

UNIVERSITÉ SORBONNE PARIS NORD

IUT de Villetaneuse

Année universitaire 2023-2024

Rapport de Stage

Encadré par PACHIS Stéphane

du 22 Janvier au 15 mars 2024

Présenté en vue d'obtenir le BUT Informatique 2

Market Factory, Baillet-en-France

JOSEPH MARIE Julien



MarketFactory
Communiquer - Innover - Valoriser

Sommaire

Introduction	3
Lexique	4
1er Jour chez Market Factory	4
Calendrier pour Market Factory	5
Un Incident à Market Factory	14
Les Outils de gestion de commande Clients	16
Conclusion	25
Annexes	26

Remerciements :

Je tiens à exprimer ma sincère gratitude envers PACHIS Stéphane pour son accompagnement attentif et ses conseils avisés tout au long de ce stage qui m'ont réellement fait progresser. Mes remerciements vont également à toute l'équipe de Market Factory pour son accueil chaleureux et sa précieuse collaboration qui m'a vraiment attaché à cette entreprise. Je souhaite aussi remercier Mme Zarg Ayouna pour son soutien constant.

Introduction:

Ce rapport est un document qui permet de représenter au mieux mes 280 heures sur deux mois de stage au sein de l'entreprise Market Factory dans le cadre de ma deuxième année de BUT informatique .

Lors de la conférence ayant eu lieu en début d'année, les dates du stage nous ont été transmises par monsieur Audibert, responsable de la formation BUT Informatique. En effet, le stage doit débuter en janvier et prendre fin en mars. Suite à cette conférence, j'ai commencé à effectuer mes recherches de stage dans le domaine de l'informatique. De nombreuses réponses se sont avérées négatives, grâce à l'aide de mon camarade David de formation, j'ai eu l'opportunité d'échanger avec son tuteur de stage ce qui a débouché vers un entretien.

J'ai ensuite tout naturellement adressé une candidature spontanée et grâce à cela j'ai réussi à avoir un entretien qui s'est avéré très favorable. À la suite de cet entretien j'ai réussi à avoir mon stage chez Market Factory qui a débuté le 22 Janvier 2024 et pris fin le 15 Mars 2024.

Market Factory est une entreprise qui a été créée en 2007 par le gérant Stéphane Pachis, cette entreprise est spécialisée dans le marquage textile donc celle-ci s'occupe du flocage, broderie, impression,etc. Cette activité compose la première branche d'activité de market factory car celle-ci possède également une deuxième branche d'activité qui s'occupe de créer des boutiques en ligne gratuitement pour ses clients et les produits disponibles sur cette boutique en ligne sont produits par l'entreprise. L'entreprise est localisée au 1 Avenue du Bosquet à Baillet-en-France.

L'entreprise fonctionne en management à l'horizontale pour garantir que toutes les personnes qui y travaillent sont considérées au même niveau. En termes de structure de l'entreprise, Market Factory compte deux employés. J'ai choisi cette entreprise car j'ai été mis à l'aise par le discours de monsieur Stéphane Pachis.

Grâce à mon stage, J'ai pu faire de grands progrès en développement web et j'ai pu notamment travailler sur des projets passionnants.

Il y a un autre point qui m'a attiré l'attention, en effet contrairement à ce qu'on a appris lors de notre formation, j'ai appris à coder en orienté objet notamment en PHP en utilisant l'architecture MVC.

Plusieurs arguments concernant cette procédure m'ont été donnés par mon tuteur en s'appuyant sur l'exemple sur les stagiaires qu'il a eu auparavant, m'a convaincu davantage d'adopter ceci.

Lors de ce stage, j'ai eu l'opportunité de découvrir de plus près le métier de développeur full stack. Cette expérience m'a aussi sensibilisé aux enjeux de cette profession.

Dans un premier temps, nous aborderons mon premier jour de stage. En deuxième temps, le calendrier que j'ai réalisé pour market factory. Troisièmement, l'incident qui a eu lieu à market factory. Pour finir, je vais vous présenter les Outils de gestion de commande Clients.

Lexique:

Durant mon développement, j'utilise les termes "méthode POST" et "méthode GET" pour transmettre des données entre le client (le navigateur web de l'utilisateur) et le serveur où est hébergée l'application web.

- La méthode POST envoie les données au serveur dans le corps de la requête HTTP, souvent utilisée pour des données sensibles ou volumineuses.
- La méthode GET renvoie les données dans l'URL du navigateur, principalement utilisée pour récupérer des données non sensibles.

Quant aux sessions, elles permettent de stocker des informations utilisateur sur le serveur entre différentes requêtes HTTP, offrant ainsi une manière de maintenir l'état d'une application web pour un utilisateur donné.

1er Jour chez Market Factory

Ma première journée de stage chez Market Factory a été une immersion directe dans le monde professionnel, offrant une expérience concrète et pratique. Accueilli par monsieur Stéphane Pachis, le gérant, ainsi que par l'équipe composée de Sonny Tony et Muriel Celestin. Dans le but d'évaluer mon niveau en développement et pour savoir où j'en suis dans l'apprentissage du langage PHP,SQL,HTML. Mon tuteur m'a confié ma première mission de stage, cette tâche consiste à créer un formulaire qui ajoute des personnes à une table dans une base de données et permet aux personnes de cette table d'être affichés via une liste où à chaque ligne de la liste on a la possibilité de modifier et supprimer la personne de la table.

Pour réaliser cette tâche j'ai noté les trois attributs qui seront utiles pour ma table personne qui sont id_personne,nom,prénom sur une feuille afin de demander à mon tuteur si cela répondait à ses attentes. Après avoir eu sa validation lorsque j'ai voulu faire la table dans la base de données j'ai rencontré mon premier problème lors de mon stage car jusqu'à ce jour j'ai uniquement utilisé la base de données locale qui était présente sur les ordinateurs fixes de l'IUT et donc je savais pas comment accéder à la base de données sur mon ordinateur portable. Grâce à l'aide de mon tuteur j'ai compris que avec le serveur web local que j'ai installé avant de débuter le stage il y avait PHPMYADMIN qui été inclus avec ce serveur web local et donc phpmyadmin me permettait de faire de la gestion de bases de données locales. Après avoir réglé ce problème j'ai créé ma base de données et ma table personne comme mon tuteur l'a suggéré.

Ensuite pour effectuer le formulaire, l'ajout dans la base de donnée et afficher la liste pour modifier et supprimer, c'était une tâche qui était assez simple car on a eu une tâche similaire lors de la SAÉ 3.01 donc j'ai pas eu de difficulté à réaliser cette tâche mis à part le problème lié à la base de données.

Nom:

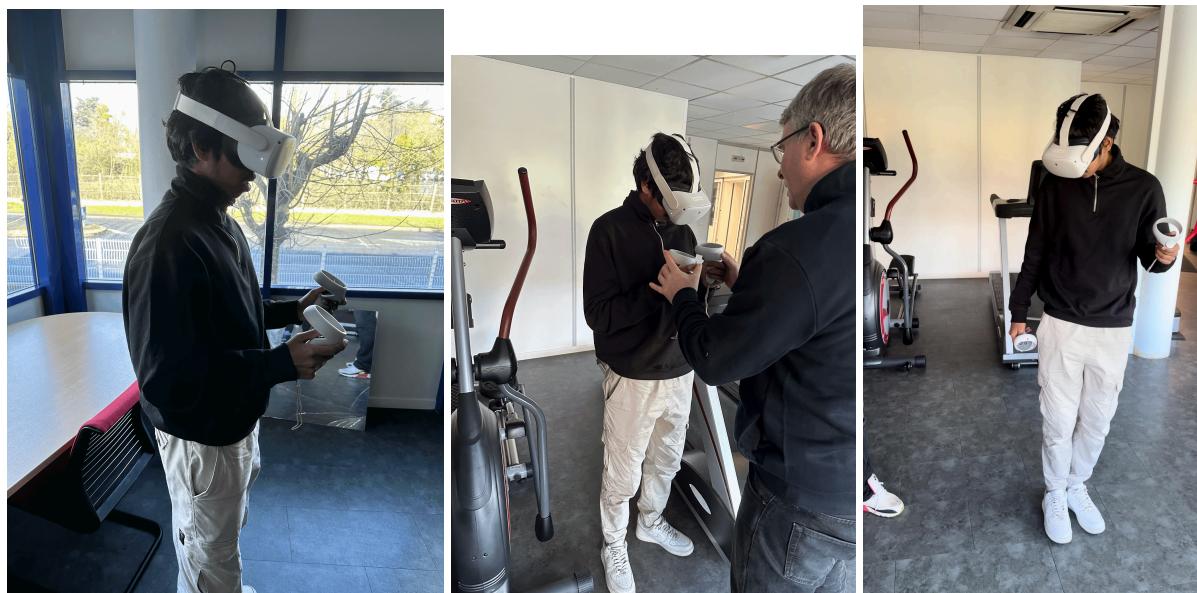
• Julien Jm [ModifierSupprimer](#)

Prénom:

• test test [ModifierSupprimer](#)

Durant cette tâche j'ai codé en procédurale alors que j'ai l'habitude de coder en architecture MVC mais au fur et à mesure je me suis habitué et le fait de coder en procédurale était plus bénéfique pour moi car c'est mon tuteur qui me corrigera lorsque je bloquerai sur quelque chose donc il comprendra mieux alors que si je coder en MVC ça sera plus dur pour de me corriger car il ne maîtrise pas le MVC.

Après avoir fait cette tâche, mon tuteur m'a proposé de faire du Casque Virtuelle, que j'ai tout naturellement accepté car j'en avais jamais fait jusqu'à aujourd'hui.



C'est une expérience très agréable dont j'ai eu l'occasion de faire grâce à mon tuteur, avec une vraie immersion.

Cette première journée de stage a été pour moi une vraie expérience en entreprise, elle a été concrète et m'a donné l'impression d'être un employé de l'entreprise.

J'ai également remarqué que l'atmosphère de l'entreprise était très familiale, ce qui m'a beaucoup plu et a facilité mon intégration. Après cette journée dite "d'adaptation", mon tuteur Pachis m'a informé que par la suite je travaillerais sur un vrai projet qui servira à Market Factory.

