



Cahier des Charges du Noyau de Cantina

Nom définitif à trouver



JANUARY 1, 2024
CANTINA

Table of Contents

<i>Présentation du Projet</i>	2
<i>Contraintes techniques</i>	2
<i>Les besoins fonctionnels</i>	3
Interface WEB	3
Accès à l'interface.....	3
Fonctionnalité de l'interface.....	3
Interaction via socket	4
<i>Les ressources allouées au projet</i>	5
Les moyens humains	5
Les moyens d'échanges	5
Le budget	5
La dead line	5

Présentation du Projet

Cantina n'a pas de structure supportant ce projet. Cantina est un projet libre et open source. Le **projet X** (*nom à définir*) a pour but d'être le centre du futur écosystème Cantina, il servira de base de données utilisateur, de centre d'information pour les utilisateurs ainsi que centre de gestion de l'écosystème. Une solution semblable au projet d'écosystème de Cantina est l'outil de collaboration en ligne [Nextcloud](#).

Des versions anciennes de Cantina existent déjà mais sont très instables et peuvent uniquement être utilisées comme Proof of Concept (POC). Dans les anciennes versions de Cantina, le projet qui ressemble le plus à ce qui est décrit ici est « [Olympe](#) ».

Les principales cibles sont un public qui est soucieux de savoir où ses données sont stockées. L'écosystème Cantina, et par conséquent le **projet X**, se doivent d'être facile d'utilisation et avoir une installation simple, rapide et dans la mesure du possible sans avoir à trop utiliser une invite de commande pour que le plus de monde puisse installer et utiliser cet écosystème.

L'objectif de ce projet est d'avoir un produit fonctionnel et le plus intuitif possible le plus rapidement possible afin de pouvoir continuer les différents projets de l'écosystème.

Pour ce faire, pour le moment nous utilisons les outils [Flask](#) et [Python](#) pour créer la partie interface web de l'écosystème. Pour la partie socket, nous n'avons pas encore défini de technologie/modules à utiliser. Tout changement est possible si les gérants du projet et la communauté sont d'accord.

Contraintes techniques

Les seules contraintes sont des contraintes techniques. Le **projet X** doit être léger et rapide. Il doit fonctionner sur un système Linux type serveur virtuel sur un minimum de RAM et de processeur pour fonctionner sur une multitude de matériel possible. Le code doit fonctionner le plus possible via des fonctions et des classes afin d'améliorer la lisibilité et la compréhension du code ainsi que les performances d'un programme. Le **projet X** devra pouvoir être accessible via une interface WEB (grâce à Flask) et devra contenir une API pour fonctionner comme un véritable écosystème.

Les besoins fonctionnels

Interface WEB

Accès à l'interface

Le **projet X** devra être accessible uniquement via l'intermédiaire du Single Sign-On (SSO) de l'écosystème Cantina. Pour ce faire, l'interface devra directement rediriger vers le SSO si elle ne détecte pas **TOUTES** les conditions requises pour avoir l'accès. Les conditions sont : l'obtention du cookie ayant la même valeur que celui défini pour l'utilisateur dans la base de données, l'obtention du cookie ayant pour valeur un identifiant unique défini dans un fichier config.json ou dans la base de données (**choix définitif à faire**).

Fonctionnalité de l'interface

Le **projet X** devra contenir la gestion de compte de l'utilisateur connecté, pour les utilisateurs "lambda", nommée "user-mode" et une partie, pour les utilisateurs ayant les autorisations requises, nommée "pex-mode", avec la gestion d'utilisateur, ainsi que la gestion des services de l'écosystème.

La partie de la gestion de compte devra contenir des sections permettant de (**liste à compléter**) :

- Changer les mots de passe du compte.
- Changer le nom d'utilisateur du compte.
- Changer l'adresse électronique associée au compte.
- Changer la photo de profile (si l'option est activée).
- Voir les services que l'utilisateur a une permission d'accès.
- Gérer les outils d'authentification à deux facteurs (2FA).
- Modifier des paramètres utilisateurs visuels pour améliorer le confort et l'ergonomie en fonction de chacun.

