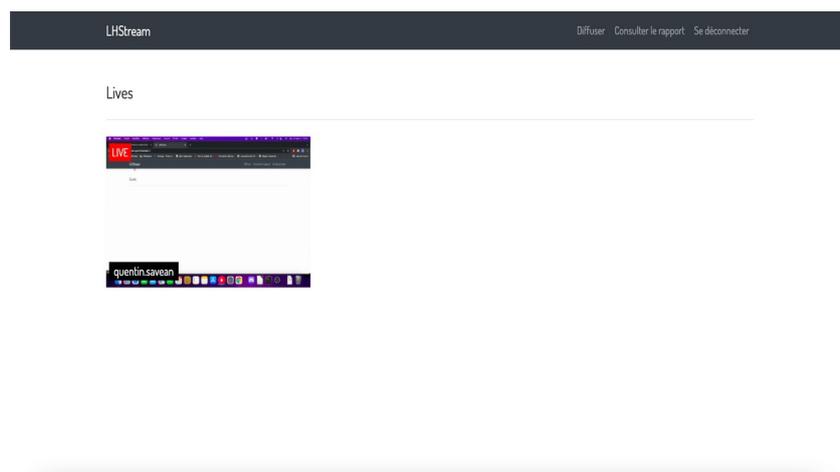


Université du Havre
IUT du Havre

M3102
Services Réseaux

Rapport de projet

**Réalisation d'un mini site de streaming
en Node.JS**



Rapport réalisé par :
Quentin Savéan

Enseignant de la matière :
Juan Luis Jimenez Laredo

ANNÉE UNIVERSITAIRE 2021 – 2022

Introduction

Lors de notre rapport pédagogique, nous avons évoqué l'utilisation du protocole RTP en expliquant son fonctionnement général à travers différents exemples jusqu'à la mise en place d'un serveur de streaming RTMP. Nous avons vu également à quel point celui-ci est présent dans notre quotidien, que ça soit depuis nos différents outils de communications par VoIP et de visionnage en ligne tel que des vidéos en direct ou à la TV. Cela sera justement l'occasion d'aborder un peu plus le sujet à travers la réalisation d'un projet plus conséquent où nous aurons l'occasion de mettre en valeur la création de notre précédent serveur RTMP en créant notre propre plateforme de vidéos en livestream!

Ce que j'entends par plateforme de livestream comme sa traduction l'indique, la création d'un site web de diffusion de vidéos de la même manière que Twitch ou encore YouTube qui représente une grande part d'audience sur internet et dans le monde en général, mais en beaucoup moins développé bien sûr, il s'agira ici de comprendre uniquement les bases du système qu'il y a derrière tout ça.

Ce rapport ne détaillera pas en détails chaque codes présent du projet mais son aspects plus général, c'est-à-dire les différents les choix que j'ai dû entreprendre et technologies à mettre en œuvre pour ce concept jusqu'à ça mise en production. Vous aurez d'ailleurs l'occasion de le tester chez vous si cela vous intéresse à travers le code mis à disposition depuis mon profil GitHub.

Dans la première partie, je parlerai de l'évolution du streaming dans le monde et de quelles façons est-il aussi présent aujourd'hui dans notre société. Dans la seconde partie, nous rentrerons dans le vif du sujet en détaillant la manière dont j'ai élaboré ce projet et les différents choix que j'ai entrepris. Puis dans la dernière partie, j'expliquerai comment fonctionne celle-ci, avec une petite conclusion. Alors je vous propose de commencer dès maintenant:)

Sommaire

Introduction.....	2
I – L’histoire de l’évolution du streaming dans le monde.....	4
1 – Qu’est-ce que le streaming ?.....	4
2 – Comment fonctionne le streaming ?.....	4
3 – Son évolution dans la société.....	5
II – Cahiers des charges.....	7
1 – Fonctionnalités.....	7
2 – Choix des technologies.....	8
3 – Structure de l’arborescence.....	8
III – Comment utiliser la plateforme ?.....	10
Étape 1 – Accéder à la plateforme.....	10
Étape 2 – Créer vous un compte.....	11
Étape 3 – Récupérer vos informations de connexions RTMP.....	12
Étape 4 – Diffuser un stream.....	13
Étape 5 - Démarrer la diffusion.....	15
Étape 6 - Visionner votre diffusion.....	16
IV – Conclusion.....	19

I – L’histoire de l’évolution du streaming dans le monde

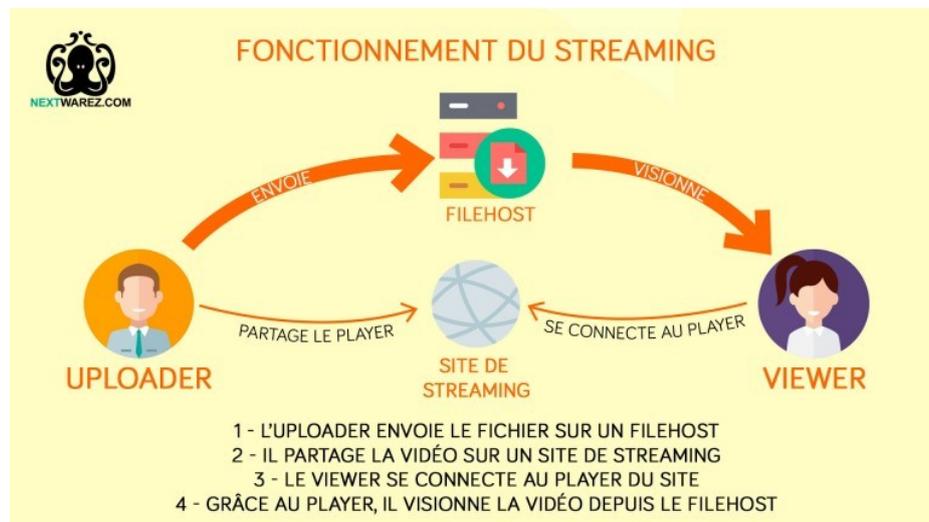
Le streaming fait partie de nos vies depuis plus d'une dizaine d'années : il est apparu quand Internet s'est démocratisé. Ce service est basé sur la transmission en continu d'un flux audio ou vidéo en ligne, sans téléchargement. Mais nous nous demandons forcément en quoi cette évolution est dû et l'envers du décor qu'il y a derrière d'une tel évolution audiovisuelle dans le notre société actuelle, ce que je t'entraî d'expliquer depuis les parties suivantes !

