



POSTE : Développeur Full-Stack

Glados, c'est quoi ?

Glados est une entreprise issue de l'évolution de l'intelligence artificielle.

À l'image des plus grandes entreprises de notre siècle, tout a démarré dans un garage. Sa créatrice, la grande Chell Aperture, s'est lancée le défi de connecter toutes les maisons de ce monde afin de les rendre autonomes.

Mais... Malheureusement la grande Chell n'aura jamais pu mener ce projet à bien car elle est décédée d'une indigestion de gâteau (à la patate) peu de temps après l'annonce de **Glados** au grand public.

Et maintenant ?

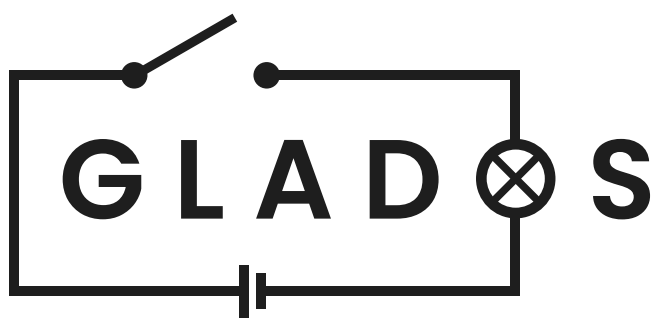
Après ces terribles événements, l'entreprise a été reprise par le merveilleux neveu de Chell : **Dave LIDE** (moi-même).

Les objectifs ont alors été revu à la hausse, afin de respecter les dernières volontés de Chell. Malheureusement, tous les développeurs qui travaillaient pour Glados ont disparu sans laisser aucune trace.

Et le poste ?

Afin de pouvoir récupérer les subventions qui nous sont dues, une première version de **Glados** est attendue dans un délai (quelque peu) restreint.

Pour cette première version, **Glados** a pour projet de proposer une interface afin de visualiser et contrôler toutes les entités d'une maison connectée.



Le projet

Pour réaliser à bien ce projet, plusieurs étapes ont été imaginées. Il est important de les réaliser dans l'ordre et de s'assurer que les étapes sont fonctionnelles avant de passer à la suivante.

Étape 1 : Récupérer et lancer le projet – DEVOPS

Pour commencer, à travailler sur le projet **Glados**, il faut récupérer le code source et lancer l'application via Docker.

Le projet est constitué de :

- Un serveur back-end **Flask** (python) : dossier glados-api-draft
- Flask : <https://flask.palletsprojects.com/en/2.1.x/>
- Une application front-end **VueJs** (js) : dossier glados-api-draft
- Vue : <https://vuejs.org/guide/introduction.html>

Les fichiers **README.md** sont une aide précieuse lorsque l'on débute sur un projet.

Étape 2 : Ajout de fonctionnalités – BACK END

Deux **routes** sont déjà présentes :

- **/** : qui nous retourne la version de l'application
- **/entities** : qui nous retourne la liste de tous les entités présentes dans la base de données

La route **/entities** contient déjà un filtre "type".

L'objectif est de modifier cet endpoint pour pouvoir également filtrer les résultats sur :

- La pièce de l'entité
- Le statut de l'entité

La meilleure façon de travailler est en TDD : <http://igm.univ-mlv.fr/~dr/XPOSE2009/TDD/pagesHTML/PresentationTDD.html>

Commence par bien écrire tes tests, puis développe ton application.



Étape 3 : Visualisation de la maison connectée – FRONT END

Pour cette étape, c'est ta créativité qui va être mise à rude épreuve. En effet, il faudra que tu améliores la vue existante. Il faudra la rendre bien plus agréable d'un point de vue UI / UX et surtout il faudra qu'elle permette d'afficher les différentes entités avec leurs informations associées (pièces, valeurs, ...).

Garde bien en tête que tu es libre sur cette étape. Si par exemple, tu penses que des icônes pourraient rendre le site incroyable : lance-toi !

De plus, l'existant n'est pas ancrée. Que ce soit les couleurs, la structure ou même les animations : tu as carte blanche !

Étape 4 : Contrôle de la maison connectée – FULL STACK

Jusqu'à présent, tu as pu toucher au back-end ou au front-end séparément. Maintenant, tu vas interagir avec les deux en même temps.

Tu vas devoir développer toute la partie front-end et back-end permettant d'éditer les entités de la base de données.

Inspire toi de l'existant et renseigne toi sur les différentes méthodes REST ainsi que sur les bonnes pratiques.

Étape 5 : Faire parler Glados – FULL STACK

Afin de rendre le site bien plus accessible, il nous est nécessaire qu'un text-to-speech soit implémenté. Celui-ci devra lister les différentes entités et donner leurs états. Sois créatif de nouveau, cette étape sera très importante pour la présentation aux différents investisseurs, un effet **WOW** peut faire la différence !

Il existe de nombreux exemples d'outils qui permettent d'utiliser cette fonctionnalité, tu es libre de trouver celui qui te convient le plus. Par exemple : <https://usefulangle.com/post/98/javascript-text-to-speech>



POSTE : Développeur Full-Stack

BONUS : Make it AWESOME !!!

Si tu as réussi à réaliser toutes les précédentes étapes, tu peux utiliser tes compétences back-end et front-end pour améliorer le projet comme bon te semble. Nouvelles fonctionnalités, amélioration UX, amélioration UI...

Attention malgré tout à ne pas détériorer les fonctionnalités de base réalisées durant les 5 premières étapes.

Besoin d'aide ?

Il se pourrait que tu aies besoin d'aide pour certaines de ces étapes. Je ne te promets pas de savoir comment faire pour te débloquent mais je ferai tout mon possible.

Dans le cas où tu aurais vraiment besoin d'un coup de pouce, envoi moi un mail avec comme sujet :
Problème - Étape [NUMÉRO DE L'ÉTAPE]

Bon courage à toi, et n'oublie pas :

"The cake is a lie."