RAPPORT DE STAGE POUR LA CERTIFICATION DEVELOPPEUR WEB /WEB MOBILE

Audrey Bucholz

TABLE DES MATIERES

**I-MAQUETTER UNE APPLICATION3**

1-Etablir le cahier des charges3

### i-Présentation du Projet …………………………………………………………………………………………….. 3

### ii-Description graphique et ergonomique ………………………………………………………………….. 3

### iii-Description technique et fonctionnelle ………………………………………………………………….. 4

### iv-Prestations attendues ……………………………………………………………………………………………. 5

2-Maquetter l’application5

**II-REALISER UNE INTERFACE STATIQUE ET ADAPTABLE6**

1-Interface statique 6

2-Interface adaptable7

**III-REALISER UNE INTERFACE DYNAMIQUE** ..……………………………………………………………………………………. **9**

1-Intégration des éléments de sécurité ..………………………………………………………………………………………. 9

2-Affichage dynamique ..…………………………………………………………………………………………………………….. 10

**I. Maquetter une application :**

La maquette est un élément important dans la conception d’un site web, elle permet au client d’avoir une ébauche de son site et de se faire un premier avis. C’est aussi un élément essentiel pour le développeur car il lui permet d’avoir une base de travail pour créer l’application.

Pour créer la maquette le développeur a besoin de connaître les besoins du client ; c’est pour cela qu’il établira un cahier des charges.

1-Etablir le cahier des charges :

i-Présentation du projet :

Il faut tout d’abord connaître l’entreprise et le but de l’application. Ici l’entreprise est une micro-entreprise dont le but est la diffusion de contenu sur internet via différentes plateformes.

Ensuite il faut connaître les objectifs principaux du site : dans notre cas le but est de faire la publicité du contenu proposés.

Puis il faut identifier la cible du site. La cible sera des personnes jeunes qui sont intéressés par les médias d’internet (Youtube, Twitch)

Après il faut avoir un objectif quantitatif. En effet le volume de visiteurs est important pour la création et la mise en place du site. Si le volume est important il faudra avoir un serveur plus important.

Aussi il faut savoir la « portée » du site, en effet si on cible un public étranger il faudra prévoir une traduction dans différents langages. Dans notre cas le public ciblé est exclusivement français donc il n’y a pas besoin de traduction.

ii-Description graphique et ergonomique :

Tout d’abord il faut savoir quelles couleurs seront sur le site. Ici le violet et le bleu ont été choisis. Aussi le bleu foncé a été choisi pour la partie admin du site. Cela permet de de bien différencier les deux parties.

Ensuite il faudra avoir les logos et différents contenus pour le site. L’entreprise m’a fourni ces ressources pour que je puisse créer le site.



Puis si le client a une idée précise de police il faudra les prendre en compte. Là nous sommes partis sur 3 polices d’écriture, le Calibri pour le corps du site, la police Chilanka-Regular les barres de navigation, pied de page et certains titre et la police Princetown pour un titre précis.

Enfin il faut connaître si le client veut un design particulier (par exemple : une barre de navigation en haut, un pied de page en bas…). Ici j’ai eu carte blanche pour la disposition des éléments du site.

iii-Description fonctionnelle et technique :

Si le client est un professionnel, il souhaitera peut-être une architecture particulière, par exemple en MVC (Modèle Vue Contrôleur). Je n’ai pas eu de d’attentes particulières vis-à-vis de ça je suis donc partie sur une architecture MVC

Aussi pour la description fonctionnelle on détermine les différents rôles s’il y a, et ce que peuvent faire ces derniers. Là il n’y aura qu’un seul rôle qui est le rôle admin, il va permettre de gérer les messages reçus dans le formulaire de contact.

S’il y a une authentification, il faudra choisir la manière (grâce à des site tiers, les champs utiles pour la connexion…). Dans notre cas l’authentification sera seulement pour l’administrateur du site et elle s’effectuera grâce à un email et un mot de passe.

Il faudra savoir les informations relatives aux contenus, par exemple comment sont-ils ajoutés ? Dans notre cas le contenu (par exemple : les vidéos Youtube) sera ajouté grâce notamment à des API

D’ailleurs il faudra prendre en compte les contraintes techniques. En effet il est possible que le client veuille des technologies particulières, des Framework précis ou pas. Pour cette application je n’ai eu aucune contrainte de technologie. J’ai donc choisi le PHP et Symfony pour faire le site. Cela me permet d’avoir plus d’expérience avec ce Framework PHP.

iv-Prestations attendues:

