Projet_VAE_Nice\index.html

```
// Interface Web Simple (HTML/CSS/JS) pour le tableau de bord des alertes NurseCall
<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>NurseCall - Tableau de Bord</title>
    <style>
        body {
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            margin: 0;
            padding: 20px;
            background: #f5f5f5;
        }

        .container {
            max-width: 1200px;
            margin: 0 auto;
        }

        .header {
            background: #2c3e50;
            color: white;
            padding: 20px;
            border-radius: 10px;
            margin-bottom: 20px;
        }

        .alertes-grid {
            display: grid;
            grid-template-columns: repeat(auto-fill, minmax(300px, 1fr));
            gap: 15px;
        }

        .alerte-card {
            background: white;
            border-radius: 8px;
            padding: 15px;
            box-shadow: 0 2px 5px rgba(0,0,0,0.1);
            border-left: 5px solid #3498db;
        }

        .alerte-card.priorite-3 {
            border-left-color: #e74c3c;
            background: #ffeaea;
        }

        .alerte-card.priorite-2 {
            border-left-color: #f39c12;
            background: #fff4e6;
        }
```

```
        .alerte-card.priorite-1 {
            border-left-color: #3498db;
        }

        .chambre {
            font-size: 24px;
            font-weight: bold;
            color: #2c3e50;
        }

        .besoin {
            font-size: 18px;
            margin: 10px 0;
        }

        .timestamp {
            color: #7f8c8d;
            font-size: 14px;
        }

        button {
            background: #2ecc71;
            color: white;
            border: none;
            padding: 10px 15px;
            border-radius: 5px;
            cursor: pointer;
            font-weight: bold;
            margin-top: 10px;
        }

        button:hover {
            background: #27ae60;
        }

        .stats {
            display: flex;
            gap: 20px;
            margin-bottom: 20px;
            flex-wrap: wrap;
        }

        .stat-card {
            background: white;
            padding: 15px;
            border-radius: 8px;
            flex: 1;
            min-width: 150px;
            text-align: center;
            box-shadow: 0 2px 5px rgba(0,0,0,0.1);
        }

        .stat-value {
```

```
            font-size: 32px;
            font-weight: bold;
            color: #2c3e50;
        }

        .stat-label {
            color: #7f8c8d;
            font-size: 14px;
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="header">
            <h1>🚨 NurseCall - Tableau de Bord</h1>
            <p>Alertes en temps réel - Hôpital Saint-Vincent</p>
        </div>

        <div class="stats">
            <div class="stat-card">
                <div class="stat-value" id="total-alertes">0</div>
                <div class="stat-label">Alertes totales</div>
            </div>
            <div class="stat-card">
                <div class="stat-value" id="alertes-actives">0</div>
                <div class="stat-label">En attente</div>
            </div>
            <div class="stat-card">
                <div class="stat-value" id="terminaux-actifs">0</div>
                <div class="stat-label">Terminaux actifs</div>
            </div>
        </div>

        <h2>📋 Alertes en cours</h2>
        <div class="alertes-grid" id="alertes-container">
            <!-- Rempli par JavaScript -->
        </div>
    </div>

    <script>
        let soignantId = 'inf_' + Math.floor(Math.random() * 1000);

        function getBesoinText(code) {
            const besoins = {
                1: '😵 Douleur',
                2: '🚽 Toilette',
                3: '🔄 Repositionnement',
                4: '❓ Question',
                5: '💧 Eau',
                99: '🚨 URGENCE'
            };
            return besoins[code] || 'Besoin inconnu';
        }

```

```
        function getPrioriteText(code) {
            return ['Basse', 'Normale', 'Haute', 'Urgence'][code] || 'Inconnue';
        }

        async function chargerAlertes() {
            try {
                const response = await fetch('/api/alertes/en-cours');
                const alertes = await response.json();

                document.getElementById('alertes-actives').textContent =
                    Object.keys(alertes).length;

                const container = document.getElementById('alertes-container');
                container.innerHTML = '';

                for (const key in alertes) {
                    const alerte = alertes[key];

                    const card = document.createElement('div');
                    card.className = `alerte-card priorite-${alerte.priorite}`;

                    const besoinText = getBesoinText(alerte.besoin);
                    const prioriteText = getPrioriteText(alerte.priorite);

                    card.innerHTML = `
                        <div class="chambre">Chambre ${alerte.chambre}</div>
                        <div class="besoin">${besoinText}</div>
                        <div class="priorite">Priorité: ${prioriteText}</div>
                        <div class="timestamp">${new
Date(alerte.timestamp).toLocaleTimeString()}</div>
                        <button onclick="prendreEnCharge(${alerte.id})">
                            👨‍⚕️ Prendre en charge
                        </button>
                    `;

                    container.appendChild(card);
                }
            } catch (error) {
                console.error('Erreur chargement alertes:', error);
            }
        }

        async function prendreEnCharge(alerteId) {
            try {
                await fetch(`/api/alerte/${alerteId}/prendre`, {
                    method: 'POST',
                    headers: {'Content-Type': 'application/json'},
                    body: JSON.stringify({ soignant: soignantId })
                });

                alert(`Alerte prise en charge par ${soignantId}`);
                chargerAlertes();
            } catch (error) {
                console.error('Erreur prise en charge:', error);
```

```
            }
        }

        // Actualisation automatique
        setInterval(chargerAlertes, 3000);
        chargerAlertes();

        // Simuler quelques terminaux actifs
        setInterval(() => {
            document.getElementById('terminaux-actifs').textContent =
                Math.floor(45 + Math.random() * 10);
        }, 10000);
    </script>
</body>
</html>

```