



Filière : Ingénierie pour la Finance

Construction d'application web

Projet Conseil en Investissement

Étudiants :

LE CALVEZ Mylène
VOLCIC Alexandre

Encadrant :

VIARDOT Sébastien

2018/2019

Table des matières

1	Présentation générale de l'application	2
1.1	Utilisation de l'application	2
2	Outils utilisés	3
2.1	Framework	3
2.2	Backend	3
3	Spécifications des fonctionnalités	6
3.1	Inscription	6
3.2	Connexion	6
3.3	Accueil	7
3.4	Catégories	7
3.4.1	Création	8
3.4.2	Suppression	8
3.5	Articles	8
3.5.1	Création	8
3.5.2	Suppression	8
3.6	Commentaires	9
3.6.1	Création	9
3.6.2	Suppression	9
3.7	Gestion des utilisateurs	10
4	Tests	11
4.1	Tests unitaires	11
4.2	Tests fonctionnels	11

1 Présentation générale de l'application

Le site « **Conseils en investissements** » a pour objectif d'afficher des conseils en investissements rédigés par des internautes. Le site permet de rechercher des conseils ou de les trier selon divers critères (date, popularité, commentaires).

N'importe qui peut consulter ou évaluer (*liker*) un article ou un commentaire. Un internaute qui souhaite faire plus que consulter le site (par exemple rédiger un conseil, laisser un commentaire,...) doit s'inscrire au site. On lui ouvre alors un compte sur le site ; lorsqu'il sera connecté sur ce compte, il pourra rédiger un article ou un commentaire. Certains utilisateurs sont en plus administrateurs : ils peuvent modifier le système de l'arbre des catégories (création, suppression de catégories), supprimer les conseils ou les commentaires. Ils ont également un rôle de gestion des utilisateurs. Les détails des spécifications de l'application sont dans la partie 3. ;

1.1 Utilisation de l'application

Après avoir lancé le serveur avec la commande `ng serve` dans un terminal, le site est accessible à l'adresse `http://localhost:4200/`.

Vous pouvez être un **lecteur** sans vous authentifier. Nous vous avons ensuite créé deux comptes de tests :

Compte auteur

- *email* : `testAuteur@mail.fr`
- *mot de passe* : `testAuteur`

Vous pouvez également vous inscrire pour créer un nouveau compte auteur.

Compte administrateur

- *email* : `testAdmin@mail.fr`
- *mot de passe* : `testAdmin`

2 Outils utilisés

2.1 Framework

Nous avons décidé de travailler avec le *framework* **Angular**. Nous avons utilisé l'IDE **Visual Studio Code** sur **Windows** pour développer l'application.

Nous avons utilisé les matériaux **Angular** dans notre code.

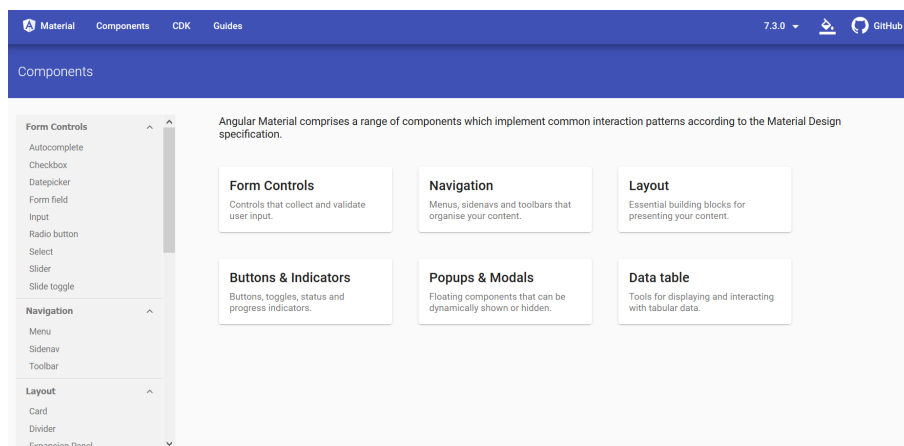
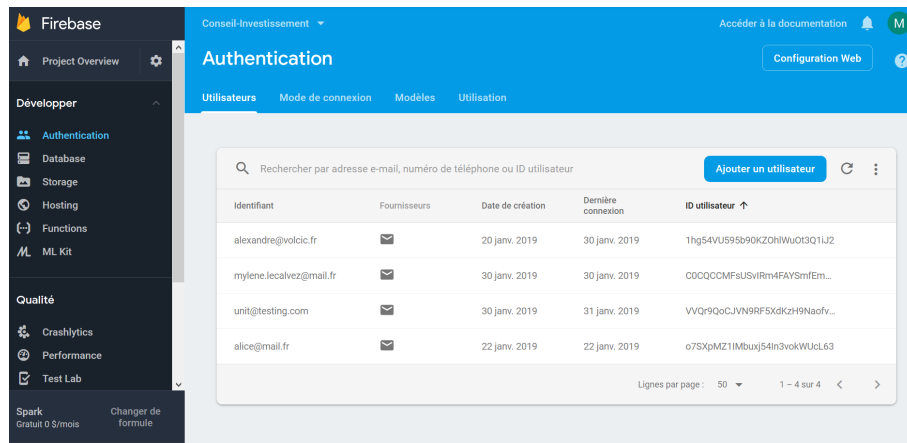


