(b) Modélisation des données avec MySQL Workbench

Le diagramme de classe exprime de manière générale la structure statique d’un système en termes de classes et de

relations entre ces classes. Une classe permet de décrire un ensemble d’objet (attribut et comportement).Il s’agit donc

de réaliser un diagramme de classe comportant les concepts suivants :

- Entité : modélisation d’un objet d’intérêt pour l’utilisateur.

- Relation : modélisation d’une association entre deux ou plusieurs entités

- Multiplicité : comparable aux cardinalités du système Merise, sert à compter le nombre minimum et maximum

d'instances de chaque classe dans la relation liant 2 ou plusieurs classes.

Graphique sans nomFigure 9 – digramme de classe via Workbench

Le digramme est composé de 4 entités et les relations associées :

Domains : relation OneToMany (1, n) avec l’entité Subrscribers

 Users : relation OneToMany (1, n) avec l’entité Campaigns

 Campaigns : relation ManyToOne (n,1) avec l’entité Users

 Subscribers : relation ManyToOne (n,1) avec l’entité Domains JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 16

III. Spécifications techniques

Graphique sans nomLe choix des technologies

Pour mener à bien notre projet et pour un souci de maintenabilité l’application sera réalisée avec les technologies

suivante :

- Framework Symfony 5.1.0

- Le langage UML

- Bundle EASYADMIN V3.3.

- Bundle KREAIT FIREBASE ADMIN SDK

- API FIREBASE de google

- PHP 7.3

- Base de données : MARIADB / MYSQL / PHPMYADMIN

- Hébergement OVH

- Versionning : GitHub

- IDE VSCODE

- Système d’exploitation Windows

Graphique sans nomPrésentation de l’architecture MVC (MODEL-VIEW-CONTROLLER)

L'architecture Modèle/Vue/Contrôleur (MVC) est une façon d'organiser une interface graphique d'un programme. Elle

consiste à distinguer trois entités distinctes qui sont, le modèle, la vue et le contrôleur ayant chacun un rôle précis dans

l'interface. L'organisation globale d'une interface graphique est souvent délicate. L’architecture MVC est l’une des

architectures logicielles les plus utilisées pour les applications Web. Elle offre aussi un cadre pour structurer une

application. Elle se compose de 3 modules :

 Modèle : noyau de l’application qui gère les données, permet de récupérer les informations dans la base

de données, de les organiser pour qu’elles puissent ensuite être traitées par le contrôleur.

 Vue : composant graphique de l’interface qui permet de présenter les données du modèle à l’utilisateur.

 Contrôleur : composant responsable des prises de décision, gère la logique du code qui prend des

décisions, il est l’intermédiaire entre le modèle et la vue

Graphique sans nomFigure 10 – interactions entre le modèle, la vue et le contrôleur.

(a) Présentation Framework Symfony

Symfony est un puissant Framework PHP respectant le design pattern MVC et utilisé pour développer des applications

web ou des sites web complexes, voire haut de gamme. Relativement simple d’accès, Symfony est un ensemble de

composants (ou « bibliothèques ») qui facilitent le développement web en réduisant de façon considérable le temps

et l’effort requis pour créer des composants génériques. Le langage de programmation PHP est celui qui est

majoritairement utilisé dans le cadre du développement web et des scripts de commande en ligne. JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 17

L’architecture MVC de Symfony :

Graphique sans nomFigure 11 – Architecture MVC du Framework Symfony

Lorsqu'un client demande une URL, celle-ci est d'abord traitée par le contrôleur frontale de Symfony avant d'être

envoyée au noyaux.

Quand le noyaux reçois la demande d'une URL, il appelle le service de ROUTING, celui-ci va alors lui indiquer le

contrôleur à appeler pour l'URL que le client demande.

Le noyaux appel alors le contrôleur qui va si il en a besoin, appeler le modèle et générer la vue.

Quand le contrôleur a fini que générer la vue, il retourne le résultat au noyaux, celui-ci va alors transmettre la réponse

au client.

