1. **INFORMATIONS GÉNÉRALES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Candidats (2 ou 3):** | Nom: | **Rodrigues Fraga** | Prénom: | Brian |
|  | Nom: | **Baseia** | Prénom: | Alexandre |
|  | Nom : |  | Prénom : |  |
| **Lieu de travail:** | Ste-Croix (CPNV) | | | |
| **Chef de projet:** | Nom: | **Benzonana** | Prénom: | Pascal |
| : | | ☎: | |
| **Expert 1:** | Nom: | **Chavey** | Prénom: | Jean-Philippe |
| : | | ☎: | |
| **Expert 2:** | Nom: |  | Prénom: |  |
| : | | ☎: | |
| **Période de réalisation :** | 4ème trimestre – année 2017-2018  30.04.2018 au 22.06.2018 – 24h00 | | | |
| **Horaire de travail :** | Voir intranet | | | |
| **Nombre d'heures :** | 5h25 \* 8 \* 2 = 84 heures (les jours de congé sont comptés) | | | |
| **Planning (en H ou %)** | 10% Gestion de projet (séance, planification, coordination)  10% Analyse préliminaire  20% Conception  50% Implémentation  10% Réserve  Note :   * Le choix de la méthode de gestion du projet est à discuter entre les développeurs et le chef de projet * La rédaction de la documentation est incluse dans chaque position | | | |

1. **PROCÉDURE**

* Les candidats réalisent un travail en équipe sur la base d'un cahier des charges reçu le 1er jour.
* Le cahier des charges est approuvé par le CPNV. Il est en outre présenté, commenté et discuté avec les candidats. Par leurs signatures, les candidats acceptent le travail proposé.
* Les candidats ont connaissance de la feuille d’appréciation avant de débuter le travail.
* Les candidats sont entièrement responsables de la sécurité de ses données.
* En cas de problèmes graves, le candidat averti au plus vite les deux experts et son chef de projet.
* Les candidats ont la possibilité d’obtenir de l’aide mais doivent le mentionner dans leur dossier de projet.
* A la fin du délai imparti pour la réalisation du Projet Web + BDD, les candidats doivent livrer sur OneDrive le dossier de projet.

1. **TITRE**

GalaxSat : Les étoiles à portée de main.

1. **SUJET**

GalaxSat est une plateforme de vente de contrat et de CubeSats en ligne pour les entreprise et les instituts.

1. **MATÉRIEL ET LOGICIEL À DISPOSITION**

**Physique**

* Un PC tournant sous Win7
* Réseau intranet et internet du CPNV
* Un PC à la maison sous Win10 (et Linux)
* Internet

**Logiciel et Framework**

* PhpStorm (avec toutes les extensions possibles comme PHP, Git, PhpUnit)
* Atom.io (les packages de base + les packages de la communauté : atom-ide-ui, ide-php, php-server)
* MySQL Workbench pour la modélisation et l’aide à la création de script SQL
* Serveur MySQL pour héberger la base de données du site web
* Suite office 2016 de Microsoft pour la rédaction de la documentation, de graphiques
* Draw.io pour la réalisation des diagrammes UML (navigation, classes, flux)
* Dataeo (générateur de documentation technique pour la BDD)
* Doxygen (générateur de documentation technique pour le code)
* Bootstrap pour rapidement obtenir des composants graphiques responsifs, à moindre coût de développement
* JQuery pour simplifier JavaScript
* Balsamiq éditeur de Wireframe payant
* GanttProject pour la planification du projet

**Outils collaboratifs**

* GitHub (versioning, ticketing)
* Trello outil de gestion des tâches. L’éventuel apprentissage de cet outil ne fait pas partie des heures du projet.
* OneDrive pour le stockage du projet (Documentation, site web, etc…)
* moodle.cpnv.ch (information sur le projet et base de connaissances)

1. **PRÉREQUIS**

Les développeurs travaillant sur ce projet doivent avoir suivi les modules ICT relatifs aux bases de données ainsi qu’au développement web. Voici une liste de compétences requises :

* Expériences de PHP ainsi que d’un IDE similaire à PHP Storm
* Avoir déjà codé un projet selon le patron de conception MVC
* Connaissances du versioning
* Connaissances d’UML et de Microsoft Visio

**7a DESCRIPTIF DU PROJET**

**7.1 Généralités**

Ce module Prog Web + BDD a pour objectifs d'entraîner les étudiants à la réalisation de projet de type développement.