FIGURE 1 – Extrait de la page Angular Material

2.2 Backend

Nous avons choisi d'utiliser **Firestore**, la plateforme mobile de Google facilitant la création de Backend *scalable* et performant. Firestore représente la partie de communication Client-Serveur de notre application web, sur laquelle nous allons pouvoir déporter et centraliser des responsabilités comme l'authentification, la gestion des rôles et le stockage de données.

La figure 2 est un extrait de la partie **Authentication** dans la console **Firestore**.

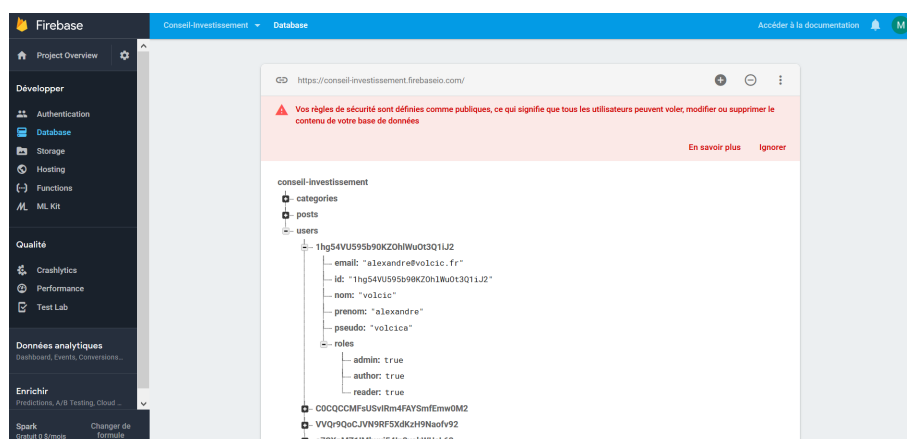


The screenshot shows the Firebase Authentication console for the project 'Conseil-Investissement'. The left sidebar contains navigation links for Project Overview, Authentication, Database, Storage, Hosting, Functions, ML Kit, and Quality (Crashlytics, Performance, Test Lab). The main area displays a table of users with columns: Identifiant, Fournisseurs, Date de création, Dernière connexion, and ID utilisateur. A search bar at the top allows filtering by email, phone number, or user ID. A table with 4 rows of user data is shown.

Identifiant	Fournisseurs	Date de création	Dernière connexion	ID utilisateur
alexandre@voicic.fr	📧	20 janv. 2019	30 janv. 2019	1hg54VU595b9KZOh1Wu0t3Q1LJ2
mylene.lecalvez@mail.fr	📧	30 janv. 2019	30 janv. 2019	C0CQCCMFsUSvIRm4FAYSmfEm...
unit@testing.com	📧	30 janv. 2019	31 janv. 2019	VVQr9QeCJVN9RF5dKzH9Naofv...
alice@mail.fr	📧	22 janv. 2019	22 janv. 2019	o7SxpMZ1IMbuxj54in3vokWUcl63

FIGURE 2 – Extrait de la base de données de l'application web

La base de données que nous avons utilisée, **Firebase Realtime Database**, est une base de données hébergée dans le *cloud*. Les données sont stockées sous forme de **JSON** et synchronisées en temps réel sur chaque client connecté. Tous les clients partagent une instance de base de données en temps réel et reçoivent automatiquement les mises à jour avec les données les plus récentes. Les figures 3, 4 et 5 sont des extraits de la base de données **Firebase Realtime Database** dans la console **Firebase**. La figure 3 montre un exemple d'utilisateur possédant les rôles **Admin** et **Author**, le rôle **Reader** étant assigné par défaut à tous les utilisateurs. Cet utilisateur a des privilèges par rapport à l'utilisateur présenté dans la figure 4 qui n'est que lecteur et auteur.



