



Université du Québec
à Chicoutimi

GÉNIE INFORMATIQUE

APPLICATIONS RÉSEAUX ET SÉCURITÉ INFORMATIQUE

6GEI466

Projet de conception Application de suivi des mises à jour des locaux et horaires de cours à l'UQAC

Préparé par :

Alissa BONNEL

Jean-Sébastien ST-PIERRE

Alexis VALOTAIRE

GÉNIE INFORMATIQUE

Pour :

Monsieur Jean-Luc CYR, ing.

UNIVERSITÉ DU QUÉBEC À CHICOUTIMI

6 novembre 2019

Historique des versions

Version	Date	Auteur(s)	Modifications
1.0	04-11-2019	AB, JSSP, AV	Création du document

Définitions

Terme	Définition
<i>Framework</i>	Ensemble cohérent de composants éprouvés et réutilisables ainsi que de préconisations servant à créer les fondations et les grandes lignes d'un logiciel ou de certains de ses composants [1].
<i>Front-end</i>	Partie du développement applicatif web visant à implémenter les mécanismes d'interaction avec l'utilisateur, habituellement sous forme d'interfaces graphiques. Le Front-End s'exécute le plus souvent sur un ordinateur client et communique avec un serveur traitant les requêtes et les données [2].
<i>Back-end</i>	Partie du développement applicatif web visant à implémenter les mécanismes de traitement, de logique et d'accès aux données, qui est habituellement exécutée sur un serveur. Le Front-End et le Back-End d'une même application communiquent entre eux au moyen d'un protocole de communication réseau [2].
<i>Cloud server</i>	Terme désignant une architecture faisant appel à une infrastructure infonuagique où un serveur virtuel, dont les ressources peuvent être allouées dynamiquement par le fournisseur de services ("pay-as-you-go"), est utilisé en lieu et place d'un serveur physique disposant de ressources statiques [3].
<i>Web scraping</i>	Terme désignant un ensemble de techniques permettant l'extraction rapide de grandes quantités de données rendues disponibles par un site web, le plus souvent dans le code HTML. [4].

Abréviations et acronymes

Abréviation/acronyme	Définition
SPA	Single page application
BCAPG	Bureau canadien d'agrément des programmes de génie

Table des matières

1	Introduction	3
1.1	Vue d'ensemble du projet	3
1.2	Références	3
1.3	Portée	3
1.4	Livrables	3
2	Organisation du projet	4
2.1	Composition de l'équipe du projet	4
2.2	Stratégie de contrôle des versions	4
2.3	Directives de livraison	4
3	Ressources matérielles, calendrier et budget	5
3.1	Ressources matérielles	5
3.2	Calendrier des tâches du projet	5
3.3	Budget	5
4	Gestion des risques	6
4.1	Risque : Injection SQL	6

1 Introduction

1.1 Vue d'ensemble du projet

Le projet consiste au développement d'une application web permettant à l'utilisateur d'accéder aux locaux et horaires de cours rapidement et facilement. Dans le contexte quelque peu frénétique et, plus récemment, cahotique des débuts de trimestres à l'UQAC, il est parfois difficile pour les étudiants et enseignants de demeurer à l'affût des inévitables changements logistiques qui peuvent être apportés par l'administration sans aucun préavis. L'application qui sera réalisée dans le cadre de ce projet vise à apporter une solution simple à cette problématique récurrente en permettant une détection automatique des changements, qui pourront être notifiés à l'utilisateur selon ses préférences.

Concrètement, l'utilisateur pourra se créer un compte et y associer ses cours. Une extraction (*scraping*) sera effectuée périodiquement sur le contenu des pages web correspondantes du site institutionnel de l'UQAC dans le but de détecter tout changement apporté aux horaires et numéros de locaux. L'utilisateur aura la possibilité de s'authentifier, puis de consulter son horaire (incluant les numéros des locaux). Il pourra également choisir d'être notifié en temps réel de tout changement par le moyen de son choix (courriel, SMS, notification de l'application, etc.).

1.2 Références

Les références utilisées dans le cadre de ce rapport peuvent être consultées dans la bibliographie.

1.3 Portée

L'application sera utilisable par toute personne impliquée dans les activités de formation de l'UQAC (étudiants, professeurs, chargés de cours, etc.) L'application ne vise en aucun temps à se substituer au portail officiel de l'institution ; elle se veut plutôt une forme d'extension de l'offre existante.

Le bon fonctionnement de l'application sera dépendant de l'exactitude des informations publiées sur le site officiel de l'UQAC. Vu le recours à la technique du *scraping*, des correctifs devront lui être apportés suite à tout changement dans la structure des documents web de l'institution. En ce sens, les concepteurs se dégagent de toute responsabilité quant à la véracité des informations relayées par l'application.

1.4 Livrables

Les concepteurs s'engagent à livrer une application web conforme aux spécifications logicielles. Lors de la complétion du projet, la documentation sur le fonctionnement ainsi que les instructions de déploiement seront fournies au promoteur, accompagnées du code source.

