

# O Grande Manual: Implementação do NeuroRace (Replit + Firebase)

## Sua Arquitetura:

- **Armazenamento (O "Armário"):** Firebase Firestore (Plano Gratuito).
  - **Hospedagem (A "Vitrine"):** Replit (Plano Gratuito).
  - **Identificador Único:** Email do jogador (A Solução Híbrida).
- 



## Parte 1: O Alicerce (Configurar o Firebase)

(Faça isso primeiro. Você não pode construir o site sem o banco de dados.)

**Objetivo:** Criar o banco de dados (Firestore) e adicionar dados de teste para que seu site tenha o que ler.

1. **Acesse o Firebase Console:** Abra seu projeto neurorace-app.
2. **Vá para o Firestore:** No menu, clique em **Firestore Database**.
3. **Crie a Coleção de Jogadores:**
  - o Clique em **+ Iniciar coleção**.
  - o **ID da coleção:** players
4. **Crie o Jogador de Teste (Solução Híbrida):**
  - o **ID do documento:** teste@email.com (Use um email real de teste).
  - o Clique em **Adicionar campo**:
    - Campo: email | Tipo: string | Valor: teste@email.com
    - Campo: phone | Tipo: string | Valor: (11) 99999-8888
  - o Clique em **Salvar**.
5. **Crie a Subcoleção sessions (Corridas):**
  - o Com o documento teste@email.com selecionado, clique em **+ Iniciar coleção** (na coluna da direita).
  - o **ID da coleção:** sessions
  - o **ID do documento:** sessao\_teste\_1
  - o **Adicione os Campos da Sessão (do PDF):**
    - startedAt (Timestamp) -> escolha qualquer data
    - tzf (Number) -> 0.75
    - cvfLabel (String) -> estavel
    - ircLabel (String) -> alto

- IfoMean (Number) -> 3.5 (a nova métrica LFO)<sup>1</sup>
- badges (Map) -> Adicione um campo isTZFPB (Boolean) -> true<sup>2</sup>

## 6. Crie a Subcoleção stats (Histórico):

- Clique no documento teste@email.com novamente.
- Clique em + Iniciar coleção (na coluna da direita).
- **ID da coleção:** stats
- **ID do documento:** stats (use a palavra "stats" como ID).
- **Adicione os Campos de Histórico (do PDF):**
  - tzfPB (Number) -> 0.85<sup>3</sup>
  - trzPBsec (Number) -> 2.1<sup>4</sup>
  - tzfSeries (Array) -> Adicione os números 0.6, 0.75, 0.85<sup>5</sup>

## 7. Tranque o Banco (Segurança):

- No Firestore, clique na aba **Regras**.
- Substitua o código por este (permite que qualquer pessoa leia, mas ninguém escreva):

```
JavaScript
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    // Permite que QUALQUER pessoa (seu site) LEIA os dados.
    match /{document=**} {
      allow read: if true;
      allow write: if false; // Bloqueia a escrita (só o back-end deve escrever)
    }
  }
}
```

- Clique em **Publicar**.

**Status:** O "Armário" (Firebase) está pronto e com dados de teste.

---

## Parte 2: A Construção (Configurar o Replit)

**Objetivo:** Criar o projeto do site no Replit e organizar todos os arquivos.

### 1. Crie o Projeto Replit:

- Vá ao Replit, clique em **+ Create Repl**.
  - Template: **HTML, CSS, JS**.
  - Nome: neuro-race-site (ou o que preferir).
2. Crie a Estrutura de Pastas e Arquivos:

Na barra lateral "Files" do Replit, crie esta estrutura:

```
neuro-race-site/
├── index.html      (Home/Ideia do Projeto)
├── ranking.html    (Ranking Geral)
├── equipe.html     (Perfis LinkedIn/GitHub)
├── dashboard.html  (O "Login" e Análise Pessoal)
├── race-finished.html (Tela da TV pós-corrida)
└── votacao.html    (Redirect para o NEXT)

├── css/
│   └── style.css    (Seu CSS principal)

├── js/
│   ├── firebase-config.js (A chave de conexão)
│   ├── main.js        (Lógica do menu, se houver)
│   ├── ranking.js    (Script do ranking)
│   ├── dashboard.js  (Script da análise pessoal)
│   └── race-finished.js (Script da tela da TV)

└── images/
    └── (Aqui vão os QR codes estáticos, ex: localizacao.png)
```

**Status:** O "esqueleto" do seu site está pronto no Replit.

---

## **Parte 3: O Visual (HTML e CSS da Identidade Visual)**

**Objetivo:** Implementar o design (do style.css) e o menu de navegação em todas as páginas.

### **1. O Template Mestre (Base para todas as páginas)**

Este é o HTML base com o **menu suspenso/lateral** (que na verdade é um menu superior fixo,

o que é melhor para a navegação) e as novas seções. Copie isso para o seu index.html.