Calendrier pour Market Factory:

Mon tuteur m'a confié un projet qui avait pour objectif de faire un calendrier interactif lié à une base de données avec la possibilité d'ajouter/modifier/supprimer des événements qui sont liés à des catégories, ce calendrier devrait avoir une vue visiteur qui enlève certaines fonctionnalités et une vue admin qui en rajoute, ce calendrier doit pouvoir être configurable par l'utilisateur même si il n'a aucune connaissance en informatique. Donc après avoir pris le besoin qui est de pouvoir planifier les tâches des collaborateurs de Market Factory, je suis allé à la rencontre des collaborateurs de Market Factory qui sont les deux employés de l'entreprise pour récupérer leurs demandes pour pouvoir rédiger mon cahier des charges ([voir annexe 1.1](#)) en fonction du besoin de mon maître de stage et des collaborateurs puis apprendre beaucoup plus sur les tables que j'aurais à créer dans la base de données et leur contenu. Donc après avoir caractérisé le besoin j'ai construit ma base de données en fonction du besoin ce qui a donné ceci ci-dessous :

1. Table planning_calendrier :

- **id** : Identifiant unique de l'événement dans le calendrier (clé primaire).
- **title** : Titre de l'événement.
- **start** : Date et heure de début de l'événement.
- **end** : Date et heure de fin de l'événement.
- **categorie** : Identifiant de la catégorie à laquelle l'événement est associé (clé étrangère).
- **valid_tel** : Indicateur de validation pour le téléphone.
- **valid_courriel** : Indicateur de validation pour l'adresse email.
- **full_day** : Indicateur spécifiant si l'événement dure toute la journée.

2. Table planning_utilisateur :

- **id_utilisateur** : Identifiant unique de l'utilisateur (clé primaire).
- **mail** : Adresse email de l'utilisateur.
- **motdepasse** : Mot de passe de l'utilisateur crypté en sha1.
- **role** : Rôle de l'utilisateur dans le système (administrateur ou non).

3. Table planning_categorie :

- **id_categorie** : Identifiant unique de la catégorie (clé primaire).
- **nom_categorie** : Nom de la catégorie.
- **couleur_categorie** : Couleur associée à la catégorie pour la visualisation dans le calendrier.
- **type_categorie** : Type de catégorie (par exemple, événements professionnels, personnels, etc.).
- **valid_tel** : Indicateur de validation pour le téléphone.
- **valid_courriel** : Indicateur de validation pour l'adresse email.

- **courriel_catégorie** : Adresse email associée à la catégorie.
- **tel_catégorie** : Numéro de téléphone associé à la catégorie.

4. Table planning_parametre :

- **id_parametre** : Identifiant unique du paramètre (clé primaire).
- **vue_calendrier_hidden_days** : Jours masqués dans la vue du calendrier (par exemple, week-ends masqués).

Après avoir réalisé la base de données avec mon tuteur, on a décidé de travailler étape par étape pour implémenter chaque fonctionnalité petit à petit et pour bien prendre en main le calendrier.

Dans un premier temps, j'ai eu la présentation de la bibliothèque que je devais utiliser pour réaliser le calendrier qui se nomme fullCalendar qui est une bibliothèque JavaScript open-source permettant de créer et de personnaliser des calendriers interactifs pour les applications web. On m'a remis des exemples de calendrier utilisant cette bibliothèque pour que je puisse me familiariser avec cette bibliothèque notamment les paramètres requis pour faire un calendrier et leur utilisation tels qu'un tableau events qui contient les informations des événements pour ensuite les afficher. La première tâche que j'ai dû faire était un fichier de configuration qui permet de changer l'ordre des boutons du calendrier depuis une interface de configuration externe au calendrier, pour faire cela j'ai choisi un exemple de calendrier fait avec fullCalendar pour faire cette tâche :



Après avoir choisi mon calendrier sur lequel je vais travailler j'ai choisi de configurer pour commencer les trois boutons en haut à gauche qui permettait de se déplacer dans le calendrier je me suis pris de la façon suivante:

- formulaire avec des cases à cocher avec enregistrement de l'ordre
- envoie du formulaire
- on vide la chaîne de caractères qui contient les boutons dans l'ordre classique
- récupération des données en méthode GET pour ensuite inclure dans la chaîne de caractère qui contient les boutons en fonction de l'ordre du click sur les cases

Voici comment sont stockés les boutons:

```
headerToolbar: [
    left: 'prevYear,prev,next,nextYear today',
    center: 'title',
    right: 'dayGridMonth,dayGridWeek,dayGridDay'
],
```

Cependant, la solution n'a pas été validée car elle serait compliquée pour une personne moins compétente en informatique . J'ai donc découvert une nouvelle bibliothèque JS qui s'appelle sortable qui est utilisable sur JavaScript, qui permet à l'utilisateur de réorganiser les éléments qui sont sélectionnés. Suite à de nombreux essais de compréhension du fonctionnement de la bibliothèque ,j'ai réussi à faire en sorte que mes trois boutons soient affichés sur la page de configuration, et que l'utilisateur puisse réorganiser comme il souhaite et que ça modifie la chaîne de caractères qui contient les boutons. Donc par la suite j'ai fais pareil pour les deux autres boutons ce qui a donner cela :

Mettez dans l'ordre les boutons que vous voulez sur le headerLeft:

prev
next
today

Mettez dans l'ordre les boutons que vous voulez sur le headerRight (List Day et List Week):

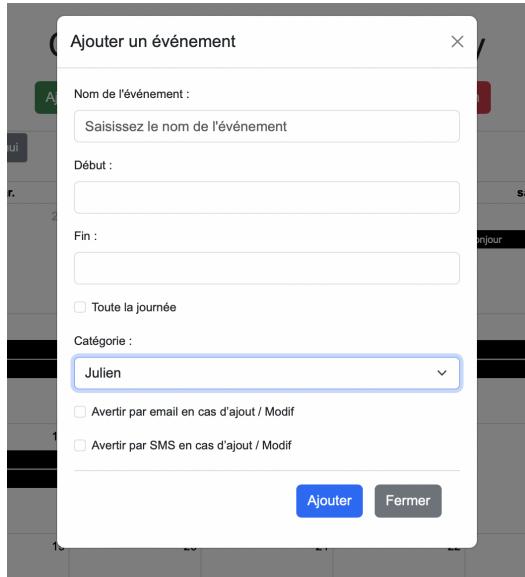
List Day
List Week

Avec cet outil on peut désormais réorganiser l'ordre des boutons et les déplacer directement depuis cette interface.

Dans un second temps, comme les données du calendrier étaient auparavant statiques, j'avais besoin de rendre le calendrier dynamique, j'ai donc lié le calendrier à une base de données. J'avais pour tâche d'ajouter les fonctionnalités ajouter,modifier,supprimer un événement.

La contrainte est que pour ce faire, je devais insérer un bouton d'ajout d' événement qui est relié au Modal bootstrap avec formulaire d'ajout. De plus, lorsque je clique sur un événement, cela affiche un modal de modification avec un formulaire de modification et un bouton de suppression.

Jusqu'à cette contrainte j'ai jamais entendu parler des Modal Bootstrap alors que j'ai déjà eu l'occasion d'utiliser bootstrap durant ma formation. Bootstrap est un outil qui facilite la création de sites web en fournissant des composants et une grille de mise en page préconçus. Cela permet aux développeurs de concevoir des interfaces utilisateur attrayantes et réactives plus rapidement et plus facilement. Je n'ai jamais entendu parler de Modal donc je me suis informé, c'est comme une petite fenêtre qui apparaît sur votre écran lorsque vous cliquez sur un bouton ou un lien. Il bloque le reste de la page, vous obligeant à interagir avec la fenêtre avant de pouvoir continuer. Ces fenêtres peuvent être utilisées pour afficher des messages importants, des formulaires à remplir ou des options à choisir sans que vous ayez à quitter la page principale. En résumé, c'est un moyen pratique de montrer des informations supplémentaires ou de demander quelque chose à l'utilisateur sans le perturber avec une nouvelle page ou fenêtre.



Donc j'ai fait les modals pour l'ajout et la modification des événements en apprenant la structure des modals sur la documentation.

Pour le bouton de l'ajout d'événement pour que lorsque qu'on clique dessus il y a le modal qui s'affiche j'ai rajouté des propriétés à mon bouton comme j'ai appris dans la documentation, pour l'ajout de l'événements après la soumission du formulaire, j'ai simplement fais une requête avec les valeurs du formulaires que je récupère en méthode post.

Pour que le formulaire de modification s'affiche il faut cliquer sur un événement, c'est à ce moment que j'ai découvert un gestionnaire d'événement inclus dans fullCalendar qui est eventClick, comme son nom l'indique quand on clique sur l'événement on peut dans mon cas afficher le modal de modification. Pour le confort de l'utilisateur j'ai eu une nouvelle contrainte qui était de pré remplir le formulaire de modification avec les informations de l'événement. Cette étape a été très compliquée pour moi car ce sont des tâches qui ne me sont pas familières et que je ne savais pas comment y procéder. J'y ai donc passé beaucoup de temps car au début je comptais pré remplir le formulaire en faisant une requête à chaque fois quand on clique sur l'événement pour récupérer ces informations mais cela semblait trop compliqué à ce moment à faire donc grâce à quelques recherches et chat gpt, je me suis rappelé qu'il y avait le tableau events qui stocke toutes les informations des événements

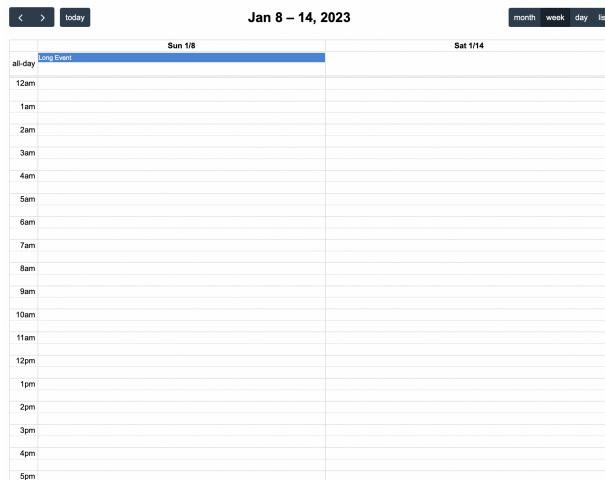
```
events: [
  {
    title: 'All Day Event',
    start: '2023-01-01'
  },
  {
    title: 'Long Event',
    start: '2023-01-07',
    end: '2023-01-10',
    display:'background',
    color: 'grey'
  },
  {
    title: 'Click for Google',
    url: 'http://google.com/',
    start: '2023-01-28'
  }
]
```

donc grâce à cela j'ai réussi à pré remplir le formulaire et avec les valeurs qu'il envoie lors de sa soumission je mets à jour la table planning_calendriers. Pour la suppression, j'ai lié un lien du calendrier avec des valeurs que je pouvais récupérer en méthode GET et donc le lien du calendrier contenait en plus l'id de l'événement avec l'action "supprimer" et donc dès que j'ai l'action "supprimer" et l'id de l'événement en méthode GET, je fais une requête de suppression. Le but étant de donner une expérience agréable à l'utilisateur j'ai dû rajouter des class bootstrap que j'ai trouver sur la documentation pour rendre mes boutons et formulaire plus beau, voici un avant vs après:

The image shows two side-by-side screenshots of an "Ajouter un évènement" (Add Event) form. The left screenshot, labeled "Avant" (Before), shows a standard HTML form with a light gray background and standard black text. The right screenshot, labeled "Après" (After), shows the same form styled with Bootstrap, featuring a green header bar, a white input field with placeholder text, and a green "Ajouter" button. Both screenshots include fields for event name, start date, end date, category selection, and notification checkboxes.

