

BTS SIO 2022

Support et mise à disposition de services informatiques (E4)

PAGE DE PRÉSENTATION DU DOSSIER

N° de candidat¹ : |__|__|__|__|__|__|__|__|__|__|__|__|

NOM : Zatric

Prénom : Nihad

Date de passage ¹ : / /2022	Heure de passage ¹ :h.....
--	---

CATEGORIE CANDIDAT ² (UNE CASE A COCHER)	
<input type="checkbox"/> Scolaire <input checked="" type="checkbox"/> Apprenti <input type="checkbox"/> Formation professionnelle continue <input type="checkbox"/> Expérience professionnelle 3 ans	<input type="checkbox"/> Ex-scolaire <input type="checkbox"/> Ex-apprenti <input type="checkbox"/> Ex-formation professionnelle continue

¹ Informations communiquées sur votre convocation envoyée en mars-avril 2022

² Informations communiquées sur votre confirmation d'inscription

Tampon de
l'établissement



BTS SIO – Dossier Etudiant
Justificatif d'acquisition des compétences

Epreuve E4
Support et mise à disposition de
services informatiques

SOMMAIRE

- 1 INTRODUCTION 4
- 2 MISSION 1 : AJOUT DE RUPTURES..... 5
 - 2.1 Cahier des charges..... 5
 - 2.2 Etude et conception de la solution 6
 - 2.3 Gestion de projet 7
 - 2.4 Mise en œuvre..... 8
 - 2.5 Bilan 11

1 Introduction

L'objectif de ce document est de vous présenter les missions professionnelles que j'ai effectué dans le cadre de ma formation BTS SIO à l'école IRIS de Strasbourg.

Ces missions peuvent être de trois types :

- Effectuées en entreprise durant une alternance
- Effectuées en stage en entreprise
- Effectuées à l'école (compte-rendu de TP, projet collaboratif)

Le type de la mission sera précisé dans chaque cahier des charges.

Ce document se compose des parties suivantes :

Chapitres	Contenu
Chapitres 2 à ...	<p>Présentation des missions, avec pour chacune :</p> <ul style="list-style-type: none">- Le cahier des charges- La solution proposée- La gestion de projet- La mise en œuvre- Le bilan du projet

2 Mission 1 : Ajout de ruptures

2.1 Cahier des charges

Type de mission
Mission effectuée en alternance.
Contexte
Dans le cadre de la réalisation d'un programme d'échanges d'informations sur les commandes en interne, formulaire permettant à l'utilisateur d'ajouter une référence d'article en rupture.
Demande du client
Formulaire d'ajout de rupture
Expression du besoin
Un champ texte permettant de renseigner la référence de l'article.
Un champ date permettant de renseigner la date de début de la rupture.
Un champ date permettant de renseigner la date de fin de la rupture (approvisionnement).
Budget disponible
Mon cout salarial sur une demi-journée.
Outils disponibles
Un poste avec un serveur local installé et configuré (Wampserver).
Contraintes
Temps : 1/2 journée.
Responsive : non.
Ressources : une demi-journée de travail, seul.
Confidentialité

2.2 Etude et conception de la solution

2.2.1 Les solutions possibles

Langages de programmation :

- Php
- Javascript
- HTML

Frameworks :

- JQuery
- VueJS

Technologies : AJAX

2.2.2 La solution retenue

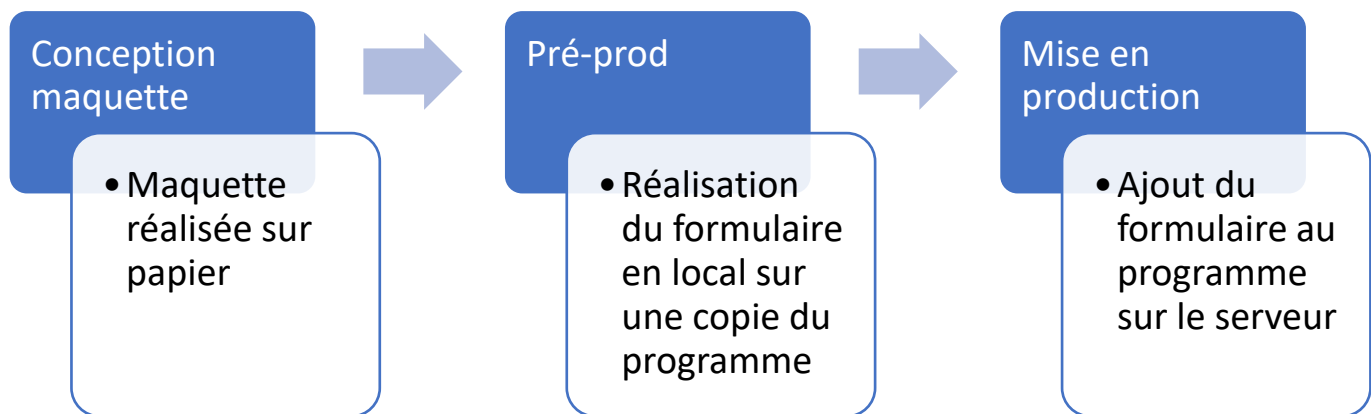
J'ai décidé de réaliser cette mission en PHP procédural, sans framework car la réalisation de ce formulaire ne requiert pas énormément de lignes de codes donc il semble superflu d'intégrer un framework pour si peu.

