

# Trivialist



**Fernando Rodríguez-Millán Pérez**

**Juan Pablo Domínguez-Fernández**  
**Sambruno**

**David Vázquez Vicario**



# Indice

## Introducción

### ➤ **Justificación del Diseño:**

- Importancia del Diseño Centrado en el Usuario
- Objetivo y Metas del Proyecto
- Beneficios esperados

### ➤ **Investigación y Análisis**

- Datos Demográficos y Segmentación
- Necesidades y Comportamientos
- Insights y Hallazgos claves

### ➤ **Diseño de la Interfaz**

- Wireframes
- Prototipos
- Guías de estilo

### ➤ **Validación y Pruebas**

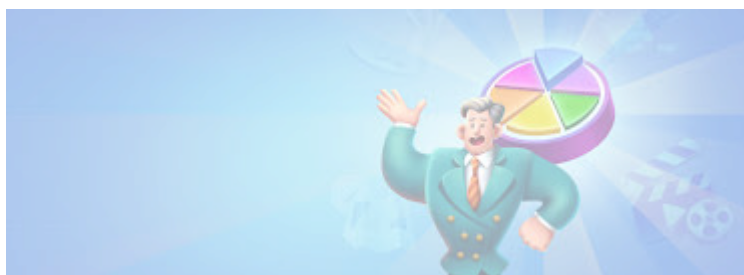
- Metodología de pruebas
- Feedback de usuarios
- Iteraciones y mejoras



# Introducción

Bienvenidos a la documentación de **Trivialist**, un juego de preguntas y respuestas diseñado para combinar diversión y conocimiento. El diseño gráfico del juego fue desarrollado en **Figma**, permitiendo crear una interfaz atractiva y una experiencia de usuario fluida. Por otro lado, la implementación técnica y las funcionalidades se realizaron en **Android Studio**, utilizando herramientas modernas para su desarrollo en la plataforma Android.

Este documento detalla el proceso de diseño, desarrollo y estructura técnica del juego, sirviendo como guía para su comprensión, mantenimiento o futura evolución. ¡Comencemos!





# 1. Justificación del diseño

## 1.1 Importancia del diseño centrado en el usuario

El diseño centrado en el usuario (DCU) ha sido el núcleo del desarrollo de *Trivialist*. Este enfoque asegura que cada decisión de diseño esté orientada a satisfacer las necesidades, expectativas y comportamientos de los jugadores. Al priorizar la experiencia del usuario:

- Se mejora la facilidad de uso y accesibilidad, permitiendo que los jugadores se enfoquen en disfrutar del juego sin complicaciones.
- Se crea una interfaz visualmente atractiva que potencia el compromiso del usuario.
- Se promueve una navegación intuitiva que facilita el aprendizaje y reduce la curva de adaptación al juego.

En un juego multijugador como *Trivialist*, donde la interacción social es clave, un diseño centrado en el usuario fomenta un entorno más amigable, inclusivo y atractivo para diferentes tipos de jugadores.

## 1.2 Objetivos y Metas del proyecto

El objetivo principal de *Trivialist* es proporcionar una experiencia de entretenimiento educativo que combine diversión y aprendizaje, creando un entorno en línea competitivo pero accesible para un rango diverso de jugadores.

Las metas específicas incluyen:

1. **Diseñar un entorno visual atractivo y funcional:** Utilizar principios de diseño gráfico para crear un juego que sea visualmente llamativo y que comunique eficazmente las funciones del juego.
2. **Optimizar la experiencia de juego multijugador en línea:** Asegurar que la conexión entre los participantes sea estable y fácil de gestionar.
3. **Promover el aprendizaje interactivo:** Utilizar preguntas de diferentes temáticas para enriquecer los conocimientos de los jugadores mientras compiten.



4. **Personalización de la experiencia de juego:** Ofrecer opciones para ajustar el número de jugadores y la dificultad de las preguntas.

## **1.3 Beneficios esperados**

El desarrollo y lanzamiento de *Trivialist* generará una serie de beneficios, tanto para los jugadores como para el mercado de juegos educativos:

- **Para los jugadores:**
  - Un espacio para aprender y entretenerse simultáneamente.
  - Mejora del trabajo en equipo y la sana competencia entre amigos o familiares.
  - Incremento en la agilidad mental y conocimientos generales a través de preguntas temáticas.
- **Para el mercado:**
  - Ampliación del catálogo de juegos educativos disponibles en dispositivos móviles.
  - Una propuesta innovadora que combina diseño gráfico de calidad y tecnología avanzada en desarrollo de aplicaciones.
  - Promoción de una comunidad de usuarios interesados en actividades recreativas que fomentan el aprendizaje.

En resumen, *Trivialist* busca posicionarse como una opción líder en el segmento de juegos educativos, destacando por su diseño centrado en el usuario, jugabilidad fluida y el valor añadido del aprendizaje interactivo.

## 2. Investigación y Análisis

### 2.1 Datos Demográficos y Segmentación

La investigación inicial identificó los grupos demográficos y segmentos de mercado que representan a los usuarios potenciales de *Trivialist*. Los resultados más relevantes son:

- **Grupo de edad:**
  - Principalmente entre 15 y 35 años, con un interés compartido por los juegos de trivia y la competencia amistosa.
  - Subsegmento adicional de 10 a 14 años para uso educativo bajo supervisión.
- **Género:**
  - El juego es inclusivo y atractivo para todos los géneros, mostrando una preferencia equitativa en las pruebas realizadas.
- **Preferencias tecnológicas:**
  - Usuarios con acceso a dispositivos móviles Android y una conexión a internet estable.
  - Preferencia por aplicaciones que ofrezcan personalización, interacción social y retos.
- **Segmentación psicográfica:**
  - Personas con interés en el aprendizaje informal, la competencia en juegos intelectuales y las actividades sociales en línea.





## 2.2 Necesidades y Comportamiento

A partir de entrevistas, encuestas y observación de patrones de uso en juegos similares, se identificaron las siguientes necesidades y comportamientos:

- **Necesidades clave:**
  - **Facilidad de acceso:** Los jugadores quieren una interfaz rápida, sencilla y sin barreras para comenzar a jugar.
  - **Conexión social:** La posibilidad de jugar con amigos o familiares en línea es un factor motivador principal.
  - **Desafíos personalizados:** Los usuarios buscan juegos que les permitan ajustar el nivel de dificultad y explorar temas de interés.
  - **Recompensa emocional:** Satisfacción al ganar o aprender algo nuevo.
- **Comportamientos recurrentes:**
  - Preferencia por partidas rápidas de 5 a 15 minutos.
  - Tendencia a invitar amigos a través de códigos o compartir en redes sociales.
  - Expectativa de retos equilibrados: Selección de dificultad en el tipo de preguntas, pudiendo ser fácil o difícil.

## 2.3 Insights y Hallazgos claves

El análisis de los datos recopilados arrojó varios hallazgos significativos que guiaron el desarrollo de *Trivialist*:

1. **El atractivo visual es crítico:** Los usuarios priorizan un diseño llamativo y claro, especialmente cuando cada categoría tiene colores diferenciados y consistentes.
2. **La personalización mejora el compromiso:** Poder elegir el número de jugadores y el nivel de dificultad aumenta el sentido de control y satisfacción en el usuario.
3. **La accesibilidad define la participación:** La funcionalidad de ingresar un código para unirse a una partida es esencial para facilitar la conexión rápida entre jugadores.