## Arquivo: index.html

HTML

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>NeuroRace - O Projeto</title>
    <link rel="stylesheet" href="css/style.css">
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;700&display=swap" rel="stylesheet">
</head>
<body>
    <nav class="menu-principal">
        <a href="index.html" class="menu-logo">NeuroRace</a>
        <ul>
            <li><a href="index.html">O Projeto</a></li>
            <li><a href="ranking.html">Ranking</a></li>
            <li><a href="equipe.html">Equipe</a></li>
            <li><a href="dashboard.html">Meu Dashboard</a></li>
            <li><a href="votacao.html" class="menu-cta" target="_blank">VOTE NO NEXT!</a></li>
        </ul>
    </nav>

    <main class="container">
        <header>
            <h1 class="title">Neurociência, Foco e Performance</h1>
            <p class="subtitle">Entenda a ideia do projeto e como o NeuroRace mede sua concentração.</p>
        </header>

        <div class="results-grid">
            <div class="card">
                <h3>🧠 Insights sobre Foco</h3>
```

```

<p>A capacidade de manter o foco é uma habilidade que pode ser treinada. Nossa sistema usa neurofeedback para te ajudar a identificar e fortalecer seus momentos de concentração.</p>
</div>
<div class="card">
  <h3>🏆 Premiação</h3>
  <p>Participe de uma competição no NeuroRace! Os pilotos com os melhores índices de performance concorrem a prêmios exclusivos no final do evento.</p>
</div>
</div>

<div class="card full-width">
  <h3>Publique sua Experiência!</h3>
  <p>Visitou nosso stand? Poste sua foto ou sua análise de performance no LinkedIn e ajude a gerar relevância para o projeto. Use as hashtags:</p>
  <div class="hashtag-container">
    <code class="hashtag">#NeuroRaceFIAP</code>
    <code class="hashtag">#NEXT2025</code>
    <code class="hashtag">#Neurofeedback</code>
  </div>
</div>
</main>
<script src="https://www.gstatic.com/firebasejs/9.6.7.firebaseio-app-compat.js"></script>
<script src="https://www.gstatic.com/firebasejs/9.6.7.firebaseio-firebase-compat.js"></script>
<script src="js/firebase-config.js"></script>
<script src="js/main.js"></script> </body>
</html>

```

**Ação:** Copie esta estrutura (o `<body>` inteiro) para ranking.html, equipe.html, e dashboard.html, alterando apenas o `<title>` e o conteúdo dentro de `<main class="container">`.

## 2. O Arquivo de Estilos Completo

Copie todo este código e cole no seu arquivo `css/style.css`. Ele já inclui os estilos do seu arquivo original (`style.css`), os estilos do menu, tooltips, hashtags, e tudo mais.

**Arquivo: `css/style.css`**

## CSS

```
/* --- ESTILOS GERAIS (do seu arquivo) --- */
body {
    background-color: #0d1117;
    color: #c9d1d9;
    font-family: 'Poppins', sans-serif;
    margin: 0;
    padding: 0;
}
.container {
    width: 100%;
    max-width: 900px; /* Aumentado para o layout lado-a-lado */
    margin: 20px auto;
    padding: 0 20px;
}
header {
    text-align: center;
    margin-bottom: 50px;
}
.title {
    font-size: 3.5rem;
    font-weight: 700;
    margin-bottom: 10px;
    background: linear-gradient(90deg, #bf46f3, #4c66f4);
    -webkit-background-clip: text;
    -webkit-text-fill-color: transparent;
    background-clip: text;
    text-fill-color: transparent;
}
.subtitle {
    font-size: 1.1rem;
    color: #8b949e;
}

