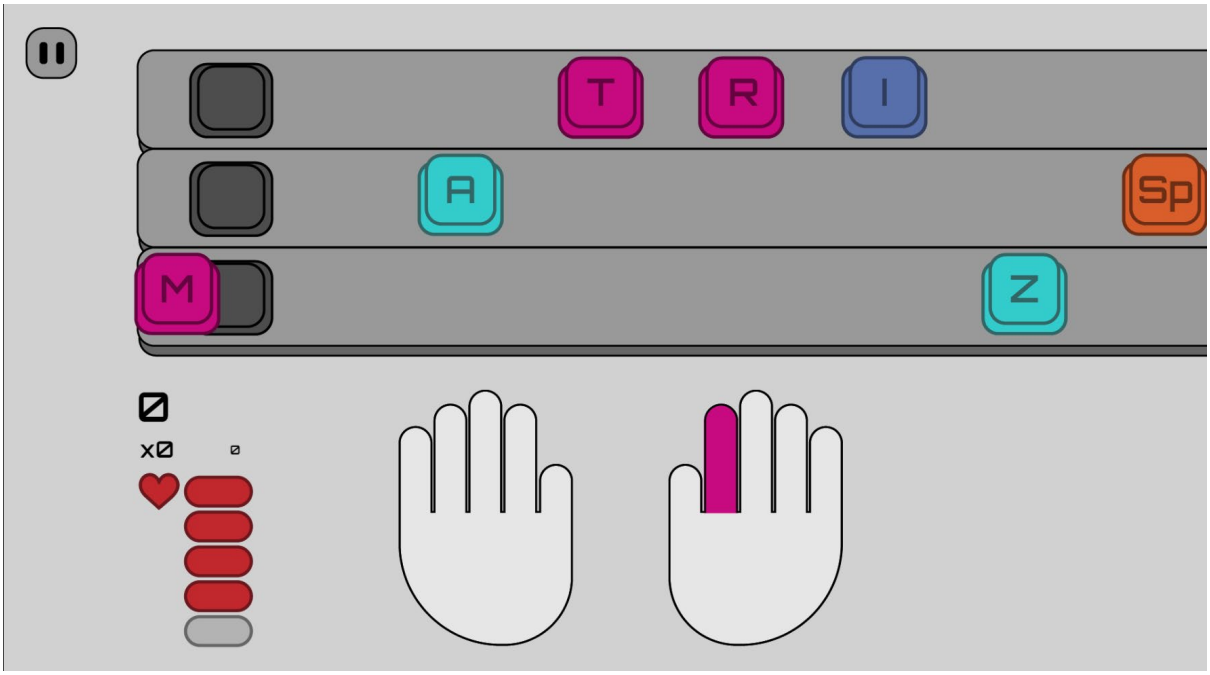


Figura 30 Captura experimento 3

Navegación

Escena pausa:

Se creó un objeto tipo sprite llamado pausa con la imagen del boton de pausa y cuando el cursor le de click o se pulse la tecla escape se activa la acción de pausar la escena y comenzar una nueva llamada pausa.

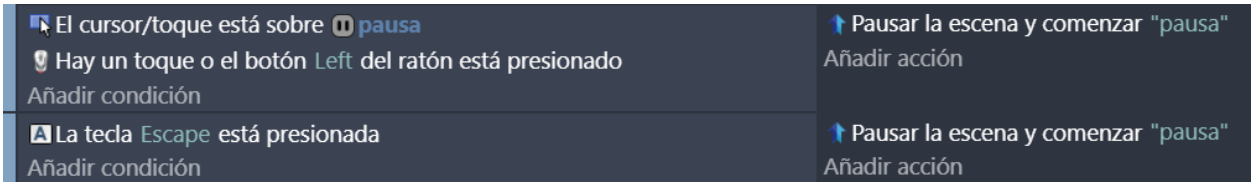


Figura 31 Captura de Evento GDevelop

por otro lado en la escena pausa para volver se crearon tres botones los cuales continuar nos regresa al gameplay, reiniciar nos lleva al principio del experimento 3 reiniciandolo y por último inicio nos lleva al inicio del juego.

Esto genera un bug cuando se da a la opción de continuar generando que la música deje de sonar.

