

Projet Cloud

Vous travaillez au sein d'une PME société d'hébergement fournissant des solutions de stockage en ligne de type cloud à ses clients.

Vous êtes chargé de mettre en place une application web permettant aux clients de consulter et de gérer l'utilisation des disques qu'ils louent.

Résumé

| | |
|------------------------------------|---|
| Projet initial à utiliser | <ul style="list-style-type: none">• Projet RT-Cloud v1.0 sur gitHub• La base de données est dans le dossier app/database du projet |
| Outils | <ul style="list-style-type: none">• Bootstrap• Micro-framework |
| Principales fonctionnalités | <ul style="list-style-type: none">• Report d'incidents• Base de connaissances |
| Livraison | <ul style="list-style-type: none">• Jusqu'au jour de la soutenance, par gitHub |

Détail des fonctionnalités à mettre en place

Les différentes pages devront gérer les droits, et n'être autorisées que pour les acteurs mentionnés. Tenir compte du fait que l'administrateur a le droit d'accéder à tout ce qui est autorisé à l'utilisateur.

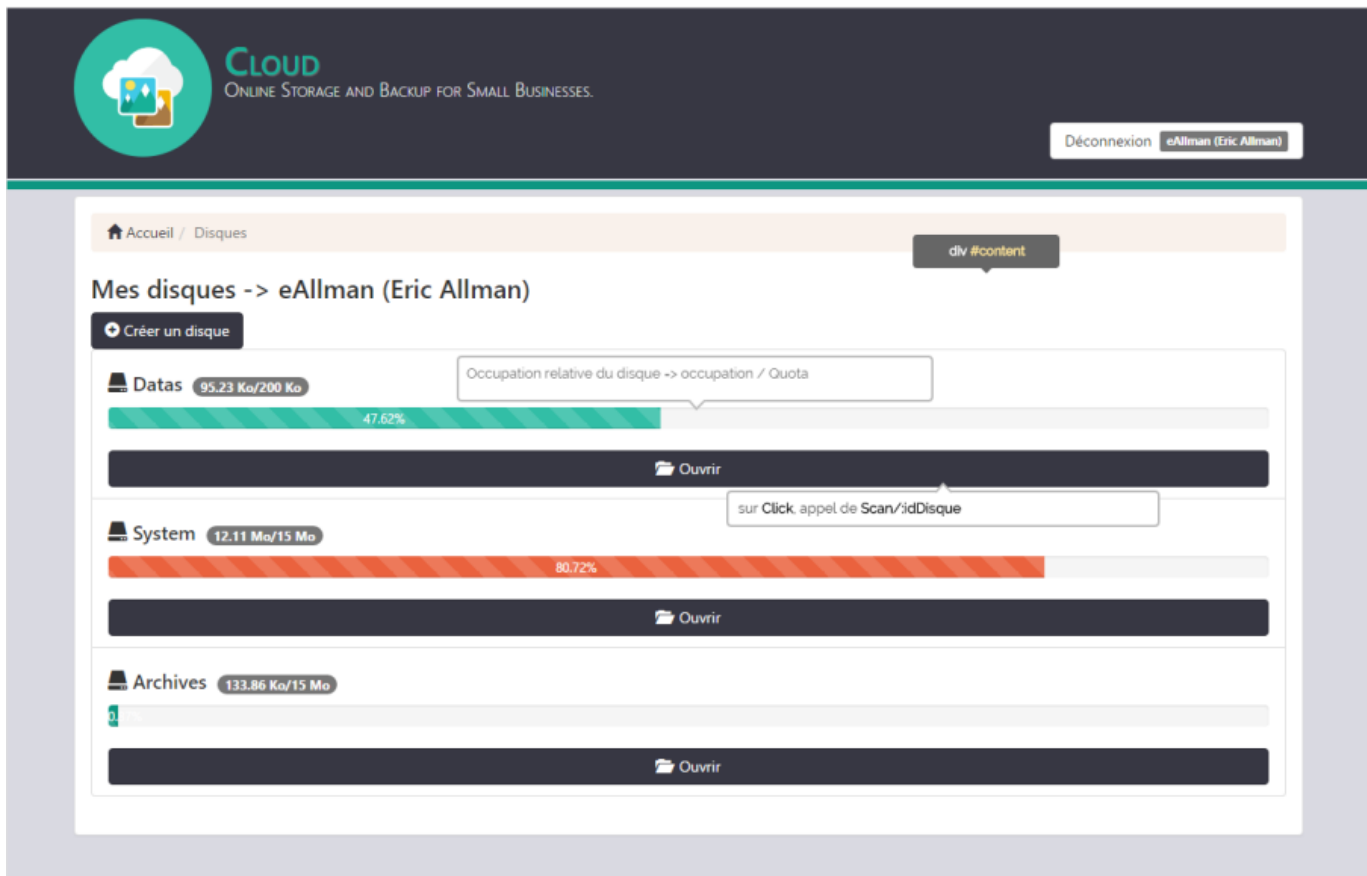
-- Module Client

-- Affichage des disques d'un client

//TODO 1.1

A l'adresse **MyDisques/index**, on souhaite afficher les disques de l'utilisateur actuellement connecté.