4. **La variedad de temáticas fomenta el aprendizaje:** Las categorías amplias como historia, deporte, geografía, matemáticas y arte fueron bien recibidas por su capacidad de atraer a diferentes perfiles de usuarios.
5. **La simplicidad tecnológica es indispensable:** Los usuarios esperan un juego que no consuma demasiados recursos del dispositivo y que sea fácil de instalar y usar.

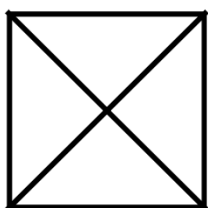
Con base en estos insights, *Trivialist* fue diseñado para satisfacer las expectativas de su público objetivo y destacar en el mercado como un juego educativo y entretenido de alta calidad.

## 3. Diseño Interfaz

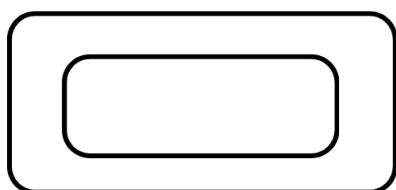
### 3.1 Wireframe

Un **wireframe** es un esquema visual o boceto de baja fidelidad que representa el diseño básico y la estructura de una página o pantalla de una aplicación o sitio web. Es una herramienta utilizada en el proceso de diseño para planificar y visualizar la disposición de elementos clave, como botones, menús, imágenes, textos y otros componentes, antes de pasar a un diseño más detallado o funcional.

Para mostrar en una imagen :



Para mostrar un botón :







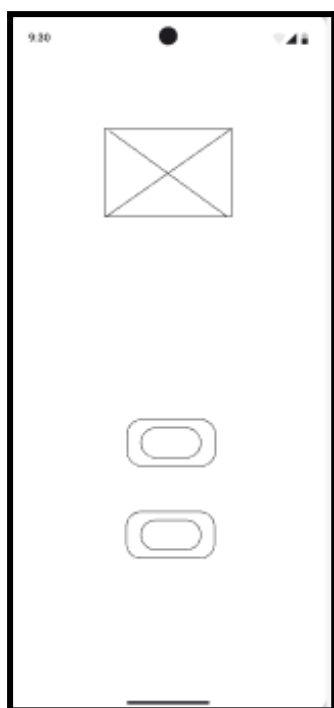
Para mostrar un texto:



Para mostrar un campo de texto o escritura:

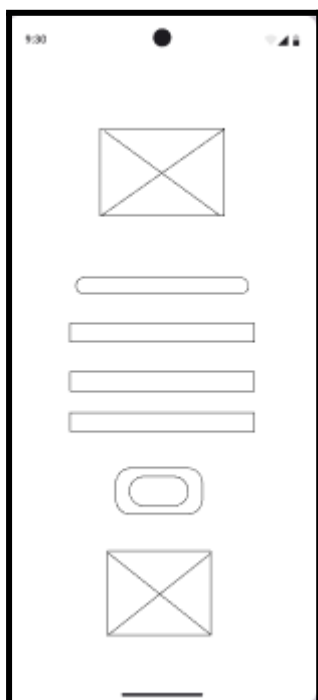


Pantalla de Inicio:

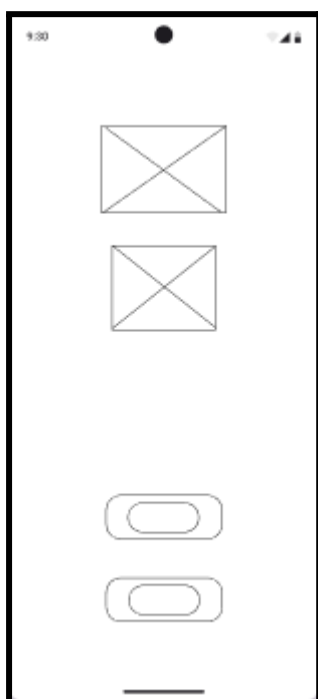




Pantalla de Registro:

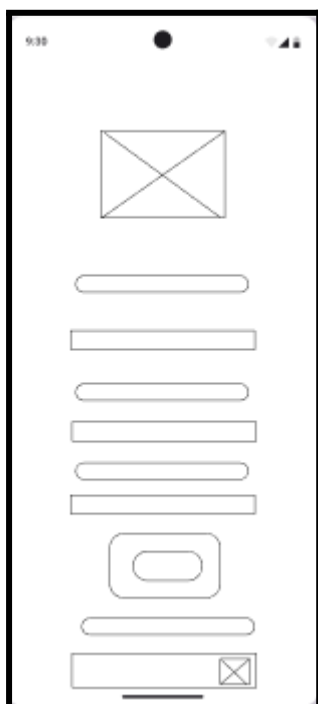


Pantalla Principal:

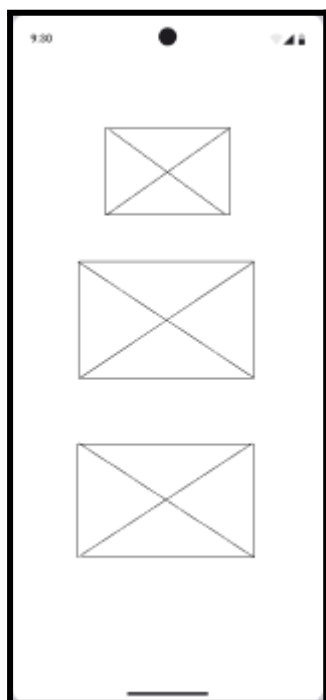




Pantalla Comenzar:

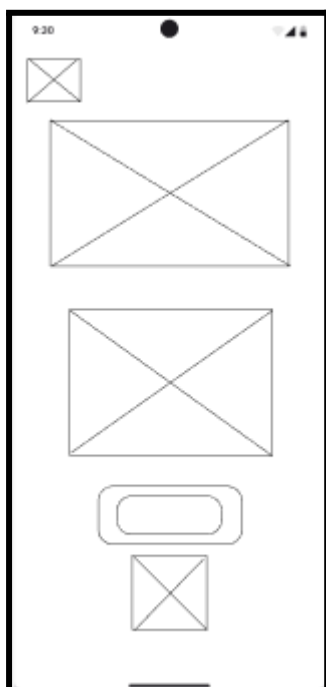


Pantalla Ayuda:

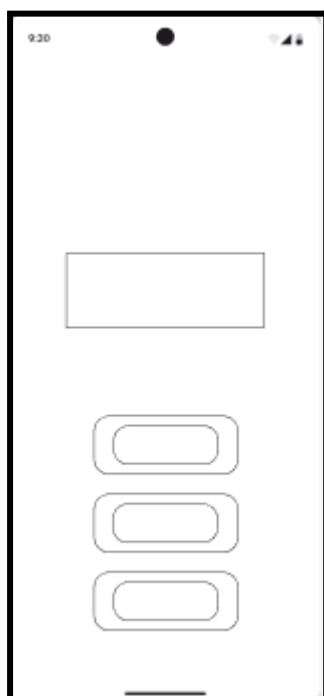




Pantalla de Partida:

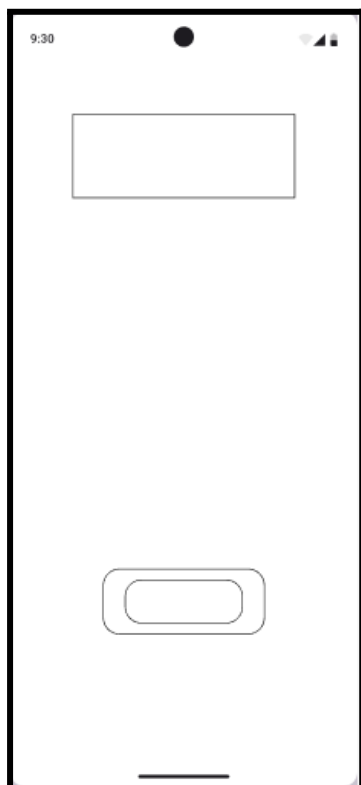


Pantalla de Preguntas:





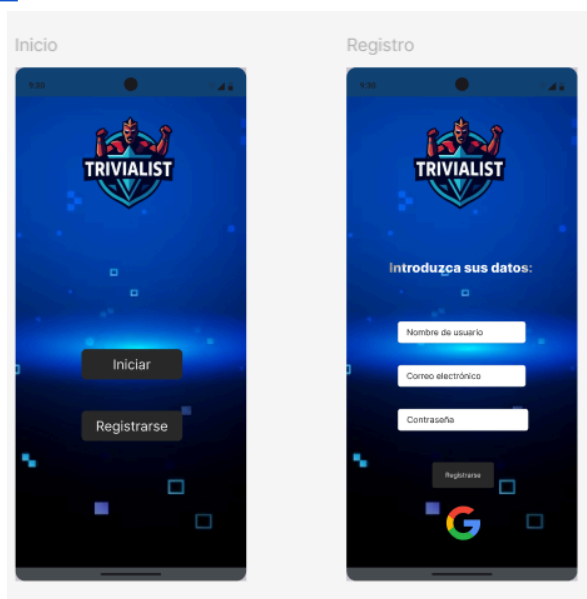
Pantalla de Victoria o Derrota:



## 3.2 Prototipos

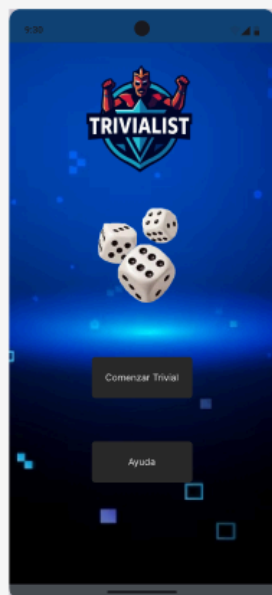
Enlace de Figma:

<https://www.figma.com/design/EwPucGmILpRFmElklKmFEv/Trivialist?node-id=0-1&m=dev&t=mfg5NEeSGekoekC2-1>

