Table des matières

[I. Introduction 2](#_Toc21305434)

[II. Technologies utilisées 3](#_Toc21305435)

[III. Maquettes 5](#_Toc21305436)

[Version 1 6](#_Toc21305437)

[Version 2 7](#_Toc21305438)

[IV. Phases de réalisation 8](#_Toc21305439)

[Tache 1 8](#_Toc21305440)

[Tache 2 9](#_Toc21305441)

[Tache 3 11](#_Toc21305442)

[Tache 4 et 5 11](#_Toc21305443)

[V. Conclusion 13](#_Toc21305444)

[Points Clé 14](#_Toc21305445)

# Introduction

* Présentation de l’entreprise

* Contexte

Dans le but de proposer à son lectorat un moyen pratique d’accéder aux articles du site newsantilels.com, M.AZEDE nous a commandé à mon collègue Oliver Gallas et moi une application mobile.

Un entretien avec le client nous a permis d’identifier les fonctionnalités principales souhaitées et d’établir un cahier des charges :

* Un écran d’accueil (splash-screen) au lancement de l’application
* Les articles du site devront être disponibles en temps réel
* Les utilisateurs recevront une notification à la publication d’un nouvel article
* Les utilisateurs pourront naviguer entre les différentes rubriques du site
* La rubrique  « direct », liée à la page youtube de NewsAntilles sera toujours accessible par l’intermédiaire d’un bouton dans la barre de titre
* L’application devra respecter la charte graphique du site web

# Technologies utilisées

Cordova

l

Cordova est un framework open-source développé par la Fondation Apache. Il permet de créer des applications pour différentes plateformes (Android, IOS, etc…) en HTML, CSS et JavaScript. Les applications qui en résultent sont hybrides, ce qui signifie qu'elles ne sont ni vraiment natives, ni purement basées sur les langages.

GitHub

GitHub est une plateforme de versioning participatif. Elle permet de synchroniser le code source entre plusieurs collaborateurs et de garder un historique des différentes versions du code ainsi que différentes forks d’un même projet.

TeamViewer



TeamViewer est un logiciel de télémaintenance permettant d’interagir avec une machine à distance. Il nous a été particulièrement utile pour résoudre les problèmes de synchronisation de versions du code.

JQUERY

****JQUERY est une bibliothèque JavaScript, qui contient un ensemble de fonction permettant de faciliter l’écriture du code. JQUERY ne permet pas de faire plus de choses que le JavaScript basique, mais permet de raccourcir le code et de faciliter la lecture grâce à sa sémantique proche du langage humain.

CORS-Anywhere

CORS-Anywhere est une API qui permet d’effectuer des requêtes cross-origin.

Le CORS, pour «Cross-origin resource sharing » (en français : partage des ressources entre origines multiples) permet à un utilisateur d’interagir avec un serveur différent de celui sur lequel est hébergée l’application ou le site sur lequel il se trouve.

Par défaut, le navigateur rejette les requêtes CORS pour des raisons de sécurité, mais CORS-Anywhere nous permet de contourner ce problème.

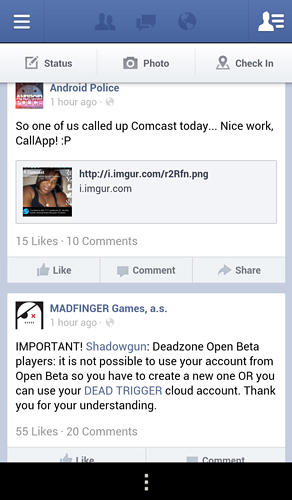
Moqups

ll

Moqups (apps.moqups.com) est une application en ligne permettant l’élaboration de maquettes grâce à une interface simple et intuitive.