/* --- ESTILO DO MENU DE NAVEGAÇÃO --- */
.menu-principal {
    display: flex;
    justify-content: space-between;
    align-items: center;
    padding: 0 40px;
```

```
background-color: #161b22;
border-bottom: 1px solid #30363d;
height: 70px;
position: sticky;
top: 0;
z-index: 1000;
}

.menu-principal .menu-logo {
    font-size: 1.5rem;
    font-weight: 700;
    color: #f0f6fc;
    text-decoration: none;
}

.menu-principal ul {
    display: flex;
    list-style: none;
    gap: 25px;
}

.menu-principal ul a {
    text-decoration: none;
    color: #8b949e;
    font-weight: 500;
    transition: color 0.2s;
}

.menu-principal ul a:hover {
    color: #f0f6fc;
}

.menu-principal .menu-cta {
    background: linear-gradient(90deg, #bf46f3, #4c66f4);
    color: white;
    padding: 8px 16px;
    border-radius: 6px;
}

.menu-principal .menu-cta:hover {
    color: white;
    opacity: 0.9;
}

/*
--- ESTILOS DO RANKING (do seu arquivo) --- */
#ranking-list {
    display: flex;
    flex-direction: column;
    gap: 15px;
}
```

```
}

.rank-item {
    background-color: #161b22;
    border: 1px solid #30363d;
    border-radius: 8px;
    padding: 16px 24px;
    display: flex;
    justify-content: space-between;
    align-items: center;
    transition: border-color 0.2s ease-in-out;
}

.rank-item.first-place {
    border-color: #f1c40f;
    box-shadow: 0 0 15px rgba(241, 196, 15, 0.2);
}

.rank-info-group {
    display: flex;
    align-items: center;
    gap: 25px;
}

.rank-position {
    font-size: 2rem;
    font-weight: 700;
    color: #8b949e;
    width: 40px;
    text-align: center;
    display: flex;
    flex-direction: column;
    align-items: center;
}

.rank-position .crown {
    font-size: 1.5rem;
    color: #f1c40f;
    margin-bottom: -10px;
}

.player-info {
    display: flex;
    flex-direction: column;
}

.player-name {
    font-size: 1.25rem;
    font-weight: 700;
    color: #f0f6fc;
```

```
}

.player-attention {
    font-size: 0.9rem;
    color: #8b949e;
}

.player-score {
    font-size: 2rem;
    font-weight: 700;
    color: #38bdf8;
}

/* --- ESTILOS RACE-FINISHED (do seu arquivo) --- */

.finished-container {
    max-width: 900px;
    margin: 20px auto;
}

.players-container {
    display: flex;
    justify-content: center;
    gap: 40px;
    flex-wrap: wrap;
    margin-top: 40px;
}

.player-card {
    background-color: #161b22;
    border: 1px solid #30363d;
    border-radius: 12px;
    padding: 30px;
    width: 320px;
    display: flex;
    flex-direction: column;
    align-items: center;
    text-align: center;
    border-top-width: 5px;
}

#player-1-card { border-top-color: #38bdf8; }
#player-2-card { border-top-color: #f43f82; }

.final-score {
    font-size: 1.1rem;
    color: #8b949e;
    margin-top: 0;
    margin-bottom: 25px;
}
```

```
.qrcode {  
    background-color: white;  
    padding: 10px;  
    border-radius: 8px;  
}  
.qrcode img {  
    display: block;  
    width: 200px !important;  
    height: 200px !important;  
}  
  
/* --- ESTILOS DASHBOARD/RESULTS (do manual anterior) --- */  
.results-grid {  
    display: grid;  
    grid-template-columns: 1fr 1fr;  
    gap: 20px;  
}  
.card {  
    background-color: #161b22;  
    border: 1px solid #30363d;  
    border-radius: 8px;  
    padding: 24px;  
    margin-bottom: 20px; /* Adicionado para espaçamento */  
}  
.full-width {  
    grid-column: 1 / -1;  
}  
.chart-container {  
    position: relative;  
    height: 300px;  
}  
 @media (max-width: 768px) {  
    .results-grid {  
        grid-template-columns: 1fr;  
    }  
}  
.email-form {  
    display: flex;  
    gap: 10px;  
    margin: 30px 0 15px 0;  
}  
.email-form input {  
    flex-grow: 1;
```

