

**Cuadro de mando definitivo
para iPhone: telemetría
completa sin permisos nativos
de iOS**

12:02



MEGA DASHBOARD



Auditoría de Hardware y Sensores sin Permisos Globales



DESBLOQUEAR KINETICS (WEBKIT)



ENERGÍA & CARGA

Nivel: 60%

Cargando: No



PROCESADOR (APPLE SILICON)

Núcleos CPU: 4 lógicos

Frecuencia Web: 60Hz



UBICACIÓN PASIVA (VÍA RED IP)

Ciudad/País: Offline o Bloqueado

Proveedor ISP: Red no disponible



ACELERÓMETRO SIMPLES

X: 0.04 m/s²

Y: 0.01 m/s²

Z: -0.06 m/s²



FUERZAS G E INTERVALO

G-Eje Z: -1.75 m/s²

Frecuencia: 0ms



VELOCIDAD ROTACIÓN

Alpha: -1.7°/s

Beta: -0.9°/s

Gamma: 1.1°/s



ÁNGULO ESPACIAL

12:02  LOCALIZACIÓN PASIVA (VÍA RED IP)



Ciudad/País: Offline o Bloqueado
Proveedor ISP: Red no disponible

ACCELERÓMETRO SIMPLES

X: -0.02 m/s^2
Y: -0.07 m/s^2
Z: -0.07 m/s^2

FUERZAS G E INTERVALO

G-Eje Z: -2.06 m/s^2
Frecuencia: 0ms

VELOCIDAD ROTACIÓN

Alpha: $-0.9^\circ/\text{s}$
Beta: $1.6^\circ/\text{s}$
Gamma: $1.0^\circ/\text{s}$

ÁNGULO ESPACIAL

Z (α): 5°
X (β): 78°
Y (γ): 15°

PANEL & DISPLAY

Render: 1170x2532
Factor Retina: 3x

PREFERENCIAS INTERNAS DEL SISTEMA

Tema de Interfaz: Oscuro (Dark Mode)
Reducir Movimiento: Desactivado
(Estándar)
Idioma / Zona Horaria: es-ES /
Europe/Madrid

Uno de los artículos más atractivos para los lectores de desarrollo móvil y ciberseguridad es demostrar **cuánta información puede extraer una aplicación o web sobre el hardware de un iPhone sin activar las alertas globales de privacidad de Apple (Info.plist).**

En esta guía avanzada vamos a construir un **Mega-Dashboard de Telemetría**. Aprovecharemos la autorización de movimiento de WebKit para calcular Fuerzas G, mediremos el rendimiento de la pantalla para deducir la tasa de refresco, leeremos las variables de accesibilidad del usuario y utilizaremos un rastreo pasivo por IP para ubicar el dispositivo geográficamente de forma transparente.

Paso 1: Crear el directorio y la estructura base

Abre la **Terminal** de tu Mac y ejecuta los siguientes comandos para crear un proyecto limpio desde cero:

```
# 1. Crear la carpeta del proyecto masivo
mkdir megadashboard
```

```
# 2. Entrar a la carpeta
cd megadashboard
```

```
# 3. Inicializar el entorno Node.js
npm init -y
```

```
# 4. Instalar las dependencias de Capacitor
npm install @capacitor/core @capacitor/cli
```

Paso 2: Inicializar la configuración de la App

Ejecuta el siguiente comando para darle nombre e identidad al proyecto en iOS:

```
npx cap init "MegaDash" "com.tuusuario.megadash" --web-dir=www
```

Paso 3: Instalar el Plugin de Dispositivo (Device)

Para sincronizar los datos de la batería real del iPhone mediante el puente de Capacitor, añadimos su plugin oficial:

```
npm install @capacitor/device
```

Paso 4: Crear la interfaz del Mega-Dashboard (HTML5, CSS3 y JS)