# Maquettes

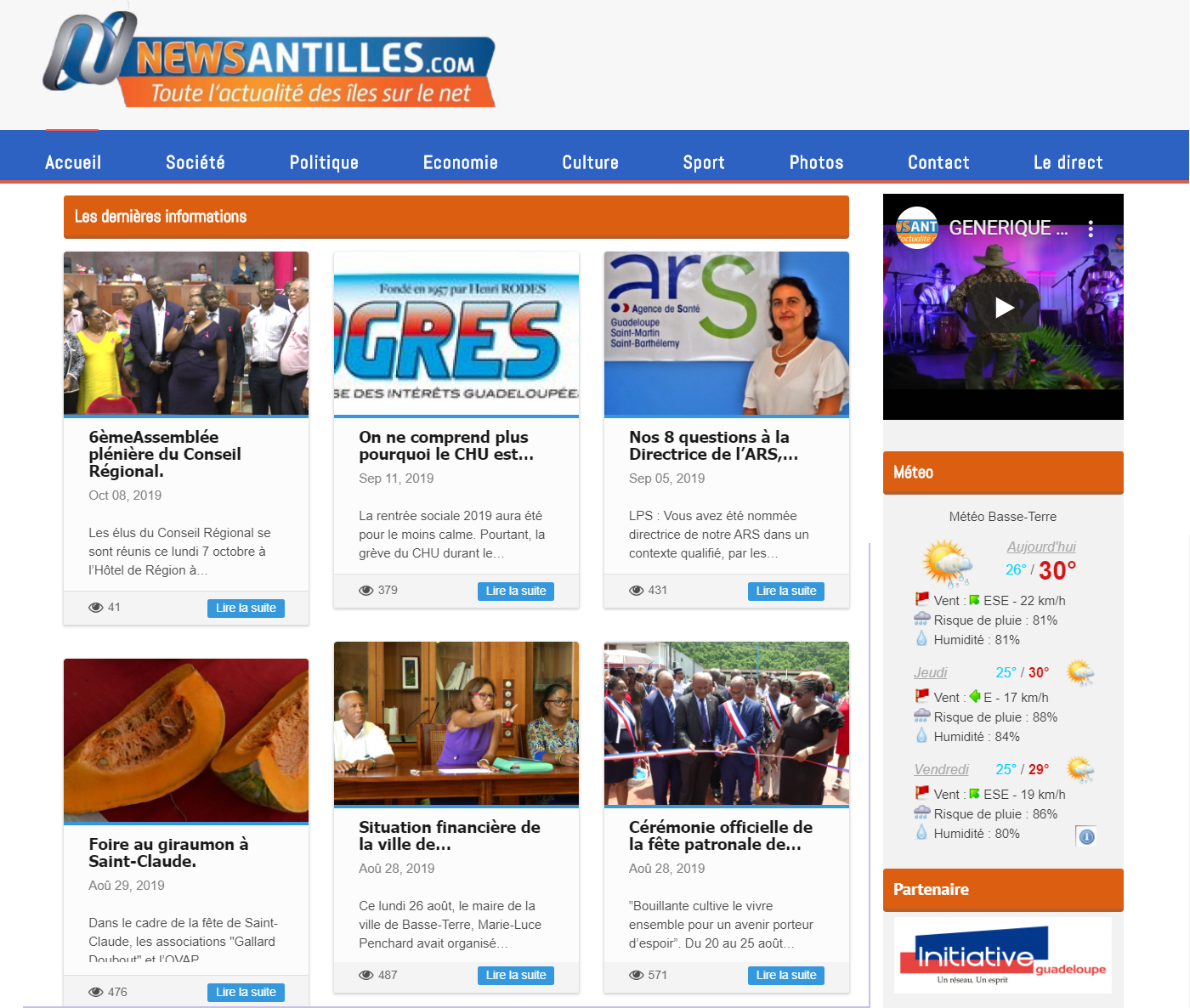
Après définition du cahier des charges, nous avons commencé à faire des recherches sur le design d’applications mobiles, en nous concentrant sur les applications de news et de flux RSS afin de mieux comprendre les habitudes d’utilisation de leurs usagers et les normes d’ergonomies fréquemment utilisées.





Nous nous sommes arrêtés sur un visuel proche de celui d’applications connues comme celles de facebook, twitter ou l’application BFMTV, avec un design sous forme de fil d’actualité défilant. Les titre des articles sont affichés dans des tuiles elles même présentés sous forme de liste, chaque tuile présente le titre de l’article et une miniature de la photo accompagnant celui-ci. Les tuiles devront être assez grandes pour pouvoir cliquer dessus avec le pouce sans déborder sur les tuiles voisines et les titres assez grands pour être lisible par les utilisateurs ayant des troubles visuels. Cliquer sur une tuile ouvre une page en plein écran où sera affiché l’article.

Ensuite, nous avons étudié la charte graphique du site NewsAntilles.