* Développement
* Design (sans les logos et images)
* Intégration

Il conviendra aussi d’établir un planning pour la création du projet

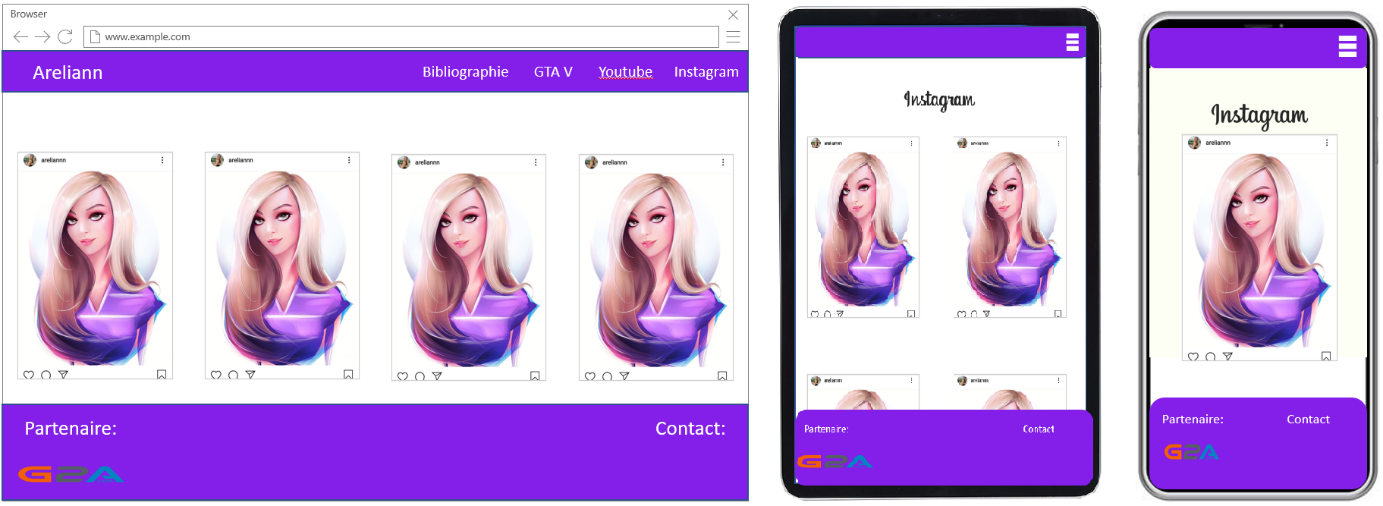
Enfin on choisira une méthodologie de suivi notamment des rendez-vous avec le client pour lui montrer l’avancée du projet.

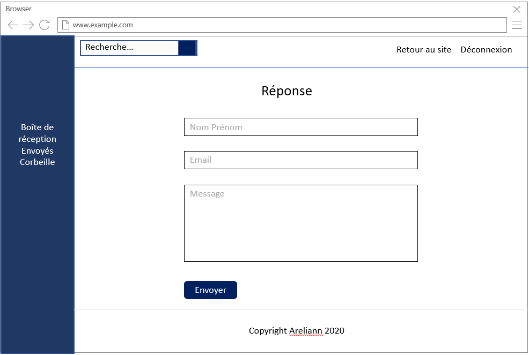
2-Maquetter l’application :

Une fois que le cahier des charges est créé, on pourra passer au maquettage de l’application. Cela consiste à établir une ébauche et d’établir les différents placements des parties constituant le site. Il faut créer une maquette pour les versions mobile, tablette et PC. Généralement on ne met pas les couleurs dans la maquette mais personnellement je les ai mise pour que le client ait une facilité à s’imaginer le rendu final.

Il existe différents logiciels pour effectuer le maquettage. Ici j’ai utilisé Power Mock’up qui est un plugin de PowerPoint.

Voici différentes images de la maquette effectuée :

