



Nom de votre projet :	Sylver.Service
Membres de l'équipe :	Sylvio Pelage Maxime
Membres de l'équipe :	Elvann Jolivel
Membres de l'équipe :	Deecleane Coralie
Membres de l'équipe :	
Membres de l'équipe :	
Niveau d'étude :	Terminale
Établissement scolaire :	Lycée général et technologique Faustin Fleret
Enseignante/enseignant de NSI :	Mr. Somerfeld

> PRÉSENTATION GÉNÉRALE :

• Idée

Faisant partie d'une spécialité complexe où il est nécessaire d'apprendre beaucoup d'informations, nous avons eu l'idée d'une application permettant de partager des cours entre élèves, afin de faciliter les révisions, notamment pour le baccalauréat, mais également pour les épreuves de français en première.

• Présentation

Sylver.Service est une application destinée aux étudiants permettant de poster des tutoriels personnalisés et de joindre des fichiers Word. Elle contient une base de données modérée par nous-mêmes qui répertorie les comptes d'utilisateurs, leurs publications et les signalements éventuels.

• Besoins

On répond souvent à un besoin d'apprentissage plus approfondi chez les élèves ou chez les gens dans leur vie de tous les jours. Parfois, il peut s'avérer difficile de trouver des informations, et parfois, même lorsque trouvées, ces informations ne sont pas expliquées de façon claire. Il est souvent dit qu'un élève qui maîtrise un sujet saura mieux l'expliquer à un autre élève. Ainsi, l'application Sylver.Service vise à mettre en relation les élèves ou les particuliers qui souhaitent approfondir leurs connaissances.

> ORGANISATION DU TRAVAIL :

En ce qui concerne l'organisation, nous nous sommes en quelque sorte séparés en deux groupes : Sylvio et Elvann se chargeaient de la structure de l'application. Ils programmaient les différentes interfaces, assuraient l'accès à la base de données et vérifiaient les bogues. Ils étaient les programmeurs.

Deecleane était chargée de concevoir et d'implémenter les designs dans le code. J'ai créé tous les émoticônes et images de l'application.

> LES ÉTAPES DU PROJET :

• Essence du projet

Après avoir échoué au projet NSI de 2023, il nous fallait un projet qui avait une chance d'être apprécié, mais également

qui pourrait aider les élèves dans leur parcours scolaire. Dès lors, nous avons pensé à une petite application en Python qui pourrait regrouper des tutoriels sur divers sujets tels que les mathématiques, l'informatique, voire le sport !

• Début

Tout d'abord, il nous a fallu imaginer les interfaces de Sylver.Service afin de les implémenter ensuite avec Pygame. Dès lors, nous avons réalisé des croquis pour avoir une base.

• Développement Générale

Le développement de chaque partie était coupé en 3 étapes :

1. Création des fonctions permettant la liaison SQL dans un autre fichier python
2. Implémentation d'une interface de test pour vérifier le bon fonctionnement des fonctionnalités de l'application
3. Implémentation d'une meilleure interface, plus efficace

• Phase de développement

- Tout d'abord, nous avons commencé par coder l'accueil, l'environnement central de l'application, car c'est depuis celui-ci qu'on peut accéder à toutes les "pages" de l'application.
- Chaque "page" a été séparée en fonction. Hormis l'accueil, chaque "page" est rangée dans une fonction à laquelle on peut faire appel quand on le souhaite, ce qui facilite grandement le code et l'éclaircit également.
- Ensuite, nous avons débuté la création d'un fichier Python à part, nommé "Sylver_class_import", qui regroupe les accès à la base de données mais également de petites fonctions permettant des interactions visuelles brèves (telles que des erreurs ou des interactions quelconques). Ce fichier est très important car sans lui, l'application ne fonctionnerait pas. C'est dans celui-ci que se font les recherches de tutoriels, les ouvertures de tutoriels, les connexions de comptes, les créations de comptes, et bien d'autres éléments essentiels à l'application. Nous avons choisi de faire un fichier à part car cela permet de décentraliser le code et c'est bien mieux pour un débogage plus efficace. Ce code comporte également toutes les classes de l'application, donc quand on a un problème de classe, on sait directement où chercher. De plus, ce

code est bien plus court que le code principal, ce qui facilite grandement la recherche de bugs.

- Après avoir grandement avancé sur les fonctionnalités de l'application, on s'est penché sur la création d'un design plus attrayant et plus efficace ! C'est un aspect très important pour une application car si une application a les meilleures fonctionnalités du monde mais que l'interface laisse à désirer, cette application restera également à désirer.

