

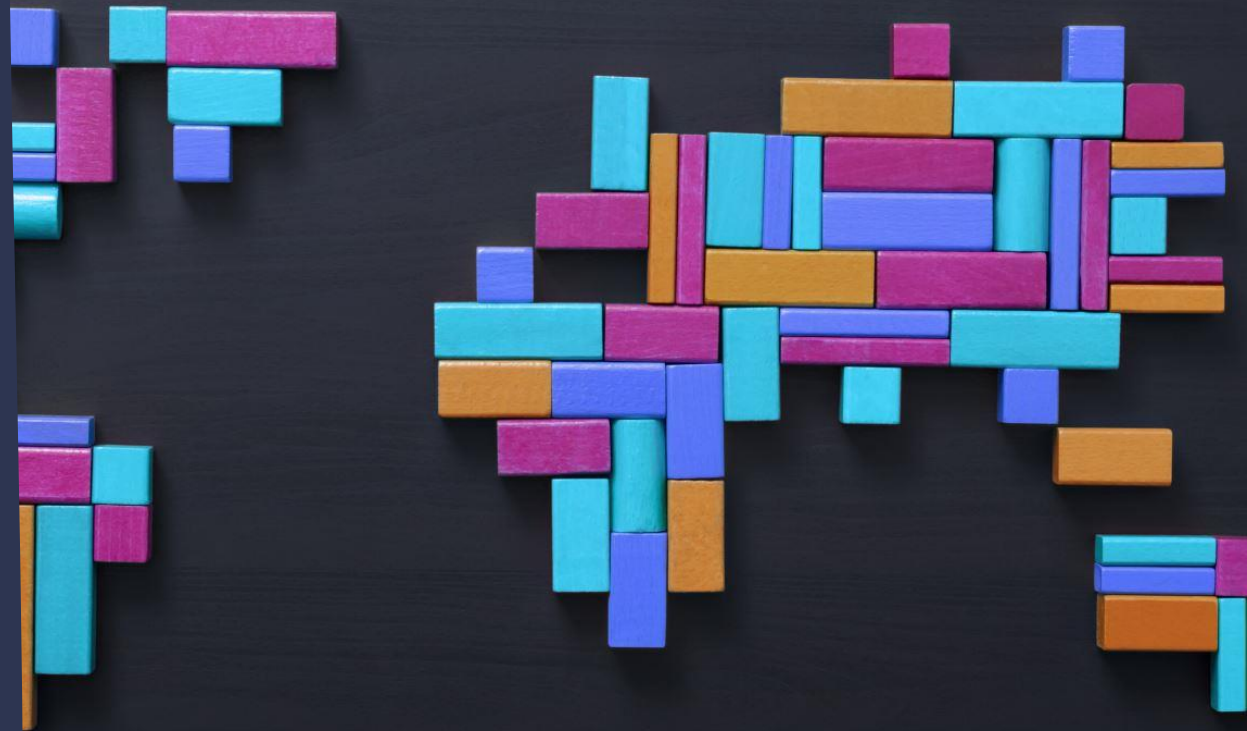
# APLICACIÓN ANDROID PARA GESTIÓN DE ESTADÍSTICAS Y UTILIDADES

Realizado por:

Jesús Manuel Domínguez Tristancho

Dirigido por:

Rafael Paz Vicente



# ÍNDICE



Introducción



Objetivos



Implementación



Planificación



Conclusiones

# ÍNDICE



Introducción



Objetivos



Implementación

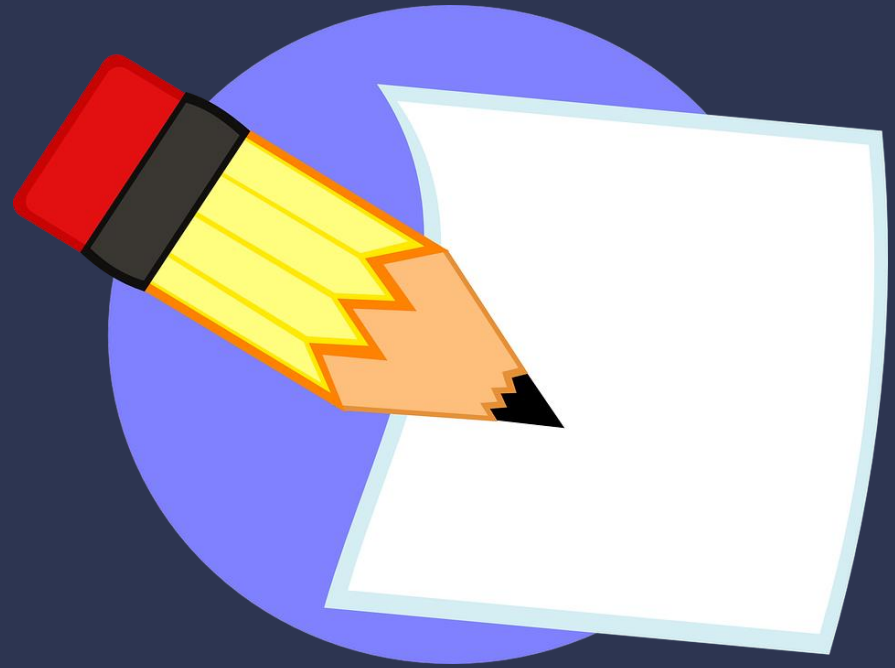


Planificación



Conclusiones

# INTRODUCCIÓN



# JUSTIFICACIÓN



# ÍNDICE



Introducción



Objetivos



Implementación

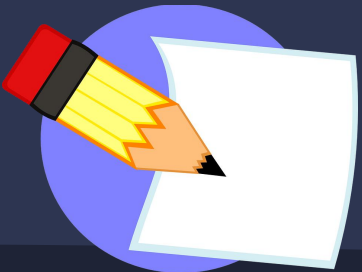
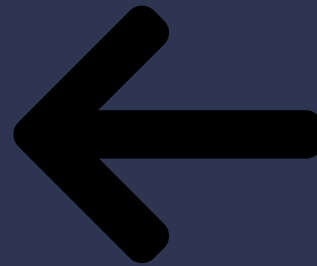
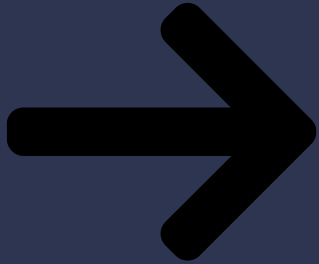