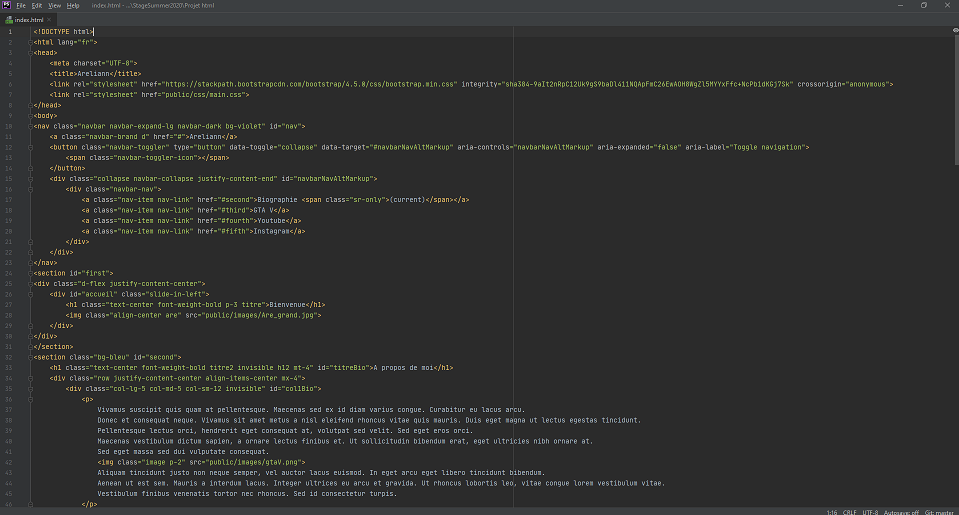




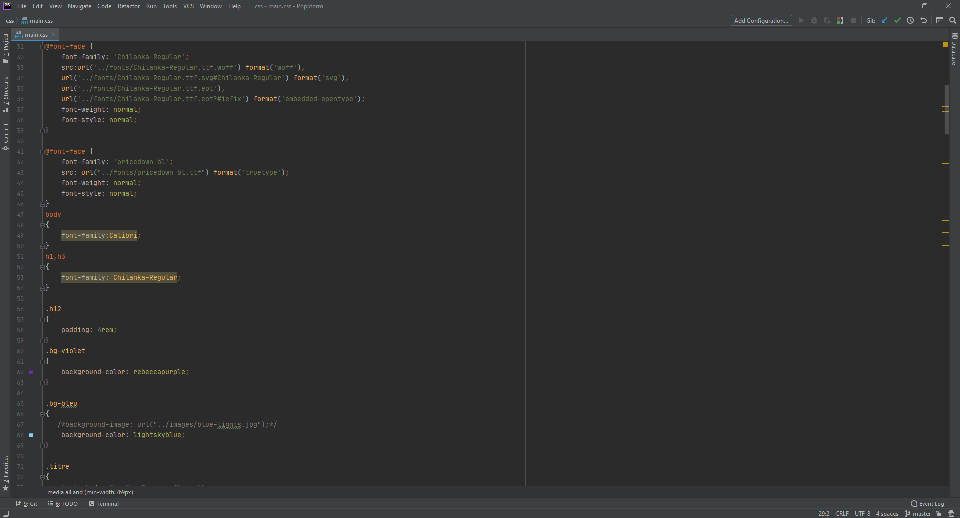
**II-Réaliser une interface utilisateur web statique et adaptable :**

1-Interface statique :

Pour effectuer une interface statique. On utilise le HTML et le CSS pour le style



Ici on peut voir la structure de la page HTML avec les balises <HTML>,<head>(pour notamment définir le Charset avec <meta>, lier une feuille de style avec la balise <link> ) et la balise <body> qui contient le contenu du site.



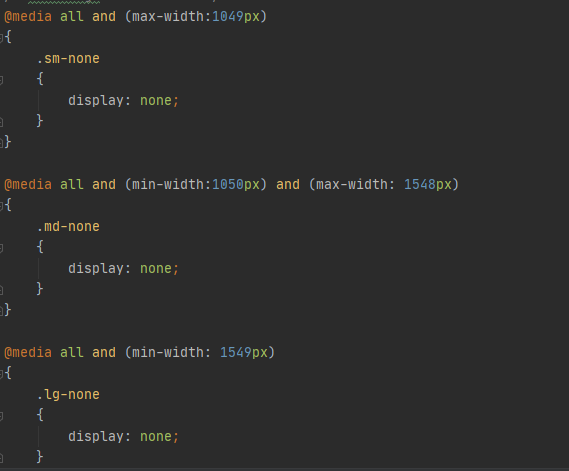
Sur cette image on peut voir une partie de la feuille de style que j’ai créée pour notamment intégrer des polices particulières (@font-face), ajouter des styles sur des balises, et créer des classes.

Pour mettre en place les éléments de manière statique on peut utiliser un module CSS qui s’appelle Grid . La page internet est quadrillée et on peut créer des lignes et des colonnes pour disposer nos éléments. De même il est possible de le faire avec Flexbox et des largeurs relatives.