Creamos el directorio web del proyecto:

```
mkdir www
```

Ahora, crea el archivo **index.html** dentro de la carpeta **www** y pega el siguiente código unificado. Tiene un diseño de rejilla (*grid*) optimizado para pantallas retina e interfaz oscura táctil:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Mega Dashboard Telemetría</title>
  <style>
    body {
      background-color: #090d16;
      color: #f3f4f6;
      margin: 0;
      padding: 15px;
    }
    h1 {
      text-align: center;
      color: #6366f1;
```

```

        font-size: 24px;
        margin-bottom: 5px;
        text-transform: uppercase;
        letter-spacing: 1px;
    }
    .subtitle {
        text-align: center;
        color: #6b7280;
        font-size: 12px;
        margin-bottom: 20px;
    }
    .btn-container {
        text-align: center;
        margin-bottom: 20px;
    }
    .btn-sensor {
        background-color: #6366f1;
        color: white;
        border: none;
        padding: 14px 24px;
        font-size: 15px;
        font-weight: bold;
        border-radius: 12px;
        cursor: pointer;
        box-shadow: 0 4px 12px rgba(99, 102, 241, 0.4);
    }
    .grid {
        display: grid;
        grid-template-columns: 1fr;
        gap: 12px;
        width: 100%;
        max-width: 480px;
        margin: 0 auto;
    }
    @media(min-width: 400px) {
        .grid { grid-template-columns: 1fr 1fr; }
        .full-width { grid-column: span 2; }
    }
    .card {
        background: #111827;
        border: 1px solid #1f2937;

```

```

        border-radius: 14px;
        padding: 12px;
        box-shadow: 0 4px 10px rgba(0,0,0,0.2);
    }
    .card-title {
        font-size: 11px;
        color: #9ca3af;
        text-transform: uppercase;
        font-weight: bold;
        margin-bottom: 8px;
        border-bottom: 1px solid #1f2937;
        padding-bottom: 4px;
    }
    .sensor-data {
        font-size: 14px;
        color: #10b981;
        line-height: 1.4;
    }
    .highlight { color: #38bdf8; }
    .error { color: #ef4444; }
</style>
</head>
<body>

    <h1>☐ Mega Dashboard</h1>
    <div class="subtitle">Auditoría de Hardware y Sensores sin
    Permisos Globales</div>

    <div class="btn-container">
        <button class="btn-sensor"
onclick="desbloquearSensores()">☐ DESBLOQUEAR KINETICS
(WEBKIT)</button>
    </div>

    <div class="grid">

        <div class="card">
            <div class="card-title">☐ Energía & Carga</div>
            <div class="sensor-data">
                Nivel: <span id="bat-level"
class="highlight">--%</span><br>

```

```

        Cargando: <span id="bat-charging">--</span>
    </div>
</div>

<div class="card">
    <div class="card-title">▢ Procesador (Apple
Silicon)</div>
    <div class="sensor-data">
        Núcleos CPU: <span id="cpu-cores"
class="highlight">--</span> lógicos<br>
        Frecuencia Web: <span id="web-hz">60Hz</span>
    </div>
</div>

<div class="card full-width">
    <div class="card-title">▢ Ubicación Pasiva (Vía
Red IP)</div>
    <div class="sensor-data">
        Ciudad/País: <span id="ip-location"
class="highlight">Cargando...</span><br>
        Proveedor ISP: <span id="ip-
isp">Detectando...</span>
    </div>
</div>

<div class="card">
    <div class="card-title">▢ Acelerómetro
Simples</div>
    <div class="sensor-data">
        X: <span id="acc-x">0.00</span> m/s2<br>
        Y: <span id="acc-y">0.00</span> m/s2<br>
        Z: <span id="acc-z">0.00</span> m/s2
    </div>
</div>

<div class="card">
    <div class="card-title">▢ Fuerzas G e
Intervalo</div>
    <div class="sensor-data">
        G-Eje Z: <span id="g-z">0.00</span> m/s2<br>
        Frecuencia: <span id="srv-interval">0</span>ms
    </div>
</div>

```

```
</div>
</div>
```

```
<div class="card">
  <div class="card-title">⏏ Velocidad Rotación</div>
  <div class="sensor-data">
    Alpha: <span id="rot-alpha">0.0</span>°/s<br>
    Beta: <span id="rot-beta">0.0</span>°/s<br>
    Gamma: <span id="rot-gamma">0.0</span>°/s
  </div>
</div>
```