De plus, je n'ai pas eu besoin de mettre en place de requête asynchrone (AJAX), puisque le contenu de la page est assez léger pour qu'une actualisation n'impacte pas négativement la fluidité du programme.

2.3 Gestion de projet

2.3.1 Planing de déploiement de la solution

Schéma de la réalisation de la mission :



2.3.2 Budget

Matériel : 1 poste avec un serveur installé et configuré (Wampserver).

Licences : PHPStorm.

Coût humain : 1/2 journée de travail pour 1 personne.

2.4 Mise en œuvre

2.4.1 Implémentation de la solution

Tout d'abord, on crée le formulaire au format HTML :

```
<form id="hidden_form_create_rupture" method="post">
  <div class="form-group rupture_form">
    <input type="text" class="rupture_input create_rupture" name="ref_rupture" id="create_rupture_numero_ref">
    <label for="create_rupture_numero_ref" class="label-name rupture_label">
      <span class="content-name">Référence de l'élément en rupture</span>
    </label>
  </div>

  <div class="form-group rupture_form rupture_date_div">
    <p>Date de début de rupture : </p>
    <input class="rupture_date" type="date" id="date_rupture" name="date_rupture">
  </div>

  <div class="form-group rupture_form rupture_date_div">
    <p>Date de réapprovisionnement : </p>
    <input class="rupture_date" type="date" id="date_reapro" name="date_reapro">
  </div>
  <button id="button_annuler_rupture" onclick="annuler_rupture()" type="button" class="btn_creer submit btn">Annuler</button>
  <button id="button_create_rupture" type="button" class="btn_creer submit btn">Créer</button>
</form>
```

Il comprend :

- Un champ texte permettant de renseigner la référence de l'article.
- Un champ date permettant de renseigner la date de début de la rupture.
- Un champ date permettant de renseigner la date de fin de la rupture.

Ensuite, on s'occupe de la réception des données.

On prépare une requête SQL qui va permettre d'effectuer l'insertion des données entrées par l'utilisateur et transmises par ce dernier via la méthode POST.

Afin de protéger l'intégrité des données et éviter toute tentative de piratage (par injection de requête SQL), on utilise la méthode bindParam.

Ensuite, on effectue l'insertion à la base de données de l'article en rupture.

```
$sql = 'INSERT INTO "rupture_entete" (ref, date_debut, date_fin, date_crea) VALUES
(:ref_rupture, :date_rupture, :date_reapro, NOW() )';

$result = $bdd->prepare($sql);
$result->bindParam( param: ':ref_rupture', &var: $_POST['ref_rupture'], type: PDO::PARAM_STR, maxLength: 255);
$result->bindParam( param: ':date_rupture', &var: $_POST['date_rupture']);
$result->bindParam( param: ':date_reapro', &var: $_POST['date_reapro']);
$result->execute();

$result->closeCursor();
```


Résultat :

Éléments jusqu'au 03/01/2022

Nombre de lignes : 670

Meuble envoyé à la sci

Assemblage des meubles

L'article est un meuble

+
○

Liste de ruptures

X

Référence de l'élément en rupture

Date de début de rupture : jj/mm/aaaa

Date de réapprovisionnement : jj/mm/aaaa

Annuler
Créer

Référence de l'article	Date début rupture	Date fin rupture	
CHA.TIO.0110.041.ZA	14/09/2021	21/09/2021	X
CHA.TIO.0110.041.ZA	14/09/2021	21/09/2021	X
SXPLANV3.SM110-120.SBM	02/09/2021	31/12/9999	X
HZPLAN70VBS	08/09/2021	15/09/2021	X
BG.LUNA01.BLB	21/01/2022	31/12/9999	X
HZPLAN140VCS	30/08/2021	15/09/2021	X
RLED01.SM2.NM	01/09/2021	31/12/9999	X
PAN.18.2800.2070.020.VCM	01/06/2021	16/09/2021	X

2.4.2 Recettage

Test fonctionnel

Contexte : Je me mets à la place de l'utilisateur, je renseigne une référence, une date de début de rupture et une date de fin de rupture :

Liste de ruptures

Référence de l'élément en rupture

CHA.EMBA.TIO.H00.NI

Date de début de rupture : 02/03/2022

Date de réapprovisionnement : 10/03/2022

AnnulerCréer

Date début rupture	Référence de l'article	Date fin rupture
--------------------	------------------------	------------------

Une fois que je valide le formulaire, la page est rechargée et la ligne est ajoutée au bloc sous le formulaire. Ce bloc étant généré via une requête SQL renvoyant toutes les ruptures en base de données, c'est la preuve que la requête d'insertion a bien été effectuée :

Liste de ruptures

Référence de l'élément en rupture

Date de début de rupture : jj/mm/aaaa

Date de réapprovisionnement : jj/mm/aaaa

AnnulerCréer

Référence de l'article	Date début rupture	Date fin rupture	
CHA.EMBA.TIO.H00.NI	02/03/2022	10/03/2022	

2.5 Bilan

2.5.1 Validation des exigences point par point

Création des trois champs ☒

Insertion à la base de données ☒

2.5.2 Axes d'amélioration

Requête en AJAX.

2.5.3 Compétences acquises

Maîtrise des requêtes de type POST.