

Funkwhale, les baleines mélomanes libres et décentralisées

Aujourd'hui nous laissons bien volontiers les commandes à Narf, une étudiante actuellement en stage chez Framasoft. Sa mission porte sur la vulgarisation des concepts de « **fédération** », notamment au travers de PeerTube, mais aussi plus largement du protocole ActivityPub. Afin de découvrir cet univers (ou ce [fediverse](#) ;)), nous lui avons proposé d'interviewer des personnes travaillant ou réfléchissant autour de cette problématique.

Comme beaucoup d'entre vous, je me pose de nombreuses questions sur l'avenir d'un internet où des géants monopolisent et centralisent l'espace public sans nous demander notre avis. La construction de nos propres médias sociaux semble être une belle manière de s'émanciper, tout en chérissant notre réseau social. Celui avec lequel nous aimons partager nos pensées, des bribes de nos joies ou tristesses, des vidéos qui nous ont fait rire, réfléchir, des musiques qui nous font frissonner. L'enjeu est de taille, vous ne trouvez pas ?

*Cependant, contrairement au modèle que les géants cherchent à nous imposer, il faudra que chaque personne puisse, si elle le souhaite, poser sa pierre pour [construire les villages](#) de ce nouvel univers appelé le fediverse. Pour ce faire, il ne suffit pas d'avoir une pierre à poser, il faut aussi pouvoir la tailler, savoir pourquoi et comment la poser. On sait bien ça chez Framasoft ! Lors de ces quelques mois que je vais passer dans l'asso, je vais apprendre sur les concepts de « **fédération** » et essayer de transmettre au maximum ce que j'aurai appris afin qu'ensemble nous rendions plus facile l'accès à cet univers !*



Avec le projet **Funkwhale**, Eliot Berriot participe, à sa manière, à la construction du fediverse. Nous avons voulu en savoir plus sur son parcours et sur ce projet : on apprend une nouvelle fois que technique et vision de société ne vont pas aller l'un sans l'autre.

Salut Eliot, est-ce que tu pourrais te présenter brièvement ?

Alors je m'appelle Eliot, j'ai 25 ans, et je suis tombé dans le développement de façon assez tardive / inhabituelle (je pense) : à la base, j'ai fait un bac L et des études pour devenir libraire ! Au cours de mes études supérieures, j'ai commencé à faire de la programmation en python pour le plaisir (sur le site du zéro, à l'époque), et petit à petit j'ai fait des sites web pour les amis, la famille etc. À tel point qu'à un moment, j'ai fini mon cursus (en licence pro) et je me suis établi à mon compte en tant que développeur web freelance. J'ai fait ça pendant 1 an ou 2, puis j'ai rejoint une SCOP, Au Fond À Gauche, où j'ai travaillé pendant plus de deux ans en tant que développeur / chef de projet technique. Après quoi, j'ai rejoint ma boîte actuelle, People Doc où je me trouve depuis Octobre 2017, toujours comme développeur.

Super intéressant ce parcours en auto-formation !

Complètement en autoformation, oui. C'est un truc qui me plaît énormément dans ce monde là : la possibilité d'apprendre à son

rythme, avec les ressources disponibles.

Nous, on t'a connu grâce à ton projet Funkwhale. Est-ce que tu pourrais le présenter aux lecteurs du Framablog ?

Le concept de Funkwhale est assez similaire à Grooveshark, d'où le nom d'ailleurs. Pour ceux qui ne connaissent pas, Grooveshark était un service de streaming musical, un peu comme Deezer ou Spotify, très axé sur les interactions entre utilisateurs, le côté social. L'expérience utilisateur était vraiment bonne, avec un player web de qualité, et la possibilité d'écouter de la musique pendant des heures sans pub. Le service a été fermé il y a quelques années suite à des soucis avec les ayants-droit.



