

RAPPORT DE PROJET

Smart'Eat



Table des matières

Introduction	3
Présentation	3
Connexion/Inscription	3
inscription.html	4
connexion.html	5
Page d'accueil.....	5
Chatbot	6
Journal	6
Quiz	7
Calendrier	7
À propos	9
Organisation du code	9
Conclusion	10

Introduction

En L2, nous avons appris les bases de la programmation web, notamment avec du HTML, du CSS et du JavaScript. Pour finir l'année, nous devons rendre un site web fonctionnel et présentant de nombreuses rubriques. C'est pourquoi nous avons eu l'idée de fusionner notre idée de projet transverse avec la création de ce site. En effet, Smart'Eat est le nom de notre projet transverse, et celui-ci propose des outils essentiels pour guider ses utilisateurs vers une meilleure organisation de leur alimentation, mais aussi pour les éduquer sur les bases de la nutrition.

Présentation

Smart'Eat propose aujourd'hui de nombreuses fonctionnalités comme un suivi de ses macronutriments personnels, un chatbot appelé Smart'Bot, des jeux éducatifs sur la nutrition, quelques recettes et un calendrier pour pouvoir planifier ses repas.

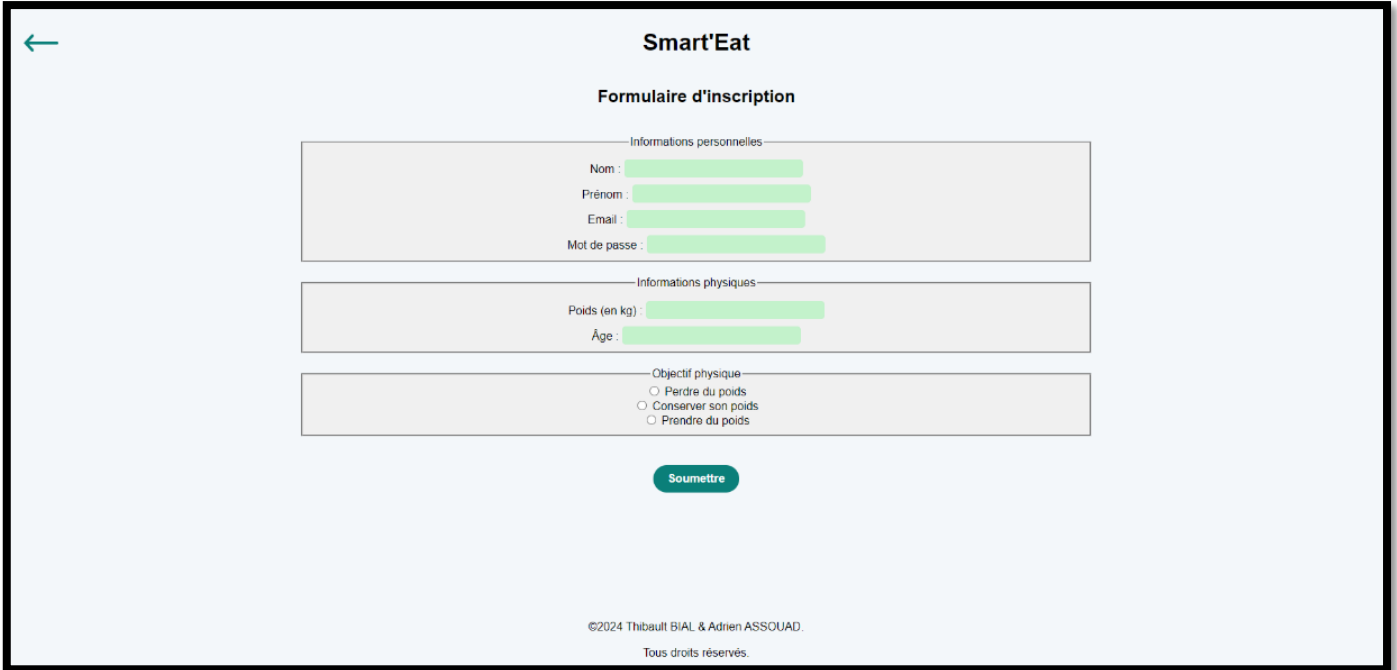
Connexion/Inscription

Pour pouvoir accéder aux diverses fonctionnalités de notre site, l'utilisateur doit d'abord créer son profil utilisateur. Pour cela, nous avons créé deux pages HTML qui sont connexion.html et inscription.html. Cependant, nous n'avons pas rendu notre site dynamique, car nous n'avons pas encore appris à le faire.

inscription.html

Sur cette page, nous affichons un formulaire (balise <form>) avec plusieurs rubriques (balises <fieldset>) contenant plusieurs inputs (text, email, password, number, radio) et un bouton « Soumettre » affichant les informations que l'utilisateur a saisi via un pop-up, et le renvoyant à la page de connexion (fonction créée dans un fichier login_signup.js).