Le projet en question se veut aussi proche possible d'un TPI, avec les nuances suivantes :

* Le nombre d'heures allouées est inférieur (environ la moitié 50 heures vs 80 heures)
* Ce projet sera réalisé par groupe de 2 ou 3 personnes (le TPI est individuel)
* L’évaluation finale se fera sous la forme d'une livraison client (démonstration réelle du développement)
* La documentation conceptuelle sera réduite à son minimum (afin de favoriser la qualité)

**7.2 Fonctionnalités minimales**

Le type d’application web demandé doit offrir les fonctionnalités minimales que l’on rencontre actuellement sur les e-commerces récents, tels que :

* La création d’un compte utilisateur
* Différents rôles (administrateur du site, responsable des ventes, utilisateur, visiteur)
* Gestion des articles et du stock
* Intégration d’un panier
* Fonction de paiement
* Envoi d’email
* Connectivité avec au moins un réseau social (de votre choix)

**7.3 Graphisme**

L’interface graphique est la carte de visite de votre application. Les composants devront tous être “responsifs”. En d’autres termes, votre application doit pouvoir fonctionner aussi bien sur un PC, une tablette ou un smartphone. Toutefois, le temps de développement du graphisme doit être réduit au maximum car c’est la partie PHP-MySQL qui sera évaluée au niveau pratique. De ce fait, l’utilisation d’un Template déjà existant est fortement conseillée.

**7.4 Sécurité**

Votre application doit respecter les standards du marché concernant la sécurité.

Le client doit y accéder en https. Les mots de passe ne doivent ni être visibles sur l’interface graphique, ni stockés dans la base de données en clair.

Les versions modernes de PHP intègre de nombreuses fonctions permettant de se protéger, notamment contre les injections SQL. Utilisez les mécanismes de sécurité à disposition.

**7.5 Persistance des données**

En vous inspirant des modules que vous avez suivi en SQL, mettez en place une base de donnée selon la conception de votre choix.

Les principes de base comme la redondance, l’obsolescence et l’incohérence doivent être particulièrement soignés.

**7b POINTS ÉVALUÉS DURANT LE PROJET**

* Le processus « Analyse préliminaire – Conception – Implémentation »
* Le soin des livrables (documentation, schéma et code)
* Régularité dans l’effort fourni par toute l’équipe de développeurs

1. **LIVRABLES**

**Récurrents (via GitHub)**

* 1 fois par semaine le **journal de travail** (vendredi 17:00 au plus tard)

Le journal de travail doit permettre de reproduire le cheminement réflexif vécu par les développeurs lors de la réalisation du projet.

Un journal de travail doit permettre de :

* Identifier le temps consommé et le temps restant, selon la répartition mentionnée au point 1 de ce document
* Situer les événements particuliers comme des changements de stratégies, des soucis techniques, des séances lors du projet
* Observer les actions et tâches réalisées par les développeurs tout au long du projet
* 1 fois par semaine le **rapport de projet** (vendredi à 17:00 au plus tard)

Le rapport de projet peut-être inspiré du canevas qui est livré (Moodle). Ce document a pour objectif de documenter le processus d’analyse, de conception et d’implémentation.

Rappel

Le rapport de projet met en valeur toutes les compétences d’informaticien en CFC sur un projet de type développement. Chaque composant implémenté, doit être présent dans la conception et identifié dans l’analyse.

