Dossier de Synthèse

Aeroclub

Une image contenant croquis, dessin, Dessin au trait, clipart

Description générée automatiquement

Al karshi Abdullrahman

Titre visé : Développement Web et Mobile

Organisme de Formation : Elan Formation

Session : 2024

**Table des matières**

1. Présentation
2. Parcours Personnel
3. Projet de formation : AeroClubBlog
4. Cahier des charges
5. Compétences couvertes par le projet
6. Technologies et Outils
7. Configuration de l’environnement de travail
8. Gestion de projet
9. Front-End : de la maquette jusqu’au projet
10. Les Définitions UX / UI
11. Charte Graphique & Typologie
12. Installation depuis Google Font
13. Types de Caractères Oswald
14. Couleurs de bases
15. Réalisation
16. Images libres de Droit avec Pixels
17. Réalisation Non Connecté
18. Barre de Navigation
19. Pied de page (Footer)
20. Page d’accueil
21. Listes des Categories
22. Listes des Articles
23. Listes des Posts
24. Bouton Inscription
25. Bouton Connexion
26. Réalisation Connecté
27. Onglet Ajout
28. Onglet Ajout d’une Catégorie
29. Onglet Ajout d’un Article
30. Onglet Déconnexion
31. Onglet Mes Catégories
32. Onglet Mes Articles
33. Onglet Lancer un Post
34. Camouflage de l’onglet S’inscrire
35. Responsive design
36. SEO
37. Définition du SEO
38. Comment augmenter la visibilité de son site
39. RGPD
40. Design Pattern
41. MVC
42. Partie Back-End
43. MCD, MLD : notions théoriques
44. Base de Donnée
45. Framework Symfony : POO et Doctrine
46. Sécurité
47. Mot de Passe et le Regex
48. Utilisation de rôles utilisateurs
49. Attaque par force brute
50. Présentation d’une fonctionnalité principale
51. Axes d’améliorations
52. Conclusion
53. Remerciement

ANNEXES

1. Annexes
2. Maquettages
3. Bureau
4. Tablette
5. Mobile
6. Modèles de données
7. MCD
8. MLD
9. Entité
10. Fonctionnalité
11. Controller
12. Template
13. Repository
14. public
15. Security
16. FormType Symfony et affichage dans un template Twig
17. Présentation
18. Parcours Personnel

Passionné d’aviation et des technologies, j’ai toujours été intéressé par l’informatique mais c’était selon moi un secteur où les mathématiques avait leurs places ce qui n’était pas mon point fort mais j’ai toujours voulu apprendre et m’y lancer curieusement.

Après plusieurs réorientations dans le Commerce, Cette fois-ci je m’y suis mis sérieusement dans l’informatique en m’y intéressant de plus prêt et notamment dans le développement Web.

J’ai commencé mon apprentissage en autodidacte à l'aide d'Internet et de documentations que je pouvais acquérir en ma possession

À la suite de plusieurs Certifications OpenClassroom, Cisco en Cybersécurité et ANSSI dans le domaine de la sécurité Informatique notamment, je me suis résolu à réaliser une formation chez Elan Formation.

1. Projet de formation : AeroClubBlog

Ce projet aura pour but de créer un blog autour de l’aviation, ou des passionnés d’aviation pourront interagir entre eux et ainsi partager leurs expériences comme pilote ou co-pilote en prodiguant des conseils et partagé leurs expériences de vols sans limite et ainsi lancer la discussion concernant leurs interrogations pour les nouveaux passionnés et ainsi leurs ouvrir la voie vers les cieux.

La source de ce travail provient d’un intérêt personnel pour l’aviation étant donné que je souhaite un jour passer le permis PPL (Licence Pilote Privé) alors j’ai décidé d’allié mes 2 passions : l’Informatique et l’Aviation.

1. Cahier des charges

Voici une liste exhaustive des fonctionnalités de mon projet Aeroblog :

• Pouvoir se créer un compte et accéder à une interface utilisateur où ce dernier pourra :

* S’Inscrire
* Se Connecter
* Voir la liste des Articles
* Voir la liste des Catégories
* Voir la liste des Posts

• Pouvoir utiliser le blog dans sa totalité, une fois connecté où ce dernier pourra :

* Se Déconnecter
* Lancer un Post
* Voir la liste des Catégories qu’il a crée avec la possibilité de les modifié et ou les Editer
* Voir la liste des Articles qu’il a créé avec la possibilité de les modifié et ou les Editer
* Ajout d’une Catégorie
* Ajout d’une Article

1. Compétences couvertes par le projet

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Le tableau ci-dessus est tiré du REAC (Référentiel Emploi Activités Compétences) du titre Développeur Web et Web mobile et montre les compétences que j’ai acquises et développées au cours de ma formation chez ELAN ainsi qu’au cours de la réalisation de mon projet. C’est un titre full Stack, j’ai donc travaillé pour réaliser le côté front-end et le côté back-end de mon projet.

Dans ce projet, la première compétence que j’ai mise en action et la première compétence du référentiel : l’installation et la configuration de l’environnement de travail en fonction du projet web ou web mobile.

La partie front-end est la partie visible à l’utilisateur de mon application. Pour la réaliser, j’ai mis en application ces compétences :

➢ Réaliser la maquette de mon projet pour les pages principales en pensant à inclure des versions tablettes et mobiles

➢ Réaliser la partie statique de mon site via du langage de balisage.

