

Servicio nacional de Aprendizaje SENA

Proyecto: Juego de adivinar número (Spider'sNumber)

ADSO 15

Ficha: 2827703

Competencia: Calidad

Miembros:

Edwin Andrés Palacios castro

Juan Felipe Colmenares Girado

James Daniel Salamanca Romero

Andrés Felipe Mosquera Álvarez

Instructor: Edwin Guerrero

Tabla de contenido

1. FORMATOS	3
1.1. Formato de registro del tiempo	3
1.2. Formato de registro de defectos	3
1.3. Resumen del plan del proyecto	5
2. INTRODUCCIÓN	6
2.1 Propósito	6
2.2 Propuesta de valor o innovación	6
3. DESCRIPCIÓN GENERAL	7
3.1 Perspectiva de Spider'sNumber	7
3.2 Funciones generales del juego Spider'sNumber	7
3.3 Lenguajes de programación y maquetación y tecnologías	8
3.4. Estándares de calidad aplicados (ISO/IEC 25010)	8
4. APLICACIÓN DE COBIT EN SPIDER'SNUMBER	8
4.1. Definición COBIT	8
4.2. APO (Alinear, Planear y Organizar) #04: Gestión de la Innovación	8
4.3. BAI (Construir, Adaptar, Implementar) #02: Identificación y Análisis de Requisitos	9
5. METODOLOGÍA DE TRABAJO ESTABLECIDA	10
5.1. Metodología	10
5.2. Actores	10
6. REQUISITOS ESPECÍFICOS	10
6.1. Requerimientos funcionales	10
6.2. Requerimientos no funcionales	13
7. Mockups y diseños de interfaz del juego Spider'sNumber	15
7.1. Mocups versión escritorio	15

7.2. Mockups versión celulares.....	18
7.3. Guía de estilos y componentes.....	22
7.4. ¿Qué es?	22
7.5. Definiciones.....	23

1. FORMATOS

1.1. Formato de registro del tiempo

Estudiantes: Edwin palacios, Juan Colmenares, James Salamanca, Andrés Mosquera

Instructor: Edwin Guerrero

Fecha: 12/11/2024

Programa: 2827703

FECHA	INICIO	TERMINO	TIEMPO INTERRU PCIÓN	TIEMPO DELTA	FASE	COMENT ARIOS
12/11/2024	1:00PM	4:30PM		4 hrs 30 min	Planeación	
19/11/2024	1:15PM	4:00PM		3 hrs 45 min	Diseño	
26/11/2024	1:00PM	5:00PM		4 hrs	Codificación	

1.2. Formato de registro de defectos

Tipos de defectos

10 = documentación: Defectos relacionados con errores en los documentos del proyecto.

20 = sintaxis: Defectos que ocurren debido a errores en la escritura del código.

30 = construcción: Errores cometidos durante la compilación o el ensamblaje del código.

40 = asignación: Problemas relacionados con la asignación de tareas, variables o recursos.

50 = interfaz: Defectos relacionados con las interfaces de usuario o de sistema

60 = chequeo: Errores en los procesos de validación o pruebas.

70 = datos: Defectos relacionados con la gestión o manipulación de datos.

80 = función: Errores en la funcionalidad específica del sistema.

90 = sistema: Defectos relacionados con el entorno en el que opera el software

100 = ambiente: Errores que surgen debido al entorno físico o digital

Fecha	Número	tipo	encontrado	removido	Tiempo de composición	Defecto arreglado
19/11/2024	1	10	documentación	documentación	30 minutos	-Falta de requerimientos no funcionales en la documentación. involucrados.
19/11/2024	2	10	documentación	documentación	30 minutos	-Ausencia de integración de COBIT en la documentación.
19/11/2024	3	10	documentación	documentación	40 minutos	-No especificación de metodología ágil ni actores / roles involucrados en la documentación.
19/11/2024	4	10	documentación	documentación	10 minutos	-Título inexistente en la documentación.
19/11/2024	5	10	documentación	documentación	10 minutos	-Falta de tabla de contenido en la documentación.

19/11/2024	6	10	documentación	documentación	120 minutos	-No especificación de metodología ágil ni actores / roles
19/11/2024	7	10	documentación	documentación	10 minutos	-Incongruencias con el formato de registro del tiempo.
25/11/2024	8	60	chequeo	chequeo	30 minutos	Falta de permisos de chequeo para el integrante de esta tarea para modificaciones de los Mockups en la plataforma de figma
25/11/2024	9	50	chequeo	chequeo	60 minutos	Se encontró un bug en la interfaz de los usuarios en dispositivos móviles
26/11/2024	10	10	documentación	documentación	30 minutos	-Falta especificación guía de estilos y componentes
26/11/2024	11	10	documentación	documentación	20 minutos	-No explicación del equipo de desarrollo en metodología scrum
26/11/2024	12	10	documentación	documentación	15 minutos	-Ausencia de estándares de calidad ISO 25010 aplicados
26/11/2024	13	10	documentación	documentación	5 minutos	-Incongruencia fases en formato de tiempo.