Le logo de Funkwhale

Il y a cependant des différences qui sont : le développement sous licence libre, la possibilité d'installer et de gérer son instance Funkwhale sur son serveur, en autonomie, comme avec Mastodon, par exemple. Et bien entendu, on n'est pas encore à parité fonctionnelle avec Grooveshark. Il y a certaines fonctionnalités qui étaient présentes (et aussi sur d'autres services de streaming, j'imagine), qui ne sont pas encore réimplémentées dans Funkwhale. Par exemple, les broadcast, une fonctionnalité que j'aimais beaucoup : un utilisateur peut live-streamer des musiques de son choix et ceux qui le

souhaitent peuvent se connecter sur ce stream et écouter la même musique, au même moment un peu comme une radio. Ça permet de partager et de découvrir de la musique sur un mode très fun.

Effectivement, c'est sympa comme fonctionnalité, ça met plus d'interaction dans le partage. Ça fait sens si tu as envie de créer une vraie communauté autour de la musique !

Tout à fait : chaque broadcast avait aussi un chat, et les utilisateurs pouvaient suggérer des chansons à jouer ensuite, c'était très participatif. Je crois même que certaines personnes faisaient des broadcasts à temps plein et en tiraient des revenus.

Qu'est ce qui t'as motivé à faire ce projet ?

C'est le résultat d'un parcours perso assez long. En parallèle de ma découverte du développement, je me suis beaucoup intéressé aux questions d'auto-hébergement. J'essaie depuis environ 5 ans d'héberger le plus possible mes données et de ne pas me reposer sur des services tiers / fermés (pour les mails, la synchro de fichiers, etc). Quand les solutions existantes ne me satisfaisaient pas, il m'est arrivé de développer des outils perso (par exemple, mon premier gros projet web était un lecteur de flux RSS / moteur de blog). A un moment, Grooveshark que j'utilisais beaucoup a fermé, et j'ai commencé à me dire « tiens, et ça, si je l'hébergeais ? ». Malheureusement, les solutions existantes ne me satisfaisaient pas, principalement pour deux raisons :

- Soit je les trouvais peu agréables d'utilisation,
- Soit elles étaient mono-utilisateur, alors que je voulais quelque chose qui puisse permettre à des personnes de se retrouver autour de la musique, comme sur Grooveshark.

Du coup, j'ai retroussé mes manches, et j'ai commencé à travailler sur Funkwhale. Ça remonte à deux ans et demi /

trois ans, à peu près. La première version était assez... pas terrible, disons-le. Mais ça marchait quand même, je pouvais importer mes artistes préférés et jouer ma musique sans dépendre de personne et sans pub, et ça, c'était cool ! Ensuite l'année dernière, j'ai entrepris une réécriture complète de l'interface (qui était ce qui posait le plus problème), pour arriver en gros sur ce que l'on a maintenant. Comme ça devenait utilisable et assez riche fonctionnellement, j'ai commencé à en parler sur Mastodon, en début d'année, et puis cela a pas mal pris ☐

Qu'est-ce qui différencie Funkwhale d'un autre site de partage comme Soundcloud ?

N'utilisant pas Soundcloud, j'aurais un peu plus de mal à être juste je pense. Mais pour ce que j'en ai vu, Soundcloud semble s'adresser plutôt aux personnes qui créent de la musique. Grooveshark (et Funkwhale) sont plus axés « auditeur » (du moins pour le moment :D)

Mais si les utilisateurs ne sont pas, a priori, « créateurs de contenu », comment ça se passe pour les droits d'auteurs ? C'est quand même ce qui a fait fermer Grooveshark. Qui est-ce qui va gérer ça ?

Je l'attendais, celle là ! Il faut évidemment aborder ces questions, d'autant que ça peut effrayer certaines personnes ! Tout d'abord, il faut bien distinguer Funkwhale en tant que logiciel, ce qui permet de faire tourner une instance, d'une instance Funkwhale, qui est un serveur qui fait tourner le logiciel Funkwhale. C'est exactement le même principe que PeerTube. Funkwhale, en soi, c'est un logiciel qui permet d'importer de la musique et de la mettre à disposition des utilisateurs d'une instance (ou potentiellement d'autres instances, avec la fédération). Aucune musique n'est livrée avec, donc le projet en lui même est une coquille vide.

Ce sont les personnes qui gèrent une instance et qui vont

mettre en ligne de la musique qui sont responsables du volet « respect de la propriété intellectuelle ». D'ailleurs, dans son mode par défaut, une instance Funkwhale et la musique qu'elle contient sont uniquement accessibles aux personnes inscrites sur l'instance (et les inscriptions sont fermées par défaut).