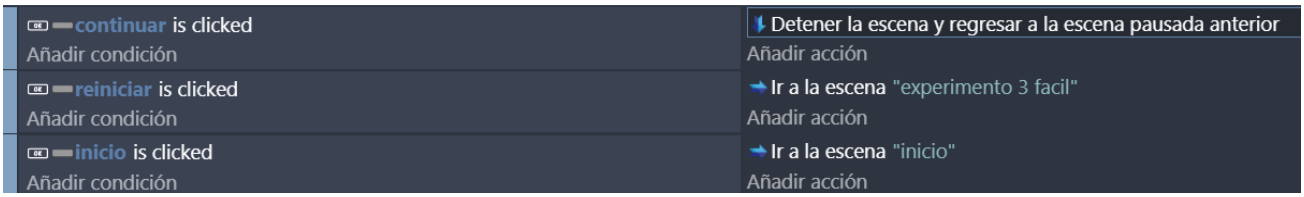


Figura 32 Captura de Evento GDevelop

Escena game over:

Al inicio de la escena se oculta un objeto tipo mosaico cambiando su opacidad a 0 y cuando la variable vidas llega a cero se activa una animación tipo interpolación la cual cambia su opacidad de 0 a 255 y cuando esta termina nos lleva a la escena “game over”

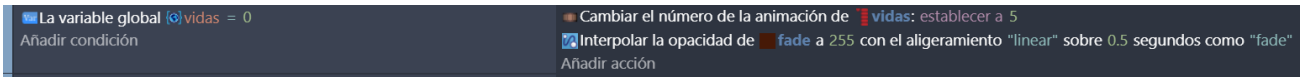


Figura 33 Captura de Evento GDevelop

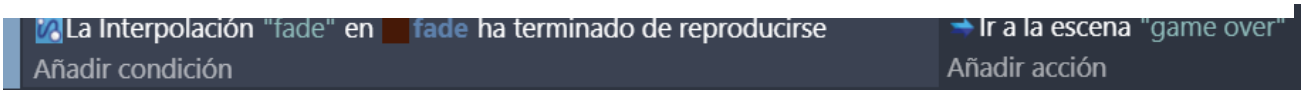


Figura 34 Captura de Evento GDevelop

Escena resultados

Esta escena aparecerá cuando el jugador haya completado exitosamente el nivel.



Figura 35 Pantalla de resultados

Para ello se utilizó la acción de cambiar el texto de un objeto de texto por su respectiva variable

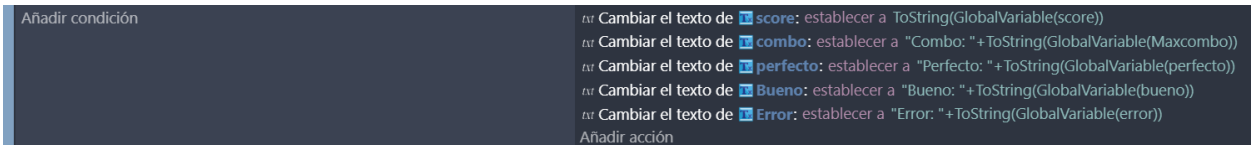


Figura 36 Captura de Evento GDevelop

Además se crearon nuevas variables que contarán el número de perfectos, buenos, errores, y el combo máximo. por otro lado se creó una variable llamada “hiscore” que representa la puntuación más alta hecha en el nivel y esta será guardada en el almacenamiento del sistema, para ello se creo que al inicio de la escena se cargará el almacenamiento para cargar esta variable

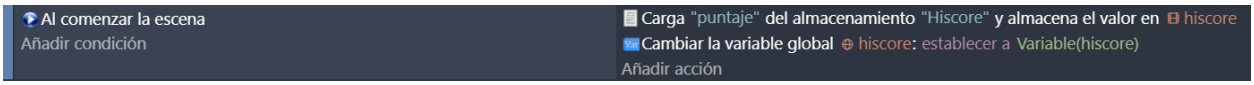


Figura 37 Captura de Evento GDevelop

luego en otro evento se hacía la comparación si la variable “hiscore” es menor que la variable “score” siendo el puntaje del jugador, en la acción se cambiaba el valor de “hiscore” por el de “score”, se guardaba en el almacenamiento y se cambiaba la animación del fondo por una que indicaba al jugador que obtuvo un nuevo puntaje.

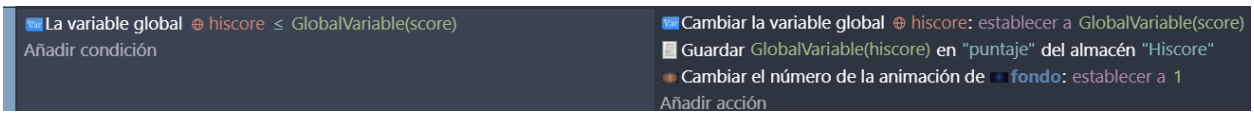


Figura 38 Captura de Evento GDevelop

Escena inicio

Para esta escena principalmente se utilizó botones para la navegación, creando la condición cuando este botón sea clickeado, ir a la respectiva escena.

