



DCI-IT-GAME

Documentation technique

Matéo Monteiro, Baptiste Roméra,
Mathéo Risselin, Lénäic Pascalis

ALTF42

Sommaire

1.	Objectif du Projet	2
2.	Technologies Utilisées	3
3.	Mise en place de l'infrastructure.....	4
4.	Mise en place de la BDD.....	5
5.	Fonctionnalités Clés de l'Outil	8
6.	Organisation du projet.....	8
7.	Proposition sur de nouveaux indicateurs à valeur ajoutée	9
8.	Comparatif on-premise et on-demand.....	10
9.	Conclusion	11

1. Objectif du Projet

L'objectif de ce projet est de concevoir une interface web dynamique dédiée à la visualisation et à l'analyse des coûts d'infrastructure informatique au sein de la DSI Infrastructure du groupe

Cette plateforme permettra d'afficher sous forme de tableaux et de graphiques interactifs des données financières et statistiques issues des métriques de consommation des infrastructures IT. Elle offrira une vision claire des coûts au service DSI en fonction des services hébergés, facilitant ainsi l'optimisation budgétaire et la prise de décision stratégique.

Cette interface aidera ainsi les responsables de la DCI à optimiser la gestion des coûts d'hébergement, à identifier les réductions budgétaires et à prendre des décisions éclairées sur le choix des applications et services à privilégier en fonction de leur rentabilité et de leur consommation de ressources.

De plus, le projet inclut une étude comparative des coûts entre une solution On-Premise et On-Demand afin d'évaluer les meilleures options d'hébergement pour la plateforme.

2. Technologies Utilisées

Technologie	Utilisation
Serveur Ubuntu 24.04.2	Hébergement du projet
Ressources	2 vCPU, 8 Go RAM, 20 Go stockage
Apache2	Serveur web pour héberger le site
MariaDB	Base de données pour stocker les données financières
My SQL sécurée	
PHP	Traitement des requêtes SQL et envoi des données en JSON
JavaScript (Fetch API, Chart.js)	Affichage dynamique et graphique
HTML / CSS	Interface utilisateur et stylisation
OpenSSL	Certificat SSL auto-signé pour HTTPS pour protocole https

3. Mise en place de l'infrastructure

1. Mise à jour du serveur

```
sudo apt update && sudo apt upgrade -y
```

2. Installation des dépendances

```
sudo apt install apache2 mariadb-server php libapache2-mod-php php-mysql unzip -y
```

3. Activation des services

```
sudo systemctl enable apache2
```

```
sudo systemctl enable mariadb
```

4. Sécurisation de la base de données

```
sudo mysql secure_installation
```

5. Sécurisation avec un Certificat SSL auto signée (Création du certificat et de la clé privé) pour ajout du protocole HTTPS.

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \ -keyout  
/etc/ssl/private/server.key -out /etc/ssl/certs/server.crt
```

4. Mise en place de la BDD

1. Connexion a Maria DB en root

Sudo mysql -u root -p

```
dsi@DCS-IT-GAME:~$ sudo mysql -u root -p
[sudo] password for dsi:
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 50
Server version: 10.11.8-MariaDB-0ubuntu0.24.04.1 Ubuntu 24.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> █
```

2. Création de la base de données

CREATE DATABASE tableau_bord;

USE tableau_bord;

```
MariaDB [(none)]> USE tableau_bord;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [tableau_bord]> █
```

3. Importation du fichier SQL

mysql -u root -p tableau_bord < ./home/bi_files/creation_bdd_tableau_bord.sql

4. Vérifier les tables

SHOW TABLES;

```
MariaDB [(none)]> USE tableau_bord;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [tableau_bord]> SHOW TABLES;
+-----+
| Tables_in_tableau_bord |
+-----+
| application            |
| centresactivite        |
| clients                 |
| famille                |
| grandclients           |
| ligne_facturation       |
| produit                |
+-----+
7 rows in set (0,001 sec)
```

5. Vérifier les données

SELECT * FROM clients LIMIT 5;

SELECT * FROM ligne_facturation LIMIT 5;

