

Galactic Cargo Logistics

Bienvenido, Cadete. La Federación de Comercio necesita un nuevo sistema para gestionar el transporte de mercancías en el borde exterior. Tu misión es desarrollar el frontend de la aplicación **Galactic Cargo** utilizando las últimas tecnologías de Angular.

🎯 Objetivos Técnicos

- Arquitectura:** Rutas, Lazy Loading y Componentes Standalone.
- Seguridad:** Autenticación JWT, Guardianes (Guards) e Interceptores.
- Formularios:** Uso híbrido (Template-Driven para Login, Reactivos para Registro/Alta).
- Reactividad:** Uso de **Signals** para la gestión del estado global.

🏗 Estructura del Proyecto

La aplicación debe seguir estrictamente esta estructura de rutas:

URL	Componente	Tipo de Carga	Protección
/home	HomeComponent	Eager (Directa)	Pública
/auth/login	LoginComponent	Lazy Loading	Pública
/auth/register	RegisterComponent	Lazy Loading	Pública
/dashboard	DashboardLayout	Lazy Loading	Privada (AuthGuard)
/dashboard/list	CargoListComponent	Lazy (Hija)	Privada
/dashboard/create	CargoCreateComponent	Lazy (Hija)	Privada
**	Redirect a Home	-	-

🔧 Requisitos Funcionales

1. Core & Seguridad (AuthService)

- Debe usar **Signals** para almacenar el usuario actual.
- Al iniciar la app (constructor o APP_INITIALIZER), debe verificar si existe un token en localStorage y validarla contra el endpoint /auth/check-status .
- **Interceptor 1 (Token)**: Debe añadir el header Authorization: Bearer ... a todas las peticiones si el usuario está logueado.
- **Interceptor 2 (Error)**:
 - Si recibe un **401**: Logout forzoso y redirigir al Login.
 - Si recibe un **403**: Mostrar una alerta (SweetAlert) de "Permisos Insuficientes".

2. Módulo de Autenticación

- **Login**: Formulario de **Plantilla (Template-Driven)**. Campos: Email y Password.
- **Registro**: Formulario **Reactivio**.
 - Validaciones: Nombre requerido, Email (formato correcto), Password (mínimo 6 caracteres).
 - **Validación Asíncrona (Opcional/Bonus)**: Comprobar si el email ya existe.

3. Dashboard (Gestión de Cargas)

- **Listado**: Mostrar las cargas en tarjetas.
 - Si el usuario es **Commander**, debe aparecer un botón de "Eliminar" en cada tarjeta.
 - Si es **Pilot**, el botón de eliminar no debe existir o estar deshabilitado.
- **Crear Carga**: Formulario **Reactivio** para dar de alta nueva mercancía.

Recursos Gráficos (Plantillas HTML)

Para que la aplicación tenga ese aspecto "Sci-Fi", añade estos estilos globales en tu styles.css y asegúrate de tener **Bootstrap 5** y **Bootstrap Icons** instalados.

Instalación rápida:

```
npm install bootstrap bootstrap-icons
```

styles.css:

CSS

```
/* Importar Bootstrap en styles.css o angular.json */
@import 'bootstrap/dist/css/bootstrap.min.css';
@import 'bootstrap-icons/font/bootstrap-icons.css';

body {
    background: radial-gradient(circle at center, #1a1a2e 0%, #16213e 50%, #0f3460 100%);
    min-height: 100vh;
    color: #e0e0e0;
}

.glass-panel {
    background: rgba(255, 255, 255, 0.05);
    backdrop-filter: blur(10px);
    border: 1px solid rgba(255, 255, 255, 0.1);
    border-radius: 12px;
}

.text-neon {
    color: #00d2ff;
    text-shadow: 0 0 10px rgba(0, 210, 255, 0.5);
}
```

1. Plantilla: Login (Template-Driven)

Archivo: login.component.html

Nota: Debes añadir las referencias `#email="ngModel"` y el `[(ngModel)]`.

HTML

```

<div class="container d-flex justify-content-center align-items-center" style="height: 90vh;">
    <div class="card glass-panel p-4 shadow-lg" style="width: 400px;">
        <div class="text-center mb-4">
            <i class="bi bi-rocket-takeoff fs-1 text-neon"></i>
            <h3 class="mt-2 fw-bold text-white">Galactic Access</h3>
        </div>

        <form>
            <div class="mb-3">
                <label class="form-label text-secondary">Identificación (Email)</label>
                <input type="email" class="form-control bg-dark text-white border-secondary" placeholder="pilot@galaxy.com">

                <div class="text-danger small mt-1">
                    El email es obligatorio.
                </div>
            </div>

            <div class="mb-3">
                <label class="form-label text-secondary">Código de Acceso</label>
                <input type="password" class="form-control bg-dark text-white border-secondary" placeholder="*****">
            </div>

            <div class="d-grid gap-2">
                <button type="submit" class="btn btn-primary fw-bold">
                    INICIAR PROPULSORES
                </button>
                <a routerLink="/auth/register" class="btn btn-outline-secondary btn-sm">
                    Solicitar Licencia de Vuelo
                </a>
            </div>
        </form>
    </div>
</div>

```

2. Plantilla: Registro (Formulario Reactivo)

Archivo: register.component.html

Nota: Vincula tu [FormGroup] y formControlName .

