



# **CleverClash**

**Jesus Garcia Calvo**

**Pablo Piedrola Muñoz**

**Jose Manuel Leon Carmona**



# Índice

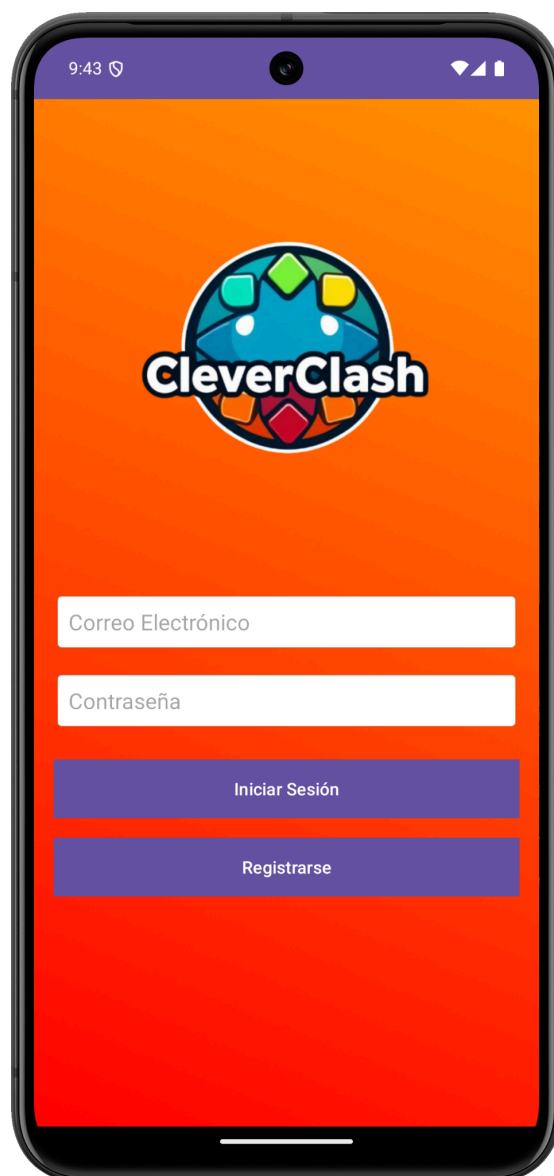
<b>Introducción.....</b>	<b>3</b>
<b>Descripción Básica de las Clases.....</b>	<b>4</b>
Actividades Principales.....	4
Otras Clases Importantes.....	10
Funcionalidad.....	11
Futuros Aspectos a Mejorar.....	11



## Introducción

Clever Clash es una aplicación interactiva que combina entretenimiento y competencia en un entorno gamificado. La plataforma está diseñada para permitir a los jugadores participar en desafiantes preguntas de trivia, organizadas en diversas categorías. Utiliza Firebase para la autenticación de usuarios y un sistema cliente-servidor basado en sockets para gestionar partidas multijugador. La aplicación permite a los usuarios registrarse, iniciar sesión, jugar partidas, girar una ruleta para determinar la categoría de preguntas y mejorar su conocimiento en distintas áreas temáticas.

Clever Clash está diseñado para ser accesible y fácil de jugar, con una interfaz intuitiva que guía a los jugadores a través de las diferentes secciones del juego. Además, el sistema de preguntas y respuestas busca no solo entretener sino también educar a los jugadores en una variedad de temas.



Repositorio GitHub del Proyecto: [CleverClash](#)



## Descripción Básica de las Clases

### Actividades Principales

1. **LoginActivity:** Maneja la autenticación de usuarios mediante Firebase Authentication. Permite a los jugadores iniciar sesión con correo y contraseña, o registrarse en la aplicación si no tienen una cuenta. También se encarga de validar los campos de entrada y mostrar mensajes de error en caso de datos incorrectos.

```
</> public class LoginActivity extends AppCompatActivity {

    private EditText emailEditText; 2 usages
    private EditText passwordEditText; 2 usages
    private Button loginButton; 2 usages
    private Button registerButton; 2 usages

    private FirebaseAuth auth; 2 usages

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Inicializar FirebaseAuth
        FirebaseAuth.initializeApp(context: this);
        auth = FirebaseAuth.getInstance();

        setContentView(R.layout.activity_login);

        // Obtener referencias a los elementos del diseño
        emailEditText = findViewById(R.id.emailEditText);
    }
}
```

```
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/background_2"
    android:padding="16dp">

    <!-- Logo -->
    <ImageView
        android:id="@+id/logoImage"
        android:layout_width="350dp"
        android:layout_height="350dp"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="5dp"
        android:src="@drawable/logo_cleverclash"
    />
```

