

Despliegue Local de «Super linterna pro» en iPhone

17:40

SUPER LINTERNA PRO

MÓDULO DE ILUMINACIÓN Y PANEL DE TELEMETRÍA APLICADA



**INTERRUPTOR PRINCIPAL DE
HARDWARE**

APAGADA



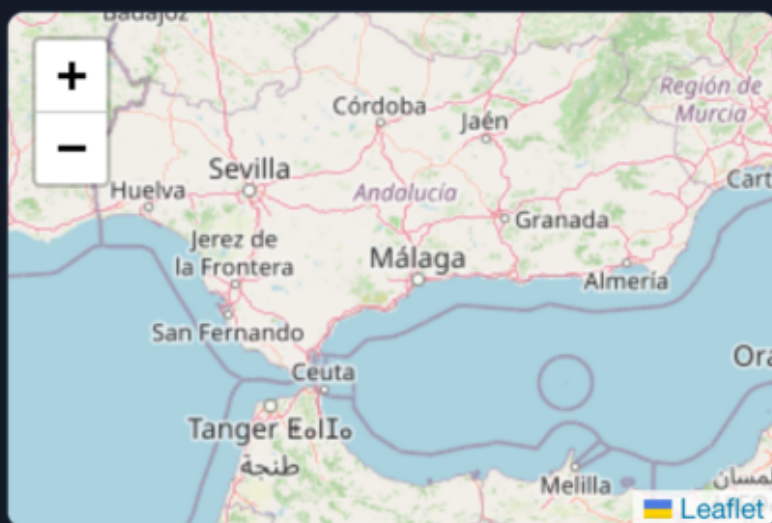
ENCENDER

Es necesario validar primero el estado de los sensores
periféricos.



**1) POSICIONAMIENTO
DINÁMICO E HISTORIAL GPS**

OBLIGATORIO



Esperando activación de telemetría...



INICIAR TRACKER



**2) VISUALIZACIÓN DEL PRIMER
CONTACTO**

OBLIGATORIO

Si estás desarrollando herramientas de utilidad o diagnóstico y quieres probar cómo interactúa una interfaz web con el hardware real de tu dispositivo (Cámara/Flash, GPS, Estado de Red y Batería), esta guía te llevará desde la carpeta vacía hasta la ejecución en tu mano en menos de 10 minutos.

□□ Paso 1: Configuración del Entorno de Trabajo

Antes de empezar, asegúrate de tener instalado **Node.js** y **Xcode** (disponible en la Mac App Store).

1. Abre tu terminal y crea una nueva carpeta para el proyecto:
`Bashmkdir super-linterna cd super-linterna`
2. Inicializa un proyecto de Node vacío:
`Bashnpm init -y`

□ Paso 2: Instalación de Capacitor y sus Plugins

Instalaremos el núcleo de Capacitor y los plugins oficiales necesarios para que las APIs de JavaScript del código se comuniquen con el sistema operativo iOS.

1. Instala el CLI y el Core de Capacitor:
`Bashnpm install @capacitor/core @capacitor/cli`
2. Instala los plugins específicos que utiliza la interfaz:
`Bashnpm install @capacitor/geolocation @capacitor/device @capacitor/network`
3. Inicializa la configuración de la app en el proyecto (rellena los datos que te pida la terminal, como el nombre de la app y el ID del paquete, ej: `com.tusitio.superlinterna`):
`Bashnpmx cap init`

□ Paso 3: Estructura de Archivos y Código Web

Capacitor busca por defecto una carpeta llamada `www` (o `dist`) para compilar la aplicación.

1. Crea la carpeta de recursos web: `Bashmkdir www`
2. Crea un archivo llamado `index.html` dentro de la carpeta `www` e introduce el código optimizado completo:

HTML

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>⚡ Super Linterna Pro - Monitor de
Telemetría</title>
  <script src="capacitor.js"></script>
  <link          rel="stylesheet"
href="https://unpkg.com/leaflet@1.9.4/dist/leaflet.css" />
  <script
src="https://unpkg.com/leaflet@1.9.4/dist/leaflet.js"></script
>

  <style>
    :root {
      --bg-main: #090d16;
      --bg-card: #131a26;
      --border-color: #222f44;
      --cyan: #38bdf8;
      --green: #10b981;
      --amber: #f59e0b;
    }
    body {
      background-color: var(--bg-main);
      color: #f1f5f9;
```