HTML

```
<div class="container mt-5">
    <div class="row justify-content-center">
        <div class="col-md-6">
            <div class="card glass-panel text-white">
                <div class="card-header border-0 bg-transparent text-center pt-4">
                    <h2 class="text-neon">Nuevo Recluta</h2>
                    <p class="text-muted">Únete a la flota de la Federación</p>
                </div>

                <div class="card-body p-4">
                    <form autocomplete="off">

                        <div class="mb-3">
                            <label class="form-label">Nombre del Piloto</label>
                            <div class="input-group">
                                <span class="input-group-text bg-dark border-secondary text-secondary"><i class="bi bi-person"></i></span>
                                <input type="text" class="form-control bg-dark text-white border-secondary" placeholder="Ej: Han Solo">
                            </div>
                            <span class="text-danger small">El nombre es requerido</span>
                        </div>

                        <div class="mb-3">
                            <label class="form-label">Correo Galáctico</label>
                            <div class="input-group">
                                <span class="input-group-text bg-dark border-secondary text-secondary"><i class="bi bi-envelope"></i></span>
                                <input type="email" class="form-control bg-dark text-white border-secondary" placeholder="user@galaxy.com">
                            </div>
                        </div>

                        <div class="mb-3">
                            <label class="form-label">Clave de Seguridad</label>
                            <div class="input-group">
                                <span class="input-group-text bg-dark border-secondary text-secondary"><i class="bi bi-key"></i></span>
                                <input type="password" class="form-control bg-dark text-white border-secondary" placeholder="*****">
                            </div>
                        </div>

                        <button type="submit" class="btn btn-success w-100 py-2 mt-3">
                            <i class="bi bi-check-circle me-2"></i> REGISTRAR PILOTO
                        </button>
                    </form>
                </div>
            </div>
        </div>
    </div>
</div>
```

```
        </button>
      </form>
    </div>
  </div>
</div>
</div>
```

3. Plantilla: Listado de Cargas (Dashboard)

Archivo: cargo-list.component.html

Nota: Usa `@for` para iterar las cargas y `getBadgeClass(status)` para los colores.

HTML

```

<div class="container mt-4">
    <div class="d-flex justify-content-between align-items-center mb-4">
        <h2 class="text-white fw-light">Manifiesto de Carga <span class="text-neon">| Sector 7</span></h2>
        <button routerLink="/dashboard/create" class="btn btn-outline-info">
            <i class="bi bi-plus-lg"></i> Nueva Carga
        </button>
    </div>

    <div class="row g-4">
        <div class="col-md-4">
            <div class="card glass-panel h-100 text-white">
                <div class="card-header border-secondary d-flex justify-content-between">
                    <span class="fw-bold font-monospace text-warning">CODE</span>
                    <span class="badge bg-success">Status</span>
                </div>
                <div class="card-body">
                    <h5 class="card-title fw-bold">Descripción Carga</h5>
                    <p class="card-text text-muted small mb-3">
                        <i class="bi bi-geo-alt"></i> Origen → <i class="bi bi-geo-fill"></i> Destino
                    </p>
                    <div class="d-flex justify-content-between align-items-center bg-dark bg-opacity-50 p-2 rounded">
                        <span class="text-secondary">Valor:</span>
                        <span class="text-neon fw-bold">฿ 5000</span>
                    </div>
                </div>

                <div class="card-footer border-0 bg-transparent text-end">
                    <button class="btn btn-sm btn-danger opacity-75">
                        <i class="bi bi-trash"></i> Desechar
                    </button>
                </div>
            </div>
        </div>
        <div class="col-12 text-center text-muted mt-5">
            <i class="bi bi-box-seam fs-1"></i>
            <p>No hay cargas en el sistema.</p>
        </div>
    </div>
</div>

```

4. Plantilla: Navbar (Componente Compartido)

Archivo: navbar.component.html

HTML

```
<nav class="navbar navbar-expand-lg navbar-dark bg-transparent border-bottom border-secondary mb-4">
  <div class="container">
    <a class="navbar-brand text-neon fw-bold" href="#">
      <i class="bi bi-stars"></i> GALACTIC CARGO
    </a>

    <div class="collapse navbar-collapse">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link" routerLink="/home" routerLinkActive="active">Noticias</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" routerLink="/dashboard/list" routerLinkActive="active">Operaciones</a>
        </li>
      </ul>

      <div class="d-flex">
        <ng-container>
          <a routerLink="/auth/login" class="btn btn-sm btn-outline-light me-2">Login</a>
        </ng-container>

        <ng-container>
          <div class="dropdown">
            <button class="btn btn-sm btn-dark dropdown-toggle border-secondary" type="button" data-bs-toggle="dropdown">
              <i class="bi bi-person-circle"></i> Usuario
            </button>
            <ul class="dropdown-menu dropdown-menu-dark">
              <li><button class="dropdown-item text-danger">Cerrar Sesión</button></li>
            </ul>
          </div>
        </ng-container>
      </div>
    </div>
  </div>
</nav>
```