



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 au l'interface.....	3

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 cet écosystème. 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 au 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.