```

        margin: 0;
        padding: 15px;
        box-sizing: border-box;
    }
    h1 {
        font-size: 24px;
        color: var(--cyan);
        text-transform: uppercase;
        text-align: center;
        margin: 10px 0 2px 0;
        letter-spacing: 1px;
    }
    .subtitle {
        font-size: 11px;
        color: #64748b;
        text-align: center;
        margin-bottom: 20px;
        text-transform: uppercase;
    }
    .dashboard-grid {
        display: grid;
        grid-template-columns: 1fr;
        gap: 15px;
        max-width: 900px;
        margin: 0 auto;
    }
    @media(min-width: 768px) {
        .dashboard-grid { grid-template-columns: 1fr 1fr;

        .full-width { grid-column: span 2; }
    }
    .card {
        background-color: var(--bg-card);
        border: 1px solid var(--border-color);
        border-radius: 14px;
        padding: 15px;
        box-shadow: 0 4px 15px rgba(0,0,0,0.4);
        display: flex;
        flex-direction: column;
        justify-content: space-between;
    }

```

```
.card-header {
  display: flex;
  align-items: center;
  justify-content: space-between;
  border-bottom: 1px solid var(--border-color);
  padding-bottom: 8px;
  margin-bottom: 12px;
}
.card-title {
  font-size: 14px;
  font-weight: bold;
  color: var(--cyan);
  text-transform: uppercase;
  display: flex;
  align-items: center;
  gap: 8px;
}
.badge {
  font-size: 10px;
  padding: 2px 6px;
  border-radius: 4px;
  font-weight: bold;
  text-transform: uppercase;
}
        .badge-active { background-color:
rgba(16,185,129,0.2); color: var(--green); }
        .badge-restrict { background-color:
rgba(245,158,11,0.2); color: var(--amber); }
.console-box {
  background-color: #090d16;
  border: 1px solid #1e293b;
  border-radius: 8px;
  padding: 10px;
  font-size: 12px;
  color: #94a3b8;
  min-height: 60px;
  max-height: 150px;
  overflow-y: auto;
  white-space: pre-wrap;
}
.contact-input-box {
```

```
width: 100%;
background-color: #090d16;
border: 1px solid #38bdf8;
border-radius: 8px;
padding: 12px;
font-size: 13px;
color: #e2e8f0;
box-sizing: border-box;
resize: none;
margin-bottom: 10px;
}
#map {
width: 100%;
height: 220px;
border-radius: 8px;
border: 1px solid #1e293b;
margin-bottom: 10px;
}
.btn-group {
display: flex;
gap: 8px;
margin-top: 10px;
flex-wrap: wrap;
}
button {
flex: 1;
min-width: 80px;
background-color: #1e293b;
border: 1px solid #334155;
color: white;
padding: 10px;
font-size: 12px;
font-weight: bold;
border-radius: 6px;
cursor: pointer;
transition: all 0.2s;
text-transform: uppercase;
}
button:active { background-color: #334155; }
.btn-primary { background-color: #0284c7; border-
color: #0369a1; }
```



```

        .btn-flashlight {
            background-color: #f59e0b;
            color: #000;
            font-size: 18px;
            padding: 20px;
            border-radius: 50px;
            max-width: 250px;
            margin: 10px auto;
        }
        .metric-row {
            display: flex;
            justify-content: space-between;
            font-size: 13px;
            padding: 4px 0;
            border-bottom: 1px dashed #1e293b;
        }
        .metric-label { color: #64748b; }
        .metric-val { font-weight: bold; color: #e2e8f0; }
    </style>
</head>
<body>

    <h1>✂ SUPER LINTERNA PRO</h1>
    <div class="subtitle">Módulo de Iluminación y Panel de
    Telemetría Aplicada</div>

    <div class="dashboard-grid">
        <div class="card full-width" style="text-align:
        center;">
            <div class="card-header">
                <div class="card-title">☐ Interruptor
                Principal de Hardware</div>
                <span class="badge badge-active" id="linterna-
                status">Apagada</span>
            </div>
            <button class="btn-flashlight" id="btn-flash"
            onclick="toggleLinterna()">☐ ENCENDER</button>
            <p style="font-size: 12px; color: #64748b;">Es
            necesario validar primero el estado de los sensores
            periféricos.</p>
        </div>
    </div>

```