Du point de vue de la loi, à ma connaissance, il n'y a pas d'interdiction à mettre en ligne de la musique que l'on a achetée légalement, pour pouvoir l'écouter sur d'autres supports / machines. Ce qui est réprimé, c'est le partage hors du cercle familial amical, par exemple avec des inconnus via torrent. Héberger une instance Funkwhale pour les copains et/ou la famille me semble (mais je ne suis pas juriste) globalement dans les clous. C'est une plateforme d'hébergement de contenus donc les personnes gérant la plateforme auront à répondre des infractions s'il y en a, si leur plateforme est publique et accessible à n'importe qui.

Tu parles d'un cercle familial mais la fédération, que tu as récemment mise en place, ne rentre plus dans ce cadre, si ?

La question se pose, oui. Depuis peu, Funkwhale permet de fédérer les catalogues musicaux de différentes instances pour les rendre accessibles à d'autres instances. Pour essayer de limiter les problèmes potentiels et leurs conséquences, sur Funkwhale, la fédération se fait par défaut sur un mode très restrictif. Les catalogues musicaux des instances ne sont pas accessibles sans autorisation. Donc une instance A doit demander l'accord d'une instance B pour accéder à son catalogue. Cela permet de révoquer l'accès en cas de besoin, par exemple. D'autre part, par défaut, les fichiers musicaux ne sont pas répliqués d'une instance à l'autre (hors pour du cache sur une période assez courte). Ainsi, on pourrait imaginer avoir des grosses instances hébergeant de la musique libre partageant leurs contenus avec de petites instances possédant du contenu protégé.

Ouh là là, certains lecteurs assidus du Framablog t'ont

sûrement suivi sur ce coup mais, pour les autres, pourrais-tu expliquer simplement ce qu'est la fédération ?

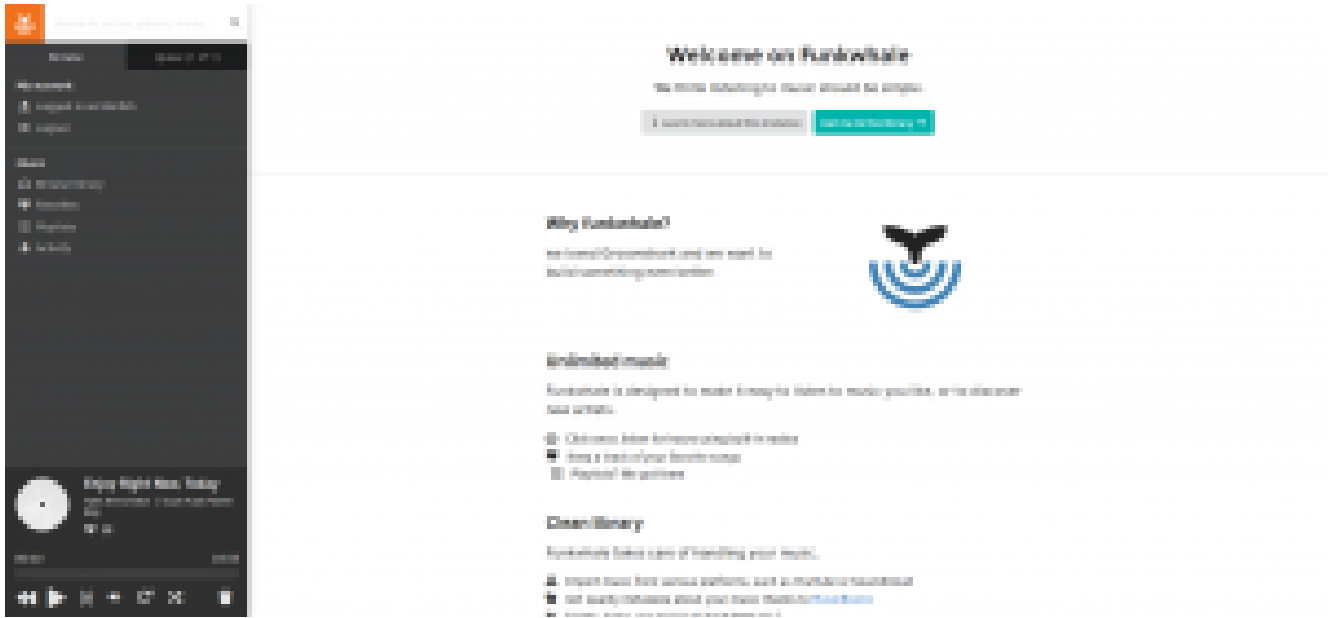
Alors la fédération, tel que je le conçois, c'est un mécanisme qui permet à différents intervenants sur un réseau de s'échanger des messages, de se comprendre. L'intérêt de la fédération, c'est également de réduire le développement de SPOF (*single point of failure*, les endroits où si ça pète, tout pète) puisque dans un réseau fédéré, la chute d'un acteur n'affecte pas les autres acteurs outre mesure. Ainsi, si le fournisseur e-mail d'un·e de vos amis ne fonctionne plus et que vous êtes chez un autre fournisseur, vous pouvez continuer à lire vos mails, car l'e-mail fonctionne sur un mode décentralisé et fédéré.