Planificación



Conclusiones

# OBJETIVOS



# OBJETIVOS



Desarrollo aplicación  
Android



Almacenar estadísticas



Compartir datos entre  
dispositivos



Facilitar carga física y  
ahorrar tiempo



Aplicación *offline* para  
utilidades y *online* para  
datos a tiempo real



# ÍNDICE



Introducción



Objetivos



Implementación



Planificación



Conclusiones

# IMPLEMENTACIÓN



Android  
Studio

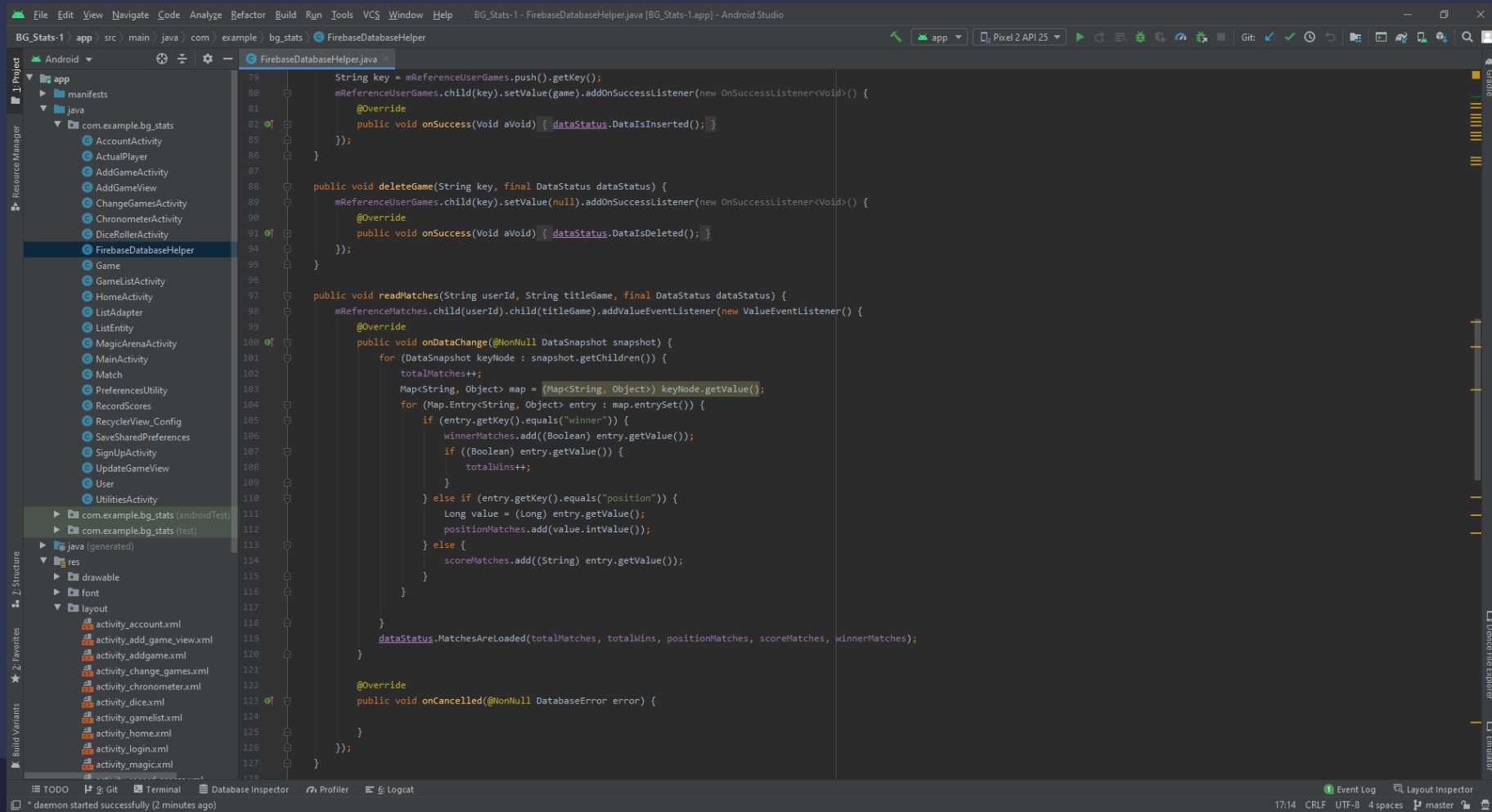


Firebase



Java

# IMPLEMENTACIÓN (Android Studio y Java)



```
79
80 String key = mReferenceUserGames.push().getKey();
81 mReferenceUserGames.child(key).setValue(game).addOnSuccessListener(new OnSuccessListener<Void>() {
82     @Override
83     public void onSuccess(Void aVoid) { dataStatus.DataIsInserted(); }
84 });
85
86
87
88 public void deleteGame(String key, final DataStatus dataStatus) {
89     mReferenceUserGames.child(key).setValue(null).addOnSuccessListener(new OnSuccessListener<Void>() {
90         @Override
91         public void onSuccess(Void aVoid) { dataStatus.DataIsDeleted(); }
92     });
93 }
94
95
96
97 public void readMatches(String userId, String titleGame, final DataStatus dataStatus) {
98     mReferenceMatches.child(userId).child(titleGame).addValueEventListener(new ValueEventListener() {
99         @Override
100         public void onDataChange(@NonNull DataSnapshot snapshot) {
101             for (DataSnapshot keyNode : snapshot.getChildren()) {
102                 totalMatches++;
103                 Map<String, Object> map = (Map<String, Object>) keyNode.getValue();
104                 for (Map.Entry<String, Object> entry : map.entrySet()) {
105                     if (entry.getKey().equals("winner")) {
106                         winnerMatches.add((Boolean) entry.getValue());
107                         if ((Boolean) entry.getValue()) {
108                             totalWins++;
109                         }
110                     } else if (entry.getKey().equals("position")) {
111                         Long value = (Long) entry.getValue();
112                         positionMatches.add(value.intValue());
113                     } else {
114                         scoreMatches.add((String) entry.getValue());
115                     }
116                 }
117             }
118             dataStatus.MatchesAreLoaded(totalMatches, totalWins, positionMatches, scoreMatches, winnerMatches);
119         }
120     });
121
122     @Override
123     public void onCancelled(@NonNull DatabaseError error) {
124
125     }
126
127
128 }
129 }
130 }
131 }
132 }
133 }
134 }
135 }
136 }
137 }
138 }
139 }
140 }
141 }
142 }
143 }
144 }
145 }
146 }
147 }
148 }
149 }
150 }
151 }
152 }
153 }
154 }
155 }
156 }
157 }
158 }
159 }
160 }
161 }
162 }
163 }
164 }
165 }
166 }
167 }
168 }
169 }
170 }
171 }
172 }
173 }
174 }
175 }
176 }
177 }
178 }
179 }
180 }
181 }
182 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
```