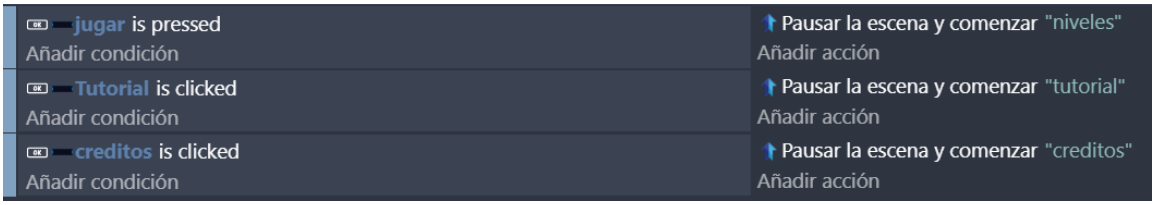


Figura 39 Captura de Evento GDevelop

Estética

Se comenzó generando un moodboard con imágenes de elementos de interfaces de ciencia ficción y cyberpunk, donde predominan las tonalidades azuladas y los gráficos rectilíneos con diversas diagonales.

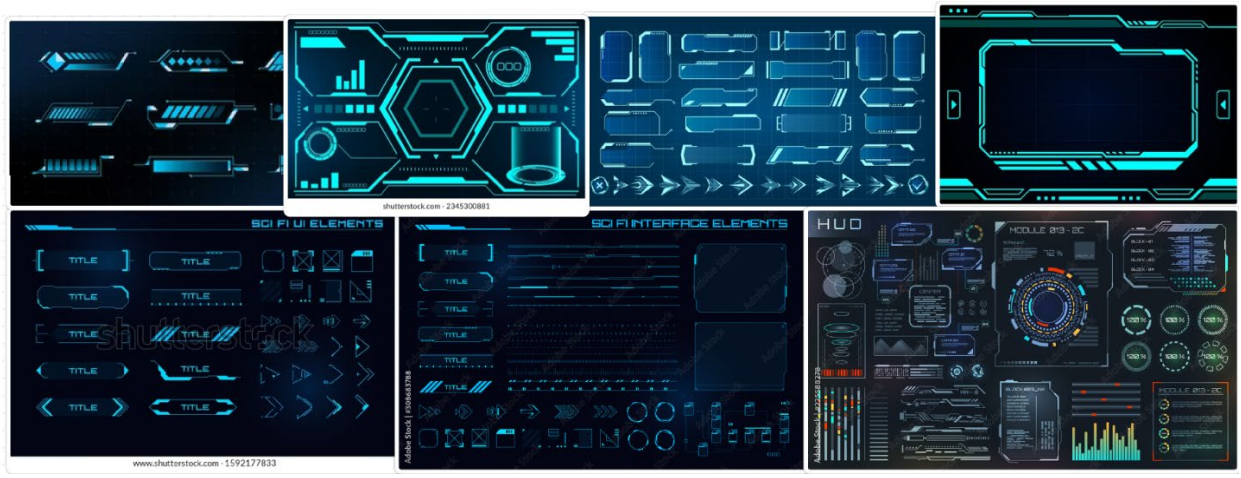


Figura 40 Moodboard

La estética futurista cyberpunk es un estilo visual y cultural que ha ganado popularidad a lo largo de las décadas, especialmente en la ciencia ficción, el arte, la moda y el diseño. Se caracteriza por combinar elementos futuristas, tecnológicos y distópicos con influencias de la cultura urbana y la alta tecnología.

La tecnología desempeña un papel central en la estética cyberpunk. Se presentan avances tecnológicos impresionantes, como la realidad virtual, la inteligencia artificial, los implantes cibernéticos y la realidad aumentada.

Además, la estética cyberpunk se caracteriza por su uso de neones y luces brillantes, que crean una atmósfera vibrante y futurista en las ciudades nocturnas. Estas luces intensas contrastan con la oscuridad y la decadencia del entorno urbano.

De esta forma se creó una paleta de colores en distribución de pentagrama, eligiendo como color principal un azul celeste, según la psicología del color de Eva Heller, este color representa la simpatía, armonía y confianza. (Heller, 2015)

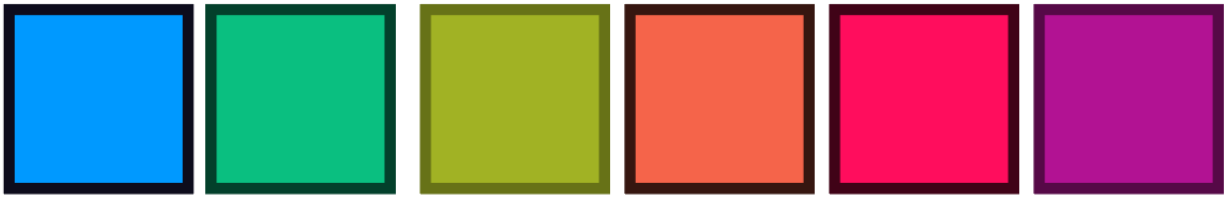


Figura 42 Paleta de color

Además, la paleta fue puesta a prueba en un simulador de daltonismo de la herramienta online Adobe color.

