<!DOCTYPE html>

<html lang="fr">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<title>GUAI BORDEL'SS 2025 - Résultats & Phase finale</title>

<style>

body {

font-family: Arial, sans-serif;

margin:0; padding:0;

background:#f4f4f4;

color:#333;

}

header {

background:#2c3e50;

color:#fff;

text-align:center;

padding:1rem;

}

img.logo {

max-height:80px;

vertical-align:middle;

}

nav {

background:#34495e;

display:flex;

justify-content:center;

flex-wrap:wrap;

padding:0.5rem;

}

nav button {

margin:5px;

padding:10px 20px;

font-size:16px;

background:#ecf0f1;

border:none;

border-radius:5px;

cursor:pointer;

transition:background-color 0.3s;

}

nav button:hover, nav button.active {

background:#bdc3c7;

}

.content {

max-width:1000px;

margin:auto;

padding:1rem;

}

.section {

display:none;

}

.section.active {

display:block;

}

.sports-buttons, .poule-buttons {

display:flex;

justify-content:center;

flex-wrap:wrap;

margin-bottom:1rem;

}

.sport-button, .poule-button {

margin:5px 10px;

padding:8px 15px;

font-size:18px;

cursor:pointer;

background:#ecf0f1;

border:none;

border-radius:5px;

transition:background-color 0.3s;

}

.sport-button:hover, .poule-button:hover,

.sport-button.active, .poule-button.active {

background:#bdc3c7;

}

table {

width:100%;

border-collapse:collapse;

background:#fff;

box-shadow:0 2px 4px rgba(0,0,0,0.1);

margin-bottom:1rem;

}

th, td {

padding:8px;

border:1px solid #ddd;

text-align:center;

}

th {

background:#34495e;

color:#fff;

}

.message {

text-align:center;

font-style:italic;

margin-top:1rem;

color:#666;

}

</style>

<script src="https://cdnjs.cloudflare.com/ajax/libs/PapaParse/5.4.1/papaparse.min.js"></script>

</head>

<body>

<header>

<img src="logo.png" alt="Logo" class="logo" />

<h1>GUAI BORDEL'SS 2025</h1>

</header>

<nav>

<button class="nav-btn active" onclick="showSection('accueil', event)">Accueil</button>

<button class="nav-btn" onclick="showSection('resultats', event)">Résultats</button>

<button class="nav-btn" onclick="showSection('phase-finale', event)">Phase finale</button>

<button class="nav-btn" onclick="showSection('contacts', event)">Contacts urgence</button>

</nav>

<div class="content">

<!-- Accueil -->

<section id="accueil" class="section active">

<h2>Bienvenue !</h2>

<p>Retrouvez toutes les informations des GUAI BORDEL'SS 2025.</p>

</section>

<!-- Résultats -->

<section id="resultats" class="section">

<h2>Résultats des matchs</h2>

<div class="sports-buttons" id="buttons-resultats"></div>

<div class="poule-buttons" id="buttons-poules-resultats"></div>

<div id="container-resultats" class="message">Veuillez choisir un sport et une poule.</div>

</section>

<!-- Phase finale -->

<section id="phase-finale" class="section">

<h2>Phase finale</h2>

<div class="sports-buttons" id="buttons-phase-finale"></div>

<div id="container-phase-finale" class="message">Veuillez choisir un sport.</div>

</section>

<!-- Contacts urgence -->

<section id="contacts" class="section">

<h2>Contacts d'urgence</h2>

<div id="contacts-container" class="message">Chargement en cours...</div>

</section>

</div>

<script>

const sports = {

handball: {

nom: "Handball",

emoji: "🤾‍♂️",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=0&single=true&output=csv"

},

rugby: {

nom: "Rugby",

emoji: "🏉",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=702928165&single=true&output=csv"

},

FootM: {

nom: "Foot M",

emoji: "⚽",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=35127129&single=true&output=csv"

},

VolleyM: {

nom: "Volley M",

emoji: "🏐",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=1516588087&single=true&output=csv"

},

FootF: {

nom: "Foot F",

emoji: "⚽",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=504548277&single=true&output=csv"

},

BasketM: {

nom: "Basket M",

emoji: "🏀",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=896073508&single=true&output=csv"

},

BasketF: {

nom: "Basket F",

emoji: "🏀",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=254686168&single=true&output=csv"

},

VolleyF: {

nom: "Volley F",

emoji: "🏐",

urlCSV: "https://docs.google.com/spreadsheets/d/e/2PACX-1vRWrOq4N-KeChvdjO4JaEyMZVm-e3G1PNxbX-gEhC7xMDo1Y2dV6-Ax28BOJ31voFSPHhR\_dNHrDzIT/pub?gid=1381709457&single=true&output=csv"

}

};

function showSection(id, event) {

document.querySelectorAll('.section').forEach(s => s.classList.remove('active'));

document.getElementById(id).classList.add('active');

document.querySelectorAll('nav button').forEach(b => b.classList.remove('active'));

event.target.classList.add('active');