1 – Qu’est-ce que le streaming ?

Le streaming est utilisé pour visionner ou écouter des contenus en ligne. Ce protocole permet la lecture instantanée de vidéos ou de musiques, directement dans le navigateur web. Le streaming s’est largement développé sur Internet au début des années 2000, avec le lancement de grandes plateformes de streaming aujourd’hui très connues telles que YouTube, Dailymotion, Netflix ou encore Spotify. Le streaming permet ainsi simplement de regarder des vidéos ou d’écouter de la musique sur Internet, sans avoir à télécharger de fichier. Il est devenu un protocole incontournable. Ainsi, en 2018, on estime que le streaming représentait près de 58 % de la bande passante mondiale.

2 – Comment fonctionne le streaming ?

Le mot «streaming» vient du verbe «stream» en anglais, qui signifie «diffuser en flux (courant) continu». Le streaming est donc une diffusion sans arrêt et régulière qui permet de transporter des données. En l’occurrence, ici, des «bits» (petites quantités d’informations numérisées) qui sont encodés et compressés (rapetissés) par rapport au fichier d’origine (la musique sur un CD ou un film sur un DVD).



Par exemple, lorsque vous regardez une vidéo ou que vous écoutez de la musique en streaming, votre ordinateur télécharge et décode de tout petits morceaux du média en temps réel. Si vous avez une connexion internet performante, il peut arriver que vous ayez terminé de télécharger le fichier avant d'avoir fini de le lire, c'est pour cette raison que lors d'une coupure internet il se peut que votre stream continue quelques secondes/minutes avant de s'arrêter. Ceci étant dit, aucun fichier streamé ne se retrouvera sur votre ordinateur de façon permanente. Certains services de streaming, comme Spotify, enregistrent cependant de petits fichiers cache pour rendre le futur téléchargement du même fichier plus rapide.

3 – Son évolution dans la société

Au début des années 90, les internautes partageaient leurs fichiers vidéo en utilisant le « téléchargement traditionnel », c'est-à-dire en cliquant sur un lien internet pour télécharger le fichier en question. Par la suite, l'utilisateur devait patienter durant une longue période, car les modems étaient lents, avant de pouvoir réellement lire le fichier et afficher le contenu.

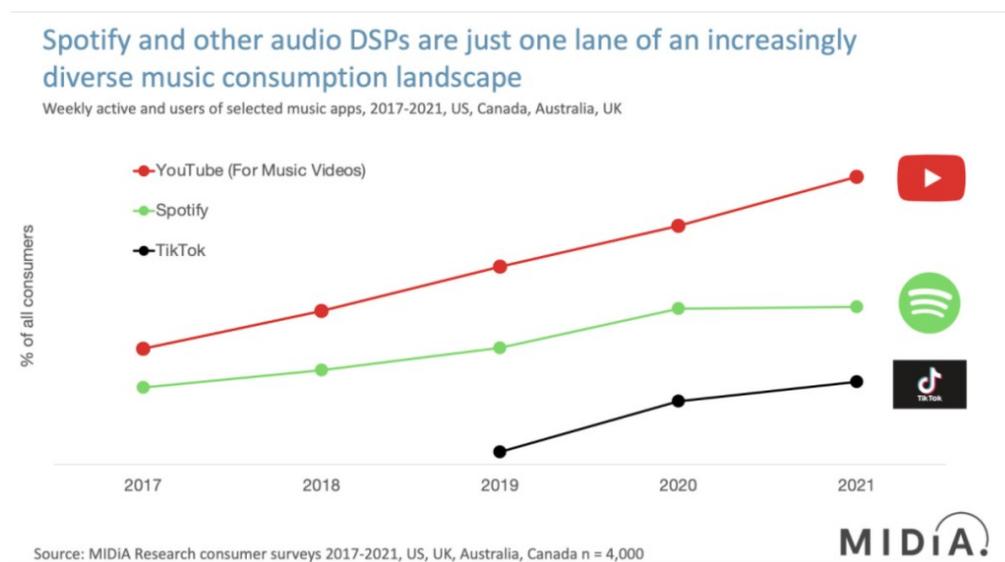
C'était de toute évidence un processus fastidieux et assez loin de la solution optimale pour les utilisateurs, les éditeurs et fournisseur de contenu, en raison de nombreux problèmes : longue attente avant d'avoir le fichier, fréquents problèmes de bande passante qui obligeaient l'utilisateur de relancer depuis le début son téléchargement ou encore, aucun contrôle sur le fichier vidéo une fois qu'il est remis à un consommateur.

Avec l'évolution de haut débit, et les fichiers vidéos qui devenaient de plus en plus volumineux il était nécessaire de trouver une solution pour faire gagner du temps aux utilisateurs. C'est pour ça qu'à partir de l'année 1994, la société Real Networks met en place la technologie du streaming, avec la première diffusion d'un audio en streaming en 1995, et la première diffusion d'une vidéo en streaming en 1997 avec le codec RealVideo.

A la fin des années 1990 et au début des années 2000, le streaming est principalement utilisé pour les retransmissions en direct d'événements sportifs.

Il faut attendre 2005, pour voir apparaître un site d'hébergement et de partage de vidéos, que l'on ne présente plus aujourd'hui, Youtube. Par la suite les volumes de connexion étant suffisants, une dernière innovation fait son entrée, en 2009 : le RTMP (Real Time Messaging Protocol), protocole propriétaire développé par Adobe Systems pour la diffusion de flux de données en streaming (audio, vidéo ou autre) entre un serveur et un client, généralement le lecteur Flash, dominant sur le marché jusqu'alors.