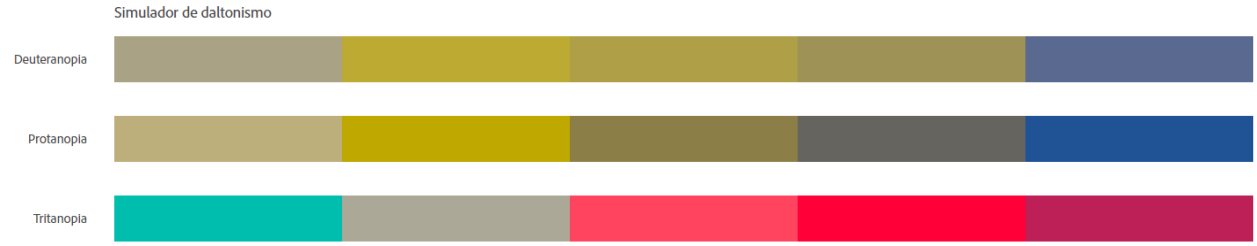


Figura 41 Simulación de daltonismo

Para la tipografía se escogieron dos Orbitron y JetBrains Mono, se escogieron por ser sans serif y ser principalmente geométricas con un estilos y toques futuristas. La Orbitron se usan para los titulares y botones mientras que la JetBrains Mono se usa en las teclas, ya que la orbitron presentaba caracteres muy similares como la D pareciendo mucho a una O lo que podría causar confusiones a la hora del gameplay.

Pruebas del prototipo con usuarios

Descripción de primera sesion con usuarios

Tras completar el Experimento 3, se llevó a cabo una prueba informal con usuarios durante una sesión de la clase Proyecto de Grado 1. En esta ocasión, dos usuarios probaron el juego.
El primer usuario mostraba una constante mirada al teclado y no logró comprender el uso de la tecla "Sp" y el indicador del combo, luego no fue capaz de completar el nivel. No obstante, sugirió la inclusión de elementos estéticos inspirados en la estética japonesa Y2K.

Por otro lado, el segundo usuario demostró un mayor dominio del teclado, aunque también presentó dificultades para entender el uso de la tecla "Sp". Este usuario sugirió que debería ocurrir algo cuando se pierde una tecla, ya que aparentemente el sonido que se reproduce cuando el jugador falla no es lo suficientemente alto y no fue percibido por este usuario.

En conclusión, se decidió cambiar la tecla del espacio, añadir la etiqueta comba y aumentar las vidas a 10 para ayudar al jugador

Conclusiones de la segunda prueba

Para esta segunda prueba de usuarios, se les preguntó si habían tenido experiencias previas con juegos de ritmo. La mayoría respondió que sí, mencionando "Just Dance" como su principal referencia. Además, se indagó acerca de su nivel de mecanografía. La mayoría conocía el término, pero no lo había estudiado formalmente. También se descubrió que los dedos índice y corazón son los más utilizados para escribir.

Con respecto al prototipo, ningún usuario fue capaz de completar el nivel. Algunos comentaron que era demasiado difícil para quienes no están familiarizados con la mecanografía. Esta prueba demostró que el nivel desarrollado tiene un grado de dificultad algo alto. Por otro lado, se detectó

un error que ocurría al regresar del modo de pausa: la música no se reanudaba y dejaba de sonar. Además, a veces, después de reiniciar un nivel, las teclas no estaban sincronizadas con la música.

Las preguntas y las respuestas exactas se podrán encontrar en los anexos del documento.

Correcciones

Cambio de la tecla espacio y añadir el indicador de combo

Se realizó el cambio visual de la tecla espacio pasando de ser su abreviatura “Sp” a “Esp” y al indicador de combo de “x10” por “combo: 10”



Figura 43 Antes y después tecla Espacio

Aumentar las vidas

Como lo demostró la primera prueba de usuarios para bajar la dificultad del juego se decidió aumentar las vidas de 5 a 10 para darle mayor margen al jugador.

Posición de las manos guia

Bajo el comentario de un participante de la prueba que le gustaría que las manos guia estuvieran alineadas con la ranura al momento del gameplay, se procedió a cambiar de posición de las manos guia y hacer que el salto sacádico que realizan los ojos fuera más corto.