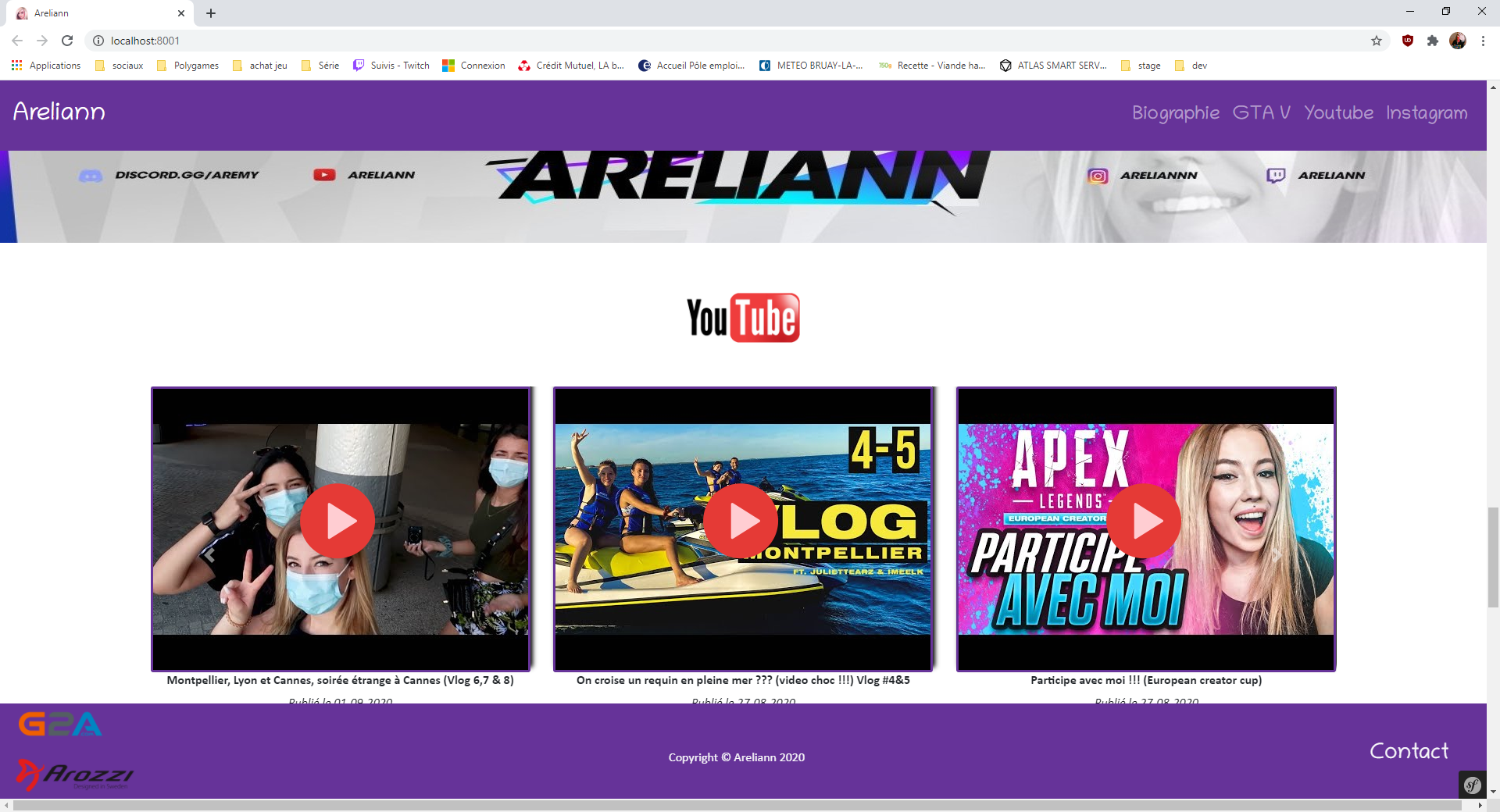
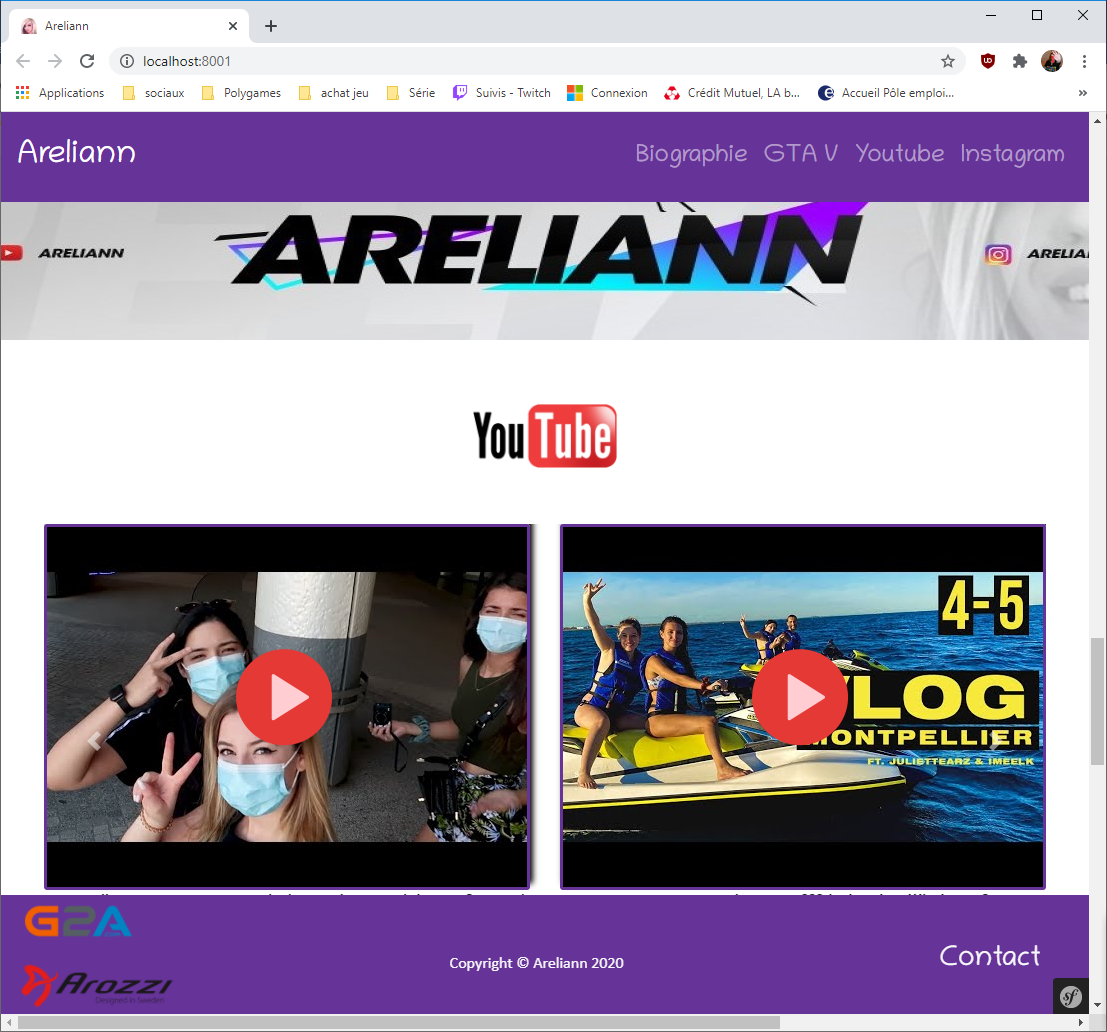
2-Interface adaptable :