ORM : signifie Object Relationnel Mapping. Il permet de créer l'illusion d'une base de données orientée objet à partir

d'une base de données relationnelle, Il fournit la persistance transparente des objets PHP, et c’est l’interface qui permet

de faire le mapping entre nos objets et les éléments de la base de données. Il existe deux types d'ORM :

Data Mapped (Doctrine) : Permet une abstraction plus puissante avec la base de données, on envoie l’objet à

un manager pour le sauvegarder en base de données.

Active Record (Propel) : Un objet est une ligne de la base de données, chaque champs est une colonne, la

méthode save() est présente sur chaque objet.

L'arborescence d'un projet Symfony est la suivante :

Graphique sans nomFigure 12 – Arborescence Symfony

Graphique sans nomPartie 1 : Le backend admin, gestionnaire de notifications

EASYADMIN est un bundle de Symfony ou un module d’administration, c’est un outil qui permet de créer

des interfaces d’administration plus ou moins avancées. On peut réaliser des opérations SCRUD

(Search/Create/Read/Update/Delete) très facilement sur nos entités Doctrine (ORM seulement).

EASYADMIN est rapide et performant dans ses tâches, permettant une navigation fluide. L’interface de JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 18

base reste très basique (basée sur Bootstrap). Et c’est tout ce qu’on demande à un module

d’administration : de la fonctionnalité, pas de l'esthétique.

(a) Etape 1 installation et configuration de Symfony

On va créer une application web complète on exécutera cette commande afin de charger toutes les dépendances

- Symfony new MobileboostProjet –full

- Depuis la racine on modifie le .env (on modifie DATABASEURL login et MDP de la BDD)

- Symfony console make:entity (on créer nos entités)

- Symfony console make : migration (ORM analyse l’entité et la BDD et génère le SQL)

- Symfony console doctrine :migration :migrate (on charge la BDD avec nos entités)

(b) Etape 2 : installation de EASYADMIN

- Symfony composer req admin

- Symfony console make :admin :Dashboard (créations du Dashboard home page)

- Symfony console make :admin :crud (on va générer un CRUD sur toutes nos entités)

- Arrivé a cette étape nous devons configurer le dashboardController.php pour pouvoir y accéder comme ceci :

public function configureMenuItems(): iterable

{

yield MenuItem::linktoDashboard('Dashboard', 'fa fa-home');

yield MenuItem::linkToCrud('Utilisateurs', 'fa fa-user', User::class);

yield MenuItem::linkToCrud('Abonnés', 'fa fa-users', Subscribers::class);

yield MenuItem::linkToCrud('Campagnes', 'fa fa-edit', Campaigns::class);

yield MenuItem::linkToCrud('Domaines', 'fa fa-link', Domains::class);

}

}

Ensuite éditer également le fichier de chaque crud controller pour configurer les champs.

On prend l’exemple ici du USERCONTROLLER :

Graphique sans nomFigure 13 – configuration des champsJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 19

(c) Etape 2 : installation du bundle KREAIT

On installe ensuite le bundle KREAIT qui va servir pour l’envoi des notifications ce bundle est spécialement conçu

pour FIREBASE

Symfony composer require KREAIT/ FIREBASE -bundle

Ensuite on va toute suite configurer le bundle pour ce faire on va dans le dossier config et on modifie le firebase.yaml.

Voici un extrait de code

Graphique sans nomDans le fichier. ENV on ajoute cette ligne c’est le chemin vers les crédenciales :

###> app ###

FIREBASE\_CREDENTIALS=mobilboostproject-firebase-adminsdk-otqie-e99724c86b.json

###< app ###

(d) Extraits de code les plus significatifs

Pour pouvoir envoyer des notifications il va falloir créer une fonction qui va permettre l’envoie des

notifications via le bundle KREAIT. Cette partie du code et la plus importante car c’est elle qui va exécuter

Graphique sans noml’envoie d’une notification. Voici le code qui se trouve dans le Controller Campaigns :

Figure 14 – fonction principale d’envoi de notificationJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 20

Figure 15 – requête findOneShoot et UpdateStatus

Graphique sans nomExplication de la fonction SENDNOTIF :

Une requête est créée depuis le repository pour récupérer les données en BDD est

 On va venir les stocker dans des variables afin de les réutiliser plus tard dans la fonction cloud

messaging ().