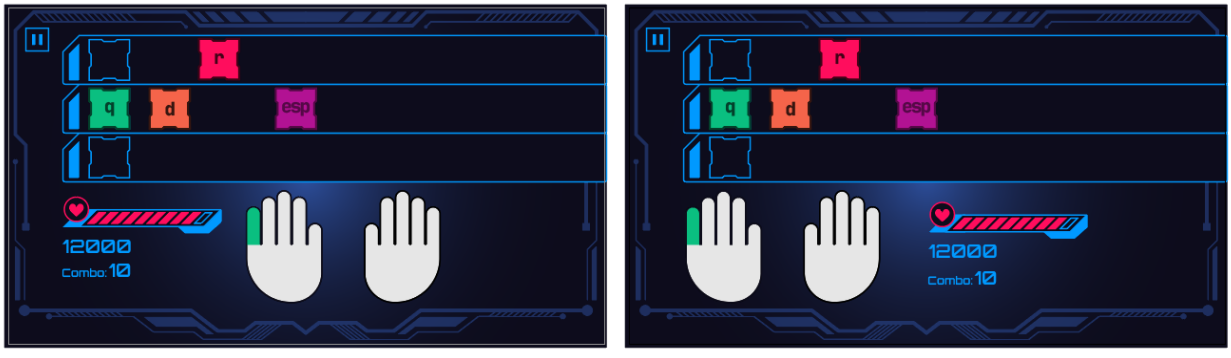


Figura 44 Posicionamiento de las manos guia

Niveles de dificultad

La segunda prueba con usuario demostró que el experimento 3 fácil no es tan fácil, entonces se replanteó como serían los niveles de dificultad del juego. De esta forma se definió que el nivel fácil fuera solo con la línea guía del teclado, osea con una sola pista, que el nivel medio fuera con las líneas superior y la fila guía , o sea dos pistas y por último el nivel difícil fuera con todas las teclas del teclado.



Figura 47 Nivel fácil

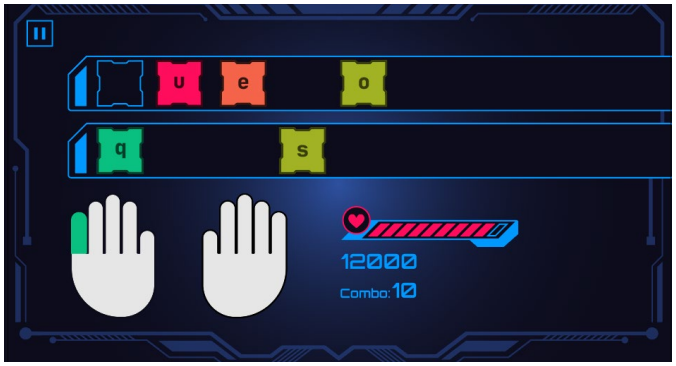


Figura 46 Nivel Medio

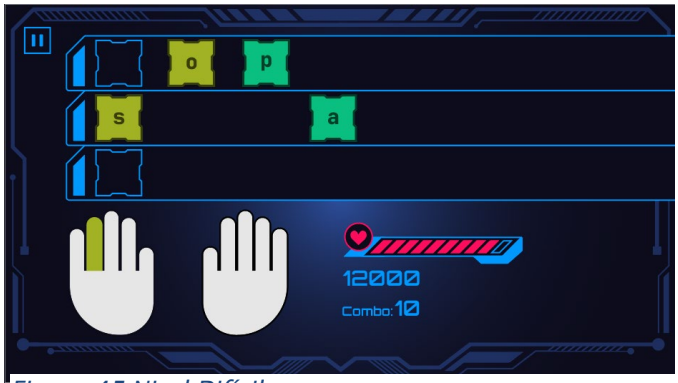


Figura 45 Nivel Difícil

Conclusiones

En definitiva se lograron los objetivos de crear un prototipo del juego Mecabeat y se pudo demostrar a través de las pruebas con usuarios que es una herramienta alternativa divertida para practicar mecanografía. Además, es recomendable tener cierto nivel de mecanografía para disfrutar este juego, pero con la corrección de los niveles ya se abre la posibilidad de que usuarios nuevos en el estudio de la mecanografía puedan disfrutar de este juego.

Por otro lado, la programación sin código del programa GDevelop a través del método de arrastrar y soltar es una alternativa bastante efectiva para desarrollar videojuegos sin conocimientos profundos en lenguajes de programación, solo es necesario nociones básicas en la lógica de programación.

Para terminar, como trabajo futuro para este juego es desarrollar más niveles, exportar el ejecutable y publicarlo en plataformas para juegos indie como itch.io.

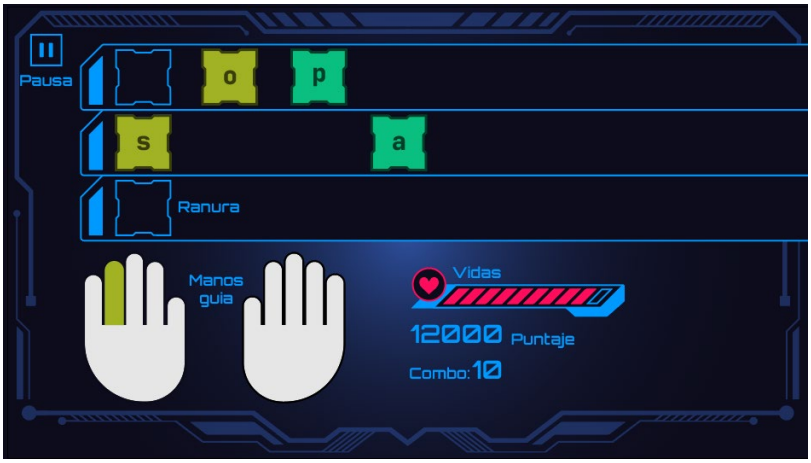
Anexos

Manual de usuario de Mecabeat

Para navegar sobre la interfaz se debe utilizar el ratón y dar click sobre los botones.



a la hora del gameplay las teclas a pulsar aparecerán desde la parte derecha de la pantalla y se desplazarán hasta el final de la pista, el jugador debe pulsar la tecla correspondiente cuando esta esté sobre la ranura.