Dans un troisième temps, afin que je puisse voir un nouveau type de calendrier et je puisse travailler avec en apprenant à remettre les fonctionnalités que j'ai pu développer sur le calendrier, :

Il m'était demandé de travailler avec un calendrier avec une perspective différente (vue) affichant uniquement samedi et dimanche.



J'ai ajouté les fonctionnalités ajouter, modifier et supprimer un événement.

Ensuite comme demandé dans le cahier des charges j'ai effectué le rajout d'un moyen qui permet de déplacer les événements dans le calendrier avec mise à jour des informations de l'événement.



Pour faire cela j'ai dû m'intéresser à AJAX qui est une technique web qui permet aux pages d'envoyer et de recevoir des données du serveur sans recharger toute la page. Cela rend les sites web plus interactifs et réactifs. Quant à la compréhension du fonctionnement de AJAX notamment la syntaxe des requêtes ajax, j'ai éprouvé beaucoup de difficulté car j'ai jamais utilisé cette technique de développement lors de ma formation, mais en trouvant une requête modèle j'ai compris comment cela fonctionne.

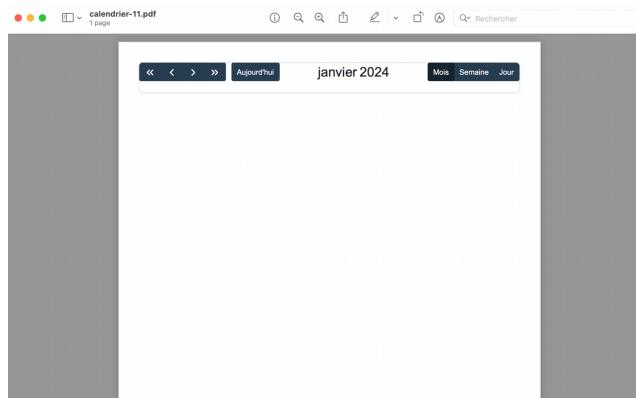
Dans mon calendrier à chaque fois qu'on déplace l'événement donc il fallait utiliser le gestionnaire d'événement eventDrop que j'ai trouvé dans la documentation du fullCalendar, il fallait envoyer une requête ajax qui récupère les informations de l'événement pour mettre à jour la date de début et de fin en fonction du déplacement, les informations sont envoyées via la requête AJAX sur un fichier que j'ai créé qui avec les valeurs que j'ai envoyées en méthode post de la méthode ajax, met à jour la table avec la nouvelle date de début et fin.

Pour le calendrier final, j'ai opté pour l'exemple de base que j'ai vu en premier car c'était plus conforme aux attentes. J'ai commencé par ajouter les fonctionnalités que j'avais développées.

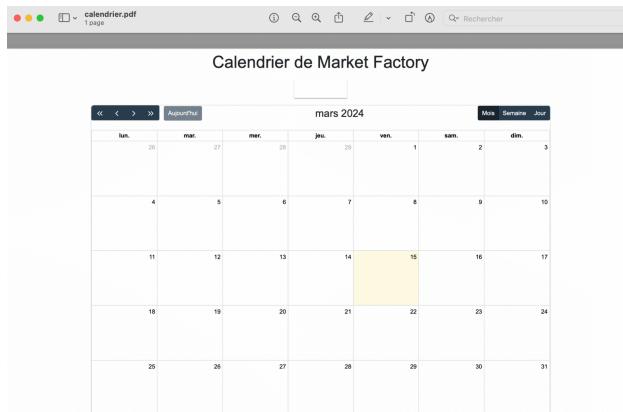
Ainsi comme l'exige le cahier des charges j'ai développé une interface d'identification pour pouvoir se connecter si on veut accéder au calendrier version admin. Lors de la configuration de la fonctionnalité d'impression des calendriers au format PDF, l'exemple d'un employé de Market Factory capable d'exporter un PDF, de l'imprimer et de l'afficher sur un mur a été donné. J'ai recherché un moyen de pouvoir imprimer le calendrier simplement et j'ai trouvé une bibliothèque qui s'appelle jsPDF qui est très populaire. Cette bibliothèque permet de générer des fichiers PDF à partir de contenu HTML, ce qui la rend adaptée à l'exportation de calendriers générés dynamiquement.

J'ai donc ajouté la bibliothèque à mon code et j'ai confectionné le code d'exportation en pdf du calendrier en m'inspirant de la documentation de jsPDF.

Sachant que durant l'utilisation j'ai eu plusieurs soucis due à la mauvaise manipulation des paramètres de la fonction de la bibliothèque afin de générer pdf donc il m'arrivait d'avoir des pdf avec que le haut du calendrier:



Après cela, j'ai réussi à mieux gérer les paramètres et la fonction de la bibliothèque grâce à cela j'ai réussi :



Pour la première fois, mon tuteur me propose de mettre mon travail en production ce qui signifie héberger un travail sur Internet où on pourrait y accéder via une URL, rendant mon travail disponible sur Internet.

Les étapes que j'ai suivis sont les suivants:

- 1. Installer un client FTP : J'ai installé un client ftp qui se nomme filezilla qui permet de transférer des fichiers entre mon ordinateur et un serveur distant.
- 2. Connexion au serveur : Pour me connecter au serveur de market factory où je vais héberger mon calendrier, je nécessitait l'adresse IP du serveur, l'utilisateur est le mot de passe et le port. Toutes ces informations utiles permettant la connexion au serveur m'ont été fournies par mon tuteur.
- 3. Transfert des fichiers : avant de transférer mon calendrier sur le serveur, j'ai eu pour tâche de supprimer tous les fichiers qui n'étaient pas utiles pour éviter d'occuper de la place dans le serveur puis j'ai déposé mon fichier directement sur le serveur.

Le calendrier était ensuite affiché directement visible via l'adresse URL lorsqu'on saisis dans le navigateur. Après avoir effectué cela, je pensais que tout allait fonctionner correctement mais il y avait un détail auquel je n'ai pas pensé que mon tuteur me l'a fait remarquer. Le problème était sur le serveur hébergeant mon calendrier qui n'avait sûrement pas le même encodage, la même version de langage de programmation semblable à ce qui est apparu sur mon ordinateur, un problème commun. Autrement dit problème d'encodage, c'est-à-dire tous les événements qu'ils avaient des caractères spéciaux tels que les caractères accentués étaient remplacés par des « ? ». Cependant, j'ai constaté que si je crée un événement à partir d'un calendrier mis en production, l'affichage serait correct.

Après avoir fait cela, j'ai dû mettre en place le fichier, que j'ai réalisé en premier qui est un fichier qui permet de configurer le calendrier mais aussi d'ajouter, modifier et supprimer des catégories comme demandé dans le cahier des charges.

J'ai dû rajouter une icône de paramétrage avec mes boutons :

[Ajouter un évènement](#)[EXPORT PDF](#)[Déconnexion](#)

Dès lors que l'on clique sur ce bouton on arrive sur la page de configuration qui permet de choisir les jours qu'on souhaite cacher et la vue du calendrier, ce sont les paramètres généraux qui peuvent être modifiés par l'utilisateur admin. Ainsi à chaque changement de paramètre la table planning_parametre est mise à jour. Cette page de configuration permet également de consulter les catégories en ayant la possibilité de modifier et supprimer :

Nom de la catégorie	Type Catégorie	Couleur de la catégorie	Courriel Contact	Avertir par email	Tel Contact	Avertir par SMS	Actions
test	Event			Non		Non	Modifier Supprimer
Julien	Event			Non		Non	Modifier Supprimer

On peut également ajouter des catégories. Pour les modifications, j'ai remplacé le formulaire d'ajout de catégorie par un formulaire de modification:

Formulaire d'ajout de catégorie

Nom de la catégorie :

Type Catégorie :

Couleur de la catégorie :

Courriel Contact :

Tel Contact :

Avertir par email en cas d'ajout / Modif

Avertir par SMS en cas d'ajout / Modif

[Ajouter](#)

Formulaire de modification de catégorie

Nom de la catégorie :

Type Catégorie :

Couleur de la catégorie :

Courriel Contact :

Tel Contact :

Avertir par email en cas d'ajout / Modif

Avertir par SMS en cas d'ajout / Modif

[Modifier](#)

J'ai pré rempli le formulaire de modification en faisant une requête qui récupère les informations de la catégorie via l'id catégorie que j'envoie en méthode GET lors de l'appuis du bouton modifier, la méthodologie que j'utilise est celle que je n'ai pas réussi à faire afin de pré remplir le formulaire de modification d'événement , entre-temps j'ai beaucoup appris.

La version finale de ce calendrier représente le résultat complet de mon travail, sur lequel j'ai consacré beaucoup de temps. Cependant, il est important de noter que ce calendrier est évolutif et pourrait être amélioré à l'avenir. Ce projet a été une expérience formatrice pour moi, car je suis passé d'un calendrier statique à un calendrier dynamique.

J'ai également appris à sécuriser mes données en utilisant des méthodes POST plutôt que GET, et j'ai acquis une meilleure compréhension des sessions en programmation. Travailler sur ce calendrier m'a permis de découvrir de

nombreuses bibliothèques JavaScript et d'explorer des techniques de développement que je n'aurais peut-être jamais utilisées autrement.

Cela m'a également enseigné à coder de manière plus pragmatique, en répondant aux besoins spécifiques du projet plutôt qu'aux exigences académiques. Avant ce projet, je sous-estimais la complexité de la création d'un calendrier, ainsi cette expérience m'a permis d'approfondir mes connaissances sur les calendriers et le développement en général car grâce à ce projet, cela m'a permis de m'améliorer dans plusieurs langages de programmation tels que JS,SQL,PHP,HTML,etc.

