Le dossier respecte ce plan type :

1. Liste des compétences du référentiels couverte par le projet

Activité type 1 « Développer la partie front-end d’une application web ou web mobile en intégrant les recommandations de sécurité »

* Maquetter une application
* Réaliser une interface utilisateur web statique et adaptable
* Développer une interface utilisateur web dynamique

Activité type 2 « Développer la partie back-end d’une application web ou web mobile en intégrant les recommandations de sécurité »

* Développer les composants d’accès aux données
* Développer la partie back-end d’une application web ou web mobile
* Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce

1. Résumé du projet en français environ 200 à 250 mots

Ce projet consiste à créer et développer une boutique en ligne de vente de vin, pour une société existante. Cette société ne souhaitant finalement pas de boutique en ligne, je me suis permis de garder leurs informations pour tout de même poursuivre ce projet. Cette application doit avoir une partie présentation de la société ainsi qu’une partie boutique en ligne ayant les fonctions suivantes : création d’un compte utilisateur ainsi que sa gestion, afficher la liste des produits disponibles, commander et payer ces produits. Ce projet est réalisé avec le framework Symfony, ainsi que l’API Stripe pour réaliser les paiement bancaire.

1. Cahier des charges, expression des besoins, ou spécifications fonctionnelles du projet
2. Présentation du projet

Le but de mon projet est de créer une boutique en ligne, pour cela je me suis appuyée sur une société existante pour obtenir les informations que contiendra mon site. Cette société ne souhaitant pas de boutique en ligne ne sont dont pas considéré comme client. Toutefois, je m’appuie sur eux pour avoir des retours vis-à-vis de l’interface et des données pour que ce soit plus concret.

1. Objectifs du projet

L’objectif de ce projet est de réaliser une boutique en ligne viable. Pour cela, j’ai établi un cahier des charges avec les fonctionnalités essentielles :

Le site web doit avoir une interface utilisateur où celui-ci peut :

* + S’inscrire, se connecter et se déconnecter
  + Obtenir la liste des produits, ainsi que ses caractéristiques
  + Accéder à un panier et pouvoir y ajouter les produits qu’il souhaite
  + Accéder à son profil avec ses informations, notamment son email, ses adresses, ses commandes effectués, où il pourra également modifier son adresse mail et son mot de passe
  + Avoir la possibilité de passer une commande et de la payer, et par la suite obtenir une facture en PDF

L’utilisateur n’a besoin de se connecter uniquement pour accéder à son profil et passer une commande.

Le site web peut avoir une partie administrateur où celui-ci peut :

* + Se connecter et se déconnecter
  + Ajouter, modifier et supprimer un produit
  + Obtenir l’historique de toutes les commandes effectuées
  + Gérer les commandes

L’administrateur a également accès à tout ce que l’utilisateur a accès.

De plus, le site possède une partie site vitrine concernant la présentation de la société.

Voici ci-dessous la partie non essentielle du cahier des charges, qui si elle n’est pas respectée peut être poussé en axe d’amélioration :

* + Un système de filtre des produits, par type, domaine ou appellation
  + Un système de tri des produits, par ordre alphabétique ou par prix
  + Un système de mail qui demande une vérification à l’inscription et envoie les factures ainsi que les confirmations de commande
  + Une version anglaise du site
  + Un système de disposition des produits (en colonne ou en tableau)

1. Spécifications techniques du projet, élaborées par le candidat, y compris pour la sécurité et le web mobile
2. Technologies utilisées

Php (HyperText Preprocessor): c’est un langage de script(interprété) open source, coté serveur. Il est généralement utilisé pour le développement de site/page web dynamique

HTML5 CSS3 :Hypertext Markup Language est un langage de balisage conçu pour représenter des pages web. Cascading Style Sheet ou les feuilles de style en cascade sont un langage informatique qui décrit la présentation des documents HTML et XML(eXtensible Markup Language : langage de balisage extensible, conçu pour structurer des fichiers)

Symfony : est un ensemble de composant PHP ainsi qu’un framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulable et adaptables qui permettent de faciliter et d’accélérer le développement d’un site web.