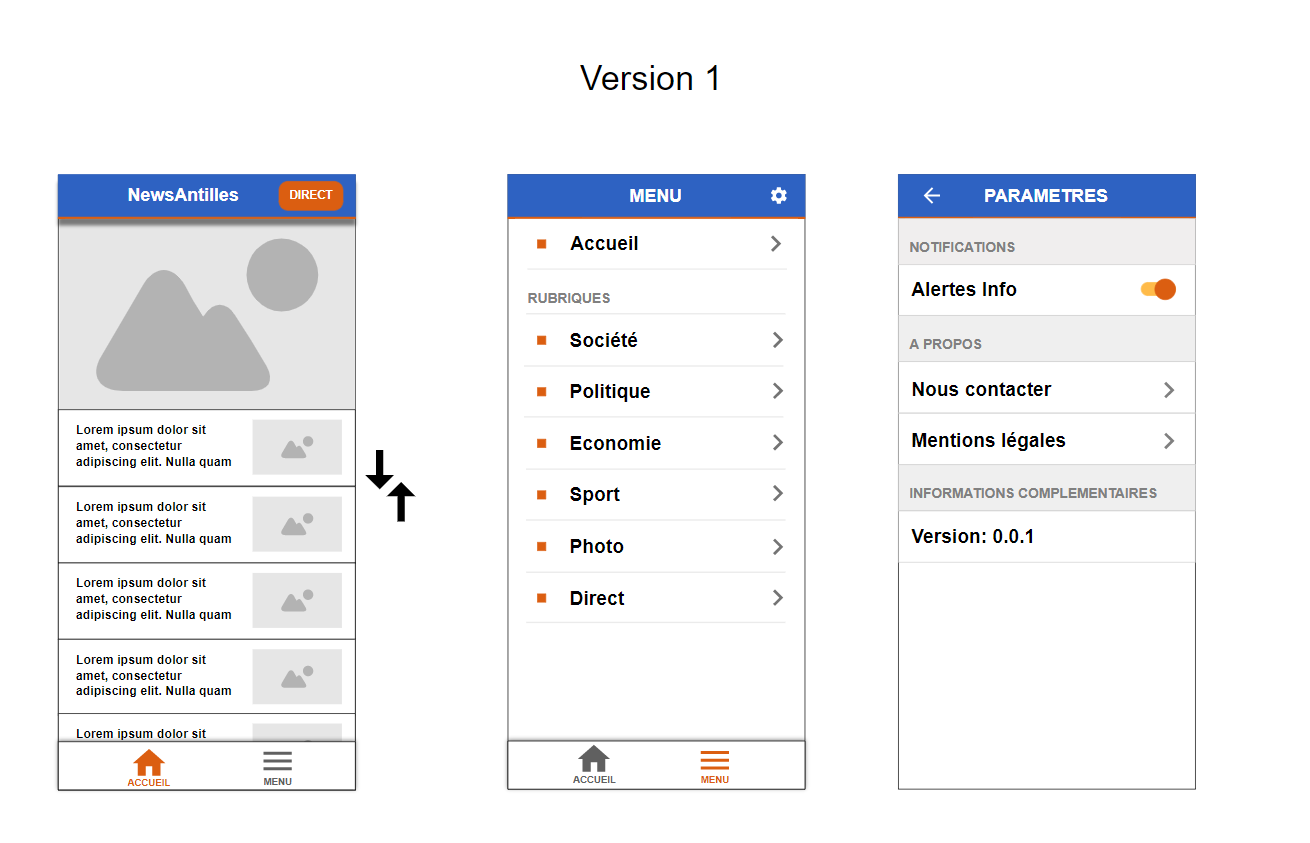


Son design se base autour de cette palette de couleur trois couleurs principales, et deux secondaires, que nous avons repris pour l’interface de l’application.



