



# Camagru

*Résumé : Le but de ce projet est de construire une application web.*

*Version : 4*

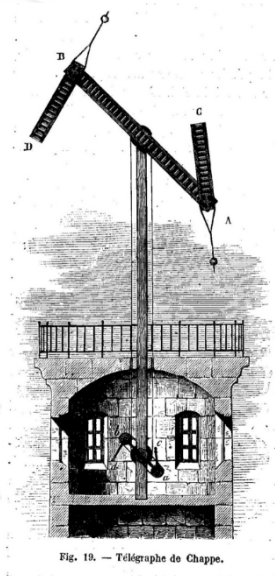
# Contenu

<b>I</b>	<b>Avant-propos</b>	<b>2</b>
<b>II</b>	<b>Introduction</b>	<b>3</b>
<b>III</b>	<b>Objectifs</b>	<b>4</b>
<b>IV</b>	<b>Instructions générales</b>	<b>5</b>
<b>V</b>	<b>Partie obligatoire</b>	<b>6</b>
V.1	Caractéristiques communes .....	6
V.2	Caractéristiques de l'utilisateur .....	7
V.3	Galerie des caractéristiques .....	7
V.4	Fonctions d'édition .....	8
V.5	Contraintes et éléments obligatoires .....	9
<b>VI</b>	<b>Partie bonus</b>	<b>10</b>
<b>VII</b>	<b>Soumission et évaluation par les pairs</b>	<b>11</b>

# Chapitre I Avant- propos

L'histoire de la communication est aussi vieille que l'histoire de l'Humanité et l'Homme a su la faire évoluer au fil des siècles à travers d'incroyables révolutions.

En 1794, Claude Chappe tente de résoudre le problème des communications à longue distance, limitées à l'époque à la vitesse des chevaux. Il met en place un ingénieux système de communication par télégraphe aérien pendant la Révolution française. Les tours de Chappe étaient couvertes par un mât mobile qui pouvait être vu avec des jumelles depuis la tour voisine la plus proche, située à environ 10 à 15 km.



La ligne Paris-Lille a été mise en service la même année et permet des transmissions entre les deux villes à une vitesse de 9 minutes par lettre à travers 15 tours. Bien entendu, le temps de transmission dépend de sa longueur.

En 1844, 534 tours étaient dispersées sur le territoire français reliant sur plus de 5000km, les villes les plus importantes.

Mais ce système présentait deux inconvénients majeurs : il ne pouvait pas fonctionner la nuit en raison, évidemment, de la mauvaise visibilité, et le nombre d'opérateurs par tour (2 tous les 15 km).

Heureusement, nous sommes au 21<sup>e</sup> siècle.

# Chapitre II

## Introduction

Vous êtes maintenant prêt à construire vos premières applications web, comme des pros. Si vous ne le saviez pas, le web est un monde vaste et riche, qui vous permet de diffuser rapidement des données et du contenu à tout le monde dans le monde entier.

Maintenant que vous connaissez les bases, le moment est venu de quitter ces vieilles listes de tâches et ces sites web de commerce électronique, et de vous envoler vers de plus grands projets.

C'est aussi le moment pour vous de découvrir de nouvelles notions et la beauté de :

- Conception adaptée
- Manipulation du DOM
- Débogage SQL
- Falsification des requêtes intersites
- Partage des ressources entre pays d'origine
- ...

# Chapitre III

## Objectifs

Ce projet web vous met au défi de créer une petite application web vous permettant faire de l'édition photo et vidéo de base en utilisant votre webcam et quelques images prédéfinies.



Il est évident que ces images doivent avoir un canal alpha, sinon votre superposition n'aurait pas l'effet escompté !

Nous allons, par exemple, imaginer le moment précis d'un lancement de chat intergalactique, en voici la preuve :



Les utilisateurs de l'application devraient pouvoir sélectionner une image dans une liste d'images superposables (par exemple un cadre photo, ou d'autres objets dont on ne veut pas savoir à quoi ils servent), prendre une photo avec leur webcam et admirer le résultat qui devrait être le mélange des deux images.

Toutes les images capturées devraient être publiques, aimables et commentables.