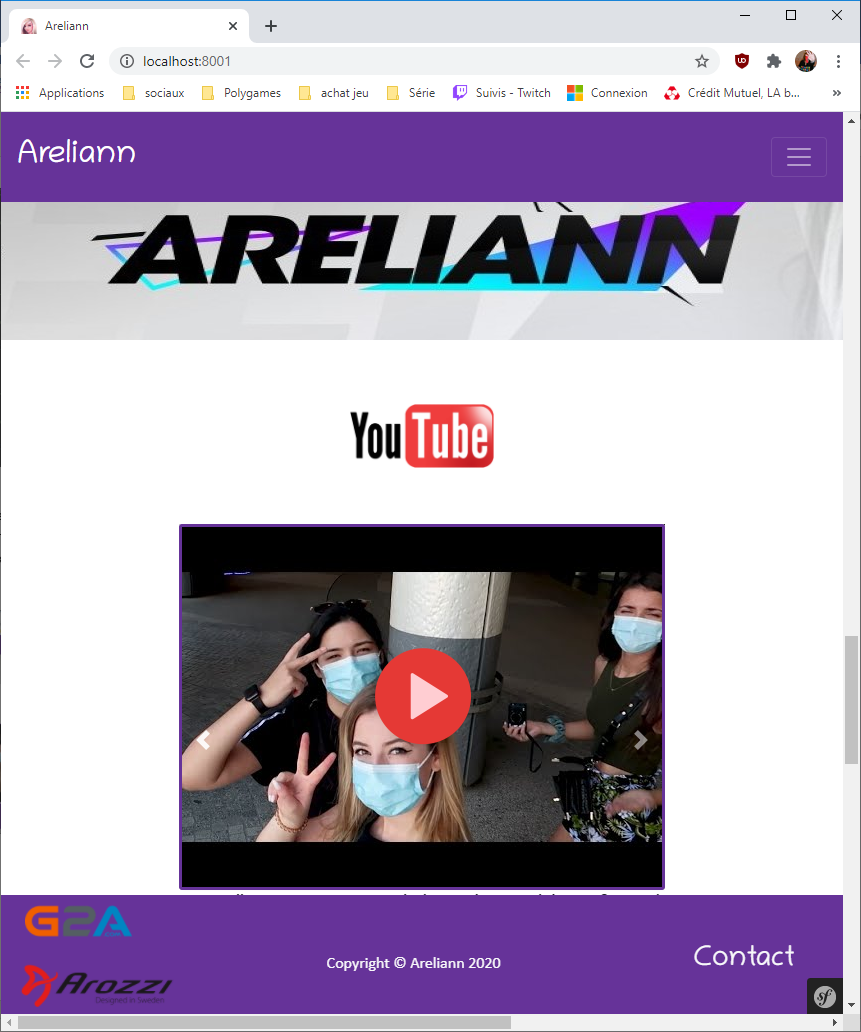
Avec l’expansion des smartphones de plus en plus de monde consulte les sites sur leur mobile, c’est pour cela que lorsqu’on crée un site adaptable on utilise la technique du Mobile First. On commence par concevoir la partie mobile du site et on développe sur des médias plus grands.

Pour avoir un site qui s’adapte on utilise les Media Queries. Ce sont des styles définis pour des cas prédéfinis.