```
padding: 12px;
font-size: 1rem;
background-color: #0d1117;
border: 1px solid #30363d;
border-radius: 6px;
color: #f0f6fc;
}

.email-form button {
    padding: 12px 20px;
    font-size: 1rem;
    font-weight: 700;
    color: #f0f6fc;
    background: linear-gradient(90deg, #bf46f3, #4c66f4);
    border: none;
    border-radius: 6px;
    cursor: pointer;
}

@media (max-width: 768px) {
    .email-form {
        flex-direction: column;
    }
}

.incentive-box {
    background-color: #161b22;
    border-left: 5px solid #bf46f3;
    border-radius: 0 4px 4px 0;
    padding: 16px;
    margin: 25px 0;
    font-size: 0.95em;
    color: #c9d1d9;
}

.incentive-box strong {
    color: #f0f6fc;
}

.tooltip {
    position: relative;
    display: inline-block;
    cursor: help;
    margin-left: 8px;
    border: 1px solid #8b949e;
    border-radius: 50%;
    width: 16px;
    height: 16px;
```

```
    font-size: 11px;
    font-weight: bold;
    line-height: 16px;
    text-align: center;
    color: #8b949e;
}
.tooltip .tooltip-text {
    visibility: hidden;
    opacity: 0;
    transition: opacity 0.3s;
    width: 250px;
    background-color: #f0f6fc;
    color: #0d1117;
    text-align: left;
    font-weight: normal;
    border-radius: 6px;
    padding: 12px;
    box-shadow: 0 4px 10px rgba(0,0,0,0.3);
    position: absolute;
    z-index: 10;
    bottom: 150%;
    left: 50%;
    margin-left: -125px;
}
.tooltip .tooltip-text strong {
    display: block;
    margin-bottom: 5px;
    color: #0d1117;
}
.tooltip:hover .tooltip-text {
    visibility: visible;
    opacity: 1;
}
.secao-reacao {
    border-top: 1px solid #30363d;
    padding: 10px 0;
}
.secao-reacao:first-child { border-top: none; }
.secao-reacao h4 { margin: 0 0 8px 0; color: #f0f6fc; }
.contagem { font-weight: normal; font-size: 0.8em; color: #8b949e; }
.metrica-reacao { display: flex; justify-content: space-between; font-size: 0.9em; margin-bottom: 5px; }
.label-reacao { color: #8b949e; }
```

```

.valor-reacao { font-weight: bold; }
.valor-positivo { color: #238636; }
.valor-negativo { color: #da3633; }
.descricao-fadiga { font-size: 0.9em; color: #8b949e; margin-top: 0; }
.display-fadiga { display: flex; align-items: center; justify-content: center; margin: 16px 0; }
.icone-fadiga { font-size: 2.5em; margin-right: 12px; }
.label-fadiga { font-size: 1.5em; font-weight: bold; }
.detalhe-fadiga { font-size: 0.8em; color: #8b949e; text-align: center; }
.fadiga-crescente { color: #da3633; }
.fadiga-estavel { color: #8b949e; }
.fadiga-decrescente { color: #238636; }

/* --- NOVOS ESTILOS (Hashtag, etc) --- */
.hashtag-container {
  display: flex;
  gap: 10px;
  flex-wrap: wrap;
  margin-top: 15px;
}
.hashtag {
  background-color: #30363d;
  color: #c9d1d9;
  padding: 5px 10px;
  border-radius: 4px;
  font-family: monospace;
}

```

**Status:** A identidade visual está pronta. Todas as páginas terão o mesmo menu e estilo.

---

## ⚡ Parte 4: A Lógica (JavaScript e Conexões)

**Objetivo:** Fazer os botões funcionarem e os dados aparecerem.

### 1. Conexão do Firebase

Copie sua chave de API do Firebase Console (Configurações do Projeto > Seus apps > App

Web).

### Arquivo: js/firebase-config.js

JavaScript

```
// ATENÇÃO: Substitua pelos valores reais do seu projeto
const firebaseConfig = {
  apiKey: "SUA_API_KEY",
  authDomain: "neurorace-app.firebaseio.com",
  projectId: "neurorace-app",
  storageBucket: "neurorace-app.appspot.com",
  messagingSenderId: "...",
  appId: "..."
};
```

## 2. Página da Equipe (equipe.html)

Use o "Template Mestre" da Parte 3 e preencha o <main>:

### Arquivo: equipe.html

HTML

