

SAÉ 3.01 : Développement d'une application Rapport du projet

FI2-B
BUI Brandon
CHARTIER Hugo
CHARTON Samuel
LAMPIN Vivien
MERABET Yanis

1.A Objectifs du projet

Le Projet se nomme SAE 1.03 Développement d'application, le projet s'articule en la réalisation d'une application web proposant des modules de simulation de calcul (l'utilisateur inscrira des données qui seront calculées) pour différents domaines :

- informatique (exemple : conversions binaires hexa, octal et décimal)
- mathématiques (exemple : calcul d'une probabilité en loi Normale)
- autre domaine (exemple : calcul d'amortissement)
- sécurité (exemple : mise en place d'un algorithme de cryptographie)

L'application gèrera des visiteurs, utilisateurs inscrits et possèdera un gestionnaire, qui fait office d'administrateur.

Le visiteur n'a accès qu'à la page d'accueil qui lui permet de voir une explication (vidéo et/ou texte) de l'application et de pouvoir s'inscrire/se connecter afin de pouvoir accéder au système de simulation de calcul.

L'identifiant d'un utilisateur est son email et son mot de passe sera composé d'au moins 8 caractères étant uniquement des caractères de la norme ASCII. L'utilisateur doit aussi renseigner son nom et prénom au moment de l'inscription.

Il devra également remplir un captcha pour valider l'inscription.

Un utilisateur peut changer son mot de passe. Il ne peut toutefois pas le récupérer s'il l'oublie.

Le gestionnaire peut gérer les utilisateurs (voir les identifiants des utilisateurs inscrits mais pas leurs mots de passe, en supprimer un) et a accès à ses modules utilisés.

Il peut aussi établir une statistique des visites et des modules utilisés par les utilisateurs inscrits sur la plateforme (exemples : module le plus utilisé, nombre de visites dans le mois...).

Le gestionnaire ne peut pas accéder aux modules pour faire des simulations de calcul, il doit se connecter avec un autre identifiant en tant qu'utilisateur standard.

Il y a aussi un fichier de log qui se met à jour à chaque connexion ratée. Il contient l'identifiant, le mot de passe tenté, l'adresse IP, et la date de la tentative de connexion.

L'application sera hébergée sur un Raspberry Pi 4, système d'exploitation Linux de la distribution Debian et aura un système de gestion de base de données dédié.

Elle possèdera aussi une charte graphique, un logo et un nom.

1.B Méthode utilisée et calendrier de travail

Nous avons travaillé en méthode agile qui consiste à rester en contact avec le client (Mr Huguin dans notre cas) en utilisant notamment les forums de la plateforme eCampus pour rester en contact avec lui. Nous étions en contact régulier avec lui surtout au début du projet notamment pour voir si les maquettes du site web ainsi que ce que nous avons marqué dans les différents documents à rendre correspondait à ses besoins.

De plus, la méthode agile que nous avons utilisé s'appelle le Scrum. Avec cette méthode, nous avons séparé notre travail par petites périodes de quelques semaines (1 semaine au début du projet puis 3 semaines une fois le développement vraiment commencé) appelées "sprints". A la fin de chaque sprint, nous sommes censés rendre une partie du projet au client que l'on appelle livrable.

Il était donc nécessaire de faire une planification des sprints à envoyer à Mr Huguin pour qu'il puisse voir ce que l'on allait rendre à chaque sprint et à quelle date. Nous avons donc fait un fichier PDF avec les dates des sprints ainsi qu'une brève description de ceux-ci que nous lui avons envoyé.

Voici la planification que nous avons mise en place :

Planification des sprints :

Sprint 7 :

Ce sprint se déroulera de la semaine du 31 octobre à la semaine du 14 novembre. Les pages de connexion et d'inscription seront effectués ainsi que leur liaison à la base de données.

Sprint 8 :

Ce sprint se déroulera de la semaine du 21 novembre à la semaine du 5 décembre. Le 1^{er} module sera effectué. Celui-ci sera consacré aux mathématiques.

Sprint 9 :

Ce sprint se déroulera de la semaine du 12 décembre à la semaine du 26 décembre. Le 2^e module sera effectué. Il sera consacré à la cryptographie.

Sprint 10 :

Ce sprint se déroulera de la semaine du 2 janvier au 16 janvier. Le 3^e module sera effectué. Toutefois, nous ne savons pas encore lequel sera le module. De plus, le serveur sera sécurisé (potentiellement avec un accès https en certificat X509) avec un rapport d'explication.

1.C Résultats obtenus

Côté modules :

Une page de modules, listant tous les modules proposés par l'application. Les utilisateurs pourront choisir celui qu'ils désirent utiliser en cliquant dessus. Ils seront redirigés vers une page dédiée à ce dernier, tout en pouvant revenir en arrière si ils le souhaitent.