La documentation technique autogénérée et le manuel d’utilisation seront en annexe de la documentation.

* Le **code source** est à jour après chaque fin de session

Aide : Une solution sur GitHub permet d’avoir la garantie que tout le monde travaille sur la même version du code. N’hésitez pas à publier (push) régulièrement. La qualité des messages accompagnant les “commits” sera évaluée comme un aspect de la documentation et de la communication.

**Ponctuels**

* 30.04.2018
  + - Les sujets et les groupes sont connus
    - Les outils de développements (PhpStorm, Github, Word, Excel) sont mis en places par les développeurs et accessibles aux chefs de projet
* 25.05.2018
  + - Le design du site est défini et implémenté
    - Les fonctions de login et création de compte sont fonctionnels
    - Les MCD et MLD initiaux sont « scriptés » et la base de données est atteignable depuis le backend

**//TO DO – l’équipe de développeur doit définir les livrables entre le 30.04 et le 22.06**

* 14.05.2018
  + - Une vision sur la phase d’analyse et le début de la phase de conception sera livrée au client pour pouvoir suivre et avoir une première vision de comment ressemblera le site.

Livraison : les experts peuvent voir l’avancement du début de projet.

* 21.05.2018
  + - La phase de conception est terminée en plus de la phase d’analyse. La phase de réalisation débutera après avoir envoyé ce livrable au client.

Livraison : les experts peuvent voir sous quel forme le site et le projet devra ressembler.

* 28.05.2018
  + - Une première version de la plateforme est livrée au client.

Livraison : les experts doivent pouvoir s’inscrire et se connecter à la plateforme, sans l’aide des développeurs.

* 04.06.2018
  + - Une deuxième version de la plateforme est livrée au client

Livraison : les experts doivent pouvoir ajouter, modifier et supprimer des articles.

* 11.06.2018
  + - Une troisième version de la plateforme est livrée au client

Livraison : les experts doivent pouvoir gérer le panier et accéder au paiement.

* 18.06.2018
  + - Une dernière version de la plateforme et livrée au client

Livraison : les experts doivent pouvoir gérer le stock des articles et recevoir des mails.

* SEM COM
  + - Démonstration de l’outil fonctionnel face à 2 experts. L’application web doit être livré sur l’environnement du client (et non sur le poste du développeur).
    - Livraison : le client reçoit le projet au complet (le site, la documentation, etc…)

1. **POINTS TECHNIQUES ÉVALUÉS SPÉCIFIQUES AU PROJET**

* L’architecture MVC a été implémentée et est respectée pour toutes les fonctionnalités du site

Il est tout à fait envisageable d’utiliser un MVC différent de celui vu en cours. Les chefs de projets conseillent néanmoins de ne pas prendre un Framework trop complexe.

* Le panier est totalement fonctionnel
  1. Ajout d’article
  2. Suppression d’article
  3. Vider le panier lors de l’achat

Note : La persistance du panier n’est pas exigée.

* La documentation est à jour par rapport au produit livré

Rapport de travail

Doit être complet et parfaitement aligné sur le produit livré.

Documentation technique (code PHP et SQL)

Doit être généré par des outils de générations de documentation automatique. Il est exigé qu’une fonctionnalité soit décrite en détail, de l’interface graphique jusqu’à la base de données. Cette fonctionnalité doit être spécifique à la plateforme. Les actions de login, création de compte ne seront pas acceptés.

Manuel d’utilisation

Le manuel d’utilisation doit permettre à un utilisateur lambda d’effectuer les actions minimales sans soucis.

1. **VALIDATION**

|  |  |  |
| --- | --- | --- |
|  | Lu et approuvé le : | Signature : |
| Candidats : |  |  |
|  |  |  |
|  |  |  |
| Expert n°1 : |  |  |
| Expert n° 2 : |  |  |
| Chef de projet : |  |  |