

# Entrega 2

## Auth, Libro Estilos, Diseño, Mapa: Dungeon Fighter



## Contenido

1.	Proceso de Autenticación y Dashboard.....	3
1.1	Dashboard Inicial:.....	4
2.	Guía de Estilos.....	6
3.	Diseño del Prototipado y Mapa de Navegación.....	7

## 1. Proceso de Autenticación y Dashboard

### 1. Preparar servicios y reglas

En la escena “LoginRegistro”, tenemos un GameObject llamado “Servicios”, con:

- FirebaseAuthCliente, con FirebaseAuthAjustes asignado.
- FirestoreCliente, con FirebaseAuthAjustes asignado.
- GameSaveServicio, con las referencias Auth y Firestore.
- BootServicios, con el GameObject Servicios.

### 2. En Firebase -> Reglas tenemos las reglas para controlar quién accederá al programa:

```
rules_version = '2';
service cloud.firestore {
  match /databases/{db}/documents {

    function isSignedIn() {
      return request.auth != null;
    }

    function isOwner(uid) {
      return isSignedIn() && request.auth.uid == uid;
    }

    // Admin si el documento en /usuarios/{uid} es esAdmin == true
    function isAdmin() {
      return isSignedIn() &&
        get(/databases/$(db)/documents/usuarios/$(request.auth.uid)).data.esAdmin ==
        true;
    }

    // Usuarios
    match /usuarios/{uid} {
      // Usuario o Admin pueden leer
      allow read: if isOwner(uid) || isAdmin();

      // Crear/actualizar su propio perfil
      // Un admin también puede crear/actualizar cualquier perfil.
      allow create, update: if isOwner(uid) || isAdmin();

      // Borrar un perfil, solo admin
      allow delete: if isAdmin();
    }

    // Partidas
    // Documento por UID
    match /partidasGuardadas/{uid} {
      // El usuario gestionará su partida; el admin también.
      allow read, write: if isOwner(uid) || isAdmin();
    }

    // Permitir al admin listados sobre la colección completa
    match /partidasGuardadas/{anyDocId} {
      allow list: if isAdmin();
    }

    // Cierra el resto
    match /{document=**} {
      allow read, write: if false;
    }
  }
}
```

3. El usuario ejecuta el programa, y automáticamente se dirige al Login / Registro.
4. Si no pulsa nada, directamente se abre el panel de Login, si pulsa en “Cambiar”, se le desbloquearán las opciones para el registro.
5. El Login / Registro, llama a Firebase Auth, y obtiene el idToken y uid del usuario.
6. Se guarda/actualiza usuarios/{uid}: si isAdmin = true, mostrará el Panel de Admin, si no, el Panel de Usuario.

### 1.1 Dashboard Inicial:

Ventana Login:



Ventana Registro:



Panel Login:



Panel Registro:



## 2. Guía de Estilos



**Logo:**



**Colores Principales:**



**Colores Secundarios:**



**Tipografía:**

**Tiny 10 (H1)**

**Tiny 10 (H2)**

**Tiny 10 (H3)**

**Botones:**



**Reglas Pixel Art:**

**Tile base: 16x16**

**PPU: Unity 16**

### 3. Diseño del Prototipado y Mapa de Navegación

Inicio



Usuario

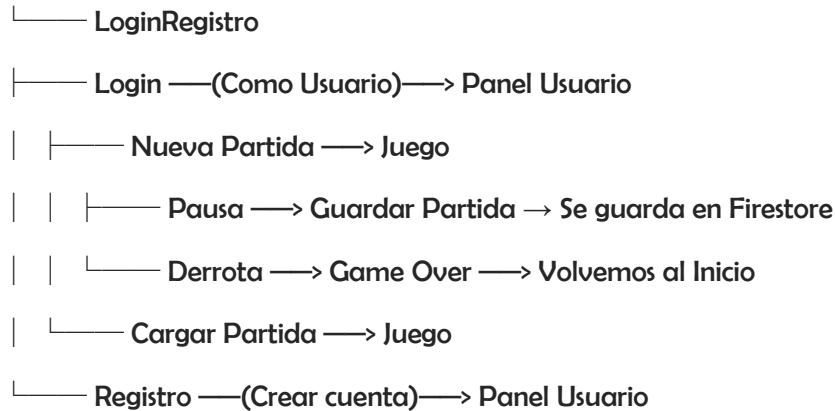
Admin



Si seleccionamos “Nueva/Cargar Partida”, nos llevará al HUD del Juego principal, para comenzar la partida,

### Flujo del Usuario:

#### **Inicio**



#### **Admin**

