Gestion de projet

*420-0SH-SW*

Présentation formelle

*Contrôle d’environnement avec module Arduino*

Présenté à :

*Marco Guilemette*

*France Jean*

Présenté par :

*Dave Grenier*

*Tommy Landry*

*Jade Phaneuf*

Remis le :

*5 février 2020*

Table des matières

[Proposition 3](#_Toc31699202)

[But général 3](#_Toc31699203)

[Lien avec la réalité 3](#_Toc31699204)

[Résultat attendu 3](#_Toc31699205)

[*But ultime* 3](#_Toc31699206)

[*But au symposium* 4](#_Toc31699207)

[Auditoire visé 4](#_Toc31699208)

[Ressources nécessaires 4](#_Toc31699209)

[Grandes étapes du projet et répartition des heures 4](#_Toc31699210)

[Outils nécessaires 5](#_Toc31699211)

[Formations nécessaires : 6](#_Toc31699212)

[Exigences particulières 6](#_Toc31699213)

## Proposition

Nous proposons un projet de gestion d’environnement contrôlé réalisé à une échelle domestique (opposée à l’échelle commerciale) où il possible d’obtenir des données en temps réels dans un environnement que l’on souhaite contrôler (serre, terrarium, etc.) et d’automatiser une partie de la gestion de cet environnement à l’aide des données obtenues, comme par exemple, gestion de la température, activation de la ventilation, brumisation, etc.

## But général

Création d’un système de gestion d’environnement contrôlé.

## Lien avec la réalité

La gestion d’un environnement défini est sources de problèmes pour plusieurs. Par exemple, dans nos maisons, les thermostats gèrent les sources de chauffage en fonction de la température souhaitée. Cependant, lorsque vient le temps de créer un environnement propice à d’autres espèces, par exemple des plantes tropicales ou des reptiles dans un terrarium, cela peut devenir plus complexes et nécessiter davantage de surveillance.

Des solutions pouvant prendre en compte un seul facteur existent; par exemple, une minuterie sur une lumière, un thermostat sur un élément chauffant. Cependant, ces éléments individuels ne considèrent pas l’environnement à contrôler comme un tout. Par exemple, une ventilation est souvent nécessaire après une période de brumisation pour éviter les maladies fongiques ou une baisse de température peut être géré par un démarrage de la ventilation, mais en tenant en compte de la baisse d’humidité relative de l’air subséquente.

Des systèmes sont déjà présents sur le marché pouvant réaliser ce type de gestion complexes ou pouvant être programmés selon les besoins mais la plupart sont coûteux, volumineux et/ou pour des besoins à grande échelle. Le but est donc de concevoir un prototype accessible, relativement peu coûteux et portable pouvant être utilisé dans un petit environnement.

## Résultat attendu

## *But ultime*

Créer un système de gestion d’environnement contrôlé afin de gérer autant la température, l’humidité, voire même l’arrosage de plantes à l’aide d’un module Arduino. Les données recueillies seront ensuite entreposées dans une base de données qui sera accessible sur une application de type web, permettant d’avoir accès à un historique des données.

Des fonctions de gestion d’environnement seront programmées afin de gérer les paramètres selon les données reçues. Par exemple, le démarrage de la brumisation si l’humidité relative est trop basse pendant un temps donné, suivi d’une période du départ de la ventilation et une mise en veilleuse de la fonction de brumisation pendant un certain temps pendant que les conditions se stabilisent.

L’application quant à elle recevra les notifications en temps réelles des besoins de la serre ainsi qu’un historique afin de pouvoir par exemple préprogrammer quand arroser ou non.

## *But au symposium*

Démontrer la gestion d’environnement contrôlé à l’aide d’un prototype simulant une serre ou un terrarium. Les fonctions visées sont similaires à celles de la section précédente, mais dans le cas où des contraintes de budget, temps ou de faisabilité (électricité, etc.) viendrait limiter le projet, l’ébauche du prototype vise à remplacer les périphériques (éclairage, pompe et buses de brumisation, ventilateurs, etc.) par des ampoules de types LED et des croquis des pièces d’équipement afin de démontrer la logique et la séquence d’activation selon les conditions présentes dans l’environnement.