1. Module de probabilité

La loi normale a été conçue pour modéliser certains faits de la réalité, démontrant quelques événements de la vie pouvant être vus par des nombres où nous pouvons calculer leurs probabilités de se réaliser. Ici, l'utilisateur entrera un jeu de données dans trois formulaires. Le module calculera la probabilité $P(x < t)$ selon les valeurs entrées par ce dernier. Pour finir, il aura le choix de choisir entre trois méthodes de calcul, chacune renvoyant des valeurs approximatives de la probabilité.

2. Module de cryptographie

De nos jours, le cryptage de données est un enjeu majeur pour garder de manière sécurisée les données. C'est pourquoi ce module est à la disposition des utilisateurs. Ils entreront leurs données, le programme lui renverra les données chiffrées tout en gardant la clef secrète. Quand le moment viendra, ils pourront renvoyer ces données chiffrées pour récupérer les données originales.

Côté site :

1. Connexion/Inscription

Avant de pouvoir bénéficier des différents modules proposés par notre application, les utilisateurs devront se connecter. Pour se faire, une page dédiée à la connexion est mise en place. Côté client : ils entreront leur identifiant et mot de passe. Côté serveur : des requêtes dans la base de données des utilisateurs seront effectuées, afin d'accepter ou non la demande d'accès des utilisateurs à l'application. Cependant, si de nouveaux utilisateurs arrivent, ils devront créer un compte. Un lien permettant d'accéder à la page d'inscription est à leur disposition. Là, les utilisateurs entreront leurs données (nom, prénom, mail, mot de passe) et pourront accéder à la page principale du site afin de profiter des modules. Le serveur quant à lui se charge de mettre à jour sa base de données pour ajouter ses nouveaux utilisateurs.

2. Conditions d'utilisation

Les utilisateurs voulant utiliser notre application devront accepter nos conditions d'utilisation. Cela permet de les informer de leurs engagements. Ils pourront accéder à cette page quand ils le souhaitent.

1.D Bilan

1. Difficultés rencontrées et solutions trouvées :

Choix du langage du module de probabilité. Beaucoup de possibilités s'offraient à nous, tel que le R, Python, Java, JavaScript ou encore php. Chacun avait ses avantages : la simplicité de python qui n'est pas un langage typé. Le php permettait de faire directement le code du module sur la page, simplifiant ainsi la récolte et l'envoi des données. Quant à Java, nous étions très familiarisés grâce à la dernière saé de notre première année.

Le choix des données stockées dans notre base de données. Une des solutions était de stocker le nom et l'identifiant dans une table. Une autre était d'associer une autre table pour des éventuelles données des utilisateurs, « liées » entre elles grâce à l'identifiant. Cela pourrait être possible si on souhaitait stocker d'autres données moins importantes.

Problème de connexion sur la base de données. Il nous est arrivé que la base de données, pour une raison que nous ignorons encore, nous empêche de s'y connecter.

Donner la possibilité aux utilisateurs de modifier leurs identifiant et mot de passe. Mais avec réflexion on avait compris que ça demanderait beaucoup trop d'investissement, ce qui pourrait entraîner un retard sur le rendu du projet.

Pour réaliser la connexion à notre base de données, on a dû s'inspirer de nos cours de programmation web. Notre professeur nous a enseigné une méthode pour se connecter à une base de données depuis php. On a donc créé notre base de données, puis nous nous sommes connecté depuis php. Pour enfin préparer des requêtes SQL lorsque les utilisateurs voudront se connecter ou s'inscrire.

Mauvaise organisation, ainsi qu'une communication approximative. Les tâches attribuées n'étaient pas explicites, ce qui nous a causé du tort et a mené à un décalage du rendu, nous pénalisant. Cependant elle a entraîné une entraide pour notre camarade en difficulté, c'est tout le principe de notre projet.

L'utilisation d'un langage qui n'est pas au programme nous a demandé une étude plus approfondie. Son utilisation demande un serveur, ce qui a poussé un membre à s'investir davantage, afin de comprendre l'hébergement des données, l'envoi et la réception des données client/serveur.

2. Les ressources mobilisées

- Développement web, pour le site web de manière générale
- SQL et programmation, pour la base de données
- Probabilités, pour le module de probabilités
- Cryptographie et sécurité, pour le module de cryptographie
- Droit des contrats et du numérique, pour la partie juridique du site
e.g RGPD
- Qualité de développement, pour l'organisation du projet et l'architecture du site
- Anglais, pour la présentation du projet en anglais

3. Les apports en termes de compétences

Nous pensons que ce projet ne concerne pas les compétences du BUT 1 et uniquement celles du BUT 2, ce qui paraît logique car celles de première année sont acquises, donc non réévaluées et le projet se déroule en deuxième année, et donc seules celles-ci seront traitées :

- UE3.1 - Partir des exigences et aller jusqu'à une application complète :

Il nous est demandé de créer une application web à partir d'exigences vagues d'un client, nous avons donc dû demander des précisions, convenir avec lui d'un cahier des charges que les deux parties prenantes du projet comprennent.

- UE3.2 - Sélectionner les algorithmes adéquats pour répondre à un problème donné :

Nous avons pris beaucoup trop de temps avant de se décider sur un langage dans lequel développer les modules à cause des différences propres à chaque langage et à nos connaissances avec celui-ci.

Le client nous a proposé des formules mathématiques pour le module de probabilité et un algorithme pour celui de cryptographie, mais c'était à nous de traduire cela en lignes de code, de choisir une bonne implémentation, qui serait simple à expliquer et à comprendre pour d'éventuelles modifications dans le futur, sans pour autant sortir des choses proposées par le client.

- UE3.3 - Déployer des services dans une architecture réseau :

Nous étions fourni directement une machine relativement pré-configurée (système d'exploitation déjà installé, connectée au réseau, avec quelques choses en plus comme un serveur ssh) pour héberger notre site web, n'avions pas à le faire nous-mêmes. Nous avons eu à installer des services pour faire fonctionner notre site web cependant : un serveur web pour pouvoir traiter des demandes, une base de données pour stocker nos données...

- UE3.4 - Optimiser une base de données, interagir avec une application et mettre en œuvre la sécurité :

Nous avons besoin d'une base de données pour stocker les données personnelles des utilisateurs du site et donc implique logiquement une sécurisation de cela : nous chiffons les données stockées dedans de manière à ce que même si quelqu'un arrive à y accéder ça ne va pas lui servir à grand-chose. Naturellement, notre site web fait des requêtes à cette base de données lors de l'inscription ou de la connexion d'un utilisateur.

- UE3.5 - Appliquer une démarche de suivi de projet en fonction des besoins métiers des clients et des utilisateurs :

- UE3.6 - Situer son rôle et ses missions au sein d'une équipe informatique :

Le rôle de chaque membre du projet a été relativement clair depuis le départ, où chacun a été assigné à la tâche à laquelle il excelle le mieux, même s'il y a eu des modifications en cours de route.

2.A Charte graphique

- Couleur de fond : blanc
- Couleur des cadres : gris foncé
- Couleur du texte : blanc
- Police de caractère :
- Taille de la police :
- Couleurs des boutons : couleurs similaires à celles du logo
- Couleur des champs de saisie : noir

Nous avons décidé la mise en place d'un fond blanc car il représente le principe d'une feuille de papier. En effet, les logiciels proposant des services de bloc-notes tel que Word utilisent cette même page blanche. Ce standard de fond permet d'indiquer inconsciemment aux utilisateurs qu'une zone de texte sera présentée, et l'aide à se concentrer sur l'essentiel en minimisant les distractions visuelles. Pour donner de la valeur à notre contenu, on a décidé de placer une barre de navigation et les formulaires de la même couleur (gris foncé). Ainsi, on utilise cette couleur pour tout élément où l'utilisateur pourra interagir avec. Ce mélange entre blanc et noir permet de donner un aspect moderne à notre site, sans oublier que ce mélange n'est pas agressif aux yeux.

Comme l'utilisation du blanc et noir est limitée, on a décidé de reprendre les couleurs du logo pour les boutons. Car en effet, notre logo représente avant tout des boutons et il est de notre intérêt de l'exploiter pour favoriser la familiarité qu'a l'utilisateur sur notre application. Par ce fait, les boutons sont mis en valeur et se démarquent des composants comme les formulaires de saisie. Et comme on y vient, les champs de saisie : des composants permettant de recevoir des chaînes de caractères par l'utilisateur qui seront envoyés ultérieurement. Ils font partie du formulaire, il est donc préférable d'utiliser une couleur qui puisse se démarquer sans pour autant trop le mettre en valeur pour ne pas perdre le regard de l'utilisateur. Car des textes explicatifs, permettant d'expliquer quelles données entrer dans le formulaire, sont présents.

2.B Présentation des deux logos



Ce premier logo est un logo de type abstrait qui représente des modules qui appartiennent à plusieurs domaines (on peut le voir à la couleur qui change entre certains carrés), enfin ces carrés sont sur un fond blanc pouvant rappeler une page web.



Modulia

Ce deuxième logo est le même que le précédent à l'exception que du texte est en plus présent pour appuyer le nom de l'application, les couleurs du texte rappelant les couleurs du logo abstrait.

Le logo finalement choisi était le deuxième, car il rappelle le nom de l'application Modulia et était le plus complet des deux.