}

function createSportButtons() {

const btnsResultats = document.getElementById('buttons-resultats');

const btnsPhaseFinale = document.getElementById('buttons-phase-finale');

Object.entries(sports).forEach(([key, sport]) => {

const btn1 = document.createElement('button');

btn1.className = 'sport-button';

btn1.textContent = `${sport.emoji} ${sport.nom}`;

btn1.onclick = () => showPoules(key, sport.urlCSV);

btnsResultats.appendChild(btn1);

const btn2 = document.createElement('button');

btn2.className = 'sport-button';

btn2.textContent = `${sport.emoji} ${sport.nom}`;

btn2.onclick = () => showPhaseFinale(key, sport.urlCSV);

btnsPhaseFinale.appendChild(btn2);

});

}

function showPoules(sportKey, csvURL) {

const poules = ['Poule A', 'Poule B'];

const btns = document.getElementById('buttons-poules-resultats');

btns.innerHTML = '';

poules.forEach(poule => {

const btn = document.createElement('button');

btn.className = 'poule-button';

btn.textContent = poule;

btn.onclick = () => {

// Marquer le bouton actif

document.querySelectorAll('.poule-button').forEach(b => b.classList.remove('active'));

btn.classList.add('active');

fetchAndDisplayWithClassement(csvURL, poule, 'container-resultats');

};

btns.appendChild(btn);

});

document.getElementById('container-resultats').innerHTML = "Veuillez choisir une poule.";

}

function fetchAndDisplayWithClassement(url, pouleName, containerId) {

fetch(url)

.then(resp => resp.text())

.then(csv => {

const parsed = Papa.parse(csv.trim(), { skipEmptyLines: true }).data;

const headers = parsed[0];

const rows = parsed.slice(1).filter(row => row[headers.indexOf('Type de match')] === pouleName);

const matchs = rows.map(row => ({

equipeA: row[headers.indexOf('Équipe A')],

equipeB: row[headers.indexOf('Équipe B')],

scoreA: parseInt(row[headers.indexOf('Score A')]),

scoreB: parseInt(row[headers.indexOf('Score B')])

}));

// Calcul classement

const teams = {};

matchs.forEach(m => {

[m.equipeA, m.equipeB].forEach(e => {

if (!teams[e]) teams[e] = { pts: 0, joues: 0, diff: 0, marque: 0 };

});

teams[m.equipeA].joues++;

teams[m.equipeB].joues++;

teams[m.equipeA].marque += m.scoreA;

teams[m.equipeB].marque += m.scoreB;

teams[m.equipeA].diff += m.scoreA - m.scoreB;

teams[m.equipeB].diff += m.scoreB - m.scoreA;

if (m.scoreA > m.scoreB) teams[m.equipeA].pts += 3;

else if (m.scoreA < m.scoreB) teams[m.equipeB].pts += 3;

else {

teams[m.equipeA].pts += 1;

teams[m.equipeB].pts += 1;

}

});

let classement = Object.entries(teams).map(([equipe, stats]) => ({ equipe, ...stats }));

classement.sort((a, b) => b.pts - a.pts || b.diff - a.diff || b.marque - a.marque);

const container = document.getElementById(containerId);

container.innerHTML = '';

// Tableau classement

let htmlClassement = `<h3>Classement - ${pouleName}</h3>

<table>

<thead>

<tr>

<th>Rang</th><th>Équipe</th><th>Pts</th><th>J</th><th>Diff</th><th>Marqué</th>

</tr>

</thead>

<tbody>

${classement.map((c, i) => `<tr>

<td>${i+1}</td>

<td>${c.equipe}</td>

<td>${c.pts}</td>

<td>${c.joues}</td>

<td>${c.diff}</td>

<td>${c.marque}</td>

</tr>`).join('')}

</tbody>

</table>`;

// Tableau résultats

let htmlMatchs = `<h3>Résultats - ${pouleName}</h3>

<table>

<thead>

<tr>

<th>Équipe A</th><th>Score A</th><th>Score B</th><th>Équipe B</th>

</tr>

</thead>

<tbody>

${matchs.map(m => `<tr>

<td>${m.equipeA}</td><td>${m.scoreA}</td><td>${m.scoreB}</td><td>${m.equipeB}</td>

</tr>`).join('')}

</tbody>

</table>`;

container.innerHTML = htmlClassement + htmlMatchs;

})

.catch(err => {

document.getElementById(containerId).textContent = "Erreur lors du chargement des données.";

console.error(err);

});

}

function showPhaseFinale(sportKey, csvURL) {

fetch(csvURL)

.then(resp => resp.text())

.then(csv => {

const parsed = Papa.parse(csv.trim(), { skipEmptyLines: true }).data;

const headers = parsed[0];

const rows = parsed.slice(1);