2. **RegisterActivity:** Facilita el proceso de registro para nuevos usuarios. Recoge los datos de nombre de usuario, correo y contraseña y los almacena en Firebase Authentication. Además, verifica que las contraseñas coincidan y que el correo tenga un formato válido.

```
> </> public class RegisterActivity extends AppCompatActivity {  
  
    private EditText usernameEditText, emailEditText, passwordEditText, confirmPasswordEditText;  
    private Button registerButton; 2 usages  
    private FirebaseAuthManager firebaseAuthManager; 2 usages  
  
    @Override  
    protected void onCreate(@Nullable Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_register);  
  
        try {  
            // Inicializa los componentes de la vista  
            usernameEditText = findViewById(R.id.usernameEditText);  
            emailEditText = findViewById(R.id.emailEditText);  
            passwordEditText = findViewById(R.id.passwordEditText);
```

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="vertical"  
    android:gravity="center_horizontal">  
  
    <!-- Logo -->  
    <ImageView  
        android:id="@+id/logoImage"  
        android:layout_width="250dp"  
        android:layout_height="250dp"  
        android:layout_marginTop="32dp"
```

3. **MainActivity:** Es la pantalla principal que aparece tras el inicio de sesión. Incluye opciones para iniciar una nueva partida, acceder a futuras funcionalidades y cambiar el idioma de la aplicación. También permite a los jugadores editar su perfil y visualizar su historial de juegos.

```
public class MainActivity extends AppCompatActivity {  
  
    private FirebaseAuth auth; 3 usages  
    private Toolbar toolbar; 4 usages  
    private EditText userNameEditText; 4 usages  
    private Switch switchLanguage; 6 usages  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);
```

```
        android:layout_width="match_parent"  
        android:layout_height="match_parent"  
        android:background="@drawable/background_2">  
  
        <!-- Barra superior (Toolbar) -->  
        <androidx.appcompat.widget.Toolbar  
            android:id="@+id/toolbar"  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:background="@color/white"  
            android:elevation="4dp"
```

4. **RuletaActivity:** Contiene la ruleta interactiva, que los jugadores pueden girar para seleccionar aleatoriamente una categoría de preguntas. Utiliza animaciones para mejorar la experiencia de usuario y ofrece un efecto visual atractivo.

```
</> public class RuletaActivity extends AppCompatActivity {  
  
    private RuletaDrawable ruletaDrawable; 3 usages  
    private ImageView ruletaImageView; 2 usages  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_ruleta);  
    }  
}
```

```
android:layout_width="match_parent"  
android:layout_height="match_parent"  
android:padding="16dp">  
  
<ImageView  
    android:id="@+id/ruletaImage"  
    android:layout_width="0dp"  
    android:layout_height="0dp"  
    android:layout_marginTop="32dp"  
    app:layout_constraintDimensionRatio="1:1"  
    app:layout_constraintEnd_toEndOf="parent"
```

5. **EntretenimientoActivity:** Muestra preguntas relacionadas con la categoría "Entretenimiento" y permite al jugador seleccionar una respuesta entre varias opciones. Implementa un temporizador para limitar el tiempo de respuesta y aumentar la dificultad.



```
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class EntretenimientoActivity extends AppCompatActivity {

    private TextView questionBox; 2 usages
    private Button btnGreen, btnYellow, btnBlue, btnRed; 2 usages
    private String userName = "Usuario 1"; // Esto se puede pasar desde la MainActivity con

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_entretenimiento);

        // Configurar el nombre del usuario
        TextView userNameView = findViewById(R.id.userName);
        userNameView.setText(userName);
    }
}
```

```
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#9C5DD6"> <!-- Color de fondo (morado para entretenimiento) -->

    <!-- Barra superior con el nombre del usuario -->
    <LinearLayout
        android:id="@+id/topBar"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:orientation="horizontal"
        android:background="FFFFFF"
        android:padding="10dp">

        <ImageView
            android:id="@+id/userIcon"
            android:layout_width="40dp"
```

6. **SportsActivity:** Similar a EntretenimientoActivity, pero con preguntas específicas sobre deportes. También incluye la posibilidad de recibir pistas o sugerencias en caso de preguntas difíciles.

