|  |
| --- |
| Projet tutoré dans le cadre de la première année du master TI en collaboration avec CapGemini |
| Projet Tutoré |
| [Sous-titre du document] |

|  |
| --- |
| Cazalis-von Tolkacz-Joret  [Date] |

Couverture du rapport :

Logo UPPA / Logo CapGemini

Auteurs

Rapport de projets tutorés « intitulé »

Noms des responsables

Date

Remerciements

Table des matières

Introduction

Problématique

Descri travail effectué

Descri apport étudiants (oral ou conclusion)

Conclusion

Bibliographie

Annexes ?

Remerciements :

Nous tenons à remercier nos encadrants d’avoir proposé ce sujet très intéressant et de nous avoir prêté les capteurs.

Table des matières :

Introduction :

Durant notre second semestre de master technologie de l’internet à l’université de Pau et des pays de l’Adour, nous avons été amenés à réaliser un projet tutoré dans le cadre de notre module de gestion et réalisation de projet.

Ce projet tutoré devait être réalisé en groupe, le nôtre était composé de 3 personnes :

Pauline Cazalis

Eddy Joret

Karol von Tolkacz

Notre projet a été proposé et encadré par Fabien Lamas, Nicolas Dubois, Mickaël Lefèvre et Guillaume Fauvet des ingénieurs logiciel de l’entreprise CapGemini à Pau.

Capgemini est une entreprise de service et de numérique française qui est défini par ses métiers en quatre grandes catégories :

Le conseil et transformation à travers Capgemini Invent

Les services applicatifs

Les services de technologies et d’ingénierie

Les services restant d’infogérance

Cette entreprise a remodelé son organisation en de grandes unités opérationnelles nommées Strategic Business Units (ou SBU) qui sont composées de Business Unit (BU) et eux-mêmes qui sont un regroupement de plusieurs Market Units (MU) sectorielles :

Biens de consommation, commerces et distributions

Energie, Utilities et chimie

Services financiers

Industrie, Automobile et Science de la vie

Secteur public

Télécommunications, Médias et Divertissement

Il a pour but de mettre en place tout une plateforme ayant pour objectif la réduction de la consommation électrique, ainsi que l’amélioration de la qualité de vie de la ville de Pau.

Travail demandé

Voici l’intitulé du travail demandé par nos encadrants : *Nous vous proposons de reprendre le sujet « Ville Connectée » mis en place lors du TER 2020. La base du système IOT ainsi qu’une IHM sous NodeRed seront vos fondations pour aller encore plus loin ! Avec la motivation d’étendre notre système à une ville, et non plus une simple agence connectée, il vous faudra mettre en place un système dynamique d’ajout et de suppression d’objets connectés, ajout et suppression de capteurs sur un arduino déjà en place en interrompant le système le moins possible ainsi qu’une utilisation astucieuse et innovante des différents capteurs à l’échelle d’une ville.  
La rigueur et de l’imagination seront les maîtres mots du projet, nous vous accompagneront dans les choix techniques et architecturaux que vous ferez dans un cadre Agile (kanban, événements agiles) en vous offrant tous les outils nécessaire pour réussir ce projet ambitieux.*

A partir de cette base d’informations nous avons donc commencé la mise en place de notre plateforme en respectant tout le long la problématique suivante : Qu’elles changements et solutions pouvons-nous apporter pour améliorer la qualité de vie de Pau ?

Afin de structurer notre rapport écrit, nous allons dans un premier temps faire une présentation globale de notre projet, nous ferons ensuite la description des capteurs et logiciels utilisés et enfin nous présenterons de façon détaillé le travail réalisé.

1. Présentation globale du projet (via une problématique et un l’état de l’art)

La problématique

Pour répondre à la problématique nous avons pour objectif de mettre en place une « ville connectée » permettant l’amélioration de la qualité de vie de la ville via la surveillance et le traitement de données de différents secteurs d’activités.

Une « Ville Connectée » ou « Ville intelligente » est un type de ville ou l’utilisation de capteurs permet la récupération de données avec pour objectif l’amélioration de la qualité des services urbains et la réduction de leurs couts.