## Version 1

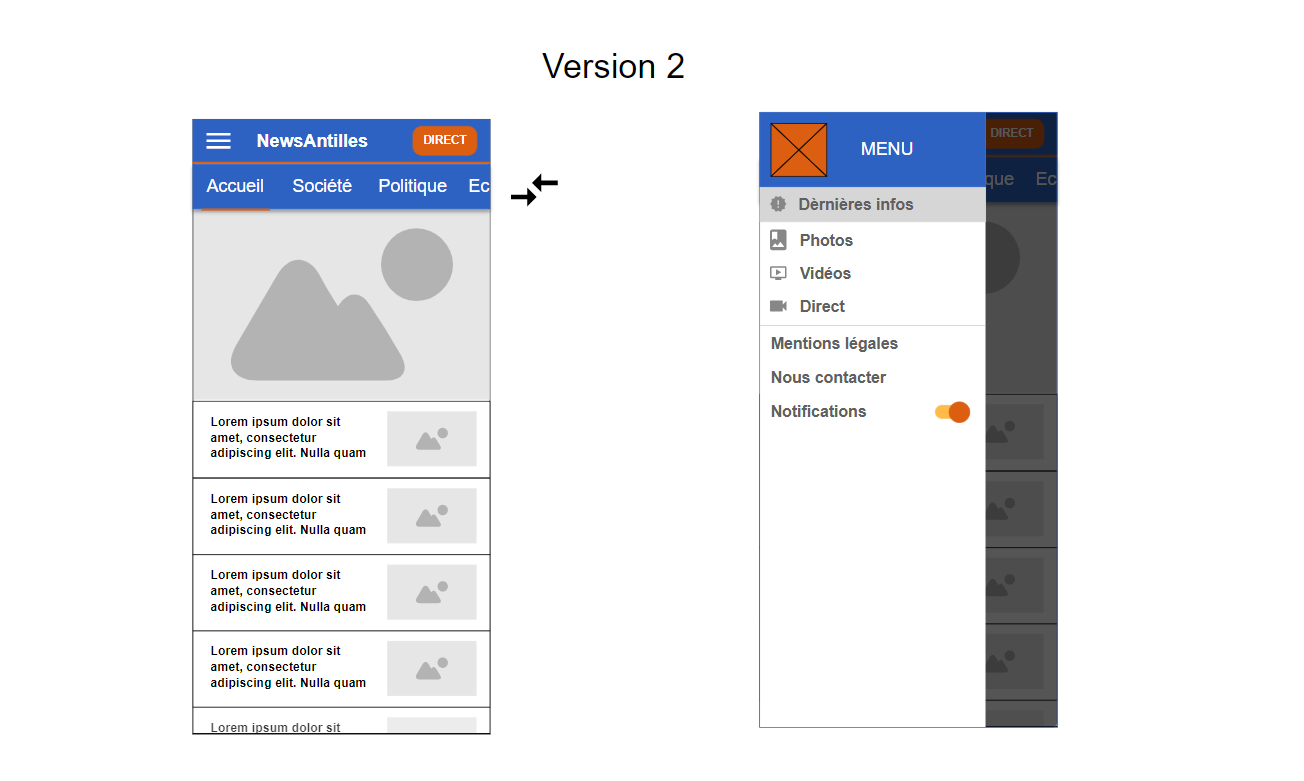
La première version de la maquette possède un écran d’accueil ou s’affichent les derniers articles et un menu occupant tout l’affichage du téléphone accessible sur la page d’accueil en cliquant sur une icône. Celui-ci permet de naviguer entre les différentes sections et donne l’accès à un second menu, qui lui donne accès à la gestion des notifications, la possibilité d’envoyer un mail à l’association directement depuis l’application ou de laisser un commentaire ou une note sur le Play Store, ainsi que les mentions légales et le numéro de version.



Cette première version a un design simple, et aussi proche que possible de la description de M.AZEDE