Voici un aperçu :



The image shows a web form titled "Smart'Eat" with the subtitle "Formulaire d'inscription". The form is divided into three sections: "Informations personnelles", "Informations physiques", and "Objectif physique". The "Informations personnelles" section contains fields for "Nom", "Prénom", "Email", and "Mot de passe". The "Informations physiques" section contains fields for "Poids (en kg)" and "Âge". The "Objectif physique" section contains three radio buttons: "Perdre du poids", "Conserver son poids", and "Prendre du poids". A "Soumettre" button is located below the form. At the bottom of the page, there is a copyright notice: "©2024 Thibault BIAL & Adrien ASSOUAD. Tous droits réservés."

127.0.0.1:5500 indique

Bonjour Thibault Bial, votre email est : thibault.bial@efrei.net et votre mot de passe est : Smart123.

Votre poids est de 70 kg et vous avez 19 ans.

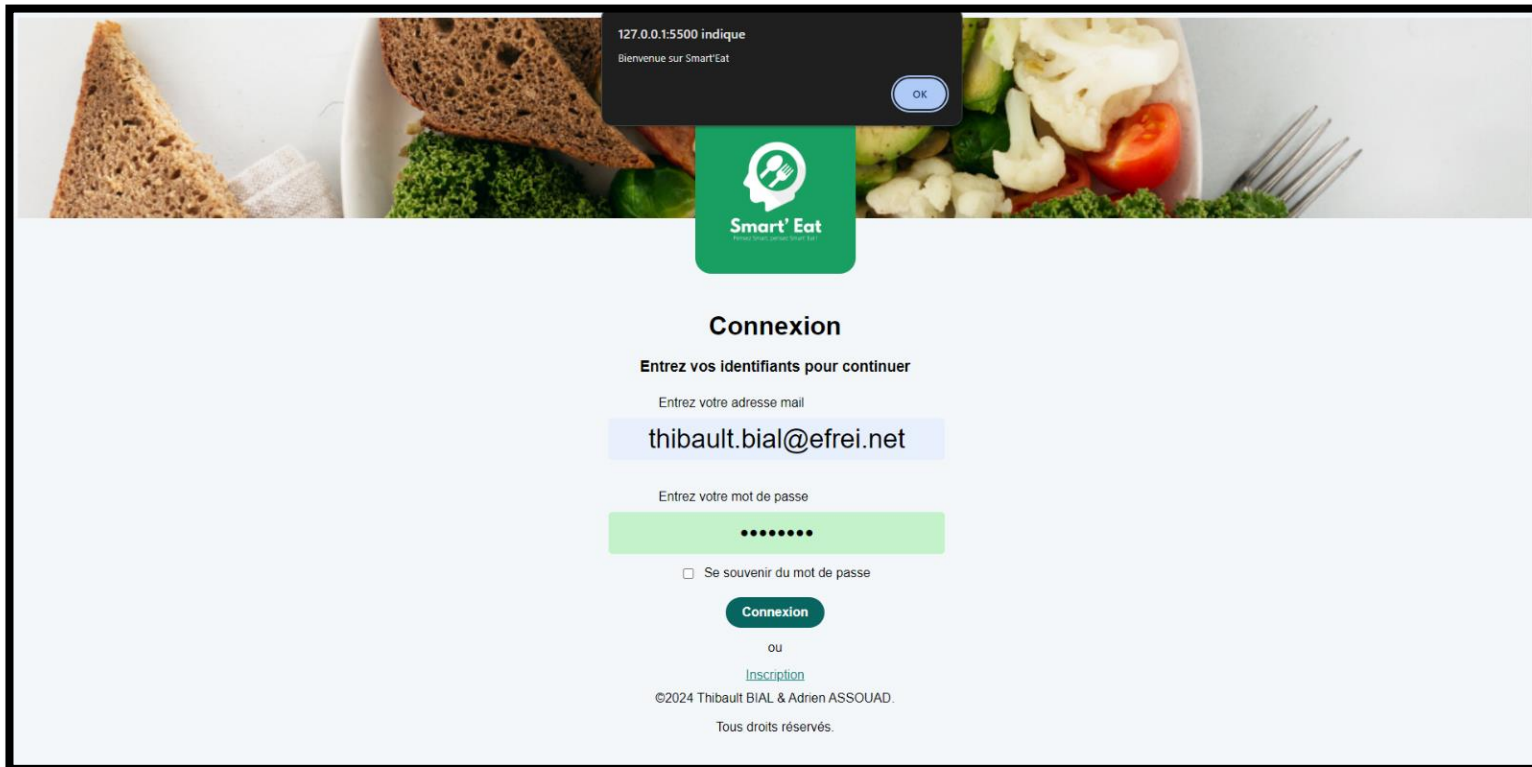
Votre objectif est de prendre du poids.