➢ Réaliser la partie dynamique des interfaces utilisateur web ou web mobile via des langages dynamiques comme PHP.

La partie back-end est la partie invisible à l’utilisateur de mon application. Pour la réaliser, ce sont ces compétences qui sont concernées :

➢ Mettre en place une base de données relationnelle

➢ Développer des composants d’accès aux données SQL et NoSQL .

Pendant la rédaction de mon dossier et au courant de la formation, j’ai eu l’opportunité de développer des compétences transverses dans le métier de Développeur Web et Web Mobile :

➢ Utilisation de l’anglais technique pour se documenter et communiquer.

➢ Développer sa gestion de projet.

➢ Rédiger de la documentation et des rapports.

1. Technologies et Outils

Pour la réalisation de ce projet, j’ai utilisé :

Un langage de balisage

▪ HTML (HyperText Markup Langage) : pour tous les templates. HTML constitue le squelette de mes pages et permet d’interagir avec des langages de programmation.

Des langages de programmation

▪ PHP (HyperText Preprocessor) : génération de données dynamiques. Partie back-end de l’application, non visible aux utilisateurs

▪ CSS (Cascading Style Sheets) : apparence et habillage des pages web.

▪ SQL (Structured Query Language – Langage de requêtes structurées) : communication avec la base de données.

Un Framework PHP (bibliothèque logicielle)

• **Symfony** : permet de mieux structurer son projet puisqu’il nous donne un cadre de travail précis qui suit le design pattern Modèle-Vue-Présentation. Permet d’avoir une séparation de la logique métier. Le framework propose aussi une grande quantité d’atouts pour pouvoir sécuriser correctement son application. Symfony permet d’accélérer sa productivité via des fonctions natives du framework.

Symfony comprend aussi :

▪ Doctrine, un ORM (Object-Relational Mapping) qui permet de travailler avec des objets plutôt qu’avec des tableaux associatifs.

▪ Twig : moteur de template qui permet d’écrire du code contenu dans les templates. Twig propose aussi beaucoup de filtres qui permettent des fonctionnalités diverses comme formater une date.

• Un gestionnaire de dépendances : Composer. Il permet de gérer les librairies ou les paquets sur lesquels reposent le projet. Cet utilitaire permet d’être sûr que le projet a besoin de toutes ses ressources pour fonctionner. Composer est nécessaire pour pouvoir travailler avec Symfony.

Des moteurs de recherche et des site web

▪ Chrome : pour les recherches et comme moteur de recherche

• Outil de maquettage : **Figma** qui est un Outil qui sert de maquettage en ligne.

Environnement de développement :

• Editeur de Texte : Visual Studio Code

Serveur Local

• Wampserver est une plateforme de développement Web permettant de faire fonctionner localement des scripts PHP.

C’est un environnement permettant l’interprétation des scripts et on y trouve également phpMyAdmin pour l'administration Web des bases MySQL

1. Configuration de l’environnement de travail

Je commence par installer **Visual Studio Code**, mon environnement de développement, à cette adresse : <https://code.visualstudio.com/>. J’utilise la dernière version stable pour Windows.

Également je vais devoir installer **Wampserver**, qui va stocker mon projet en local sur mon PC, je dois donc l’installer via ce lien <https://wampserver.aviatechno.net/>

Celui-ci est une plateforme de développement Web, permettant de faire fonctionner localement des scripts PHP. WampServer est un environnement permettant d’interpréter un script, ainsi que phpMyAdmin pour l'administration Web des bases MySQL

J’ai dû télécharger de **scoop** à cette adresse, <https://scoop.sh/> , utilitaire en ligne de commande qui permet d’installer d’autres programmes.

Puis je vais télécharger **Symfony** CLI : <https://symfony.com/download> permet d’installer le framework.

J’ai dû télécharger **Composer** à cette adresse <https://getcomposer.org/download/> car c’est un élément vital pour faire fonctionner Symfony

Je consulte la documentation Symfony et je tape cette commande dans mon terminal Windows

Commande permettant d’installer **scope**

*Set-ExecutionPolicy RemoteSigned -scope CurrentUser*

Une fois que scope est installé, j’ai vais pouvoir installé symfony.

Commande permettant d’installer **symfony**

*scoop install symfony-cli*

Commande permettant d’installer **composer**

*composer install*

J’ai encore besoin de quelques étapes supplémentaires pour configurer mon projet, notamment l’ajout d’extensions sur VSCode qui permettent d’être plus productif comme par exemple TWIG Pack.

- TWIG pack (par Bajdzis) : VScode ne prend pas en charge naturellement la syntaxe de Twig. Cette extension permet donc d’y voir plus clair dans son code et de colorer sa syntaxe.

1. Gestion de projet

Dans ce projet, j’ai utilisé une méthode assez spéciale qui consiste à faire une liste de tâche quotidiennement entre 1 à 3 tâches par jour afin d’avoir une certaine régularité.De plus, cette méthode m’a permis de lutter contre la procrastination.

1. Front-End : de la maquette jusqu’au projet

Le maquettage est une étape de préparation où l’on visualise le projet par des moyens techniques différents. Le maquettage permet aussi de réfléchir à comment va se dérouler la navigation pendant la consultation du site.

