

****Plan de mémoire****

****REMERCIEMENTS****

- Expression de gratitude envers les personnes et entités ayant contribué à la réussite de la période d'alternance et à la réalisation du dossier projet.

****INTRODUCTION****

- Présentation du contexte et des objectifs du dossier projet.

****CHAPITRE 1: CONTEXTUALISATION****

- ****Compétences couvertes par le dossier projet (Compétences acquises)****

- {
 - Maquetter une application Web (figma)
 - Développer des composants d'accès aux données (svelte)
 - Développer la partie front-end d'une interface utilisateur web (sveltestrap)
 - Développer la partie back-end d'une interface utilisateur web (SvelteKit)
 - Concevoir une base de données (firebase)
 - Mettre en place une base de données (firebase)
 - Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement (??)
 - Concevoir une application (svelte)
 - Développer des composants métier (SvelteKit)
 - Construire une application en couches (cf: pipeline gitlab)}

- ****Abstract****

- {

En anglais ??

- ****Présentation de l'entreprise**** :

- {

EasyTools est une entreprise innovante spécialisée dans la fourniture de solutions de gestion de la vidéo infrastructure. Avec une approche flexible, EasyTools offre la possibilité d'exécuter ses outils aussi bien sur site (on premises) que dans le cloud privé ou public. Cette polyvalence permet aux clients de bénéficier d'une infrastructure vidéo adaptée à leurs besoins, tout en utilisant le même logiciel, la même API et la même interface, quelle que soit la plateforme choisie.

- ****Environnement technique****

- {

- Environnement Node.js (<https://nodejs.org/en>).
- Framework front-end : Svelte (<https://svelte.dev/>).
- Frameworks assurant la responsivité des composants : Sveltestrap (<https://sveltestrap.js.org/>)
- Framework d'interface : SvelteKit (<https://kit.svelte.dev>).

- Framework vidéo : ffmpeg (url)
- Environnement Back-end : Firebase
(https://firebase.google.com)

}

- Organisation de l'application Visual Flow

{

Visual Flow est protégé par un composant d'authentification (Login.svelte) donnant accès aux composants Status, Schedules, Ingestion, Collection, ainsi que le UserSettings permettant à l'utilisateur de sauvegarder ses préférences d'accessibilité en fonction de l'utilisation spécifique qu'il a de l'application. Ces composants contiennent la totalité des fonctionnalités utilisateurs de l'application web.

}

****CHAPITRE 2: AMÉLIORATION DE L'ERGONOMIE DE L'INTERFACE DES PARCOURS****

- ****Analyse de l'existant et expression du besoin****
- ****Analyse technique du projet****
- ****Réalisation du projet****
- ****Gestion du projet****

****CHAPITRE 3: ENRICHISSEMENT DE L'ESPACE UTILISATEURS****

- ****Analyse de l'existant et expression du besoin****
- ****Analyse technique du projet****
- ****Réalisation du projet****
- ****Gestion du projet****

****CHAPITRE 4: SOLUTION POUR L'EXPLOITATION DES DONNÉES VIDEO****

- ****Analyse de l'existant et expression du besoin****
- ****Travail de recherche****
- ****Analyse technique du projet****
- ****Réalisation du projet****
- ****Gestion du projet****

****CONCLUSION****

- Synthèse des principaux résultats et contributions.
- Perspectives futures et recommandations d'accessibilité.

****RÉFÉRENCES****

****Architecture en couches de l'application Visual Flow : ****

1. ****Couche de présentation (Interface utilisateur) : ****

{

 - Cette couche représente l'interface utilisateur de l'application EasyTools, où les utilisateurs interagissent avec le système pour gérer leur infrastructure vidéo.
 - Elle utilise des technologies modernes telles que Svelte, Sveltestrap et SveteKit pour offrir une expérience utilisateur intuitive et réactive.
 - Les composants de cette couche sont responsables de la présentation des données et de la réception des entrées de l'utilisateur.

}

}

2. **Couche Métier (Logique métier) :**

{

- La couche métier contient la logique métier de l'application Visual Flow. Elle traite les données reçues de la couche de présentation, exécute des opérations d'ingestion spécifiques et interagit avec la couche d'accès aux données.

- Cette couche inclut des fonctionnalités telles que la gestion des médias vidéo ainsi que les playlists composant la programmation des diffusions de chacune des chaînes gérées par les utilisateurs depuis l'application, la configuration de l'infrastructure vidéo et la gestion des autorisations d'accès aux ressources de l'application.

}

3. **Couche d'accès aux données (Accès aux données) :**

{

- La couche d'accès aux données est responsable de l'interaction avec la base de données de l'application Visual Flow.

- Elle utilise des outils tels que Firebase et SveteKit pour fournir une interface d'accès aux données, permettant ainsi à la couche métier d'effectuer des opérations de lecture, d'écriture et de mise à jour sur la base de données.

}

4. **Couche Infrastructure (Infrastructure technique) :**

{

- La couche infrastructure englobe les composants techniques de l'application Visual Flow, notamment les services cloud.

- Elle assure le bon fonctionnement et la disponibilité de l'application en fournissant les ressources nécessaires à son exécution dans le cloud.

}

SYNTHESE:

{

En résumé, l'application Visual Flow est organisée en différentes couches logiques, chacune ayant un rôle spécifique dans le fonctionnement global de l'application. Cette architecture en couches permet une séparation claire des responsabilités, favorisant ainsi la maintenabilité, la scalabilité et la flexibilité de l'application.

}

ANNEXES

- Éléments complémentaires nécessaires à la compréhension du dossier projet (schémas, tableaux, code source, assets).