 On passe par la fonction Cloud Message qui va se charger d’envoyer la notification à tous les abonnés

inscrit.

 On utilise la méthode d’envoi via les Topics. J’ai commencé par utiliser la fonction CURL de PHP dans

mon Controller et après quelques recherches je suis tombé sur le bundle KREAIT qui s’occupe de

l’envoi des notifications et qui s’intégrer parfaitement. Ainsi on génère moins de code dans le

contrôleur. Ainsi on va pouvoir faire de l’injection de dépendance, on va venir chargée la classe

messaging dans le constructeur et utiliser les méthodes du bundle.

 Et on fait une update de la table lors de l’envoie de la notification ainsi le champs STATUS passe a 1

ce qui signifie que le message a été envoyé.

(e) Extrait de code sur la sécurité

Seul l’administrateur aura accès au site en tant que ROLE ADMIN, un seul rôle a été créer car le client et le

seul à utiliser l’application mais d’autres Rôles pourraient s’ajouter au niveau du Security.yaml afin de

d’établir la hiérarchie des rôles si le client souhait ajouter d’autre utilisateurs et de restreindre certains

menus.

On accède directement à la page l’authentification via l’url https://phonipush.com https://phonipush.com. Un login et mot de passe

est requis afin d’accéder à l’administration. Comme cette application est privé et donc il n’y pas de lien

d’inscription, j’ai dû insérer un l’utilisateur en base afin de se connecter. L’administrateur pourra ensuite

modifier son mot de passe directement depuis l’interface utilisateur. Le mot de passe est crypté on utilise ici

la classe UserPasswordEnconderInterface. On appelle la méthode setPassword dans l’entité User qui vient

prendre en paramètre $passwordencoder et on appelle la méthode encodepassword qui hérite de l’interface JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 21

et qui prend en paramètre l’entité user et le mot de passe à haché bien sûr on effacera au niveau du Security

Controller la persistance des données, afin de pas générer d’erreur.

Illustration du Code Controller Security : insertion de l’administrateur

Graphique sans nomFigure 15 – insertion de l’administrateur en base de données

Comment Restreindre l’accès aux autres rôles : Dans cet exemple on va prendre le Campagne Controller,

dans la fonction configure Actions on va tous simplement ajouter une ligne de code au niveau du return

$action comme sur la figure ci-dessous ainsi on supprimera les bouton DELETE et EDIT pour tous les

utilisateurs qui n’ont pas le rôle de super admin.

Graphique sans nom(f) Explication détaillée sur les deux entités de l’application :

Abonnés : En ce qui concerne les abonnés on stock les informations de son USERAGENT, IP, Pays,

opérateur mobile, TOKEN, url, toutes ces données seront récupérées depuis la page html en GET via une

fonction jQuery qui va appeler un script PHP et qui se chargera d’insérer en base toutes les données de

l’utilisateur. (Ce script PHP sera sur la racine de tous les nom domaines)

Un champ date de création qui sera généré automatiquement depuis le constructeur dans

l’entité abonnée

 Une date de suppression (dans le cas où l’utilisateur se désabonne)

 L’IP de l’abonné

 La Provenance (exemple : France)

 L’opérateur mobile (SFR, ORANGE, Bouygues, FREE)

 Appareil utilisé (iOS, androïde, Mozilla, Chrome, opéra…) JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 22

La clef TOKEN qui sera récupère depuis la page html et envoyer en base depuis une fonction en

JQUERY

 Et le nom domaine associés à l’abonnés site sur lequel l’utilisateur a navigué

Campagnes : Cette espace est spécialement dédiée à la création de campagne et à l’envoie de la

notification, une fois le formulaire soumis on récupère les données depuis une fonction dans le

Controller campagne (description détaillée de cette fonction dans la section extrait de code au-dessus)

Un Utilisateur

 Un Titre (obligatoire)

 Un Message texte area (obligatoire)

 Un menue déroulant avec la fréquence d’envoie avec en premier lien : (obligatoire)

 Quotidien (une fonction qui doit gérer l’envoie automatique mais faute de temps elle

n’a pas été créée sur cette première version de l’application)