Enfin, j'ai réalisé l'importance de la maintenance continue des projets ainsi j'ai compris l'importance de l'optimisation du code et ceci s'avère très important. Par exemple, au départ j'avais deux fichiers contenant les calendriers des visiteurs et un fichier contenant le calendrier administrateur qui était personnel. Je me suis connecté et j'ai remarqué que cela nécessitait deux fois plus de maintenance. Il était donc clair que le code pouvait être optimisé même après son achèvement initial, ce qui a été très utile pour faire prendre conscience du rôle du développeur dans la satisfaction des exigences

Pour voir le rendu final (voir [annexe 1.2](#))

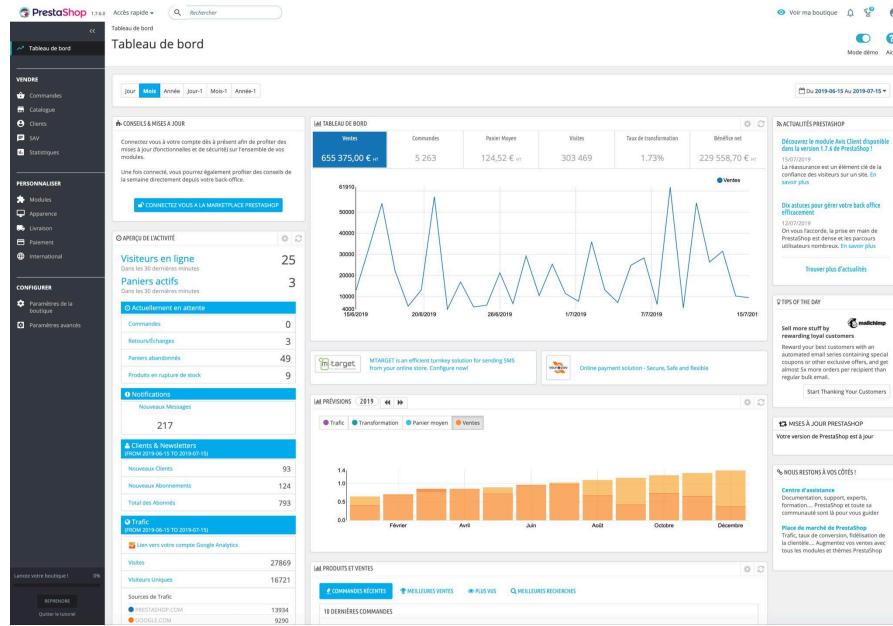
Un Incident à Market Factory:

Il y a eu un incident qui a beaucoup impacté la société Market Factory ce qui a fait perdre sept ans de travail au gérant de Market Factory qui est Stephane Pachis. Cet incident a surtout impacté la deuxième branche de Market Factory qui, comme je l'ai expliqué durant l'introduction consistait à créer des boutiques en ligne pour ses clients et s'occuper de la production des produits mise en ligne sur la boutique en ligne. L'incident qui a eu lieu est le crash des deux disques dur de Market Factory qui fait perdre des factures, boutique en ligne de clients, et informations liées aux clients.

Market Factory créait des boutiques en ligne auparavant sur Magento qui est une plateforme de commerce électronique qui permet aux entreprises de créer et de gérer des boutiques en ligne. C'est similaire à un magasin virtuel où les entreprises peuvent vendre leurs produits et gérer les commandes, les paiements et les expéditions. Sachant qu'il a perdu toutes les boutiques en ligne des clients pour recréer ses boutiques, il a décidé de changer de plate-forme et d'aller sur prestashop qui est également une plateforme de commerce électronique, similaire à Magento.

C'est un logiciel qui aide les entreprises à créer et à gérer des boutiques en ligne où elles peuvent vendre leurs produits, gérer les commandes et les paiements. C'est similaire à un magasin virtuel où les entreprises peuvent faire des affaires sur Internet. Il a donc changé de plateforme car il n'avait pas la dernière version. Ainsi effectuer la mise à jour de Magento et d'apprendre les nouvelles fonctionnalités étaient compliquées pour lui. La solution la plus simple a été de passer sur PrestaShop, qui était une plate-forme qui était beaucoup plus simple à apprendre et à comprendre.

J'ai par la suite assisté à un appel de 30 minutes avec mon maître de stage qui était en Visio avec un développeur PrestaShop, qui a configuré le back office PrestaShop de Market Factory, selon les besoins de mon tuteur. Un back office PrestaShop est l'interface d'administration où le tableau de bord est réservé aux propriétaires de boutiques en ligne. C'est là où les administrateurs gèrent tous les aspects de leur boutique, y compris la gestion des produits, des commandes, des clients, des paiements, des expéditions, etc.



J'ai ensuite aidé Market Factory à reconstruire cette branche d'activité.

Pour commencer, j'ai appris à faire des fiches produits sur PrestaShop donc une fiche produit PrestaShop est une page spécifique dans une boutique en ligne créée avec la plateforme PrestaShop, dédiée à la présentation détaillée d'un produit disponible à la vente. Cette fiche contient des informations essentielles sur le produit pour aider les clients à prendre une décision d'achat.

Elle comprend généralement :

1. Nom du produit.
2. Description détaillée du produit, incluant ses caractéristiques et ses avantages, etc.
3. Images haute résolution du produit, montrant différents angles et perspectives.
4. Prix du produit, y compris les éventuelles réductions ou promotions.
5. Options de personnalisation ou de choix de variantes (taille, couleur, etc.).
6. Disponibilité en stock.

Ensuite, j'ai appris à créer des boutiques pour des clients donc les étapes que j'ai suivie pour créer des boutiques en ligne sur PrestaShop sur les suivantes:

- création de la catégorie pour la boutique
- Création de la boutique ensuite lié à la catégorie auparavant
- Création des produits qui sont ensuite ajouté à la boutique

- Configuration de la mise en page de la boutique

Mon tuteur m'a enseigné la gestion du Back Office, grâce à cela j'ai eu l'opportunité de m'occuper de la maintenance des boutiques en ligne donc dès qu'il y avait un problème sur une des boutiques, je m'occupais de résoudre ceci, tel que les problèmes d'affichage.

Cette étape m'a permis d'apprendre beaucoup plus sur cette branche de Market Factory mais aussi sur la gestion et le fonctionnement de la boutique e-commerce.

Les Outils de gestion de commande Clients

Dans le but de pouvoir satisfaire une cliente qui souhaite suivre ses commandes passées chez Market Factory et aussi faciliter le travail des collaborateurs de Market Factory notamment dans la gestion et suivi des commandes. Mon tuteur nous a confié un projet à David et moi qui consiste à développer un outil qui aura une vue client et admin nécessitant une interface de connexion, la différence c'est que l'admin aura accès à toutes les boutiques et pourra changer les états de commandes, créer un utilisateur client, alors que le client peut uniquement accéder aux informations de sa boutique. L'outil doit permettre de voir les commandes, les ventes, les produits disponibles dans les boutiques.

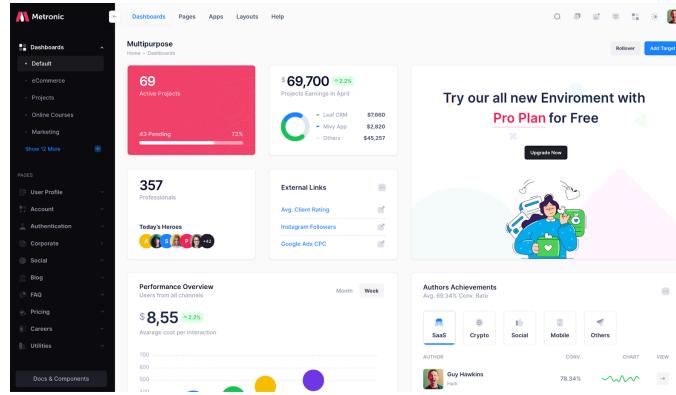
L'outil permet de créer des nouvelles boutiques sous forme de flyer avec les produits de Market Factory ainsi que de rajouter des logos. Si le mode d'expédition est colissimo, nous devons imprimer le bordereau qui convient à ce dernier.

Après avoir identifié les demandes, j'ai rendu visite aux deux employés afin de répondre à leurs besoins et permettre de créer un cahier des charges (voir annexe 1.3) adapté aux besoins de Mon tuteur et de ses collaborateurs.

Contrairement au calendrier, mon tuteur nous a envoyé directement la base données qui contenait toutes les tables nécessaires pour réaliser le projet. C'est la première fois que j'utilisais une aussi grande base de données, mais on était obligés d'utiliser cette base de données car il y avait toutes les informations concernant les boutiques, ventes, commandes, livraisons, produits, et pleins d'autres.

Pour commencer David et moi on a eu des tâches séparées, celle de David est de faire un tableau des commandes de la boutique choisi via un menu déroulant contenant toutes les boutiques des clients de Market Factory.

La mienne étant de prendre en main le template que mon tuteur a acheté pour ce projet, expliquant que celui-ci servirait de base pour la construction du site web. Pour par la suite mettre le travail de David dans mon interface. Ce template consiste en un ensemble de fichiers pré-conçus qui définissent l'apparence générale du site, y compris les couleurs, les polices et la mise en page. Utiliser ce template nous offrait une base solide sur laquelle nous pouvions construire et personnaliser le code en fonction de nos besoins spécifiques. Cependant, travailler directement sur ce fichier unique était complexe, car toutes les parties du code étaient regroupées sans distinction.



Pour optimiser le code et rendre la gestion plus facile, mon tuteur m'a demandé de diviser le fichier en plusieurs parties et d'utiliser des inclusions avec des directives « include » dans un fichier principal. Cette tâche s'est avérée dure, car le fichier initial contenait environ 12 000 lignes de code. En divisant le fichier de manière méthodique, j'ai pu réduire sa taille à 260 lignes, rendant ainsi sa manipulation beaucoup plus simple. J'ai aussi lors cette étape, remarqué que à la moindre erreur cela peut causer des problèmes d'affichage. Cette expérience m'a véritablement appris à repérer facilement les différentes parties du template et à organiser le code de manière à faciliter sa gestion et sa maintenance à l'avenir.

Pour commencer, j'ai dû faire du tri en retirant les éléments inutiles afin de réserver le template uniquement pour notre outil et le détacher des autres pages du template. Cela signifiait : vider le menu de la barre latérale pour y intégrer nos rubriques, enlever les éléments en haut et les widgets de l'affichage à l'exception de l'icône utilisateur.

Après avoir fait cela, j'ai dû initialiser des variables en SESSION afin de savoir si on est connecté ou non pour que si on est pas connecté on affiche un formulaire de connexion et dans le cas contraire on affiche l'interface de base.