La notion de **front-end** désigne dans une application logicielle ou web tous les éléments qui font les interfaces entre les Humains et les machines. Le maquettage est donc la première étape de la construction de la partie front-end du projet.

1. Les Définitions UX / UI

**(User Expérience (Expérience de l’Utilisateur) / User Interface (Interface Utilisateur)**

L’UI (User Interface – Interface utilisateur) dans sa définition large est la façon dont est pensée l’affichage utilisateur de l’application lors de son utilisation. Voici une UI traditionnelle en guise d’exemple : Une image contenant texte, Appareils électroniques, capture d’écran, Page web

Description générée automatiquement

L’UX est un terme anglophone (User eXperience) désignant l’expérience de l’utilisateur avec l’objet de son interaction. C’est un élément important à optimiser pour permettre une navigation fluide et agréable. C’est un élément qui comprend les concepts d’ergonomie et d’accessibilité.

Voici quelques éléments pour rendre une navigation agréable, ergonomique et accessible :

- Avoir la barre de navigation accessible facilement : il faut pouvoir naviguer facilement. - Dans un formulaire, indiquer quels champs sont nécessaires et quels sont les conditions du mot de passe.

- Avoir un responsive design efficace

- Précision concernant les champs mots de passe

B) Charte Graphique & Typologie

1. Installation depuis Google Font

Pour réaliser cela je me rend sur Google Font est une vaste bibliothèque contenant un grand nombre de polices de caractères libres.

Depuis ce site <https://fonts.google.com/>

Et insérer dans cette ligne



1. Types de Caractères Oswald

Taper le caractère voulu en barre de recherche sur le site Google Font

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Afin de rendre fonctionnel sur mes onglets



1. Couleurs de fond

J’ai utilisé le site Code couleur qui m’a permis de choisir la couleur de mon choix pour les onglets et le footer (bas de page).

Une image contenant texte, capture d’écran, logiciel, Système d’exploitation

Description générée automatiquement

1. 5) Réalisation

La maquette doit donc être conforme au projet fini, sans les animations et les interactions. Voici un exemple de la page d’accueil avec les vues bureau, tablette et mobile en Annexe.

Images libres de Droit avec Pexels

Pexels fourni des photos de haute qualité et entièrement gratuites.

6)Réalisation Non Connecté

1. Barre de Navigation

La barre de navigation est présente en haut de la page. Des onglets Aeroclub qui redirige vers la page d’accueil, un onglet déroulant nommé « **Listes** » une fois dessus qui comporte et affiche **Listes des Categories**, **Listes des Articles**, **Listes des Posts** , un onglet « **S’Inscrire** » et également un onglet « **Se Connecter** »

1. Pied de page (Footer)

Le pied de page permet d’accéder aux conditions générales d’utilisation, d’accéder à un réseaux sociaux (Linkedin) et également d’accéder au portfolio du créateur du blog pour ainsi entrer en contact avec celui-ci.

- Les Conditions Générales d’Utilisations permettent de spécifier l’utilisation et la récolte des données.

- Présence des mentions légales.

1. Page d’accueil

La page d’accueil se présente avec notre barre de navigation présenté précédemment en haut, une image que j’ai prise du site [pexels.com](https://www.pexels.com/fr-fr/)  qui nous montre un avion à terre et également un petit paragraphe écris par mes soins présentant le blog et le but de ce blog qui est un lieu de discussion virtuel. Et pour finir le Footer en pied de page pour terminer cette description de ma page d’accueil.

1. Liste des Categories

La liste des catégories montre les catégories qui ont été lancé par les autres utilisateurs et par soi-même si cela a déjà été fait. Ses catégories sont bien sûr issues de la base de donnée. Cette liste est visible que l’on soit connecté ou non.

1. Liste des Articles

La liste des articles montre les articles qui ont été lancé par les autres utilisateurs et par soi-même si cela a déjà été fait. Ses articles sont bien sûr issus de la base de données également. Cette liste est visible que l’on soit connecté ou non.

1. Liste des Posts

La liste des posts nous affiche l’ensemble des posts qui ont été publié par les utilisateurs. On peut y voir le titre de l’article qui est un lien cliquable qui va permettre à l’utilisateur de cliqué sur celui-ci afin de le redirigé vers le post en question afin d’y apercevoir les commentaires qui ont été publié par les utilisateurs intéressé par ce post. On peut également y voir le pseudo du créateur de ce post ainsi que la date de création mais aussi la catégorie auxquelles ce post à été attribué par le fondateur de ce post.

Et pour finir une image si celle-ci à était ajouté par le créateur de ce post, celle-ci est facultative.

1. Bouton Inscription

On peut y voir un formulaire avec des champs de saisies Pseudo, Email et Mots de Passe et d’une case à coché qui est obligatoire de cocher sinon l’inscription ne se fera pas. Tout ceci accompagné d’un bouton S’inscrire. Le pseudo sera visible ainsi que l’Email mais mot de passe saisi sera lui visible mais avec des caractères en forme de point noirs.

1. Bouton Connexion

On y voit 2 champs de saisies Pseudo et Mot de Passe et d’un bouton Se connecter. Le pseudo sera visible et le mot de passe saisi sera lui visible mais avec des caractères en forme de point noirs.

1. Réalisation Connecté
2. Onglet Ajout

Une fois connecté, cet onglet est donc visible et une fois la souris dessus, on peut y voir une liste déroulante nous montrant 2 autres onglets : Ajout d’une Catégorie et Ajout d’un Article.

