



NOM	BENKIRANE
Prénom	DRISS
Date de naissance	12/09/1992

PARTIE 1 : REFLEXION INITIALE SUR LE SUJET

La réflexion initiale technologique sur l'application web que l'on nous a donné à développer se doit d'être un mélange entre ce que l'on nous a appris en formation jusque-là, ce que j'en avais compris en pratiquant, en lisant, et grâce à la documentation officielle présente sur le web sur les technologies aborder en cours. Une période de deux mois et demi s'est écoulé entre le début des cours, et à ce moment je n'avais que très peu de connaissance en matière de programmation, et le début de notre Évaluation en Cours de Formation. Une période pendant laquelle les technologie front-end de programmation ont été profondément étudié et pratiqué grâce à des petits projets tel que le « projet fil rouge ». Le « projet fil rouge » m'a donné une bonne base pour me lancer dans la programmation d'une application et savoir comment structurer l'arborescence d'un site web. J'ai naturellement voulu m'inspirer de ce travail pour développer le projet d'Eco-ride. En parallèle, quand le sujet d'Eco-ride est sorti, cela faisait quelque semaine que je travaillé sur comment développer la partie back-end d'une application. Autant dire que je ne comprenais pas tout à fait certains aspects du back-end tellement cela est riche et compliqué à saisir pour un débutant. Ne sachant pas comment m'y prendre, testant un à un les Framework et manière de se connecter à une banque de donné, je me suis heurté à un mur complexe de configuration. J'ai donc penché ma réflexion vers quelques formateur, présent en ligne, sur YouTube, pour élargir ma compréhension sur les sujets qui étais à mes yeux important et pourquoi pas apprendre des choses en plus. La chaine Grafikart a retenu mon attention, grâce à ses vidéo captivantes, simple d'application, sur des thèmes très intéressant, j'ai réussi à élargir mon champ de pratique et comprendre la programmation de manière plus pédagogique. Après quelques échec et réussite avec le Framework Symfony, même s'il est vu en cours, l'assimilation de son utilisation, est à première vue très compliqué à intégrer,



L'arborescence est riche et complexe, on s'y perd très facilement en tant que junior. J'ai donc penché ma réflexion vers un back-end plus simple, et en même temps sécurisé avec PHP avec l'utilisation de la PDO. La PDO était une manière recommander de faire. En effet, dans l'exemple de stack technique possible, PDO était dans la liste des stack technique back-end. Je n'avais alors aucune idée de la syntaxe et du fonctionnement de celui-ci. Je me suis documenté pour comprendre, regarder des vidéos explicatives, et décide de lancer les premiers tests. Non seulement, je sens la technologie plus simple que Symfony, mais aussi, je me suis tout de suite senti à l'aise avec l'utilisation de ce dernier. J'ai donc décidé après plusieurs recherches sur quel SGBD utilisé, d'utilisé La PDO, pour mon application avec DB Browser for SQLite. Je suis donc parti sur une structure simple en HTML, CSS et javascript pour le front-end comme vu dans le « projet fil rouge » qui m'a donné quelques cordes pour travailler, et l'utilisation de la PDO en back-end avec PHP, comme conseillé dans la liste des stacks technique possible. Un choix réconfortant, car ce sont des technologies avec lesquelles je me suis senti à l'aise avec leur logique.

PARTIE 2: CONFIGURATION DE VOTRE ENVIRONNEMENT DE TRAVAIL

J'ai travaillé sur ce projet web, sur une machine apple, un macbook Air, sequoia 15.3.1. avec l'aide de manager OSX pour travailler en local sur VS code. Pour la partie programmation et langages, voici la liste de mon stack technique :