Ensuite j'ai dû rajouter le tableau de David en ajouter des class que j'ai récupéré dans les tableaux qui sont donnés dans le template pour que le tableau soit plus esthétique :

Projects Stats					
Updated 37 minutes ago					
ITEM	BUDGET	PROGRESS	STATUS	CHART	VIEW
Mivy App Jane Cooper	\$32,400	↑ 9.2%	In Process		→
Avonica Esther Howard	\$256,910	↓ 0.4%	On Hold		→
Charto CRM Jenny Wilson	\$8,220	↑ 9.2%	In Process		→
Tower Hill Dusty Fisher	\$74,000	↑ 9.2%	Completed		→
9 Degree Savannah Nguyen	\$183,300	↓ 0.4%	In Process		→

Avant

Choisissez une option : Market Factory Soumettre				
id_order	addr	Nom/prénom	Email	Tél
1				
2				
3				
4				

Après

Tableau des commandes Albert de Mun		
DATE DE COMMANDE	NOM/PRENOM	PRODUIT
07/03/2024 16:32:53		Veste bleu marine Logo manche (Tailles : M) Non traité
15/02/2024 11:38:29		T-shirt blanc Logo cœur (Tailles : M) Remboursé
09/02/2024 16:56:23		T-shirt Blanc Logo face (Tailles : L) Non traité

Par la suite, David m'a informé qu'il a mis une fonctionnalité qui change l'état des commandes lors du changement dans le menu déroulant via une requête ajax qui met à jour la base de données.

Les commandes sont récupérées dans la table ps_orders.

Par la suite, j'ai donc réussi à planter le tableau dans mon interface.

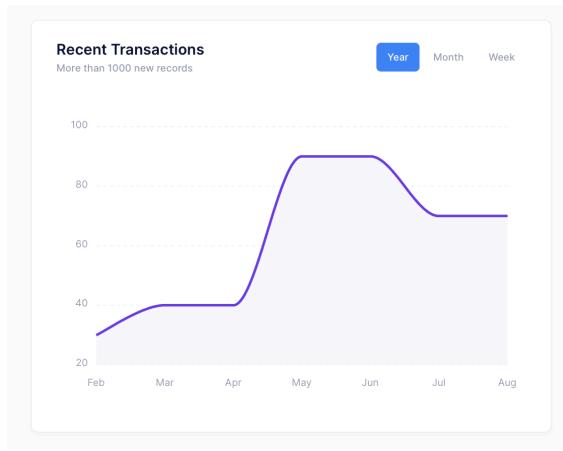
Ensuite, j'ai dû faire pareil mais par la liste des clients des boutiques qui sont dans la table ps_customer sur une nouvelle page. Lors de cette tâche j'ai rencontré une difficulté qui faisait des doublons de clients car ils avaient à chaque fois plusieurs contacts pour une et même personne. Pour que cela se reproduise, j'ai effectué une requête pour afficher un contact par client en les distinguant par l'email.

J'ai mis en place la vue client et admin, bien évidemment sur le même fichier. Pour faire cela j'ai émis des conditions en fonction du rôle de l'utilisateur qui est stocké dans la base de données ps_client que j'ai créé pour l'authentification de l'utilisateur

id_client admin id_shop_client nom_client prenom_client email_client password_client tel_client valid_client

En fonction de cela afficher ou non certaines fonctionnalités, par exemple un client ne verra pas la liste déroulante pour choisir la boutique et changer l'état.

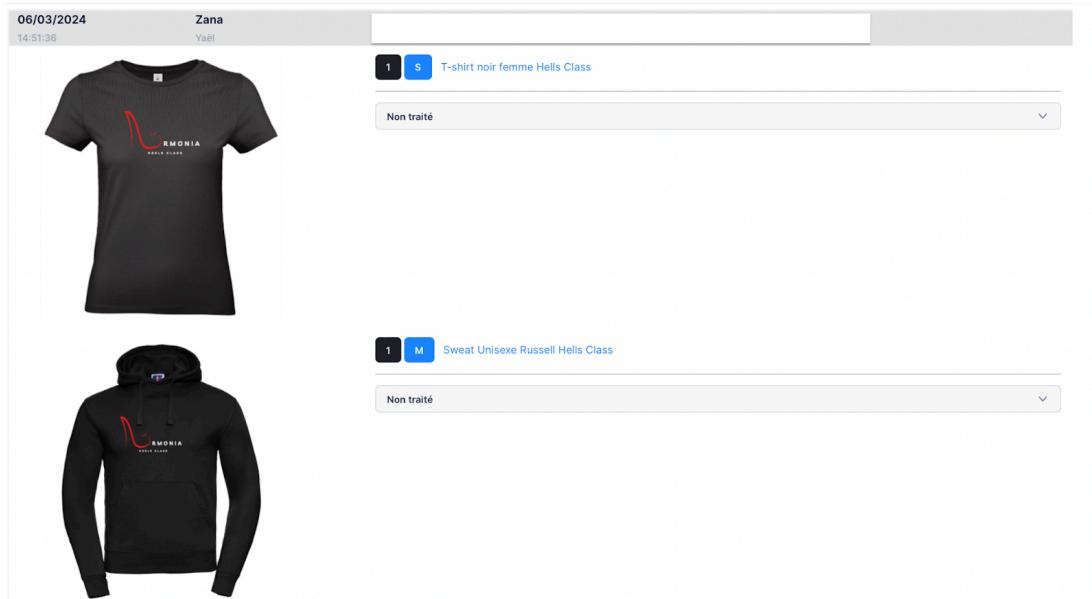
Ensuite j'ai une tâche pour afficher les ventes sur une nouvelle page, donc pour cela j'ai utilisé un graphique que j'ai trouvé dans le template :



J'ai appris que ce graphique se remplissait grâce à un tableau dans le java script des widgets donc ce que j'ai fait j'ai uniquement modifié le visuelles du graphiques et remplace le tableau par les données de ventes que j'ai récup dans la base de données dans la tables ps_orders pour ensuite les mettre dans un tableau dans l'ordre des mois. Mais le plus dur dans cette tâche est de trouver l'emplacement du tableau pour remplacer le contenu par le contenu de mon tableau des ventes de Market Factory car voici une petite partie du fichier en question :

Mais je suis parvenu à trouver le tableau grâce au raccourci **ctrl+f** pour chercher avec l'id du tableau.

Entre-temps mon tuteur a dû changer l'affichage des commandes pour remplacer le tableau par l'affichage

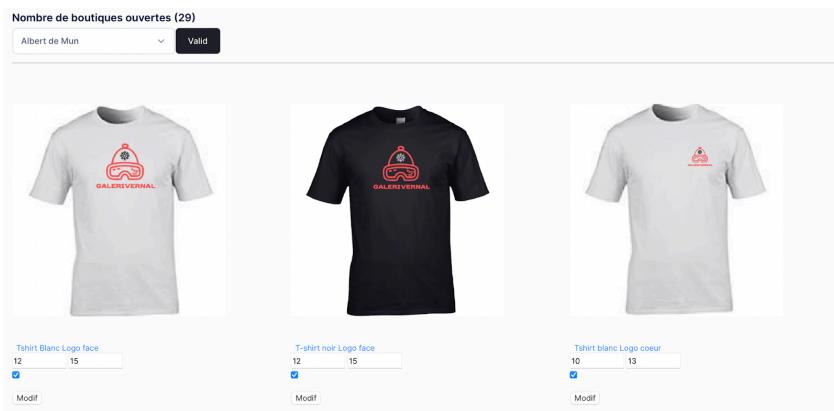


suivant.

Il a également ajouté les images et il m'a expliqué comment sont stockés les images . En effet, le chemin vers l'image est stocké dans une table ps_image et grâce à ce chemin on peut récupérer les images sur le répertoire. Avec cette technique j'ai ajouté une page pour afficher la liste des produits des boutiques qui m'a fait un schéma de ce qu'il voulait et comment je devais utiliser les les <div> et la largeur des col, même si j'ai déjà eu un cours sur les col en Bootstrap, il m'a enseigné les colonnes Bootstrap et aussi comment je devais mettre mes <div> pour éviter d'avoir des problèmes d'affichages.

Ensuite j'ai récupéré les images comme il m'a montré et il m'a demandé de mettre un formulaire pré rempli avec le prix de production et de ventes que l'utilisateur peut modifier. L'utilisateur peut également choisir si la fiche produit s'affiche dans la boutique e-commerce. Pour voir si j'ai bien compris j'ai dû faire un cahier des charges fonctionnels qui a été validé par mon tuteur (annexe 1.4).

Et voici le rendu:



J'ai mis mon travail en production en suivant les même étapes que le calendrier sauf que c'est sur un autre serveur de Market Factory, contrairement au Calendrier lors de ma première mise en production je n'ai pas eu d'erreur car j'ai appris de mes erreurs lors de la première mise en production.

Ensuite j'ai ajouté quelques fonctionnalités pour satisfaire les collaborateurs de Market Factory tels que les couleurs aux états qui ont été définies par les eux-même pour chaque états,

En commande Fournisseur

un bouton pour afficher les messages des clients par exemple lors du erreur de commande que je récupère dans la base de données dans une table qui s'appelle ps_message qui contient les messages des clients et je prends le messages uniquement quand c'est pas le message par défaut et quand on clique dessus y'a le message affiché par un Modal,

The screenshot shows a web-based shopping cart interface. At the top left, it says "Olivet Basket 07/03/2024 09:16:13". In the center, there's a product image of a white tote bag with a green logo. To its right, a list item shows "1 Tot bag logo olivet" with "Personnalisation: ['Marquage noir']" and a dropdown menu set to "Non traité". A red "ATTENTION" button is visible at the top right. On the right side, a modal window titled "ATTENTION MESSAGE DU CLIENT !" displays the message "LIVRAISON CLUB CHEZ MME TALBORDET - JE SOUHAITE LE BAS DE SURVET NOIR ET ECRITURE BLANCHE". At the bottom right of the modal is a "Fermer" button.