L'utilisateur connecté est obtenu par l'appel de la méthode **Auth::getUser()**.



Indications

| Elément | Indications |
|------------------------------|---|
| Contrôleur | MyDisques |
| Action | index |
| Utilisateur connecté | L'utilisateur connecté est obtenu par l'appel de la méthode Auth::getUser() . |
| Occupation, Quota | le quota est obtenu sur le tarif actuel du disque, la classe ModelUtils permet de connaître l'occupation en cours du disque |
| Composants visuels Bootstrap | on pourra utiliser les composants bootstrap Progressbar, GlyphButton, Listgroup |
| Accès à la config du cloud | Les disques clients sont localisés dans le dossier /files de l'application, et les disques y sont localisés sous le nom : \$cloud->prefix-[disque.nom] . Cette configuration est définie dans le fichier de config.php dans la variable cloud , accessible depuis les contrôleurs par \$GLOBALS["config"]->cloud |
| Tailles en octets | Vous pourrez utiliser la méthode sizeConverter de ModelUtils |

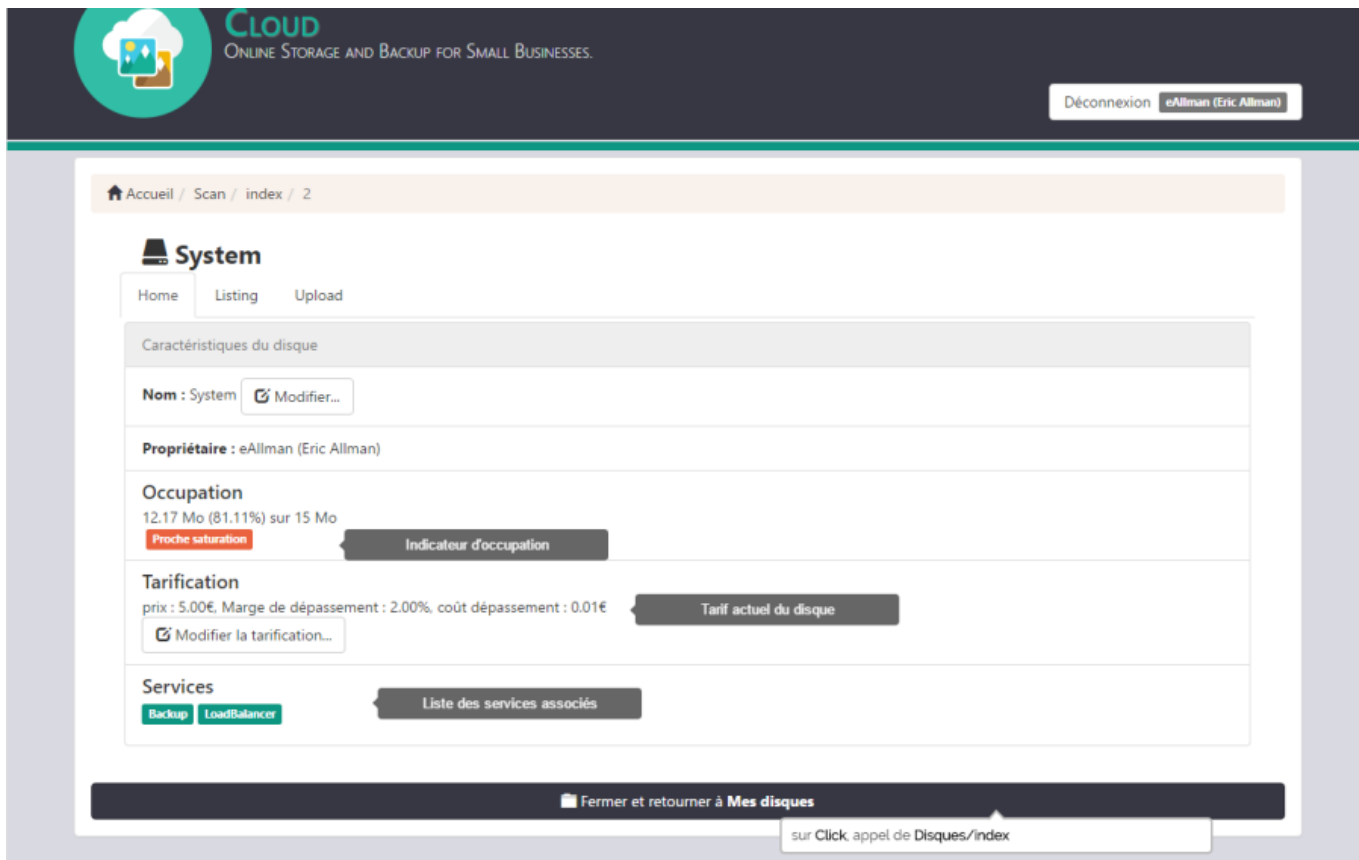
Le style des progressbars doit donner une indication sur le taux d'occupation en % :

| Style | Valeurs |
|---------|-------------|
| info | de 0 à 10% |
| success | de 10 à 50% |
| warning | de 50 à 80% |
| danger | plus de 80% |

