**Conception d’un système de monitoring et de maintenance**

# INTRODUCTION

Au cours de ma formation, j'ai eu l'opportunité de réaliser un stage de **3 mois** au sein de **DigiPlus Consulting**, une entreprise spécialisée dans le développement de solutions informatiques pour différents secteurs d'activités. Mon stage a été axé sur le développement d'un système de monitoring et de prédiction des incendies dans une usine grâce à des capteurs, un microcontrôleur, la plateforme Microsoft Azure avec les services tels que IoT Hub, Stream Analytics, PostgreSQL, et en y intégrant le machine learning pour la prédiction et la conception d'un dashboard avec la librairie Dash de Python.

L'objectif principal de ce projet était de concevoir un système capable de surveiller en temps réel les conditions environnementales dans une usine et de prédire les risques d'incendie. Le projet a été réalisé en collaboration avec une équipe de développeurs expérimentés, qui m'ont guidé tout au long du processus de développement et m'ont permis d'acquérir de solides compétences en matière de développement de solutions IoT.

Dans ce rapport de stage, je vais présenter en détail les différentes étapes de développement du projet, ainsi que les outils et technologies utilisés pour le réaliser. Je vais également analyser les résultats obtenus et discuter des perspectives de ce projet.

# PARTIE 1 : PRESENTATION DE L’ENTREPRISE ET DEROULEMENT DU STAGE

## CHAPITRE 1 : PRESENTATION DE L’ENTREPRISE

### GENESE

DIGIPLUS SARL a été créé le 17 Aout 2016 par Monsieur CHOUAMO NJO Bertin, directeur général de ladite structure depuis la création. Au départ avec un effectif de 5. DIGIPLUS compte maintenant 4 employés et six (06) partenaires et de nos jours ont réalisé deux projets.

### STRUCTURE DE DIGIPLUS

#### Structure fonctionnelle

#### Structure hiérarchique

#### Adresses de l’entreprise

#### Localisation

Consulting SARL est une société situe dans la ville de DOUALA, plus précisément à Bonamoussadi dans la rue situe derrière l’hôtel de France, à l’étage d’un bâtiment R+1.

### CONTEXTE

DIGIPLUS SARL est une société exerçant dans la proposition des solutions informatique pour le quotidien, firme dynamique qui livre ses produits en respectant des spécifications de base, elle se charge de la création d’une identité numérique des entreprises (création des sites web).

#### Partenaires

* NETINAFRICA ;
* MICROSOFT PARTNER ;
* ITIL V3 CERTIFIED ;
* TMFORUM.

#### Concurrents

#### Fournisseurs

* CAMTEL : fournit une connexion à internet
* ENEO : fournit en énergie électrique
* CAMWATER : fournit de l’eau
* MICROSOFT : Les services offices 365 et d’hébergement
* MTN : fournit en connexion internet

#### Clients

* PROMODIS
* Le grand public avec ses différents produits

#### Différents services

* **Les services digitaux :** elle crée une identité de votre entreprise a l’aide des ressources telles qu’un site web, des plateformes de médias sociaux, un annuaire d’entreprise et d’autre sources en ligne
* **Sécurité et réseaux :** Nous créons, administrons et assurons la sécurité de votre réseau virtuel accessible uniquement via internet.
* **Développement et intégration :** elle dispose d’une équipe expérimentée et dévoué de développeurs qui s’occupent de la conception, du développement et de la mise en production des projets dans les plus brefs délais fixés par le client
* **It support :** Nous assistons nos clients au quotidien afin d’assurer la disponibilité de leurs ressources (applications, ressources matérielles).
* **Services cloud :** en tant que partenaire de Microsoft, nous facilitons la procédure de souscription aux services cloud propose par Microsoft. Nous nous occupons de l’achat des licences, de la configuration et l’administration de vos comptes et vous consommez juste vos services.

#### Produits

* **DIGISHOP**
* **DIGIXPRESS**

#### Perspectives avenir

* **DIGIROOM**

#### **Conclusion**

Après avoir parcouru les coins et recoins de cette entreprise futuriste qu’est Digiplus Consulting, nous pouvons ainsi dire qu’elle a tout le nécessaire pour survivre encore très longtemps dans ce monde brutalement concurrentiel qu’est l’entrepreneuriat.