```
<title>NeuroRace - A Equipe</title>
<body>
  <nav class="menu-principal">...</nav> <main class="container">
    <header>
      <h1 class="title">Nossa Equipe</h1>
      <p class="subtitle">Conheça os desenvolvedores por trás do projeto.</p>
    </header>

    <div class="card">
      <h3>João Victor Moreira (RM 563608)</h3>
```

```
<p>Scrum Master | Firebase e Front-end</p>
<a href="https://www.linkedin.com/in/seu-linkedin" class="menu-cta" target="_blank">LinkedIn</a>
<a href="https://github.com/tavares-fiap/NeuroRace" class="menu-cta"
target="_blank">GitHub</a>
</div>
</main>
</body>
</html>
```

### 3. Página de Votação (votacao.html)

Este arquivo é apenas um redirecionamento.

**Arquivo:** votacao.html

HTML

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
    <meta charset="UTF-8">
    <title>Votação - Redirecionando...</title>
    <meta http-equiv="refresh" content="0; url=https://www.fiap.com.br/next/">
</head>
<body>
    <p>Redirecionando para a página de votação do NEXT...</p>
</body>
</html>
```

### 4. Página do Ranking (ranking.html e js/ranking.js)

**Arquivo:** ranking.html

## HTML

```
<title>NeuroRace - Ranking</title>
<body>
  <nav class="menu-principal">...</nav> <main class="container">
    <header>
      <h1 class="title">Ranking Geral</h1>
      <p class="subtitle">Veja os pilotos mais focados do evento em tempo real.</p>
    </header>
    <div id="ranking-list">
    </div>
  </main>

  <script src="https://www.gstatic.com/firebasejs/9.6.7.firebaseio-app-compat.js"></script>
  <script src="https://www.gstatic.com/firebasejs/9.6.7.firebaseio-firebase-compat.js"></script>
  <script src="js/firebase-config.js"></script>
  <script src="js/ranking.js"></script> </body>
</html>
```

## Arquivo: js/ranking.js

## JavaScript

```
// Este script é executado na tela ranking.html
document.addEventListener("DOMContentLoaded", () => {
  firebase.initializeApp(firebaseConfig);
  const db = firebase.firestore();
  carregarRanking(db);
});

async function carregarRanking(db) {
  try {
    const rankingList = document.getElementById("ranking-list");
    rankingList.innerHTML = "<p>Carregando ranking...</p>";

    // 1. Busca os dados de 'stats' de todos os jogadores
    const snapshot = await db.collectionGroup("stats")
      .orderBy("tzfPB", "desc") // Ordena pelo recorde de foco
```

```

        .limit(10) // Pega os 10 melhores
        .get();

    if (snapshot.empty) {
        rankingList.innerHTML = "<p>Nenhum dado de ranking encontrado.</p>";
        return;
    }

    rankingList.innerHTML = ""; // Limpa o "Carregando"
    let posicao = 1;
    snapshot.forEach(doc => {
        const dados = doc.data();
        const itemHTML = criarItemRanking(posicao, dados);
        rankingList.innerHTML += itemHTML;
        posicao++;
    });

} catch (error) {
    console.error("Erro ao carregar ranking:", error);
    rankingList.innerHTML = "<p>Erro ao carregar ranking.</p>";
}
}

function criarItemRanking(posicao, dados) {
    // Tenta pegar o email do campo 'email' ou do ID do documento pai
    const playerId = dados.playerId || dados.email || doc.ref.parent.parent.id;
    const mediaAtencao = (dados.tzfPB * 100).toFixed(1) || "0";

    const firstPlaceClass = (posicao === 1) ? "first-place" : "";
    const crown = (posicao === 1) ? '<span class="crown">👑</span>' : "";
    const score = mediaAtencao;

    return `
        <div class="ranking-item ${firstPlaceClass}">
            <div class="rank-info-group">
                <div class="rank-position">
                    ${crown}
                    ${posicao}
                </div>
                <div class="player-info">
                    <span class="player-name">${playerId}</span>
                    <span class="player-attention">Recorde de Foco: ${mediaAtencao}%</span>
                </div>
            </div>
        </div>
    `;
}

```

```
        <div class="player-score">${score}%</div>
    </div>
    ;
}
```

## 5. Página do Dashboard (Login e Análise)

Este é o coração do seu site. Ele lê o email da URL (se vier do QR Code) ou do campo de digitação.