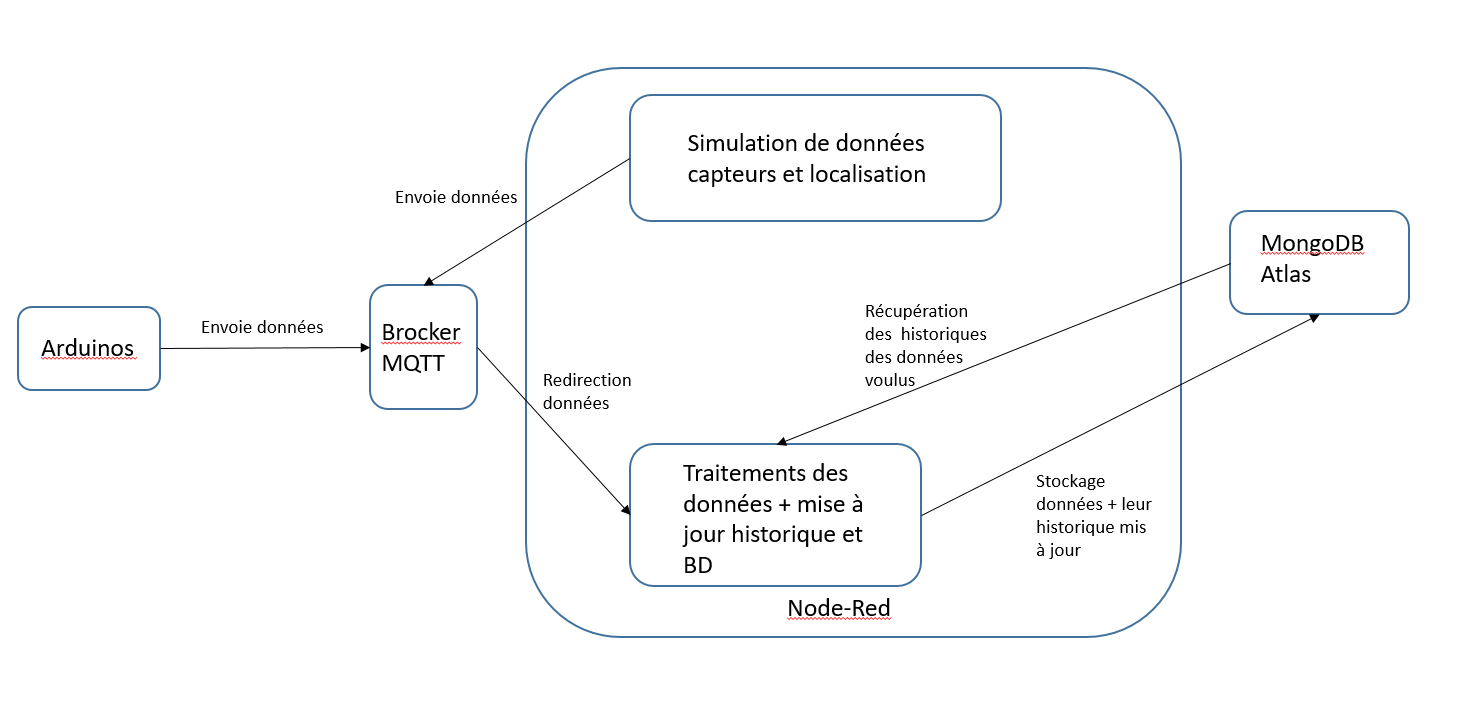
La réponse

Pour cela nous avons repris comme base le projet proposé l’année dernière qui consistait en la création d’un bâtiment d’une agence, équipé d’une solution de surveillance de données, se voulant une prémisse a l’élaboration d’une solution à l’échelle d’une ville.

Celle-ci nous a permit de construire la base de notre projet tout en apportant des changements et des améliorations

Ainsi, pour pouvoir extraire et interpréter les données voulus nous utilisons des Arduino, différents capteurs et logiciels servant à récupérer, traiter et stocker ces données qui seront présentés dans la partie suivante.