La partie "pex-mode" devra contenir ces sections permettant de (**liste à compléter**) :

- Pour la partie gestion de compte :
 - Voir la liste complète de tous les utilisateurs.
 - Voir en détail le compte d'un utilisateur.
 - Modifier les informations d'un compte spécifique.
 - Modifier les informations de plusieurs comptes à la fois.
 - Créer un utilisateur avec une liste de mots de passe à génération aléatoires et un identifiant unique aléatoire à changer lors de la connexion ou basé sur les informations définies (nom.prénom ou prénom.nom, choix laissé à l'utilisateur final).
 - Pouvoir activer/désactiver le changement de nom d'utilisateur, de mots de passe, d'adresse électronique, de photo de profile.
 - Suppression et désactivation de compte en grande quantité ou un utilisateur à la fois.
 - Gérer les permissions de différents comptes en grande quantité en même temps ou d'un utilisateur particulier.
 - Gérer les accès aux différents modules de l'écosystème pour un utilisateur particulier, un groupe d'utilisateurs ou de plusieurs comptes en même temps.
- Pour la partie gestion de l'écosystème :
 - Voir la liste de tous les services de l'écosystème connecté à l'écosystème.

- Visualiser en temps réel si les services sont déconnectés, si une erreur a été intercepté ou si les services fonctionnent correctement.
- Pouvoir éteindre ou allumer plusieurs services en même temps.
- Pouvoir gérer individuellement un service : voir les derniers logs, éteindre le service, allumer le service (**marche à suivre à définir**).
- Pouvoir installer et supprimer des modules de l'écosystème.
- Pouvoir mettre en maintenance, un ou plusieurs modules de l'écosystème en même temps.

Interaction via socket

Pour pouvoir fonctionner comme un véritable écosystème, le **Projet X** doit pouvoir communiquer avec l'ensemble des modules de l'écosystème Cantina connecté au **Projet X**.

Pour ce faire, le Projet X devra (**liste à compléter**) :

- Pouvoir communiquer avec d'autres instances de l'écosystème.
- Pouvoir réceptionner les erreurs des différentes instances de l'écosystème.
- Pouvoir transmettre les ordres (maintenance, les extinctions, les redémarrages, etc...) ordonnées par l'utilisateur via l'interface web.

Les ressources allouées au projet

Les moyens humains

Le **Projet X** n'a pas de moyen humain fixe définis, ce qui veut dire que toute personne voulant apporté son expertise et sa connaissance à ce projet est la bienvenue. Cependant, ces contributeurs doivent respecter certaines règles.

Les moyens d'échanges

L'utilisation de [Github](#) et du dépôt utilisé par le **Projet X** est obligatoire. Si le contributeur veut faire proposer une amélioration, celui-ci devra ouvrir [une Pull Request](#) (PR), sur le dépôt en ligne. Cette PR pourra, en cas de non-conformité, être refusé par les administrateurs s'ils jugent que le contenu du code n'est pas correcte (manque de lisibilité, manque de documentation, non utilisation des fonctions, etc...), si le code rajouté est malveillant, si la fonctionnalité ajoutée n'est pas utile ou si la fonctionnalité n'est pas conforme aux conditions d'utilisations et de créations de Cantina.

Afin de faciliter les échanges entre contributeurs, il est fortement recommandé de rejoindre le serveur discord de Cantina ([lien ici](#)). Des salons, vocaux et textuels, sont disponibles pour discuter, d'autres salons peuvent être mis en place si besoin.

Le budget

Le Projet X n'a aucun budget, et repose uniquement sur la base du volontariat. C'est le but de faire un projet open source.

La dead line

Ce projet n'a pas de dead line à proprement parler, il faut juste que le projet fonctionne le plus rapidement possible afin de pouvoir créer les autres projet rapidement. Il faudrait que dans l'idéal, le projet doit pouvoir être mis sur une branche de production vers le premier juillet deux-mille-vingt-quatre (01/07/2024).