
Area - Cahier des charges

Alexandre Cochet | Johan Le Nagard | Jérémy Quemener | Thibaut Rassouli | Gervan Menguy | Mathis Calonnec

VUE D'ENSEMBLE

L'AREA est un projet de gestion de la vie numérique en temps réel. Les utilisateurs de l'Area peuvent créer un compte à l'AREA via l'application web ou l'application mobile. L'utilisateur, une fois inscrit sur le site, pourra souscrire à des services: Facebook, Google, Twitter, qui offriront différentes actions et différentes réactions. L'acronyme AREA signifie "Action REAction". Les utilisateurs peuvent alors lier des actions à des réactions. Les réactions se déclenchent alors quand l'événement de l'action est activé.

CARACTÉRISTIQUES

Le projet se découpe en trois grandes parties:

- Un serveur API RESTFUL en C#, avec les technologies ASP.NET Core et Firebase: Le serveur se divise en deux parties: Gestion des utilisateurs d'un côté, et gestion des événements actionnés/réactions enclenchées de l'autre. L'utilisation de l'API est précisée dans le fichier API.md.
 - **Développeurs: Alexandre Cochet**
- Une Application web en VueJS via laquelle les utilisateurs peuvent s'inscrire, se connecter, souscrire à des services et lier des actions à des réactions. L'application web doit seulement s'occuper de l'affichage en appelant l'API fournie par le serveur, et ne doit pas contenir de logique métier.
 - **Développeurs: Jérémy Quemener | Thibaut Rassouli**
- Une application mobile en Kotlin réalisant les mêmes actions que l'application web, et possédant les mêmes contraintes.
 - **Développeurs: Johan Le Nagard**

LES SERVICES

1) Facebook

Nom de code de l'action	Description de l'action
-------------------------	-------------------------

GotMessage	L'action se déclenche lorsque l'utilisateur reçoit un message sur Facebook.
GotLike	L'action se déclenche lorsque l'utilisateur reçoit un like sur Facebook.
GotComment	L'action se déclenche lorsque l'utilisateur reçoit un commentaire sur un de ses posts.
GotNewPublish	L'action se déclenche lorsque quelqu'un publie dans le journal de l'utilisateur.
GotNewGroupPost	L'action se déclenche lorsque quelqu'un poste dans l'un des groupes de l'utilisateur.

Nom de code de la réaction	Description de la réaction	Paramètres de la réaction	Actions associées
WillSendMessage	Envoie un message à l'utilisateur passé en paramètres.	target: L'utilisateur ciblé content: Le contenu du message	*
WillLikePost	Envoie un like sur le message posté.	/	GotComment, GotNewPublish, GotNewGroupPost

- 2) Google
- 3) Twitter
- 4) Microsoft

GRANDES ÉTAPES

I - Back-end

- 1) Appréhender la technologie C# et ASP.NET Core.
- 2) Créer un controller d'API RESTFUL simple avec les verbes GET, POST, PUT et DELETE, pour obtenir, créer, modifier, supprimer des utilisateurs et créer un système d'inscription basique.

-
- 3) Appréhender l'API Firebase pour pouvoir stocker les utilisateurs dans une base de données plutôt que dans le cache du serveur.
 - 4) Représenter les services dans la base de données et créer un objet Service et un objet RegisteredService qui permettent aux utilisateurs de souscrire à des services.
 - 5) Appréhender les API utilisées par les différents services.
 - 6) Créer un CRON dans le serveur pour recharger les token API et pour obtenir les événements d'actions.
 - 7) Créer les actions et les réactions.

II - Front-end

- 1) Appréhender les technologies VueJS et Kotlin.
- 2) Appréhender l'API RESTFUL fournie par le serveur (API.md).
- 3) Créer les formulaires d'inscription/connexion et l'interface utilisateur.
- 4) Utiliser les API associées aux différents services pour récupérer les tokens d'accès.
- 5) Utiliser l'API serveur pour récupérer la liste des services, la proposer aux utilisateurs et leur permettre d'y souscrire.
- 6) Proposer à l'utilisateur de lier des actions et des réactions.
- 7) (Optionnel) Créer un CRON pour notifier l'utilisateur lorsqu'une action s'est déclenchée.