## CHAPITRE 2 : DEROULEMENT DU STAGE

### Accueil

### Objectif du stage

D’un point de vue purement pédagogique, le stage académique a pour but de permettre à l’étudiant de voir les applications pratique et quotidiennes des connaissances théoriques vues pendant les cours en salle de classe et de mieux préparer son insertion dans la vie professionnelle.

Ainsi nos objectifs durant le stage à **DIGIPLUS** étaient donc déjà prédéfinis par nos pédagogues. Nous pouvons lister ici ses objectifs.

* Apprendre à rédiger un rapport.
* Comprendre les réalités du terrain.
* Se familiarisé avec les équipements industriels.
* Comprendre la structure hiérarchique d’une entreprise.

Mais ne pouvant pas se limiter à ces objectifs, nous nous sommes imposés les notre qui était les suivants :

* Apprendre une nouvelle chose chaque jour
* Arriver au travail avant 8h

### Synthèse de déroulement du stage

#### Conditions de travail

#### Difficultés rencontrées

### Regard critique sur DIGIPLUS

#### Problèmes observés

#### Solutions proposées

#### Conclusion

Cette première partie principalement centrée sur **DIGIPLUS CONSULTING** nous a permis de mieux la cerner et d’avoir un aperçu de notre future vie (la vie professionnel), ainsi nous pouvons retenir de cette belle aventure en tant que stagiaire, que le monde académique dans lequel nous sommes actuellement et très doux et ne nous prépare pas psychologiquement au monde impitoyable qui nous attend à la fin de notre formation. Donc il est nécessaire et même obligatoire pour chaque étudiant de faire le maximum de stage possible pendant sa formation.

# PARTIE 2 : PROJET DE STAGE

## CHAPITRE 1 : ETUDE DE L’EXISTANT

### INTRODUCTION

L'IoT, ou Internet des objets (en anglais, Internet of Things), fait référence à la connectivité des objets du quotidien à Internet et à leur capacité à échanger des données entre eux. Cela permet aux objets physiques, tels que les appareils domestiques, les véhicules, les capteurs, les montres intelligentes, les réfrigérateurs, etc., de collecter, d'analyser et de partager des informations.

L'IoT repose sur l'utilisation de capteurs, de dispositifs embarqués et de technologies de communication sans fil pour permettre aux objets de se connecter et d'interagir avec le monde numérique. Les données collectées par ces objets peuvent être analysées pour obtenir des informations utiles, automatiser des processus, améliorer l'efficacité, offrir de nouveaux services et améliorer la qualité de vie.

### GENERALITES

#### Evolution de l’internet des objets

Depuis ses débuts dans les années 1990, l'Internet des objets (IoT) a connu une évolution remarquable. Grâce aux progrès technologiques, tels que la miniaturisation des capteurs, les avancées des communications sans fil et l'amélioration de la gestion de l'énergie, l'IoT a pu se développer et se diversifier dans de nombreux secteurs. Les applications de l'IoT se sont étendues à la domotique, à la santé, à l'industrie, à l'agriculture, aux transports et bien plus encore. Les objets connectés sont devenus plus répandus, permettant la collecte et l'analyse de données en temps réel, l'automatisation des processus, l'amélioration de l'efficacité et la création de nouveaux services. Cependant, l'IoT continue de faire face à des défis tels que la sécurité des données, la gestion des normes et l'interopérabilité. Malgré cela, l'IoT reste une technologie en constante évolution qui promet de transformer de nombreux aspects de notre vie quotidienne et de nos industries, offrant de nouvelles opportunités et innovations à mesure que le temps passe.

Dans la suite de notre devoir, nous allons nous consacrer certains aspects de l’IoT qui sont la collecte, le stockage et l’affichage des données prises dans un environnement.

#### Fonctionnement d’un module de collecte et d’affichage des données

Dans le cadre de l'IoT, un module de collecte et d'affichage des données permet de récupérer les informations captées par un capteur dans un environnement spécifique, puis de les afficher de manière lisible pour l'utilisateur. Voici un aperçu simplifié de son fonctionnement :