Twig : est un moteur de template souple, rapide et sécurisé pour le langage de programmation PHP, il est utilisé par défaut par le framework Symfony.

Doctrine : il s’agit d’un Object Relational Mapping (ORM : Technique de programmation faisant le lien entre la base de données et la programmation objet) il est utilisé par défaut par le framework Symfony.

Composer : est un logiciel de gestionnaire de dépendances libre écrit en PHP. Il permet de gérer les dépendances d’un projet.

Visual Studio Code : est un éditeur de code extensible développé par Microsoft pour Windows.

Adobe XD : est un outil de conception d’expérience utilisateur basé sur le vecteur pour les applications web et les applications mobiles.

Jmerise : c’est un logiciel dédié à la modélisation des modèles conceptuels de données, il permet la généralisation et la spécialisation des entités, la création des relations et des cardinalités ainsi que la généralisation des modèles logiques de données. (MLD) et des script SQL.

API Stripe est une solution très simple d’API bancaire pour effectuer des paiements en lignes.

Laragon est un environnement de développement web dédié au système d’exploitation Windows. Il est composé de différentes technologies : Apache (serveur web), PHP (langage interprété coté serveur), MySQL (base de données)

JavaScript c’est un langage de programmation de scripts, principalement employé dans les pages web interactives.

~~Jquery est une bibliothèque JavaScript libre et multiplateforme créer pour facilité l’écriture de scripts côté client.~~

github

1. MCD

Un modèle conceptuel de données est la représentation la plus abstraite des données d’un système d’information. Les données sont représentées sous forme d’entités et d’associations entre entités. Il permet de représenter les données d’un système d’information.

J’ai réalisé mon MDC à l’aide de Jmerise.

L’entité User contient l’essentiel l’email comme moyen d’authentification, un mot de passe, ainsi qu’un rôle. Elle est reliée à l’entité Ship\_address avec une relation qui permet qu’un utilisateur puissent avoir plus adresse de livraison mais qu’une adresse de livraison ne puisse appartenir qu’a un seul utilisateur. L’entité Ship\_address contient toutes les informations relatives à une adresse, elle est reliée à l’entité Order de façon à ce que l’adresse de livraison puisse correspondre avec plusieurs Order mais qu’un Order ne soit relié qu’a une seule adresse de livraison. L’entité Order contient une référence de commande, une date de création, un status qui correspondra à l’état de la commande (en cours de traitement, payé, annulé), elle est également reliée à l’utilisateur, un user peut avoir plusieurs commandes, mais une commande n’est faite que par un seul utilisateur.

L’entité Order est en relation avec l’entité facture qui elle contiendra sa propre référence et adresse de facturation, ainsi lorsque l’utilisateur souhaite supprimer ses données, je pourrais les supprimer sans problème en gardant uniquement le nécessaire qui sera inclut dans l’entité Facture. Cette entité possède également une propriété lienPDF qui sera générer lorsque l’utilisateur achèvera sa commande.

L’entité Order est en relation avec l’entité Product qui elle contient toutes les informations relatives à un produit : la référence, le nom, la description, le prix à l’unité, la quantité en stock, la disponibilité du produit et la photo. Cette relation est de type : 0,n ce qui inclut donc une table de jointure que j’appellerai ProductOrder, elle correspondra à l’association d’un ou plusieurs produits à une ou plusieurs commande. J’y inclus la quantité afin d’obtenir une entité que me permettra d’avoir une commande, un produit, et la quantité de ce produit.

L’entité Product est reliée à 3 entités qui sont des catégories différentes : Appellation, Type, Domain. Un produit ne possède qu’une appellation, qu’un type et qu’un domaine exemple : Corton Charlemagne étant le nom du produit, il a l’appellation Grand Cru, de type vin rouge et du domaine Chapuis.

*IMG du MCD à refaire*

MLD

Le modèle Logique de Données est la représentation des données d’un système d’information. Jmerise génère le MLD à partir du MCD, il met en évidence les relations existantes entres les entités. Par exemple il nous permet de mieux concevoir la relation entre Order & Product, la relation est bien devenue une table de jointure et correspond donc à une nouvelle entité. On peut également voir une relation de type 1,1 entre Order et Facture, en effet, une facture n’appartient qu’à une commande et une commande n’a qu’une facture.