## Version 2

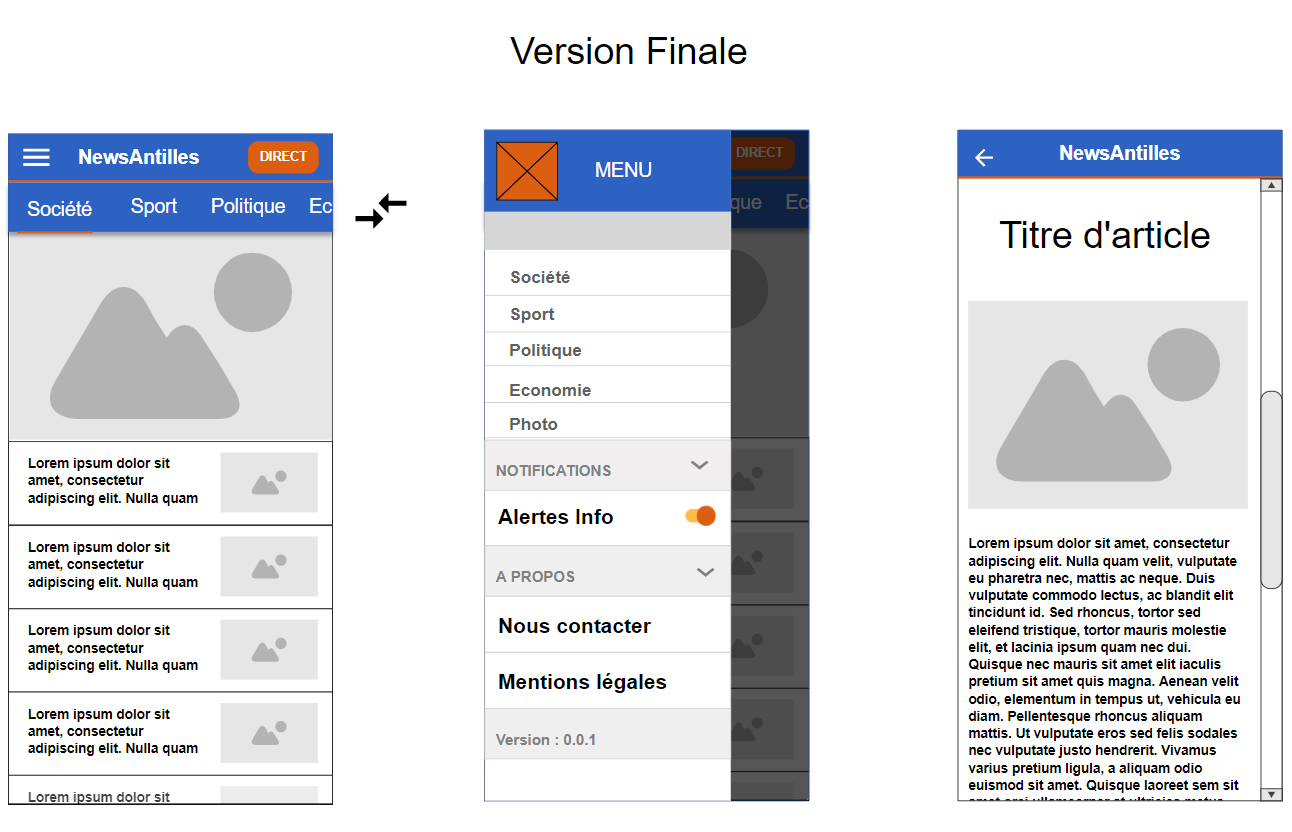
La seconde version de la maquette supprime la barre de menu pour la remplacer par un bouton dans la barre de titre. Aussi, le menu est remplacé par une barre de navigation horizontale, permettant de changer de section en un clic au lieu de deux. De ce fait, la fonction de navigation entre les sections a été supprimée du menu et seules les sections les moins fréquentées y ont été laissées. L’espace ainsi libéré a alors été attribué aux options du menu paramètres, afin de rassembler les fonctionnalités et limiter les menus dans des menus. A notre avis cela faciliterai la navigation et éviterai à l’utilisateur de devoir chercher les options qu’il souhaite, puisqu’elles seront toutes immédiatement accessibles.



Après un second entretien, M.AZEDE a apprécié notre second design mais a insisté pour que nous lui rajoutions le menu de navigation de la première maquette. Nous avons émis une remarque concernant la redondance du menu de navigation, qui serai affiché à la fois sur l’écran principale et dans le volet des paramètres. Mais à sa demande nous l’avons laissé.

Après un dernier passage sur moqups, nous avons donc réalisé une version finale de la maquette, correspondant à ses attentes

## Version finale



# Phases de réalisation

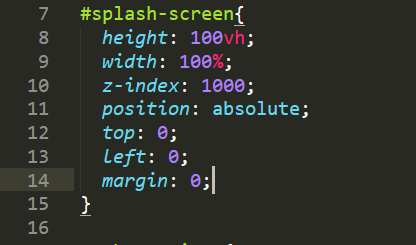
A ce point de projet, mon collègue et moi nous sommes divisés le travail ; il s’occupera de la partie HTML et CSS et moi de la partie JavaScript et PHP.