* Capteur et collecte des données : Le capteur est chargé de mesurer et de détecter des informations spécifiques dans l'environnement, telles que la température, l'humidité, la luminosité, la pression, etc. Le capteur convertit ces informations en signaux électriques exploitables.
* Connectivité et transmission des données : Le module de collecte dispose d'une connectivité réseau, généralement sans fil, pour transmettre les données collectées. Il peut utiliser des technologies telles que le MQTT, le HTTP, le HTTPS ou d'autres protocoles de communication adaptés à l'application.
* Plateforme IoT et stockage des données : Les données captées par le module de collecte sont généralement envoyées à une plateforme IoT dédiée, qui peut être hébergée dans le cloud ou sur un serveur local. La plateforme IoT stocke et gère les données de manière sécurisée, prête à être traitée et analysée.
* Analyse et traitement des données : Les données collectées peuvent être analysées en temps réel ou de manière périodique pour en extraire des informations pertinentes. Cela peut inclure des analyses statistiques, des algorithmes de détection de tendances, des seuils d'alerte, etc. L'objectif est d'obtenir des informations utiles à partir des données collectées.
* Affichage des données : Les résultats de l'analyse sont ensuite présentés à l'utilisateur de manière compréhensible et conviviale. Cela peut être sous la forme de tableaux de bord, de graphiques, de rapports ou d'autres moyens visuels, en fonction des besoins de l'application.

Il convient de noter que la mise en œuvre exacte d'un module de collecte et d'affichage des données peut varier en fonction des spécificités de l'application, de la connectivité utilisée et des plateformes IoT sélectionnées. Les étapes décrites ici représentent une vue générale du fonctionnement de base d'un tel système dans le contexte de l'IoT.

### Exemple d’une plateforme mettant en œuvre ce système : EnergyCAP

EnergyCAP est une plateforme basée sur le cloud qui permet aux utilisateurs de surveiller, d'analyser leur consommation d'énergie en temps réel et qui fonctionne sous forme d’application web.

En tant qu'application web, EnergyCAP offre une interface conviviale accessible via un navigateur web. Voici quelques fonctionnalités clés offertes par EnergyCAP :

#### Tableau de bord personnalisable

Les utilisateurs peuvent créer leur propre tableau de bord en sélectionnant les graphiques, les widgets et les informations qu'ils souhaitent afficher en priorité. Ils peuvent personnaliser leur vue pour visualiser les données de consommation d'énergie les plus pertinentes pour eux.

#### Visualisation des données :

EnergyCAP propose une variété de graphiques interactifs pour représenter les données de consommation d'énergie, tels que des graphiques en barres, en lignes, en secteurs, des diagrammes de Sankey, etc. Ces graphiques permettent de visualiser les tendances, les variations et les relations entre les différents paramètres de consommation.

#### Suivi en temps réel

Les utilisateurs peuvent accéder aux données de consommation d'énergie en temps réel, ce qui leur permet de surveiller leur consommation à tout moment et d'identifier les comportements inhabituels ou les dépassements de seuils.

#### Rapports automatisés

EnergyCAP permet de générer des rapports automatisés selon des fréquences prédéfinies (quotidiennes, hebdomadaires, mensuelles, etc.) ou sur demande. Ces rapports peuvent inclure des données de consommation, des analyses de performance, des comparaisons, des recommandations, etc.

#### Gestion des factures et des budgets

EnergyCAP offre des fonctionnalités de gestion des factures d'énergie, ce qui permet aux utilisateurs d'importer, de valider et de suivre les données de facturation. Ils peuvent également définir des budgets énergétiques et surveiller les dépenses par rapport à ces budgets.

#### Intégration des données

EnergyCAP permet l'intégration avec différents systèmes de gestion des données, tels que les compteurs intelligents, les systèmes de surveillance du bâtiment (BMS), les systèmes de contrôle d'accès, etc. Cela permet une collecte automatique des données et une vue consolidée de la consommation d'énergie.

En utilisant EnergyCAP en tant qu'application web, les utilisateurs peuvent accéder à leurs données de consommation d'énergie de n'importe où, à tout moment, via une connexion Internet. Cela facilite la surveillance, l'analyse et la gestion de la consommation d'énergie, tout en permettant une collaboration efficace entre les parties prenantes impliquées dans la gestion de l'énergie d'une organisation.

#### Les manquements observés sur la plateforme EnergyCap

Bien qu'EnergyCAP soit une plateforme populaire et largement utilisée pour le suivi de la consommation d'énergie, il présente également certains manquements potentiels. Voici quelques-uns des points à prendre en compte :

* Complexité initiale : La configuration et la mise en place initiales d'EnergyCAP peuvent être complexes, en particulier pour les utilisateurs moins expérimentés. La création de tableaux de bord personnalisés et la configuration des différentes fonctionnalités nécessitent une certaine expertise technique, ce qui peut être un défi pour les utilisateurs novices.
* Courbe d'apprentissage : En raison de la richesse des fonctionnalités offertes par EnergyCAP, les utilisateurs peuvent avoir besoin d'une période d'apprentissage pour maîtriser toutes les possibilités offertes par la plateforme. L'exploitation maximale de ses fonctionnalités peut nécessiter une formation et une familiarisation approfondies.
* Limitations de personnalisation : Bien qu'EnergyCAP permette une personnalisation des tableaux de bord, des rapports et des visualisations, il peut y avoir certaines limitations quant à la personnalisation avancée. Certains utilisateurs peuvent souhaiter des options de personnalisation plus poussées pour répondre à leurs besoins spécifiques.
* Dépendance aux données de facturation : EnergyCAP dépend des données de facturation pour générer des rapports et des analyses. Si les données de facturation sont inexactes, manquantes ou tardives, cela peut affecter la précision des résultats fournis par la plateforme.
* Coût : L'utilisation d'EnergyCAP peut impliquer des coûts associés, notamment des frais de licence, d'abonnement et de support technique. Pour certaines organisations, en particulier les petites entreprises ou les particuliers, le coût de l'utilisation de cette plateforme peut constituer un obstacle.
* Limitations de l'intégration des données : Bien qu'EnergyCAP offre des fonctionnalités d'intégration avec d'autres systèmes, il peut y avoir des limitations en termes de compatibilité et de flexibilité avec certains systèmes spécifiques. L'intégration avec des dispositifs de collecte de données tiers peut nécessiter des efforts supplémentaires de configuration et de développement.

Il est important de noter que ces manquements potentiels peuvent varier en fonction des besoins et des exigences spécifiques de chaque utilisateur. Avant de choisir une plateforme de suivi de la consommation d'énergie, il est recommandé d'évaluer attentivement ces facteurs et de s'assurer que la solution correspond aux besoins spécifiques de l'organisation.

## CHAPITRE 2 : PRESENTATION DU PROJET DE STAGE

### DESCRIPTION DU PROJET

Le projet consiste en la mise en place d'un système complet de collecte et de visualisation des données à partir de capteurs. Grâce à l'utilisation de technologies telles que l'ESP32, MongoDB, PostgreSQL, Flask et le protocole HTTP, ce système permettra de surveiller et d'analyser différentes mesures environnementales en temps réel.

Les capteurs sélectionnés sont spécifiquement choisis pour répondre aux besoins de collecte de données, qu'il s'agisse de mesurer la température, l'humidité, la pression ou d'autres paramètres pertinents. L'ESP32, une carte de développement polyvalente et dotée de la connectivité Wi-Fi, joue un rôle essentiel en permettant la collecte des données à partir des capteurs.

Les données collectées par les capteurs sont stockées de manière efficace et évolutive grâce à l'utilisation de MongoDB, une base de données NoSQL flexible. Cette solution permet de gérer facilement de grandes quantités de données, garantissant ainsi la scalabilité du système.

Parallèlement, PostgreSQL, une base de données relationnelle robuste, est utilisée pour stocker les données des utilisateurs. Cette base de données est conçue pour fournir une structure organisée et fiable pour gérer les informations personnelles et les préférences des utilisateurs.

L'application web est développée en utilisant le framework Flask, qui offre une approche légère et flexible pour la création d'applications web. Cette application web permet aux utilisateurs de visualiser facilement les données collectées à partir des capteurs, en leur fournissant des graphiques et des tableaux clairs et intuitifs. Des fonctionnalités avancées d'analyse et de génération de rapports peuvent également être intégrées selon les besoins spécifiques du projet.

Le système utilise le protocole HTTP pour établir une communication fiable entre l'ESP32 et le serveur. Grâce à cela, les données collectées par les capteurs peuvent être envoyées au serveur de manière sécurisée et efficace, garantissant une transmission fluide des informations.

La sécurité est une préoccupation majeure de ce projet. Des mesures telles que l'authentification des utilisateurs, le chiffrement des données et la protection contre les attaques par injection sont mises en place pour garantir l'intégrité et la confidentialité des données collectées.

