

Rapport de Projet : Application Client/Serveur avec SuperSet

21/06/2024

Une Analyse et Mise en Œuvre d'une Solution de Visualisation des Données

Résumé : Ce rapport présente le développement d'une application client/serveur utilisant Apache Superset pour la visualisation des données. L'objectif du projet était de créer une plateforme interactive permettant aux utilisateurs d'explorer et d'analyser des ensembles de données complexes. La méthodologie adoptée inclut la configuration de l'infrastructure serveur, le développement de l'interface client, et l'intégration de Superset pour la visualisation. Les résultats obtenus démontrent l'efficacité de la solution dans la facilitation de l'analyse des données.

Encadré par :

MR LIONEL

Réalisé par :

- TH027: TIEMELE Adolphe Venceslas Kouamé
- TH038 : KOUADIO TIA EMMANUEL
- TH076 : GOLI YAO JEAN-JAURES
- TH039 : Yapo Beda Eric Abed-nego
- *TH050*: ANZIAN BROU ULRICH BERANGER KOFFI

Table des Matières

1. Introduction.....	p01
2. Définition des Besoins Fonctionnels.....	p01
2.1 Visualisation de Données à partir de Différentes Sources de Données.....	p01
2.2 Création et Personnalisation de Tableaux de Bord Interactifs.....	p02
2.3 Exploration et Analyse des Données avec Filtres et Visualisations Variées..	p02
2.4 Gestion des Utilisateurs et des Autorisations.....	p03
3. Conception Technique.....	p04
3.1 Architecture de l'Application.....	p04
3.2 Configuration de SuperSet.....	p04
4. Plan de Mise en Œuvre.....	p05
4.1 Phase de Planification.....	p05
4.2 Phase de Développement.....	p05
4.3 Phase de Test.....	p05
4.4 Phase de Déploiement.....	p05
5. Conclusion.....	p06

1 Introduction

Ce rapport présente le développement d'une application client/serveur utilisant Apache Superset pour la visualisation et l'analyse de données. L'application vise à fournir une interface intuitive pour la création et la personnalisation de tableaux de bord interactifs, tout en gérant les utilisateurs et les autorisations de manière sécurisée.

2 Définition des Besoins Fonctionnels

2-1. Visualisation de Données à partir de Différentes Sources de Données

Description :

L'application doit se connecter à plusieurs sources de données telles que des bases de données relationnelles (MySQL, PostgreSQL), des bases de données NoSQL (MongoDB), des fichiers CSV et des API.

Fonctionnalités :

- Connexion et configuration de diverses sources de données.
- Importation et synchronisation des données.
- Mise à jour automatique ou manuelle des données.

2.2 Création et Personnalisation de Tableaux de Bord Interactifs

Description :

Les utilisateurs doivent pouvoir créer et personnaliser des tableaux de bord avec différents types de visualisation (graphiques, cartes, tableaux).

Fonctionnalités :

- Interface intuitive pour la création de tableaux de bord.
- Large choix de types de visualisation (barres, lignes, secteurs, cartes géographiques, etc.).
- Personnalisation des visuels (couleurs, titres, étiquettes).
- Sauvegarde et partage des tableaux de bord.

2.3. Exploration et Analyse des Données avec Filtres et Visualisations Variées

Description :

Les utilisateurs doivent pouvoir explorer et analyser les données en utilisant des filtres dynamiques et des visualisations interactives.

Fonctionnalités :

- Filtres dynamiques pour affiner les jeux de données.
- Drill-down pour explorer des niveaux de données plus détaillés.
- Interactions entre visualisations.
- Exportation des données et des visualisations.

2.4. Gestion des Utilisateurs et des Autorisations

Description :

Gestion des utilisateurs avec définition de rôles et permissions pour contrôler l'accès aux données et fonctionnalités.

Fonctionnalités :

- Système d'authentification (connexion, inscription).
- Gestion des rôles et permissions (administrateur, éditeur, visualiseur).
- Attribution de permissions spécifiques aux utilisateurs et groupes.
- Audit des actions des utilisateurs pour la sécurité.

3 Conception Technique

3-1 Architecture de l'Application

- **Frontend** : Application web en Node.js pour l'interface utilisateur.
- **Backend** : Serveur en Flask (Python) pour la gestion des utilisateurs, des rôles et des autorisations.
- **SuperSet** : Utilisé pour la visualisation des données et la création de tableaux de bord.
- **Base de données** : Mysql pour la gestion des utilisateurs, des rôles et des autorisations.

3-2 Configuration de SuperSet

*** Installation et Configuration :**

- Installer Apache Superset.
- Configurer la connexion à diverses sources de données.
- Créer des exemples de tableaux de bord pour démonstration.

*** Intégration avec le Backend:**

- Utiliser l'API REST de SuperSet pour automatiser certaines tâches (gestion des utilisateurs et des rôles).

*** Sécurité et Permissions:**

- Configurer les rôles et les permissions dans SuperSet.

- Synchroniser ces permissions avec celles gérées par le backend Flask.

4 Plan de Mise en Œuvre

4-1. Phase de Planification

- Réunions initiales pour définir les spécifications détaillées.
- Attribution des rôles et des tâches parmi les participants.

4-2. Phase de Développement

- **Semaine 1-2** : Installation et configuration de SuperSet et du backend Flask.
- **Semaine 3-4** : Développement des fonctionnalités frontend en node.js.
- **Semaine 5-6** : Intégration et tests des différentes parties du système.
- **Semaine 7-8** : Débogage, optimisation et documentation.

4-3. Phase de Test

- Tests unitaires et intégrés pour chaque composant.
- Tests utilisateurs pour validation des fonctionnalités et de l'ergonomie.

4-4. Phase de Déploiement

- Déploiement sur un serveur cloud (AWS, Azure, GCP).
- Formation des utilisateurs finaux.
- Maintenance et mises à jour régulières.

5 Conclusion

Ce projet a permis aux participants de se familiariser avec la création d'une application client/serveur en utilisant SuperSet pour la visualisation des données. L'application offre une plateforme robuste et sécurisée pour l'analyse des données, avec une architecture bien définie et une gestion rigoureuse des rôles et des permissions. Le projet a été structuré en phases claires, allant de la planification au déploiement, en passant par le développement et les tests, assurant ainsi une mise en œuvre efficace et une transition fluide vers l'utilisation par les utilisateurs finaux.