Afin de planifier l’organisation de mon code, j’ai décomposé le fonctionnement de l’application en taches simples, que j’ai décomposé en suite d’instructions, elles même rassemblées en fonctions :

1. Au lancement de l’application le splash-screen apparait pour une seconde puis disparait.
2. Après le splash-screen, l’utilisateur se retrouve sur la page d’accueil qui lui présentera les derniers articles
3. Si l’utilisateur clique sur un des liens de la barre de navigation, le contenu du corps de l’application est effacé et est remplacé par le fil des articles de la section correspondante.
4. Si l’utilisateur clique sur le bouton du menu, la fenêtre du menu apparait avec une animation.
5. Si l’utilisateur clique à côté du menu celui-ci disparait avec la même animation, à l’envers cette fois ci.

## Tache 1

Le splash screen est composé d’une <div> faisant toute la taille de l’écran contenant le logo de l’application, la version du build, et une animation de chargement.



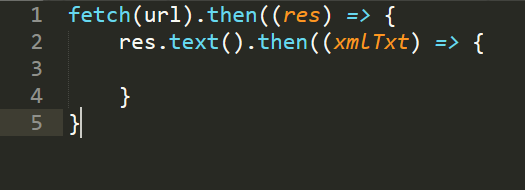
Au chargement de l’application, une fonction se lance qui identifie cette <div> et après une seconde, la fait disparaitre en lui appliquant une valeur à son element de style display égale à « none » et une animation grace à la fonction fadeOut() incluse dans Jquery .

D:\PC\Dropbox\Captures d'écran\Capture d'écran 2019-10-07 00.44.42.png

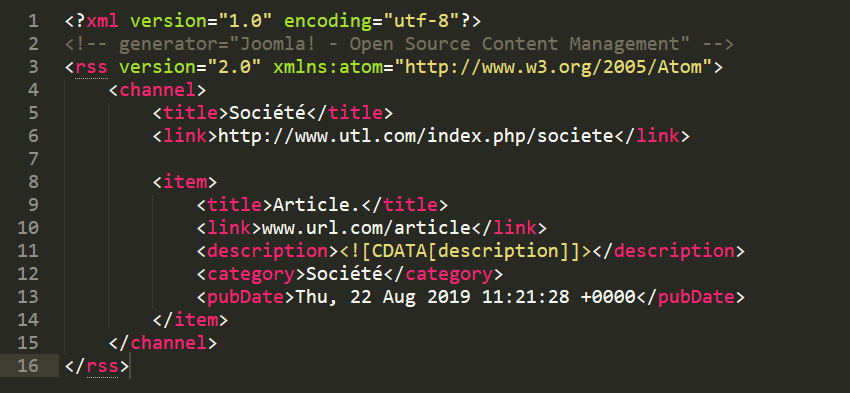
## Tache 2

J’ai utilisé la fonction fetch() pour faire une requête http vers le lien du flux RSS en rentrant l’adresse du flux RSS en paramètres, puis ai transformé la réponse au format XML en une suite de caractères grâce à la fonction text() avant d’affecter le résultat à une variable de type string.

La fonction fetch()

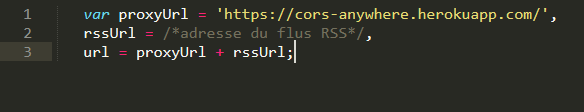


Un exemple de code XML



Cependant, le serveur de NewsAntilles rejette les requêtes de sources extérieures. A ce moment du développement, il était prévu que nous ayons accès au serveur, ce qui était censé me permettre d’autoriser les requêtes de l’application, mais l’information ne nous avait pas encore été communiquée. J’ai donc utilisé l’API CORS-anywhere pour contourner ce problème.

La requête CORS-API



Une fois le résultat de la requête recue,d, je parcours à l’aide d’une boucle toutes les balises item du fichier XML, et crée une nouvelle tuile pour chacune d’entre elles. Puis je récupère les éléments qu’elle contient ; la balise titre pour le titre de l’article, la balise <link> pour le lien, et la balise description pour les images, et grâce à la fonction document. createElement () je crée une élément enfant à la tuile en leur attribuant une classe avec element.className() qui permettra au CSS de définir le style graphique de l’élément.