OK

connexion.html

Cette page permet à l'utilisateur de s'identifier via son adresse mail et son mot de passe. Une fois, les champs remplis un pop-up apparaît et lui souhaite la bienvenue sur Smart'Eat et le redirige vers la page d'accueil.

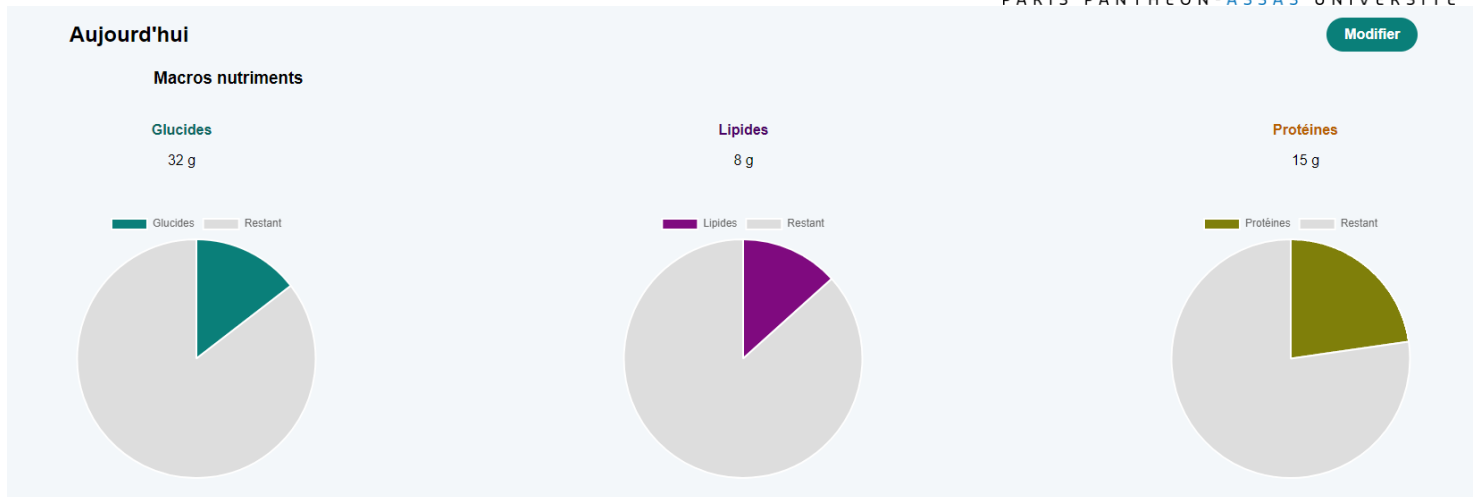
Voici un aperçu :



Page d'accueil

Sur cette page d'accueil, on remarque d'abord un <div> ayant comme id « tools » qui permet de se diriger vers les différentes page html. Ce « header » qui n'en est pas vraiment un (car notre header est le nom de notre site avec notre logo) sera disponible sur l'ensemble des pages.

Et juste en dessous, se trouve trois camemberts permettant de visualiser la progression des macronutriments. En cliquant sur le bouton modifier, un petit formulaire apparaît,



laissant à l'utilisateur la possibilité de rentrer manuellement ses macronutriments, mettant à jour sa progression visible sur les camemberts.