1. Onglet Ajout d’une Catégorie

Une fois connecté, cet onglet est accessible, en cliquant dessus on peut y voir un champ de saisi permettant d’ajouter une catégorie et un bouton Ajouter permettant de l’ajouter à la base de données. Après avoir cliqué sur ce bouton, il redirige automatiquement vers une autre page permettant cette fois de créer un nouvel article.

1. Onglet Ajout d’un Article

Une fois connecté, cet onglet est accessible, en cliquant dessus on peut y voir des champs de saisi et également des listes déroulantes notamment pour Categorie qui liste les catégories enregistrées précédemment, un champ Title permettant d’y insérer le Titre de l’article, c’est celui-ci qui sera cliquable et qui redirigera vers ses post. Et également un champ Date de Création afin d’y noté le jour de création ainsi que l’heure. Accompagné d’un bouton Enregistrer permettant d’enregistrer tout ceci en base de données et permettre de voir ce nouvel article dans la liste des Articles.

Cette page va être accompagné d’un astérix en rouge qui précise que si l’utilisateur ne trouve pas sa catégorie il peut cliqué sur un lien qui redirige vers la page d’ajout d’une catégorie.

1. Onglet Déconnexion

Une fois connecté, cet onglet apparait et c’est celui-ci qui va fermer la session.

1. Onglet Mes Catégories

Une fois connecté, cet onglet apparait et c’est celui-ci va lister toutes les catégories créer par l’utilisateur.

1. Onglet Mes Articles

Une fois connecté, cet onglet apparait et c’est celui-ci qui va lister tout les articles créer par l’utilisateur.

1. Onglet Lancer un Post

Une fois connecté, cet onglet apparait et c’est celui-ci permettre de créer un post, on va y trouver une liste déroulante permettant de choisir un article, un bouton permettant d’ajouter une image (celle-ci est facultatif) et également le contenu d’un post et également un bouton enregistré afin de transmettre ses informations à notre base de données.

1. Camouflage de l’onglet S’inscrire

Une fois connecté, l’onglet s’inscrire ne sert pas à grand-chose étant donné que l’utilisateur est déjà inscrit et même connecté alors pour gagner de la place et rendre notre blog plus ergonomique j’ai décidé de tout simplement le camouflé.

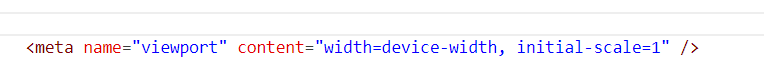
Ne vous inquiétez pas, il sera bien de retour une fois que vous serez déconnecté.

1. Responsive design

Le « responsive design » désigne la façon de concevoir son application de façon que son affichage se comporte différemment en fonction du support sur lequel il est visionnée et de la taille de la fenêtre du navigateur web.

Il existe plusieurs moyens de rendre son site adaptable :

Viewport :



L’attribut content permet de mettre la largeur de la page à la même largeur d’écran de l’appareil à partir duquel la page est visionnée.

La partie initial scale mis à 1 nous permet de ne pas zoomer sur la page au chargement. Ces attributs évitent ainsi l’effet « dezoom » qui se produit lorsque l’on conçoit un site en bureau et qu’on le visionne en mobile.

Tailles d’unités

Il est possible d’utiliser plusieurs types d’unité.

• Pixels : dimensions fixes : unité absolue. Un pixel est toujours la taille du pixel : un carré de 2x2 pixels aura toujours la même taille indépendamment du contexte.

• Pourcentages : expression en pourcentage d’écran. Un paragraphe dont la width est 50% prendra la moitié de la largeur de la page

Media queries

Elles permettent de conditionner des propriétés CSS à certaines propriétés comme la taille d’écran. Il existe des tables récapitulatives et des statistiques qui donnent la taille moyenne d’un écran bureau, tablette et mobile, qui permettent avec les médias queries d’ajuster des comportements spécifiques pour la moyenne de chaque support.

Dans mon projet, j’ai adopté une autre stratégie : « au pixel minimum » où je redimensionne jusqu’aux points où l’affichage des éléments est désorganisé et casse l’agencement de la page.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Dans mon projet, j’ai utilisé ces média queries pour changer l’orientation des propriétés flex. Par exemple, un élément un onglet par exemple qui s’affichait en ligne sur la version bureau va s’afficher en colonne sur la version mobile.

Le responsive design est un élément primordial pour pouvoir référencer son site. Aujourd’hui, un site dont le visionnage est totalement non adaptable n’est même pas référencé pour le SEO (Référencement Naturel).

9 – SEO

1. Définition du SEO

Le SEO est un acronyme qui correspond à Search Engine Optimization, ce qui correspond à l’optimisation de la découverte du site par un moteur de recherche appelé également Référencement Naturel. Il existe aussi un autre moyen d’augmenter le référencement d’un site via les publicités, ce procédé est appelé le SEA pour Search Engine Advertising.

1. Comment augmenter la visibilité de son site

Il existe plusieurs moyens d’optimiser son site pour qu’il apparaisse plus haut dans les résultats de moteurs de recherche.

- Utilisation de mots-clés (« keywords »).

Une image contenant texte, Police, capture d’écran, ligne

Description générée automatiquement

- Utilisation de titres via la balise (« title ») : uniques, précis, courts. Exemple avec de la syntaxe Twig :

Une image contenant Police, blanc, capture d’écran, conception

Description générée automatiquement

- Utilisation de la balise meta avec l’attribut description

- Utilisation des titres avec les balises (« h1 », « h2 », etc.) pour hiérarchiser son contenu.

- Utiliser des balises pertinentes.

- Utiliser du texte pour la navigation.

- URL simples compréhensibles par un humain.



- Contenu pertinent et nouveau

- Affichage des images via la balise et non via du CSS.

- Site accessible (propriété « alt=’’ ‘ ’ » sur les balises images pertinentes), contraste colorimétrique afin qu’elle soit visible par tous.

Le RGAA (Référentiel Général d’Amélioration d’Accessibilité) propose des tests pour vérifier l’accessibilité d’un site.

- Responsive design efficace.

- Performance du site : c’est le paramètre qui a le plus de poids.

- Réaliser une carte du site pour les robots google.

- Utiliser des liens extérieurs (ex : réseaux sociaux).

D’où l’importance de les faire apparaître dans la maquette. Google Lighthouse permet de tester son site par rapport à de nombreux paramètres et donne un score SEO, avec des pistes d’amélioration. Il y a divers outils de notation également GT Metrix.

Une image contenant texte, Police, capture d’écran, logo

Description générée automatiquement

10 – RGPD

Le Règlement Général sur la Protection des Données est le texte qui régit la façon dont les données des utilisateurs européens sont traitées. En France, c’est un texte dans la prolongation de la loi Française Informatique et Libertés de 1978.

Le RGPD met l’accent sur la protection de la personne physique qu’est l’utilisateur via plusieurs volets :

- Principe d’utilisation licite, loyale et transparente des données au regard de la personne concernée : il faut le consentement de l’utilisateur, ne pas induire ce dernier en erreur et utiliser ses données uniquement pour la chose précise dont on demande les données.

- Utilisation finie des données : il faut définir pourquoi on recueille tel donnée, dans quel cadre on l’utilise et pourquoi. On ne peut pas revenir rétroactivement dessus sans modifier les C.G.U et notifier l’utilisateur.

- Minimisation du recueillement des données : il faut demander ce qui est uniquement nécessaire à l’application.

Dans mon projet Aeroblog, je ne demande qu’un email pour pouvoir donner la possibilité à l’utilisateur de récupérer son compte via une demande de récupération de mot de passe sur Mailtrap qui est un serveur SMTP qui permet d’envoyer des emails, j’ai dû le configuré et le relier à mon projet afin qu’il puisse envoyé des mails aux destinataires et ainsi permettre la modification du mot de passe et ainsi résoudre le problème du mot de passe oublié.

- Principe d’exactitude : les données personnelles doivent être exactes et tenues à jour.

- Limite de la conservation des données : l’administrateur du site ne peut pas conserver indéfiniment des données personnelles. Selon la sensibilité de celles-ci, la durée de conservation diffère et les données doivent être effacées au moment où celles-ci tombent hors de leur date de conservation.

- Intégrité et confidentialité : les données personnelles doivent être sécurisées, c’est-à-dire non accessibles par n’importe qui, et protégées selon la nature de ces dernières. Il faut aussi les protéger contre la destruction, la perte ou les dégâts d’origine accidentels.

- Droit à l’oubli : L’utilisateur doit pouvoir supprimer son compte et son identité dans le restant du site doit être anonymisé.

11 – Design Pattern

1. Définition

Un design pattern en informatique est un moyen de résoudre un problème de conception courant. C’est une solution réutilisable. Un bon design pattern permet de faciliter la maintenance du logiciel et accélère son développement.

Les designs patterns ne sont liés à aucun langage de programmation spécifique puisque ce sont des stratégies conceptuelles. La traduction en français, « Patron de Conception », permet d’illustrer ce concept. Pour synthétiser, c’est une formalisation de bonnes pratiques.

Il y a 3 catégories de design pattern :

- Création (Creational) : nous nous concentrons sur la création de classes et l’instanciation de ces dernières. Exemples :

Factory : modèle de conception le plus fréquemment utilisé. Une classe se charge d’instancier d’autres classes nécessaires au fonctionnement du logiciel ou de l’application. Symfony utilise ce principe.

Singleton : Un Objet est lié à une classe instanciée

Structuration (Structural) : Ce sont des concepts qui se concentrent sur l’assemblage de classes et d’objets pour en faire de grosses structures en les gardant flexibles et efficaces. Exemples :

Adaptateurs : permettre à des objets avec des interfaces différentes d’interagir ensemble.

2) Façade : création d’une interface Façade simplifiée qui couvre tout un soussystème d’autres classes plus complexes. Ce design pattern est utile pour des systèmes complexes qui demandent l’appel à plusieurs méthodes.

3) Composite : agencement en « arbre » : les objets sont rangés dans des arborescences pour qu’ils soient traités comme des objets individuels.

- Comportement : cette famille comprend les design pattern qui décrivent les modes de communication entre les classes et la distribution de leurs responsabilités.

Exemples :

1) Chaîne de responsabilité : passage de requêtes à travers d’éléments qui permettent de les transmettre d’un élément à un autre. Quand un élément reçoit une requête, il décide s’il la passe ou prochain ou s’il la laisse tomber.

2) Commande : une requête devient un objet en tant que tel et contient toutes les informations de la requête. Cette transformation permet de transmettre la requête via une méthode avec des arguments, de retarder ou mettre en attente l’exécution de la requête.

3) Observer : Une classe « observateur » est lié à d’autres classes et ce pattern fonctionne d’une façon similaire à un système d’abonnement. L’observateur va écouter d’autres classes qu’on peut considérer comme des abonnées et recevra des notifications quand ces derniers changent.

4) Stratégie (strategy) : patron qui permet de définir une famille d’algorithmes rangés dans des catégories séparées et où les objets sont interchangeables.

12) MVC

Le pattern MVC est le plus connu de tous.Il permet une meilleur organisation autour de 3 piliers que son Model , View et les Controller.

Le Model représente la structure des données.

La View représente la vue ou plutôt l’interface graphique la partie visibile.

Le Controller represente lui le cœur de l’application,il se situe aux centres des 2 autres piliers que son Model et View. Les requêtes du client fait dans l’interface graphique vont être redirigé. Les views vont être dirigé vers le controller qui va manippuler les données qui les redirige vers le model et celui-ci va mettre à jour la vue avec la requête demandé par l’utilisateur. (Image en Annexe MVC)

Note : quelques exemples de codes pour les requêtes reçues très récurrentes

- 200 : tout est OK

- 3XX : redirection

- 4XX : erreur

- 403 : accès refusé

- 404 : ressource non trouvée

- 5XX : erreur dans le serveur distant

1. Partie Back-End
2. MCD, MLD : notions théoriques

Avant de commencer à écrire du code, je me suis interrogé sur la nature de mes données, comment je vais les organiser et les ranger. La représentation de la relation des différentes entités entre elles dans le projet est le Modèle Conceptuel de données. Il fait partie de la méthodologie Merise utilisée dans ce projet.

La méthode Merise n’est pas propre au domaine informatique : elle est utile dans tout système d’information et de gestion.

Elle utilise trois niveaux d’abstraction :

- Le MCD : la représentation des entités et des relations entre elles est effectuée en respectant ces règles :

➢ Une entité = un rectangle.

➢ Les propriétés d’une entité (nom, prénom, id, etc.) sont listées dans le rectangle de l’entité correspondante.

➢ Les relations sont représentées par des ovales liés par des traits pleins. Par convention, elles sont nommées avec des verbes à l’infinitif.

➢ Les cardinalités sont les nombres inscrits dans les traits qui représentent les relations.

Elles ont un sens de lecture et permettent de déterminer combien de fois une instance d’une entité peut être liée avec d’autres instances d’une autre entité.

- Le MLD : modèle Logique de Données. Il est utile pour se représenter les tables présentes dans le projet et la façon dont elles devraient interagir entre elles. Le MLD fait apparaître chaque clef primaire d’une entité et éventuellement ses clefs étrangères. C’est dans le MLD qu’apparaissent le type de données.

- Le MPD : Modèle Physique de Données. C’est l’étape où l’on commence à sortir de l’abstrait et du concept. Les entités sont écrites en tables comme dans un SGBD (Système Gestion Base de Données), les propriétés sont visualisées en champ, le type de données est précisé.

Il existe plusieurs types de relations :

❖ OneToOne (Une à une)

❖ OneToMany (Une à Plusieurs)

❖ ManyToOne (Plusieurs à Une)

❖ ManyToMany (Plusieurs à Plusieurs) générant une table associative.

Les règles du MCD à MLD

Règle numéro 1

Une association de type 1 :1, c’est un cas exceptionnel

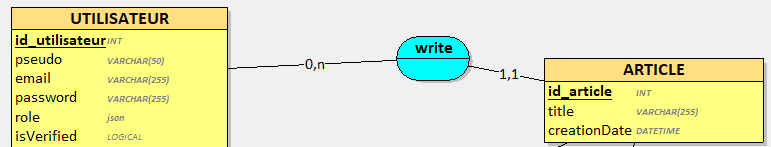
Règle numéro 2 :

Une association de type 1:N (c’est à dire qui a les cardinalités maximales positionnées à « 1 » d’une côté de l’association et à « n » de l’autre côté) se traduit par la création d’une clé étrangère dans la relation correspondante à l’entité côté « 1 ». Cette clé étrangère référence la clé primaire de la relation correspondant à l’autre entité.

Règle numéro 3 :

Une association de type N :N (c’est à dire qui a les cardinalités maximales positionnées à « N » des 2 côtés de l’association) se traduit par la création d’une table dont la clé primaire est composée des clés étrangères référençant les relations correspondant aux entités liées par l’association. Les éventuelles propriétés de l’association deviennent des attributs de la relation.

Exemple de MCD :



Il y a un sens de lecture du point de vue de l’entité Utilisateur, je veux qu’un Utilisateur puisse écrire plusieurs articles, donc la relation Utilisateur -> Article est de Un Utilisateur à plusieurs Articles (OneToMany). La cardinalité est de ce fait notée 0,n dans le sens Utilisateur->Article.

Le deuxième chiffre donne le nom de la relation (oneToMany).

On lit donc dans l’autre sens la relation inverse : un Article appartient essentiellement à un Utilisateur : cette relation est notée 1.1 pour cette raison. La relation dans ce sens est donc une ManyToOne.

Exemple de MLD:

Une image contenant texte, ligne, capture d’écran

Description générée automatiquement

Dans le cas de la relation **ManyToMany**, il y a création d’une table associative. C’est une table qui met en liaison les entités à partir de laquelle cette table associative est née : elle contient les clefs primaires des tables qu’elle met en liaison en tant que clefs étrangères.

Dans notre cas ici en MCD, nous avons une relation **OneToMany** depuis l’entité Article vers Utilisateur ce qui à pour effet quand on passe en MLD d’insérer la Clé Primaire de Utilisateur (**id\_utilisateur**) à l’entité Article et cette clé primaire deviendra clé étrangère dans cette entité Article.

Vous pouvez retrouver le MCD et MLD complet en page Annexe

1. Base de Donnée

Une base de données est un ensemble organisé d’informations. Il en existe plusieurs sortes :

- BDD relationnelles (SQL) : organisation en tables composées d’enregistrements et de champs avec des schémas.

- BDD non relationnelles est une Base de données qui n’utilise pas de schéma tabulaire sous forme de lignes et de colonne

Ci-dessous des bases de données plus anciennes qui ont évolué vers les BDD relationnelles.

Pour mon projet, je vais utiliser une base de données relationnelles.

Une base de données SQL présente les avantages d’être :

- Simple visuellement : visions globales de la base de données

- Faciles à interroger avec le langage SQL pour chercher une information.

- Garantes de l’intégrité des données :

▪ Utilisation de clés

❖ Primaires : elles permettent d’identifier chaque ligne dans une table. Elles permettent de s’assurer que chaque enregistrement (une ligne) soit unique. Exemple : Tout les Utilisateurs contient un ID unique à chaque utilisateur. Ils seront identifiables via leur ID : 1,2,3…. (Clé primaire).

Par convention les ID sont des Integer (**nombres entiers**) pour faciliter la mémorisation et sont plus légers en termes de mémoire. Les Integer permettent aussi la mise en place de l’auto incrémentation.

❖ Secondaires : elles permettent d’établir un lien entre différentes tables, elles garantissent l’intégrité référentielle de la relation. Ce sont des points d’accroche utiles pour pouvoir effectuer des jointures de tables. En cas de tentative de suppression d’un enregistrement dans une table parente, la contrainte de clef étrangère vérifie s’il existe des enregistrements correspondants dans une table enfant. Cette contrainte est par défaut notée FK pour Foreign Key dans des logiciels de SGBD.

1. Framework Symfony : POO et Doctrine

Symfony est un framework qui repose sur un modèle de programmation spécifique : la programmation orientée objet (POO).

La notion d’objet sont des instances d’une classe, qui ont elles-mêmes chacune des propriétés (exemple pour une voiture : portières, sièges, etc.) avec un statut (publiques, protégées ou privées) et des méthodes (comme démarrer et s’arrêter, qui elles aussi ont un statut).

Voir des exemples de propriétés et de méthodes en annexes

L’ORM Doctrine simule une base de données relationnelles en utilisant des objets PHP. Il sert d’abstraction entre la base de données relationnelle et le code PHP, permettant aux développeurs de manipuler des objets PHP au lieu de travailler directement avec des requêtes SQL. Les entités PHP symbolisent les tables de la base de données et les relations entre elles.

En outre, Doctrine utilise une couche d’abstraction de base de données (DBAL, Database Abstraction Layer) pour interagir avec la base de données. DBAL traduit les requêtes orientées objet de Doctrine en SQL pour MySQL.

Cela permet à Doctrine de communiquer efficacement avec la base de données tout en maintenant l’indépendance du code PHP par rapport à la technologie de base de données spécifique utilisée. Ainsi, DBAL joue un rôle crucial dans le processus de récupération et de modification des données entre Doctrine et la base de données MySQL.

14) Sécurité

A) Mot de Passe et le Regex

L’OWASP (Open Web Application Security Project, organisation de sécurité américaine) recommande un minimum de 8 caractères Unicode minimum avec un maximum de 64 caractères pour éviter les attaques DDOS par déni de mot de passe.

En effet, comme nous allons l’étudier, un mot de passe n’est pas stocké de façon claire dans la base de données.

- Ce n’est pas sécurisé d’un point de vue technique

- La personne responsable de la base de données à accès au mot de passe de l’utilisateur : violation du RGPD.

La CNIL recommande un mot de passe qui comporte au minimum 12 caractères, dont au moins un numérique, une lettre minuscule et majuscule, un caractère spécial.

Dans mon projet Symfony se fait via une regular Expression ou Regex qui donne un motif (pattern) que doit respecter ma chaîne de caractères.

1.   **'min' => 12** : La chaîne fait au minimum 12 caractères.

2. **(?=.\*[A-Z])** : Il faut qu’il y a au moins un caractère unicode capitale (upperCase).

3. **(?=.\*[a-z])** : Il faut qu’il y a au moins un caractère unicode minuscule (lowerCase).

4. **(?=.\*\d)** : Il faut qu’il y au moins un chiffre (digit).

5. **(?=.\*[\W\_])** : Il faut au moins la présence d’un caractère présent dans ce set.

Un mot de passe fort comme celui-ci permet d’en plus du throtling, de réduire le risque de succès d’une attaque par force brute. Un mot de passe fort contribue à allonger le temps de calcul nécessaire pour cracker ce dernier.

Une image contenant texte, capture d’écran, menu, Police

Description générée automatiquement

La totalité des mots de passe en base de données sont bien sur haché.