# Chapitre IV

## Instructions générales

- Ce projet sera corrigé par des humains uniquement. Vous pouvez organiser et nommer vos fichiers comme bon vous semble, mais vous devez respecter les règles suivantes.
- Votre application web ne doit produire aucune erreur, aucun avertissement ou ligne de journal dans n'importe quelle console, côté serveur et côté client. Néanmoins, en raison de l'absence de HTTPS, toute erreur liée à `getUserMedia()` est tolérée.
- Vous êtes libre d'utiliser n'importe quel langage pour créer votre application côté serveur, mais, pour chaque fonction que vous utilisez, vous devez vérifier qu'un équivalent existe dans la bibliothèque standard de PHP.
- Côté client, vos pages doivent utiliser HTML, CSS et JavaScript.
- Une conteneurisation à jour est indispensable.
- N'oubliez pas que certains choix peuvent vous rendre plus attractif sur le marché du travail.
- Tous les frameworks, micro-framework ou bibliothèques que vous ne créez pas et qui n'ont pas d'équivalent dans la bibliothèque standard de PHP sont totalement interdits, à l'exception des frameworks CSS qui ne nécessitent pas de JavaScript interdit.
- Votre application doit être exempte de toute faille de sécurité. Vous devez traiter au moins les cas mentionnés dans la partie obligatoire. Néanmoins, vous êtes encouragés à aller plus loin dans la sécurité de votre application, pensez à la confidentialité de vos données !
- Vous êtes libre d'utiliser le serveur web de votre choix, comme Apache, Nginx ou même le serveur web intégré<sup>1</sup>.
- Votre application web doit au moins être compatible avec Firefox ( $\geq 41$ ) et Chrome ( $\geq 46$ ).



Pour des raisons évidentes de sécurité, toutes les informations d'identification, clés API, variables env, etc... doivent être sauvegardées localement dans un fichier `.env` et ignorées par git. Les informations d'identification stockées publiquement vous conduiront directement à l'échec du projet.

---

<sup>1</sup> <http://php.net/manual/en/features.commandline.webserver.php>

# Chapitre V Partie obligatoire

## V.1 Caractéristiques communes

Vous allez développer une application web. Même si cela n'est pas nécessaire, essayez de structurer votre application (comme une application MVC, par exemple).

Votre site web doit avoir une bonne mise en page (c'est-à-dire au moins un en-tête, une section principale et un pied de page), pouvoir s'afficher correctement sur les appareils mobiles et avoir une mise en page adaptée aux petites résolutions.

Tous vos formulaires doivent être correctement validés et l'ensemble du site doit être sécurisé. Ce point est OBLIGATOIRE et sera vérifié lors de l'évaluation de votre demande. Pour avoir idée, voici quelques éléments ne sont pas considérés comme sécurisés :

- Stocker les mots de passe en clair ou non cryptés dans la base de données.
- Offrir la possibilité d'injecter du HTML ou du JavaScript "utilisateur" dans des variables mal protégées.
- Offrir la possibilité de télécharger du contenu indésirable sur le serveur.
- Offrir la possibilité de modifier une requête SQL.
- Utiliser un formulaire externe pour manipuler des données dites privées

## **V.2 Caractéristiques de l'utilisateur**

- L'application doit permettre à un utilisateur de s'inscrire en demandant au moins une adresse électronique valide, un nom d'utilisateur et un mot de passe d'un niveau de complexité minimal.
- À la fin du processus d'enregistrement, l'utilisateur doit confirmer son compte au moyen d'un lien unique envoyé à l'adresse électronique qu'il a indiquée dans le formulaire d'enregistrement.
- L'utilisateur doit alors pouvoir se connecter à votre application, en utilisant son nom d'utilisateur et son mot de passe. Il doit également pouvoir demander à l'application d'envoyer un message de réinitialisation du mot de passe, s'il oublie son mot de passe.
- L'utilisateur doit pouvoir se déconnecter en un seul clic, à tout moment et sur n'importe quelle page.
- Une fois connecté, l'utilisateur peut modifier son nom d'utilisateur, son adresse e-mail ou son mot de passe.

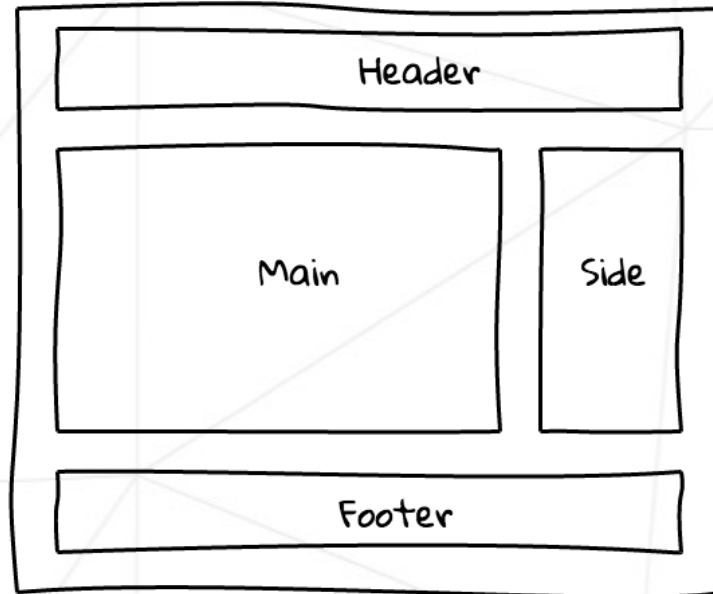
## **V.3 Caractéristiques de la galerie**