À l'inverse, sur des services centralisés, en silos, comme YouTube ou Spotify, si le service devient indisponible ou disparaît, plus aucun utilisateur ne peut en bénéficier. C'est ce qui fait que Funkwhale ne disparaîtra jamais à la manière de Grooveshark : si l'on ferme une instance, les autres continueront de fonctionner et, même si l'on fait disparaître le projet et les sources, les instances existantes continueront de fonctionner. Là encore, on est sur un concept assez proche de PeerTube, dont le fonctionnement parle probablement plus aux lecteurs et lectrices de ce blog.

Pour une personne non initiée pourrait-on simplifier en disant : une instance = Funkwhale auto-hébergé / les messages = la musique que l'on met sur Funkwhale ?

À l'heure actuelle plus ou moins oui. En fait une instance va notifier ses « followers » (les instances qui ont accès à son catalogue) quand elle importe de la musique. Celles-ci, en fonction de leurs paramètres, vont pouvoir la rendre accessible immédiatement et en temps réel dans leur catalogue. Mais à terme, la fédération ira plus loin, et inclura aussi le contenu des utilisateurs. Typiquement, si tu crées une playlist sur ton instance A et que je te follow sur mon

instance B, je pourrai écouter ta playlist ou bien la partager avec mes followers. Je pourrais également envoyer un message à un·e ami·e sur une instance C pour lui conseiller d'écouter telle ou telle chanson.



Interface officielle de Funkwhale

Est-ce que tu avais développé Funkwhale en ayant déjà la fédération derrière la tête ?

En fait je voulais que les instances communiquent, mais je n'avais aucune idée du comment. Très concrètement, c'est grâce à mon année passée sur Mastodon et aux discussions que j'ai vu passer régulièrement sur ActivityPub que je me suis rendu compte que c'était faisable. Auparavant, je voyais ça comme une fonctionnalité extrêmement complexe, ça me refroidissait beaucoup.

Est-ce que tu peux expliquer aux lecteurs du framablog ce qu'est ActivityPub ? À quoi ça sert, comment ça marche ?

Alors tout à l'heure je disais que la fédération, c'était le mécanisme qui permettait à des acteurs d'un réseau de s'échanger des messages. ActivityPub, qui est maintenant un standard du W3C, c'est un protocole qui permet de faire de la fédération, et qui définit notamment la structure des

messages, leur contenu possible, l'endroit où il faut les envoyer, etc. Ça permet à ceux qui l'implémentent [les différents logiciels] de parler la même langue et donc de communiquer. Une des grandes forces du protocole, c'est qu'il réutilise des choses qui existaient déjà (HTTP pour l'envoi des messages, Json-LD pour la structure, Activity Streams pour les vocabulaires, c'est-à-dire le contenu des messages). Il rajoute finalement assez peu de choses ce qui permet de réutiliser plein d'outils déjà existants.

Pourquoi as-tu choisi ce protocole ?