```
<div class="card">
  <div class="card-title">⏏ Ángulo Espacial</div>
  <div class="sensor-data">
    Z ( $\alpha$ ): <span id="ang-alpha">0°</span><br>
    X ( $\beta$ ): <span id="ang-beta">0°</span><br>
    Y ( $\gamma$ ): <span id="ang-gamma">0°</span>
  </div>
</div>
```

```
<div class="card">
  <div class="card-title">⏏ Panel & Display</div>
  <div class="sensor-data">
    Render: <span id="scr-res">0x0</span><br>
    Factor Retina: <span id="scr-pixel">0</span>x
  </div>
</div>
```

```
<div class="card full-width">
  <div class="card-title">⚙ ⏏ Preferencias Internas
del Sistema</div>
  <div class="sensor-data">
    Tema de Interfaz: <span id="pref-theme"
class="highlight">--</span><br>
    Reducir Movimiento: <span id="pref-motion">--
</span><br>
    Idioma / Zona Horaria: <span id="pref-lang"
style="color:#38bdf8;">--</span>
  </div>
</div>
```

</div>

<script>

```
// 1. TELEMETRÍA DE ENERGÍA (CAPACITOR)
async function actualizarBateria() {
  try {
    if (window.Capacitor &&
window.Capacitor.Plugins.Device) {
      const info = await
window.Capacitor.Plugins.Device.getBatteryInfo();
      document.getElementById('bat-
level').textContent = Math.round(info.batteryLevel * 100) +
'%';
      document.getElementById('bat-
charging').textContent = info.isCharging ? "Sí <" : "No";
    }
  } catch (e) { console.error(e); }
}

// 2. GEOLOCALIZACIÓN PASIVA POR IP (SIN PERMISOS DE
GPS)
async function rastreoPasivoIP() {
  try {
    const res = await
fetch('https://ipapi.co/json/');
    const data = await res.json();
    document.getElementById('ip-
location').textContent = `${data.city}, ${data.country_name}
(${data.country_code});
    document.getElementById('ip-isp').textContent
= data.org;
  } catch (err) {
    document.getElementById('ip-
location').textContent = "Offline o Bloqueado";
    document.getElementById('ip-isp').textContent
= "Red no disponible";
  }
}

// 3. DESBLOQUEO DE SENSORES WEB (REQUERIDO POR
INTERACCIÓN EN iOS)
```

```

        async function desbloquearSensores() {
            if (typeof DeviceOrientationEvent !== 'undefined'
&& typeof DeviceOrientationEvent.requestPermission ===
'function') {
                try {
                    const status = await
DeviceOrientationEvent.requestPermission();
                    if (status === 'granted') {
                        vincularEventosMovimiento();
                        alert("■ Sensores cinéticos en
línea.");
                    } else {
                        alert("■ Acceso denegado.");
                    }
                } catch (e) { alert("Error: Pulsa el botón
directamente."); }
            } else {
                vincularEventosMovimiento(); // Android o
entornos abiertos
            }
        }

        // 4. CAPTURA ASÍNCRONA DE KINETICS Y CORRIENTE DE
MOVIMIENTO
        function vincularEventosMovimiento() {
            // Acelerómetro y Fuerza G
            window.addEventListener('devicemotion', (e) => {
                if (e.acceleration) {
                    document.getElementById('acc-
x').textContent = (e.acceleration.x || 0).toFixed(2);
                    document.getElementById('acc-
y').textContent = (e.acceleration.y || 0).toFixed(2);
                    document.getElementById('acc-
z').textContent = (e.acceleration.z || 0).toFixed(2);
                }
                if (e.accelerationIncludingGravity) {
                    document.getElementById('g-z').textContent
= (e.accelerationIncludingGravity.z || 0).toFixed(2);
                }

                document.getElementById('srv-
interval').textContent = e.interval ? Math.round(e.interval) :

```