J'ai trouvé une solution en utilisant une requête qui joint en fonction d'une condition de jointure, mais si la commande comporte plusieurs produits et qu'elle contient un produit

La commande avait le perso1, les autres produits avaient le perso2, et tous les produits de la commande étaient personnalisés avec le perso1, j'ai donc dû utiliser une jointure pour créer la demande, mais c'est plus précisément, elle était basée sur deux conditions contraignantes.

```
Première requête : SELECT ps_order_detail.*, ps_wk_product_customer_options.option_value  
FROM ps_order_detail  
LEFT JOIN ps_wk_product_customer_options  
ON ps_order_detail.id_order = ps_wk_product_customer_options.id_order  
WHERE ps_order_detail.id_order_detail = :id_order_detail
```

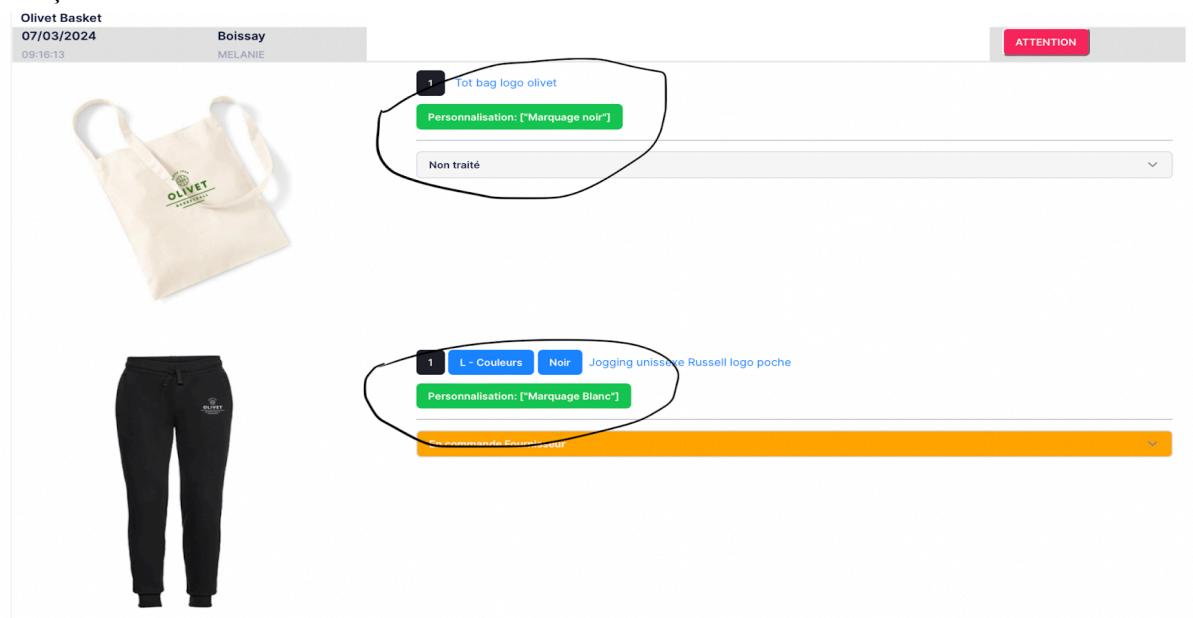
Pour ce résultatat:

This screenshot shows a shopping cart with two items. The top item is a white tote bag with a green logo, with a callout highlighting the personalization message "Personnalisation: ['Marquage noir']". The bottom item is a pair of black joggers with a green logo, also with a callout highlighting the personalization message "Personnalisation: ['Marquage noir']". Both items have a status bar below them indicating they are not yet treated ("Non traité"). At the very bottom of the page, a yellow bar contains the text "En commande Fournisseur".

Par la suite j'ai eu l'idée de faire une condition de jointure avec deux attributs:

```
SELECT ps_order_detail.*, ps_wk_product_customer_options.option_value  
FROM ps_order_detail  
LEFT JOIN ps_wk_product_customer_options  
ON ps_order_detail.id_order = ps_wk_product_customer_options.id_order AND  
ps_order_detail.product_attribute_id = ps_wk_product_customer_options.id_product_attribute  
WHERE ps_order_detail.id_order_detail = :id_order_detail
```

Pour ça :



Après avoir répondu aux demandes des collaborateurs de Market Factory, j'ai ajouté une fonctionnalité qui permet d'ajouter un utilisateur client depuis l'interface admin, pour faire cela j'ai pris le code du formulaire d'ajout de catégorie du calendrier et je l'ai mis dans ma page car il faut comme pour le calendrier un formulaire d'ajout et un tableau avec tous les comptes qu'on peut modifier et supprimer :

Gestion des comptes admin

Accueil - Commandes

--Choisissez une boutique--

Nom	Prénom	Email	Téléphone	Mot de passe
-----	--------	-------	-----------	--------------

Avertir

Créer Admin Client

Nom	Prenom	Email	Numéro	Boutique	Actions

Pour finir, j'avais pour tâche de créer une page qui permet à l'utilisateur de créer sa propre boutique en créant son flyer avec les produits de market Factory et aussi avoir la possibilité de mettre des logos.

Au début lorsque l'on arrive sur la page de création de flyer on a un formulaire :

Créer une boutique

Nom du flyer

Nom de la boutique

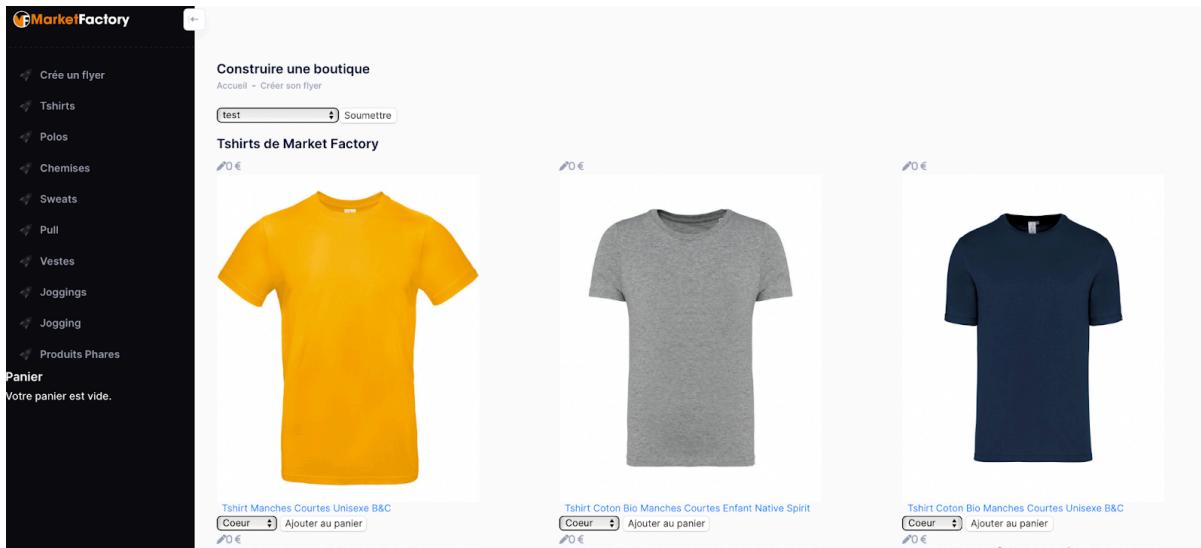
Description de la boutique

Logo 1: Choisir le fichier aucun fichier sélectionné
 Logo 2: Choisir le fichier aucun fichier sélectionné
 Logo 3: Choisir le fichier aucun fichier sélectionné

Créer

Pour les images que je récupère du formulaire, j'ai décidé de récupérer les images et de les enregistrer dans un répertoire pour faciliter l'insertion du chemin d'accès dans le tableau.

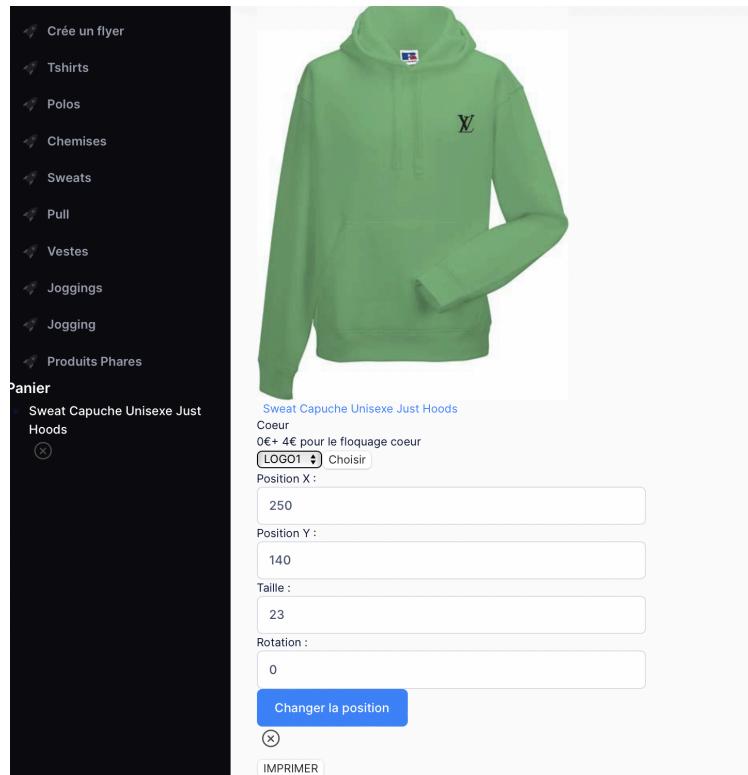
Si vous êtes un nouveau client, rendez-vous directement sur cette page pour créer votre boutique. Une fois cette opération terminée, vous aurez accès au grand catalogue Market Factory.



Avec la possibilité pour chaque produit de choisir la position de son marquage et pouvoir l'ajouter au panier, à chaque fois qu'on ajoute le produit au panier, il est ajouté à la base de données dans la tables ps_cart_flyer que j'ai créée pour le projet :

id_cart	id_product	quantity	id_client	created_at	nom_produit	prix_produit	position_logo	id_logo_flyer	id_flyer	x	y	taille	rotation
-------------------------	----------------------------	--------------------------	---------------------------	----------------------------	-----------------------------	------------------------------	-------------------------------	-------------------------------	--------------------------	-------------------	-------------------	------------------------	--------------------------

Ensute quand on est dans le panier on peut changer la position du logo, choisir le logo parmi ceux qu'on a ajouté dans le formulaire.

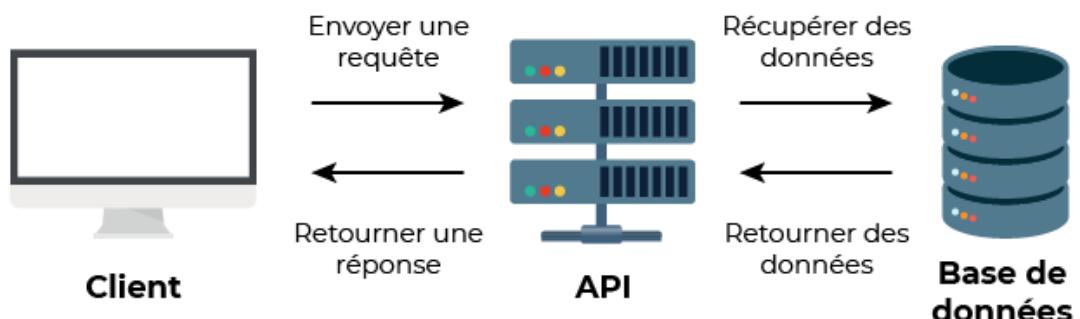


La superposition de l'image avec le logo, j'ai utilisé GD qui est une bibliothèque logicielle qui permet la création, la manipulation et l'affichage d'images sur les sites web. Grâce à un tutoriel j'ai appris à créer une fonction qui utilise GD pour superposer l'image du produit avec le logo selon la position.

Le but de cette page est tout de même faire un flyer en pdf qui devait ressembler à ça (voir annexe 1.5).