Je l'ai choisi principalement parce que je ne suis pas hyper calé sur la question, qu'il était relativement récent, et qu'il permettait potentiellement de communiquer avec le *fediverse*. On peut imaginer que si un jour Mastodon ou Pleroma intègrent un player audio, un utilisateur de Funkwhale puisse recommander de la musique à un utilisateur de Mastodon, et que celui-ci pourra lire la musique en question directement dans Mastodon.

Tu dis que tu n'es pas hyper calé sur la question mais tu l'as quand même implémenté rapidement ! Quels sont tes retours pour l'instant sur la fédération des différentes instances Funkwhale ?

Les retours sont assez positifs, il y eu quelques couacs et bugs dus à des problèmes de tuyauterie, mais sinon ça fonctionne, ce qui fait plaisir. Ce qui manque principalement, à mon avis, ce sont des instances avec lesquelles fédérer mais il y en a régulièrement de nouvelles, donc le problème va disparaître avec le temps !

Tu m'as dit être émerveillé que ce projet intéresse tant de monde : on pourrait dire que le projet prend bien, alors ? Comment cela se fait d'après toi ?

Est-ce que ça prend bien ? Je dirais que oui : le chan matrix (#funkwhale:matrix.org) est assez vivant, il y a régulièrement

des nouvelles idées, demandes, et des instances qui ouvrent (il doit y en avoir une petite dizaine je dirai). J'ai également quelques personnes qui contribuent sur des fonctionnalités spécifiques. J'espère arriver à réduire la difficulté d'accès pour les contributions externes, parce que le projet étant quand même assez complexe, ce n'est pas toujours évident. En ce moment je rédige beaucoup de documentation (installation, contribution, etc).

Pour moi, si le projet prend bien, c'est d'une part parce que le public auquel je me suis adressé jusqu'à maintenant (sur Mastodon) est assez sensible aux questions d'auto-hébergement, de décentralisation. C'est aussi grâce à la hype, à mon avis tout à fait justifiée, autour de PeerTube. Il y a deux mois, avant l'arrivée de la fédération dans Funkwhale, la question de la fédération était quasi systématique : « est-ce que Funkwhale peut se fédérer comme PeerTube ? ». J'ai passé beaucoup de temps sur cette fonctionnalité, parce que j'ai vu qu'il y avait une vraie demande et le fait d'y répondre en partie a contribué à susciter l'intérêt je crois ☐

Pour les technos qui nous lisent : quels sont les conseils que tu pourrais donner à d'autres développeurs qui souhaiteraient implémenter de la fédération dans leurs logiciels ?

Aux personnes qui veulent se lancer dans de la fédération je dirai :

1. Définissez bien en amont ce que vous voulez, et commencez par quelque chose de petit, dans le doute,
2. N'ayez pas peur de demander de l'aide, car il n'y a pas encore beaucoup de ressources sur le sujet (sur Mastodon, il y a des tas de personnes prêtes à vous répondre dont moi-même).

Le 1. est important car c'est beaucoup plus dur de faire évoluer une fédération une fois qu'elle est lancée.

Quand tu conseilles de commencer par quelque chose de petit,

tu parles de ce qui est fédéré ?

Je veux dire petit sur le plan fonctionnel. Par exemple, pas la peine d'essayer de ré-implementer Mastodon du premier coup. Mais peut-être juste une fonctionnalité qui permette de favoriser un toot du fediverse et de notifier son auteur que l'on a favorisé. Cela évite de se confronter à tous les problèmes d'un coup. Quand j'ai commencé sur la fédération, j'ai juste fait un bot coté Funkwhale qui pouvait recevoir des follows et répondre « pong » quand on lui envoyait le message « /ping ».

Tu as utilisé la doc de W3C ou tu considères que c'est vraiment en échangeant sur Mastodon que tu as pu avancer ?

Non, la spécification m'a énormément servi ! Le principal souci, c'est que la partie « gestion de l'authentification et des autorisations » n'est absolument pas spécifiée, donc là il a fallu aller creuser dans le code de Mastodon / PeerTube.

Est-ce que c'est important pour toi que le logiciel que tu as développé soit libre ?

C'est une question très importante pour moi. Je me suis fixé comme contrainte avec Funkwhale et depuis le début de ne bosser qu'avec des composants libres, et autant que possible, auto-hébergés. Ceci est valable tant pour la partie développement pure (le projet en lui même utilise des langages / bibliothèques libres), que pour la partie infrastructure autour du projet, qui est souvent complètement oubliée quand on pense « logiciel libre ». Ainsi, Funkwhale est développé sur mon Gitlab perso (en attendant un Gitlab dédié au projet ?), je communique dessus exclusivement sur des réseaux libres (Matrix, Mastodon...), l'intégration continue est également sur mes serveurs perso, etc. J'essaie de réfléchir, mais je ne vois pas une seule brique non-libre ou centralisée qui intervient dans le processus de développement du projet ou dans son fonctionnement au sens large.

Le principal souci que je vois avec ce montage, c'est que Funkwhale est du coup très lié à mon infrastructure d'auto-hébergement perso. Si le projet continue, il faudra mettre en place une structure administrative et des outils et moyens dédiés au projet je pense. Quant au projet lui-même, il est sous licence BSD, qui est une licence extrêmement permissive et que j'ai choisie un peu au hasard. J'avoue ne pas être très calé sur la question des licences et je suis ouvert à l'idée d'utiliser une autre licence plus adaptée au projet et à sa pérennité s'il le faut.

Pourquoi t'être donné toutes ces contraintes ?

Des raisons de placer le projet sous licence libre, j'en vois plein ! En vrac :

- m'assurer que le projet puisse continuer même si j'arrête ou que je disparaiss,
- permettre à d'autres personnes de participer et d'enrichir Funkwhale, ce dont je profiterai directement, étant utilisateur de l'outil et qui ne serait pas possible sur un modèle fermé,
- rendre dans une certaine mesure une petite partie de ce que j'ai « pris » à la communauté. J'utilise Seafile, Mailcow, Firefox, Debian, et des centaines d'autres logiciels libres, chaque jour. Je ne contribue pas forcément à ces logiciels, mais si je peux enrichir ce patrimoine commun avec quelque chose qui répond à un besoin tant mieux !
- cela m'a servi au cours de ma vie pro, pour montrer ce sur quoi je bosse facilement, pour que des personnes puissent lire du code que j'ai écrit

Et sur l'utilisation du libre dans le cadre du projet, je dirai que c'est par esprit de cohérence. Je me sens toujours très mal à l'aise de contribuer sur des projets libres sur Github, par exemple. Je n'ai rien contre Github en

particulier, c'est une entreprise qui fournit un service de qualité, mais c'est devenu un SPOF énorme. Si Github tombe un jour, pour une raison ou une autre, le logiciel libre prendra un sacré coup ! Idem, je ne me vois pas faire un compte Twitter pour Funkwhale. Enrichir des publicitaires ? Très peu pour moi ! C'est aussi un moyen de garder le contrôle sur les moyens de production. Si je n'aime pas quelque chose dans Gitlab, je peux forker Gitlab, ou contribuer au projet pour le faire évoluer dans un sens qui me convient. Ce n'est tout simplement pas possible avec Github.

J'applique le même raisonnement pour les contributions financières. J'aurais pu ouvrir une page Patreon, par exemple, mais j'aurais été constamment tributaire de leurs décisions, de leur business-model etc. Avec Liberapay, je sais que je peux participer à l'évolution du projet, que ça ne sera pas racheté du jour au lendemain pour faire des sous.

Tout ça, ça donne de la tranquillité d'esprit, permet de travailler d'une façon plus posée, en accord avec ses principes. C'est important pour moi, même si du coup cela prend un certain temps pour mettre les pièces du puzzle en place, trouver et configurer des alternatives, etc. Mais je crois qu'il faut plus le voir comme un investissement que comme un coût. C'est une vision de la société, des interactions humaines et de ce qu'elles pourraient être. J'ai envie de participer à l'émergence de ça, une société qui soit basée sur le don, la bonne volonté, le positif en fait. Pour moi, cela passe notamment par le fait de développer Funkwhale sur ce mode, et d'utiliser Funkwhale comme levier pour mettre en valeur / utiliser d'autres projets qui fonctionnent sur ce même mode, pour créer un écosystème.