En résumé, ce projet offre un système complet et évolutif de collecte et de visualisation des données à partir de capteurs. Grâce à l'utilisation de l'ESP32, de MongoDB, de PostgreSQL, de Flask et du protocole HTTP, les utilisateurs peuvent surveiller et analyser facilement les données environnementales en temps réel, tout en garantissant la sécurité des informations collectées.

### OBJECTIFS DU PROJET

Dans le cadre du projet de collecte et de visualisation des données à partir de capteurs, voici quelques objectifs clés :

* Collecte des données : L'objectif est de configurer et de mettre en œuvre les capteurs appropriés pour collecter les données environnementales souhaitées de manière précise et fiable. Les capteurs doivent être en mesure de fournir des mesures précises des paramètres tels que la température, l'humidité, la pression, etc.
* Stockage des données : L'objectif est de mettre en place une infrastructure de stockage des données efficace et évolutif. Les données collectées par les capteurs doivent être stockées de manière structurée et accessible pour permettre leur utilisation ultérieure. Cela implique l'utilisation de bases de données comme MongoDB pour stocker les données collectées par les capteurs et PostgreSQL pour stocker les informations des utilisateurs.
* Visualisation des données : L'objectif est de développer une application web conviviale pour permettre aux utilisateurs de visualiser les données collectées. L'application doit fournir des graphiques, des tableaux et des représentations visuelles claires pour une compréhension facile des données. Elle doit également permettre aux utilisateurs de personnaliser les vues et d'explorer les données selon leurs besoins spécifiques.
* Communication sécurisée : L'objectif est de mettre en place une communication sécurisée entre les capteurs, l'ESP32 et le serveur. Les données collectées par les capteurs doivent être transmises de manière sécurisée pour éviter toute altération ou interception indésirable. Des mesures de sécurité telles que l'authentification et le chiffrement des données doivent être mises en place pour garantir l'intégrité et la confidentialité des informations
* Intégration du protocole HTTP : L'objectif est d'utiliser le protocole HTTP pour permettre la communication entre les différents composants du système, notamment l'ESP32, le serveur et l'application web. Le protocole HTTP doit être utilisé de manière efficace pour transmettre les données collectées du périphérique aux bases de données et à l'application web.
* Sécurité des données des utilisateurs : L'objectif est de garantir la sécurité des informations des utilisateurs stockées dans la base de données PostgreSQL. Cela inclut des mesures telles que l'authentification des utilisateurs, la protection contre les attaques par injection SQL et le chiffrement des données sensibles.
* Analyse des données : L'objectif est de permettre l'analyse des données collectées pour obtenir des informations pertinentes. Cela peut inclure l'utilisation d'algorithmes ou de techniques d'analyse de données pour détecter des tendances, des anomalies ou des modèles spécifiques dans les données collectées
* Évolutivité du système : L'objectif est de concevoir un système évolutif capable de gérer efficacement une grande quantité de données et de s'adapter à des besoins futurs. Le système doit être conçu de manière à permettre l'ajout de nouveaux capteurs, l'extension des fonctionnalités et la gestion de charges de travail croissantes sans compromettre les performances.

Ces objectifs fournissent une orientation claire pour le développement du projet et permettent de garantir que le système répondra aux besoins des utilisateurs tout en offrant une expérience conviviale et sécurisée.

### Planification du projet

Voici une planification générale du projet de collecte et de visualisation des données à partir de capteurs :