**Arquivo: dashboard.html**

HTML

```
<title>NeuroRace - Meu Dashboard</title>
<body>
    <nav class="menu-principal">...</nav> <main class="container">
        <header>
            <h1 class="title">Meu Dashboard Pessoal</h1>
            <p class="subtitle" id="dashboard-subtitle">Digite seu email para carregar sua análise de desempenho.</p>

            <div class="email-form">
                <input type="email" id="email-input" placeholder="Digite seu email de cadastro">
                <button id="search-button">Buscar Análise</button>
            </div>
            <p id="loading-message" style="display: none;">Carregando dados...</p>
        </header>

        <main id="dashboard-content" class="results-grid" style="display: none;">

            <blockquote class="incentive-box full-width">
                 <strong>Curiosidade:</strong> O cérebro é como um músculo. Quanto mais você treina seu foco, mais forte e resiliente ele fica!
            </blockquote>

            <div class="card">
```

```

<h3>Resumo da Última Sessão</h3>
<div class="metric-item">
  <span class="label">⌚ Foco (TZF)
    <span class="tooltip">(?)<span class="tooltip-text"><strong>Tempo em Zona de Foco (TZF):</strong> Mede o percentual da corrida que você passou acima do seu limiar de foco.</span></span>
  </span>
  <span class="value" id="resumo-tzf">--%</span>
</div>
<div class="metric-item">
  <span class="label">⚖️ Consistência (CVF)
    <span class="tooltip">(?)<span class="tooltip-text"><strong>Consistência do Foco (CVF):</strong> Indica se seu foco foi 'Estável' ou se 'Oscilou' muito durante a prova.</span></span>
  </span>
  <span class="value" id="resumo-cvf">--</span>
</div>
<div class="metric-item">
  <span class="label">🧠 Resiliência (IRC)
    <span class="tooltip">(?)<span class="tooltip-text"><strong>Índice de Resiliência Cognitiva (IRC):</strong> Calcula sua capacidade de se recuperar mentalmente de eventos, como colisões.</span></span>
  </span>
  <span class="value" id="resumo-irc">--</span>
</div>
</div>

<div id="card-recordes" class="card">
  <h3>Meus Recordes (PBs)</h3>
  <div class="metric-item"><span class="label">🚀 Melhor Foco (TZF)</span><span class="value" id="recorde-tzf">--</span></div>
  <div class="metric-item"><span class="label">⌚ Maior Sequência de Foco</span><span class="value" id="recorde-streak">--</span></div>
  <div class="metric-item"><span class="label">⚡ Melhor Recuperação (TRZ)</span><span class="value" id="recorde-trz">--</span></div>
</div>

<div class="card full-width">
  <h3>Evolução do Foco (TZF)</h3>
  <div class="chart-container">
    <canvas id="performance-chart"></canvas>
  </div>
</div>

```

```

<div id="card-analise-reacao" class="card">
    <h3>Análise de Reação Pós-Evento</h3>
    <div class="secao-reacao">
        <h4>Após Colisões 💥 <span id="colisao-n" class="contagem">(n=--)</span></h4>
        <div class="metrica-reacao"><span class="label-reacao">Variação da
Atenção:</span><span id="reacao-att-colisao" class="valor-reacao">---</span></div>
        <div class="metrica-reacao"><span class="label-reacao">Variação da
Calma:</span><span id="reacao-med-colisao" class="valor-reacao">---</span></div>
        <div class="metrica-reacao"><span class="label-reacao">⚡ **Recuperação
(LFO):**</span><span id="reacao-lfo-colisao" class="valor-reacao">---</span></div>
        </div>
        <div class="secao-reacao">
            <h4>Após Ultrapassagens 🚗 <span id="ultrapassagem-n"
class="contagem">(n=--)</span></h4>
            <div class="metrica-reacao"><span class="label-reacao">Variação da
Atenção:</span><span id="reacao-att-overtake" class="valor-reacao">---</span></div>
            <div class="metrica-reacao"><span class="label-reacao">Variação da
Calma:</span><span id="reacao-med-overtake" class="valor-reacao">---</span></div>
            </div>
        </div>

        <div id="card-fadiga" class="card">
            <h3>Análise de Fadiga</h3>
            </div>
        </main>
    </main>

    <script src="https://www.gstatic.com/firebasejs/9.6.7.firebaseio-app-compat.js"></script>
    <script src="https://www.gstatic.com/firebasejs/9.6.7.firebaseio-firebase-compat.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script> <script
src="js/firebase-config.js"></script>
    <script src="js/dashboard.js"></script> </body>
</html>