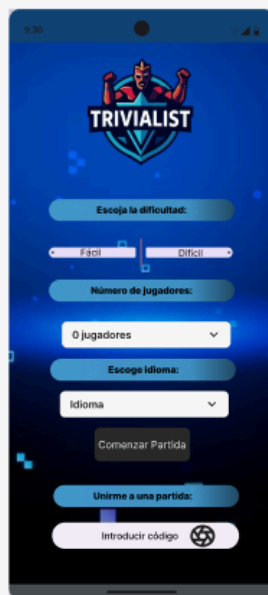




Pantalla Principal



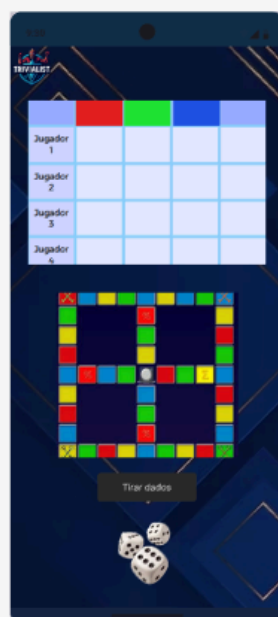
Comenzar trivial

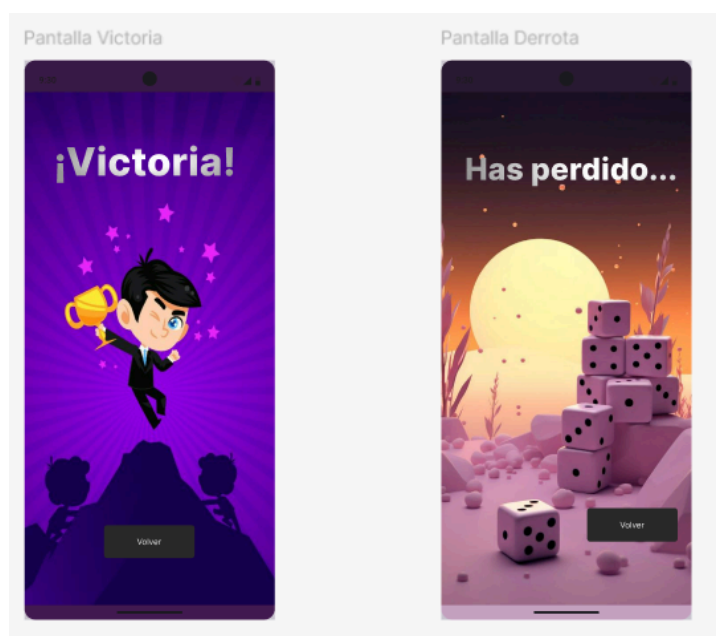


Ayuda



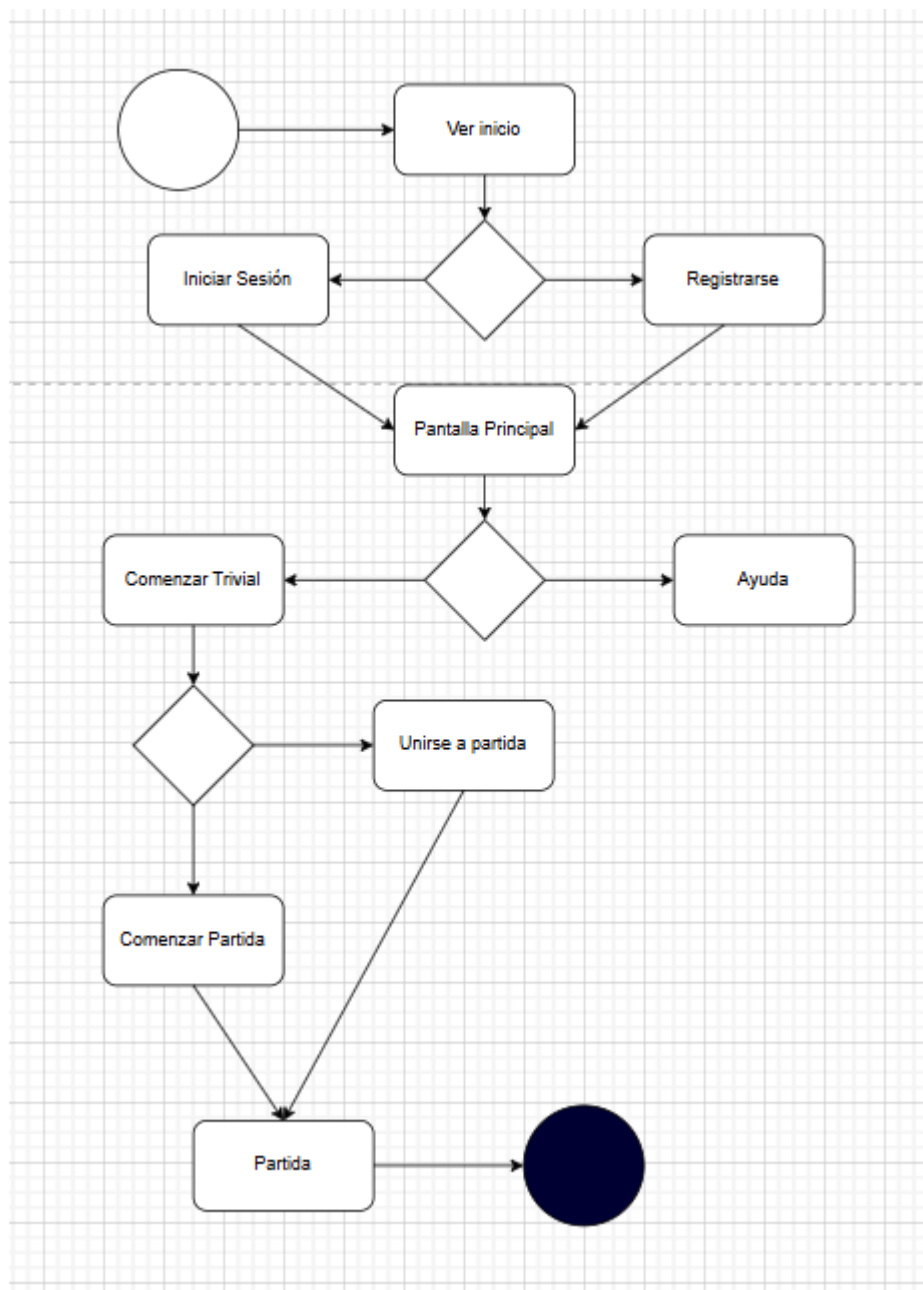
Partida



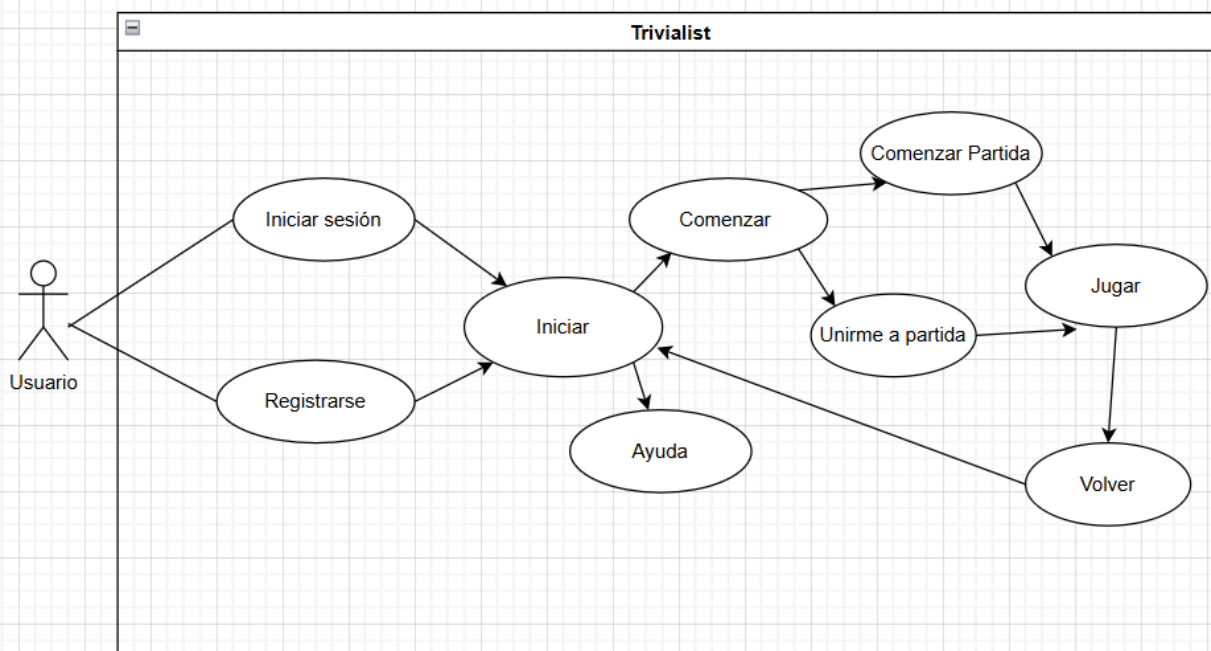


## Diagramas

### Diagrama de Actividad:



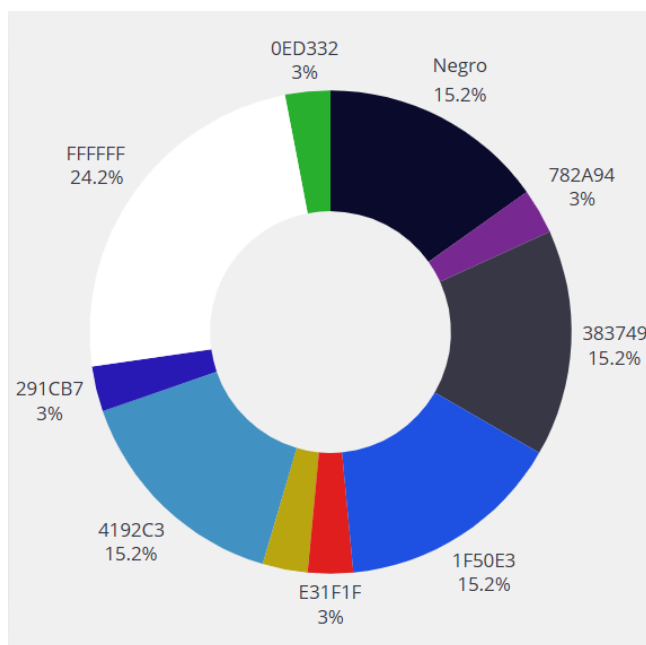


**Diagrama de casos de usos:**



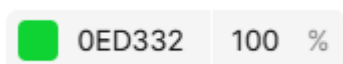
## 3.3 Guías de estilo

### Paleta de colores



### Categorización de los colores

#### 1- Verde vibrante (#0ED332):

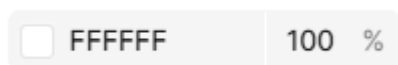


Se utiliza para las preguntas de Geografía.

Aparece en los botones de selección de respuesta en las pantallas de preguntas específicas de esa categoría.



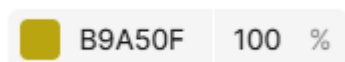
## 2- Blanco (#FFFFFF):



Se usa como fondo de los botones de respuestas no seleccionadas.

También podría estar presente en los textos o contornos para mantener contraste con los colores más vivos.

## 3- Amarillo (#B9A50F):



Representa las preguntas de **Historia**.

Vemos este color en los botones de selección de respuestas para preguntas de esa categoría.

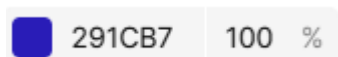
## 4- Azul claro (#4192C3):



Puede usarse para algunos detalles **secundarios** o **efectos visuales** en el diseño.

Este color parece no estar directamente visible en las pantallas principales, pero podría complementar los elementos visuales relacionados con tonos fríos.

## 5- Azul oscuro (#291CB7):

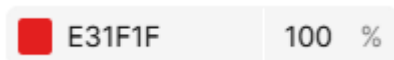


Representa las preguntas de **Matemáticas**.

Se usa en los botones de selección de respuestas para preguntas de esta categoría.



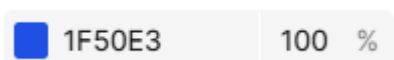
### 6- Rojo (#E31F1F):



Representa las preguntas de **Deportes**.

Se encuentra en los botones de selección de respuestas para preguntas de esta categoría.

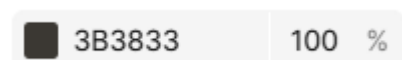
### 7- Azul medio (#1F50E3):



Puede usarse como un color de soporte para otros elementos azules en las pantallas.