* Phase de définition et de planification (2 semaines) :
* Définir les objectifs du projet et les résultats attendus.
* Identifier les parties prenantes et leurs besoins.
* Établir une équipe de projet et assigner des rôles et des responsabilités.
* Élaborer un plan de projet détaillé, y compris les étapes suivantes.
* Phase d'analyse des besoins (1 semaine) :
* Analyser les exigences fonctionnelles et techniques du projet.
* Identifier les capteurs nécessaires et les paramètres à collecter.
* Définir les spécifications techniques pour l'ESP32, les bases de données, et l'application web.
* Phase de conception (2 semaines) :
* Concevoir l'architecture globale du système, y compris les composants matériels et logiciels.
* Définir les interfaces entre les capteurs, l'ESP32, les bases de données et l'application web.
* Concevoir l'interface utilisateur de l'application web pour la visualisation des données.
* Phase de développement (6 semaines) :
* Configurer les capteurs et l'ESP32 pour la collecte des données.
* Créer les bases de données MongoDB et PostgreSQL avec les schémas appropriés.
* Développer les fonctionnalités de stockage des données dans les bases de données.
* Programmer l'ESP32 pour collecter les données des capteurs et les transmettre aux bases de données.
* Développer l'application web avec Flask pour la visualisation des données.
* Phase de tests et de validation (2 semaines) :
* Effectuer des tests unitaires pour vérifier le bon fonctionnement des composants individuels.
* Réaliser des tests d'intégration pour s'assurer de la bonne communication entre les différents modules.
* Effectuer des tests de performance et de sécurité pour garantir la robustesse du système.
* Valider le système en comparant les données collectées avec des sources de référence.
* Phase de déploiement (1 semaine) :
* Déployer le système sur les serveurs ou les dispositifs appropriés.
* Configurer les paramètres de connexion et les autorisations nécessaires.
* Mettre en place des mesures de sécurité pour protéger les données et les utilisateurs.
* Phase de formation et de documentation (1 semaine) :
* Fournir une formation aux utilisateurs finaux sur l'utilisation du système.
* Préparer une documentation complète comprenant des guides d'utilisation et des manuels techniques.
* Phase de mise en production (1 semaine) :
* Déployer le système en production pour les utilisateurs finaux.
* Assurer le suivi du système pour détecter les problèmes potentiels et effectuer des ajustements si nécessaire.
* Phase de suivi et de maintenance (en continu) :
* Surveiller l'utilisation du système et résoudre les problèmes rencontrés par les utilisateurs.
* Effectuer des mises à jour régulières pour améliorer les fonctionnalités et la sécurité du système.

Il est important de noter que cette planification est indicative et peut varier en fonction des spécificités du projet, de la disponibilité des ressources et des contraintes de temps. Il est recommandé de diviser le projet en tâches plus petites et de les planifier avec des délais réalistes pour une exécution efficace.

## CHAPITRE 3 : METHODOLOGIE DE DEVELOPPEMENT

### DESCRIPTION DES METHODES UTILISEES

Le projet de collecte et de visualisation des données utilisera différentes méthodes pour atteindre ses objectifs. Voici quelques-unes des méthodes couramment utilisées dans ce type de projet :

#### Configuration des capteurs

Les capteurs sélectionnés seront configurés en fonction des besoins spécifiques du projet, tels que la fréquence d'échantillonnage, la précision, etc. Cette configuration peut impliquer la programmation des capteurs ou l'utilisation de bibliothèques spécifiques pour communiquer avec eux.

#### Programmation de l'ESP32

L'ESP32 sera programmé pour collecter les données provenant des capteurs. Cela peut être réalisé en utilisant le langage de programmation Arduino, qui est couramment utilisés pour programmer l'ESP32.

#### Communication avec les bases de données

Pour stocker les données collectées, le système utilisera des bases de données MongoDB et PostgreSQL. Des bibliothèques spécifiques comme pymongo (pour MongoDB) et psycopg2 (pour PostgreSQL) seront utilisées pour établir des connexions et effectuer des opérations de lecture et d'écriture dans les bases de données.

#### Développement de l'application web avec Flask

Le framework Flask sera utilisé pour développer l'application web permettant aux utilisateurs de visualiser les données collectées. Flask utilise des méthodes et des décorateurs pour gérer les routes, les requêtes HTTP et les réponses correspondantes.

#### Utilisation du protocole HTTP

Le système utilisera le protocole HTTP pour permettre la communication entre l'ESP32 et le serveur. L'ESP32 enverra les données collectées au serveur en utilisant des requêtes HTTP, et le serveur enverra les réponses appropriées à l'ESP32 et aux utilisateurs de l'application web.

#### Sécurité des données

Des méthodes de sécurité seront mises en place pour protéger les données collectées et les informations des utilisateurs. Cela peut inclure l'authentification des utilisateurs, le chiffrement des données sensibles, la validation des entrées utilisateur, etc.

#### Analyse et visualisation des données

Des méthodes d'analyse de données peuvent être utilisées pour extraire des informations pertinentes à partir des données collectées. Ces informations peuvent ensuite être visualisées sous forme de graphiques, de tableaux ou d'autres représentations visuelles pour une compréhension facile et une interprétation des données.

Il est important de noter que les méthodes spécifiques utilisées peuvent varier en fonction des technologies, des bibliothèques et des langages de programmation choisis pour implémenter le projet.