Pour finir, la fonction append() permet d’intégrer l’élément envoyé en paramètre dans le document, ici, l’élément tuile, contenant un élément titre et un élément image sera intégré dans la <div> ayant la classe output

La boucle est placée dans une fonction qui est lancée au chargement de la page, en même temps que le script du splash-screen de façon à commencer le chargement des éléments avant que l’utilisateur ne puisse voir la page.

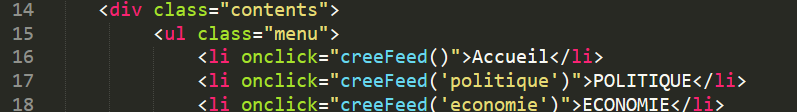
On appellera cette fonction creeFeed()

## Tache 3

Chaque catégorie d’articles possède son flux RSS dont l’adresse suit le schéma suivant :

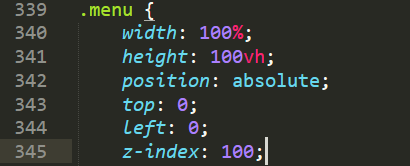


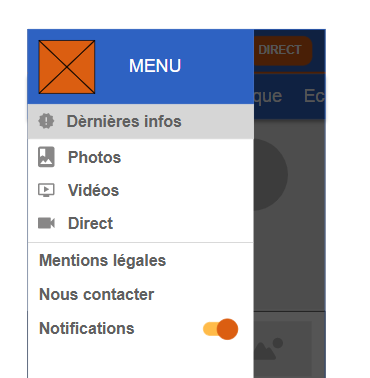
Les liens des menus de navigation, renvoient onClick, un appel à la fonction de creeFeed() avec pour argument le nom de la section.



Celui-ci sera stocké dans la variable nomSection, qui possède par défaut la valeur «», avant d’être ajouté à la variable RSSUrl.

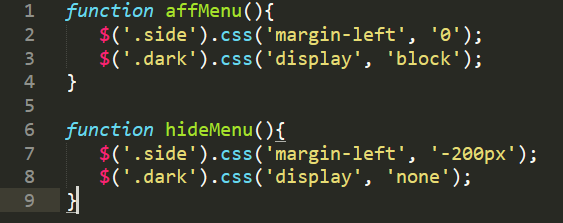
## Tache 4 et 5



Tout comme pour le splash-screen, le menu est une <div> en « position : absolute » avec un « z-index : 100 ».

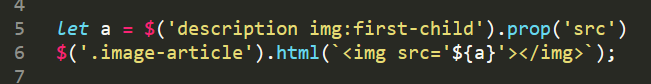
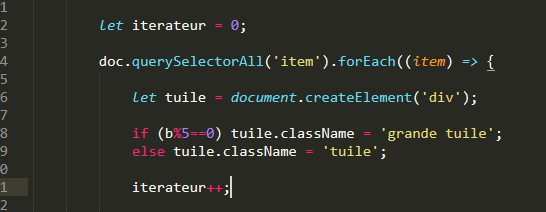
Cette div en contient deux autres, l’une contenant les éléments visuels du menu, et l’autre, vide ayant une couleur sombre et une opacité de 50% permettant de fermer le menu onClick. La partie visuelle possède une marge gauche de -200px et la partie sombre un « display : none ».

Quand l’utilisateur clique sur le bouton hamburger  dans la barre des titres, il lance une fonction afficheMenu() qui fait passer la valeur de la marge gauche de l’élément de -200px à 0 et le display de la div sombre à « block ». une fois le menu ouvert, en cliquant sur la <div> sombre, la fonction cacheMenu() se lance, qui rend à chaque élément ses valeurs initiales.



## Améliorations

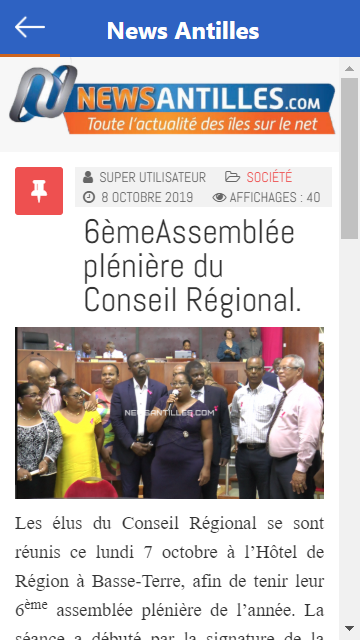
Une fois les fonctionnalités de base implémentées, et l’application fonctionnelle, nous l’avons présentée à M.AZEDE en tant que version béta afin qu’il la valide. Une fois fait, nous avons continué à la peaufiner.