Podría ser parte de los detalles o efectos visuales en el fondo.

### 8- Gris oscuro (#3B3833):



Usado como fondo en varias pantallas (por ejemplo, en las preguntas).

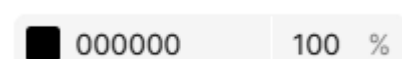
Sirve para dar contraste a los elementos brillantes y textos.

### 9- Morado (#782A94):

Representa las preguntas de **Arte**.

Presente en los botones de selección de respuesta para preguntas de esta categoría.

### 10- Negro (#000000):



Presente en los fondos de todas las pantallas.

Ayuda a resaltar los colores vivos y los elementos interactivos.



### Tipografía:

La tipografía en este proyecto combina tres fuentes principales las cuales son un inter extra bold, font weight regular/ body font family, Roboto, Righteous y por último body strong, cada uno de ellos será mostrado por separado con una muestra de su funcionamiento en el trabajo como también otros detalles los cuales pueden ser su tamaño, etc.

#### **Inter extra bold:**

Inter

▼

Extra Bold

▼

24

▼

**Introduzca sus datos:**

**Fuente:** Inter.

**Estilo:** Extra bold.

**Tamaño:** 24

Utilizamos esta tipografía para los enunciados de diferentes acciones como puede ser para la introducción de datos en el registro o la frases al finalizar la partida de victoria o derrota.

#### **Font weight regular/Body font family:**

T

Body/Font Family

#

Body/Font Weight Regula

15.14

Nombre de usuario

**Fuente:** Font Family.

**Estilo:** Font Weight Regular.



**Tamaño:** 15.14

Esta tipografía ha sido utilizada para las cajas de texto en los apartados de datos.

**Roboto:**

Roboto

ExtraBold



15

trivialistsoporte@gmail.com

**Righteous:**

Righteous

Regular



15

Soporte Técnico:



## **Iconografía:**

### **Iconografía de la primera Activity (Inicio):**

#### **Logotipo de "Trivialist":**

En la parte superior, el mismo diseño gráfico del superhéroe visto en la imagen anterior. Representa fuerza, conocimiento y un tema lúdico, con un escudo en forma de diamante, usando colores llamativos como rojo, azul y negro.

#### **Botón "Iniciar":**

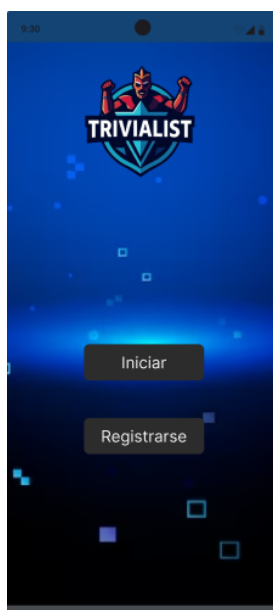
Un botón rectangular de color oscuro con el texto "Iniciar", indicando la acción de iniciar sesión o empezar a jugar.

#### **Botón "Registrarse":**

Un segundo botón similar al anterior, con el texto "Registrarse", que probablemente lleva a una pantalla de creación de cuenta o registro de usuario.

#### **Fondo digital:**

Un fondo en tonos de azul oscuro con un efecto de iluminación degradada y pequeños elementos visuales como píxeles y cuadros flotantes, que sugieren un estilo moderno y tecnológico.





### Iconografía de la segunda Activity (Registro)

**Logotipo de "Trivialist":** En la parte superior, un gráfico que representa un superhéroe con estilo caricaturesco, usando colores llamativos como rojo, negro y azul, con un escudo en forma de diamante, asociado al tema del juego o trivia.

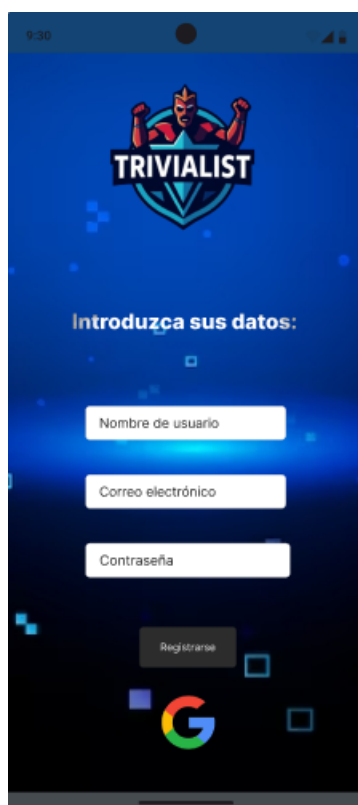
**Campos de entrada:** Tres cajas de texto:

- "Nombre de usuario" (para escribir un alias o nickname).
- "Correo electrónico" (para ingresar una dirección de email).
- "Contraseña" (campo para ingresar una clave, posiblemente enmascarada).

**Botón de registro:** Una caja negra con el texto "Registrarse", lo que indica una acción primaria.

**Ícono de Google:** En la parte inferior, el logotipo de Google, representando una opción de registro o inicio de sesión rápido utilizando una cuenta de Google.

**Fondo digital:** Un diseño en tonos azules oscuros, con elementos visuales como cuadrados y pixeles luminosos flotantes, que sugieren un tema tecnológico o futurista.







### **Iconografía de la tercera Activity (Pantalla Principal)**

#### **Logotipo de "Trivialist":**

En la parte superior, nuevamente el superhéroe con un escudo en forma de diamante, representando el tema principal del juego. Los colores rojo, azul y negro destacan como en las imágenes previas.

#### **Dado flotante tridimensional:**

Un conjunto de dados blancos con puntos negros flotando en el centro de la pantalla, que refuerzan la temática de juego o azar, posiblemente relacionado con dinámicas de trivia o selección aleatoria.

#### **Botón "Comenzar Trivial":**

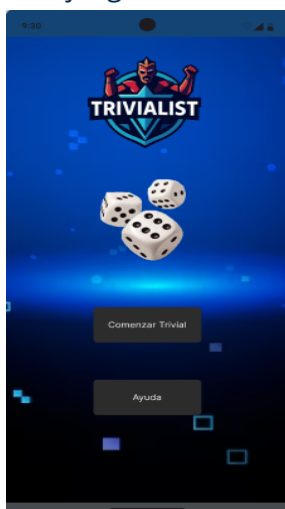
Un botón rectangular oscuro con el texto "Comenzar Trivial", que indica una acción principal para empezar el juego o acceder al contenido principal de la aplicación.

#### **Botón "Ayuda":**

Un segundo botón más pequeño y con el texto "Ayuda", que probablemente lleva a una sección con información o soporte para el usuario.

#### **Fondo digital:**

Un diseño en tonos azules oscuros con gradientes de luz y pequeños cuadros flotantes que sugieren un estilo tecnológico o de videojuego, consistente con las pantallas anteriores.





### Iconografía de la cuarta Activity (Comenzar Trivial)

#### Logotipo de “Trivialist”:

En la parte superior, nuevamente el superhéroe con un escudo en forma de diamante, representando el tema principal del juego que es la competencia o el desafío intelectual. Los colores rojo, azul y negro destacan como en las imágenes previas.

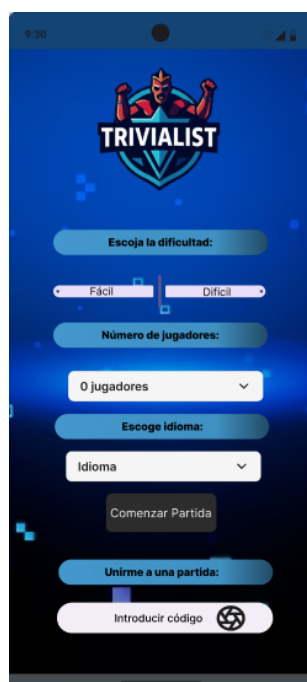
**Tipografía fuerte y llamativa:** Da una sensación de dinamismo y competencia.