*IMG du MLD*

1. Maquettes

Une maquette permet d’avoir un premier rendu d’un site web en indiquant l’emplacement des blocs afin de concevoir la structure d’un site. Elle permet également d’avoir une première idée de l’interface utilisateur (UI) et de l’expérience Utilisateur (UX). Elle est également nécessaire lorsque l’on travail avec un client pour qu’il est rapidement une idée du projet. Ici, j’ai choisi de réalisé mes maquettes avec Adobe XD, il s’agit d’un logiciel gratuit et complet qui permet de réaliser des maquettes pour tout type de formats, très simplement et rapidement.

J’ai d’abord réalisé les maquettes pour le format téléphone de mon site web. J’ai passé plusieurs heures à optimiser les maquettes afin que l’expérience utilisateur soit le meilleur possible. Ensuite, j’ai réalisé les maquettes pour la version desktop avec l’essentiel des changements.

*IMG Exemple de maquette*

Différence à la fin du projet

1. Symfony

J’ai choisi d’utiliser le framework Symfony pour réaliser mon projet afin de faciliter et donc de gagner du temps dans le développement de mon projet. Ce framework possède de nombreux avantages notamment la sécurité native à Symfony ainsi qu’une facilité d’utilisation grâce à sa documentation complète et le soutien de sa communauté.

Présentation du MVC

Symfony se présente comme framework possédant une architecture Modèle Vue Controller (MVC). Cela nous permet de séparer 3 points essentiels :

* Le Modèle qui correspond aux requêtes de la base de données
* Le Controller qui gère la logique relative aux traitements des demandes sert également à effectuer des vérifications et des autorisations
* La vue qui correspond à l’affiche

L’utilisateur interagit en envoyant une requête au controller, c’est le point d’entrée de l’application. Le controller demande les données au modèle qui traduit cette demande par une requête SQL, il récupère les données et les renvois au controller. Le controller transmet les données à la vue qui se charge d’afficher les données. La vue ici n’a aucune logique. Le point d’entrée de l’application

Cependant, Symfony est capable de créer n’importe quel type de framework. C’est pourquoi pour ce projet nous utiliserons plutôt l’architecture MVP qui découle de l’architecture MVC. Le P signifie ici Presenter qui correspond au controller. Dans cette architecture, l’utilisateur envoie sa requête à la vue qui fait elle-même appel au Presenter pour traiter la demande de l’utilisateur. Le Présenter fait appel au modèle qui traite la demande du controlleur et lui renvoies des données. Le controller retourne alors une vue avec les données. Autrement dit ici, la vue et le modèle ne sont pas directement en relation. Et le point d’entrée de l’application c’est la vue.

Gestion des utilisateurs

Symfony nous permet de créer rapidement un système de gestion des utilisateurs.

1. User

Grâce au MakerBundle, nous pouvons utiliser la commande « php/bin console make :user » qui nous permet de créer une entité User, elle possèdera les propriétés suivantes par défaut un identifiant unique tel que l’email, un mot de passe ainsi qu’un rôle. Lorsqu’une entité est créée avec la console, le repository de l’entité est également créer, elle correspond au modèle (de l’architecture MVP)

ainsi que le SecurityController qui contiendra deux méthodes, l’une pour s’authentifier et l’autre pour se déconnecter.

1. Registration

Gestion des utilisateurs

Sécurité Native

1. RGPD
2. EAsy Admin ??maybe
3. Réalisations du candidat comportant les extraits de code les plus significatifs et en les argumentant, y compris pour la sécurité et le web mobile
4. Présentation du jeu d’essai élaboré par le candidat de la fonctionnalité la plus représentative (données entrée, données attendues, données obtenus)
5. Description de la veille, effectuée par le candidat durant le projet, sur les vulnérabilités de sécurité
6. Description d’une situation de travail ayant nécessité une recherche, effectuée par le candidat durant le projet, à partir d’un site anglophone
7. Extrait du site anglophone, utilisé dans le cadre de la recherche décrite précédemment, accompagné de la traduction en français effectuée par le candidat sans traducteur automatique, environ 750 signes