1.3. Resumen del plan del proyecto

resumen del plan del proyecto	
Estudiantes	Edwin Andrés Palacios Castro James Daniel Salamanca Andrés Felipe Mosquera Juan Felipe Ángel Colmenares
Programa	Juego de adivinar número (Spider'sNumber)
Instructor	Edwin Alexander Guerrero Martínez
Fecha	

Programa #	01		
lenguajes	JavaScript		
Plan: 720 min			
Tiempo en la fase(min):	Actual	A la fecha	A la fecha %
Planeación	270 min	270 min	37,5%
Diseño	225 min	495 min	68,75%
Codificación	300 min	795 min	110,41%
total	795 min	795 min	110,41%

FASE PLANEACIÓN

2. INTRODUCCIÓN

En este documento se tratarán las especificaciones técnicas que se verán dentro del proyecto **Spider'sNumber** contando con la descripción de las funciones de este, los requerimientos, interfaces graficas, planeación del mismo, tecnologías a usarse y lenguajes, metodologías de trabajo, estándares de calidad, marco de trabajo y dominios elegidos para el proyecto así como la planeación de la utilización de los recursos a disponerse en pro de los lineamientos y metas establecidas.

2.1 Propósito

El propósito del sistema a realizar es la fabricación de un software/juego el cual va a generar un numero al azar el cual estará limitado del 1 al 42 para que el usuario lo adivine, este tendrá 5 intentos y se le dará pistas diciendo si es mayor o menor del número elegido por el sistema.

2.2 Propuesta de valor o innovación

El juego tendrá soporte para usuarios discapacitados visualmente. A través TTS (text to speech).

3. DESCRIPCIÓN GENERAL

3.1 Perspectiva de Spider'sNumber

El juego está pensado para correr en plataforma web. Los usuarios podrán participar en un juego de adivinanza simple, con una interfaz amigable donde podrán ingresar sus intentos, recibir pistas en tiempo real.

El diseño será visualmente sencillo pero atractivo, con colores e íconos fáciles de entender para los usuarios de todas las edades. No se requerirá experiencia previa, ya que la mecánica es intuitiva.

3.2 Funciones generales del juego Spider'sNumber

- **Generación de Número Aleatorio:** El sistema genera un número aleatorio entre 1 y 42 que será el número secreto que el usuario debe adivinar.
- **Entrada de Intentos del Usuario:** El jugador ingresa un número para adivinar. El sistema valida que el número esté dentro del rango de 1 a 42.
- **Validación de Intentos:** El sistema verifica si el número ingresado es correcto, mayor o menor que el número secreto.
- **Pistas en Tiempo Real:** El sistema proporciona pistas luego de cada intento, indicando si el número es mayor o menor que el número secreto.
- **Contador de Intentos:** El sistema lleva un registro del número de intentos que el jugador ha realizado (máximo 5 intentos). Una vez alcanzados los 5 intentos, el juego termina.
- **Mensaje de Resultado Final:** Al finalizar los intentos o adivinar el número, el sistema muestra un mensaje con el resultado (si el jugador adivinó el número o no) y el número secreto.
- **Reinicio del Juego:** Permite reiniciar una nueva partida, generando un nuevo número aleatorio y restableciendo el contador de intentos.
- **Lectura de Instrucciones Iniciales:** Al comenzar el juego, el sistema leerá en voz alta las instrucciones básicas del juego, como el objetivo (adivinar el número entre 1 y 42) y los intentos disponibles.
- **Lectura de Intentos y Pistas:** Cuando el jugador ingresa un número, el sistema leerá el resultado de ese intento (por ejemplo, "El número es mayor que el número secreto" o "El número es menor que el número secreto") y actualizará la información sobre los intentos restantes.

- **Mensaje Final (Victoria/Derrota):** Cuando el jugador adivina el número o se quedan sin intentos, el sistema leerá en voz alta si han ganado o perdido el juego, y también el número secreto.

3.3 Lenguajes de programación y maquetación y tecnologías

Los lenguajes establecidos a utilizarse en el desarrollo del programa serán: JavaScript, HTML, CSS. Otra tecnología usada en el desarrollo del juego es la API “Web speech API” integrada en la web y que es compatible en la mayoría de los navegadores.

Web speech API: El lenguaje base de esta API es JavaScript, por lo que toda implementación de esta en un programa se realiza a través de sintaxis o comandos de este lenguaje. Debido a que esta API es una herramienta estándar de la W3C (World Wide Web Consortium) no hace falta instalar ni frameworks ni librerías, ni software externo.

3.4. Estándares de calidad aplicados (ISO/IEC 25010)

Marco con el cual se evalúa qué tanto se cumple con las características de calidad en el mercado en el producto tratado (Juego Spider’sNumber). En este proyecto las características de calidad aplicadas son:

- **Funcionalidad:** Ya que el sistema cuenta con las funciones que se requieren en los requisitos establecidos.
- **Usabilidad:** Porque los usuarios pueden interactuar fácilmente con el juego.
- **Mantenibilidad:** Debido a que el sistema en su codificación y desarrollo hace uso de modularidad.

4. APLICACIÓN DE COBIT EN SPIDER’SNUMBER

4.1. Definición COBIT

COBIT 5 (Control Objectives for Information and Related Technologies, versión 5) es un marco de referencia global diseñado para ayudar a las organizaciones a gobernar y gestionar la tecnología de información (TI) de manera efectiva y eficiente. Fue desarrollado por ISACA (Information Systems Audit and Control Association) y se enfoca en alinear las metas estratégicas de la organización con sus capacidades tecnológicas.

4.2. APO (Alinear, Planear y Organizar) #04: Gestión de la Innovación

Esta metodología incluye seis componentes clave:

- APO04.01:** Crear un entorno propicio para la innovación.
- APO04.02:** Mantener un entendimiento del entorno empresarial.
- APO04.03:** Monitorear y analizar el entorno tecnológico.
- APO04.04:** Evaluar el potencial de tecnologías emergentes e ideas innovadoras.
- APO04.05:** Recomendar iniciativas adicionales apropiadas.
- APO04.06:** Monitorear la implementación y el uso de la innovación.