David a développé un moyen de génération de pdf grâce à FPDF est une bibliothèque PHP qui permet de générer des fichiers PDF à partir de données dynamiques. Elle offre des fonctionnalités pour créer des documents PDF personnalisés en ajoutant du texte, des images, des tableaux, des formes et d'autres éléments graphiques. J'ai uniquement rajouté les images générées dans le panier dans le pdf.

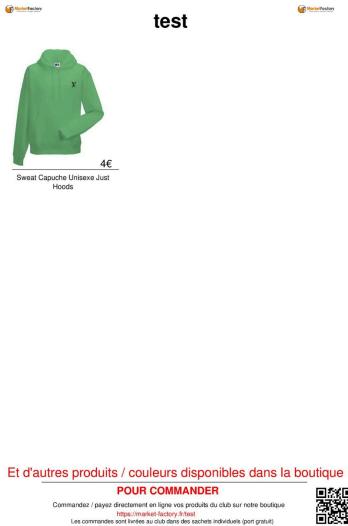
Mon tuteur m'a ensuite donné une contrainte qui consistait à afficher un qr code dynamique lié à l'adresse "market-factory.fr/nom_boutique_client" pour cela j'ai utiliser une api :



Pour cela, j'ai initialement utilisé qrcode Monkey comme API

Cependant, il était difficile de modifier le code inclus dans le document pour l'adapter à mes besoins, car le document ne comprenait pas le lien que je souhaitais inclure dans le code QR à la fin.

J'ai fini par faire le pdf :



La version finale de cet outil représente le résultat complet de mon travail, dans lequel j'ai mis beaucoup d'efforts. Je suis vraiment fier de cet outil car c'était le fruit de ma première demande d'un client .C'est vraiment impressionnant de dire que nous sommes partis presque de zéro et que nous avons obtenu des résultats comme ces derniers.

Nous avons fait de grands progrès en matière de HTML, notamment en ce qui concerne la structure des modèles et un certain nombre de problèmes d'affichage

De plus, cette première expérience avec un client m'a permis de mieux comprendre les attentes et les besoins de l'utilisateur final, ce qui a été une étape importante dans mon évolution professionnelle

J'ai également eu l'opportunité de découvrir les API et d'apprendre à les utiliser, ce qui a élargi mes horizons de développement. Je suis reconnaissant pour les leçons tirées de ce projet

Pour voir la dernière version de l'outil voir ([annexe 1.6](#)).

Conclusion:

Durant ce stage de deux mois que j'ai pu faire chez Market Factory, j'ai énormément appris à travers des projets passionnants. Durant ce stage, j'ai réalisé deux projets qui vont servir à Market Factory, un calendrier et un outil de gestion de commandes. A travers cette expérience, j'ai pu voir une progression notable dans tous les langages de programmation que j'ai utilisés tels que PHP,HTML,SQL,JAVASCRIPT, j'ai également découvert l'étendue du métier de développeur full stack et ainsi que j'avais un avenir dans ce métier. Ce stage a pleinement accompli son objectif qui est de m'initier au monde professionnel. La chose qui me rend le plus heureux est que les projets que j'ai réalisés au sein de l'entreprise vont être utilisés par l'entreprise et peut être mis à jour par la suite.

Annexes

Annexe 1.1:

Cahier des charges : Calendrier Interactif

1. Introduction

Ce cahier des charges vise à définir les spécifications nécessaires à la réalisation d'un calendrier interactif pour l'entreprise Market Factory. Ce calendrier devra permettre aux utilisateurs d'ajouter, modifier et supprimer des événements associés à des catégories, offrant ainsi une gestion flexible et intuitive des rendez-vous et des tâches.

2. Objectifs

- Créer un calendrier interactif permettant aux utilisateurs d'ajouter, modifier et supprimer des événements.
- Intégrer des catégories pour classer les événements et faciliter leur organisation.
- Proposer deux vues distinctes : une vue visiteur avec des fonctionnalités limitées et une vue admin avec des fonctionnalités étendues.
- Assurer la configurabilité du calendrier, même pour les utilisateurs non techniques.
- Utiliser FullCalendar pour la mise en œuvre du calendrier dynamique.
- Déplacer les événements sur le calendrier avec mise à jour dans la base de données
- Exporter le calendrier en pdf

3. Spécifications fonctionnelles

Le calendrier interactif doit répondre aux exigences suivantes :

3.1 Fonctionnalités de base :

- Affichage des événements sous forme de calendrier dynamique.
- Possibilité d'ajouter, modifier et supprimer des événements.
- Classification des événements par catégories pour une meilleure organisation.
- Vue mensuelle permettant de visualiser les événements sur l'ensemble du mois.
- Déplacer les événements sur le calendrier avec mise à jour dans la base de données

3.2 Fonctionnalités avancées :

- Vue visiteur : accès restreint aux fonctionnalités de base (consultation des événements uniquement).
- Vue admin : accès complet aux fonctionnalités de gestion des événements.
- Interface conviviale permettant une configuration facile du calendrier, y compris l'ajout de catégories et la personnalisation des paramètres.
- Possibilité d'exporter le calendrier en pdf pour les collaborateurs de l'entreprise

4. Spécifications techniques

- Utilisation de FullCalendar pour la conception du calendrier dynamique.
- Intégration avec une base de données pour le stockage des événements et des catégories.
- Développement d'une interface intuitive et réactive pour une expérience utilisateur optimale.

5. Contraintes de développement

- Respect des normes de sécurité pour protéger les données des utilisateurs.
- Conception responsive pour garantir une expérience utilisateur cohérente sur tous les appareils (ordinateurs de bureau, tablettes, smartphones).

6. Livrables attendus

- Une version fonctionnelle du calendrier interactif répondant aux spécifications définies.

Annexe 1.2:

Calendrier de Market Factory

Connexion

<< < > >>		Aujourd'hui	mars 2024					Mois	Semaine	Jour
lun.	mar.	mer.	jeu.	ven.	sam.	dim.				
					1					3
26	27	28	29		2					
4	5	6	7	8	9	10				
11	12	13	14	15	16	17	• 12 h test			
18	19	20	21	22	23	24				
25	26	27	28	29	30	31				

Calendrier de Market Factory

Ajouter un événement EXPORT PDF ☰ Déconnexion

<< < > >>		Aujourd'hui	mars 2024					Mois	Semaine	Jour
lun.	mar.	mer.	jeu.	ven.	sam.	dim.				
					1					3
26	27	28	29		2					
4	5	6	7	8	9	10				
11	12	13	14	15	16	17	• 12 h test			
18	19	20	21	22	23	24				
25	26	27	28	29	30	31				

Ajouter un événement

Nom de l'événement :

Saisissez le nom de l'événement

Début :

Fin :

Toute la journée

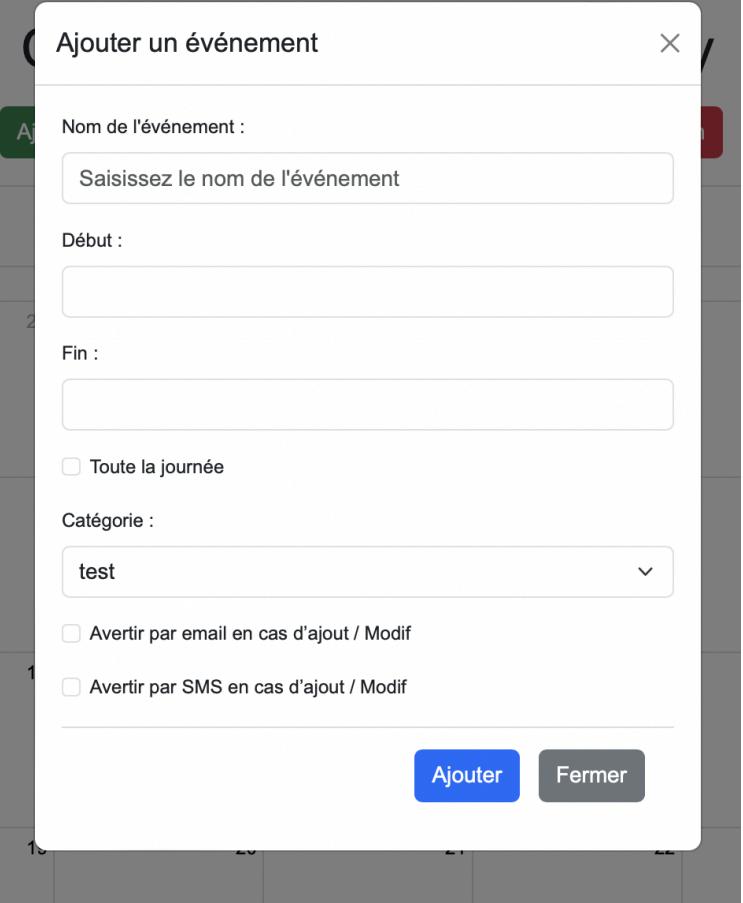
Catégorie :

test

Avertir par email en cas d'ajout / Modif

Avertir par SMS en cas d'ajout / Modif

Ajouter **Fermer**



MarketFactory

Paramètres généraux

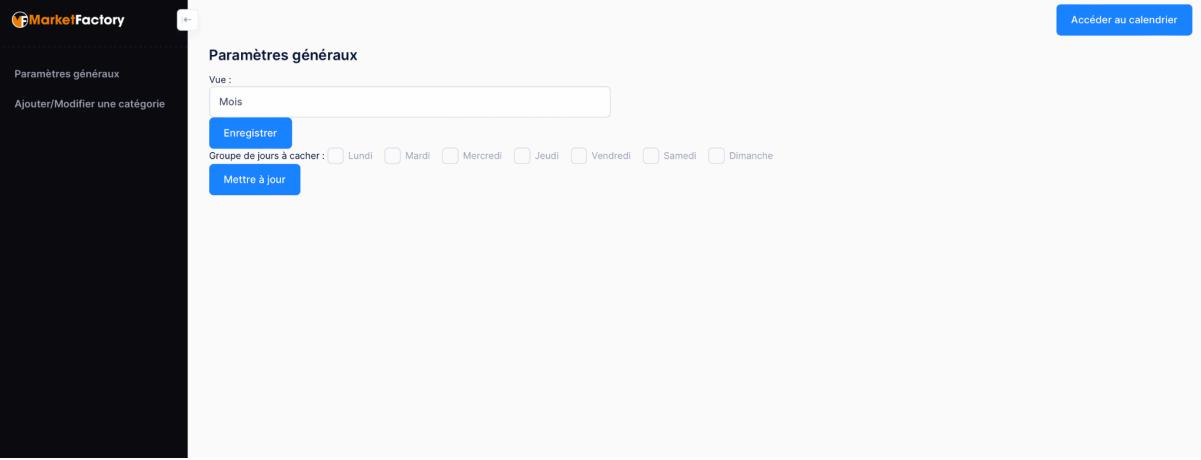
Vue : Mois

Enregistrer

Groupe de jours à cacher : Lundi Mardi Mercredi Jeudi Vendredi Samedi Dimanche

Mettre à jour

Accéder au calendrier



MarketFactory

Paramètres généraux

Ajouter/Modifier une catégorie

Nom de la catégorie : Saisissez le nom de la catégorie Type Catégorie : Event Couleur de la catégorie :

Courriel Contact : Courriel Contact Tel Contact :

Avertir par email en cas d'ajout / Modif Avertir par SMS en cas d'ajout / Modif

[Ajouter](#)

Nom de la catégorie	Type Catégorie	Couleur de la catégorie	Courriel Contact	Avertir par email	Tel Contact	Avertir par SMS	Actions
test	Event	■■■■■	test@gmail.com	Non	Non	Non	Modifier Supprimer
Julien	Event	■■■■■		Non	Non	Non	Modifier Supprimer

Accéder au calendrier

test

Voulez-vous supprimer ou modifier l'événement ?