Depuis, les sites d'hébergements se sont multipliés, avec la rapidité des débits actuels et l'entrée de la fibre optique en France, le streaming ne peut que cesser de progresser vers des formats toujours plus rapides où la qualité de la vidéo sera toujours supérieure, avec une utilisation de plus en plus fréquente, surtout lors des dernières restrictions sanitaires où les utilisateurs ne trouver que ça comme seule occupation de leur journée dû aux confinements, comme nous les montres ces chiffres :



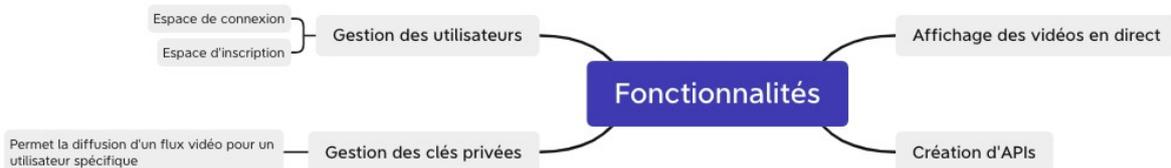
II – Cahiers des charges

Nous allons dès à présent rentrer dans le vif du sujet en commençant à créer notre propre plateforme de vidéos en direct à l'aide de technologies JavaScript dont nous verrons prochainement. Comme je suis une personne originale, je lui donnerai le nom de LHStream (comme Le Havre Stream) au projet.

1 – Fonctionnalités

Avant de réfléchir à l'ensemble des technologies que nous utiliseront, je vous propose de réfléchir tout d'abord à l'ensemble des fonctionnalités de notre plateforme.

Dans le contexte, nous chercherons à rester dans le simple, et vous propose donc ainsi les éléments suivants qu'inclura notre site internet:



C'est relativement court, mais pas si simple à mettre en place, ici il s'agira des fonctionnalités principales qui pourront être utilisées dans notre application.

2 – Choix des technologies

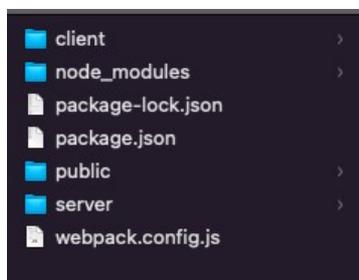
Pour ce projet, j'ai décidé de me baser sur des technologies JavaScript moderne, offrant ainsi de meilleures performances qu'un langage comme le PHP et pouvant inclure un plus grand nombre de données simultanément grâce à une base de données de type noSQL. Les bibliothèques JS principalement utilisées seront les suivantes :

VPS	
OS	Ubuntu 20.04
Langages/librairies	
Côté client	ReactJS, bootstrap (interface)
Côté serveur	Express.js (gestion des requêtes HTTP)
Base de données	MongoDB (base de données noSQL)

Il y aura bien évidemment d'autres modules qui interviendront dans notre projet, mais ici je ne fait qu'énumérer les principaux. Vous retrouverez l'ensemble des modules via le fichier package.json.

3 – Structure de l'arborescence

L'arborescence de notre projet sera la suivante :



En résumant, chaque répertoire correspondra aux caractéristiques suivantes :

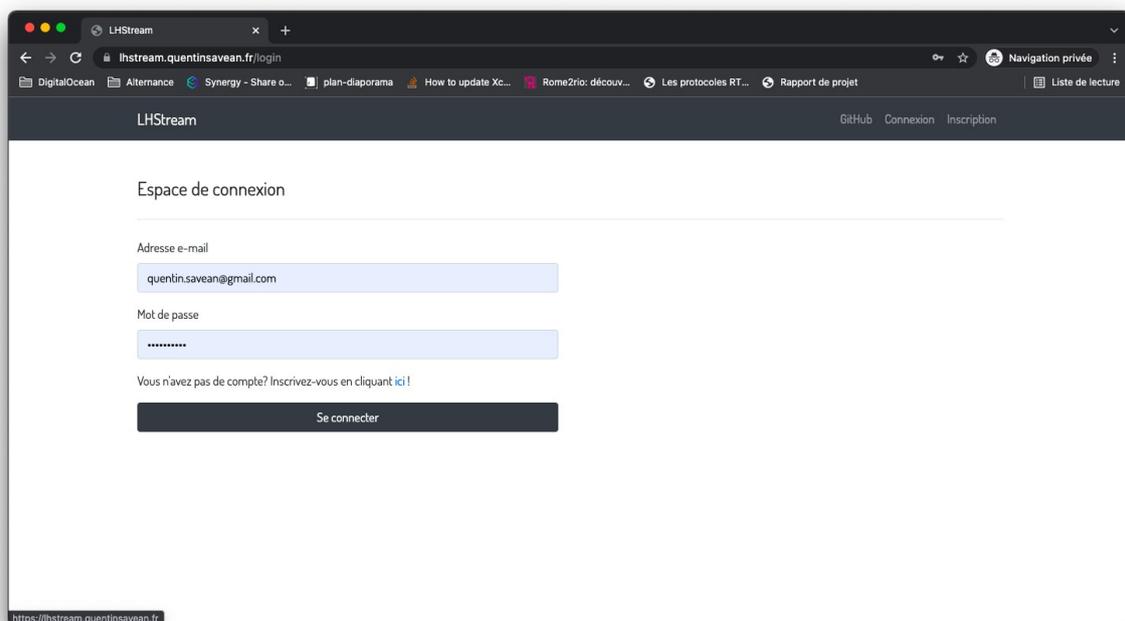
Client	Ensemble des fichiers statiques ReactJS de notre applications. C'est le côté client.
node_modules	Ensemble de nos modules node.js
Package-lock.json	Fichier généré par la commande d'installation des modules (npm i), agrège une version immuable du package.json, de façon à ce que l'on puisse, par exemple, aller chercher du vieux code
Package.json	Contient l'ensemble de la liste de nos modules node au format json (tableau)
Public	Fichiers publics accessibles par le navigateur (ex : fichiers css, bundle js...)
Server	Répertoire constituant l'ensemble des éléments de notre serveur (routes, apis, gestion du protocole...). C'est le côté serveur qui communiquera avec le client.
Webpack.config.js	Permet de compacter (build) nos ressources javascript ReactJS, pour que celles-ci soient visibles par l'utilisateur.