▪ $2y indique quel algorithme de hachage a été utilisé. Le hachage est le procédé qui permet de transformer un texte en clair en une série de caractères inintelligibles pour un humain. C’est une opération à un sens (destructive), une fois la transformation effectuée on ne peut pas revenir en arrière.

▪ L’algorithme de hachage de base de Symfony est **bcrypt** et $2y indique la version de bcrypt utilisée.

▪ 13 est le coût du mot de passe : c’est un indicateur qui détermine la complexité de l’algorithme. Plus il est élevé, plus il faudra de puissance de calcul pour pouvoir le cracker via une attaque par force brute.

▪ **XVCRh6tGJeSzKQILyy** est le sel. C’est un élément de sécurité supplémentaire, qui augmente la taille d’une table précalculée (type rainbow table) pour pouvoir pirater le mot de passe. C’est aussi un ensemble aléatoire de chiffres et de lettres.

B) Utilisation de rôles utilisateurs

A partir du moment où un site dépasse le stade de site vitrine et progresse dans son interactivité avec l’utilisateur (ex : création d’un compte utilisateur), il est courant de discriminer les utilisateurs selon un rôle qui leur attribuera des permissions leur permettant ou non de réaliser tel ou tel action.

Dans mon projet, un **utilisateur non inscrit** pourra lui **voir les articles et les posts** mais ne pourra pas interagir avec les autres utilisateurs.

Un **utilisateur inscrit** lui pourra **réagir à un post et même lancer un post, ajouté une catégorie et un article**.

C) Attaque par force brute

L’attaque par force brute est un jeu d’essais et d’erreurs qui consiste à tester des combinaisons. Elles sont souvent opérées par des programmes informatiques, et leur efficacité est proportionnelle à la puissance de calcul du processeur sur lesquels ils sont lancés.

Une image contenant texte, capture d’écran, diagramme, conception

Description générée automatiquement

Pour s’en prémunir, la CNIL recommande de mettre en place un blocage à la tentative de connexion après un nombre n d’essais infructueux.

Voici comment elle a été mise en place sur mon projet : App\config\packages\security.yaml

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Ceci s’appelle en termes techniques le « login throttling ». « null » ici signifie que je laisse les paramètres prédéfinis par Symfony 5.4 , c’est-à-dire à 3 essais par minute.

Avoir un mot de passe fort permet aussi de réduire le succès d’une attaque par force brute,

1. Présentation d’une fonctionnalité principale

J’ai choisi de vous présentez la partie de création d’un Post.

Nous devons tout d’abord nous connecter, cette connexion va permettre d’afficher l’onglet qui lui va afficher l’onglet **Lancer un post**. Celui-ci n’étant visible que si l’utilisateur est connecté, Après avoir cliqué sur cet onglet, on y voit une liste déroulante permettant de choisir un article et également l’ajout d’image qui lui est facultatif est va inciter ou non à choisir une image. Ensuite, on retrouve une zone de texte permettant d’écrire du contenu et pour finir on retrouve un bouton Enregistrer qui va permettre d’envoyer ce post. Par la suite, le post qui vient tout juste d’être crée sera lui bien listé dans la liste des posts. Aussi ce post qui vient d’être crée sera aussi visible dans l’onglet

Mes Posts qui affiche et liste les posts crée par un utilisateur et sur ce même page, nous pourront supprimer le post en cliquant sur un bouton.

16)Axes d’amélioration

Voici quelques axes d’amélioration pour mon projet :

• Traduction du site en anglais via bundle PHP Symfony Translation

• Passage à Symfony 7.2

1. Conclusion

Au cours de la réalisation de ce projet ainsi que tout au long de ma formation, j’ai rencontré de multiples obstacles. Pour les surmonter et en tirer des enseignements, j’ai eu l’opportunité de développer mes compétences de recherche documentaire et de communication lors de mes échanges avec l’équipe de formation d’ELAN.

J’ai réalisé l’importance d’une gestion de projet efficace et de la réflexion préparatoire à propos d’un projet avant d’en écrire le code notamment en faisant le maquettage et le Modèle Conceptuelle de Donnée.

En travaillant sur la partie technique, j’ai acquis des compétences de programmation dans différents langages. J’ai réalisé ma formation dans un cadre valorisateur avec des collègues chaleureux et j’en sors très satisfait, avec des idées concrètes de réalisations et de mon futur dans ce domaine.

**Remerciements**

J’aimerai remercier chaleureusement toute l’équipe de formation d’ELAN :

Stéphane SMAIL, Mickaël MURMANN, Quentin MATHIEU,

À toute son équipe administrative,

Merci.

ANNEXE

Version Bureau

Une image contenant texte, avion, capture d’écran, plein air

Description générée automatiquement

Version Tablette

Une image contenant texte, capture d’écran, Site web, Page web

Description générée automatiquement

Version Mobile

Une image contenant texte, capture d’écran, Police, Site web

Description générée automatiquement

Architecture MVC

Une image contenant texte, capture d’écran, diagramme, affichage

Description générée automatiquement

MCD

Une image contenant texte, diagramme, capture d’écran, ligne

Description générée automatiquement

MLD

Une image contenant texte, capture d’écran, diagramme, ligne

Description générée automatiquement

Connexion à Mailtrap

Connexion à la BDD

Mise en place de la Regex