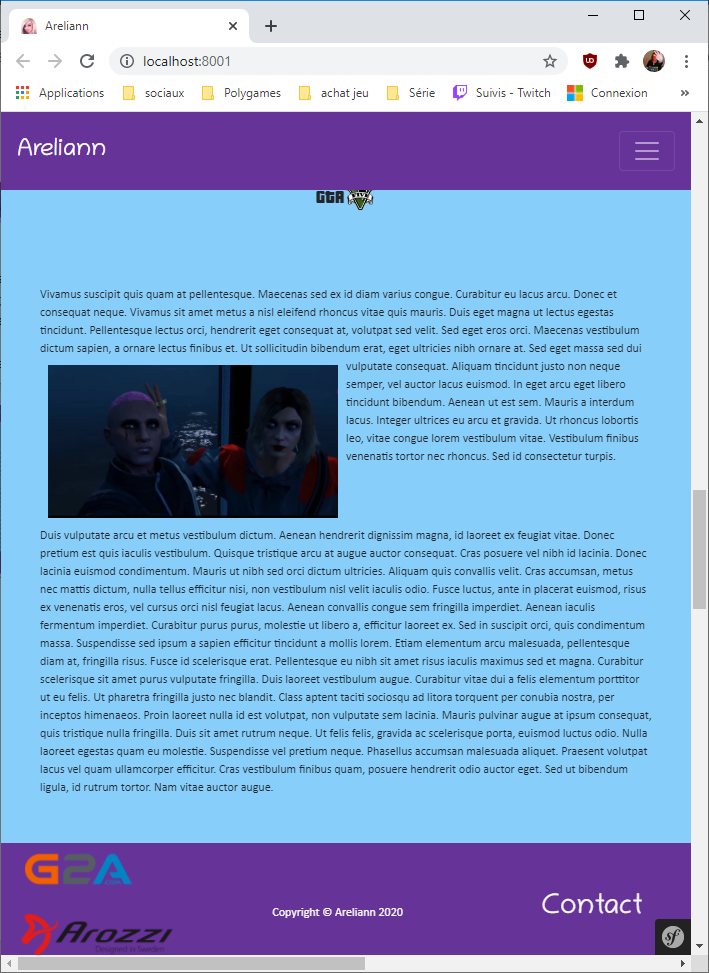
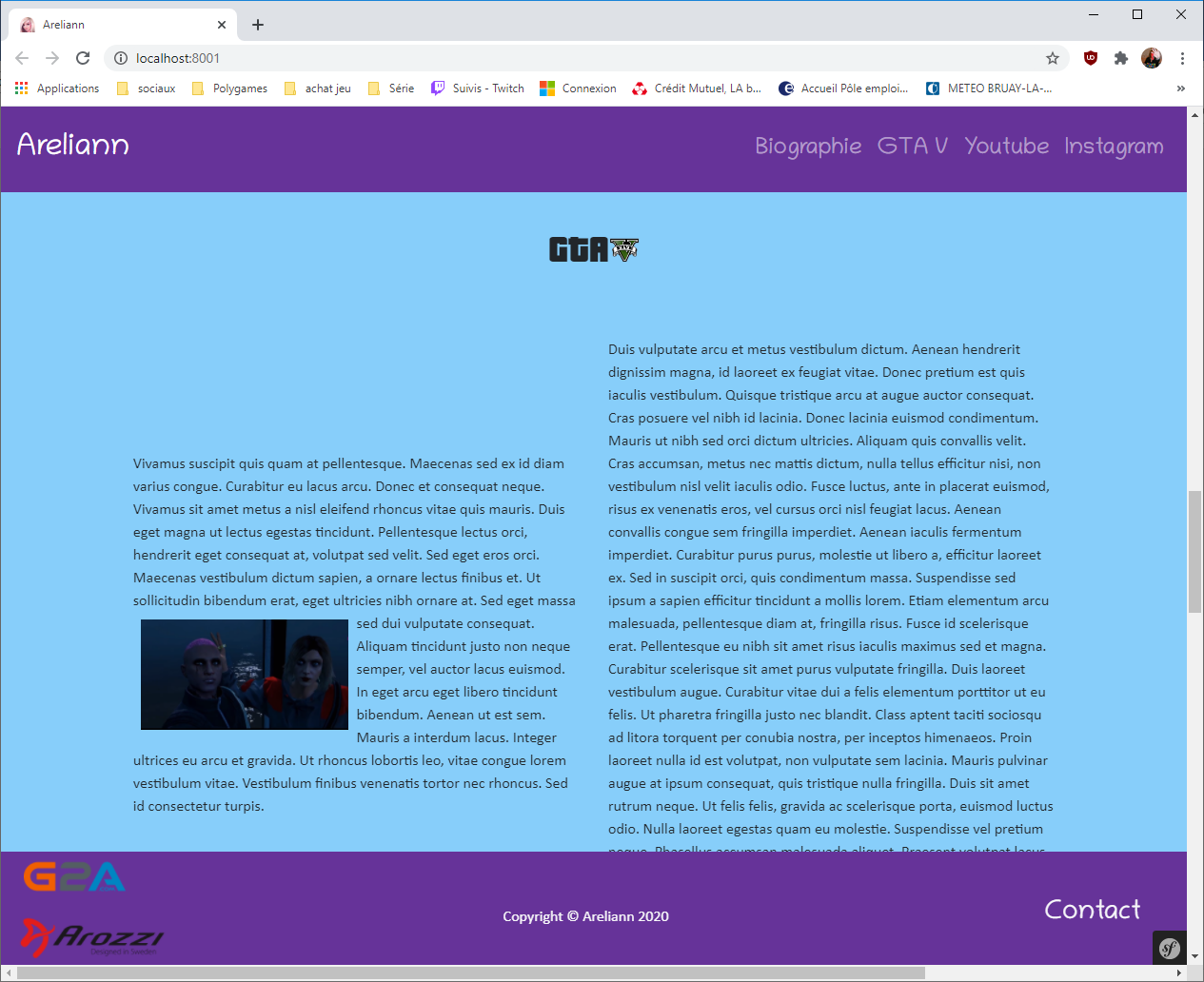
Sur cet exemple on peut voir que j’ai créé une classe « sm-none » qui existe lorsque n’importe quel média à une largeur maximale de 1049 px. On peut ainsi grâce à ça, créer une disposition des éléments différentes en fonction de la largeur que largeur que l’on a

Voilà le rendu, on a un carrousel à un élément pour les largeurs moins de 1049px, à 2 éléments pour les largeurs entre 1050 et 1548px et à3 éléments pour celle supérieur à 1549px





Pour nous faciliter la tâche des Framework CSS ont été créés (Ex : Bootstrap). Ils sont basés sur un système de « quadrillage » avec des lignes et des colonnes. Il utilise des Media Queries avec des points de rupture (breakpoint) La largeur est découpée en 12 colonnes. Dans les images ci-dessous les 2 paragraphes sont en col-lg-6 col-md-6 et col-sm-12. Donc en version PC et tablette il y a 2 colonnes width= 50% (6/12\*100) et en version mobile il n’y a qu’une seule colonne car la largeur est à 100%. Bootstrap est le plus populaire mais il existe d’autre Framework comme TailWind, Mobirise.