## Auditoire visé

Outre les amateurs de plantes tropicales et de reptiles, l’auditoire visé se composent aussi des amateurs de sciences en herbe, de curieux et de gens intéressés à voir les possibilités que peut offrir la connaissances et la maîtrise de notions de programmations en informatique et en électroniques.

## Ressources nécessaires

* Site de référence Arduino (<https://www.arduino.cc/en/Guide/HomePage>);
* Site de référence W3S (<https://www.w3schools.com/>);
* Formations web DYMA (<https://dyma.fr/>);
* Formation Udemy (<https://www.udemy.com/>);
* Formations OpenClassRoom (<https://openclassrooms.com/fr/>);
* Personnes de référence en électronique : M. Nicolas Bourré et M. Stevens Gagnon;

## Grandes étapes du projet et répartition des heures

La répartition des 270 heures du projet est détaillée dans le tableau 1.

Au niveau des notions théoriques, une bonne partie des heures (135 h) seront utilisées au sein du cours de Gestion de projet alors qu’environ 10 h de formation seront utilisées pour les volets *front end* (HTML, javascript), back end (base de données) et acquisition de données (Arduino), totalisant 162 h pour ce volet (Tableau 1).

Pour sa part, le volet technique du projet obtiendra près de 90 h pour la programmation et le montage électronique vers un prototype fonctionnel (Tableau 1).

Pour terminer, la rédaction d’un rapport et le montage du kiosque pour le salon des Sciences devraient accaparer environs 20 h de travail. Le temps de présentation au Salon des Sciences n’est cependant pas comptabilisé (Tableau 1).

**Tableau 1. Répartition des heures pour la réalisation d'un projet d'environnement contrôlé en fonction le volet et types de tâches.**

|  |  |
| --- | --- |
| **Tâches** | **Nb d'heures estimées (h)** |
| **Apprentissage et théorie** | |
| Formation en classe (3h/semaine) | 135 |
| Formation/Approfondissement *front end* | 7 |
| Formation/Approfondissement *back end* | 10 |
| Formation/Approfondissement Arduino/acquisition de données | 10 |
| *Sous-Total Théorie* | *162* |
| **Réalisation et technique** | |
| Réalisation *front end* | 25 |
| Réalisation *back end* | 29 |
| Réalisation montage | 5 |
| Réalisation programmation modules | 29 |
| *Sous-Total Technique* | *88* |
| **Communication et rapport** | |
| Rédaction du rapport | 15 |
| Création prototype et kiosque salon des Sciences | 5 |
| *Sous-Total Communication* | *20* |
| **Total** | **270** |

## Outils nécessaires

Matériel logiciel

* Arduino IDE et librairies associées ;
* Visual Studio Code;
* Logiciel de gestion de base de données (à déterminer);
* Suite Microsoft Office ;
* Langages de programmations appropriés (ex : html, javascript, etc.)

Matériel physique

* Matériel électronique (ex : fils, breadboard, résistances, etc.)
* Carte NodeMCU Lua 12E;
* Soudeur
* Relais
* Modules de conversion I2C logic
* Source d’alimentation 5V;
* Sondes température/humidité DHT22;
* Autre sonde si jugé nécessaire (humidité du sol, etc);
* Ventilateurs d’ordinateur;
* Lumières LED;
* Symposium : aquarium ou contenant (prototype), plante.

## Formations nécessaires :

* Arduino et langage approprié pour le développement de notre matériel et logiciel (possiblement du C/C++), utilisation de librairies;
* Formation web : bootstrap, php, JQuery, JS, HTML5, CSS, NodeJS (révision et approfondissement de la matière);
* Formation BDD : utilisation du php pour envoyer les données vers la base de données, choix d’une version de logiciel de base de données pertinent pour nos besoins;

## Exigences particulières

Pour le symposium : Avoir accès à une prise de courant, extension et écran;