Nous avons utilisé des balises <canvas> pour pouvoir les représenter, et des fonctions mettant à jour les variables correspondantes dans un fichier nutriments.js.

Chatbot

Nous avons repris le chatbot que nous avons réalisé lors du TP4 et nous avons modifié celui-ci pour que les questions soient adaptées à notre projet. Nous avons rajouté un `addEventListener` pour que lorsque nous appuyons sur la touche « entrée » après avoir écrit une réponse cela active le bouton « envoyer ». De plus, un texte nous indique quelques exemples de questions à poser au Smart'Bot. Autre chose à savoir, vous pouvez cliquer sur le logo en haut à gauche pour revenir à l'accueil.

Smart'Bot

09/06/2024 23:46:10
Bonjour

09/06/2024 23:46:10
Bonjour ! Comment puis-je vous aider ?

09/06/2024 23:46:19
Combien de calories dois-je consommer par jour ?

09/06/2024 23:46:19
L'apport calorique quotidien varie en fonction de l'âge, du sexe, du poids, de la taille et du niveau d'activité physique. En moyenne, un homme adulte a besoin de 2500 calories par jour et une femme adulte de 2000 calories par jour. Pour des conseils personnalisés, il est préférable de consulter un professionnel de la santé.

Posez votre question...

Envoyer

Smart'Bot est un outil proposé par Smart'Eat
Avez-vous des questions sur la nutrition ?

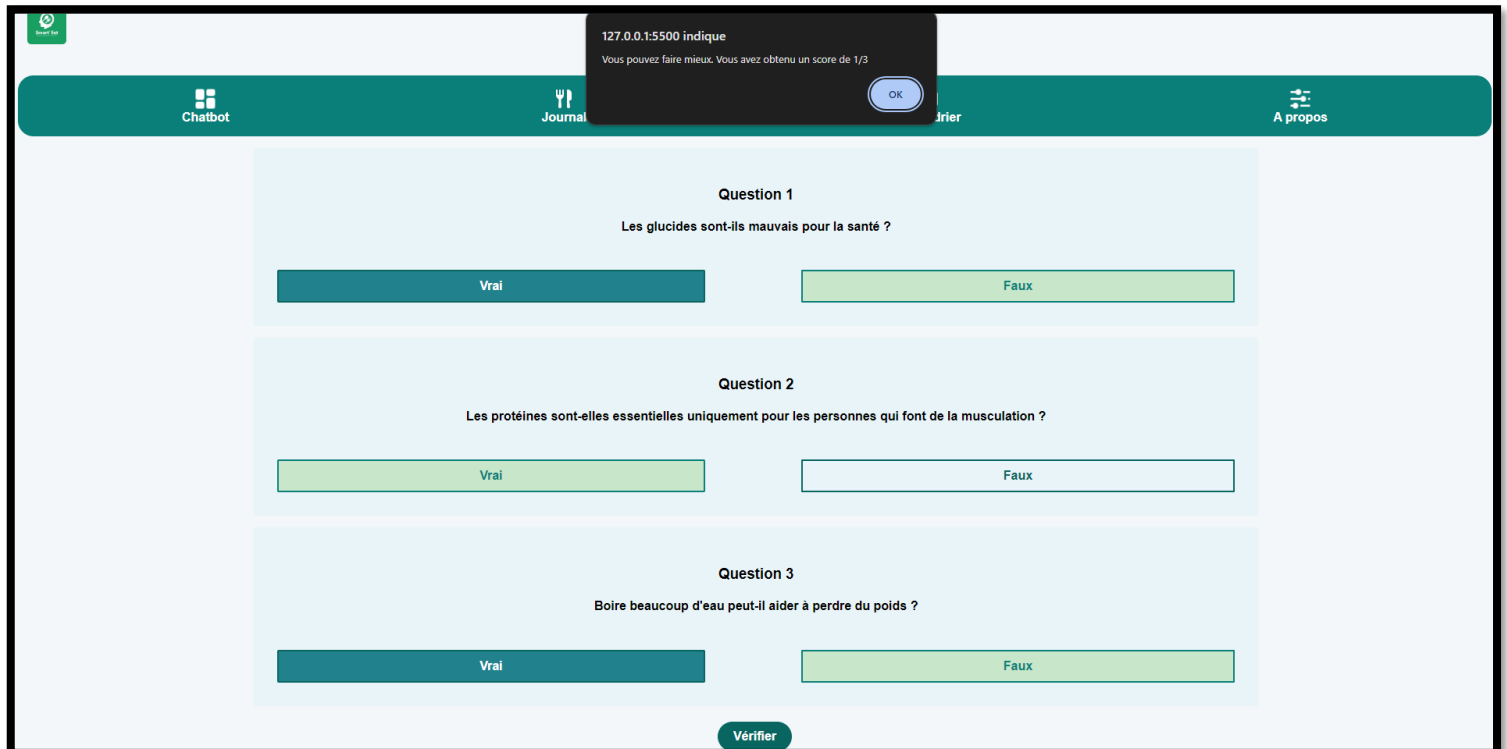
Exemple :
- Combien de calories dois-je consommer par jour ?
- Quelles sont les bonnes sources de protéines ?