III – Comment utiliser la plateforme ?

Après avoir vu les éléments constituant notre projet, commençons à comprendre son utilisation.

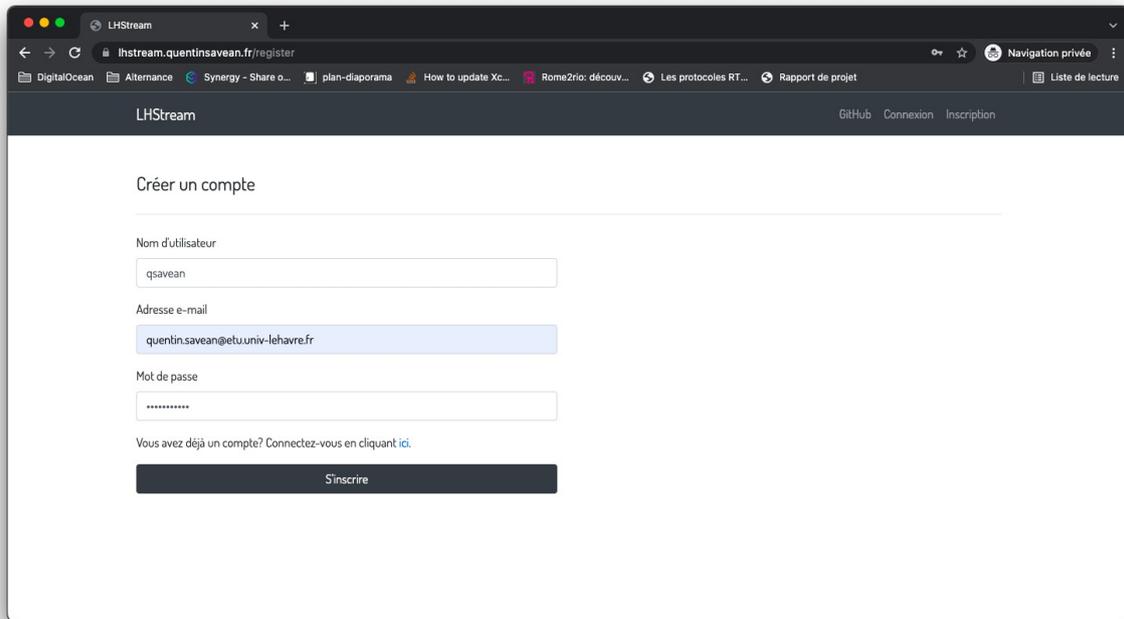
Étape 1 – Accéder à la plateforme

La plateforme est disponible via l'URL suivante : <https://lhstream.quentinsavean.fr>



Étape 2 – Créer vous un compte

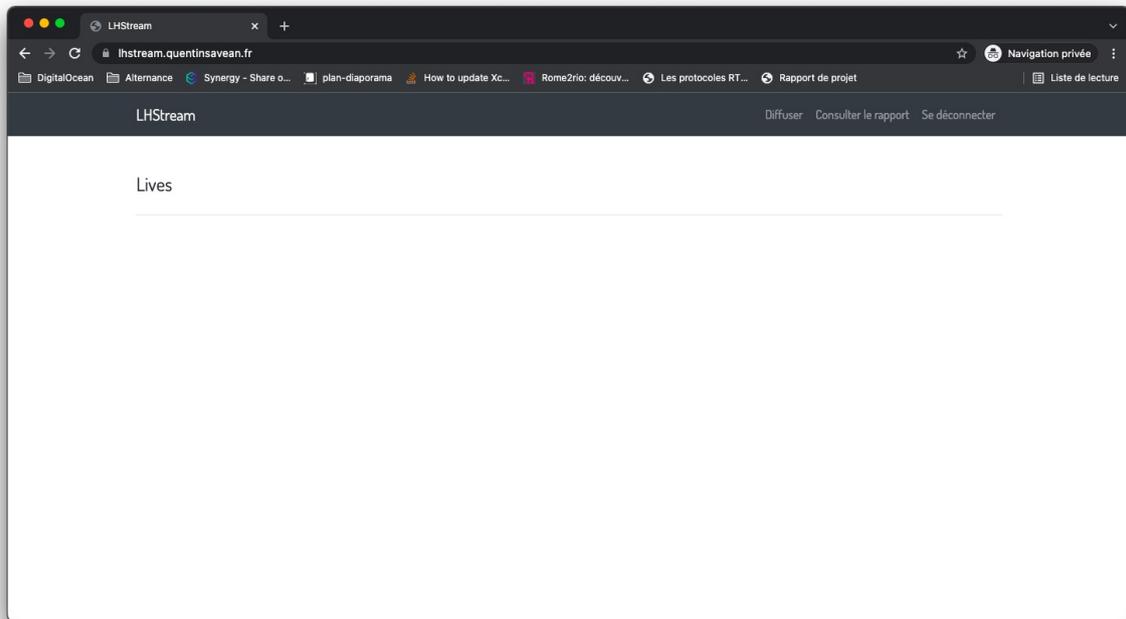
Si vous ne possédez pas de compte, il sera nécessaire d'en créer un via le formulaire d'inscription depuis le menu « [Inscription](#) ».

A screenshot of a web browser showing the registration page for LHStream. The browser's address bar shows the URL 'lhstream.quantinsavean.fr/register'. The page has a dark header with the LHStream logo and navigation links for 'GitHub', 'Connexion', and 'Inscription'. The main content area is titled 'Créer un compte' and contains three input fields: 'Nom d'utilisateur' with the value 'qsavean', 'Adresse e-mail' with the value 'quentin.savean@etu.univ-lehavre.fr', and 'Mot de passe' with a masked password '*****'. Below the fields is a link that says 'Vous avez déjà un compte? Connectez-vous en cliquant [ici](#).' and a dark 'S'inscrire' button.