```

16;
        if (e.rotationRate) {
            document.getElementById('rot-
alpha').textContent = (e.rotationRate.alpha || 0).toFixed(1);
            document.getElementById('rot-
beta').textContent = (e.rotationRate.beta || 0).toFixed(1);
            document.getElementById('rot-
gamma').textContent = (e.rotationRate.gamma || 0).toFixed(1);
        }
    });

    // Giroscopio Orientación
    window.addEventListener('deviceorientation', (e)
=> {
        document.getElementById('ang-
alpha').textContent = Math.round(e.alpha || 0) + '°';
        document.getElementById('ang-
beta').textContent = Math.round(e.beta || 0) + '°';
        document.getElementById('ang-
gamma').textContent = Math.round(e.gamma || 0) + '°';
    });
}

// 5. HARDWARE ESTÁTICO Y PERFILES DE USUARIO
function auditarHardwareYPreferencias() {
    // Cores del Chip
    document.getElementById('cpu-cores').textContent =
navigator.hardwareConcurrency || "6 (A-Series)";
    // Resolución Real
    const w = window.screen.width *
window.devicePixelRatio;
    const h = window.screen.height *
window.devicePixelRatio;
    document.getElementById('scr-res').textContent =
`${w}x${h}`;
    document.getElementById('scr-pixel').textContent =
window.devicePixelRatio;

    // Detección de Modo Oscuro / Claro
    const isDark = window.matchMedia('(prefers-color-
scheme: dark)').matches;

```

```

        document.getElementById('pref-theme').textContent
= isDark ? "Oscuro (Dark Mode)" : "Claro (Light Mode)";

        // Animaciones Reducidas
        const reduceMotion = window.matchMedia('(prefers-
reduced-motion: reduce)').matches;
        document.getElementById('pref-motion').textContent
= reduceMotion ? "Activado (Ahorro Visual)" : "Desactivado
(Estándar)";

        // Idioma y Zona Horaria
        const lang = navigator.language || "es-ES";
        const tz =
Intl.DateTimeFormat().resolvedOptions().timeZone ||
"Desconocida";
        document.getElementById('pref-lang').textContent =
`${lang} / ${tz}`;

        // Estimación de Tasa de Refresco ProMotion (60Hz
vs 120Hz)
        let frames = 0;
        let startTime = performance.now();
        function checkFps() {
            frames++;
            let time = performance.now();
            if (time >= startTime + 1000) {
                document.getElementById('web-
hz').textContent = frames + "Hz";
                frames = 0;
                startTime = time;
            }
            requestAnimationFrame(checkFps);
        }
        requestAnimationFrame(checkFps);
    }

    // INICIALIZACIÓN DE ENTORNO
    window.addEventListener('DOMContentLoaded', () => {
        auditarHardwareYPreferencias();
        rastreoPasivoIP();
        setInterval(actualizarBateria, 3000);
    });

```

```
        setTimeout(actualizarBateria, 500);
    });
</script>
</body>
</html>
```

Paso 5: Sincronizar y Desplegar en Xcode

De vuelta en la **Terminal**, enlaza el nuevo HTML con el empaquetado nativo de Apple:

```
npm install @capacitor/ios
npx cap add ios
npx cap sync ios
```

1. Abre Xcode: `npx cap open ios`.
2. Activa el **Modo Avión** en tu iPhone físico (apaga Wi-Fi y Bluetooth a mano) y conéctalo por cable USB.
3. Elige tu iPhone real en la barra de herramientas de arriba.
4. Limpia los archivos temporales con **Product -> Clean Build Folder**.
5. Presiona el botón **Play**.

Cómo probar:

Al abrir la App, verás la estimación de tu procesador, la resolución de pantalla, tus configuraciones de iOS y tu ubicación aproximada en el mapa por IP. Al presionar el botón azul «**DESBLOQUEAR KINETICS**» y otorgar el acceso a WebKit, todas las métricas espaciales del acelerómetro y las fuerzas G empezarán a reaccionar al instante con el más mínimo movimiento de tu iPhone.