Por otro lado, las manos ilustran con que tecla se debe pulsar cada tecla y las pistas indican en que linea del teclado se encuentra cada tecla. Para pausar el juego se debe pulsar la tecla "esc" o pulsar el boton de la esquina superior izquierda Los textos en pantalla corresponden a la

puntuación del jugador, el combo, el cual es el número de veces seguidas que el jugador pulsa correctamente una tecla, y el indicador de vidas

Resultados segunda prueba con usuarios

Objetivo:

- Analizar el nivel de conocimiento sobre el tema de los usuarios
- Opinión de los usuarios sobre la estética e interfaz
- Observar cómo los usuarios interactúan con el prototipo
- Descubrir si hay errores o bugs en el prototipo

Apertura

Buenos días/tardes

Gracias por tomarte el tiempo para participar en esta sesión de prueba hoy. Mi nombre es Jessica Murillo soy estudiante de Diseño gráfico.

El propósito de esta sesión es obtener tu opinión y tus comentarios sobre el prototipo que hemos desarrollado. Este prototipo es el concepto de un juego de ritmo basado en la mecanografía

Antes de comenzar, me gustaría aclarar que no hay respuestas correctas o incorrectas en esta prueba. Lo que realmente nos interesa es tu opinión honesta y tus experiencias mientras interactúas con el prototipo. No estás siendo evaluado; es el prototipo el que estamos evaluando.

Durante la sesión, te pediremos que completes algunas tareas específicas utilizando el prototipo y nos des tu opinión sobre tu experiencia. También te animamos a pensar en voz alta y a compartir cualquier pensamiento o pregunta que surja mientras trabajas con él.

¿Tienes alguna pregunta antes de que comencemos?

Deimar Andre valencia Murillo - 11 años

Preguntas de contextualización

¿Sueles ver el teclado al escribir en un computador?

Algunas veces

¿Alguna vez has estudiado mecanografía?

No, lo conozco porque me lo comentaste

¿Has escuchado el término mecanografía?

¿Alguna vez has jugado un videojuego de ritmo? por ejemplo Dance dance revolution, just dance o Guitar hero

si

En caso de que sea afirmativo ¿Qué títulos has jugado?

nada más he jugado guitar hero

¿Utilizas todos tus dedos al escribir en un teclado?

no siempre uso los dedos índices

¿Alguna vez has estudiado mecanografía?

no

Pasamos a la prueba del prototipo

¿Qué te parece la estética del juego?

Es fea, no me gusta

¿Qué opinas sobre la dificultad del juego?

para mí que no se nada de mecanografía medianamente difícil

¿Cuál es tu opinión general de este prototipo?

una chimba

¿Hay algo que mejorarías de este prototipo?

la estética, es fea. me gustaría paisajes retro me gustaría como la gráfica del juego hop tiles 3d hit music game

Para mi muy difícil, el nivel fácil(el experimento 1) me pareció de dificultad media

Observaciones sobre el usuario:

miraba constantemente el teclado e ignoraba las indicaciones de las manos guía pulsando las teclas con cualquier dedo principalmente con el dedo índice.

el usuario no entendió la tecla ESP

se burlaba el tutorial, le parecía muy obvio

Observaciones sobre el prototipo

Cuando se reiniciaba el nivel se generaba una nota fuera de tiempo y en varias ocasiones las teclas perdían sincronía con la música

¿Cuál es tu nombre y tu edad?

Tiana Sofia Valencia Murillo y tengo 8 años

¿Alguna vez has jugado un videojuego de ritmo? por ejemplo Dance dance revolution, just dance o Guitar hero

si

En caso de que sea afirmativo ¿Qué títulos has jugado?

He jugado Just dance

¿Has escuchado el término mecanografía?

no, ¿qué es eso?

¿Alguna vez has estudiado mecanografía?

No

¿Has tenido el interés de aprender mecanografía?

si

¿Sueles ver el teclado al escribir en un computador?

si

¿Utilizas todos tus dedos al escribir en un teclado?

no principalmente uso mis dedos índices

Pasamos a la prueba del prototipo

¿Qué opinas sobre la dificultad del juego?

muy rápido las teclas, es muy dificil para mi

¿Qué te parece la estética del juego?

Buena

¿Cuál es tu opinión general de este prototipo?

Me parece buenísimo

¿Hay algo que mejorarías de este prototipo?