Journal

Cette page n'est pas encore disponible, son but sera d'afficher les informations sauvegardées sur les repas des utilisateurs à chaque fois que ceux-ci se connectent sur le site. Pour le moment, le site n'est pas dynamique. Ce n'est donc pas encore possible.

Quiz

Sur cette page, nous avons fait 3 questions de type vrai/faux sur l'alimentation. Jusqu'ici rien de très compliqué. Cependant, au lieu de faire de simples boutons de type radio, nous avons mis des types button (pour plus d'esthétique) que nous avons dû associer à



127.0.0.1:5500 indique
Vous pouvez faire mieux. Vous avez obtenu un score de 1/3

Chatbot Journal Ajouter A propos

Question 1
Les glucides sont-ils mauvais pour la santé ?

Vrai Faux

Question 2
Les protéines sont-elles essentielles uniquement pour les personnes qui font de la musculation ?

Vrai Faux

Question 3
Boire beaucoup d'eau peut-il aider à perdre du poids ?

Vrai Faux

Vérifier

un fichier JavaScript pour les faire fonctionner comme des boutons radio. En plus de cela, dans notre fichier JavaScript, nous avons fait en sorte que lorsqu'on soumet nos réponses cela nous affiche notre score avec un message via une alerte.

Calendrier

La page calendrier.html propose tout d'abord des recettes qui peuvent être triées par catégorie. En faisant passer la souris par-dessus, nous pouvons découvrir les apports en macronutriments de celle-ci et un bouton « Ajouter au calendrier » qui pour l'instant n'est pas encore fonctionnelle, mais qui a pour objectif d'ajouter un événement au calendrier situé juste en dessous. Il est possible de découvrir la recette en cliquant rapidement sur l'image de la carte avant qu'elle ne se retourne.