 Ou ONE-SHOOT (c’est un envoi immédiat de la notification)

 Une date d’envoi (programmer une la date et l’heure d’envoi de la notification : obligatoire)

 Une bannière (petite image qui vas se trouver sous le texte de la notification)

 Une icône (une icône qui va se trouver sur la gauche juste à côté du texte de la notification)

(obligatoire)

 Nom domaine (on choisira le nom domaine sur lequel on enverra la notification : obligatoire)

 Url (on associe à chaque campagne une url, lorsque qu’on affichera la notification à l’utilisateurs

celui-ci pourra cliquer sur la notif et sera rediriger vers la page voulue : obligatoire)

Graphique sans nomFigure 16 – formulaire principale de création de notification

JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 23

Graphique sans nomPartie 2 : frontend et analyse du code source de l’API FCM

(a) Graphique sans nomExplication détaillée d’un notification web push

Dans cette partie je vous explique la notification dans sa globalité et certains avantages à l’utiliser

Tout d’abord vous accéder au site web, un pop-up s’affiche et le site vous demande si vous voulez vous

abonner aux notifications, on est libre du choix mais a chaque fois que vous accéder au site le pop-up

s’affichera de nouveau. On peut depuis les paramètres du navigateur accéder à la liste des sites abonnés et

se désabonner au service.

La web push notification, aussi appelée notification web, est une technologie qui permet d’envoyer des

notifications aux internautes via leur navigateur, sans qu’ils consultent votre site. Elle fonctionne sur

Chrome, Firefox, Safari, Edge et Opéra, mais aussi sur Android. C’est une fenêtre qui permet d’envoyer un

message avec texte, image et logo.

La notification web est envoyée au navigateur par le système d’exploitation de son ordinateur ou de son

mobile. Son but est de rediriger les internautes sur votre site lorsqu’ils sont en ligne en leur adressant des

messages courts et ciblés au bon moment. Cela permet d’augmenter votre taux de conversion et de

fidéliser votre clientèle en leur envoyant des offres personnalisées.

Pour fonctionner, la notification web a besoin de l’autorisation de l’internaute. Puis, lorsqu’il reçoit une

web push et quand il clique dessus, il sera redirigé vers le site web.

(b) Les avantages à utiliser les notifications push

Cette technologie innovante est un excellent canal de communications qui présente de nombreux

avantages :

Une technologie 100 % RGPD READY

Pour qu’un internaute reçoive des notifications web, il faut d’abord obtenir son consentement. Cela veut

dire que c’est une technologie d’annonceur 100 % RGPD READY.

Une innovation anti-fraude qui respecte l’image de marqueJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 24

Comme les notifications web apparaissent directement sur l’écran de l’internaute, les risques de fraude ou

de dépréciation de l’image de marque sont éliminés.

Une technologie qui fonctionne malgré les Adblockers

Les notifications web permettent de toucher 100 % de votre audience, avec ou sans bloqueurs de

publicités.

Une solution sans Cookies

Les cookies tiers sont amenés à disparaître. La web push notification permet de toucher les internautes qui

ont visité votre site ou des pages sans utiliser de cookies.

Une totale maîtrise de la pression marketing et du budget

L’équipe chargée de l’acquisition client est seule pour décider à qui et quand notifier d’un message

personnalisé à ses abonnés. Cela permet un contrôle total de la pression marketing et donc, de son budget.

Des messages ciblés et personnalisés

La web push notification permet de collecter des informations sur les internautes abonnés, comme le fait

de savoir à partir de quelle page Internet ils ont cliqué ou s’ils ont déjà acheté sur le site.

Cela permet de cibler de façon précise les personnes d’une base de données et ainsi de mettre en place

des scénarios marketing automatisés et précis, d’améliorer les taux de conversion, ou encore de fidéliser

ses abonnés.

Des communications planifiées

Voici un des avantages qui rend la web push notification très singulière. Étant donné que c’est le navigateur

qui envoie la notification et qu’elle est affichée par le système d’exploitation du terminal, l’abonné n’a pas

besoin d’être sur votre page Internet pour recevoir une notification web. Ainsi, elle peut être planifiée et

donc envoyée à tout moment.