**Colores azul y negro con detalles en blanco:** Transmiten tecnología, modernidad y un ambiente digital.

**Icono de un balón con red en el botón de código:** Puede simbolizar la conexión en línea o el acceso a partidas mediante un código.

**Barras y botones con diseño futurista:** Refuerzan la idea de un juego interactivo y moderno.

**Uso de controles deslizantes y desplegables:** Muestra que hay opciones configurables para personalizar la experiencia de juego.





## Iconografía de la quinta Activity (Ayuda)

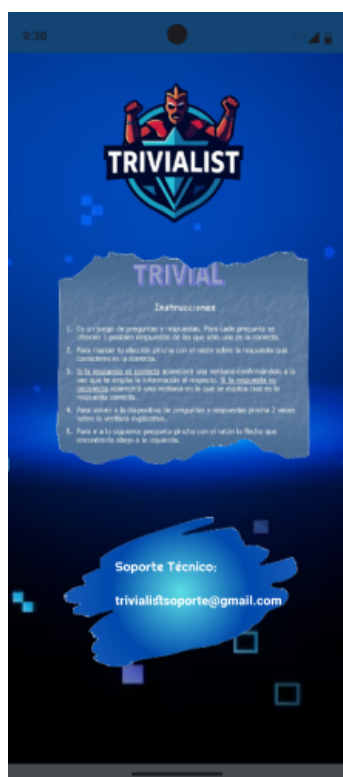
### Logotipo de "Trivialist":

En la parte superior, nuevamente el superhéroe con un escudo en forma de diamante, representando el tema principal del juego que es la competencia o el desafío intelectual. Los colores rojo, azul y negro destacan como en las imágenes previas.

**Título "TRIVIAL" con efecto de papel rasgado:** Rompe con el estilo digital y evoca las reglas del juego del que va a disfrutar. Usa una fuente clara para facilitar la lectura y comprensión del juego.

**Fondo azul con efectos luminosos y píxeles dispersos:** Refleja un ambiente tecnológico y dinámico.

**Sección de soporte técnico con efecto de pincelada:** Destaca visualmente el correo de contacto, diferenciándolo del resto.



### Iconografía de la sexta Activity (Partida)

**Tablero de colores con casillas rojas, verdes, amarillas y azules:** Sugiere que cada color puede representar un tipo de pregunta o desafío diferente, similar a los juegos de trivia clásicos.

**Tabla de jugadores con colores asignados:** Indica el orden y progreso de cada participante, facilitando la organización del juego.

**Dado tridimensional en la parte inferior:** Representa el mecanismo de avance en el juego, sugiriendo un factor de azar en la jugabilidad.

**Botón "Tirar dados":** Elemento interactivo que permite a los jugadores avanzar en el tablero.

**Fondo azul con patrones geométricos:** Mantiene una estética elegante y tecnológica, alineada con el estilo del juego.





## **Iconografía de las Actividades de la preguntas**

### **Signos de interrogación en neón:**

Colores llamativos (azul, rosa, amarillo) representan la diversidad de preguntas y la incertidumbre del juego.

El efecto de neón aporta un estilo moderno y dinámico.

### **Fondo de pared oscura con textura de ladrillos:**

Crea un contraste con los elementos brillantes, mejorando la visibilidad y el impacto visual.

Da un ambiente de concurso o desafío intelectual.

### **Categorías de preguntas diferenciadas por colores:**

Rojo (Deportes), Verde (Geografía), Azul (Matemáticas), Amarillo (Historia), Morado (Arte).

Permite una rápida identificación de la temática de cada pregunta.

### **Diseño de botones:**

Resaltan las opciones de respuesta con un diseño claro y accesible.

Su forma rectangular con bordes redondeados facilita la interacción táctil.





## **Iconografía de las Actividades de Victoria o Derrota**

### **Pantalla de Victoria**

#### **Color predominante: Morado y violeta con destellos**

Representa éxito, energía y emoción.

El fondo radiante y las estrellas refuerzan la sensación de triunfo.

#### **Personaje saltando con un trofeo**

Simboliza el logro y la celebración.

Su expresión alegre transmite la satisfacción de ganar.

#### **Tipografía grande y enérgica ("¡Victoria!")**

Refuerza la emoción del momento con un mensaje claro y positivo.

#### **Botón "Volver"**

Permite al jugador regresar al menú, con un diseño sobrio para no restar protagonismo a la celebración.

### **Pantalla de Derrota**

#### **Color predominante: Tonos cálidos (naranja, marrón, rosado) al atardecer**

Transmite calma, nostalgia y reflexión tras la derrota.

#### **Dados apilados en un paisaje desértico**

Representan el azar y la incertidumbre en el juego.

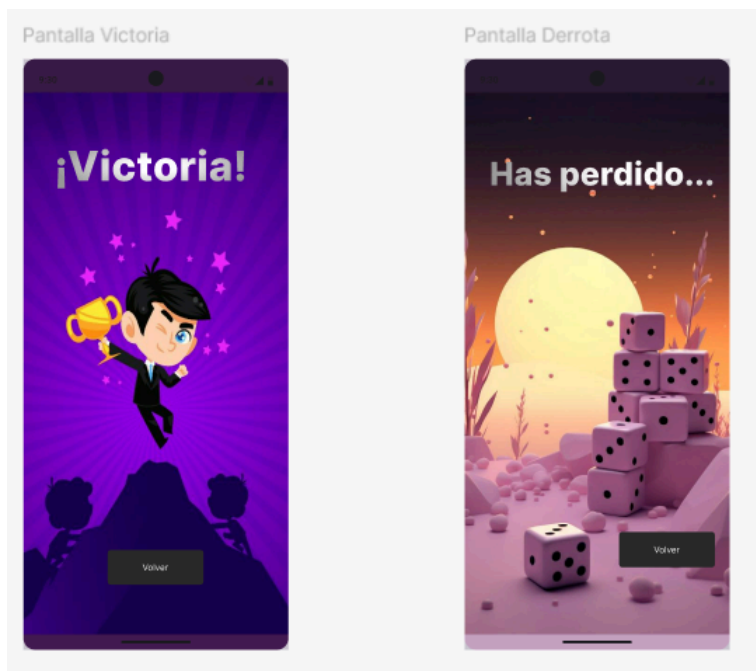
El entorno solitario refuerza el sentimiento de pérdida.

#### **Tipografía con tono suave ("Has perdido...")**

Usa un estilo menos impactante, alineado con la tristeza del momento.

## Botón "Volver"

Igual que en la pantalla de victoria, facilita la navegación de regreso.



## 4. Validaciones y pruebas

### 4.1 Pruebas Unitarias

Hemos realizado las pruebas unitarias en la **LobbyActivity**, la cual hemos utilizado las pruebas unitarias de JUnit y Mockito.

El siguiente código crea pruebas unitarias que validan el comportamiento de **LobbyActivity**, en particular, la recepción de datos a través de un **Intent** y las acciones de los botones **buttonPlay** y **buttonHelp**. Usaremos Mockito para simular el comportamiento de la clase **LobbyActivity** y sus dependencias.