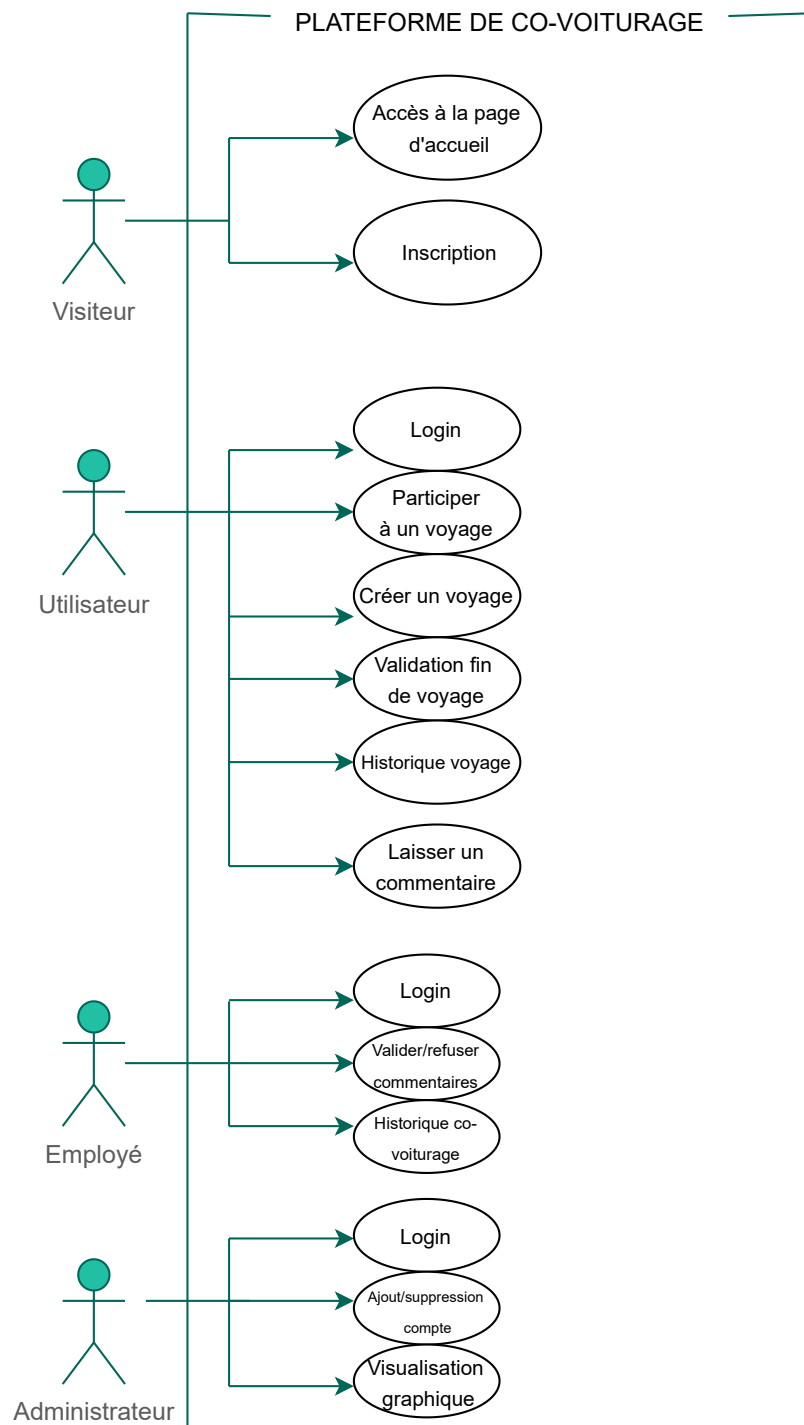
- Front-end : HTML5, CSS, JAVASCRIPT
- Back-end : PHP avec utilisation de PDO
- Base de donnée : DB Browser for sqlite
- Déploiement : infomaniak (FTP)

Pour plus d'information sur les versions utilisées, voir la partie Information complémentaires.