- Dès que l'application est devenue stable, on s'est lancé dans de petites améliorations sympas, telles que la mise en place d'un système de signalement, la mise en place de catégories dans l'application, ...

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

À l'heure actuelle, le projet est dans sa phase de finalisation. Des fichiers exécutables (.exe) sont fréquemment créés et mis en ligne sur GitHub afin de permettre une utilisation facile de l'application. Pour vérifier l'absence de bugs, nous avons établi une liste de tâches à exécuter que nous avons suivie dans différentes situations. De plus, pour nous assurer que l'application s'adapte suffisamment aux différents écrans, nous avons fréquemment modifié le zoom de l'écran de l'ordinateur utilisé pour observer comment l'application se rendait sur différents formats. Pour faciliter l'utilisation de l'application, nous avons inclus différents fichiers d'aide accessibles depuis l'application, qui peuvent aider à son utilisation. Durant le développement de l'application, nous avons rencontré différents problèmes, inhérents au métier de développeur, mais deux d'entre eux étaient fondamentaux :

1. L'adaptation aux écrans :

Pour remédier à ce problème d'adaptation, nous avons revu une bonne partie des designs en modifiant un élément très important. Auparavant, les tailles étaient définies de manière fixe, soit avec une valeur spécifique, soit de manière moins fixe mais assez problématique, avec par exemple une variable moins un nombre fixe. Le problème était que cette méthode entraînait souvent des rendus très différents sur différents écrans. Nous avons donc décidé d'adopter des

ratios/proportions pour les tailles des différents éléments présents dans l'application.

2. La connexion à la base de données et sa reconnexion :

La connexion à la base de données prenait parfois un temps excessivement long, ce qui nuisait au bon fonctionnement de l'application. Initialement, nous avons tenté de modifier la méthode de connexion utilisée, mais les changements apportés étaient minimes, voire introduisaient davantage de bugs. Nous avons donc finalement décidé de changer de module et de revoir les algorithmes de connexion, ce qui s'est avéré être une solution efficace. À présent, la connexion s'effectue plus rapidement et le temps de chargement entre différentes recherches est plus constant.

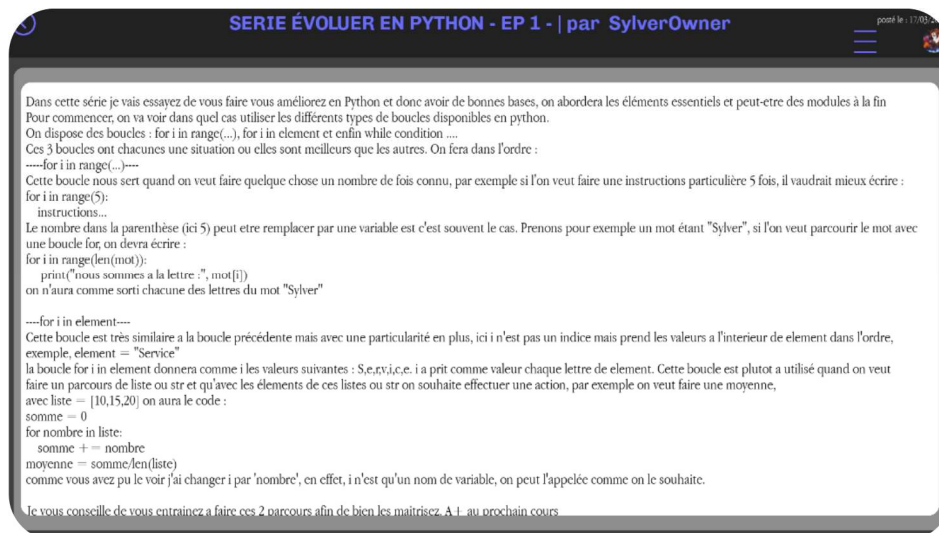
VOICI QUELQUES IMAGES DE L'APPLICATION :