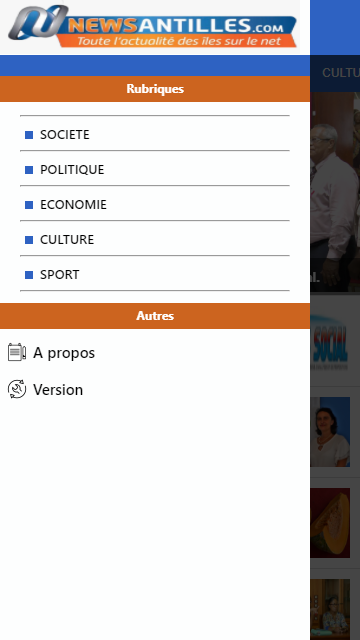
* Nous avons appliqué la propriété CSS animation-duration aux éléments du menu ; celle-ci a permis d’appliquer les modifications apportées au CSS par le code de façon graduelle sur une durée définie, créant un effet d’animation.
* A chaque fois qu’une requête est effectuée, l’affichage du fil est réinitialisé et les articles affiches lors de la requête précédente sont effacés grace à la fonction Jquery html() à laquelle j’envoie comme argument un string vide.
* Pour afficher l’image de chaque article, je collais tout le contenu de la balise description (qui contient à la fois le lien de l’image et du texte) et appliquais dans le CSS une taille de police de 0. Cependant, certains articles ayant plus d’une image, le layout de l’application buggais. Comme solution j’ai récupéré la propriété src du premier élément de type <img> de chaque article et l’ai appliqué à l’élément de classe « .image-article », qui deviendra une balise <img> au lieu d’une balise <div>
* Après avoir vu une version en progrès de l’application M.AZEDE A demandé à ce que la photo du premier article soit plus grande que les autres, et qu’une photo sur cinq par section aussi. J’ai donc modifié la boucle qui affiche les articles pour qu’elle incrémente une variable à chaque itération, et qu’elle applique une classe CSS différente pour tous les multiples de cinq.

## Conclusion

Voici le rendu final de l’application telle que nous l’avons présentée à M.Azede à la fin du stage.







Nous avons cependant dû modifier le design initial en raison d’imprévus lors du développement :

1. La chaine YouTube n’étant pas une priorité de NewsAntilles et n’ayant pas été mise à jour depuis plusieurs mois, nous avons décidé de retirer le bouton du direct
2. Contrairement à ce qui était prévu nous n’avons pas reçu l’accès au serveur. Notre plan initial était de récupérer les informations des articles directement sur le serveur et ensuite de les afficher dans l’application. Cette option n’étant plus possible, nous avons décidé d’afficher la page web de l’article directement dans l’application. Nous avons donc créé un <iframe> prenant la totalité de l’affichage, et ayant un « display : none ». lorsqu’un article est sélectionné par l’utilisateur, le display devient « block » et l’url de l’article est attribuée à sa propriété src.
3. Suite à un bug sur le serveur, le flux RSS de la page d’accueil ne fonctionne pas, nous avons donc dû retirer la page des derniers articles de l’application.
4. La fonctionnalité de notifications n’a pas pu être implantée car elle necessitait l’emploi de technologies que nous ne maitrisons pas

Dans les mois qui suivent, nous resterons en contact avec M.Azede afin de palier à ces problèmes.

# Points Clé

* BDD !!!!!!!!
  + mettre du php dans le powerpoint
  + parler de la sécuité
* Imprimer un CV a jour (ajouter le projet de stage)
* se présenter au jury
* présentation powerpoint
* site dédié, responsive web design
* lire la doc, les coms, api , vs
* référencement naturel, seo, rapidité du charfement, compatibilité avec les mobile, mots clés bien choisis,
* vulnérabilités: injection sql, Cross Script Scripting
* accessibilité: vitesse de chargement, design, responsive
* mesure de sécurité pr les users: login
* verifier la conformité à la maquette: recette (site conforme, tests)
* verifier la conformité de la bdd à la maquette: clé primaires