La relevancia de esta metodología en nuestro proyecto radica en que establece criterios para integrar tecnologías emergentes y fomenta una planificación estratégica basada en innovación. Un ejemplo claro es la incorporación del **Text to Speech (TTS)**, que proporciona soporte a usuarios con discapacidades visuales, haciendo del proyecto una solución accesible, moderna e inclusiva.

4.3. BAI (Construir, Adaptar, Implementar) #02: Identificación y Análisis de Requisitos

Esta metodología comprende cuatro componentes principales:

- BAI02.01:** Definir y mantener requisitos funcionales y técnicos.
- BAI02.02:** Realizar un estudio de viabilidad y formular soluciones alternativas.
- BAI02.03:** Gestionar riesgos relacionados con los requisitos.
- BAI02.04:** Obtener la aprobación de requisitos y soluciones.

La importancia de esta metodología para nuestro proyecto radica en que proporciona directrices claras para alinear el juego con los objetivos estratégicos y las necesidades de los usuarios desde las etapas iniciales. Esto incluye la definición de requisitos específicos, como las pistas en tiempo real, el contador de intentos y el soporte TTS, asegurando que el producto final sea funcional y cumpla con las expectativas tanto de los usuarios como de los **stakeholders**.

El uso de esta metodología minimiza riesgos e incompatibilidades durante el desarrollo, promoviendo un enfoque estructurado y colaborativo. Además, fomenta un desarrollo consensuado con todas las partes interesadas, garantizando que las prioridades de todos sean consideradas y evitando cambios inesperados que puedan afectar el progreso del proyecto.

5. METODOLOGÍA DE TRABAJO ESTABLECIDA

5.1. Metodología

La metodología elegida para llevar a cabo la realización del proyecto, juego Spider'sNumber es la Scrum.

5.2. Actores

Los actores que toman relevancia en la planificación, diseño y desarrollo del juego son:

- **Product owner:** es el encargado de dar valor al producto, asignando las tareas de cada uno de los actores, la funcionalidad del sistema y el cómo se debe construir el sistema
- **Scrum master:** es el encargado de facilitar y administrar las tareas del equipo de desarrollo
- **Equipo de desarrollo:**
 - Responsables de construir el producto:
 - **Desarrollador Backend:** es el encargado de la lógica, la construcción y el mantenimiento del servidor web
 - **Desarrollador Frontend:** transforma el trabajo del diseñador en interfaces con las que el usuario puede interactuar
 - **Desarrollador móvil:** adapta el programa para los dispositivos móviles
 - **Diseñador:** Idea y hace bocetos de cómo se debería ver la aplicación según los requerimientos y los estándares que se deben manejar
- **Stakeholders:** persona, grupo u organización que tiene interés o puede verse afectado por un proyecto, producto o decisión.

6. REQUISITOS ESPECÍFICOS

6.1. Requerimientos funcionales

Código	Nombre	Fecha	Grado Necesidad
RF001	Generación de número aleatorio	12/11/2024	Alta
Descripción	El sistema generará un número aleatorio a adivinar, el cual permanecerá oculto hasta el momento en que el jugador escoja el número correcto.		
Proceso	El sistema genera un número aleatorio, que será oculto por el mensaje "Adivina el número".		

Aprobado			Fecha aprobación	
-----------------	--	--	-----------------------------	--

Código	Nombre	Fecha	Grado Necesidad
RF002	Entrada de intentos del usuario	12/11/2024	Alta
Descripción	El sistema deberá validar que el número ingresado esté dentro del rango permitido (1-42)		
Proceso	El sistema recoge el número ingresado por el usuario y valida las condiciones necesarias.		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF003	Validación de intentos	12/11/2024	Alta
Descripción	El sistema debe comprobar si el número ingresado es correcto, mayor o menor que el número secreto		
Proceso	el sistema comprueba el número y lo compara con el número secreto		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF004	Pistas a tiempo real	19/11/2024	Alta
Descripción	El sistema proporcionara pistas después de cada intento indicándole al jugador si el número es mayor o menor que el secreto.		
Proceso	el sistema comprueba el número y lo compara con el número secreto		
Aprobado		Fecha aprobación	

Código	Nombre	Fecha	Grado Necesidad
RF005	Contador de intentos	19/11/2024	Alta
Descripción	El sistema contará los intentos hechos por el jugador tratando de adivinar el número		

Proceso	El sistema contara los intentos hechos por el jugador		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF06	Limitación de intentos	19/11/2024	Alta
Descripción	El sistema limitará al jugador a solo realizar 5 intentos		
Proceso	El sistema solo dejara realizar al jugador 5 intentos para que acabe el juego		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF07	Mensaje de Resultado Final	12/11/2024	Alta
Descripción	Al finalizar el juego el sistema deberá indicarle al jugador si Ganó o perdió el juego, posterior a esto el sistema revelara el número oculto		
Proceso	El jugador adivina correctamente el numero o se le acaban los intentos, El sistema genera un mensaje indicándole al jugador si gano o perdió, el sistema revela el numero oculto		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF08	Reinicio del Juego	12/11/2024	Alta
Descripción	El sistema debe permitir que el jugador reinicie el juego con un botón, restableciendo intentos y generando un nuevo número oculto		
Proceso	El jugador Acciona un botón que tenga la función de reiniciar el contador de intentos, y generar un nuevo número oculto		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF09	Lectura de pistas (TTS)	12/11/2024	Alta
Descripción	El sistema deberá de reproducir un audio de lo que el mensaje de pista dice cada vez que el jugador falle.		
Proceso	El jugador selecciona un número, este siendo incorrecto, en pantalla aparecerá el mensaje de pista y una voz del sistema lo leerá.		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF010	Lectura de resultado final (TTS)	12/11/2024	Alta
Descripción	Cuando el juego termine (ya sea por adivinación o por agotarse los intentos), el sistema debe leer en voz alta el resultado final, informando si el jugador ha ganado o perdido y cuál fue el número secreto.		
Proceso	El jugador adivina el número secreto y junto con el mensaje de ganador y el número secreto descubierto, el sistema indica con una voz el mismo mensaje y el número. Curso alterno: El jugador agota sus intentos incorrectamente, y el sistema junto con el mensaje de pérdida y el número descubierto indica con una voz este mensaje y el número secreto.		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RF011	Lectura de números en pantalla (TTS)	12/11/2024	Alta
Descripción	El sistema deberá, cada vez que el jugador ponga el cursor encima de uno de los números en pantalla, reproducir un audio que indique ese número.		
Proceso	El jugador pone el cursor encima de un número y el sistema reproduce un audio indicando este número gracias a una función.		
Aprobado			Fecha aprobación