**III-Développer un interface web dynamique :**

Pour créer une interface dynamique il faut utiliser un langage qui sera interprété par le serveur où sera hébergé le site. Aussi on pourra utiliser JQuery pour dynamiser l’affichage avec des animations notamment.

1-Intégration des éléments de sécurité :

Faire des contrôles sur la sécurité du site, notamment sur les formulaires est un premier pas essentiel pour créer une interface web dynamique. En effet il faut éviter que l’utilisateur insère des scripts via les formulaires : c’est la faille XSS. Ici en PHP j’ai utilisé la fonction htmlspecialchars. Celle-ci permet de remplacer les caractères inférieur et supérieur par leur code et donc éviter d’avoir des blocs « <script>…</script> » C’est pour cela que l’on vérifie ce que l’utilisateur rentre avant de pouvoir l’utiliser.

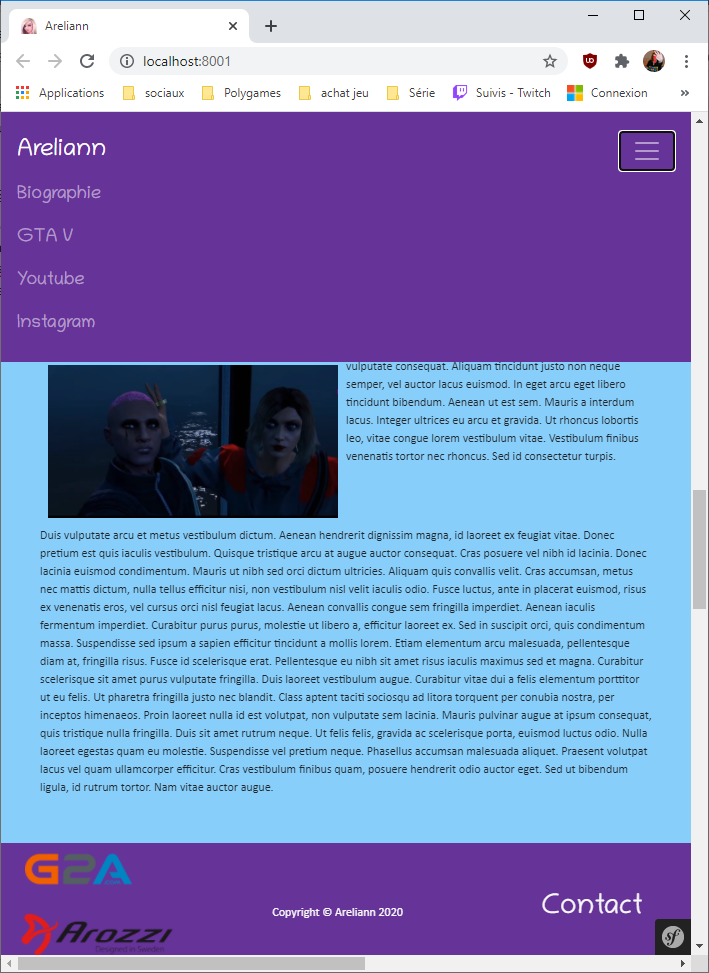
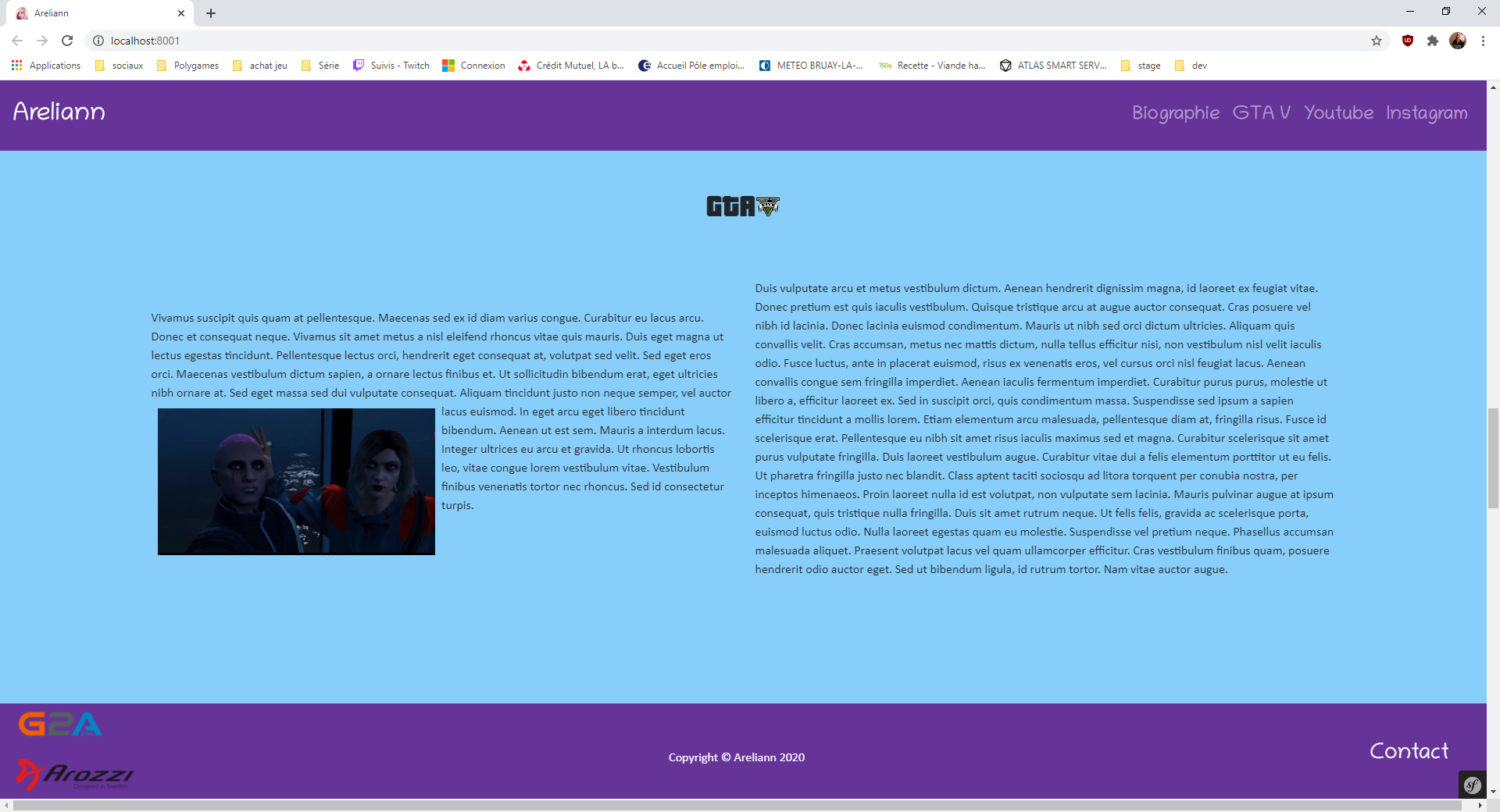
Ici on peut voir que dans mon Entité Contact j’ai appliqué dans mes setters, la fonction htmlspecialchars ainsi que la fonction trim qui permet de supprimer les espaces avant et après (cela permet de gagner de la place). Grâce à ça quand une personne va remplir le formulaire de contact, elle ne pourra pas insérer de script.

