

Livrable Technique Code du site

Choix de Langages :

Le choix des langages s'est fait dans le but de simplifier la conception afin de fournir un résultat remplissant les conditions de départ dans les plus brefs délais tout en proposant une solution esthétique et fonctionnelle. Pour mener à bien ce projet les langages utilisés sont donc le **HTML 5** afin de construire la structure de notre code, le **CSS** pour rendre le tout plus esthétique et attractif ainsi que le **Java Script** pour permettre de rendre la page interactive et lui permettre de faire des calculs.

HTML :

Nous sommes partis sur une base de page en 3 compartiments afin de rendre l'utilisation sur téléphone plus simple, avec une colonne centrale qui propose la sélection de l'année ainsi que les informations qui en découle. Pour les deux autres colonnes elles permettent l'ajout de fonctionnalité disponible seulement sur Ordinateurs pour ne pas surcharger la vue sur téléphone.

CSS :

Le CSS quant à lui nous a permis la mise en forme de notre solution pour proposer un site agréable à l'œil et adapter au thème associé grâce à l'ajout de photos en arrière-plan et d'image/vidéo sur les côtés ? Nous avons aussi ajouté une fonctionnalité de jeu qui grâce au CSS permet de maintenir les utilisateurs sur la page.

Java Script :

La partie en JS, ou Java Script, nous a permis de donner du volume à notre site grâce à un algorithme de calcul des années bissextiles, un algorithme de calcul des fêtes mobiles avec un affichage dynamique. Deux un deuxième temps le JS a permis la mise en place du clicker et ses différentes fonctionnalités.

Fonctionnalités du code :

Pour plus d'informations sur le code vous pouvez le consulter avec le lien suivant :

<https://github.com/Axel-plt/PROJET-VIERGE>

Algorithme de calcul des années bissextiles :

```
function isBissextile(year) {  
  if (year === 0) {  
    // Si l'année est 0, on retourne false ici, mais la logique de texte sera gérée plus tard  
    return null; // Utilisation de null pour signaler un cas spécial (année 0)  
  }  
  if (year < -46) {  
    // Les années avant -46 ne sont jamais bissextiles  
    return false;  
  }  
  if (year <= 1582) {  
    // Avant ou en 1582, les années de siècles (divisibles par 100) sont aussi bissextiles,  
    // et toutes les années divisibles par 4 sont bissextiles (y compris l'année 4)  
    return year % 4 === 0 || year % 100 === 0;  
  } else {  
    // Après 1582, appliquer la règle moderne  
    return (year % 4 === 0 && year % 100 !== 0) || year % 400 === 0;  
  }  
}
```

Cet algorithme permet d'exclure l'an 0 qui n'existe pas ainsi que d'enlever les années avant l'an -46 du calcul car l'histoire nous apprend qu'avant l'an -46 il n'y avait pas d'années bissextile. Pour finir l'histoire nous apprend qu'avant 1582 les années de siècle été aussi bissextile par conséquent une condition est ajoutée.

Algorithme de calcul de la date de Paques :

```
function calculerPaques(year) {  
    // Si l'année est inférieure à 100, la traiter comme une année du 20ème siècle  
    if (year < 100) {  
        year += 1900;  
    }  
  
    const a = year % 19;  
    const b = Math.floor(year / 100);  
    const c = year % 100;  
    const d = Math.floor(b / 4);  
    const e = b % 4;  
    const f = Math.floor((b + 8) / 25);  
    const g = Math.floor((b - f + 1) / 3);  
    const h = (19 * a + b - d - g + 15) % 30;  
    const i = Math.floor(c / 4);  
    const k = c % 4;  
    const l = (32 + 2 * e + 2 * i - h - k) % 7;  
    const m = Math.floor((a + 11 * h + 22 * l) / 451);  
    const month = Math.floor((h + 1 - 7 * m + 114) / 31);  
    const day = ((h + 1 - 7 * m + 114) % 31) + 1;  
  
    // Retourner la date corrigée (le mois commence à 0 dans JavaScript, donc on soustrait 1 du mois)  
    return new Date(year, month - 1, day);  
}
```

Cet algorithme calcul la date de paques et par conséquent nous permet de calculer la date des fêtes mobiles par la suite car celle-ci découle de Pâques.

Algorithme de calcul des fêtes mobiles :

```
function calculerFetesMobiles(year) {  
    const paques = calculerPaques(year);  
  
    function addDays(date, days) {  
        const result = new Date(date);  
        result.setDate(result.getDate() + days);  
        return result;  
    }  
  
    return {  
        'Mardi Gras': addDays(paques, -47),  
        'Mercredi des Cendres': addDays(paques, -46),  
        'Dimanche des Rameaux': addDays(paques, -7),  
        'Jeudi Saint': addDays(paques, -3),  
        'Vendredi Saint': addDays(paques, -2),  
        'Pâques': paques,  
        'Lundi de Pâques': addDays(paques, 1),  
        'Divine Miséricorde': addDays(paques, 7),  
        'Jeudi de l'Ascension': addDays(paques, 39),  
        'Pentecôte': addDays(paques, 49),  
        'Lundi de Pentecôte': addDays(paques, 50),  
        'Sainte Trinité': addDays(paques, 56),  
        'Fête-Dieu': addDays(paques, 60)  
    };  
}
```

À partir de ces trois Algorithmes nous avons réussi à ressortir les informations liées aux dates des fêtes mobiles.