Une solution efficace et peu onéreuse

Les notifications sont un élément à avoir en tête pour plusieurs raisons. Et contrairement au mail, il n’y a pas

de problématique de délivrabilité : tous les inscrits aux notifications sont adressables.

Autre avantage de poids : la possibilité de communiquer en temps réel. Vous pouvez à n’importe quel

moment envoyer des push personnalisés qui s’adaptent au comportement du visiteur (offres spéciales,

abandon de panier, offre de bienvenue…). Le re ciblage de vos abonnés devient totalement gratuit.JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 25

Vous avez aussi la possibilité de cibler et recibler vos visiteurs qui n’utilisent pas le mobile via les notifications

desktop. Une part non négligeable de votre audience. Enfin, vous avez la main sur la pression marketing, et

sur l’environnement de diffusion, contrairement au display. Rappelons qu’il est possible de se désabonner

des notifications à tout moment, l’intrusion est donc très limitée pour l’utilisateur.

(c) Graphique sans nomSchéma technique de l’envoie d’une notification

Figure 17 : représentation technique d’une notification

(d) Implémentation du code de l’Api FIREBASE

Tout d’abord avant tout commencement, on doit créer un projet sur FIREBASE.google.com ainsi on pourra

récupérer la configuration du projet. On pourra ainsi récupérer la clef FCM server pour envoyer des requêtes

https cotés server à l’aide du CURL sur un script en PHP.

Sur la partie Html on va implémenter le code javascript de l’Api FIREBASE, pour ainsi récupérer les données

de l’utilisateur. On viendra ajouter le Sdk FIREBASE et JQUERY.

Sur cette figure on implémente les SDK jQuery et le SDK FIREBASE et on récupère depuis notre espace notre

configuration. Le SDK FCM est pris en charge uniquement dans les pages diffusées via HTTPS. Cela est

important car si le site n’est pas en https le SDK ne fonctionnera pas.JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 26

Figure 18 – SDK configuration FIREBASE

Graphique sans nom(e) Souscrire aux notifications

Avant de pouvoir établir la communication entre le serveur de notifications et le navigateur, il faut demander

à l'utilisateur s'il accepte d'être notifié. Cela se fait en appelant la méthode requestPermission de

l'objet Notification

Graphique sans nomFigure 19 – 1

er partie de la fonction qui permet à être notifié

La promise renvoyée par la méthode contient le résultat de la demande. Ce résultat peut avoir trois valeurs

: granted (permission accordée), denied (permission refusée) ou default (décision inconnue, à considérer

comme un refus).

Si l'utilisateur nous donne l'autorisation, on récupère le token ainsi nous allons pouvoir créer la souscription

aux notifications. Cette souscription permet d'accéder aux informations nécessaires à l'envoi de

notifications à l'utilisateur. Ici on récupère le pays (countryiso), carrier (réseau), devisetype (iOS , chrome..)

et enfin le token. On utilise la méthode saveToken() pour l’insertion des données en base. Et pour la

fcmKey() celle-ci va servir pour souscrire l’utilisateur a un Topic. On fait le choix d’utiliser les topics car Les

messages thématiques sont optimisés pour le débit plutôt que pour la latence. La réception du message

sera plus rapide. JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 27

Figure 20 – partie 2 : sauvegardes des données utilisateurs

Graphique sans nomPour recevoir une notification quand l’application est active nous devons utiliser la

méthode onMessage() deFirebase, sans cette méthode, si une notification est envoyée et que l’utilisateur

navigue sur l’application, il ne se passera rien.

Graphique sans nomFigure 21 – fonction qui permet d’afficher la notification

La méthode onTokenRefresh()sera appelée chaque fois qu'un nouveau jeton est généré. Chaque fois qu’un

utilisateur Graphique sans noml'utilisateur supprime les données de l'application et redemande à être abonnée.

Figure 22 – méthode qui génère d’autres token

La fonction fcmkey() qui va récupérer le TOKEN afin de souscrire un utilisateur au topic depuis

curl\_firebase.phpJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 28

Figure 23 Graphique sans nom–fonction jQuery qui récupère le Token

Le script PHP CURL\_FIREBASE.PHP qui va abonner un utilisateur via CURL ici on va appeler le script cotés

serveur pour éviter d’afficher la clef serveur et de permettre a une personne de mal attentionner

d’envoyer des requêtes sans autorisation.