The screenshot shows the Firebase Realtime Database console for the project 'Conseil-Investissement'. The left sidebar is the same as in Figure 2. The main area displays a JSON tree structure for the database. A warning message at the top states: 'Vos règles de sécurité sont définies comme publiques, ce qui signifie que tous les utilisateurs peuvent voler, modifier ou supprimer le contenu de votre base de données.' The JSON structure shows a root node 'conseil-investissement' with children 'categories', 'posts', and 'users'. The 'users' node contains a user object with fields: email, id, nom, prenom, pseudo, and roles. The roles array includes 'admin', 'author', and 'reader'.

```

conseil-investissement
├── categories
├── posts
└── users
    ├── 1hg54VU595b9KZOh1Wu0t3Q1LJ2
    │   ├── email: "alexandre@voicic.fr"
    │   ├── id: "1hg54VU595b9KZOh1Wu0t3Q1LJ2"
    │   ├── nom: "voicic"
    │   ├── prenom: "alexandre"
    │   ├── pseudo: "voicic"
    │   └── roles
    │       ├── admin: true
    │       ├── author: true
    │       └── reader: true
    ├── C0CQCCMFsUSvIRm4FAYSmfEmw0M2
    ├── VVQr9QeCJVN9RF5dKzH9Naofv92
    └── o7SxpMZ1IMbuxj54in3vokWUcl63
  
```

FIGURE 3 – Extrait de la base de données de l'application web

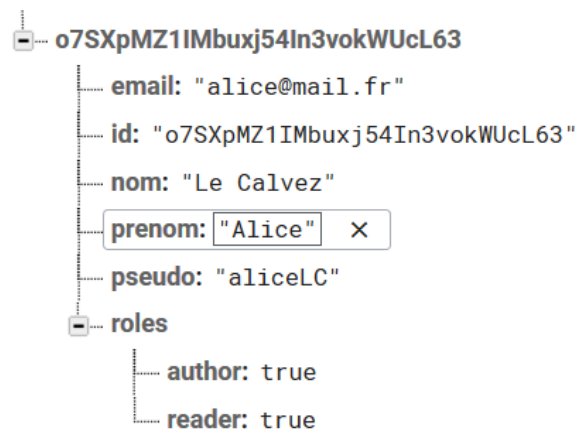


FIGURE 4 – Extrait de la base de données de l'application web

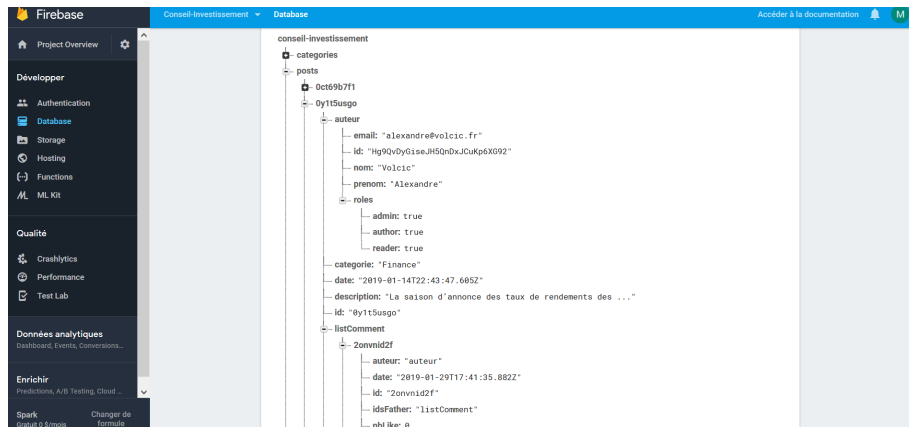


FIGURE 5 – Extrait de la base de données de l'application web

3 Spécifications des fonctionnalités

3.1 Inscription

La page d'inscription est accessible à tous. Un utilisateur a besoin d'indiquer différents éléments pour s'inscrire (*cf fig 6*). Le pseudo et le mot de passe doivent avoir une longueur minimale de 6 caractères chacun. L'adresse mail doit être au bon format. L'unicité du compte est assurée par l'adresse mail : une même adresse mail ne peut servir que pour un unique compte (sinon un message d'erreur est levé).