Este es el código sin haberse realizado las pruebas:

```
13
14 import com.example.trivialist.databinding.ActivityLobbyBinding;
15
16 <?> public class LobbyActivity extends AppCompatActivity {
17
18     @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21         ActivityLobbyBinding binding = DataBindingUtil.setContentView(this, R.layout.activity_lobby);
22
23         Intent intent = getIntent();
24         String nameuser=intent.getStringExtra("nameUser");
25
26         binding.buttonPlay.setOnClickListener(new View.OnClickListener() {
27             @Override
28             public void onClick(View view) {
29                 Intent intent = new Intent(packageContext, LobbyActivity.this, ConfigLobbyActivity.class);
30                 intent.putExtra("nameUser",nameuser);
31                 startActivity(intent);
32             }
33         });
34         binding.buttonHelp.setOnClickListener(new View.OnClickListener() {
35             @Override
36             public void onClick(View view) {
37                 Intent intent = new Intent(packageContext, LobbyActivity.this, HelpActivity.class);
38                 startActivity(intent);
39             }
40         });
41     }
42 }
```





Este es el código cuando se le realizan las pruebas unitarias:

```
package com.example.trivialist;

import android.content.Intent;
import android.view.View;

import org.junit.Before;
import org.junit.Test;
import org.mockito.Mock;
import org.mockito.MockitoAnnotations;

import static org.junit.Assert.assertEquals;
import static org.mockito.Mockito.mock;
import static org.mockito.Mockito.verify;
import static org.mockito.Mockito.when;

public class LobbyActivityTest {
```

```
public class LobbyActivityTest {

    @Mock
    private LobbyActivity lobbyActivity;

    @Mock
    private Intent mockIntent;

    @Before
    public void setup() {
        // Inicializar Mockito
        MockitoAnnotations.openMocks(this);

        // Crear un mock de LobbyActivity e Intent
        lobbyActivity = mock(LobbyActivity.class);
        mockIntent = mock(Intent.class);

        // Simular el valor del Intent
        when(mockIntent.getStringExtra("nameUser")).thenReturn("TestUser");
```

```
@Test
public void testReceiveIntentExtra() {
    // Simular que LobbyActivity recibe un Intent con "nameUser"
    String nameUser = mockIntent.getStringExtra("nameUser");

    // Validar que el nombre recibido sea correcto
    assertEquals("TestUser", nameUser);
}

@Test
public void testPlayButtonAction() {
    // Simular un clic en el botón Play
    View mockView = mock(View.class);
    lobbyActivity.findViewById(R.id.buttonPlay).performClick();

    // Verificar que se llama a startActivity con el Intent adecuado
    verify(lobbyActivity).startActivity(mockIntent);
}
```

```
@Test
public void testHelpButtonAction() {
    // Simular un clic en el botón Help
    View mockView = mock(View.class);
    lobbyActivity.findViewById(R.id.buttonHelp).performClick();

    // Verificar que se llama a startActivity con el Intent adecuado
    verify(lobbyActivity).startActivity(mockIntent);
}
}
```



## Explicación de código:

### 1. Configuración con Mockito:

- **@Mock:** Anotación que marca los objetos que serán simulados (mocks). Aquí simulamos **LobbyActivity** y **Intent**.
- **MockitoAnnotations.openMocks(this):** Inicializa los mocks antes de ejecutar las pruebas.
- **mock(...):** Crea un objeto simulado de una clase, como **LobbyActivity** o **View**.

### 2. Prueba de recepción de datos en el Intent:

```
@Test
public void testReceiveIntentExtra() {
    String nameUser = mockIntent.getStringExtra("nameUser");
    assertEquals("TestUser", nameUser);
}
```

Simulamos un Intent que contiene el valor "**TestUser**" bajo la clave "**nameUser**".

Usamos **assertEquals** para validar que el dato recibido coincide con el esperado.

### 3. Pruebas de los botones:

#### Botón Play (buttonPlay):

```
@Test
public void testPlayButtonAction() {
    View mockView = mock(View.class);
    lobbyActivity.findViewById(R.id.buttonPlay).performClick();

    verify(lobbyActivity).startActivity(mockIntent);
}
```



- Simulamos que el usuario hace clic en el botón **buttonPlay**.
- Usamos **verify(...)** para asegurarnos de que se llama a **startActivity** con un **Intent** adecuado.

#### Botón Help (buttonHelp):

```
@Test
public void testHelpButtonAction() {
    View mockView = mock(View.class);
    lobbyActivity.findViewById(R.id.buttonHelp).performClick();

    verify(lobbyActivity).startActivity(mockIntent);
}
```

Similar al caso del botón **buttonPlay**, verificamos que se llama a **startActivity** cuando el usuario hace clic en el botón **buttonHelp**.

#### Ventajas de Este Enfoque

1. **Aislamiento:** Con Mockito, no necesitamos ejecutar la aplicación completa ni interactuar con el sistema Android, lo que hace las pruebas más rápidas y controladas.
2. **Validación de Interacciones:** Usamos **verify** para confirmar que ciertos métodos, como **startActivity**, se llamaron con los parámetros correctos.
3. **Simulación:** Evitamos dependencias reales, como instancias de **Intent**, al usar mocks.



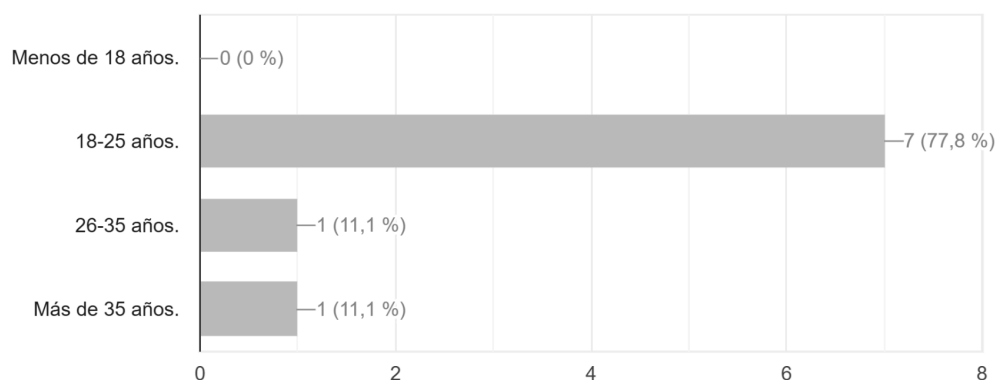
## 4.2 Feedback del usuario

Formulario para los usuarios

[https://docs.google.com/forms/d/e/1FAIpQLSdmXNq3g8DPn1YG\\_fLVg2hedOlvYgMAObOjISf1Hg--4N\\_exg/viewform?usp=header](https://docs.google.com/forms/d/e/1FAIpQLSdmXNq3g8DPn1YG_fLVg2hedOlvYgMAObOjISf1Hg--4N_exg/viewform?usp=header)