Es muy rápido y es muy difícil

Observaciones del usuario

tuvo problemas en el tutorial, le costó reconocer la tecla de avanzar en el tutorial

no entendió el término de ranura y tuvimos que hacer una explicación más extensa

en la hora del gameplay observaba el teclado y no pulsaba las teclas al ritmo de la música, la pulsaba apenas las veía en la pantalla. Después se frustró al perder varias teclas y comenzó a pulsar de manera aleatoria las teclas del teclado con el fin de acertar

observaciones del prototipo

Al reanudar un nivel después de pasar por la escena pausa, la música no se escucha

¿Cuál es tu nombre y tu edad?

Gabriela Mejia 22 años

¿Alguna vez has jugado un videojuego de ritmo? por ejemplo Dance dance revolution, just dance o Guitar hero

si

En caso de que sea afirmativo ¿Qué títulos has jugado?

just dance

¿Has escuchado el término mecanografía?

si

¿Alguna vez has estudiado mecanografía?

si

¿Cuándo escribes en un computador sueles mirar el teclado?

si

¿Qué dedos sueles utilizar al escribir en un teclado?

utilizas el índice y el corazón

Pasamos a la prueba del prototipo

¿Qué opinas sobre la dificultad del juego?

es difícil

¿del 1 al 10 qué tan divertido te pareció el juego?

7

¿Qué te parece la estética del juego?

la veo muy bien, la tipografía ni me convence

¿Cuál es tu opinión general de este prototipo?

se necesita saber de mecanografía para disfrutarlo al 100 por cien

¿Hay algo que mejorarías de este prototipo?

me gustaría algún sonido cuando pierdo, en el tutorial cualquier tecla sirva para pasar las páginas

¿Cuál es tu nombre y tu edad?

Daniela Posada 20años

¿Alguna vez has jugado un videojuego de ritmo? por ejemplo Dance dance revolution, just dance o Guitar hero

si

En caso de que sea afirmativo ¿Qué títulos has jugado?

just dance

¿Has escuchado el término mecanografía?

si

¿Alguna vez has estudiado mecanografía?

en el colegio

¿Cuándo escribes en un computador sueles mirar el teclado?

no

¿Qué dedos sueles utilizar al escribir en un teclado?

suelo utilizar el índice, el corazón y el pulgar para el espacio

Pasamos a la prueba del prototipo

¿Qué opinas sobre la dificultad del juego?

para mi difícil porque me costaba ubicar las teclas

¿del 1 al 10 qué tan divertido te pareció el juego?

9

¿Qué te parece la estética del juego?

me encanta

¿Cuál es tu opinión general de este prototipo?

me parece muy bueno y me falta la carita del mecanet

¿Hay algo que mejorarías de este prototipo?

en el tutorial especificar todas las teclas para ubicarnos en el teclado

¿Cuál es tu nombre y tu edad?

Valentina Molano 23 años

¿Alguna vez has jugado un videojuego de ritmo? por ejemplo Dance dance revolution, just dance o Guitar hero

si

En caso de que sea afirmativo ¿Qué títulos has jugado?

guitar hero para celular

¿Has escuchado el término mecanografía?

no

¿Alguna vez has estudiado mecanografía?

no

¿Cuándo escribes en un computador sueles mirar el teclado?

si

¿Qué dedos sueles utilizar al escribir en un teclado?

utilizo el índice, el anular, el corazón y de vez en cuando el menique

Pasamos a la prueba del prototipo

¿Qué opinas sobre la dificultad del juego?

para mi muy difícil

¿del 1 al 10 qué tan divertido te pareció el juego?

9 pero es estresante

¿Qué te parece la estética del juego?

me gustó mucho, muy futurista

¿Cuál es tu opinión general de este prototipo?

Está muy elaborado para ser un prototipo

¿Hay algo que mejorarías de este prototipo?

cambiaría la ubicación de las manos y la pondría al mismo nivel de la ranura

¿Cuál es tu nombre y tu edad?

Juan pablo chacón 22 años

¿Alguna vez has jugado un videojuego de ritmo? por ejemplo Dance dance revolution, just dance o Guitar hero

si

En caso de que sea afirmativo ¿Qué títulos has jugado?

piano tiles

¿Has escuchado el término mecanografía?

si

¿Alguna vez has estudiado mecanografía?

no

¿Cuándo escribes en un computador sueles mirar el teclado?

a veces

¿Qué dedos sueles utilizar al escribir en un teclado?

utilizo el índice, el corazón y el anular

Pasamos a la prueba del prototipo

¿Qué opinas sobre la dificultad del juego?

para un desafía. Está bien esta complejo para alguien que no sabe mecanografia

¿del 1 al 10 qué tan divertido te pareció el juego?