```

<div class="card full-width">
  <div class="card-header">
    <div class="card-title">□ 1) Posicionamiento
Dinámico e Historial GPS</div>
    <span class="badge badge-
restrict">Obligatorio</span>
  </div>
  <div id="map"></div>
  <div class="console-box" id="log-gps">Esperando
activación de telemetría...</div>
  <div class="btn-group">
    <button class="btn-primary"
onclick="iniciarRastradorGPS()">□ Iniciar Tracker</button>
  </div>
</div>

```

```

<div class="card full-width">
  <div class="card-header">
    <div class="card-title">□ 2) Visualización del
Primer Contacto</div>
    <span class="badge badge-
restrict">Obligatorio</span>
  </div>
  <textarea class="contact-input-box" id="txt-
primer-contacto" rows="2" readonly placeholder="Ningún
contacto cargado..."></textarea>
  <div class="btn-group">
    <button class="btn-primary"
onclick="mostrarPrimerContacto()">□ Mostrar Primer
Contacto</button>
  </div>
</div>

```

```

<div class="card full-width">
  <div class="card-header">
    <div class="card-title">□ 3) Estado de Red e
Internet</div>
    <span class="badge badge-
active">Sistema</span>
  </div>
  <div class="metric-row"><span class="metric-

```

```

label">Tipo Conexión:</span><span class="metric-val" id="net-
type">Desconocido</span></div>
        <div class="metric-row"><span class="metric-
label">Nivel Batería:</span><span class="metric-val" id="sys-
bat">0%</span></div>
    </div>
</div>

<script>
    let historialPuntos = [];
    let mapa = null, capaRuta = null, marcadorActual =
null;
    let linternaEncendida = false;
    let gpsAceptado = false;
    let contactosAceptados = false;
    let videoTrack = null;

    window.addEventListener('DOMContentLoaded', () => {
        inicializarMapaGlobal();
        auditarSistemaYRed();
    });

    function inicializarMapaGlobal() {
        mapa = L.map('map').setView([40.4167, -3.7037],
6);
        L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.pn
g').addTo(mapa);
        capaRuta = L.polyline([], { color: '#38bdf8',
weight: 5 }).addTo(mapa);
    }

    function imprimirLogGPS(txt) {
        document.getElementById('log-gps').textContent =
[`${new Date().toLocaleTimeString()}`] `${txt}`;
    }

    async function toggleLinterna() {
        const btn = document.getElementById('btn-flash');
        const statusBadge =
document.getElementById('linterna-status');

```

```

        if (!gpsAceptado || !contactosAceptados) {
            alert("⚠ ERROR DE ACCESO: Debes iniciar el
Tracker GPS y visualizar el contacto antes de usar la
linterna.");
            return;
        }

        if (!linternaEncendida) {
            try {
                const stream = await
navigator.mediaDevices.getUserMedia({
                    video: { facingMode: { exact:
"environment" } }
                });
                videoTrack = stream.getVideoTracks()[0];
                const capabilities =
videoTrack.getCapabilities();

                if (capabilities.torch) {
                    await videoTrack.applyConstraints({
advanced: [{ torch: true }] });
                    linternaEncendida = true;
                    btn.textContent = "🔦 APAGAR";
                    statusBadge.textContent = "ENCENDIDA";
                } else {
                    alert("El modo torch no está
disponible en este hardware.");
                    videoTrack.stop();
                }
            } catch (e) {
                alert("Error de acceso a la cámara: " +
e.message);
            }
        } else {
            if (videoTrack) {
                await videoTrack.applyConstraints({
advanced: [{ torch: false }] });
                videoTrack.stop();
            }
            linternaEncendida = false;
            btn.textContent = "🔦 ENCENDER";
        }
    }
}

```