# IMPLEMENTACIÓN (Firebase)

The screenshot displays the Firebase Realtime Database interface in a web browser. The left sidebar contains the Firebase logo and navigation links for 'Información general', 'Desarrollo' (with sub-links for Authentication, Cloud Firestore, Realtime Database, Storage, Hosting, Functions, and Machine Learning), 'Calidad', 'Analytics', 'Crecimiento', and 'Extensions'. The main content area is titled 'Realtime Database' and shows the 'bg-stats-hm' database. A breadcrumb trail at the top reads 'BG Stats > Realtime Database'. Below this, there are tabs for 'Datos', 'Reglas', 'Copias de seguridad', and 'Uso'. A message banner states: 'Crea prototipos y pruébalos por completo con la Suite del Emulador Local, que ahora también incluye Firebase Authentication. Empezar'. The database structure is visualized as a tree:

- bg-stats-hm
  - Games
    - Matches
      - LDgjt9YeDuYToJOM9THKvmJBk203
      - OdBdC81KpYXn0QpReg3LE7mewX13
      - WqE2Cc1I6EWYTYBoKEW3q0jo9fk1
      - sqAQj1peQBhSMgJM1OAlDVTw3wz1
        - 10 Days in Africa
        - 7 Wonders Duel
        - Agricola: All Creatures Big and Small
        - Carcassonne
        - Valeria: Card Kingdoms
  - Users
    - lq5G9JKXUEgppXEtQmCuiHZyqsa2
    - LDgjt9YeDuYToJOM9THKvmJBk203
    - OdBdC81KpYXn0QpReg3LE7mewX13
    - WqE2Cc1I6EWYTYBoKEW3q0jo9fk1
    - jZpCijJUgWbwmaJJpWBxknHCFHw2
    - sqAQj1peQBhSMgJM1OAlDVTw3wz1
      - Games
        - email: "juanmanuelmaa@gmail.com"

At the bottom, a location indicator shows 'Ubicación de la base de datos: Estados Unidos (us-central1)'.

# ÍNDICE



Introducción



Objetivos



Implementación

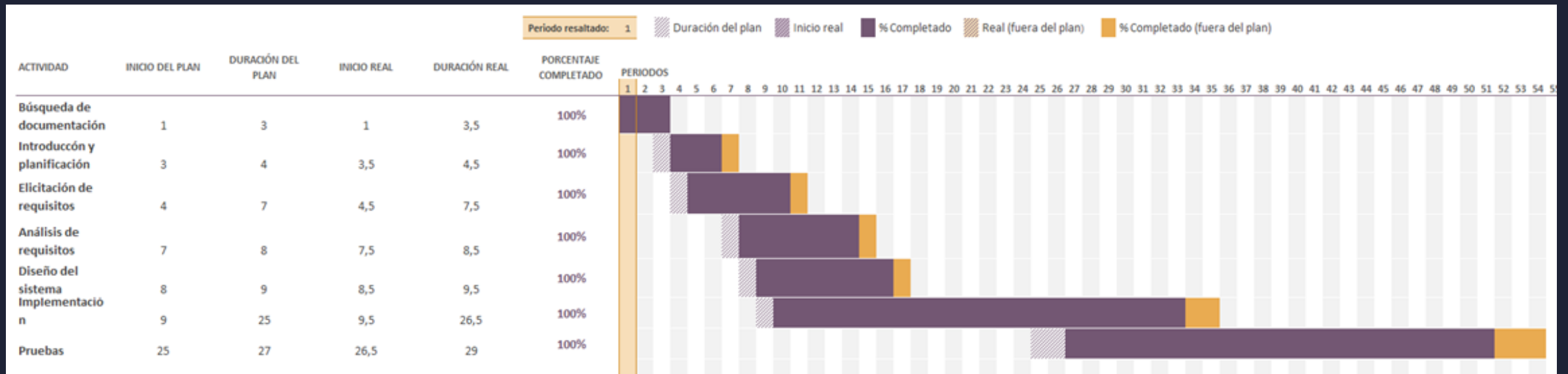


Planificación



Conclusiones

# PLANIFICACIÓN



# ÍNDICE



Introducción



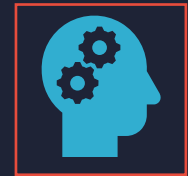
Objetivos



Implementación



Planificación



Conclusiones

# CONCLUSIONES



Estadísticas de  
partidas guardadas en  
una base de datos



Estadísticas  
diferenciadas por  
usuario



Datos compartidos  
entre dispositivos



Herramientas a mano  
del usuario



Aplicación *offline* para  
utilidades y *online* para  
datos a tiempo real



# TRABAJO FUTURO



Aplicación  
completamente  
*offline*



Gráficas para  
puntuaciones



Más conectividad  
entre usuarios



Animaciones y  
sonidos



Lista de juegos  
expandible por  
usuarios



PlayStore



GRACIAS POR VUESTRA ATENCIÓN