2 Organisation du projet

2.1 Composition de l'équipe du projet

Notre équipe se compose de trois étudiants en génie informatique à leur dernière année du baccalauréat :

Alexis Valotaire

Compétences/Préférences :

- Maîtrise du Python
- Connaissance en HTML, CSS, JavaScript
- Connaissances en design web
- Connaissance en configuration et maintenance de serveurs
- Connaissance en gestion de base de données
- Connaissance en réseautique
- Initié en industrie aux bases de l'industrie 4.0, IoT, cloud computing, etc

Jean-Sébastien St-Pierre

Compétences/Préférences :

- Habilités avec les langages C/C++
- Connaissances en Python, Javascript, VB, C#, HTML, CSS et SQL
- Connaissances en gestion de bases de données relationnelles et autres
- Connaissances en conception de bases de données
- Connaissances en design d'applications et d'infrastructures
- Connaissances en réseautique
- Initié en industrie aux bases de l'industrie 4.0, IoT, cloud computing, etc.

Alissa Bonnel

Compétences/Préférences :

- Maîtrise des langages C/C++, Python
- Connaissances en Web ; PHP, Javascript, HTML, CSS, Classic ASP
- Connaissances générales en bases de données ; MySQL, SQLServer

2.2 Stratégie de contrôle des versions

Le contrôle des versions se fera avec le système de gestion de version qu'est GitHub autant pour la documentation que pour le code du projet.

2.3 Directives de livraison

La livraison au promoteur de l'application et des documents connexes aura lieu en personne, au moment fixé par le plan de cours. Une présentation avec démonstration et période de questions permettra de valider la satisfaction du promoteur à l'égard des livrables soumis.

3 Ressources matérielles, calendrier et budget

3.1 Ressources matérielles

Aucune ressources matérielle particulière ne sera nécessaire puisque le développement se fera à partir des équipements appartenant aux concepteurs et/ou à l'UQAC, le cas échéant. Par contre, un serveur infonuagique (*Cloud server*) serait idéal afin d'héberger l'application, particulièrement en cours de développement. Ce choix présente l'avantage d'être accessible par les développeurs en tout temps et en tous lieux, sans imposer de confinement à l'intérieur de l'environnement informatique de l'Université.

3.2 Calendrier des tâches du projet

Le projet a officiellement commencé dans la semaine du 23 octobre et devrait s'échelonner sur 8 semaines. La liste qui suit montre l'allure générale du déroulement du projet.

- Semaine 1 : Discuter en équipe pour déterminer le sujet du projet ;
- Semaine 2 : Rédiger un rapport de conception de projet ;
- Semaine 3 : Déterminer les tâches de chacun des membres de l'équipe et faire des recherches individuelles ;
- Semaine 4 : Travail individuel et en équipe pour faire progresser les tâches de chacun ;
- Semaine 5 : Travail individuel et en équipe pour faire progresser les tâches de chacun ;
- Semaine 6 : Travail individuel et en équipe. Mise en forme du projet final.
- Semaine 7 : Effectuer plusieurs tests et apporter les correctifs nécessaires ;
- Semaine 8 : Conclure et remettre le projet.

À noter que les différentes tâches pour chacun des membres seront à déterminer. Celles-ci comprennent la mise en place du serveur, programmer la communication entre le côté serveur et le côté client, récupérer les données sur le site de l'UQAC, gérer la base de données et les comptes des utilisateurs, etc.

3.3 Budget

Pour ce qui est du budget, n'ayant aucun matériel physique requis, rien est à prévoir de ce côté. Un montant peut être prévu pour le serveur Cloud. Les prix de ceux disponibles sont à vérifier. Étant un projet dans le cadre d'un cours universitaire, aucun honoraire ne sera facturé pour le temps consacré par les membres de l'équipe.

4 Gestion des risques

4.1 Risque : Injection SQL

Description

Une injection SQL peut se produire lorsqu'un utilisateur est appelé à entrer des données quelconques utilisées par une application dans l'exécution d'une requête SQL. Lorsque cette vulnérabilité n'est pas gérée convenablement, un attaquant pourrait tenter d'inclure des fragments de code SQL dans les champs de saisie et éventuellement parvenir à faire exécuter une requête au contenu malicieux.

Dans le cas où une injection SQL est possible, un utilisateur malveillant pourrait facilement effacer toutes les tables de la base de données utilisée par l'application. Ceci entraînerait le non fonctionnement de l'application puisque les utilisateurs ne pourraient plus se connecter en raison du fait qu'ils n'existeraient tout simplement plus dans la table utilisée pour l'authentification.

Probabilité

3/5

Conséquence

4/5

Exposition

5/5

Méthodes de contention

La solution afin de contenir ce risque est de bien nettoyer toutes données en provenance des utilisateurs, par exemple à l'aide d'expressions régulières exécutées du côté serveur. Il est aussi impératif d'ajouter des validations côté client à l'aide des types d'*input* des *form*. Or, par exemple, si dans une entrée allant directement dans une requête SQL un utilisateur entrait un apostrophe comme dernier caractère, les validations devraient supprimer cette apostrophe avant d'exécuter la requête, sinon une erreur surviendrait.

Références

- [1] Formation Django. (s. d.) Qu'est-ce qu'un framework ?. [En ligne]. Disponible : <http://www.formation-django.fr/generalites/framework.html>
- [2] CONCEPTANEXT. (2019) What Is the Difference Between Front-End and Back-End Development?. [En ligne]. Disponible : <https://conceptainc.com/blog/difference-front-end-back-end-development/>
- [3] Cloudflare. (s. d.) What is serverless computing?. [En ligne]. Disponible : <https://www.cloudflare.com/learning/serverless/what-is-serverless>
- [4] Kho, J. (2018) How to Web Scrape with Python in 4 Minutes. [En ligne]. Disponible : <https://towardsdatascience.com/how-to-web-scrape-with-python-in-4-minutes-bc49186a8460>