Parler du rapport de l’année dernière (lien github)



1. Présentation des capteurs, logiciels et environnement de travail utilisés

Dans cette seconde partie, nous allons

Concernant le choix des solutions utilisées, en partant de la base imposée par nos encadrants, nous avons également utiliser d’autres solutions diverses et variées pour répondre au mieux aux besoins de notre projet.

Pour traiter notre projet il nous fallait donc partir d’une base simple, laquelle était un système IOT et une IHM basée sur NodeRed, et permettre à travers l’utilisation de tout un ensemble de capteurs de créer un système moderne de gestion de la ville.

Arduino

Nous disposons d’une carte Arduino NodeMCU ESP8266 de la marque Velleman, avec son cable d’alimentation USB/Micro-USB.

C’est une carte avec circuit intégré a un microcontrôleur avec connexion Wi-Fi, que l’on utilisera en langage C++ avec l’IDE Arduino, composé de 10 broches GPIO (General Purpose Input/Output). Chaque GPIO peut être PWM (Pulse Width Modulation), I2C (Inter-Integrated Circuit) ou encore 1-Wire (adressage et utilisation de plusieurs capteurs à communication série sur une seule et même broche numérique).

Elle est aussi composée de deux boutons poussoirs rst (reset, redémarrage de la carte) et flash (permet de lancer une séquence de flasharge de la mémoire) et d’une antenne PCP qui permet d’améliorer la portée.



Une carte Arduino nous permet de recevoir des informations lancées par un ou plusieurs capteurs et pour transmettre ces informations vers un logiciel externe.

L’Arduino ESP8266 étant équipée d’un module wifi, cela nous permettra d’envoyer les données reçus sous un format type via une connexion wifi, bien plus pratique qu’une connexion filière a l’échelle d’une ville.

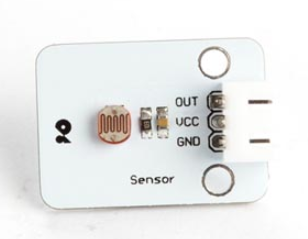
Nous disposons aussi de 21 câbles de connexion de l’Arduino aux capteurs.

Capteurs

Trois capteurs nous ont étaient fournis, 2 capteurs permettant de récupérer des informations concernant la température et l’humidité, et un capteur photosensible.

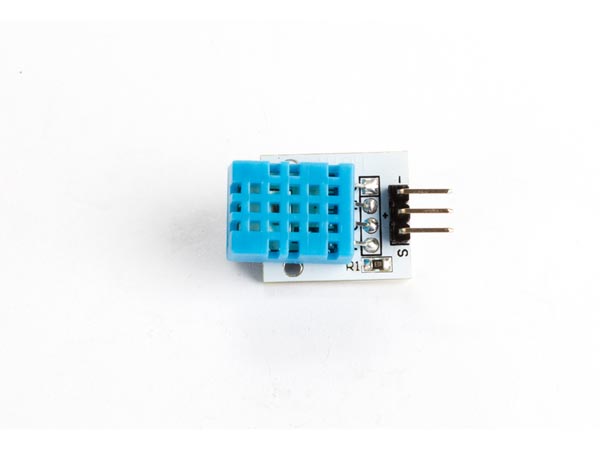
Nous disposons de trois capteurs de la marque Velleman :

1x capteur photosensible qui utilise une photorésistance pour produire une tension de sortie.



Ce capteur nous servira pour connaitre la luminosité du ciel de la ville.

2x capteurs de température (NTC) et d’humidité (DHT11).



Ce capteur nous permet de récupérer la température et l’humidité présente dans l’air.

Le reste des capteurs ont étaient simulés.

Logiciels et environnement de travail

Pour développer et déployer notre plateforme nous avons utilisés plusieurs logiciels que nous allons présenter en suivant.

Arduino IDE

Logiciel de codage de l’interface de notre Arduino.

Il s’agit de la plateforme qui va se connecter au broker MQTT

Broker MQTT

Le protocole de messagerie type publish-subscribe par topics qui nous permet de faire le lien entre notre Arduino et la plateforme Node-Red.

NodeRed

L’application reposant sur NodeJS et qui nous permet de récupérer les données reçues via MQTT, pour les restructurer de manière a pouvoir les stocker dans la base de donnée cloud MongoDB Atlas.