Nom de l'événement : test

Début : 2024-03-14 12:00

Fin : 2024-03-14 13:00

Toute la journée

Catégorie : test

Avertir par email en cas d'ajout / Modif
 Avertir par SMS en cas d'ajout / Modif

[Modifier](#) [Fermer](#) [Supprimer](#)

Annexe 1.3:

Cahier des charges - Outil de suivi des commandes pour Market Factory

Objectif :

Le projet vise à développer un outil de suivi des commandes pour satisfaire les besoins d'une clientèle souhaitant suivre ses commandes passées chez Market Factory. L'outil doit également faciliter le travail des collaborateurs de Market Factory en simplifiant la gestion et le suivi des commandes.

Fonctionnalités requises :

1. Vue client :

- Accès aux informations de sa boutique.
- Consultation des commandes passées.
- Visualisation des ventes réalisées.
- Consultation des produits disponibles dans sa boutique.

2. Vue administrateur :

- Accès à toutes les boutiques.
- Possibilité de changer les états des commandes.
- Création et gestion des utilisateurs clients.
- Consultation des commandes, des ventes et des produits pour toutes les boutiques.

3. Création de nouvelles boutiques :

- Création de nouvelles boutiques sous forme de flyer.
- Intégration des produits de Market Factory dans les flyers.
- Possibilité d'ajouter des logos personnalisés.

4. Impression de bons de livraison Colissimo :

- Génération et impression de bons de livraison Colissimo pour les commandes expédiées via ce mode de livraison.

5. Personnalisation des produits :

- Possibilité pour les clients de personnaliser certains produits lors de leur commande, avec des options telles que la taille, la couleur, etc.

6. Messagerie client :

- Intégration d'un système de messagerie pour permettre aux clients de contacter le service clientèle pour poser des questions, signaler des problèmes, etc.

7. Couleur des états de commande :

- Attribution de couleurs spécifiques à chaque état de commande (en attente, en cours de traitement, expédiée, etc.) pour une meilleure visualisation et compréhension.
- Affichage de ces couleurs dans l'interface utilisateur pour une identification rapide de l'état de chaque commande.

Technologies utilisées :

1. HTML, CSS et JavaScript pour le développement de l'interface utilisateur.
2. PHP pour la gestion des requêtes serveur et l'interaction avec la base de données.
3. MySQL pour la gestion de la base de données.

- 4. Utilisation de frameworks ou bibliothèques JavaScript pour les fonctionnalités dynamiques et l'amélioration de l'expérience utilisateur.**
- 5. Utilisation de bibliothèques d'impression pour générer et imprimer les bons de livraison Colissimo.**

Contraintes techniques :

- 1. Compatibilité : L'outil doit être compatible avec les principaux navigateurs web et les appareils mobiles.**
- 2. Modularité : Le code doit être bien structuré et modulaire pour faciliter la maintenance et l'ajout de nouvelles fonctionnalités à l'avenir.**

Livrables attendus :

- 1. Interface utilisateur fonctionnelle et intuitive, avec des vues distinctes pour les clients et les administrateurs.**

annexe 1.4:

Cahier des charges fonctionnel pour liste-produits

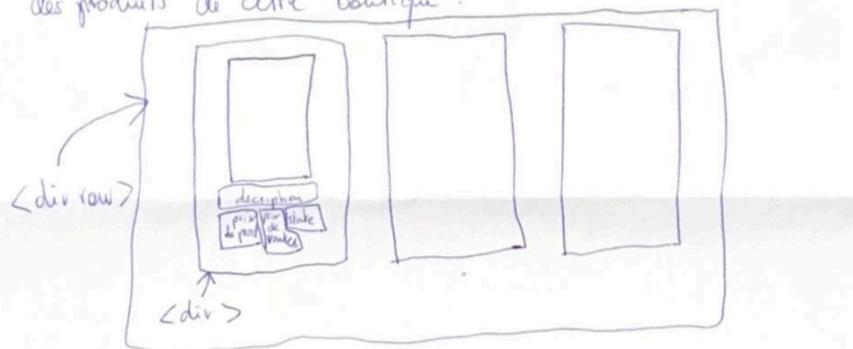
Alors il faut pour clients.php ajouter une rubrique "Liste des produits" qui quand on clique dessus on est redirigé vers une page liste-produits.php.

Alors si on est admin:

menu déroulant qui permet de sélectionner une boutique

[Choisir une boutique] [Valid]

et quand l'admin a choisi une boutique et a cliqué sur Valid. Alors il y a la liste des produits de la liste des produits de cette boutique :



Et quand la personne n'est pas admin
on a pas le menu déroulant, on a juste la liste des produits de la boutique à laquelle elle est liée.

annexe 1.5 :



LA BOUTIQUE



Sac à dos ERREA
Logo Face 30€



Sweat Russell Unisexe / Femme
Logo Face 30€



Veste Russell Unisexe / Femme
Logo Coeur 28€



Veste Running
Logo Coeur 34€



Tshirt Russell Unisexe / Femme
Logo Face 12€



Polo A&Q Unisexe / Femme
Logo Coeur 14€



Legging long
Logo Poche 24€

Legging 3/4
Logo Poche 24€



Casquette NewEra
Logo brodé Face 25€



Masque UNS avec pince nez
Logo Droite 10€

D'autres couleurs et produits disponibles dans la boutique !

POUR COMMANDER

Commandez / payez directement en ligne vos produits de l'association sur notre boutique

annexe 1.6:

FMarketFactory

VOS PRODUITS

- Liste des produits

LES COMMANDES

- Suivi des commandes
- Information
- Analyse des ventes

ADMINISTRATION

- Gestion des comptes

FLYERS

- Faire un flyer

Panier

Votre panier est vide.

Suivi des commandes

Accueil > Commandes

Nombre de boutiques ouvertes (29)

Choisir une boutique

Filtrer l'affichage

Les commandes

82 commandes

Notre Dame de France Malakoff
10/03/2024 Mehdid 21:42:19 Adel

1	7/8 ans	T-shirt unisexe bleu marine Notre dame de France
Non traité		

FMarketFactory

VOS PRODUITS

- Liste des produits

LES COMMANDES

- Suivi des commandes
- Information
- Analyse des ventes

ADMINISTRATION

- Gestion des comptes

FLYERS

- Faire un flyer

Panier

Votre panier est vide.

Suivi de la liste des produits

Accueil > Liste des produits

Nombre de boutiques ouvertes (29)

Albert de Mun

Tshirt Blanc Logo face
12 15

T-shirt noir Logo face
12 15

Tshirt blanc Logo cœur
10 13

MarketFactory

VOS PRODUITS

- [Liste des produits](#)

LES COMMANDES

- [Suivi des commandes](#)
- [Information](#)
- Analyse des ventes**

ADMINISTRATION

- [Gestion des comptes](#)

FLYERS

- [Faire un flyer](#)

Panier
Votre panier est vide.

Analyse des ventes

Accueil - Ventes

Mois

Mois	Ventes
Janv	0
Févr	~60
Mars	~20
Avr	0
Mai	0
Juin	0
Jul	0
Août	0
Sep	0
Oct	0
Nov	0
Déc	0

Tableau des ventes

Accueil - Commandes

Mois	Nombre commandes	CA	Panier Moyen
Mars	26	[REDACTED]	[REDACTED]
Février	1	[REDACTED]	[REDACTED]
Février	[REDACTED]	[REDACTED]	[REDACTED]

MarketFactory

VOS PRODUITS

- [Liste des produits](#)

LES COMMANDES

- [Suivi des commandes](#)
- [Information](#)
- [Analyse des ventes](#)

ADMINISTRATION

- Gestion des comptes**

FLYERS

- [Faire un flyer](#)

Panier
Votre panier est vide.

Gestion des comptes admin

Accueil - Commandes

--Choisissez une boutique--

Nom	Prénom
Email	Téléphone

Avertir

Créer Admin Client

Nom	Prénom	Email	Numéro	Boutique	ACTIONS
admin	julien	email		Albert de Mun	Supprimer

MarketFactory

- Crée un flyer
- Tshirts
- Polos
- Chemises
- Sweats
- Pull
- Vestes
- Joggings
- Jogging
- Produits Phares

Panier
Votre panier est vide.

Construire une boutique

Accueil - Crée son flyer

ss Soumettre

Tshirts de Market Factory

0 €

Tshirt Manches Courtes Unisex B&C
[Cœur] Ajouter au panier

Tshirt Coton Bio Manches Courtes Enfant Native Spirit
[Cœur] Ajouter au panier

Tshirt Coton Bio Manches Courtes Unisex B&C
[Cœur] Ajouter au panier

0 €

MarketFactory

- Crée un flyer
- Tshirts
- Polos
- Chemises
- Sweats
- Pull
- Vestes
- Joggings
- Jogging
- Produits Phares

Panier
Sweat Capuche Unisex Just Hoods

Sweat Capuche Unisex Just Hoods
Cœur
0€ + 4€ pour le flocage cœur
LOGO1 Choisir
Position X :
250
Position Y :
140
Taille :
23
Rotation :
0
Changer la position

MarketFactory

- Crée un flyer
- Tshirts
- Polos
- Chemises
- Sweats
- Pull
- Vestes
- Joggings
- Jogging
- Produits Phares

Panier
Sweat Capuche Unisex Just Hoods

Construire une boutique

Accueil - Crée son flyer

Nom du flyer

Nom de la boutique

Description de la boutique

Logo 1 Choisir le fichier aucun fichier sélectionné
Logo 2 Choisir le fichier aucun fichier sélectionné
Logo 3 Choisir le fichier aucun fichier sélectionné

Créer