```

## Arquivo: js/dashboard.js

JavaScript

```
// Este script é executado na tela dashboard.html
```

```

document.addEventListener("DOMContentLoaded", () => {
  firebase.initializeApp(firebaseConfig);
  const db = firebase.firestore();

  const searchButton = document.getElementById("search-button");
  const emailInput = document.getElementById("email-input");

  // LÓGICA DO QR CODE: Verifica se o email veio na URL
  const urlParams = new URLSearchParams(window.location.search);
  const emailFromURL = urlParams.get('pid') || urlParams.get('email');

  if (emailFromURL) {
    // Se o email veio do QR code, esconde o formulário e busca direto
    document.querySelector('.email-form').style.display = 'none';
    document.getElementById('dashboard-subtitle').innerText = "Carregando sua análise...";
    handleEmailSearch(db, emailFromURL);
  } else {
    // Se não, espera o usuário digitar e clicar
    searchButton.addEventListener("click", () => {
      const email = emailInput.value.trim().toLowerCase();
      if (email) {
        handleEmailSearch(db, email);
      }
    });
  }
});

/**
 * Função principal que busca os dados do jogador pelo email.
 */
async function handleEmailSearch(db, email) {
  const loadingMessage = document.getElementById("loading-message");
  const dashboardContent = document.getElementById("dashboard-content");
  const subtitle = document.getElementById("dashboard-subtitle");

  loadingMessage.style.display = 'block';
  dashboardContent.style.display = 'none';
  subtitle.innerText = "Buscando seus dados...";

  try {
    const playerRef = db.collection("players").doc(email);
    const playerDoc = await playerRef.get();
  }
}

```

```

if (!playerDoc.exists) {
    loadingMessage.style.display = 'none';
    subtitle.innerText = `Nenhum dado encontrado para "${email}". Verifique o email.`;
    return;
}

// Busca a ÚLTIMA sessão de corrida
const sessionsRef = playerRef.collection("sessions").orderBy("startedAt", "desc").limit(1);
const sessaoSnap = await sessionsRef.get();

// Busca o documento de ESTATÍSTICAS
const statsRef = playerRef.collection("stats").doc("stats");
const statsSnap = await statsRef.get();

if (sessaoSnap.empty) {
    loadingMessage.style.display = 'none';
    subtitle.innerText = `Bem-vindo, ${email}! Nenhuma corrida encontrada no seu histórico.`;
    return;
}

// SUCESSO! TEMOS OS DADOS
const dadosSessao = sessaoSnap.docs[0].data();
const dadosStats = statsSnap.exists ? statsSnap.data() : null;

// Preenche a página
preencherCardResumo(dadosSessao);
preencherAnaliseEventos(dadosSessao);
// preencherAnaliseFadiga(dadosSessao); // (Cole a função do manual anterior)

if (dadosStats) {
    criarGraficoEvolucao(dadosStats);
    preencherQuadroRecordes(dadosStats);
} else {
    document.getElementById("card-recordes").style.display = 'none';
    document.querySelector(".full-width").style.display = 'none';
}

// Mostra o dashboard
loadingMessage.style.display = 'none';
subtitle.innerText = `Exibindo a análise de ${email}`;
dashboardContent.style.display = 'grid';

} catch (error) {

```

```
        console.error("Erro ao buscar dados do jogador:", error);
        loadingMessage.style.display = 'none';
        subtitle.innerText = "Erro ao conectar. Tente novamente.";
    }
}

// =====
// == COLE AQUI AS FUNÇÕES: ==
// 1. preencherCardResumo(dadosSessao)
// 2. preencherQuadroRecordes(dadosStats)
// 3. criarGraficoEvolucao(dadosStats)
// 4. preencherAnaliseEventos(dadosSessao)
// 5. preencherAnaliseFadiga(dadosSessao)
// 6. formatarDelta(valor, elemento)
//
// (Elas estão todas no guia anterior)
// =====
```

## 6. Página da TV (A "Ponte")

Esta página gera o QR code dinâmico.