10

¿Qué te parece la estética del juego?

me gusto se entiende bastante

¿Cuál es tu opinión general de este prototipo?

está muy bien diseñado las indicaciones se encuentran facil

¿Hay algo que mejorarías de este prototipo?

no

Referencias

AgileFingers. (s. f.). *Vale la pena aprender mecanografia. He aquí por qué.* Recuperado 20 de abril de 2024, de https://agilefingers.com/es/articulos/razones-por-las-que-deberias-aprender-mecanografia

Asale, R.-., & Rae. (s. f.). *mecanografía | Diccionario de la lengua española*. «Diccionario de la Lengua Española» - Edición del Tricentenario. https://dle.rae.es/mecanograf%C3%ADa#AdlYUtW

Cáceres García, C. (s. f.). *Curso MecaNet* (V21.01.01) [Software]. https://www.cursomecanet.com/

Correa, J. D. C. (2015). *Digitopolis II: Creación de videojuegos en GDevelop*. Jose David Cuartas Correa. https://books.google.com.co/books?id=CLaJDAAAQBAJ&printsec=copyright#v=onepage&q&f=false

Free Touch Typing software. (s. f.). TypingClub. https://www.typingclub.com/

Gallego, F., Molina, R., & Llorens, F. (2014, 1 julio). *Gamificar una propuesta docente: Diseñando experiencias positivas de aprendizaje*. XX Jornadas Sobre la Ensenanza Universitaria de la Informatica (JENUI), Oviedo, España. https://rua.ua.es/dspace/bitstream/10045/39195/1/Gamificacio%CC%81n%20(definicio%CC%81n).pdf#:~:text=%22Gamificaci%C3%B3n%20(gamification%20en%20el%20%C3%A1mbito%20anglosaj%C3%B3n)%20es,positivos%20comunes%20a%20todos%20los%20juegos%22%20http://www.gamificacion.com/que%2D%E2%80%90es%2D%E2%80%90la%2D%E2%80%90gamificacion.

GDevelop. (2008). *GDevelop* (V5.3.199) [Software]. https://gdevelop.io/

Heller, E. (2015). *Psicología del color: Cómo actúan los colores sobre los sentimientos y la razón*.

Just1337. (2017). *Slash it 2* [Videojuego]. https://store.steampowered.com/app/583760/Slash_It_2/Videojuego

Karl Kapp. (2024, 2 mayo). *What is Gamification? A Few Ideas (Updated for 2024 Version)* [Vídeo]. YouTube. https://www.youtube.com/watch?v=TjNVd3Z0Ul8

Keyboard practice - a game example from the GDevelop game making app | GDevelop. (s. f.). Recuperado 22 de abril de 2024, de https://gdevelop.io/game-example/free/keyboard-practice

Lacasa, P. (2011). *Los videojuegos: aprender de mundos reales y virtuales*. Ediciones Morata. https://www.antupload.com/filed/moqKVfJv/Los+videojuegos+-+Pilar+Lacasa

Molich, R., & Nielsen, J. (1990). Improving a human-computer dialogue. *Communications Of The ACM*, *33*(3), 338-348. https://doi.org/10.1145/77481.77486

NotITG (v4.2.0). (2021). [Software]. https://www.noti.tg/Videojuego

Open Ai. (2024). *ChatGpt* (GPT-3.5) [Software]. https://chat.openai.com/

Ortíz, A. P. R., Carrera, M. F. O., Ruiz, H. L., López, M. J. M., & García, J. J. B. (2017). El uso de los videojuegos como herramienta didáctica para mejorar la enseñanza-aprendizaje: una revisión del estado del tema. *IngenieríA, InvestigaciÓN y Desarrollo/Ingeniería, Investigación y Desarrollo*, *17*(2), 36-46. https://doi.org/10.19053/1900771x.v17.n2.2017.7184

Patrick_Yu. (2022, 27 julio). *What are Rhythm Games?* Acer Corner. https://blog.acer.com/en/discussion/188/what-are-rhythm-games

ppy. (2007). *Osu!* (b20240411.1) [Software]. https://osu.ppy.sh/Videojuego

Purita, G. (2023, 19 septiembre). Experiencia de Usuario: Qué es y por qué es tan importante. *OBS Business School*. https://www.obsbusiness.school/blog/experiencia-de-usuario-que-es-y-por-que-es-tan-importante

Rhythm - a game example from the GDevelop game making app | GDevelop. (s. f.). https://gdevelop.io/game-example/free/rhythm

Schell, J. (2008). *The Art of Game Design: A book of lenses*. CRC Press.

Wikipedia. (2023, 24 marzo). *Videojuego de ritmo*. Wikipedia, la Enciclopedia Libre. Recuperado 21 de abril de 2024, de https://es.wikipedia.org/wiki/Videojuego_de_ritmo

Yu, P. (2022, 27 julio). *What are Rhythm Games?* Acer Corner. Recuperado 20 de abril de 2024, de https://blog.acer.com/en/discussion/188/what-are-rhythm-games