The image shows a web interface for a site titled "Conseils en investissement". At the top, there is a navigation bar with three links: "Accueil", "S'inscrire", and "Se connecter". The "S'inscrire" link is highlighted. Below the navigation bar, there is a registration form titled "S'inscrire". The form contains five input fields: "Nom", "Prénom", "Pseudo", "Adresse mail", and "Mot de passe". At the bottom right of the form is a button labeled "Inscription".

FIGURE 6 – Extrait de la page d'inscription

3.2 Connexion

L'utilisateur se connecte avec son adresse mail et son mot de passe (*cf fig 9*). S'il se trompe de mot de passe, une erreur est levée.

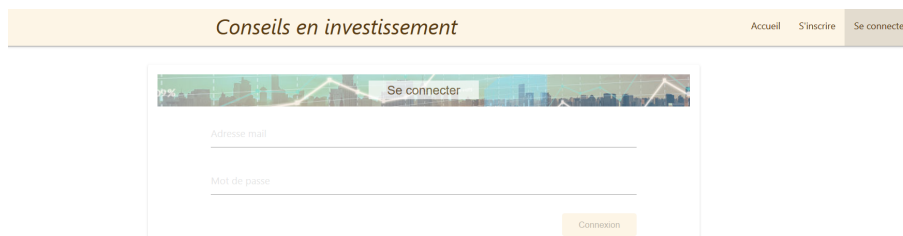
The image shows a web interface for a site titled "Conseils en investissement". At the top, there is a navigation bar with three links: "Accueil", "S'inscrire", and "Se connecter". The "Se connecter" link is highlighted. Below the navigation bar, there is a login form titled "Se connecter". The form contains two input fields: "Adresse mail" and "Mot de passe". At the bottom right of the form is a button labeled "Connexion".

FIGURE 7 – Extrait de la page de connexion

3.3 Accueil

La page d'accueil du site permet d'accéder rapidement aux conseils en investissement : on y trouve notamment la liste des 10 derniers conseils, un outil de recherche, et l'arbre des catégories. Il est possible de choisir le nombre d'articles à afficher et de trier selon la date, la popularité ou le nombre de commentaires.



FIGURE 8 – Extrait de la page d'accueil

3.4 Catégories

Un article peut appartenir à une catégorie ou à une sous-catégorie. Il ne peut pas être répertorié dans plusieurs catégories à la fois. La rubrique "Tous" permet de voir tous les articles. Les catégories par défaut sont :

- Généralités sur l'investissement
 - Développement personnel
 - Retour d'expérience
- Finance
 - Assurance Vie
 - Rendement du fonds euro
 - Actualité assurance vie
- Immobilier
 - Crédit immobilier
 - Investissement immobilier
- Internet

— Bourse

Les catégories sont disponibles sur la barre latérale à gauche de l'écran d'accueil. Il est possible de fermer cette section. Lorsque l'utilisateur choisit une catégorie, on affiche tous les conseils de cette catégorie.

3.4.1 Création

Seul un administrateur peut créer une catégorie. Il doit simplement indiquer le nom de la nouvelle catégorie.

3.4.2 Suppression

Seul un administrateur peut supprimer une catégorie.

3.5 Articles

Un article de conseil en investissement est de la forme suivante :



FIGURE 9 – Exemple d'un article

3.5.1 Création

Les auteurs et les administrateurs peuvent rédiger des articles.

3.5.2 Suppression

Les administrateurs seulement peuvent supprimer des articles.

3.6 Commentaires

Une liste de commentaire est associée à un article. Les commentaires peuvent être hiérarchiques (il est possible de commenter l'article ou de commenter un autre commentaire). Les figures



FIGURE 10 – Exemple de commentaires - vue base de données



FIGURE 11 – Exemple de commentaires - vue site

3.6.1 Création

Les auteurs et les administrateurs peuvent rédiger des commentaires.