D'ailleurs, c'est développé en quel langage ?

Funkwhale est un projet découpé en deux composants : la partie API, autrement dit le back-end, qui gère les données, la musique, etc, et le front-end, qui est l'interface officielle

du projet. Ce découpage complique certaines choses, mais a aussi énormément d'avantages. Premièrement, il est totalement possible d'utiliser Funkwhale avec d'autres clients (Desktop, Android, etc.), ou même avec une interface web différente de l'interface fournie avec le projet.

Quand on parle de contribuer à Funkwhale, on va donc parler généralement de contributions à l'un ou l'autre de ces composants. Le back-end est écrit en Python 3 avec Django (c'est un framework web sur lequel je suis très à l'aise et productif). Le front-end est écrit en VueJS, un framework Javascript assez simple à prendre en main mais très puissant. Et j'utilise la bibliothèque Semantic UI pour les styles, car je la trouve très complète et jolie (c'est un peu le même principe que Bootstrap, ça fournit des composants et des outils pour obtenir une interface cohérente et fonctionnelle).

Le fait que Funkwhale soit séparé en plusieurs composants permet donc aux personnes voulant contribuer de participer sur le volet qui les intéresse. Jusqu'à maintenant, c'est la partie front qui a reçu le plus de contributions externes, avec par exemple un travail qui a été mené par Baptiste sur l'internationalisation et le design de la sidebar. Une autre personne a récemment contribué à la rédaction de la documentation pour permettre l'installation de Funkwhale avec le serveur web Apache. Je sais qu'une autre personne a également commencé à travailler sur la prise en charge du format Flac. Je demeure toujours le principal contributeur du projet, comme tu peux le voir [ici](#). Néanmoins, j'espère que cela bougera dans les mois à venir, avec l'arrivée de nouveaux utilisateurs et utilisatrices et l'amélioration de la documentation.

Tu dis que l'auto-hébergement t'es venu par l'envie de reconstruire tes données : cette préoccupation semble traverser de plus en plus de personnes en ce moment. Cependant,

installer Funkwhale sur son serveur, ça ne s'adresse pas à tout le monde, si ?

Effectivement il faut avoir un minimum de compétence et de curiosité pour installer une instance Funkwhale (ou PeerTube, ou Mastodon...). Ce n'est donc pas à la portée de tout le monde, mais cela reste malgré tout à la portée de beaucoup de personnes, je pense. D'autre part, chacun n'a pas besoin d'avoir *son* instance. Funkwhale étant multi-utilisateur, on peut envisager des déploiements par famille, par collectif, par CHATONS...

Enfin, une des grandes nouveautés de ces dernières années, c'est à mon avis Docker qui réduit grandement les difficultés à installer un service tel que Funkwhale. Tout le monde n'est pas forcément convaincu par cette techno, qui a aussi ses problèmes, mais la simplicité pour les déploiements est quand même un atout assez fort. Très concrètement, si tu consultes la [doc d'installation de Funkwhale sur Docker](#), tu pourras constater qu'il suffit d'une dizaine de commandes à exécuter pour installer Funkwhale sur son serveur.

Ceci étant, oui, il y a un travail de pédagogie à faire pour rendre l'installation de ce genre de services plus simple, moins effrayante. Et aussi du boulot à faire pour intégrer ça avec les systèmes de packaging existant (Yunohost, Cloudron, etc.). D'ailleurs, un dernier mot sur l'auto-hébergement, mais un utilisateur a réussi à installer et à utiliser Funkwhale sur Raspberry Pi ! Je pense que c'est un bon indicateur du fait qu'une instance Funkwhale peut tourner sur des systèmes pas forcément hyper puissants, donc avec un coût réduit.

Est-ce qu'il existe un endroit où l'on peut être mis en relation avec des personnes qui hébergent leur instance (un peu comme joinmastodon.org ou joinpeertube.org) ?

Oui, c'est par [ici](#).



Join Funkwhale

Comment est-ce que tu arrives à t'en sortir avec ce projet et ton boulot ?