+ uniquement utilisé en tant que simulateur

MongoDB Atlas

Basée sur le serveur NoSQL MongoDB, Atlas est un service de cloud de base de données.

NodeJS

Angular

Docker

Visual Studio Code

Il s’agit de l’IDE que nous avons décidé d’adopter pour développer notre projet tutoré.

Il s’agit d’un des éditeurs de code les plus populaires, développé par Microsoft. De part sa légèreté et le grand nombre d’extensions qu’il propose il nous a semblé être l’IDE le plus adapté à nos besoins.

GitHub

Pour le stockage et le partage de notre code nous avons utilisé GitHub.

Trello

Enfin nous avons utilisé la plateforme Trello, un outil de gestion de projet, à partir d’une base mise en place par nos encadrants nous avons au fil des semaines remplis notre Trello en précisant l’avancement de notre projet et puis tout ce qu’il nous restait à faire.

Organisé en tableaux, auxquels on ajoute des taches, l’outil est simple à prendre en main et permet un suivi clair et précis du projet.

1. Présentation du travail réalisé

Présentation des solutions pour répondre a notre problématique (idées et mise en œuvres)

Justification des choix (est-ce vraiment dans la partie état de l’art ?)

(Ressource utilisées, choix retenus)

Notre concept de ville connectée, consisté en la mise en place d’une solution basé sur plusieurs critères :

La gestion des déchets

La gestion de l’éclairage public

Mise en place des différentes technologies

1.1 Nous partons donc de capteurs qui a terme seront positionnés dans l’ensemble de la ville de Pau, par exemple les premiers sont des capteurs de pression qui seraient placés sur l’ensemble des poubelles de la ville et permettraient de récupérer la valeur de pression de la poubelle.

Celle-ci sera transmise via le broker MQTT au serveur Web qui affichera ou non le besoin de vider la poubelle en question.

Puis détailler l’aspect technique :

Pour la mise en place de cette solution, nous avons utilisé un capteur de pression qui envoie la pression actuelle de la poubelle a notre Arduino.

Celui-ci ayant le code téléversé nécessaire a la récupération et a l’envoie de la valeur fournie par le capteur. En premier lieu le code va connecter l’Arduino au réseau wifi, puis le connecter au broker MQTT pour ainsi envoyer la valeur introduite dans un message au format JSON, et envoyer ce message au broker. (suite a compléter allant réception du msg jusqu’au serveur NodeJS)

Angular va chercher à récupérer les données stocker dans NodeJS pour pouvoir les afficher. Pour cela, NodeJS va stocker les données (récup ou ?) dans le but de les envoyer via une socket.

Le serveur ouvre une connexion socket (protocole réseau permettant la création d’un canal de communication à double sens entre un serveur et un navigateur) et recoit la demande de connexion d’Angular via web socket et lui enverra ensuite les données.

Angular pourra ainsi manipuler et afficher les données reçues dans le component lié a l’affichage de la page web.

1.2 Après avoir géré nos déchets, nous avons mis en place une solution de gestion de l’éclairage public via un capteur de luminosité.

L’idée est de

Planning et évolution au fil des semaines

Compte tenu de la situation actuelle ils nous étaient impossible de nous retrouver en physique pour travailler le projet tutoré, pour se préparer a ce travail nous avons fonctionné de la façon suivante : Nous fixions des rendez-vous hebdomadaires souvent le dimanche, à l’issu de ces rendez vous sur Discord chaque membres du trinôme pouvait expliquer et présenter ses avancées mais également les problèmes qu’il avait rencontrés.

En plus de cette réunion entre nous, chaque semaine (le mardi à 17h10), nous nous réunissions sur Discord avec tous les membres du projet et les encadrants pour débriefer et présenter l’avancée du projet.

A la fin de l’appel, un programme était mis en place par les encadrants pour nous guider sur le travail a fournir la semaine suivante.

Difficultés techniques rencontrées

Conclusion

Ce projet a été un succès, en effet nous avons réussis à développer la solution attendue.

A travers de nombreuses recherches nous avons réussis à mettre en place une solution correct et validée par nos encadrants.

+Apports personnels concernant ce projet tutoré

Bibliographie