

XCloud

Votre solution de stockage, clé en main, où que vous soyez.

Mohamed Souissi
Grégory Brunet
Stanislas Durand
ESGI - 4AL2
10/02/2020

Sommaire :

Introduction	1
Utilisation de l'application	2
Votre première connexion	2
Uploader un document	3
Architecture technique	4
Architecture	4
Conclusion	5

1. Introduction

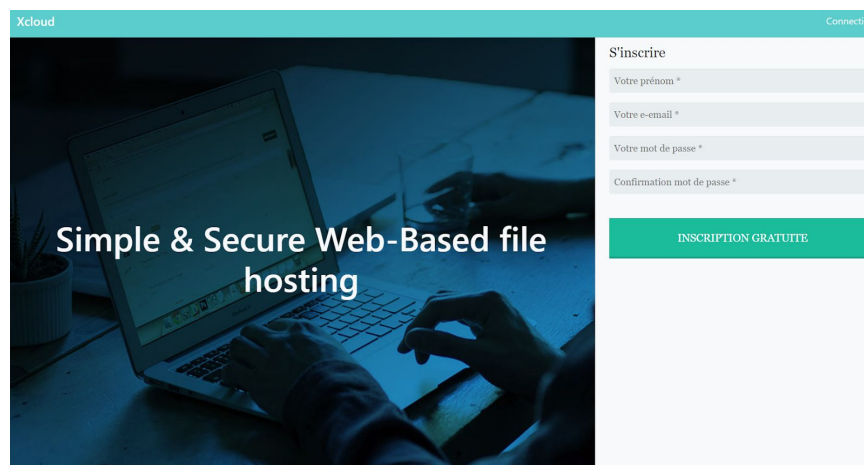
XCloud a pour vocation de vous offrir une solution de stockage pour vos fichiers où que vous soyez. Il s'agit de la documentation fonctionnel du site. Vous pouvez accéder à la documentation technique via notre repository github.

- Adresse du site public : <https://cloudxxappservice.azurewebsites.net/>
- Le repository du projet et la documentation technique :

2. Utilisation de l'application

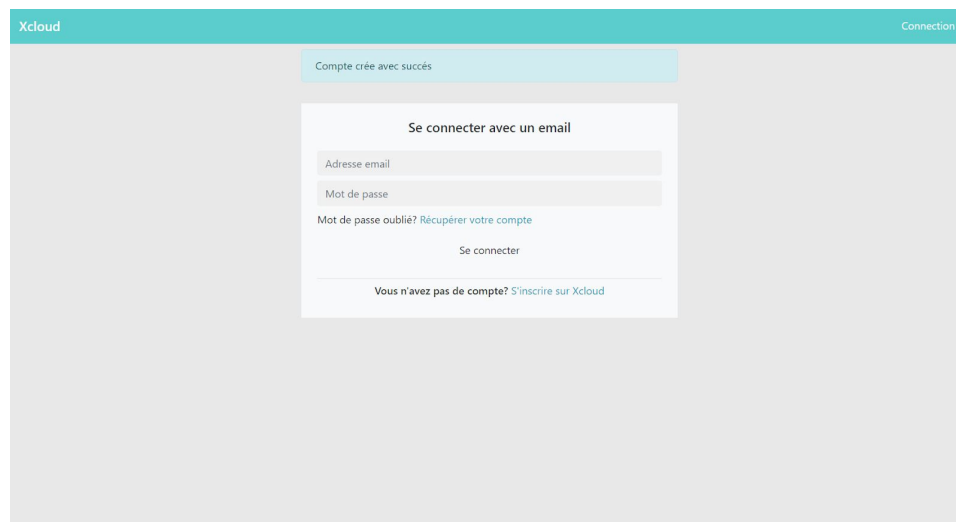
a. Votre première connexion

- Commencez par accéder à l'URL : <https://cloudxxappservice.azurewebsites.net/>
- Remplissez le formulaire d'inscription puis cliquez sur "inscription gratuite"