3.6.2 Suppression

Les administrateurs seulement peuvent supprimer des commentaires.

3.7 Gestion des utilisateurs

Les administrateurs ont un droit de gestion des utilisateurs : consulter les inscrits et leurs rôles, supprimer un utilisateur si besoin.





Conseils en investissement					
Accueil Gestion des utilisateurs Se déconnecter					
Rechercher					
Nom	Prénom	Pseudo	Adresse mail	Rôles	Supprimer
volcic	alexandre	volcica	alexandre@volcic.fr	lecteur, auteur, admin	
Le Calvez	Mylene	myleneLC	mylene.lecalvez@mail.fr	lecteur, auteur, admin	
Volcic	Alexandre	volcica	unit@testing.com	lecteur	
Le Calvez	Alice	aliceLC	alice@mail.fr	lecteur, auteur	
Items per page: 5 1 - 4 of 4 < >					

FIGURE 12 – Extrait de la page de gestion des utilisateurs

4 Tests

4.1 Tests unitaires

Un test unitaire est un procédé permettant de s'assurer du bon fonctionnement d'une unité de programme. Le testing unitaire repose au fond sur un principe très simple : un système aussi gros et complexe qu'il soit est toujours composé de différentes parties plus petites que lui. Il est indispensable (mais pas suffisant) que chacune de ces parties remplisse correctement sa fonction pour que l'ensemble soit fonctionnel et fiable.

Nous avons testé les méthodes des services (authentification, catégories, articles, commentaires). Les tests permettent de vérifier le bon fonctionnement de la gestion des droits, ainsi que de la création ou suppression des différentes entités de notre application web. Les tests sont accessibles par la page `test` de notre application (après avoir lancé le serveur avec `ng serve`, se rendre à l'adresse `http://localhost:4200/test`). A l'exécution de chaque test, un message s'affiche pour indiquer si le test est validé ou non ("**Test validé [...]**" ou "**Test échoué [...]**"). Lorsque l'on teste un scénario qui doit provoquer une erreur, le message d'erreur renvoyé est affiché. Lorsque l'on teste la création d'un objet (un article par exemple), nous avons pris soin de le supprimer à la fin du test afin de ne pas créer des objets abusivement sur l'application web. La figure 13 est un extrait de la page de test lorsque le serveur est lancé.

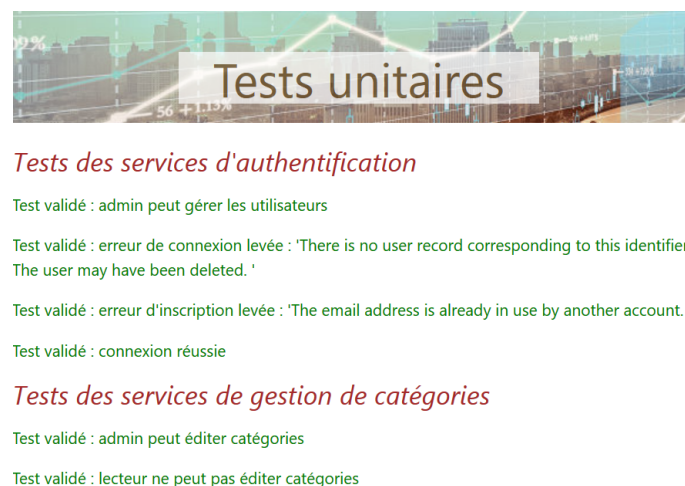


FIGURE 13 – Extrait de la page de tests unitaires des méthodes de service

4.2 Tests fonctionnels

Un test fonctionnel consiste à s'assurer qu'un système (ou un de ses composants) fonctionne adéquatement. Les tests fonctionnels peuvent être effectués durant la phase de développement, au cas par cas pour des sections spécifiques développées par l'équipe, ou

au terme du développement du projet, une fois qu'il est complet. L'objectif principal de ce type de test est d'analyser le produit fini et de déterminer s'il répond bien aux attentes du client, sur le point technique.

Nous avons essayé de réaliser des tests *end-to-end* avec **Protractor**, qui est un *framework* de test *e2e* développé pour Angular. Nous avons créé quelques fichiers dans le dossier `conseil-investissement/e2e/src/`. Cependant, nous n'avons pas réussi à résoudre les erreurs soulevées par ces tests fonctionnels.