

Rapport du **Projet** **Social** en équipe



Projet!

Résumé du projet en 2 lignes

- Application de chat entre de multiples utilisateurs
- Chats publics et privés
- Autres fonctionnalités: statistiques, réglages de paramètres divers





1

Le projet

Contributions individuelles

Contributions individuelles

- Maxime Daigle Vigneault:
 - Écran statistiques, dao statistiques, service de données statistiques.
- Hy-Vong Georges Dit Rap:
 - Écran chat privé, récupérer messages/noms utilisateurs
- Brandon Marceau:
 - Écran liste salons, construire interface et navigation
- Kenny Maréchal:
 - Écran paramètres, service et base de données
- Guillaume Vernaz
 - Écran chat public

À remplir avec les écrans, systèmes ou fonctionnalités travaillés

Données maître :

salon

Données transactionnelles :

message

Autres données :

Utilisateur, autorisation

À remplir avec le nom
des tables ou autres
(caches, entité)

Liste de vérifications

À répondre avec oui
ou non

SAUVEGARDE

Avez-vous vous bien placé chaque import ou export de données dans un DAO ? OUI

Avez-vous vous bien sauvegardé chaque SQL ou dépôt dans le git du projet ? OUI

SÉCURITÉ

Avez-vous utilisé les requêtes préparées à chaque fois que c'est possible ? OUI

Avez-vous filtré les inputs du service de données le cas échéant ?

Service statistiques : OUI



2

Les optimisations

Une par participant

Optimisation **Redis** pour cacher la liste de salons

J'ai créer une nouvelle classe qui se nomme JedisMain, Jedis est une des fonctionnalités de Redis, ensuite j'ai utiliser la liste des salons pour remplir la cache et j'ai fait en sorte que la vue utilise les informations stocker dans la cache. Pour que Redis soit fonctionnelle il faut l'executer sur son Windows.

<https://docs.google.com/presentation/d/1pdXP-BNPycdX6TfxTnNEZ3h54WbkxVxykb14wklPbc/edit#slide=id.p5>

<https://github.com/cegepmatane/projet-social-20-Brandon-Kenny-Guillaume-HyVong-Maxime/commit/e10e92c8c337dd4cbf9bf26fe553765ab32e64f5>



PROGRAMMEUR : Prénom Nom

Optimisation **Cache DAO** cachant les options

Explication bla bla bla

Lien vers cette partie dans le site github



Optimisation **SQL** Simplifier la requête

Séparation de la requête sql qui génère les statistiques en deux fonctions. Cela permet de créer des des variables et de les utiliser plusieurs fois sans avoir à les recalculer. Demande moins de ressources à la BD postgres.

<https://github.com/cegepmatane/projet-social-20-Brandon-Kenny-Guillaume-HyVong-Maxime/tree/main/Fichier%20PHP/Maxime>



PROGRAMMEUR : Prénom Nom

Optimisation de **taille de la table** message

Explication bla bla bla

Lien vers cette partie dans le site github



PROGRAMMEUR : Kenny Marechal

Optimisation de la **bases de donnée**

Le but de mon optimisation et d'augmenter les performances de la base de donnée. On pourrait pour cela. optimiser les requêtes SQL, Le système de gestion des bases de données et encore Les structures de bases de données.

Via les différents services utiliser j'ai décidé de limiter le nombre de lignes et de colonnes retournées.

Création de fonction sql pour générer les timestamps automatiquement

[Lien vers cette partie dans le site github](#)



Bonnes vacances !

Gardez bien votre travail pour votre futur portefeuille.

“ there is no such thing as a one-size-fits-all portfolio.”