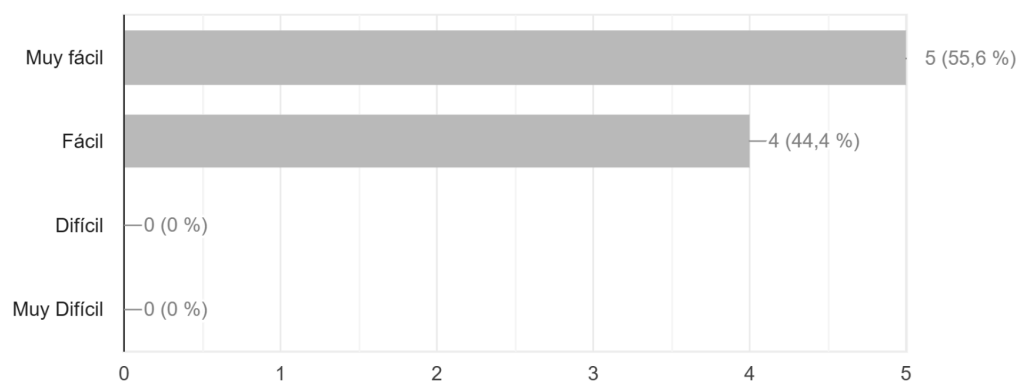
¿Cuál es tu rango de edad

0 de 9 respuestas correctas



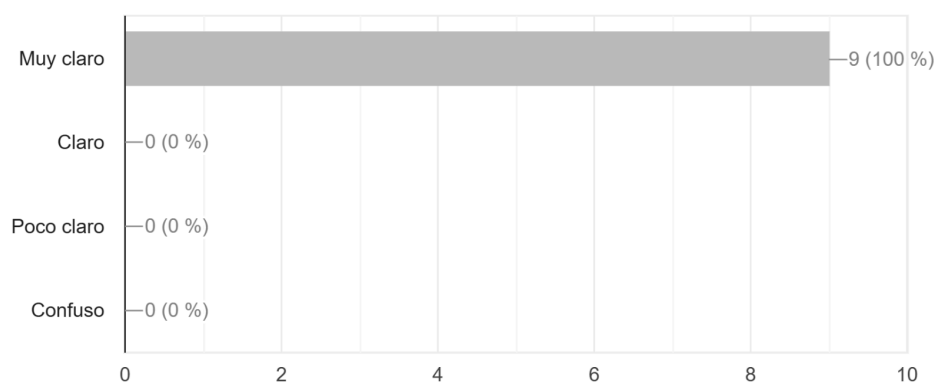
¿Qué tan fácil fue registrarse o iniciar sesión en el juego?

0 de 9 respuestas correctas



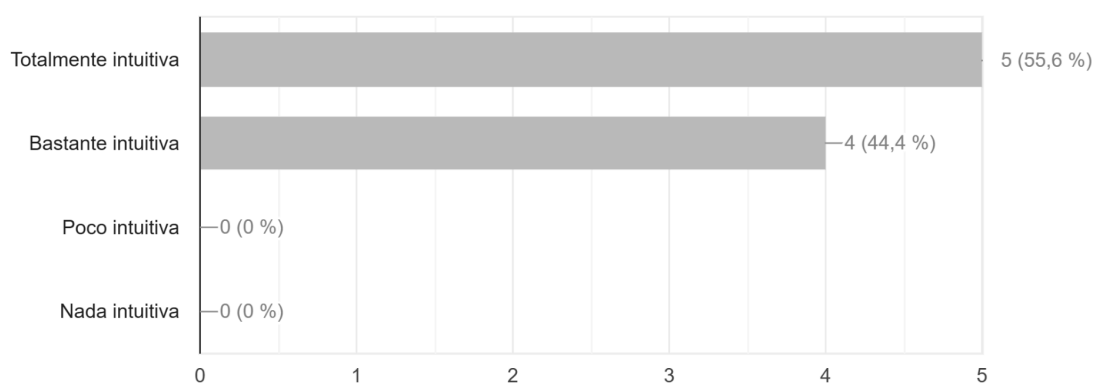
Viendo la imagen, ¿cómo de intuitiva se ve la interfaz para crear o unirse a una partida?

0 de 9 respuestas correctas



¿Te resultó intuitivo el diseño de la interfaz (botones, menús, colores)?

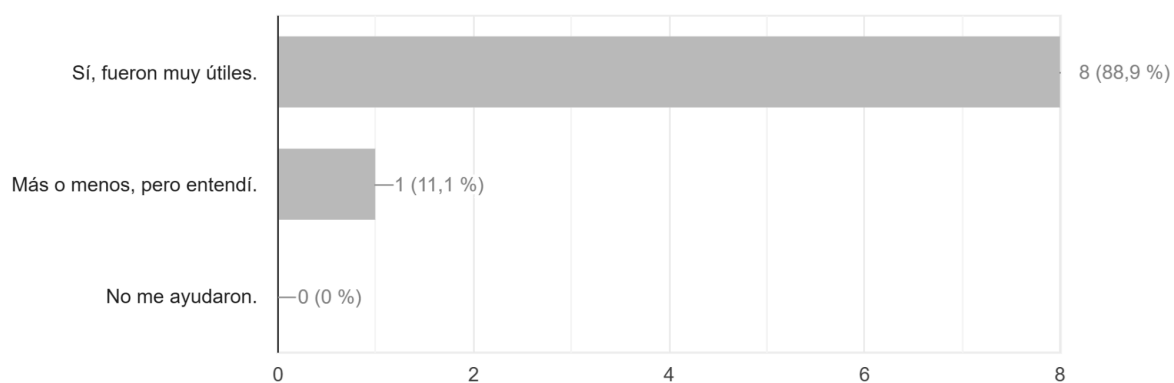
0 de 9 respuestas correctas





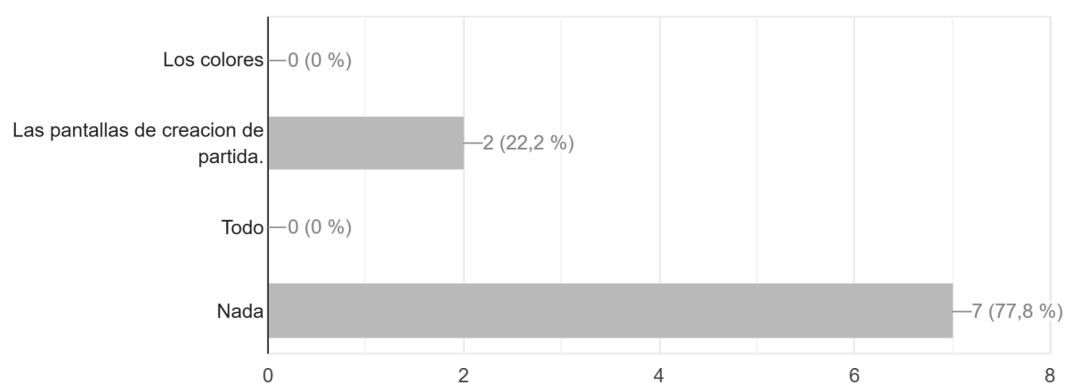
### ¿Los colores utilizados ayudaron a diferenciar las categorías de las preguntas?

0 de 9 respuestas correctas



### ¿Qué mejorarías?

0 de 9 respuestas correctas





## 5. Bibliografía

### Herramientas e Información para el desarrollo del proyecto

Behance: <https://www.behance.net>

Figma: <https://www.figma.com>

Stack Overflow: <https://stackoverflow.com>

YouTube: <https://www.youtube.com>

GitHub: <https://github.com>

## 6. Anexo

### **Enlace de Figma:**

<https://www.figma.com/design/EwPucGmILpRFmElklKmFEv/Trivialist?node-id=0-1&m=dev&t=mfg5NEeSGekoekC2-1>

### **Enlace de canvas:**

[https://www.canva.com/design/DAGdAGvOoK8/YqNw0kbXURDQ1B54Gw5YQA/edit?utm\\_content=DAGdAGvOoK8&utm\\_campaign=designshare&utm\\_medium=link2&utm\\_source=sharebutton](https://www.canva.com/design/DAGdAGvOoK8/YqNw0kbXURDQ1B54Gw5YQA/edit?utm_content=DAGdAGvOoK8&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)