**Arquivo: race-finished.html**

HTML

```
<script src="https://cdn.jsdelivr.net/npm/qrcodejs@1.0.0/qrcode.min.js"></script>
<script src="js/race-finished.js"></script> </body>
</html>
```

**Arquivo: js/race-finished.js**

JavaScript

```
// Este script é executado na tela race-finished.html
```

```

document.addEventListener("DOMContentLoaded", () => {
  firebase.initializeApp(firebaseConfig);
  const db = firebase.firestore();

  // 1. Pega os IDs da URL
  const urlParams = new URLSearchParams(window.location.search);
  const p1_Id_Email = urlParams.get('p1id'); // O ID agora é o email
  const p1_Sid = urlParams.get('p1sid');
  // ... (lógica para p2) ...

  // 2. Processa o Jogador 1
  if (p1_Id_Email && p1_Sid) {
    // Busca os dados da sessão (TZF, etc.) e preenche o card
    // preencherCardJogador(db, p1_Id_Email, p1_Sid, 1); // (Cole a função)

    // GERA O QR CODE DINÂMICO
    gerarQRCodeDinamico(p1_Id_Email, p1_Sid, 1);
  }
  // ... (faça o mesmo para o Jogador 2) ...
});

// (Cole aqui a função preencherCardJogador do manual anterior)

function gerarQRCodeDinamico(playerIdEmail, sessionId, playerNum) {
  const qrElement = document.getElementById(`qr-code-player${playerNum}`);
  qrElement.innerHTML = ""; // Limpa qualquer imagem estática

  // ATENÇÃO: Substitua 'neuro-race-site.replit.dev' pelo link real do seu Replit
  const urlDestino =
`https://neuro-race-site.replit.dev/dashboard.html?pid=${playerIdEmail}&sid=${sessionId}`;

  new QRCode(qrElement, {
    text: urlDestino,
    width: 200,
    height: 200,
    colorDark: "#0d1117", // Cor do seu tema
    colorLight: "#ffffff",
    correctLevel: QRCode.CorrectLevel.H
  });

  // Atualiza o texto abaixo do nome
  document.getElementById(`player-${playerNum}-score`).innerText = "Escaneie para ver sua análise!";
}

```

}

---

## Parte 5: Teste e Deploy (Go-Live)

### 1. Configure o Replit para Rodar:

- o No Replit, clique nos 3 pontos > "Show hidden files".
- o Abra o arquivo .replit.
- o Garanta que a linha entrypoint seja o seu index.html.

```
entrypoint = "index.html"  
run = "npm run dev"
```

(O Replit com "HTML, CSS, JS" geralmente já faz isso automaticamente).

### 2. Clique em "Run":

- o Seu site estará ao vivo no link .replit.dev.

### 3. Faça o Teste de Fluxo Completo:

- o **Teste 1 (Páginas Estáticas):** Abra seu link ...replit.dev. Clique no menu e navegue para "Equipe", "Projeto" e "Votação". Veja se tudo funciona.
- o **Teste 2 (Ranking):** Abra ...replit.dev/ranking.html. Veja se o teste@email.com aparece no ranking com o recorde de 85%.
- o **Teste 3 (Dashboard/Login):** Abra ...replit.dev/dashboard.html. Digite teste@email.com e clique em "Buscar". Veja se todos os gráficos e métricas (LFO, TZF, etc.) são carregados.
- o Teste 4 (A Ponte/TV): Abra o link da tela da TV, passando os dados de teste na URL: ...replit.dev/race-finished.html?p1id=teste@email.com&p1sid=sessao\_teste\_1
- o **Teste 5 (O Scan):** Veja se um QR code dinâmico apareceu na tela. Escaneie-o com seu celular.
- o **Teste Final:** Seu celular deve abrir o dashboard.html, e por causa do ?pid=teste@email.com na URL, ele deve pular o login e carregar sua análise automaticamente.

Se tudo isso funcionar, seu projeto está 100% implementado.