

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 1 sur 9</b>

**Cahier des charges**  
Stage en entreprise

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 2 sur 9</b>

## Sommaire :

- [Mise en place d'un formulaire d'ajout de partenaire.](#)
- [Mise en place d'un projet d'entraînement PHP \(Jeu de Bataille\).](#)
- [Mise en place d'un système de lecture et de récupération de données d'un tableau excel](#)
- [Développement d'une API en C# pour la synchronisation de bases de données :](#)
- [Gestion des Sociétés Partenaires \(Sage & MySQL\)](#)
- [Gestion et synchronisation des Anciennes Données Sage vers MySQL](#)

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 3 sur 9</b>

## 1 : Mise en place d'un formulaire d'ajout de partenaire :

**Objectif :** Mettre en place un formulaire d'ajout de partenaire pour l'entreprise sur une application web sous Symfony 7 et ajouter un stockage des données dans une base MariaDB.

### 1. Livrables :

- Un formulaire fonctionnel permettant l'ajout d'un partenaire.
- Un stockage sécurisé des données collectées dans MariaDB.
- Un chiffrement des informations sensibles via bcrypt.
- Une interface utilisateur développée en React.js.


### 2. Contraintes :


- Respect du RGPD pour la gestion des données personnelles.
- Limitation des champs à un nombre maximal de caractères (ex : Siret limité à 15 caractères).
- Intégration avec les systèmes existants (base de données, back-end Symfony 7).
- Sécurisation des données sensibles par chiffrement (bcrypt).

### 3. Validation et Tests :

- Vérification de la conformité au RGPD.
- Tests d'intégration avec la base de données MariaDB.
- Validation du fonctionnement du chiffrement des données sensibles.
- Tests d'ergonomie et d'accessibilité de l'interface utilisateur.

**Conclusion :** Ce projet vise à fournir un système sécurisé et ergonomique permettant l'ajout et la gestion des partenaires de l'entreprise tout en respectant les normes de protection des données.

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 4 sur 9</b>

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 5 sur 9</b>

## 2 : Mise en place d'un projet d'entraînement PHP (Jeu de Bataille) :

**Objectif :** Développer un jeu de carte simple en PHP permettant à deux joueurs de s'affronter en comparant des cartes et en redistribuant celles-ci selon les règles classiques du jeu de bataille.

### 1. Livrables :

- Un système de génération aléatoire des cartes pour chaque joueur.
- Une interface HTML affichant les cartes et les actions à chaque tour.
- Un mécanisme de comparaison et de redistribution des cartes.
- Un système détectant la fin de partie lorsqu'un joueur n'a plus de cartes.


### 2. Contraintes :

- Respect des bonnes pratiques PHP.
- Architecture orientée objet (classes pour les joueurs, les cartes et la gestion du jeu).
- Code extensible pour permettre des ajouts futurs (variantes de règles, interface graphique avancée, etc.).

### 3. Validation et Tests :

- Tests unitaires des classes et des fonctionnalités de jeu.
- Validation de l'affichage des cartes et des actions en HTML.
- Vérification du bon déroulement d'une partie complète.
- Analyse de performance et optimisation du code.

**Conclusion :** Ce projet permettra de renforcer mes compétences en PHP en mettant en place un système de jeu respectant les principes de la POO et les bonnes pratiques de développement.

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 6 sur 9</b>

### 3 : Mise en place d'un système de lecture et de récupération de données d'un tableau excel:

**Objectif :** Développer un outil permettant d'extraire et d'exploiter les données contenues dans des fichiers Excel au format .xlsx à l'aide de la bibliothèque PHPSpreadsheet.

#### 1. Livrables :

- Un système de lecture des fichiers .xlsx et d'extraction des données.
- Une intégration des données récupérées dans une base de données.
- Un module de validation du format et de gestion des erreurs.


#### 2. Contraintes :

- Support exclusif des fichiers .xlsx.
- Vérification du format du fichier avant traitement.
- Gestion des erreurs en cas de fichier corrompu ou inexistant.
- Désactivation de l'exécution de macros pour des raisons de sécurité.

#### 3. Validation et Tests :

- Vérification de la compatibilité des fichiers .xlsx.
- Tests de robustesse face aux fichiers corrompus ou incorrects.
- Validation de l'intégration des données extraites dans la base de données.
- Tests unitaires et fonctionnels du module.