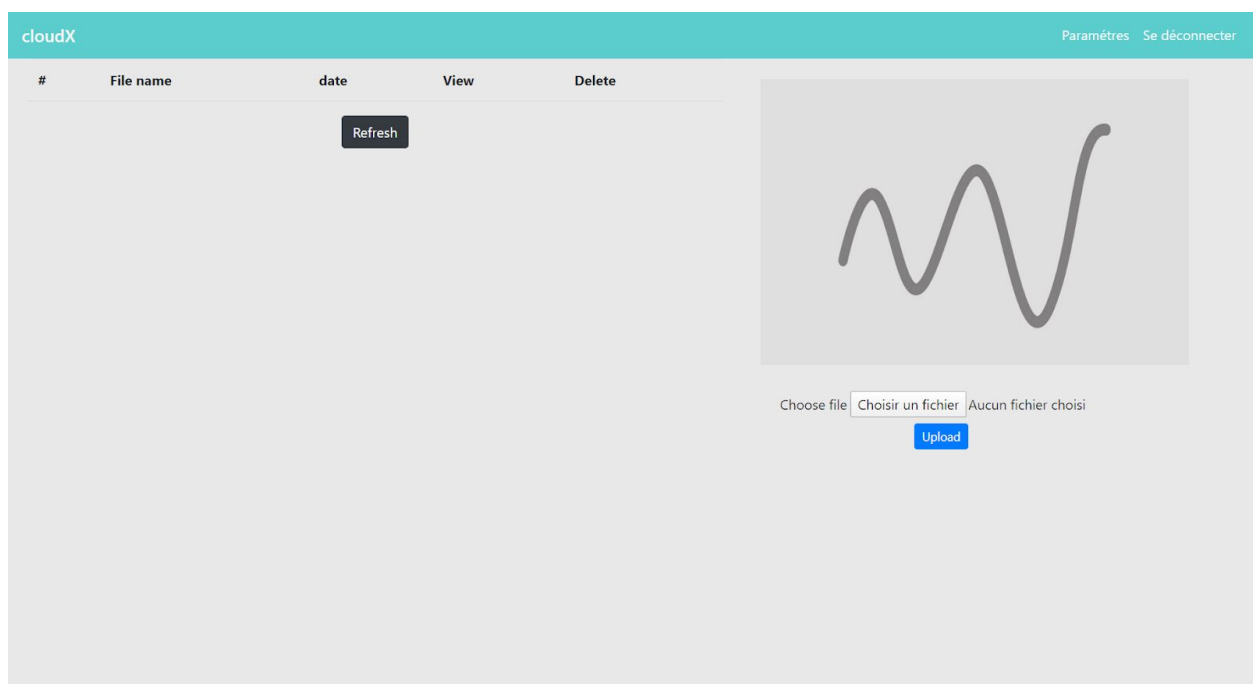
- Vous accédez maintenant à la page de connexion de l'application. Entrez dans le formulaire vos identifiants puis cliquez sur "se connecter".

-

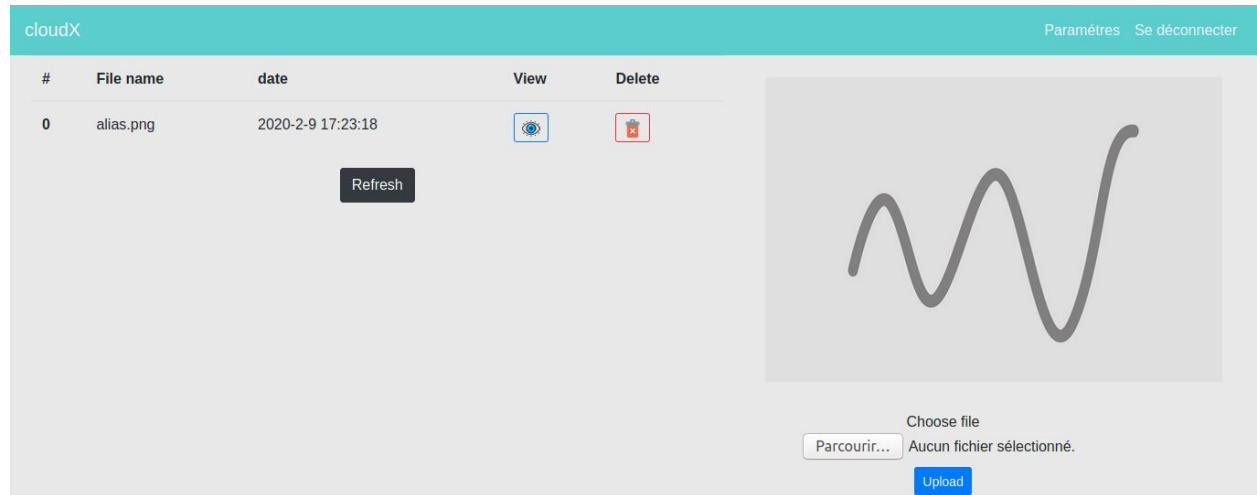


Uploader un document

Vous voilà maintenant connecté à XCloud. Vous pouvez utiliser l'application. Commençons par uploader votre premier document.



- Sur l'écran devant vous sélectionnez "choisir un fichier". Choisissez un document à uploader sur votre système puis cliquez sur "Upload".
- Après quelques secondes le document est téléchargé. Il apparaît maintenant dans le tableau à gauche de l'écran. Dans le cas contraire, sélectionnez "refresh".
- Vous pouvez maintenant utiliser XCloud à votre convenance.



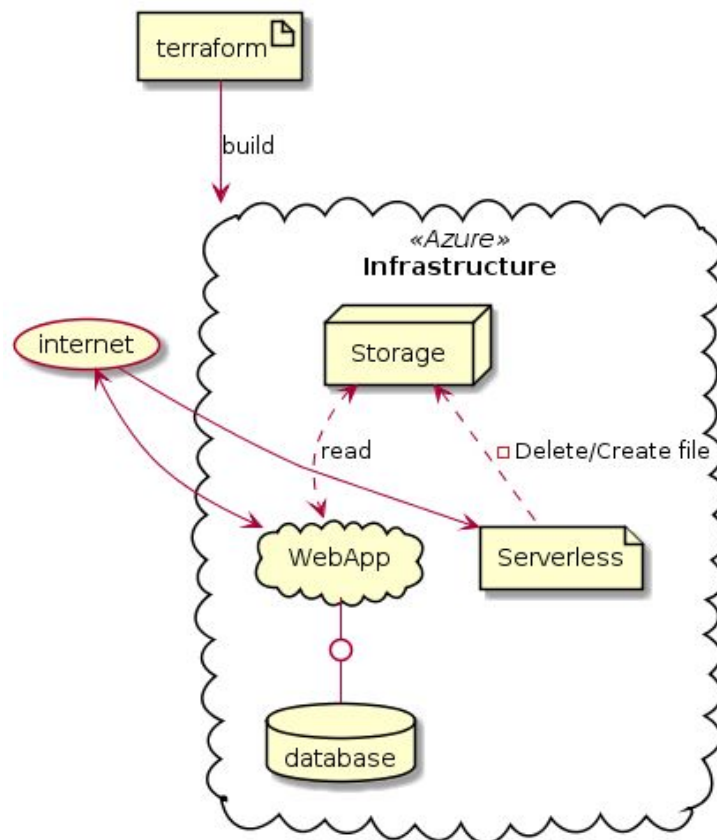
Une fois le fichier uploader, l'utilisateur aura la possibilité soit de supprimer le fichier soit de le visualiser.

Architecture technique

Architecture

Grâce à un script terraform qui construit les ressources nécessaire dans Azure. nous pouvons installer notre application et la rendre disponible rapidement. Voici la liste de nos ressources :

- Web app sous NodeJs avec pour base le framework Express. Elle a pour fonction de fournir une interface de gestion à l'utilisateur. Elle a un accès en lecture au serveur de stockage.
- Une base de donnée Mysql pour le stockage des comptes utilisateur.
- Un serveur de stockage contenant les fichiers crée par les clients.
 - Les fichiers uploads sont représentés par des blob contenu dans le serveur de stockage ils auront par défaut pour nom : "userId-BlobName".
 - Chaque blob est placé dans un fichier propre à son propriétaire. Un fichier aura comme structure de nom : "userId-fileName"
- Une fonction Serverless contenant les fonctionnalités de création et de suppression d'un document dans le composant Storage.



Utilisation d'un container jenkins dans un cloud.

Dans un contexte de déploiement continue, notre script terraform vous donne également accès à un serveur Jenkins grâce à notre image présente sur le Docker Hub : [moxxa/project-orchestration](https://hub.docker.com/r/moxxa/project-orchestration).

Il vous suffira de configurer Jenkins en fonction du système de cloud que vous utilisez comme Aws ou Azure, et de cloner notre repository github. Vous aurez alors la possibilité de développer votre propre version de xCloud.

Conclusion

Pour installer votre propre version de XCloud, consultez notre repository github. Vous y trouverez la documentation et les spécification technique directement dans le fichier "readme".

Reade.me :

XCloud a pour vocation de vous offrir une solution de stockage pour vos fichiers où que vous soyez. Il s'agit de la documentation fonctionnel du site. Vous pouvez accéder à la documentation technique via notre repository github.

- Adresse du site public : <https://cloudxxappservice.azurewebsites.net/>
- Le documentation du projet :
<https://docs.google.com/document/d/13qyA8Ne7cLA5rd7EzKQJ4Tlpopohpyc5uhkp7tCx0YM/edit?usp=sharing>

Déployer le projet :

- 1 : Cloner le projet depuis github : https://github.com/moxxxa/projet_cloud.git
- 2: Pour démarrer le projet en local. Il vous faudra installer NodeJs et Npm. Puis de démarrer les commandes suivante :
 - npm i
 - node server.js || npm start
- 3 : Accédez au fichier variable.tf de configuration terraform puis entrez vos propres identifiants Azure puis ajouter la chaîne de connexion du compte du stockage dans le fichier ".env".
- 4 : Entrez ensuite dans le fichier config/db.js les identifiants de connexion au serveur mySql. Installez MysqlWorkbench puis connectez vous à votre base de donnée.
- 5 : Vous pourrez ensuite accéder au projet en accédant à votre