-- Affichage d'un disque

//TODO 1.2

A l'adresse **Scan:/idDisque**, afficher l'interface suivante :



| Elément | Indications |
|------------------------------|--|
| Contrôleur | Scan |
| Action | index |
| Paramètre | idDisque |
| Utilisateur connecté | L'utilisateur connecté est obtenu par l'appel de la méthode Auth::getUser(\$controller) où \$controller est le contrôleur actif. |
| Occupation, Quota | le quota est obtenu sur le tarif actuel du disque, la classe ModelUtils permet de connaître l'occupation en cours du disque |
| Composants visuels Bootstrap | on pourra utiliser les composants bootstrap Label, GlyphButton, Listgroup |

L'indicateur d'occupation (**Label**) doit donner une indication sur le taux d'occupation en % :

| Texte | Style | Valeurs |
|-------------------|---------|-------------|
| Peu occupé | info | de 0 à 10% |
| RAS | success | de 10 à 50% |
| Forte occupation | warning | de 50 à 80% |
| Proche saturation | danger | plus de 80% |

Conseils importants :

- Ajax est recommandé pour toutes les requêtes

-- Création d'un disque

//TODO 1.3

A partir de l'adresse **Disques/frm**

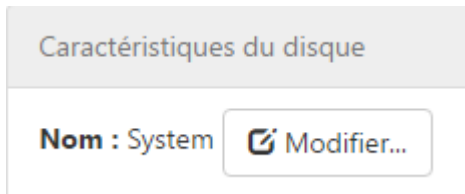
Afficher le formulaire de création d'un disque.

Soumettre le formulaire à l'adresse **Disques/update** pour créer le disque puis rediriger vers l'adresse **Scan/index/:idDisque**

Penser à le créer physiquement.

-- Renommage disque

A partir de l'adresse **Scan/index/:idDisque**



Implémenter la modification du nom d'un Disque (avec renommage éventuel du dossier associé + vérification du nom)

-- Module administration

-- Page d'accueil

-- Accès aux données

-- Accès aux disques

-- Module commun

-- Contrôle d'accès

| URL | Accès | Résultat/message |
|-----------------------------|---|---|
| Disques/index | Accessible uniquement pour un utilisateur connecté | Affiché message sur l'absence de connexion et proposer la connexion |
| Scan/index/:idDisque | Accessible uniquement pour un disque appartenant à l'utilisateur connecté | Afficher un message d'erreur "Accès à une ressource non autorisée" |

-- Fil d'ariane

Base de données

Contraintes techniques

- L'application sera développée en PHP objet, et utilisera un [micro-framework](#) facilitant les échanges avec la base de données.
- Elle respectera au mieux la séparation des couches (objets Métiers), classes techniques et vues

- (interfaces web de saisie et d'affichage).
- Elle utilisera la base de données Mysql fournie en annexe. Cette base pourra évoluer en fonction des besoins du développement.
- L'utilisation de scripts côté client (javascript et ajax) pourra compléter les validations côté serveur. L'utilisation du micro-framework est obligatoire, d'autres librairies ou frameworks peuvent le compléter..
- [Bootstrap](#) sera utilisé pour la partie présentation.

Fichiers

- [Projet Github initial à cloner](#)
- [Grille d'évaluation projet](#)

Modalités de remise du travail

- Date remise : rendre projet PHP + readme.md voir [Modalités de remise de votre travail](#)
- Date passage : soutenances

Déroulement de l'oral

Durée

- 5 minutes max de présentation par membre de l'équipe
- 5 minutes max de questions

Contenu

Il s'agit de montrer, le travail effectué, ainsi que les concepts maîtrisés :

- En présentant les fonctionnalités implémentées (démonstration du fonctionnement)
- En donnant des explications techniques sur le fonctionnement (contrôleurs, vues, classes, sécurisation...)

Compléments

Bonnes pratiques

- Alimenter correctement la base de données en ajoutant des enregistrements valides et en nombre suffisant, mettant en valeur votre travail
- respecter la Normalisation HTML 5/Css 3
- Structurer les fichiers et dossiers de manière cohérente
- Nommer en respectant les normes et de manière significative (Contrôleurs, vues, méthodes, variables...)

From:
<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:
<http://slamwiki2.kobject.net/php-rt/projets/projet-2016?rev=1459011579>

Last update: **2019/08/31 14:26**