En termes d'organisation, c'est vrai que c'est assez chronophage. En trois ans, je pense que j'ai consacré plusieurs centaines d'heures à ce projet, probablement aux alentours de 500 ou 600. Cela se fait sur mon temps libre, puisque je suis également salarié à temps plein. Fort heureusement, c'est avant tout un plaisir pour moi de faire évoluer cet outil et de l'utiliser au quotidien. C'est un travail entièrement bénévole, même si j'ai récemment ouvert une page Liberapay sur laquelle les contributeurs et contributrices du projet peuvent recevoir des dons.

À moyen terme, si je reçois suffisamment de dons via ce biais, j'envisage de réduire mon temps de travail salarié pour consacrer plus de temps au projet.

Autrement, un message à faire passer aux personnes intéressées par le projet et qui voudrait y contribuer ?

Oui : les contributions peuvent être de toutes sortes, et pas seulement financières ! Corriger une erreur typographique, remonter un bug, faire le tri dans les tickets, commenter des

discussions pour apporter des compléments d'information, ce sont des contributions valables qui vont faire avancer le projet. Le code ne fait pas tout, loin de là. Le simple fait de parler du projet, comme tu le fais, ou de dire merci, c'est également une contribution qui booste le moral et la motivation, attire d'autres contributeurs, etc. Bref c'est un cercle vertueux !

Et si une personne est intéressée pour contribuer au code, [je maintiens également une liste d'issues « faciles »](#) sur le Gitlab. Ce sont des issues faciles pour une première contribution, pour découvrir le projet. Je n'ai pas spécialement d'expérience en gestion de projet libre, Funkwhale est mon premier projet avec cette ampleur, donc je découvre chaque jour, j'essaie d'améliorer ce qui peut l'être pour réduire la barrière à l'entrée pour de nouvelles personnes.

Tu me disais qu'on pourrait te retrouver aux Rencontres Mondiales du Logiciel Libre (RMLL) ?

Tout à fait ! Suite aux nombreux échanges qui ont eu lieu après l'annonce de l'arrivée de la fédération dans Funkwhale, je suis entré en contact avec Jérémie qui m'a proposé de venir présenter le travail effectué dans Funkwhale aux RMLL 2018 qui auront lieu à Strasbourg début juillet. La présentation a pour titre « Web fédéré avec ActivityPub et WebFinger » et sera une introduction au fonctionnement concret de ces deux protocoles qui permettent de construire des applications fédérées. J'en profiterai également pour faire le lien avec ce qui a été fait dans Funkwhale, partager les techniques employées, les soucis rencontrés, etc. Les modalités pratiques et la date ne sont pas encore définies mais je pense que le programme définitif sera diffusé prochainement sur le site des [RMLL](#).

Merci pour tes réponses Eliot ! Comme de coutume, on te laisse le mot de la fin :

Tout d'abord, merci de m'avoir consacré du temps pour parler de ce projet. Cela me touche énormément, d'autant plus que je me sens assez proche de l'état d'esprit Framasoft, dont je suis les actions depuis plusieurs années: essayer de travailler avec les gens, de faire émerger autre chose, d'accompagner les personnes, plutôt que d'être en mode « Lis le fichu manuel!!!! ». Si vous souhaitez vous impliquer ou tout simplement en savoir plus, je voudrais donc vous dire ceci : n'ayez pas peur de venir discuter sur Matrix ou sur Mastodon. Je prends personnellement énormément de plaisir à accueillir les nouveaux et nouvelles venu et à répondre aux questions, techniques ou non. Et si vous souhaitez mettre la main à la pâte mais n'êtes pas sûr·e de savoir comment faire, on vous aidera !

Liens utiles

- Le site officiel <https://funkwhale.audio/>
- Où trouver des instances <https://join.funkwhale.audio/>
- Code source :
<https://code.eliotberriot.com/funkwhale/funkwhale>
- Le chat du projet #funkwhale:matrix.org
(<https://riot.im/app/#/room/#funkwhale:matrix.org>)
- Le compte Mastodon de Funkwhale :
<https://mastodon.eliotberriot.com/@funkwhale>
- le compte Mastodon d'Eliot :
<https://mastodon.eliotberriot.com/@eliotberriot>
- Soutenir financièrement :
<https://liberapay.com/funkwhale>