Graphique sans nomFigure 24 – script abonné un visiteurJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 29

Le script PHP désabonner. PHP qui va permettre de désabonner un utilisateur aux notifications via CURL

La différence entre le script abonner et le désabonner c’est la ligne CURLOPT\_CUSTOMREQUEST, « DELETE »

cette dernière va désabonner un utilisateur aux notifications.

Graphique sans nomFigure 25 – script php pour désabonner un visiteur

La fonction saveToken() qui va récupérer les données de l’utilisateurs et les stocké en base depuis le fichier

PHP Graphique sans nomsavetoken.php

Figure 26 - fonction qui récupère les données utilisateursJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 30

Le script savetoken.php : ici on va insérer les donner utilisateur qu’on récupère depuis le fichier

index afin de les insérer en base de données

Graphique sans nomFigure 29 – fonction qui va permettre d’afficher les notifications en arrière-planJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 31

(f) Quand l’application est inactive (en background ou fermée)

L’application n’étant pas visible, la fonction onMessage() n’est pas appelée à la réception d’une

notification, pour permettre la réception d’une notification et avoir un traitement personnalisé, il faut

utiliser le SERVICE-WORKER, on créer le fichier firebase-messaging-sw.js a la racine des site web.

On va créer une fonction notification clique afin de pouvoir cliquer sur la notif et de permettre de rediriger

l’utilisateurs vers une url voulue.

Graphique sans nomFigure 28 – fonction qui va permettre d’afficher les notifications en arrière-planJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 32

IV. Recherche en anglais

J’ai effectué plusieurs recherches en anglais car la documentation FIRBASE et en anglais et la

documentation du bundle KREAIT également, mais l’un des problèmes que j’ai eu c’était de comprendre

comment fonctionner la méthode SubcribeToTopic () de FIREBASE. Dans la documentation FIREBASE on

nous dit qu’avant de vouloir utiliser cette fonction il faut initialiser le Sdk ADMIN sur le serveur sur un

projet NODEJS. N’ayant pas trop la maitrise de NODEJS j’ai essayé de trouver une autre alternative. En

premier lieu j’ai récupéré cette fonction et j’ai insérer ma clef server en dur. Cela a fonctionner mais le

problème c’est que la clef server doit rester privé et ici je l’affiché coté client. Apres des journée de

recherche j’ai compris comment fonctionner le procédé, il fallait tous simplement faire un appel ajax sur un

script PHP coté server. Mes recherches mon mener à une source sur le site stackoverflow que voici :

How to subscribe to topics with web browser using Firebase Cloud Messaging

There is no direct API to subscribe clients to topics in the Firebase Cloud Messaging SDK for JavaScript.

Instead you can subscribe a token to a topic through the REST API. Calling this API requires that you specify

the FCM server key, which means that you should only do this on a trusted environment, such as your

development machine, a server you control, or Cloud Functions. This is necessary, since having the FCM

server key allows sending messages on your app's behalf to all of your app's users.

Comment s'abonner à des sujets avec un navigateur Web en utilisant Firebase Cloud Messaging ?

Il n'y a pas d'API directe pour abonner les clients aux sujets dans le SDK FIREBASE Cloud Messaging pour

JavaScript. Au lieu de cela, vous pouvez abonner un jeton à un sujet via l'API REST. L'appel de cette API

nécessite que vous spécifiiez la clé du serveur FCM, ce qui signifie que vous ne devez le faire que dans un

environnement de confiance, comme votre machine de développement, un serveur que vous contrôlez ou

Cloud Functions. C'est nécessaire, car la clé de serveur FCM permet d'envoyer des messages au nom de votre

application à tous les utilisateurs de votre application.

function subscribeTokenToTopic(token, topic) {

fetch('https://iid.googleapis.com/iid/v1/'+token+'/rel/topics/'+topic, {

method: 'POST',

headers: new Headers({

'Authorization': 'key='+fcm\_server\_key

})

}).then(response => {

if (response.status < 200 || response.status >= 400) {

throw 'Error subscribing to topic: '+response.status + ' - ' + response.text();

}

console.log('Subscribed to "'+topic+'"');

}).catch(error => {

console.error(error);

})

}

Any one looking for php solution find below since you are going to use Api key of server side

code by php curl

Si vous cherchez une solution en php, vous trouverez la solution ci-dessous puisque vous allez utiliser

la clé Api du code côté serveur avec curl.