Vegan

Végé

Tous


Carnivore

Liste des repas




Salade de lupins et haricots verts

Calories : 185kcal



Pizza aux légumes d'été

Calories : 466kcal



Escapopes de dindes minceur aux champignons

Calories : 263kcal


Vegan

Végé

Tous

Carnivore

Liste des repas



Salade de lupins et haricots verts

Calories : 185kcal


Apports

Glucides : 52 g

Protéines : 16 g

Lipides : 22 g

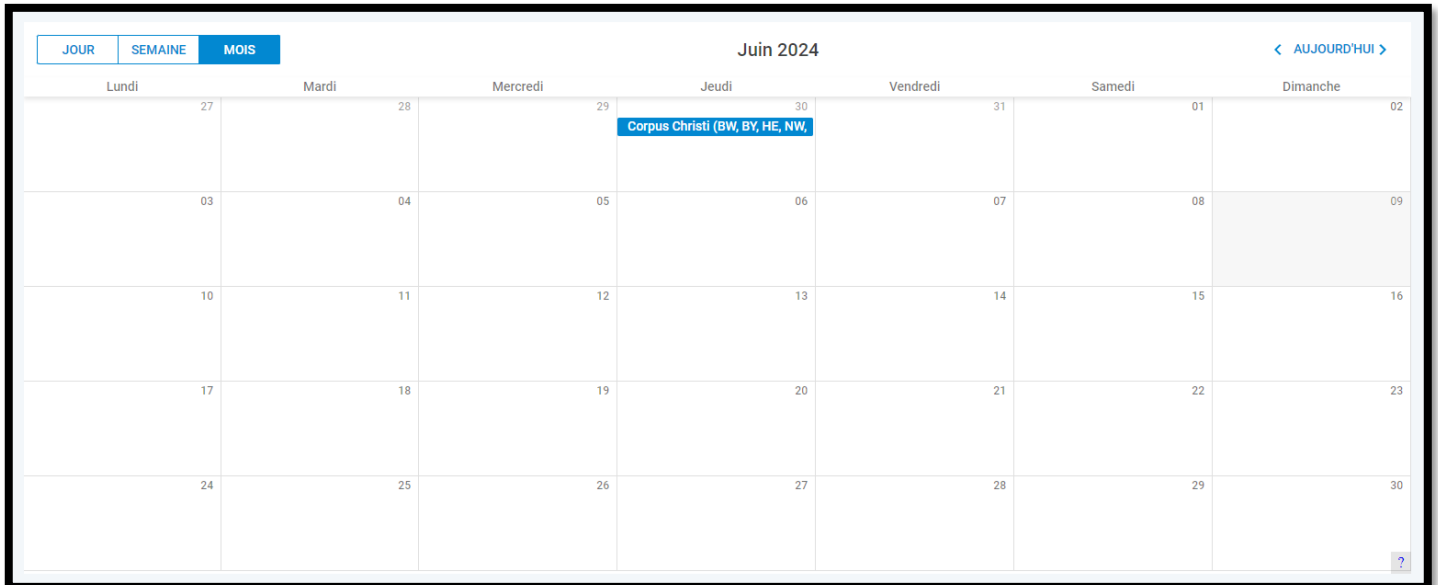
[Ajouter au calendrier](#)



Escapopes de dindes minceur aux champignons

Calories : 263kcal

Pour le calendrier, nous avons trouvé un open-web-calendar sur github (source : <https://github.com/niccokunzmann/open-web-calendar>). En effet, à notre niveau nous ne savons pas encore comment implémenter ce genre de fonctionnalité, c'est pourquoi nous nous sommes dirigés vers ce moyen.



À propos

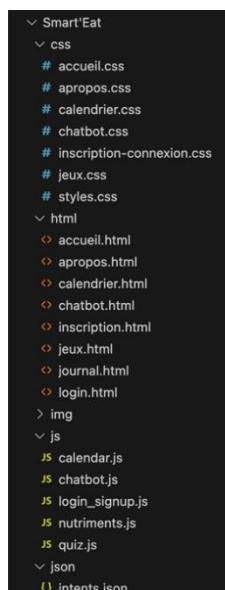
Cette page est là pour nous présenter et expliquer notre site web. On y trouve aussi une liste de fonctionnalités à venir.

Organisation du code

Nous avons réparti notre code en 5 dossiers (css, html, js, json, img). Le dossier html contient un fichier pour chaque page et notre dossier css contient un fichier styles.css commun à toutes les pages et des fichiers de styles uniques pour chacune des pages.

Notre dossier JavaScript contient :

- Un fichier nutriments.js qui sert à faire fonctionner l'ajout des glucides, lipides et protéines aux camemberts.
- Un fichier chatbot.js similaire à celui du TP4.
- Un fichier login_signup.js pour remplir les formulaires d'inscription et de connexion.
- Un fichier calendar.js qui permet de faire un tri des repas (végan, végétarien ou carnivore).
- Un fichier quiz.js pour faire fonctionner les boutons et afficher un score après avoir soumis les réponses.



Notre dossier json contient le fichier intents.json où sont stockées les questions et réponses auxquelles le chatbot sait/peut répondre.

Notre dossier img contient un dossier icons avec toutes les icônes des différentes pages, un autre dossier Plat qui sert à afficher les plats proposés sur la page calendrier et pour finir nous avons 3 images qui servent pour le logo la page de connexion et pour la page journal.

Conclusion

Ce projet nous a permis de réellement nous projeter sur la création d'un site web. Nous avons appris à bien organiser notre code en respectant les règles W3C. Le JavaScript était une épreuve à certains moments, mais cela nous a permis de mieux comprendre ce langage. La réalisation de ce site nous a aussi permis de capter l'attention de notre jury lors de notre évaluation du projet transverse, en montrant un projet concret à l'aide de notre site même si celui-ci n'est pas encore dynamique, cela reste toujours qu'un prototype. Enfin, nous en garderons un très bon souvenir.