6.2. Requerimientos no funcionales

Código	Nombre	Fecha	Grado Necesidad
RNF001	Usabilidad	12/11/2024	Alta
Descripción	El juego deberá ser fácil de interactuar para jugadores de todas las edades, con una interfaz simple y clara. Habrá retroalimentación visual y auditiva para guiar a los usuarios en cada paso del juego. La accesibilidad será fundamental, esto para usuarios con discapacidades visuales. El TTS debe ser fácil de activar y desactivar, y debe ser claro y comprensible.		
Aprobado			Fecha aprobación

Código	Nombre	Fecha	Grado Necesidad
RNF002	Accesibilidad	12/11/2024	Alta

Descripción	El juego debe ser accesible para personas con discapacidad visual, integrando soporte TTS para que los usuarios puedan escuchar las instrucciones, intentos y pistas en lugar de depender de la visualización en pantalla.			
Aprobado			Fecha aprobación	

Código	Nombre		Fecha	Grado Necesidad
RNF003	Compatibilidad web		12/11/2024	Alta
Descripción	El juego/ sitio web deberá estar disponible y funcionar correctamente en la web, sin inconvenientes siempre y cuando se cumpla la ejecución en condiciones aptas.			
Aprobado			Fecha aprobación	

Código	Nombre		Fecha	Grado Necesidad
RNF004	Rendimiento		12/11/2024	Alta
Descripción	El sistema deberá disponer de tiempos de respuesta rápidos, como en la validación de intentos y la entrega de pistas. El tiempo de carga de la página debe ser mínimo.			
Aprobado			Fecha aprobación	

Código	Nombre		Fecha	Grado Necesidad
RNF005	Fiabilidad		12/11/2024	Alta
Descripción	El sistema debe ser robusto y fiable, asegurando que no haya caídas del servidor y que el juego funcione correctamente durante toda la experiencia de juego.			
	La infraestructura en la nube debe tener redundancia para prevenir caídas inesperadas.			
Aprobado			Fecha aprobación	

Código	Nombre		Fecha	Grado Necesidad
RNF006	Mantenimiento y Actualizaciones		12/11/2024	Alta
Descripción	El sistema debe ser fácil de mantener y actualizar, con un código bien estructurado y modular.			
Aprobado			Fecha aprobación	

Código	Nombre	Fecha	Grado Necesidad
RNF007	Compatibilidad con TTS	12/11/2024	Alta
Descripción	El sistema deberá de permitir que la tecnología integrada TTS en el juego sea compatible y funcione en múltiples navegadores, Chrome, Edge, Firefox, Brave etc.		
Aprobado			Fecha aprobación

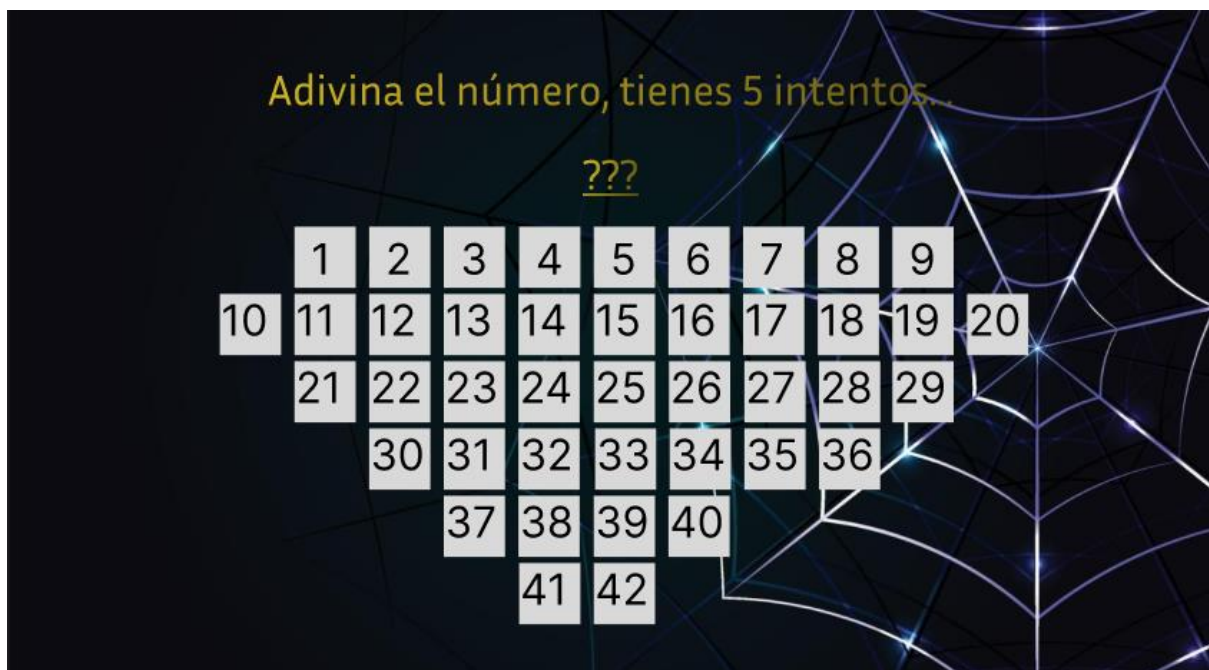
FASE DISEÑO

7. Mockups y diseños de interfaz del juego Spider'sNumber

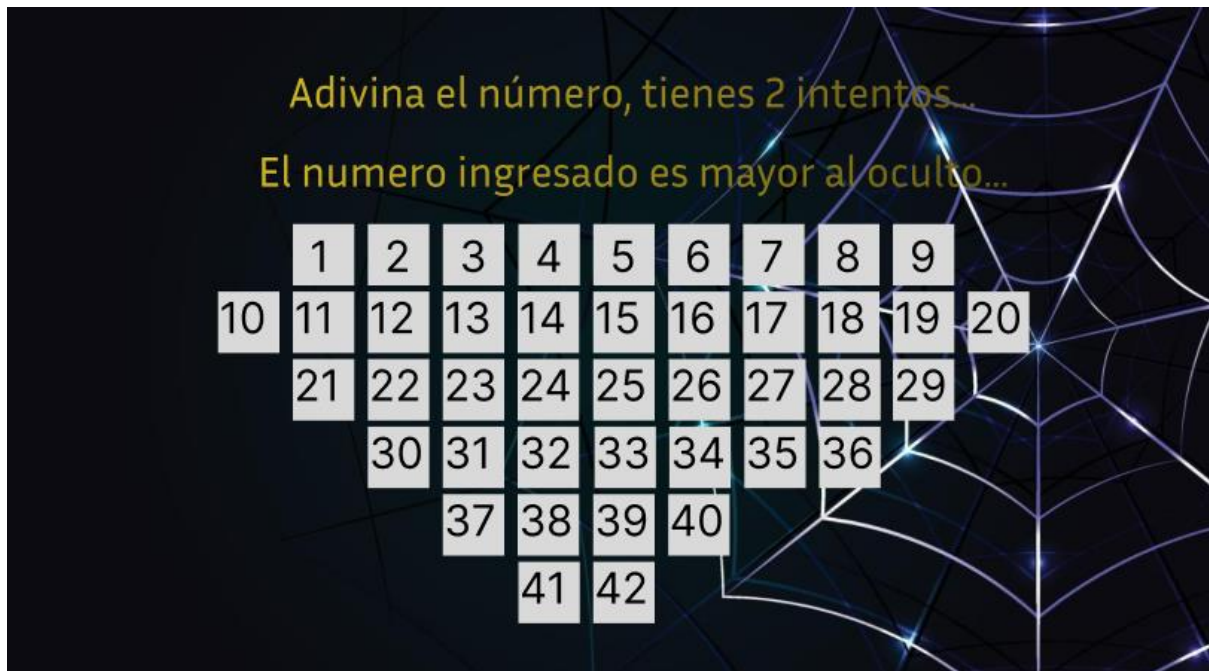
En este proyecto / juego “Spider'sNumber” se representaron los mockups siendo estos de mediana fidelidad, tratando de que quede lo más similar posible a el juego real y cómo se verá este una vez ejecutado, ya sea en computadora o en móvil.

7.1. Mocups versión escritorio

- Generación de número aleatorio



- Contador de intentos



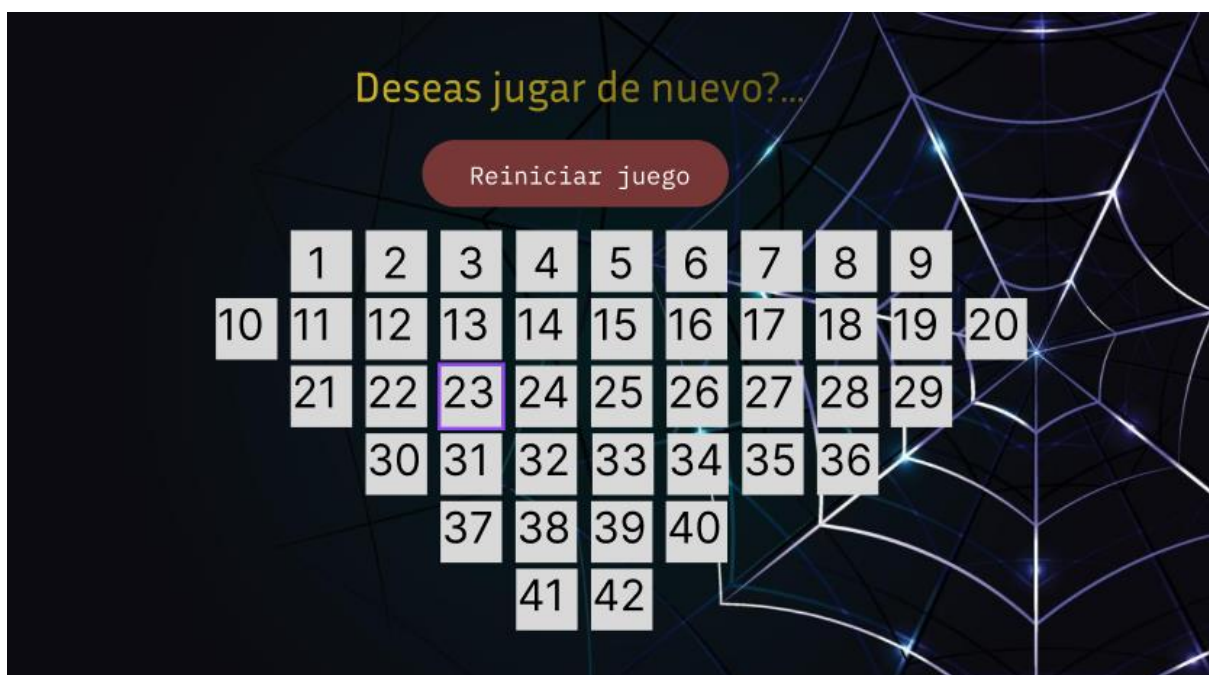
- Mensaje resultado final



- Mensaje resultado final alternativo

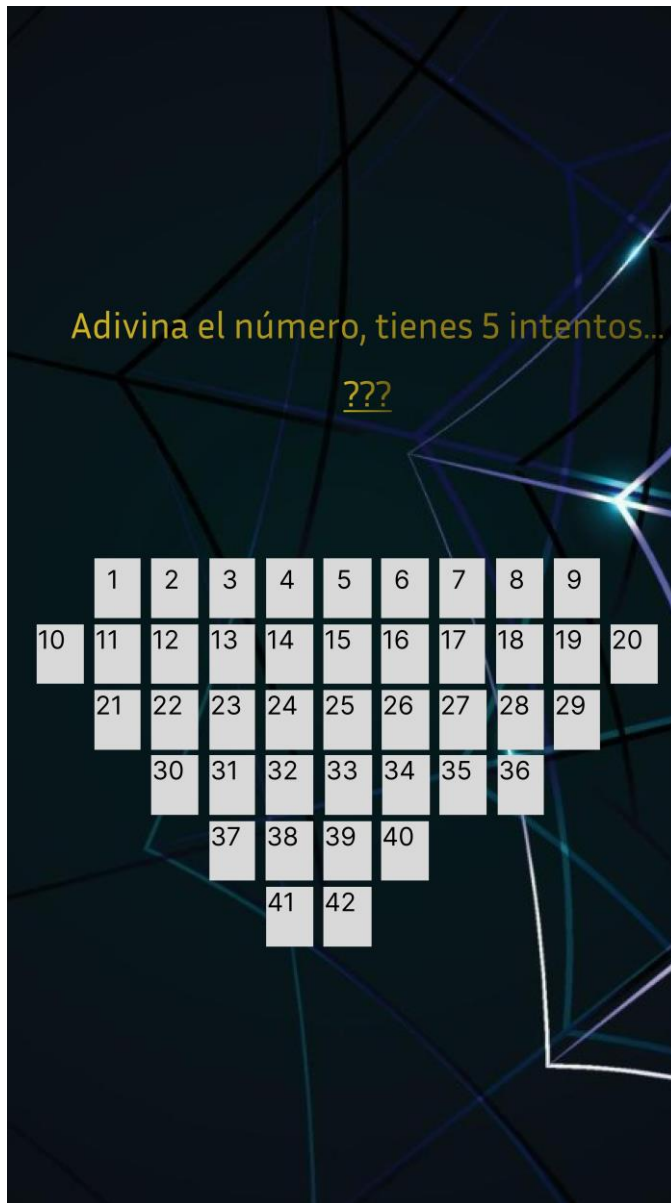


- Reinicio del juego

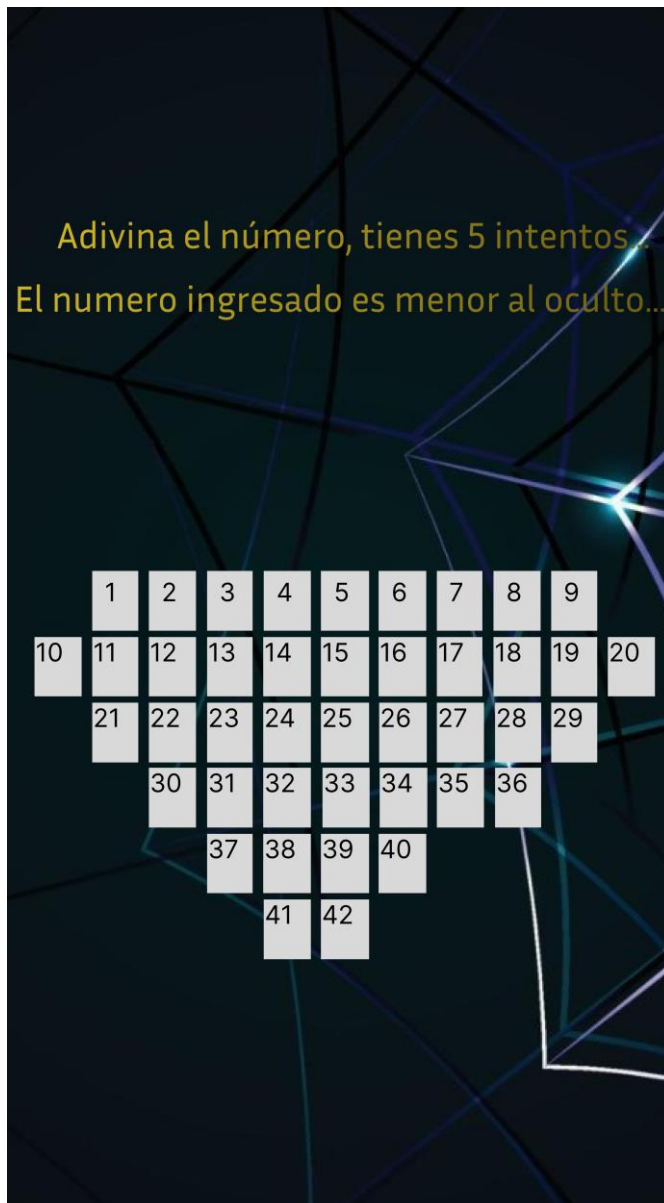


7.2. Mockups versión celulares

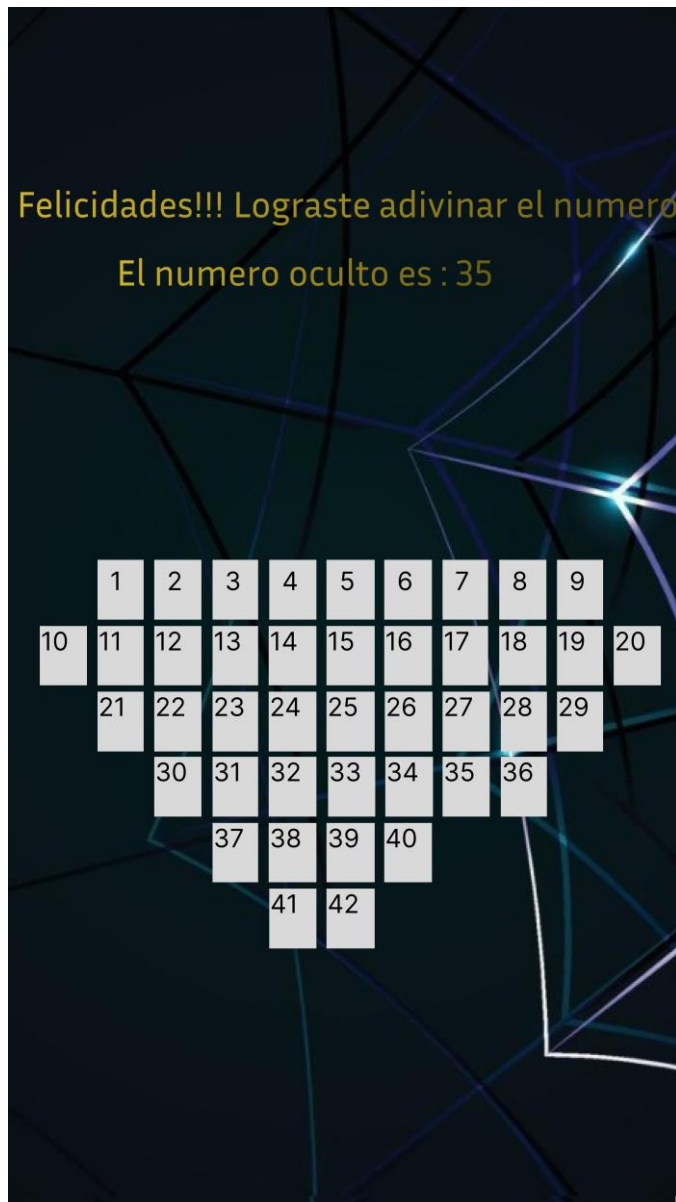
- Generación de número aleatorio



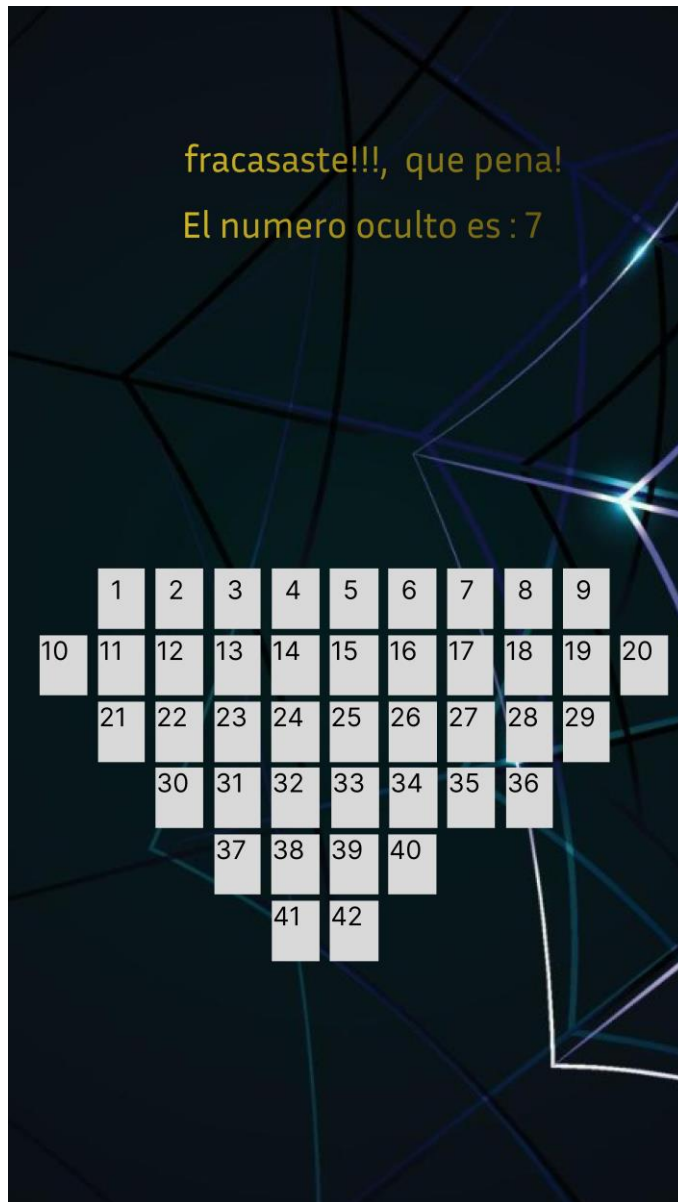
- **Contador de intentos**



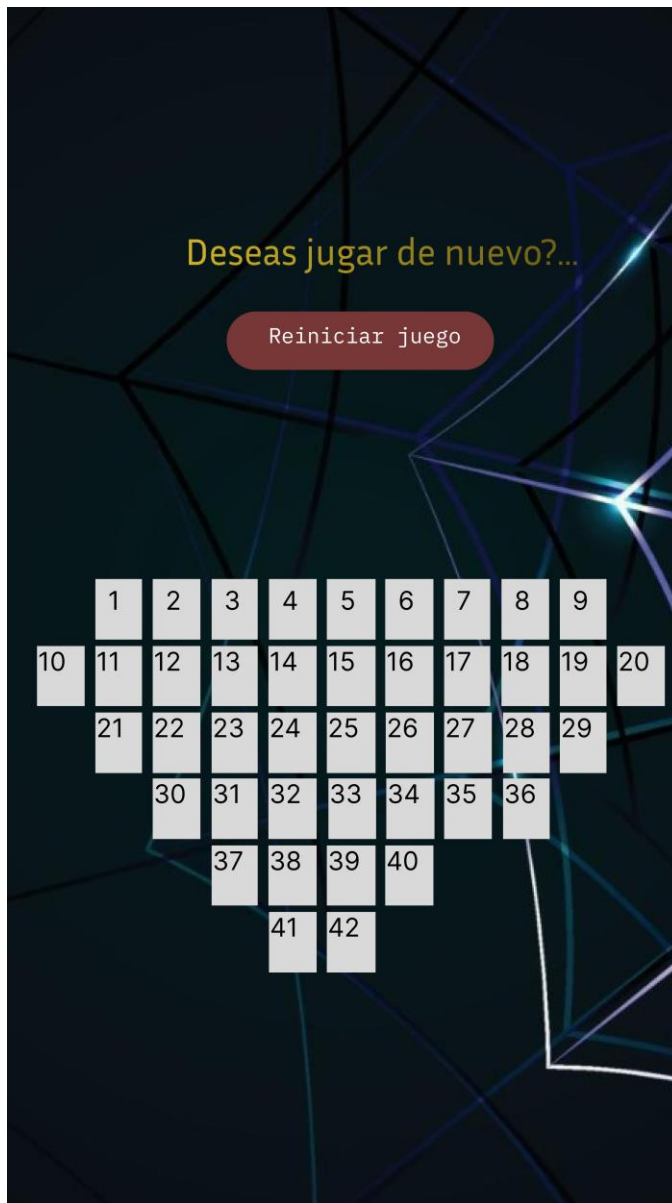
- Mensaje resultado final



- **Mensaje resultado final alternativo**



- **Reinicio del juego**



7.3. Guía de estilos y componentes

7.4. ¿Qué es?

La guía de estilos y componentes es un formato en el cual se plasman, como su nombre lo dice, todos los elementos en términos de diseño que conforman o hacen parte de la interfaz del producto. Se resumen qué fuente de texto se usará, qué tamaños de texto, botones a implementarse y sus propiedades (color, dimensiones etc). Logos, paleta de colores, fondos, imágenes predeterminadas.