$headers = array

('Authorization: key=' . API\_ACCESS\_KEY,

'Content-Type: application/json');

$ch = curl\_init();JAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 33

Glossaire

API : Une API permet de rendre disponibles les données ou les fonctionnalités d’une application

existante afin que d’autres applications les utilisent. Voici qui devrait rendre la notion d’interface de

programmation applicative plus claire

CURL: (Client URL Request Library) est un outil en ligne de commande auquel s’ajoute une bibliothèque

permettant d’effectuer des requêtes d’URL sur divers protocoles. La fonction principale de cURL est de

télécharger des éléments à partir d’URL. Le nombre de protocole supporté par cURL (DICT, file, FTP, FTPS,

Gopher, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTSP, SCP, SFTP, SMB, SMBS, SMTP, SMTPS,

Telnet et TFTP)

SERVICE-WORKER : Les SERVICE-WORKER jouent essentiellement le rôle de serveurs proxy placés

entre une application web, et le navigateur ou le réseau (lorsque disponible.) Ils sont destinés (entre

autres choses) à permettre la création d'expériences de navigation déconnectée efficaces, en

interceptant les requêtes réseau et en effectuant des actions appropriées selon que le réseau est

disponible et que des ressources mises à jour sont à disposition sur le serveur. Ils permettront aussi

d'accéder aux APIs de notifications du serveur (push) et de synchronisation en arrière-plan.

RGPD READY : est un outil de sensibilisation au Règlement Européen sur la Protection des Données

personnelles (RGPD), destiné à tous les salariés des entreprises.

FIREBASE : Google FIREBASE est une plateforme de développement d’applications pour le web ou

pour mobiles. Elle fournit des outils sous forme de services pour la création d’applications mobiles.

ORM : Un mapping objet-relationnel (en anglais object-relational mapping ou ORM) est un type de

programme informatique qui se place en interface entre un programme applicatif et une base de

données relationnelle pour simuler une base de données orientée objet. Ce programme définit des

correspondances entre les schémas de la base de données et les classes du programme applicatif. On

pourrait le désigner par-là, « comme une couche d'abstraction entre le monde objet et monde

relationnel ». Du fait de sa fonction, on retrouve ce type de programme dans un grand nombre

de Framework sous la forme de composant ORM qui a été soit développé, soit intégré depuis une

solution externe.

Doctrine : est un ORM (couche d'abstraction à la base de données) pour PHP. Il s'agit d'un logiciel

libre sous licence GNU LGPL.

 Un token d’authentification, aussi appelé « trusted device » (dispositif de confiance) en anglais, est

utilisé pour accéder à une ressource à accès restreint (généralement une application ou un réseau

d’entreprise). Il peut être considéré comme une clé électronique qui permet à un utilisateur de

s’authentifier, en prouvant son identité, grâce au stockage d’un certain nombre d’informations

personnelles. Un token d’authentification permet de fournir un niveau de sécurité supplémentaire pour

accéder aux donnéesJAOIAD LAACHIR / DEVELOPPEUR WEB ET WEB MOBILE 34

Conclusion

Le stage a été très enrichissant pour ma part, avec beaucoup de recherche et quelques blocages parfois,

mais mon tuteur de stage très pédagogue a su m’aiguiller tout au long du projet. J’ai beaucoup appris

notamment sur le PHP et l’utilisation de CURL ainsi que le fonctionnement de l’api FIREBASE et le Framework

Symfony. Ce stage très intéressant m'a permis d’augmenter mes connaissances et d’apprendre d’autres

technologies et des notions qu’on ne voit pas en formation. Ce stage s'est déroulé dans d'excellentes

conditions.