// On récupère les résultats des poules A et B (poule A et B)

const getRowsByPoule = (poule) => rows.filter(r => r[headers.indexOf('Type de match')] === poule);

// Calcul classement poule A

const classementA = calculClassement(getRowsByPoule('Poule A'), headers);

const classementB = calculClassement(getRowsByPoule('Poule B'), headers);

// Récupérer les résultats des matchs de phase finale s'ils existent (demi-finales, petite finale, finale)

const getByType = (type) => rows.filter(r => r[headers.indexOf('Type de match')] === type);

// Texte des matchs

const getScoreText = (type) => {

const match = getByType(type)[0];

if (match) {

const eA = match[headers.indexOf('Équipe A')];

const eB = match[headers.indexOf('Équipe B')];

const sA = match[headers.indexOf('Score A')];

const sB = match[headers.indexOf('Score B')];

return `${eA} ${sA} - ${sB} ${eB}`;

}

return "À venir";

};

const df1 = getScoreText("Demi-finale 1") !== "À venir"

? getScoreText("Demi-finale 1")

: `${classementA[0]?.equipe || '?'} vs ${classementB[1]?.equipe || '?'}`;

const df2 = getScoreText("Demi-finale 2") !== "À venir"

? getScoreText("Demi-finale 2")

: `${classementB[0]?.equipe || '?'} vs ${classementA[1]?.equipe || '?'}`;

const petiteFinale = getScoreText("Petite finale");

const finale = getScoreText("Finale");

const container = document.getElementById('container-phase-finale');

container.innerHTML = `

<div><strong>Demi-finale 1 :</strong><br>${df1}</div><br>

<div><strong>Demi-finale 2 :</strong><br>${df2}</div><br>

<div><strong>Petite finale :</strong><br>${petiteFinale}</div><br>

<div><strong>Finale :</strong><br>${finale}</div><br>

`;

})

.catch(err => {

document.getElementById('container-phase-finale').textContent = "Erreur lors du chargement des données.";

console.error(err);

});

}

function calculClassement(rows, headers) {

const teams = {};

rows.forEach(row => {

const equipeA = row[headers.indexOf('Équipe A')];

const equipeB = row[headers.indexOf('Équipe B')];

const scoreA = parseInt(row[headers.indexOf('Score A')]);

const scoreB = parseInt(row[headers.indexOf('Score B')]);

[equipeA, equipeB].forEach(e => {

if (!teams[e]) teams[e] = { pts: 0, joues: 0, diff: 0, marque: 0, equipe: e };

});

teams[equipeA].joues++;

teams[equipeB].joues++;

teams[equipeA].marque += scoreA;

teams[equipeB].marque += scoreB;

teams[equipeA].diff += scoreA - scoreB;

teams[equipeB].diff += scoreB - scoreA;

if (scoreA > scoreB) teams[equipeA].pts += 3;

else if (scoreA < scoreB) teams[equipeB].pts += 3;

else {

teams[equipeA].pts += 1;

teams[equipeB].pts += 1;

}

});

const classement = Object.values(teams);

classement.sort((a, b) => b.pts - a.pts || b.diff - a.diff || b.marque - a.marque);

return classement;

}

// Charger les contacts d'urgence depuis la Google Sheet

function loadContactsUrgence() {

const urlContacts = "https://docs.google.com/spreadsheets/d/e/2PACX-1vQVfiZHnma6ia55oyYBiQwl-s-m-krOtKAaNZEg2kURaFlSy1tamOQ-CDYFGVr6WHzdmNBRO4Kms-am/pub?gid=1955615812&single=true&output=csv";

fetch(urlContacts)

.then(resp => resp.text())

.then(csv => {

const parsed = Papa.parse(csv.trim(), { skipEmptyLines: true }).data;

const headers = parsed[0];

const rows = parsed.slice(1);

const table = document.createElement('table');

const thead = document.createElement('thead');

const tbody = document.createElement('tbody');

// En-têtes

const headRow = document.createElement('tr');

headers.forEach(h => {

const th = document.createElement('th');

th.textContent = h;

headRow.appendChild(th);

});

thead.appendChild(headRow);

// Lignes

rows.forEach(row => {

const tr = document.createElement('tr');

row.forEach(cell => {

const td = document.createElement('td');

td.textContent = cell;

tr.appendChild(td);

});

tbody.appendChild(tr);

});

table.appendChild(thead);

table.appendChild(tbody);

const container = document.getElementById('contacts-container');

container.innerHTML = '';

container.appendChild(table);

})

.catch(err => {

document.getElementById('contacts-container').innerHTML = "Erreur de chargement.";

console.error("Erreur chargement contacts :", err);

});

}

// Charger les contacts au démarrage

document.addEventListener('DOMContentLoaded', loadContactsUrgence);

// Initialisation des boutons à l'ouverture de la page

createSportButtons();

</script>

</body>

</html>