**Conclusion :** Ce projet vise à fournir un outil efficace et sécurisé permettant de traiter des fichiers Excel et d'intégrer leurs données dans un système d'information tout en garantissant la fiabilité du traitement.

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 7 sur 9</b>

#### **4 : Développement d'une API en C# pour la synchronisation de bases de données :**

**Objectif :** Utiliser une API en C# permettant de connecter et synchroniser plusieurs bases de données, incluant MSSQL, MariaDB, PostgreSQL et Sage.

##### **1. Livrables :**

- Une API fonctionnelle en C# permettant la synchronisation des bases de données.
- Un système de gestion des connexions et d'exécution des requêtes.


##### **2. Contraintes :**

- Complexité accrue liée à l'intégration de Sage.
- Adaptation difficile du code existant développé par le maître de stage.
- Contraintes techniques propres à l'API Sage.
- Manque de temps ayant conduit à l'abandon du projet en l'état.

##### **3. Validation et Tests :**

- Vérification de la connexion avec chaque type de base de données.
- Tests de performance sur les opérations de synchronisation.
- Identification et résolution des problèmes liés à l'API Sage.
- Validation du bon fonctionnement des requêtes et de la cohérence des données synchronisées.

**Conclusion :** Ce projet vise à utiliser une API robuste et extensible permettant la synchronisation efficace de bases de données hétérogènes, bien que des défis techniques et temporels aient freiné sa finalisation.

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	Prenom : Joshua	Nom : BETHOULE-VOISIN	Page 8 sur 9

## 5 : Gestion des Sociétés Partenaires (Sage & MySQL) :

### Objectif :

Mettre en place un système de gestion des sociétés partenaires avec synchronisation et vérification entre les bases de données Sage et MySQL.

### 1. Livrables :

- Un système de liste des sociétés partenaires.
- Une vue Twig fonctionnelle pour l'affichage des sociétés.
- Un système de vérification de l'implémentation des sociétés dans Sage et MySQL.
- Un projet de synchronisation des sociétés entre Sage et MySQL.

### 2. Contraintes :


- Base de données MySQL pour le stockage des données.
- Intégration avec Sage.
- Développement des interfaces avec Twig.
- Respect des bonnes pratiques de développement et de sécurité.

### 3. Validation et Tests :

- Vérification de la récupération des données depuis Sage.
- Tests unitaires et fonctionnels pour valider les implémentations.
- Validation de l'affichage et de la synchronisation des sociétés dans MySQL.

### Conclusion :

Ce projet vise à améliorer la gestion des sociétés partenaires en assurant une synchronisation fiable entre Sage et MySQL.

	<b>Cahier des charges</b>		<b>Création :</b> 24/01/2025
	<b>Prenom :</b> Joshua	<b>Nom :</b> BETHOULE-VOISIN	<b>Page 9 sur 9</b>

## 6 : Gestion et synchronisation des Anciennes Données Sage vers MySQL :

### Objectif :

Mettre en place un système de synchronisation massive des anciennes données Sage vers une base MySQL, permettant une migration en une seule opération tout en garantissant l'intégrité des informations.

### 1. Livrables :

- Un script de récupération des données depuis Sage.
- Un module de transformation et préparation des données pour MySQL.
- Une fonction d'insertion des données dans MySQL.
- Une interface d'affichage des données synchronisées via Twig.
- Un système de gestion des erreurs et des logs pour assurer la traçabilité du processus.

### 2. Contraintes :

- Base de données MySQL pour le stockage des données.
- Accès aux bases de données Sage et MySQL avec les privilèges nécessaires.
- Développement en PHP avec le framework Symfony.
- Utilisation de l'EntityManagerInterface pour garantir la persistance des données.

### 3. Validation et Tests :

- Vérification de la correspondance des données entre Sage et MySQL après l'extraction l'insertion.
- Tests de performance sur la rapidité d'exécution de la synchronisation massive.
- Vérification de l'intégrité des données après migration.
- Test de l'interface utilisateur et validation de l'affichage correct des informations.

### Conclusion :

Ce projet vise à garantir une migration fiable et efficace des anciennes données Sage vers MySQL.

En structurant et automatisant le processus, il permet de centraliser l'historique des informations tout en assurant leur exactitude et leur intégrité.

Cette solution s'inscrit dans une démarche d'optimisation du système de gestion des données et pourra être améliorée pour inclure des mises à jour régulières ou des fonctionnalités supplémentaires selon les besoins de l'entreprise.