```
MariaDB [tableau_bord]> SELECT * FROM clients LIMIT 5;
+-----+-----+-----+-----+
| ClientID | NomClient | GrandClientID | CentreActiviteID |
+-----+-----+-----+-----+
| 1        | Client2_1 | 689           | 3055              |
| 2        | Client2_1 | 689           | 3056              |
| 3        | Client2_1 | 689           | 3057              |
| 4        | Client2_1 | 689           | 3058              |
| 5        | Client1_4 | 690           | 3059              |
+-----+-----+-----+-----+
5 rows in set (0,001 sec)
```

6. Configurer des .PHP pour l'importation des CSV dans la base de données

Modification du code au niveau des lignes d'identification de maria DB pour donner accès au code a la BDD.

```
$servername = "localhost";
$username = "mateo";
$password = "123456+aze";
$dbname = "tableau_bord";

// Créer la connexion
$pdo = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
```

7. Importation des fichiers CSV dans la base de données

< > ↺ ⚠ Non sécurisé 192.168.106.132/insérer_code_applicatif.php

```
#####  
recuperation ligne IRT = A0166 nomAPPLI = Appli_1
```

8. Creation du code PHP pour faire le lien entre la BDD et l'interface WEB dans le dossier

Sudo nano ./var/www/html/data_dashboard.php

Exemple code data_dashboard.php

```
// Encodage UTF-8 clair  
header('Content-Type: application/json; charset=utf-8');  
  
$mysqli = new mysqli('localhost', 'mateo', '123456+aze', 'tableau_bord');  
mysqli_set_charset($mysqli, "utf8");  
  
if ($mysqli->connect_errno) {  
    die(json_encode(['error' => $mysqli->connect_error]));  
}  
  
// Top 10 applications  
$query_top10 = "  
SELECT gc.NomGrandClient, a.nomAppli, SUM(lf.prix) AS total_facture  
FROM ligne_facturation lf  
JOIN application a ON lf.IRT = a.IRT  
JOIN centresactivite ca ON lf.CentreActiviteID = ca.CentreActiviteID  
JOIN clients c ON ca.CentreActiviteID = c.CentreActiviteID  
JOIN grandclients gc ON c.GrandClientID = gc.GrandClientID  
GROUP BY gc.NomGrandClient, a.nomAppli  
ORDER BY total_facture DESC  
LIMIT 10;  
";  
$res1 = $mysqli->query($query_top10);  
$top10apps = $res1->fetch_all(MYSQLI_ASSOC);  
  
// 5 plus grands clients  
$query_top5_clients = "  
SELECT gc.GrandClientID, gc.NomGrandClient, SUM(lf.prix) AS total  
FROM ligne_facturation lf  
JOIN centresactivite ca ON lf.CentreActiviteID = ca.CentreActiviteID  
JOIN clients c ON ca.CentreActiviteID = c.CentreActiviteID  
JOIN grandclients gc ON c.GrandClientID = gc.GrandClientID  
GROUP BY gc.GrandClientID  
ORDER BY total DESC LIMIT 5;  
";  
  
$res_top5 = $mysqli->query($query_top5_clients);  
$top5_clients = [];  
while ($row = $res_top5->fetch_assoc()) {  
    $top5_clients[$row['GrandClientID']] = $row['NomGrandClient'];  
}  
$ids_top5 = implode(',', array_keys($top5_clients));  
  