Ces vérifications sont aussi possibles avec Javascript et des Expressions Régulières.

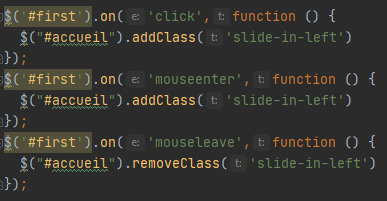
Aussi Javascript et notamment JQuery peuvent être utilisé pour afficher de manière dynamique.

2-Affichage dynamique :

On peut rendre notre site dynamique avec l’affichage d’éléments. Ici j’ai utilisé implicitement JQuery pour ma barre de navigation. En effet avec Bootstrap on peut créer une navbar qui s’affiche pleinement en grand écran et qui va afficher un bouton qui lorsque l’on clique sur celui-ci une liste des éléments de la barre s’affiche.



J’ai aussi utilisé JQuery pour afficher le texte quand on rentre dans zone et le cacher quand on sort de la zone (voir le site)

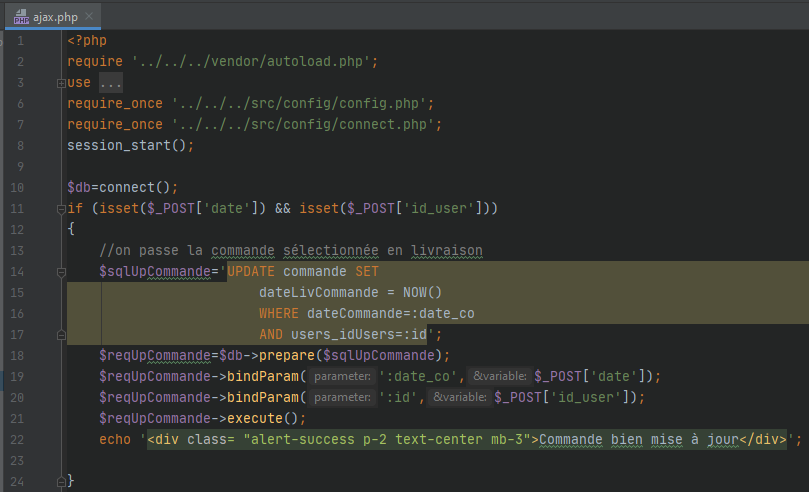


3-Méthode AJAX :

Asynchronous Javascript and XML :

De manière classique dès que l’utilisateur fais quelque chose, une requête est envoyée au serveur puis le serveur fais le calcul et envoi une réponse. En ajax, les demandes sont faites de manière asynchrone, le navigateur n’attend pas la réponse et l’utilisateur peut continuer à interagir avec la page pendant ce temps.

Dans un précédent projet j’ai utilisé Ajax et JQuery pour mettre en livraison des produits.



**IV-Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce :**