

Carpeta de Campo- INDIVIDUAL

Integrantes: Felipe Charabora- Leonel Gualini

¿De qué se trata el Proyecto?

Este proyecto tiene como objetivo principal la creación de un juego de dados interactivo mediante el uso de la plataforma Arduino. En lugar de depender exclusivamente de la fortuna al lanzar un dado convencional, explicaremos cómo programar un dado virtual que se ejecutará en un entorno electrónico controlado. La esencia del proyecto radica en fusionar los principios de la programación en Arduino con la emoción y la incertidumbre asociadas con los juegos de azar.

Explicación Tinkercad (Felipe Charabora)

Introducción a Tinkercad:

Tinkercad es una herramienta en línea que facilita la creación de prototipos virtuales en 3D y es especialmente útil para diseñar y simular circuitos electrónicos. En este proyecto, utilizaremos Tinkercad para visualizar y probar nuestro diseño de juego de dados antes de implementarlo en hardware real. Esta fase de prototipado virtual nos permitirá identificar posibles problemas y

perfeccionar nuestra idea antes de pasar a la fase de construcción física.

Creación del Circuito:

En Tinkercad, comenzaremos diseñando el circuito que respalda nuestro juego de dados. En lugar de utilizar LEDs, incorporaremos un display de siete segmentos y un botón virtual para simular la interfaz de usuario. Configuraremos la conexión entre estos componentes para replicar el comportamiento que queremos lograr en el mundo real.

Programación en Tinkercad:

Tinkercad no solo es una herramienta de diseño visual, sino que también permite la programación de Arduino directamente en la plataforma. Aquí, escribiremos y probaremos nuestro código en un entorno virtual antes de cargarlo en la placa Arduino física. Este enfoque nos proporciona una visión previa del funcionamiento del programa y nos ayuda a depurar posibles errores antes de la implementación real.

Simulación del Juego:

Una de las ventajas clave de Tinkercad es su capacidad para simular el funcionamiento de un circuito. En esta sección, explicaremos cómo simular el juego de dados,. Esto nos permitirá ajustar y refinar nuestro diseño antes de pasar a la fase de construcción física.

Evaluación de Rendimiento:

Tinkercad nos proporcionará datos valiosos sobre el rendimiento de nuestro diseño. Analizaremos aspectos como el tiempo de respuesta, la secuencia de luces y cualquier comportamiento inesperado. Esta evaluación nos ayudará a perfeccionar tanto el diseño del circuito como el código antes de llevar nuestro juego de dados al mundo real.

Selección de Componentes Virtuales Específicos:

Exploramos la librería de Tinkercad y seleccionamos un display de siete segmentos y un botón para nuestro circuito virtual. Estos elementos serán esenciales para recrear la experiencia táctil de lanzar un dado real.

Diseño del Circuito con Display y Botón:

Utilizando la interfaz de diseño de circuitos de Tinkercad, conectaremos el display y el botón de manera lógica. Estableceremos la relación entre el botón y la activación del display para simular la acción de lanzar el dado de forma virtual.

Configuración de la Interacción Usuario-Circuito:

Nos aseguraremos de que el botón sea la entrada que desencadena la generación de un número aleatorio en el display de siete segmentos. Este diseño nos permitirá simular el acto de lanzar un dado presionando el botón, ofreciendo una representación más realista en nuestro entorno virtual de Tinkercad.

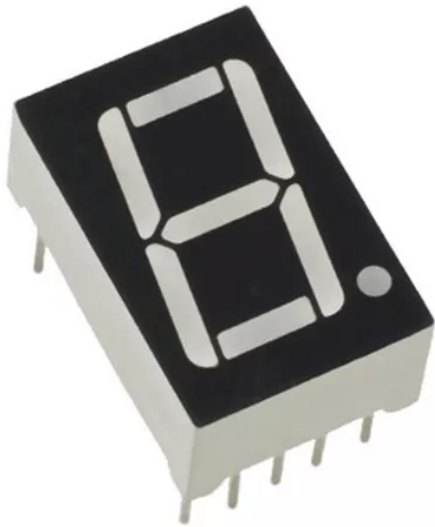
Placa Arduino:

La pieza central de este proyecto es la placa Arduino, una plataforma de desarrollo de código abierto que actúa como el cerebro de nuestro juego de dados. La placa proporciona la interfaz para cargar y ejecutar el código que controla el comportamiento del juego. En este caso, utilizamos la placa Arduino para gestionar el display de siete segmentos y la interacción con el botón.



Display de Siete Segmentos:

El display de siete segmentos es nuestro medio visual para representar los números generados por el juego de dados. Cada segmento (A a G) iluminará de manera específica para visualizar los números del 1 al 6. Este componente añade una dimensión visual al proyecto, convirtiendo los resultados numéricos en una experiencia más palpable.



Botón:

El botón es el elemento de entrada que desencadena el lanzamiento del dado. Al presionar el botón, se genera un número aleatorio y se muestra en el display de siete segmentos. Este componente añade la interactividad al juego, permitiendo al usuario iniciar cada ronda.



ALGUNAS FUNCIONES DE LO QUE PODRÍAN SER LAS FUNCIONES DEL BOTÓN:

Activación del Juego:

El botón sirve como el disparador principal para activar el juego de dados. Cuando el jugador presiona el botón, se inicia el proceso de generación de un número aleatorio y la representación visual en el display de siete segmentos. Esta acción simula el acto de lanzar un dado convencional, agregando una capa táctil y participativa al juego.

Control de la Frecuencia de Juego:

La función del botón no solo se limita a iniciar el juego, sino que también controla la frecuencia de los lanzamientos. El jugador tiene el poder de decidir cuándo lanzar el dado al presionar el botón. Esta característica agrega un elemento estratégico y de participación activa al juego, permitiendo que cada lanzamiento sea una elección consciente.

Implementación de

Eventos Especiales:

A través de la programación en Arduino, el botón puede ser utilizado para desencadenar eventos especiales en el juego. Por ejemplo, se podría programar para que un largo mantenimiento del botón active una función especial o reinicie el juego.

Esta flexibilidad en la programación permite ampliar las posibilidades del juego y añadir capas adicionales de emoción.

Interacción Continua con el Jugador:

El botón mantiene una interacción continua con el jugador, proporcionándole un papel activo en cada ronda. En lugar de depender únicamente de la aleatoriedad, el jugador decide cuándo lanzar el dado, creando una experiencia más participativa. Además, el botón actúa como un punto de contacto tangible, conectando al jugador directamente con la dinámica del juego de dados Arduino.

Conexiones:

Las conexiones entre los componentes son cruciales para el funcionamiento del juego. A través de cables, conectamos la placa Arduino con el display de siete segmentos y el botón. La disposición y la configuración de estas conexiones son esenciales para garantizar una comunicación efectiva y un comportamiento coherente del juego.

Propósito de las Conexiones en el Proyecto:

Descripción del propósito central de las conexiones en el juego de dados Arduino. Se enfoca en cómo las conexiones establecen la comunicación entre los componentes para lograr la funcionalidad deseada.

Configuración de Conexiones en el Circuito:

Detalles sobre cómo se configuraron y establecieron las conexiones en el circuito. Se abordan las interconexiones

específicas entre la placa Arduino, el display de siete segmentos, el botón, las resistencias y otros componentes clave.

Pruebas y Verificación de Conexiones:

Descripción de cómo se llevaron a cabo pruebas específicas para verificar la integridad y la eficacia de las conexiones. Se destacan los procesos de identificación y corrección de posibles problemas en esta fase.

Importancia de las Conexiones en el Funcionamiento:

Enfoque en la importancia crucial de las conexiones para el funcionamiento del juego. Se destaca cómo las conexiones sólidas y adecuadas son fundamentales para garantizar la transmisión eficiente de señales eléctricas y la respuesta correcta del circuito.

Mini Placa de Pruebas:

La mini placa de pruebas es un espacio donde realizamos conexiones temporales para probar y depurar nuestro circuito antes de implementarlo de manera definitiva. Es un entorno seguro para experimentar y ajustar las conexiones sin afectar la integridad de los componentes principales. Esta fase de prueba en la mini placa de pruebas contribuye a un proceso de desarrollo más suave y libre de errores.



Propósito de la Mini Placa de Pruebas:

Explicación del propósito y función de la mini placa de pruebas en el proyecto. Enfocada en su papel como espacio de experimentación y ajuste antes de la implementación definitiva.

Configuración de Componentes en la Miniplaca:

Detalles sobre cómo se dispusieron y conectaron los componentes en la mini placa de pruebas. Se aborda la disposición física de la placa Arduino, el display de siete segmentos, el botón y las resistencias para realizar pruebas y ajustes.

Experimentación y Verificación:

Descripción de cómo se realizaron pruebas en la mini placa de pruebas para verificar la funcionalidad y la interacción de los componentes. Se destacan los procesos de ajuste y corrección durante esta fase.

Importancia de la Miniplaca en la Fase de Desarrollo:

Enfoque en la importancia de la mini placa de pruebas como parte integral del desarrollo del proyecto. Se destaca cómo contribuyó a una implementación más suave y a la identificación de posibles problemas antes de pasar a la fase de construcción física.

Al explorar cada uno de estos elementos, comprendemos cómo se combinan para crear un juego de dados interactivo impulsado por

Arduino. Desde la lógica del código hasta la visualización en el display, cada componente desempeña un papel clave en la experiencia general del juego.

Tinkercad: Simulación Virtual del Circuito:

Antes de llevar nuestro proyecto a la realidad física, utilizaremos Tinkercad para simular y perfeccionar el circuito virtualmente. Esta plataforma nos permite diseñar, conectar y probar nuestro juego de dados de manera interactiva en un entorno virtual.

Aprovecharemos Tinkercad para asegurarnos de que todas las conexiones estén correctamente establecidas y de que el comportamiento del juego sea coherente con nuestro código.

Diseño del Circuito en Tinkercad:

En Tinkercad, arrastraremos y conectaremos visualmente los componentes que hemos utilizado en nuestro proyecto: la placa Arduino, el display de siete segmentos y el botón. Configuraremos las conexiones de acuerdo con nuestro diseño físico para reflejar fielmente el circuito real.

Simulación de la Interacción del Usuario:

Utilizaremos la herramienta de simulación de Tinkercad para observar cómo interactúa el usuario con el juego de dados virtual. Comprobaremos que al presionar el botón, se genere un número aleatorio y que este número se represente correctamente en el

display de siete segmentos. La simulación nos permitirá identificar posibles problemas antes de pasar a la implementación física.

Ajustes y Optimización en Tinkercad:

Durante la simulación en Tinkercad, realizaremos ajustes en el diseño y la lógica del circuito según sea necesario. Este proceso de optimización nos ayudará a garantizar un funcionamiento fluido y eficiente antes de pasar a la fase de construcción real.

Exportación del Código para Arduino desde Tinkercad:

Una vez satisfechos con la simulación en Tinkercad, exportaremos el código generado por la plataforma para Arduino. Este código servirá como base para nuestra implementación física, asegurando una transición sin problemas del entorno virtual al mundo real.

Documentación de la Simulación en Tinkercad:

Para mantener un registro claro de nuestra simulación y los ajustes realizados, documentamos el proceso en Tinkercad. Describiremos las configuraciones, conexiones y cualquier cambio significativo para tener una referencia clara cuando construyamos el circuito físico.

Estos pasos en Tinkercad nos permitirán afinar y validar nuestro diseño virtualmente antes de comprometernos con la construcción física, garantizando un proceso de desarrollo más eficiente y libre de errores.

RESISTENCIAS



Introducción a las Resistencias en el Proyecto:

Descripción del papel de las resistencias en el juego de dados Arduino. Introducción a su función en la limitación de corriente y estabilización del circuito.

Tipo y Valor de las Resistencias:

Detalles sobre el tipo específico de resistencias utilizadas en el proyecto y sus valores. Explicación de cómo estos valores se seleccionaron para adaptarse al circuito.

Conexión de Resistencias en el Circuito:

Explicación detallada sobre cómo se conectan las resistencias en el circuito. Se aborda su disposición física y su conexión específica a la placa Arduino, el display de siete segmentos y otros componentes.

Función de las Resistencias en el Display de Siete Segmentos:

Desglose de cómo las resistencias contribuyen a la correcta operación del display de siete segmentos. Se explica su papel en la limitación de corriente y la protección de los LEDs.

Protección y Estabilidad del Circuito:

Enfoque en cómo las resistencias contribuyen a la estabilidad del circuito. Se describe cómo protegen los componentes de corrientes excesivas y contribuyen a mantener un entorno eléctrico seguro.

Ajustes y Optimización de Resistencias:

Descripción de cualquier ajuste o optimización realizado en las resistencias durante la fase de diseño y simulación. Se destaca su papel en la optimización general del proyecto.

A continuación.

Explicacion del codigo (Leonel Gualini)

Definición de Pines:

Se establecen constantes para representar los pines conectados al display de siete segmentos (A-G) y al botón (bot).

Cada letra (A-G) representa un segmento del display.

```
1 # define A 2
2 # define B 3
3 # define C 4
4 # define D 5
5 # define E 6
6 # define F 7
7 # define G 8
8
```

Funciones para Configuración de Segmentos:

Se crean funciones específicas (uno(), dos(), ..., seis()) para definir las secuencias de encendido y apagado de los segmentos del display, personalizando la configuración de cada número.

```
11
12 int boton, numero;
13
14 void uno ()
15 {
16     digitalWrite (A, LOW);
17     digitalWrite (B, HIGH);
18     digitalWrite (C, HIGH);
19     digitalWrite (D, LOW);
20     digitalWrite (E, LOW);
21     digitalWrite (F, LOW);
22     digitalWrite (G, LOW);
23     delay(500);
24 }
25
26 void dos ()
27 {
28     digitalWrite (A, HIGH);
29     digitalWrite (B, HIGH);
30     digitalWrite (C, LOW);
31     digitalWrite (D, HIGH);
32     digitalWrite (E, HIGH);
33     digitalWrite (F, LOW);
34     digitalWrite (G, HIGH);
35     delay(500);
36
37
38 }
39
40 void tres ()
41 {
42     digitalWrite (A, HIGH);
43     digitalWrite (B, HIGH);
44     digitalWrite (C, HIGH);
45     digitalWrite (D, HIGH);
46     digitalWrite (E, LOW);
47     digitalWrite (F, LOW);
48     digitalWrite (G, HIGH);
49     delay(500);
50 }
51
52 void cuatro ()
53 {
54     digitalWrite (A, LOW);
55     digitalWrite (B, HIGH);
56     digitalWrite (C, HIGH);
57     digitalWrite (D, LOW);
58     digitalWrite (E, LOW);
59     digitalWrite (F, HIGH);
60     digitalWrite (G, HIGH);
```


Configuración de Pines como Salida:

En el bloque `setup()`, se establece que los pines definidos previamente (A-G) se utilizan como salidas mediante `pinMode()`. Esta configuración es esencial para controlar la visualización en el display.

Inicialización del Generador de Números Aleatorios:

`randomSeed(analogRead(A0))` se encarga de inicializar el generador de números aleatorios de Arduino utilizando el valor analógico de A0 como semilla, asegurando aleatoriedad en los números generados.

```
89 void setup()  
90 {  
91   pinMode(A, OUTPUT);  
92   pinMode(B, OUTPUT);  
93   pinMode(C, OUTPUT);  
94   pinMode(D, OUTPUT);  
95   pinMode(E, OUTPUT);  
96   pinMode(F, OUTPUT);  
97   pinMode(G, OUTPUT);  
98  
99   randomSeed(analogRead(A0));  
100 }
```

Bucle Principal (loop()):

Se inicia el bucle principal con la lectura del estado del botón (`botón = digitalRead(bot)`). Si el botón está presionado (`botón ==`

HIGH), se genera un número aleatorio entre 1 y 6 (inclusivo/exclusivo). Dependiendo del número generado, se llama a la función correspondiente para visualizar el resultado en el display.

```
102 void loop()
103 {
104     boton = digitalRead(bot);
105
106     if(boton == HIGH)
107     {
108         numero = random (1, 7);
109
110         if (numero == 1)
111         {
112             uno();
113         }
114         else if (numero == 2)
115         {
116             dos();
117         }
118         else if (numero == 3)
119         {
120             tres();
121         }
122         else if (numero == 4)
123         {
124             cuatro();
125         }
126         else if (numero == 5)
127         {
128             cinco();
129         }
130         else if (numero == 6)
131         {
132             seis();
133         }
134     }
135     delay(50);
136
137 }
138
```

Retardo y Espera:

Se incorpora un pequeño retardo (delay(50)) al final del bucle principal para evitar lecturas excesivas y suavizar la interacción.

Este retraso también afecta la duración de la visualización de cada número en el display.

Función uno():

Configuración específica de encendido y apagado de los segmentos del display para representar el número 1.

Función dos():

Configuración específica de encendido y apagado de los segmentos del display para representar el número 2.

Función tres():

Configuración específica de encendido y apagado de los segmentos del display para representar el número 3.

Función cuatro():

Configuración específica de encendido y apagado de los segmentos del display para representar el número 4.

Función cinco():

Configuración específica de encendido y apagado de los segmentos del display para representar el número 5.

Función seis():

Configuración específica de encendido y apagado de los segmentos del display para representar el número 6.

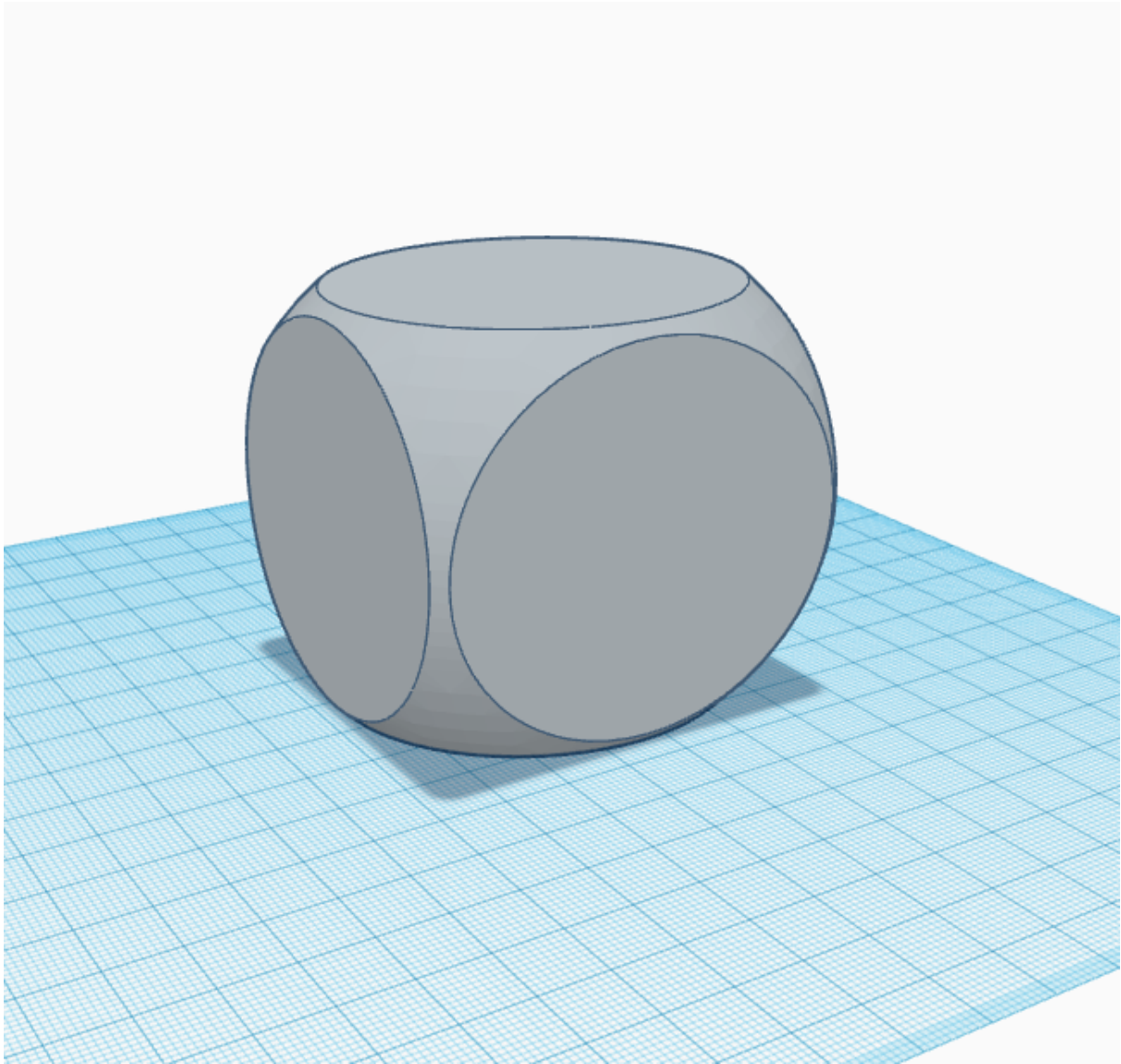
A continuación. Modelo 3D por. Felipe Charabora

Introducción al Diseño 3D del Dado Electrónico en Tinkercad:

Este proyecto en Tinkercad tiene como objetivo crear un modelo 3D de un dado electrónico. La plataforma Tinkercad proporciona una interfaz amigable para la creación y manipulación de objetos tridimensionales, lo que facilita el diseño de la caja que simulará ser un dado, con el agregado de un botón y un display de siete segmentos.

Diseño General de la Caja en Tinkercad:

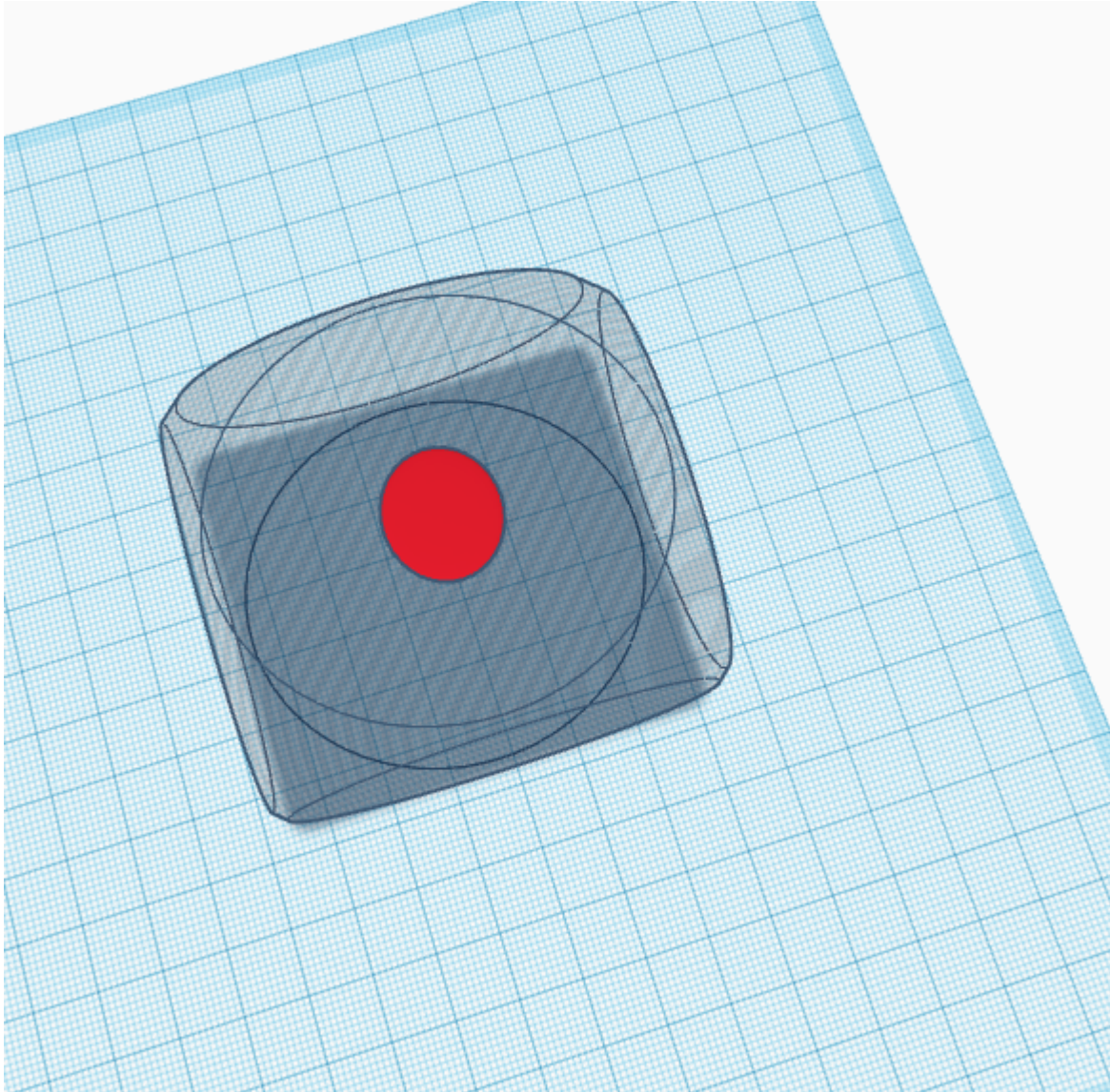
Descripción de la creación del cuerpo principal del dado electrónico en Tinkercad, utilizando formas básicas y manipulaciones para lograr una estructura sencilla y atractiva. Se destaca el uso de bloques para representar componentes clave.



(Teniendo en cuenta que puede ser tranquilamente una caja de cartón, el diseño es el insertado anteriormente)

Ubicación Estratégica del Botón en Tinkercad:

Explicación detallada sobre cómo se integra el botón en la parte superior de la caja utilizando las herramientas de posicionamiento y alineación de Tinkercad. Se abordan consideraciones de diseño para garantizar una interacción intuitiva.



Diseño del Display de Siete Segmentos en Tinkercad:

Descripción del proceso para incorporar el display de siete segmentos en una de las partes frontales de la caja utilizando formas y alineación en Tinkercad. Se destaca la precisión en la disposición para lograr una visualización efectiva.

Consideraciones Estéticas y Colores en Tinkercad:

Uso de colores y acabados en Tinkercad para mejorar la estética del dado electrónico. Se explora cómo la paleta de colores y la selección de texturas pueden contribuir a la apariencia general del diseño.

Ergonomía y Manipulación en Tinkercad:

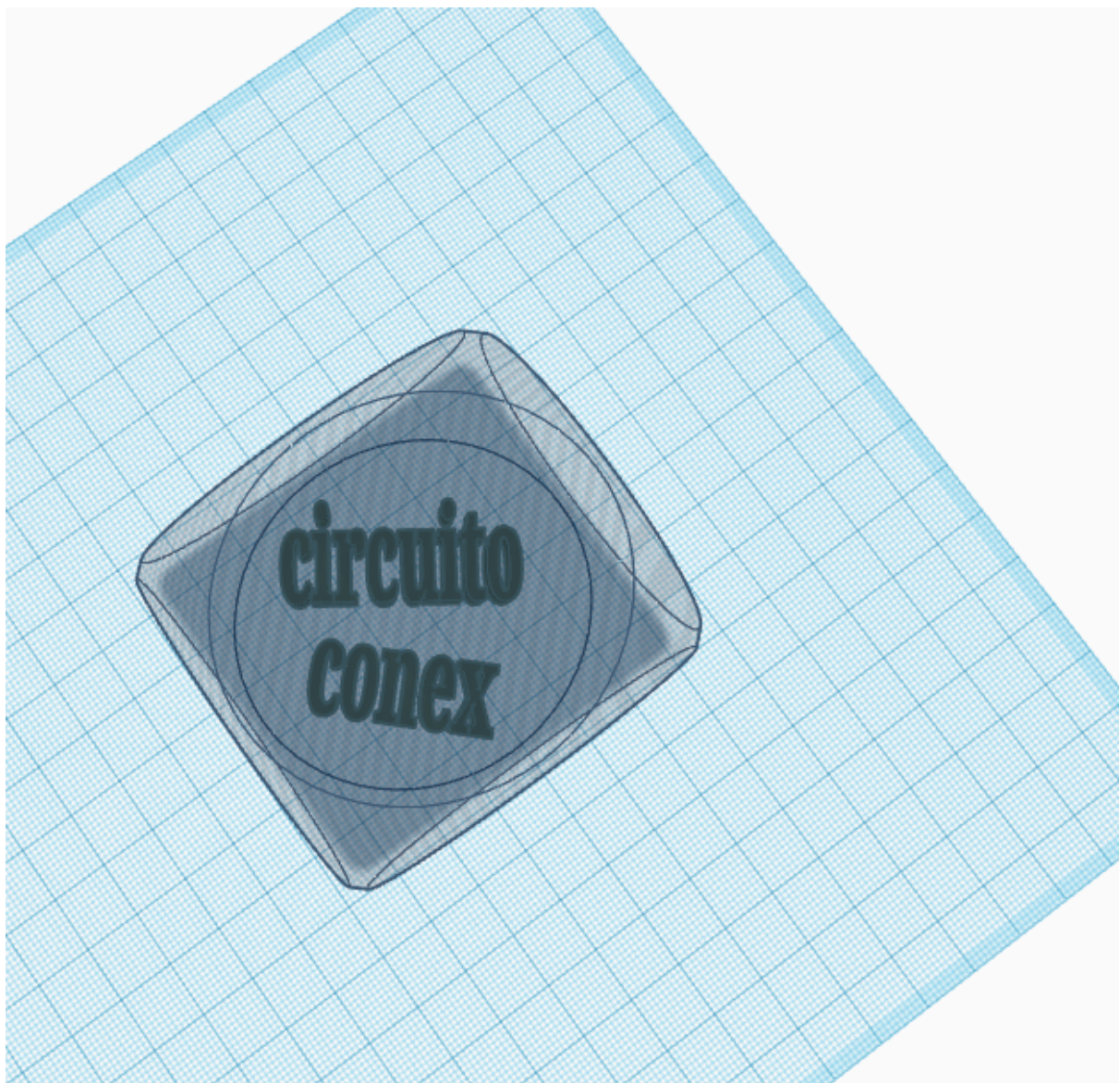
Análisis de la ergonomía del diseño utilizando Tinkercad, ajustando la forma y tamaño para garantizar una interacción cómoda con el usuario. Se muestra cómo las herramientas de manipulación en Tinkercad facilitan la adaptación del diseño.

Espacios para Conexiones y Acceso a Componentes en Tinkercad:

Creación de espacios y aberturas en Tinkercad para permitir conexiones eléctricas y facilitar el acceso a componentes internos. Se destaca el uso de formas para generar aberturas y asegurar la funcionalidad del diseño.

Integración de Componentes Internos en Tinkercad:

Este apartado se centrará en la integración eficiente de componentes internos, como la placa Arduino, en el diseño en Tinkercad. Se describirán estrategias para asegurar un ajuste adecuado y accesibilidad a estos elementos clave dentro de la estructura de la caja.



Optimización para la Impresión 3D en Tinkercad:

Guía sobre cómo optimizar el diseño para la impresión 3D en Tinkercad, incluyendo la orientación adecuada de piezas y la configuración de parámetros para minimizar la necesidad de soportes durante la impresión.

Iteración y Ajustes Virtuales en Tinkercad:

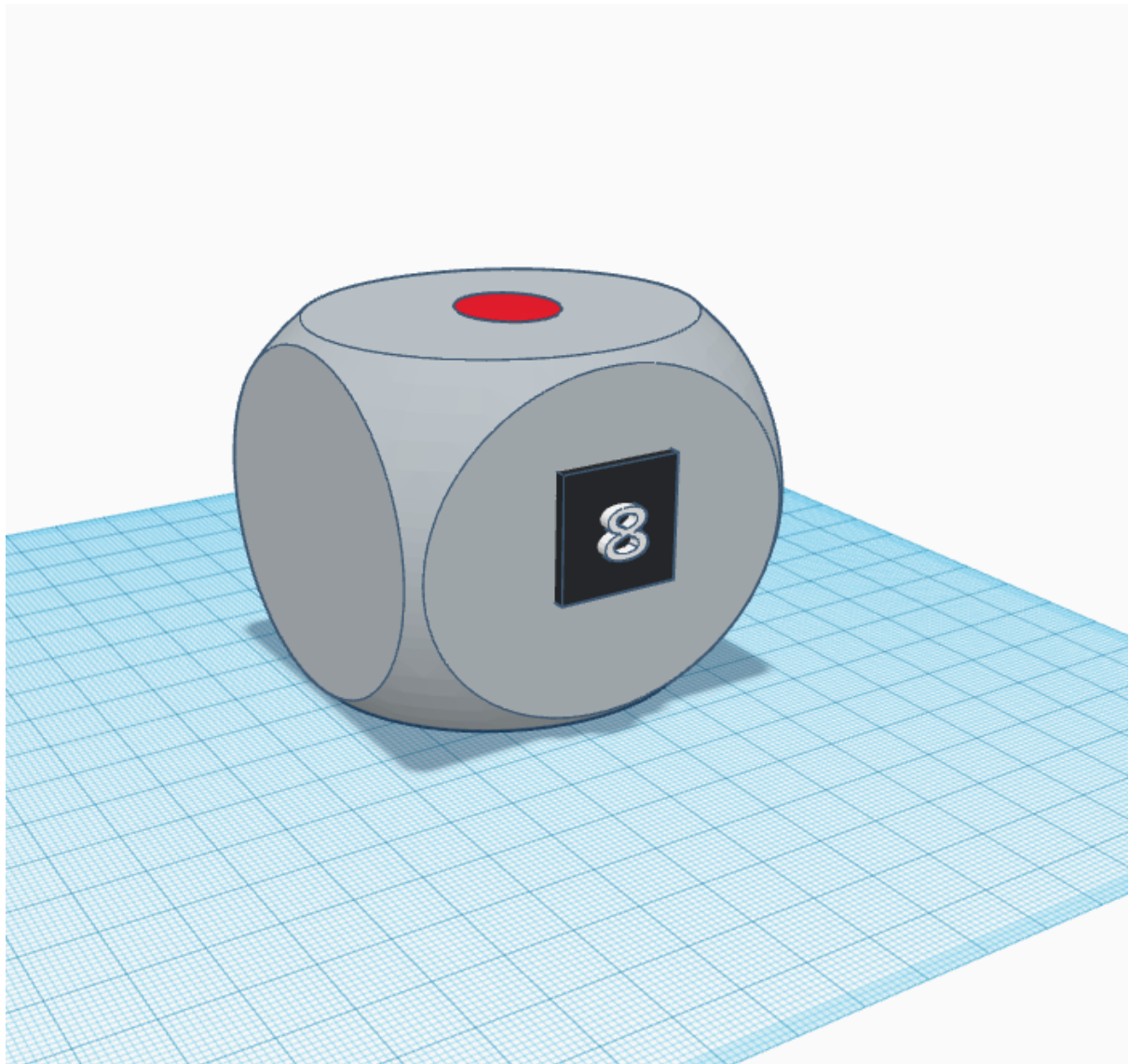
Reseña sobre el proceso iterativo en Tinkercad, realizando ajustes y modificaciones según sea necesario para mejorar la forma y la función del dado electrónico antes de la impresión física.

Pruebas Virtuales en Tinkercad:

Uso de las herramientas de simulación en Tinkercad para realizar pruebas virtuales, verificando la interacción y la apariencia del diseño. Se destaca la utilidad de estas pruebas antes de proceder a la impresión física.

Exportación de Archivos STL y Documentación en Tinkercad:

Instrucciones sobre cómo exportar los archivos STL desde Tinkercad para su posterior impresión 3D. Se resalta la importancia de documentar el diseño y compartir los archivos STL para la reproducción y mejora del dado electrónico.



(El resultado 3D NO ES LO QUE ESPERÁBAMOS, dado a la poca falta de empeño que tuvo el grupo en el tiempo para realizar el trabajo, esto es lo que pudimos hacer)

ALGUNAS CONSIDERACIONES GLOBALES A TENER
EN CUENTA SOBRE EL TRABAJO

Escalabilidad del Diseño:

Se debe considerar la capacidad del diseño para escalar en tamaño o complejidad, permitiendo futuras expansiones o adiciones de características sin comprometer la integridad del proyecto.

Facilidad de Mantenimiento:

Es crucial diseñar el proyecto de manera que facilite el mantenimiento y la reparación. Considerar accesibilidad a componentes internos, conexiones y posibles puntos de fallo para una resolución eficiente de problemas.

Eficiencia Energética:

Valorar la eficiencia en el consumo de energía, especialmente si el dado electrónico se alimenta mediante baterías. Optimizar el código y seleccionar componentes de bajo consumo puede prolongar la duración de la batería.

Compatibilidad con Futuras Actualizaciones:

Prever la posibilidad de futuras actualizaciones de hardware o software. El diseño debe ser lo suficientemente flexible para adaptarse a mejoras tecnológicas sin requerir cambios significativos.

Consideraciones Medioambientales:

Evaluar el impacto medioambiental del proyecto, desde la elección de materiales para la caja hasta la eficiencia en el uso de energía. Priorizar opciones respetuosas con el medio ambiente puede ser una consideración importante.

Adaptabilidad a Usuarios con Diversas Habilidades:

Diseñar el dado electrónico de manera que sea accesible para usuarios con diversas habilidades y necesidades. Considerar el tamaño de los botones, la visibilidad del display y otras características para garantizar una experiencia inclusiva.

Compatibilidad con Plataformas de Impresión 3D:

Verificar la compatibilidad del diseño con diversas impresoras 3D y materiales de impresión. Asegurarse de que el modelo sea fácilmente reproducible en diferentes entornos de impresión.

Documentación Integral del Proyecto:

Mantener documentación detallada que incluya instrucciones de ensamblaje, lista de materiales, enlaces a recursos y archivos STL. Facilitar la comprensión y reproducción del proyecto por parte de la comunidad.

Seguridad Eléctrica y Mecánica:

Incorporar medidas de seguridad eléctrica, como resistencias limitadoras de corriente, y asegurar que los componentes mecánicos están diseñados para evitar posibles riesgos. Cumplir con estándares de seguridad eléctrica y mecánica es esencial.

Consideraciones Estéticas y de Usabilidad:

- Evaluar la estética general del diseño y cómo esta contribuye a la experiencia del usuario. Garantizar que la interfaz sea intuitiva y estéticamente agradable para una interacción positiva.

Conclusión del Proyecto:

El diseño y desarrollo de este dado electrónico en Tinkercad ha sido un proceso que ha combinado la creatividad con la funcionalidad. La integración armoniosa de un diseño simple de caja que emula la forma de un dado convencional, junto con componentes electrónicos como el botón y el display de siete segmentos, ha dado como resultado un proyecto que no solo es estéticamente atractivo, sino también interactivo y entretenido.

A lo largo de este proceso, se han abordado consideraciones clave, desde la seguridad eléctrica hasta la optimización para la impresión 3D, asegurando un proyecto integral y versátil. La adaptabilidad a futuras

actualizaciones, la facilidad de mantenimiento y la atención a aspectos medioambientales reflejan un enfoque holístico hacia el diseño.

La documentación detallada y la exportación de archivos STL permiten compartir este proyecto con la comunidad, fomentando la colaboración y la mejora continua. Las pruebas virtuales realizadas en Tinkercad han permitido simular diversas interacciones del usuario, asegurando la efectividad y la usabilidad del dado electrónico.

En resumen, este proyecto no solo es un ejercicio de diseño 3D y electrónica, sino también una expresión de cómo la tecnología puede mezclarse armoniosamente con elementos clásicos. Con la finalización de esta explicación, se invita a la comunidad a explorar, reproducir y mejorar este dado electrónico, contribuyendo a la evolución constante de proyectos creativos y educativos.