```
public class SportsActivity extends AppCompatActivity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_sports);  
  
        // Mensaje de prueba  
        Toast.makeText(context, this, text: "Categoria: Deportes", Toast.LENGTH_SHORT).show();  
    }  
}
```

```
android:layout_height="match_parent"  
android:background="#FFA500"> <!-- Color de fondo (naranja) -->  
  
<!-- Barra superior con el nombre del usuario -->  
<LinearLayout  
    android:id="@+id/topBar"  
    android:layout_width="match_parent"  
    android:layout_height="60dp"  
    android:orientation="horizontal"  
    android:background="#FFFFFF"
```

En este caso hemos puesto la de deportes y entretenimiento, pero tenemos un activity específico para cada categoría.

## Otras Clases Importantes

7. **FirebaseAuthManager**: Clase auxiliar que encapsula las operaciones de autenticación de Firebase, facilitando el registro, inicio de sesión y recuperación de información de usuario. También maneja el restablecimiento de contraseñas y la verificación de correo electrónico.
8. **Server**: Actúa como el controlador central de la partida. Gestiona la conexión de los jugadores, administra turnos y determina el flujo del juego. Controla las condiciones de victoria y envía actualizaciones en tiempo real a los jugadores.
9. **Cliente**: Implementa un cliente TCP que se conecta al servidor del juego. Se encarga de recibir turnos, enviar respuestas y procesar mensajes del servidor. Maneja posibles errores de conexión y reconexión en caso de interrupciones.



10. **GestionClientes:** Clase que maneja individualmente a cada cliente conectado al servidor, procesando sus acciones y respuestas. Garantiza que los turnos se respeten y que la información fluya correctamente entre los jugadores y el servidor.

## Funcionalidad

- **Autenticación con Firebase:** Permite a los jugadores registrarse e iniciar sesión de manera segura con validaciones en tiempo real.
- **Gestor de partidas:** Un servidor basado en sockets que administra las conexiones y turnos de los jugadores, garantizando una experiencia fluida.
- **Interfaz interactiva:** Diseño de pantallas con preguntas categorizadas y una ruleta animada, mejorando la experiencia del usuario.
- **Cambio de idioma:** Los usuarios pueden alternar entre inglés y español desde la configuración de la aplicación, almacenando su preferencia de idioma.
- **Sistema de turnos:** Controla la secuencia de juego entre dos jugadores, asegurando la alternancia de respuestas y evitando respuestas simultáneas.
- **Historial de partidas:** Se almacena el rendimiento de cada jugador para que puedan ver su progreso a lo largo del tiempo.
- **Notificaciones en tiempo real:** Se envían alertas cuando un jugador está en su turno o cuando una nueva partida está disponible.

## Futuros Aspectos a Mejorar

- **Mejorar el diseño de interfaz:** Aplicar un diseño más moderno y atractivo con transiciones más fluidas y elementos gráficos mejorados.



- **Ampliar las categorías de preguntas:** Incluir una base de datos más amplia y permitir la generación dinámica de preguntas, así como la adición de nuevas categorías basadas en tendencias y eventos actuales.
- **Optimizar la conexión cliente-servidor:** Mejorar la estabilidad de la comunicación y reducir la latencia para una mejor experiencia de juego en tiempo real.
- **Agregar un sistema de puntuaciones y rankings:** Para fomentar la competencia entre los jugadores mediante un leaderboard global y logros personalizados.
- **Implementar un sistema de avatares:** Permitir que los jugadores personalicen su perfil con imágenes o iconos, aumentando la inmersión en el juego.
- **Integrar autenticación social:** Soporte para inicio de sesión con Google, Facebook o Twitter para facilitar el acceso de nuevos usuarios.
- **Incluir un modo multijugador ampliado:** Que permita partidas con más de dos jugadores y un sistema de eliminación basado en rondas.
- **Sistema de recompensas:** Incorporar monedas virtuales, logros o desbloqueables para mejorar la retención de usuarios y motivarlos a seguir jugando.
- **Modo de entrenamiento:** Un modo sin límite de tiempo en el que los jugadores pueden practicar y aprender sin presiones.
- **Implementación de inteligencia artificial:** Agregar un bot que pueda jugar contra los usuarios en caso de que no haya oponentes disponibles en línea.
- **Soporte para dispositivos adicionales:** Expansión a otras plataformas como tablets o integración con asistentes de voz para una experiencia más accesible.