Il sera nécessaire de rentrer :

- Un nom d'utilisateur (ex : qsavean)
- Une adresse e-mail (ex : quentin.savean@etu.univ-lehavre.fr)
- Un mot de passe (bien évidemment hashé dans la base de données grâce à bcrypt)

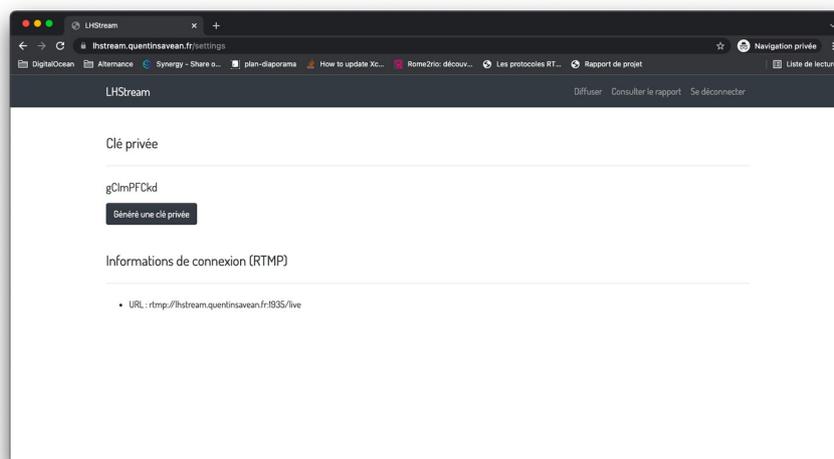
Une fois les différents champs remplis, cliquer sur le bouton s'inscrire.



Vous devriez être directement connecté par la suite avec une page qui ressemble à celle de ci-dessus.

Étape 3 – Récupérer vos informations de connexions RTMP

Via le menu « Diffuser », récupérer vos informations de connexions RTMP pour pouvoir diffuser votre flux vidéo dessus :



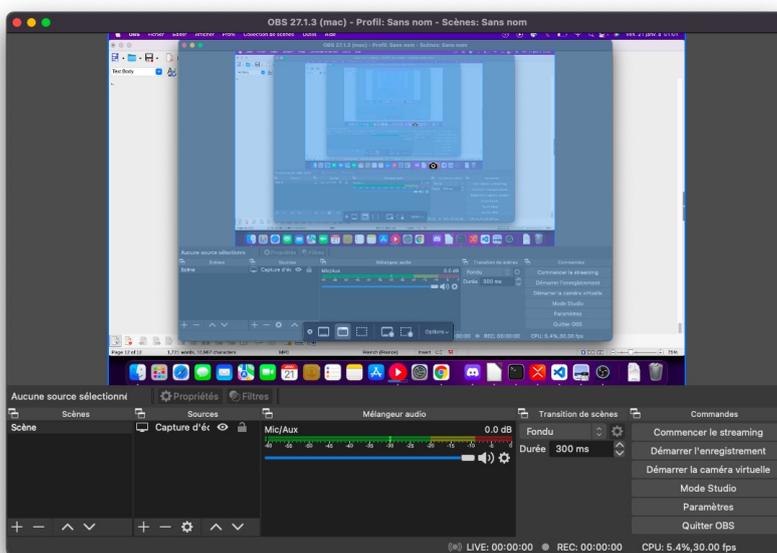
Récupérez votre clé privée (ici gCImPFCKd) et l'URL de connexion du serveur (rtmp://lhstream.quentinsavean.fr:1935/live)

Étape 4 – Diffuser un stream

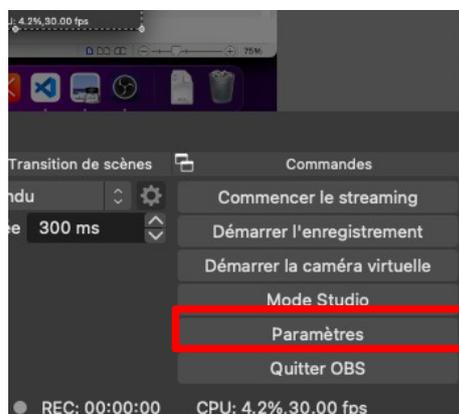
Une fois connecté, manque plus qu'avoir les outils nécessaires pour commencer à diffuser votre flux vidéo de votre machine vers la plateforme. Pour cela assurez-vous d'avoir le logiciel « [OBS](#) » sur votre PC afin de commencer à paramétrer le serveur RTMP dessus.

Vous retrouverez des explications concernant l'utilisation de ce logiciel **via le rapport pédagogique** depuis la section « **Lancement d'un stream** » via la partie concernant la « Mise en place d'un serveur simple RTMP ».

Nous avons donc cela :

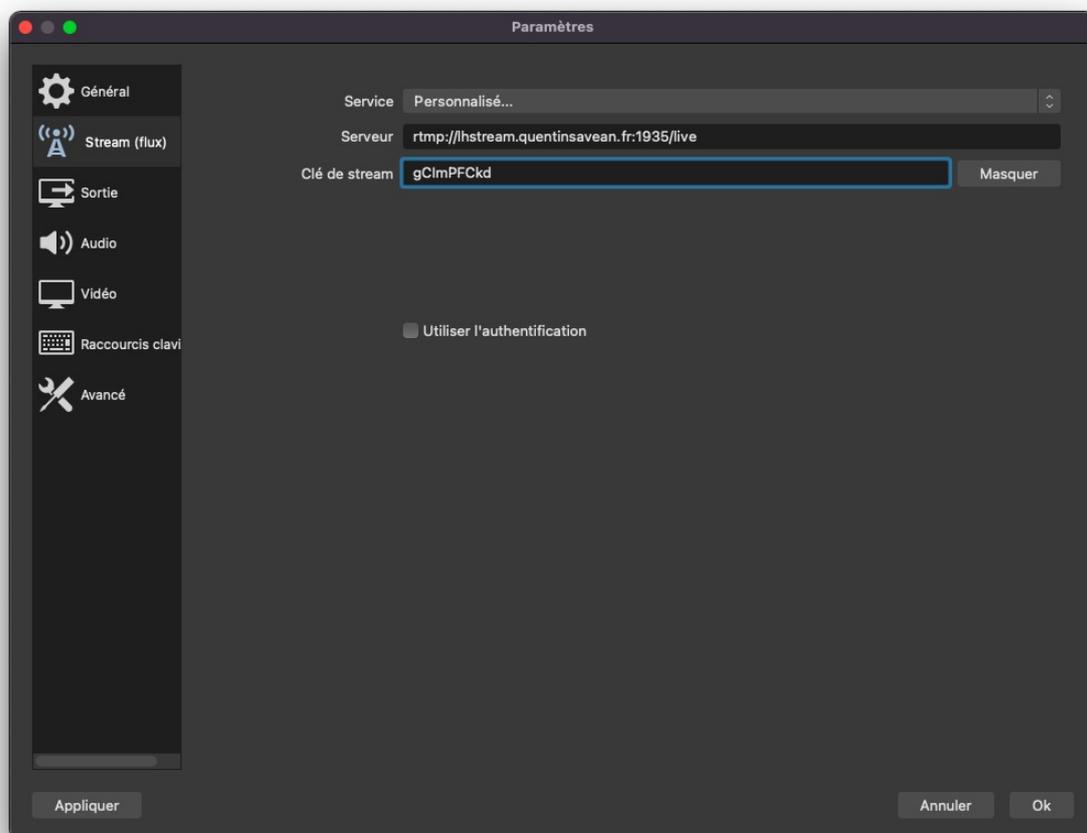


Cliquez sur le bouton « **Paramètres** », à droite, depuis le panel bas



Ensuite rentrez les informations précédentes, récupérées via le menu « [Diffuser](#) »:

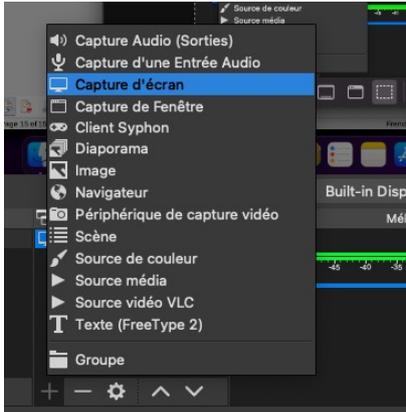
- Dans le champ « **Service** » : « Personnalisé... »
- Dans le champ « **Serveur** » : `rtmp://lstream.quentinsavean.fr:1935/live` (adresse du serveur RTMP)
- Dans le champ « **Clé de stream** »: Votre clé de stream disponible via le menu « [Diffuser](#) » de la plateforme. (ici `gCImPFCKd`)



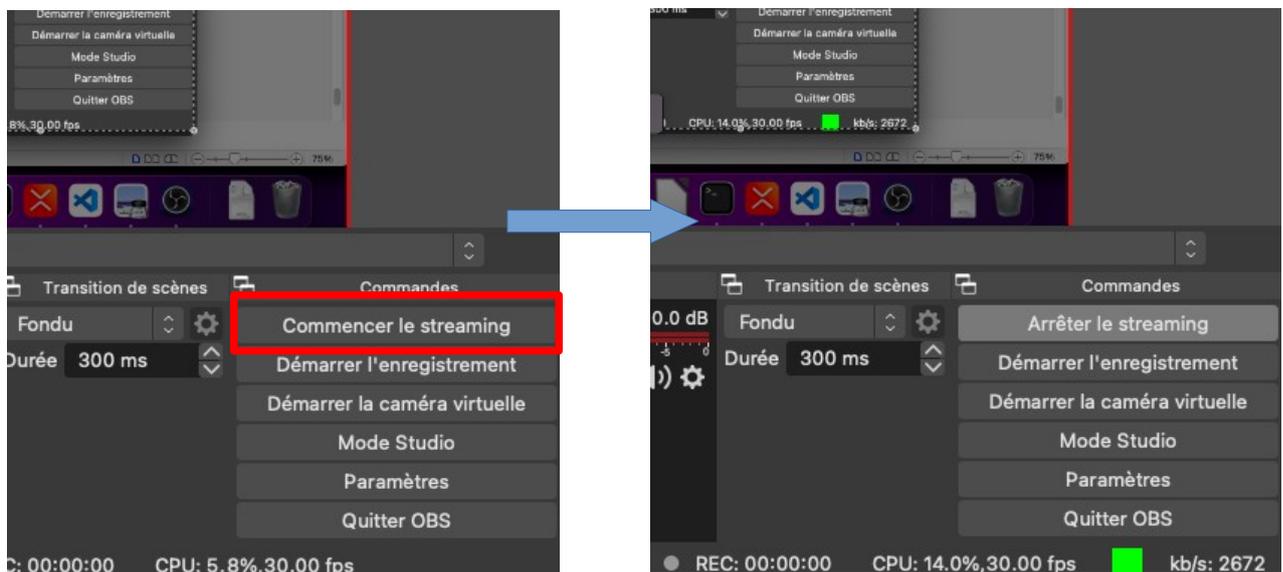
Ensuite cliquer sur le bouton « **OK** ».

Étape 5 - Démarrer la diffusion

Assurez-vous d'avoir ajouté une source vidéo avant de démarrer.



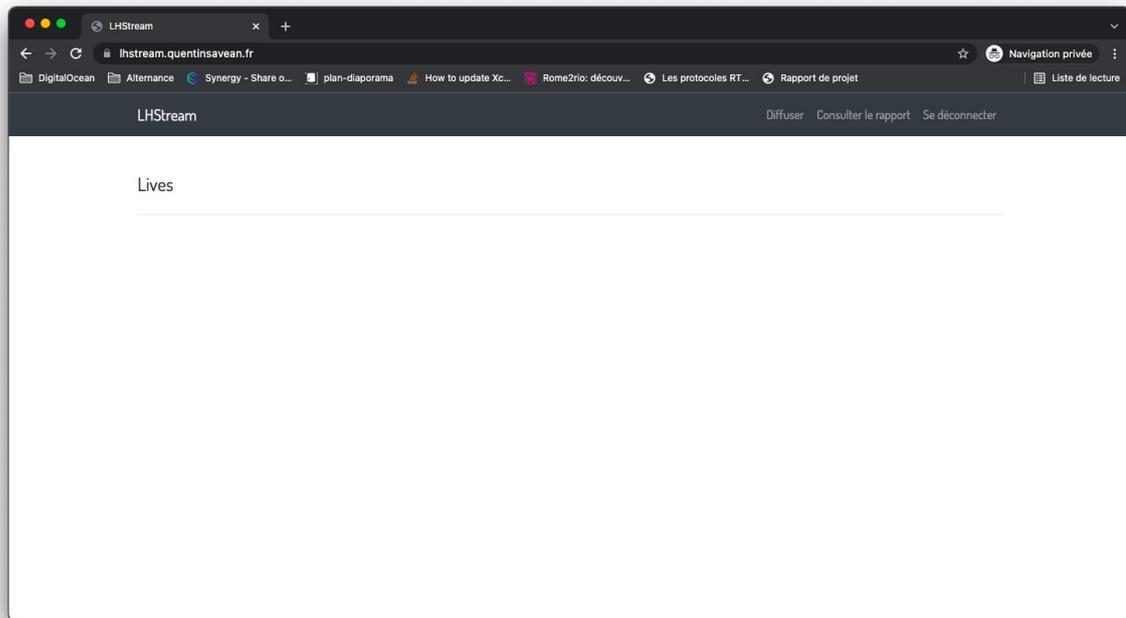
Une fois toutes les étapes précédentes faites, vous pouvez commencer à lancer votre diffusion :



Vous devriez avoir quelque chose comme cela.

Étape 6 - Visionner votre diffusion

Depuis LHStream, en vous assurant d'être toujours connecté, rendez-vous dans la page d'accueil de la plateforme :



Et là, magie ! Rien ne s'affiche...

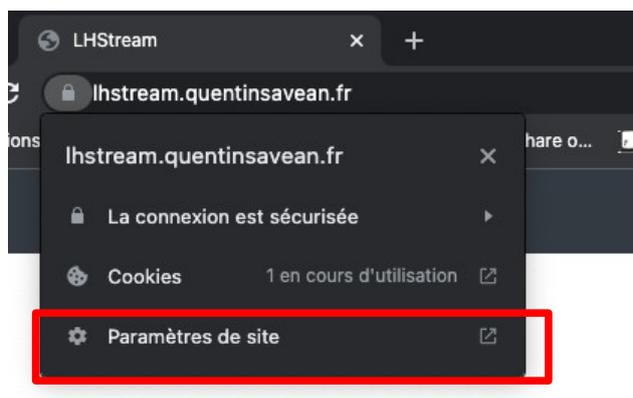
Pas d'inquiétude ! En réalité, il s'agit d'une erreur liée aux paramètres du protocole SSL (https) du serveur. En effet, le client (l'interface) et le serveur (pour la gestion des apis), utilise deux ports différents, le port 3333 pour le client et le port 8888 pour le serveur (côté apis). Or, une source https ne peut communiquer avec une source non sécurisé http, ce qui fait que l'accès aux apis est automatiquement bloqué par le navigateur (par défaut).

• Mixed Content: The page at 'https://lhstream.quentinsavean.fr/' was loaded over HTTPS, but requested an insecure XMLHttpRequest endpoint 'http://lhstream.quentinsavean.fr:8888/api/streams'. This request has been blocked; the content must be served over HTTPS. xhr.js:210

Pour pallier à ce problème, il aurait fallu héberger le serveur depuis imaginons un autre sous-domaine (ex : server.lhstream.quentinsavean.fr:8888) ou en créant un proxy virtuel nginx rédigeant vers le port 443 (port utilisé pour une connexion sécurisé à l'inverse du port 80), ce qui fait que celui-ci aurait détenu son propre certificat SSL ou redirigé vers l'existant, ce qui aurait permis une communication sécurisé entre le serveur et le client. Mais voilà, par manque de temps, je n'ai pas eu le temps de le faire.

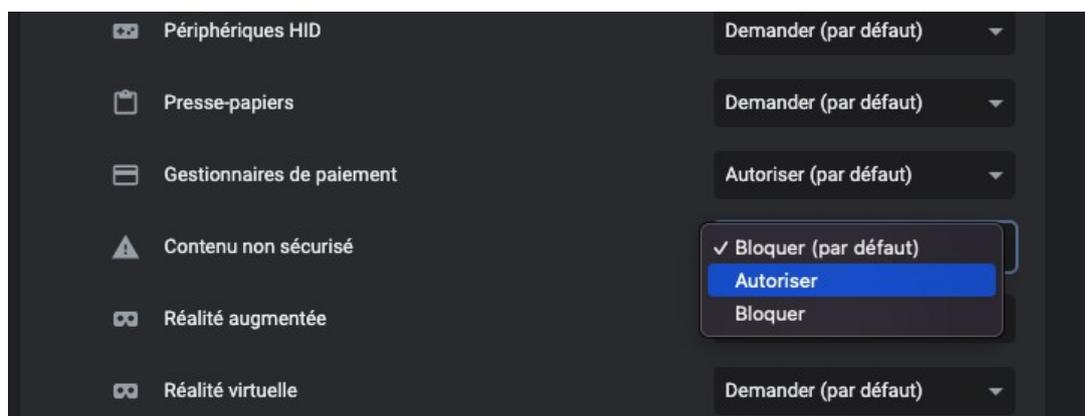
Cependant, il existe un autre moyen temporaire pour régler ce détail. Tout simplement en modifiant les paramètres du site, via votre navigateur, en autorisant notamment les contenus non sécurisés. Pour ma part j'utilise Google Chrome.

Depuis la barre d'URL, cliquez sur le cadenas situé à gauche du lien.