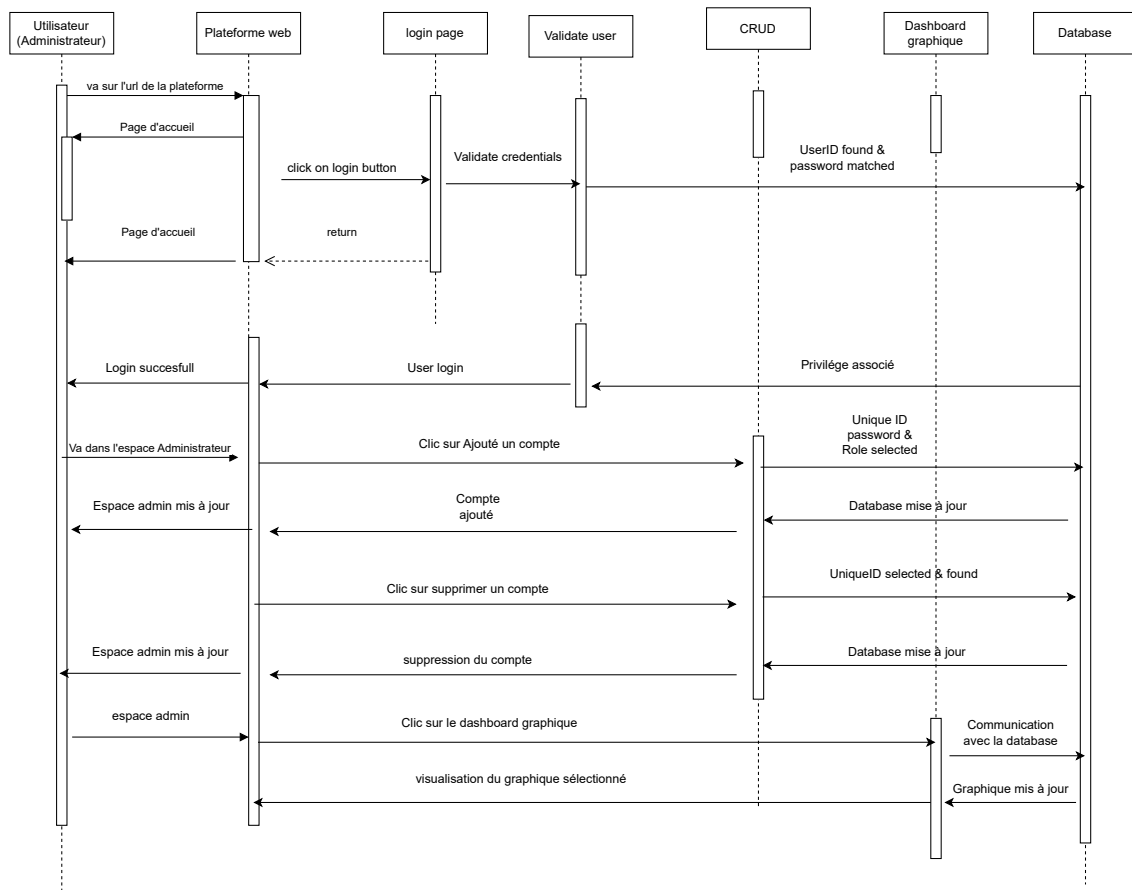
PARTIE 3: DIAGRAMMES

1- Diagramme d'utilisation



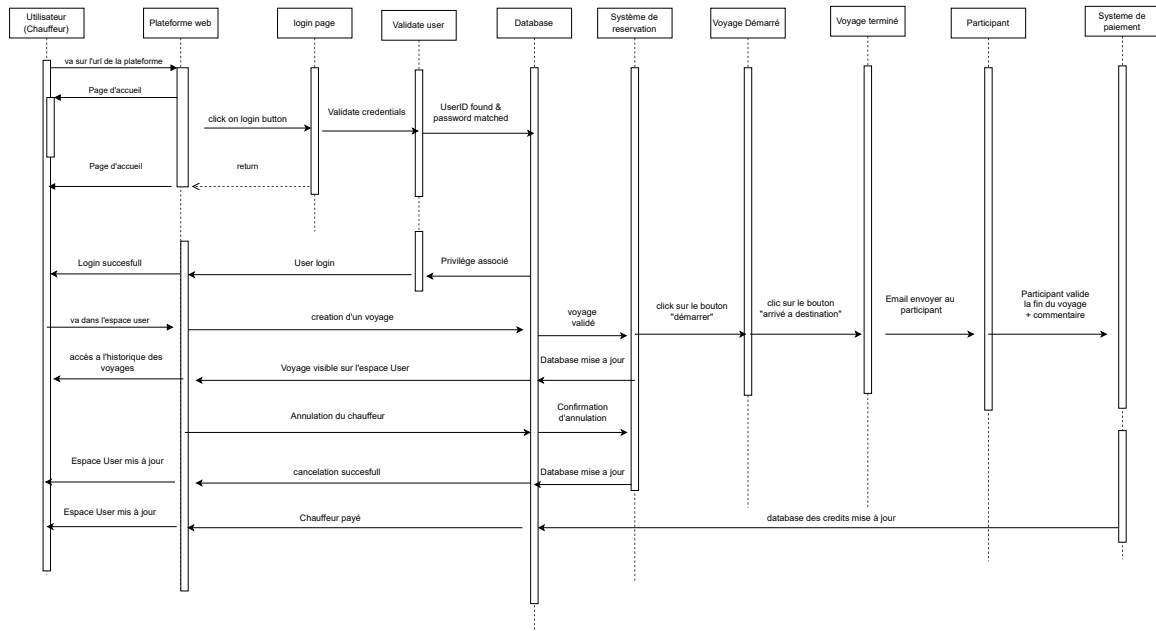


1- Diagramme administrateur

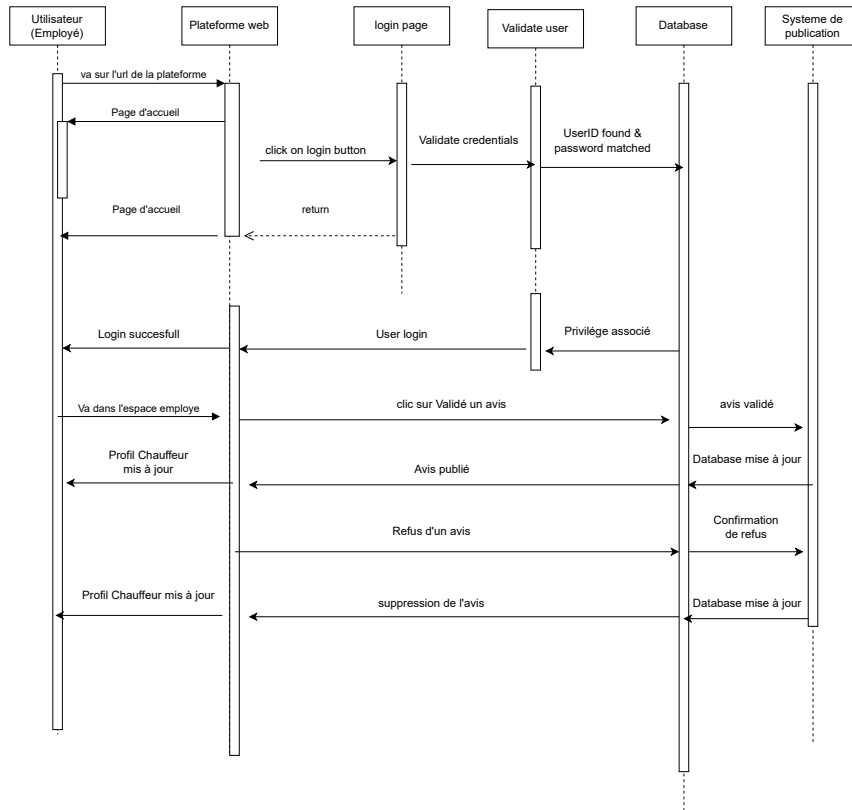




2- Diagramme chauffeur

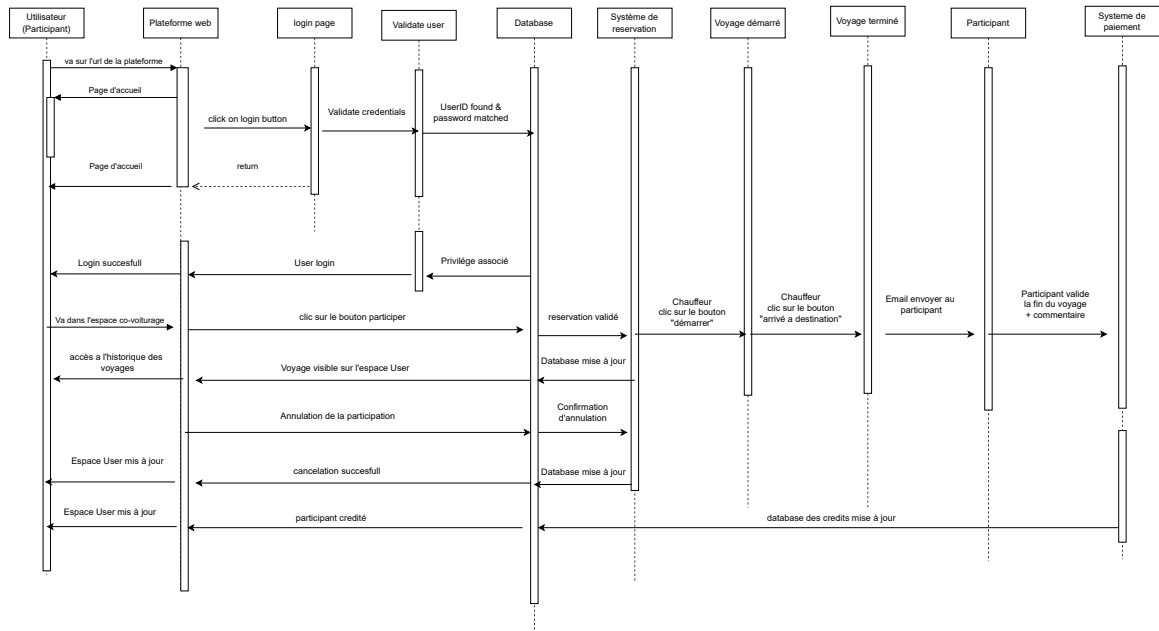


3- Diagramme employé



database des credits mise à jour

4- Diagramme participant



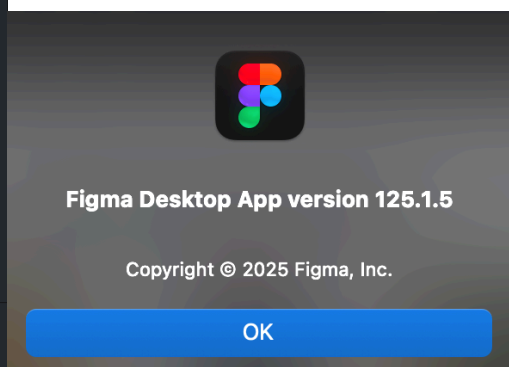
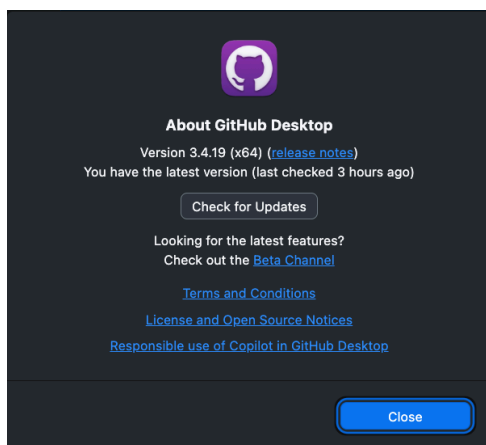
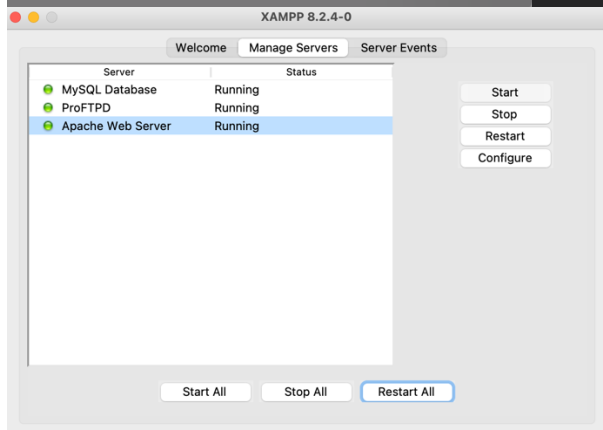
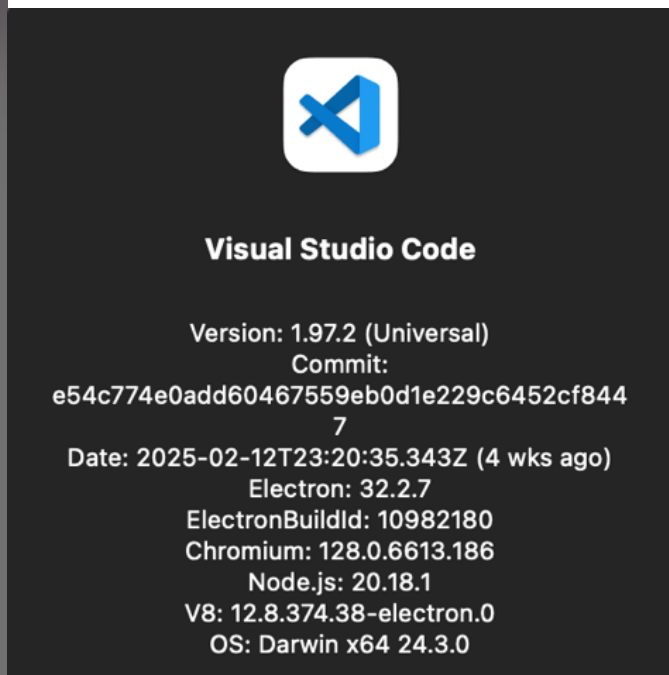
- 1- Pour la partie déploiement en local de l'application, J'ai d'abord créé un repository depuis l'application github desktop, connecter à mon compte github. Depuis github desktop j'ouvre le fichier sur vscode, tous les changements effectués sont alors visibles sur github desktop, je peux les « commit » depuis l'application puis les « push » vers github (voir informations complémentaires). J'ai aussi, crée mes branches et sous branches directement sur github, pour ensuite pouvoir versionner et naviguer facilement entre les fonctionnalités sur lesquelles je travaille depuis vs code ou github desktop. Grace à manager OS X, je peux simuler un serveur sur la machine en local. J'ai utilisé l'extension PHP server sur vs code, configurer le port, et utilisé le navigateur Chrome ainsi que Safari pour tester l'application en local et travailler dessus. Les versions des technologies utilisées sont documentées à la fin, voir la section informations complémentaires. La plupart des mises à jour ont été effectué en ligne de commande.