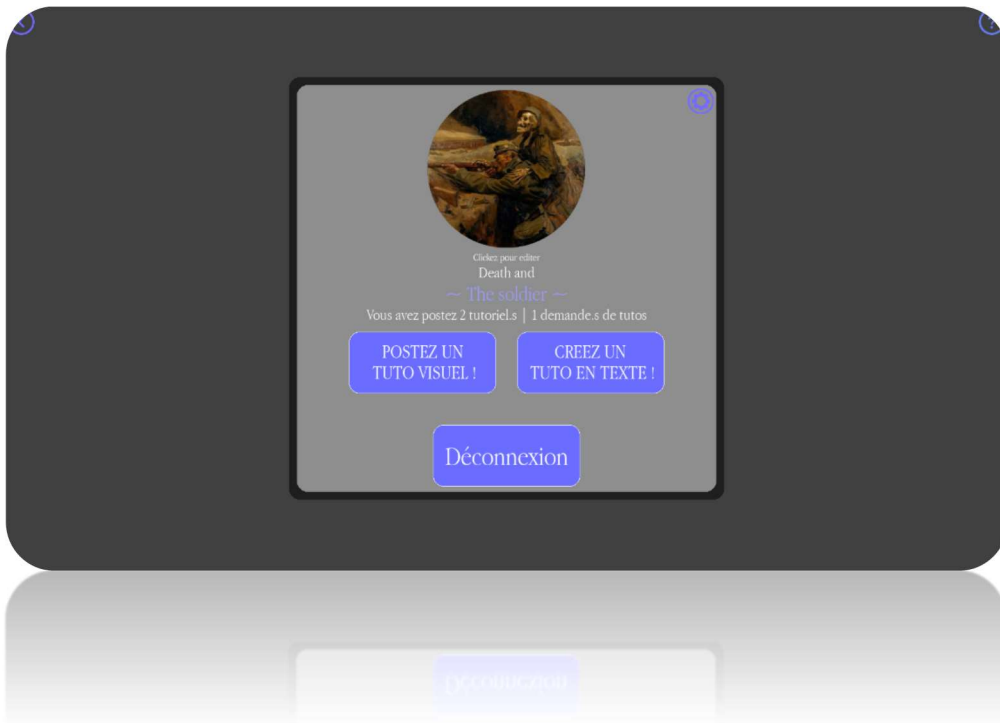
Accueil



Menu



Espace tuto



Espace compte connecté



Espace création/connexion compte

Ecrivez ici votre tutoriel The soldier

Titre

Contenu

Valider

Espace pour écrire son tuto

> OUVERTURE :

Pour améliorer le projet, nous devrions essayer d'avoir un meilleur temps de réponse du serveur, ainsi qu'un temps de réponse moins variable. Cela pourrait être réalisé en investissant dans un serveur payant.

Nous aimerions ajouter une page de contact à l'application pour que les utilisateurs puissent nous contacter directement via l'application. C'est une tâche assez simple à réaliser car nous l'avons déjà fait dans le passé (SylverDonjon), mais actuellement, nous manquons de temps, donc nous préférons reporter cette modification à plus tard. Nous souhaitons également ajouter la possibilité de corriger un tutoriel, c'est-à-dire la possibilité de le modifier après l'avoir publié. Nous avons également envisagé d'ajouter un popup affichant les statistiques du tutoriel, telles que le nombre de vues, de likes, de dislikes. Par conséquent, il faudrait également mettre en place un système de likes et de dislikes. Ce ne sont pas des tâches trop difficiles à réaliser, mais elles nécessitent du temps. Enfin, nous pourrions ajouter des titres de compte en fonction du nombre de tutoriels postés.

Si c'était à refaire, nous utiliserions plus souvent le module io de Python, qui nous permet de charger une image ou un document juste avec ses bits. Nous essayerions également de mieux séparer les fichiers de l'application dès le début afin de faciliter le repérage lors du débogage. En ce qui concerne

notre organisation, nous essaierions d'être plus réguliers dans les réunions et la planification du travail.

Grâce à ce concours, nous avons pu approfondir nos connaissances sur les bases de données abordées en Terminal. Nous avons appris à lier Python et SQL via un module, ainsi que la manière d'héberger une base de données sur internet.

Nous pensons que notre projet favorise l'inclusion, car il est coopératif par nature. Tout le monde peut contribuer pour aider les autres. Ainsi, chacun peut trouver son bonheur ici, et s'il ne le trouve pas, il peut toujours faire une suggestion qui sera peut-être prise en compte par l'équipe ou quelqu'un de bienveillant 😊. De plus, il y en a pour tous les goûts : les sportifs, les matheux, les cuisiniers. Des catégories ont été mises en place pour satisfaire un grand nombre, et ces catégories ne sont pas figées, leur nombre peut facilement évoluer dans le temps 😊

PS : En ce qui concerne la sécurité, nous aimerions que le fichier ".env" de l'application soit inaccessible, mais nous n'avons pas encore trouvé comment y parvenir malheureusement.

**NOUS ESPÉRONS QUE VOUS TROUVEREZ VOTRE BONHEUR DANS
L'APPLICATION, BONNE EXPÉRIENCE À VOUS ! 😊**