7.5. Definiciones

- **Favicon:** es una versión pequeña del logo que aparece en la pestaña del navegador junto al nombre de la pestaña.
- **Background:** es la imagen o color que se encuentra al fondo de la página y que es visible la mayor parte del tiempo.
- **Color texto:** es un texto de ejemplo que contiene la fuente y el color que llevara el texto en el juego
- **Times New Roman:** el tipo de fuente utilizada
- **Times, serif:** el tipo de fuente caracterizada por líneas unidas a las letras
- **PX:** es la abreviatura de la palabra píxel
- **Píxel:** es un pequeño cuadrado de los cuales está compuesta una imagen
- **Tamaño mínimo y máximo de fuentes (respectivamente):** 35px ,850px
- **Botón:** es un objeto con el que el usuario puede interactuar
- **Botón número:** es un botón que al presionarlo introduce un número
- **Botón número – deshabilitado:** es la forma en que se verá el botón para darle a entender al jugador que no puede interactuar con él en ese momento
- **Tamaño de los botones en versión escritorio y versión móvil (respectivamente):** 65x65px ,30x30px

Logo - Favicon



Background
Color texto

Textos

Texto1 - Times New Roman, Times, serif -
40 px - Max-width > 850px

Texto2 - Times New Roman, Times, serif -
35 px - Max-width > 850px

Texto1 - Times New Roman, Times, serif -
30 px - Max-width < 850px - Max-width > 510

Texto2 - Times New Roman, Times, serif -
20 px - Max-width < 850px - Max-width > 510

Texto1 - Times New Roman, Times, serif -
25 px - Max-width < 510

Texto2 - Times New Roman, Times, serif -
20 px - Max-width < 510

Botones

10

Botón número -
escritorio (65x65)

10

Botón número -
deshabilitado
escritorio (65x65)

30

Botón número -
móvil (40x40 -
30x30)

30

Botón número -
deshabilitado
móvil (40x40 -
30x30)

Reiniciar juego

Botón reiniciar juego