if(empty($ids_top5)){  
    die(json_encode(['error' => "Pas de données pour les 5 plus grands clients"]));  
}
```


5. Fonctionnalités Clés de l'Outil

Notre outil permet d'afficher dynamiquement les données suivantes :

Top 10 des Applications par Grand Client (€)

- Classement des applications les plus consommées en valeur.
- Affichage sous forme de tableau dynamique.

Évolution des Montants pour les 5 Clients les Plus Consommateurs

- Graphique interactif affichant les montants facturés entre **janvier 2021 et avril 2022**.
- Affichage **distinct de chaque client** pour une meilleure lisibilité.

Évolution des Volumes des Produits 1_1 et 1_4

- Suivi de l'évolution de la consommation de ces produits sur la même période.
- Affichage sous forme de courbes comparatives.

6. Organisation du projet

Mission d'installation, configuration et sécurisation de la machine :

Matéo Monteiro et Baptiste Roméra

Création de la BDD, Insertion des données, Mise en place de l'interface Web

Lénäic Pascalis et Mathéo Risselin.

7. Proposition sur de nouveaux indicateurs à valeur ajoutée

Nous avons conçu notre outil pour être évolutif et nous proposons d'ajouter les indicateurs suivants pour une analyse plus fine :

1. Ratio Facture/Volume Consommé

- Permet d'identifier l'efficacité de la facturation en calculant le prix moyen par unité consommée.

2. Taux de Croissance des Montants Facturés

- Un indicateur permettant d'identifier les variations mensuelles et annuelles des revenus.

3. Comparatif Annuel des Consommations

- Ajout d'un graphique comparant les performances de facturation entre plusieurs années.

5. Segmentation des Produits les Plus Rentables

- Identifier les produits qui génèrent le plus de chiffre d'affaires.

5. Alertes et Anomalies sur les Montants Facturés

- Détection des hausses ou baisses **anormales** d'un client ou d'une application.

8. Comparatif on-premise et on-demand

Critère	On-Premise	On-Demand
Coût initial	2 700 €	0 €
Coût mensuel	169 €	83 €
Coût total sur 3 ans	6 072 €	2 988 €
Installation	Manuel et demande des compétences	Déploiement rapide
Maintenance	À la charge de l'entreprise	Automatisée par le fournisseur
Sécurité	Contrôle total	Gérée par le fournisseur
Disponibilité	99%	99,9% à 99,999%
Mises à jour	Manuelles	Automatiques
Latence d'accès	réseau interne	Dépend d'Internet
Dépendance au fournisseur	Aucune	dépend du prestataire Cloud

9. Conclusion

Ce projet a permis de concevoir et de mettre en place un tableau de bord performant et intuitif dédié à la visualisation des données de facturation. Grâce à une architecture modulaire et sécurisée, l'outil propose une expérience utilisateur fluide et une grande adaptabilité aux besoins spécifiques des entreprises.

L'intégration des technologies Apache, PHP, MySQL/MariaDB et Chart.js a permis de garantir une interface réactive, dynamique et facile à maintenir. Le choix de Chart.js s'est avéré particulièrement pertinent pour la représentation graphique des données, offrant des visualisations claires et interactives qui facilitent l'analyse des tendances et des performances.

L'approche adoptée repose sur une séparation nette entre la récupération des données et leur affichage, via un script PHP (`data_dashboard.php`) interagissant avec la base de données et un script JavaScript (`script.js`) assurant le rendu dynamique des informations. Cette structuration permet non seulement d'améliorer la maintenabilité du projet, mais aussi de garantir une évolutivité future en intégrant de nouvelles sources de données ou en adaptant les fonctionnalités aux exigences des utilisateurs.

Par ailleurs, des optimisations ont été mises en place pour assurer de bonnes performances, notamment en réduisant les temps de requête et en optimisant l'affichage des graphiques pour une consultation fluide. La sécurité n'a pas été négligée, avec la mise en place de bonnes pratiques en matière de gestion des accès et de protection des données sensibles.

En somme, ce projet a démontré l'importance d'un tableau de bord bien structuré pour le suivi des indicateurs de facturation. Il constitue une base solide pouvant être enrichie avec de nouvelles fonctionnalités, telles que des filtres avancés, des exports de données ou encore l'intégration d'outils d'intelligence artificielle pour l'analyse prédictive.