- 2- Pour la partie déploiement en ligne, j'ai utilisé un hébergement mutualisé web payant pour une durée de 365 jours, un cloud éthique, du nom de infomaniak, recommandé par de nombreux développeur professionnel et personnes du milieu de la tech. Le cloud propose un hébergement gratuit de 1 mois, ce qui m'a laissé le temps de tester mon application en ligne, de faire les ajustements nécessaires avant de me décider définitivement. L'environnement d'Infomaniak propose une base de données MariaDB. Ainsi que les accès à phpmyadmin, au web FTP et la certification SSL pour authentifier et sécurisé le site web. Les tables de la base de données ont été créé sur phpmyadmin, puisque l'hébergement mutualisé, donné accès à phpmyadmin. Une fois la base de données créé, Le contenu de l'hébergement sur le serveur a été déployé par le biais de Filezilla, par File Transfer Protocol. Filezilla a été plutôt facile à prendre en main, il suffit de drag and drop le fichier ou dossier voulu dans le dossier eco-ride.online de l'arborescence avec le fichier index.php à la racine du dossier. Pour modifier le fichier, un clic droit suffisait pour ouvrir et modifier le fichier sur vscode, enregistrer le travail terminer sur vs code puis confirmer sur filezilla l'enregistrement pour le télécharger sur le serveur.



PARTIE 5: INFORMATIONS COMPLÉMENTAIRES

1. Informations complémentaires.



```
macosdev@repeteurwifi6-9ef0 ~ % mysql --version
mysql Ver 9.2.0 for macos14.7 on x86_64 (Homebrew)
macosdev@repeteurwifi6-9ef0 ~ %

macosdev@repeteurwifi6-9ef0 ~ % node -v
v23.7.0
macosdev@repeteurwifi6-9ef0 ~ %

macosdev@repeteurwifi6-9ef0 ~ % npm -v
11.1.0
macosdev@repeteurwifi6-9ef0 ~ %

macosdev@repeteurwifi6-9ef0 ~ % php --version
Last login: Thu Mar 13 09:25:43 on ttys004
PHP 8.4.4 (cli) (built: Feb 11 2025 15:36:20) (NTS)
Copyright (c) The PHP Group
Built by Homebrew
Zend Engine v4.4.4, Copyright (c) Zend Technologies
    with Zend OPcache v8.4.4, Copyright (c), by Zend Technologies
macosdev@repeteurwifi6-9ef0 ~ %

macosdev@repeteurwifi6-9ef0 ~ % brew --version
Homebrew 4.4.24
macosdev@repeteurwifi6-9ef0 ~ %
```

♣ Modèle conceptuel de données (ou diagramme de classe)

Résultats de la validation W3C CSS de <https://eco-ride.online> (CSS niveau 3 + SVG)