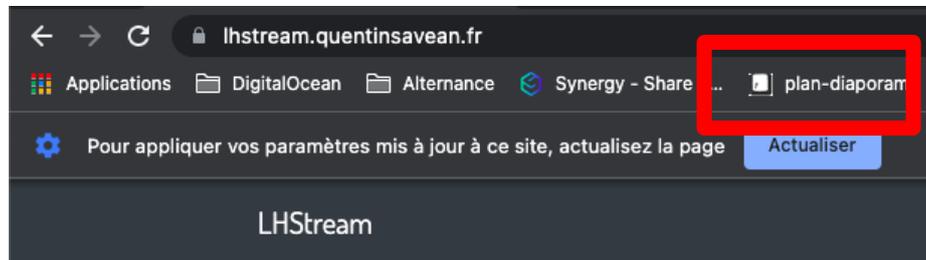


Vous verrez alors apparaître une petite fenêtre accompagnée de 3 menus. Cliquez sur le dernier « **Paramètres de site** ».

Descendez en bas de la liste, jusqu'à « **Contenu non sécurisé** », et depuis le sélecteur, choisissez « **Autoriser** ».

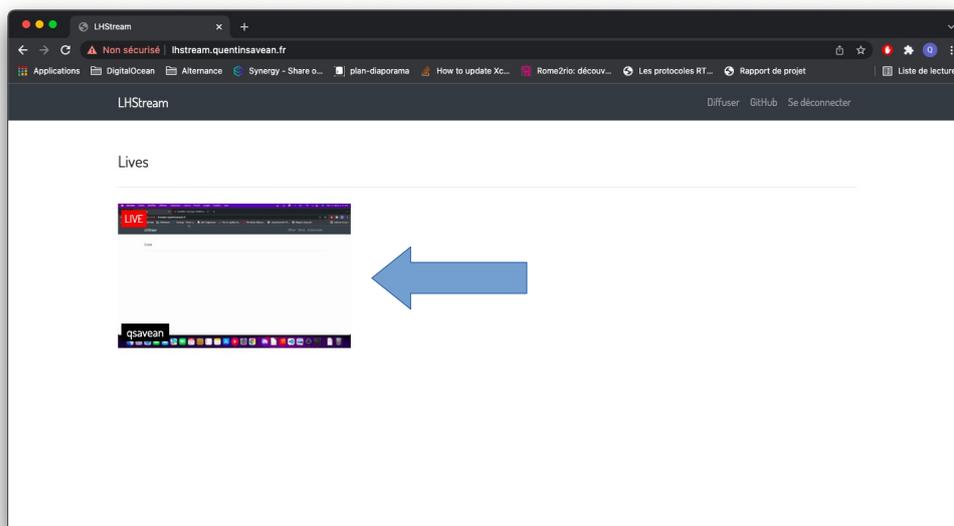


Retourner sur LHStream, et actualiser la page.

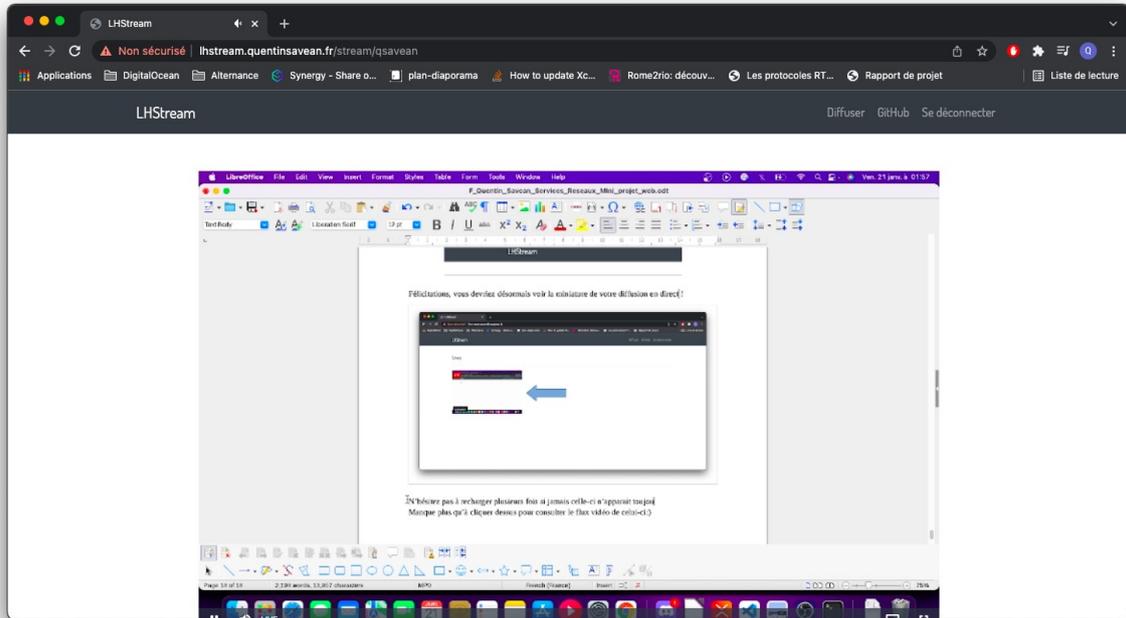


Félicitations, vous devriez désormais voir la miniature de votre diffusion en direct !

N'hésitez pas à recharger plusieurs fois la page si jamais celle-ci n'apparaît toujours pas.



Ensuite, manque plus qu'à cliquer dessus pour consulter votre flux vidéo :))



Voilà, c'est désormais la fin de ce tutoriel sur la mise en fonction de l'application. J'espère qu'elle vous aura plu:)

IV – Conclusion

Pour conclure, ce petit projet nous a permis de connaître un peu plus l'utilité du protocole RTMP et du monde du streaming à travers son histoire, ainsi que les différentes techniques utilisées pour mettre en marche un tel système.

Vous pourrez retrouver le code commenté, ligne par ligne, via mon profil GitHub depuis le lien suivant : <https://github.com/quentinsvn/lhstream> ou via le menu « GitHub » du projet.