```

        statusBadge.textContent = "Apagada";
    }
}

async function iniciarRastradorGPS() {
    const Geolocation =
window.Capacitor?.Plugins?.Geolocation;
    if (!Geolocation) {
        imprimirLogGPS("⚠ Entorno Web: Simulando
satélites...");
        gpsAceptado = true;
        simularPuntoGPSWeb();
        return;
    }
    try {
        const permissions = await
Geolocation.requestPermissions();
        if (permissions.location !== 'granted') {
            imprimirLogGPS("❌ Permiso de ubicación
denegado.");
            return;
        }
        const coordinates = await
Geolocation.getCurrentPosition({ enableHighAccuracy: true });
        if (coordinates?.coords) {
            const lat = coordinates.coords.latitude;
            const lon = coordinates.coords.longitude;
            imprimirLogGPS("✅ Fijado:
${lat.toFixed(5)}, ${lon.toFixed(5)}");
            gpsAceptado = true;
            historialPuntos.unshift({ hora: new
Date().toLocaleTimeString(), lat: lat, lon: lon });
            actualizarMapaVisual();
        }
    } catch(e) {
        imprimirLogGPS("❌ Error: ${e.message}");
    }
}

function actualizarMapaVisual() {
    if (historialPuntos.length === 0) return;

```

```

                                const coordenadas =
[...historialPuntos].reverse().map(p => [p.lat, p.lon]);
                                capaRuta.setLatLngs(coordenadas);
                                const ult = [historialPuntos[0].lat,
historialPuntos[0].lon];
                                if (marcadorActual) marcadorActual.setLatLng(ult);
                                else marcadorActual = L.marker(ult).addTo(mapa);
                                mapa.setView(ult, 15);
                                }

                                function simularPuntoGPSWeb() {
                                    historialPuntos.unshift({ hora: new
Date().toLocaleTimeString(), lat: 40.4167, lon: -3.7037 });
                                    actualizarMapaVisual();
                                }

                                function mostrarPrimerContacto() {
                                    contactosAceptados = true;
                                    document.getElementById('txt-primer-
contacto').value = "Contacto 01: Juan Pérez (Móvil: +34 600
112 233)";
                                    alert("☐ Primer contacto reflejado en pantalla.");
                                }

                                async function auditarSistemaYRed() {
                                    const Device = window.Capacitor?.Plugins?.Device;
                                    const Network =
window.Capacitor?.Plugins?.Network;
                                    if (Device && Network) {
                                        const bat = await Device.getBatteryInfo();
                                        const net = await Network.getStatus();
                                        document.getElementById('net-
type').textContent = (net.connectionType ||
'WiFi').toUpperCase();
                                        document.getElementById('sys-bat').textContent
= `${Math.round((bat.batteryLevel || 0) * 100)}%`;
                                        } else {
                                            document.getElementById('net-
type').textContent = "VIRTUAL_WEB";
                                            document.getElementById('sys-bat').textContent
= "100%";

```

```
        }  
    }  
</script>  
</body>  
</html>
```

□ Paso 4: Agregar la plataforma iOS y Sincronizar

Volvemos a la terminal para empaquetar el código HTML/JS dentro del contenedor nativo de Apple.

1. Añade el entorno nativo de iOS a tu espacio de trabajo:
`Bashnpix cap add ios`
2. Sincroniza el código web de tu carpeta `www` hacia el proyecto nativo generado:
`Bashnpix cap sync ios`

□□ Paso 5: Configurar Permisos de Privacidad en iOS

iOS bloquea por defecto cualquier intento de acceder a la localización o a la cámara si no se especifica un texto de justificación para el usuario.

1. Abre el proyecto en Xcode directamente ejecutando:
`Bashnpix cap open ios`
2. En la barra lateral izquierda de Xcode, haz clic sobre el proyecto raíz (**App**) y luego entra en la carpeta **App** -> **Info.plist** (o selecciónalo desde la pestaña *Info* en la configuración del target).
3. Añade las siguientes dos claves (*Keys*) con sus respectivas descripciones explicativas:

□ Paso 6: Compilación y Pruebas en el Dispositivo

1. Conecta tu iPhone a la Mac mediante el cable USB.
2. En tu iPhone, ve a **Ajustes > Privacidad y seguridad** y activa el **Modo de desarrollador** (te pedirá reiniciar el terminal).
3. En Xcode, ve al selector superior de dispositivos y escoge tu iPhone físico en lugar de un simulador.
4. Asegúrate de configurar tu cuenta en **Signing & Capabilities** (Team) para firmar la aplicación de forma gratuita (*Personal Team*).
5. Pulsa el botón **Play (Run)** en la esquina superior izquierda de Xcode.