- Cette partie doit être publique et doit afficher toutes les images éditées par tous les utilisateurs, classées par date de création. Elle doit également permettre (uniquement) à un utilisateur connecté de les aimer et/ou de les commenter.
- Lorsqu'une image reçoit un nouveau commentaire, l'auteur de l'image doit en être informé par courrier électronique. Cette préférence doit être définie comme vraie par défaut mais peut être désactivée dans les préférences de l'utilisateur.
- La liste des images doit être paginée, avec au moins 5 éléments par page.



## V.4 Fonctions d'édition

Figure V.1 : Juste une idée de la mise en page de la page d'édition



Cette partie ne doit être accessible qu'aux utilisateurs authentifiés/connectés et doit rejeter de manière générale tous les autres utilisateurs qui tentent d'y accéder sans avoir réussi à connecter.

Cette page doit contenir deux sections :

- Une section principale contenant l'aperçu de la webcam de l'utilisateur, la liste des images superposables et un bouton permettant de capturer une image.
- Une section latérale affichant des vignettes de toutes les photos prises précédemment.

La mise en page doit normalement ressembler à la figure V.1.

- Les images superposables doivent pouvoir être sélectionnées et le bouton permettant de prendre la photo doit être inactif (non cliquable) tant qu'aucune image superposable n'a été sélectionnée.
- La création de l'image finale (donc entre autres la superposition des deux images) doit se faire du côté du serveur.
- Comme tout le monde ne dispose pas d'une webcam, il est préférable de permettre le téléchargement d'une image d'utilisateur plutôt que d'en capturer une à l'aide de la webcam.
- L'utilisateur devrait pouvoir supprimer ses images modifiées, mais uniquement les siennes, pas celles des autres utilisateurs.

## V.5 Contraintes et éléments obligatoires

En résumé, votre application Über doit respecter les choix technologiques suivants :

- Langues autorisées :
  - **[Serveur]** Tout (limité à la bibliothèque standard de PHP)
  - **[Client]** HTML - CSS - JavaScript (uniquement avec l'API native du navigateur)
- Cadres autorisés :
  - **[Serveur]** Tout (jusqu'à la bibliothèque standard PHP)
  - **[Client]** Les cadres CSS sont tolérés, à moins qu'ils n'ajoutent du JavaScript interdit.

Votre projet doit contenir impérativement :

- Un (ou plusieurs) conteneur pour déployer votre site avec une seule commande. Tout ce qui est équivalent à docker-compose est ok.

# Chapitre VI Partie

## bonus

Si la partie obligatoire est entièrement et parfaitement réalisée, vous pouvez ajouter tous les bonus que vous souhaitez ; ils seront évalués par vos évaluateurs. Vous devez cependant respecter les exigences des parties bonus (par exemple, le traitement de l'image doit être effectué sur le serveur).

Si vous manquez d'inspiration, voici quelques pistes :

- "AJAXifier" les échanges avec le serveur.
- Proposer un aperçu en direct du résultat édité, directement sur l'aperçu de la webcam. Il est à noter que cette opération est beaucoup plus simple qu'il n'y paraît.
- Faire une pagination infinie de la partie galerie du site.
- Offrir la possibilité à un utilisateur de partager ses images sur les réseaux sociaux.
- Rendre un GIF animé.



La partie bonus ne sera évaluée que si la partie obligatoire est PARFAITE. Parfait signifie que la partie obligatoire a été intégralement réalisée et qu'elle fonctionne sans dysfonctionnement. Si vous n'avez pas satisfait à TOUTES les exigences obligatoires, votre partie bonus ne sera pas évaluée du .

# **Chapitre VII**

## **Soumission et évaluation par les pairs**

Rendez votre travail dans votre dépôt Git comme d'habitude. Seul le travail contenu dans votre dépôt sera évalué lors de la soutenance. N'hésitez pas à vérifier les noms de vos dossiers et fichiers pour vous assurer qu'ils sont corrects.