

Développement d'une application de prise de note avec ionic de A à Z



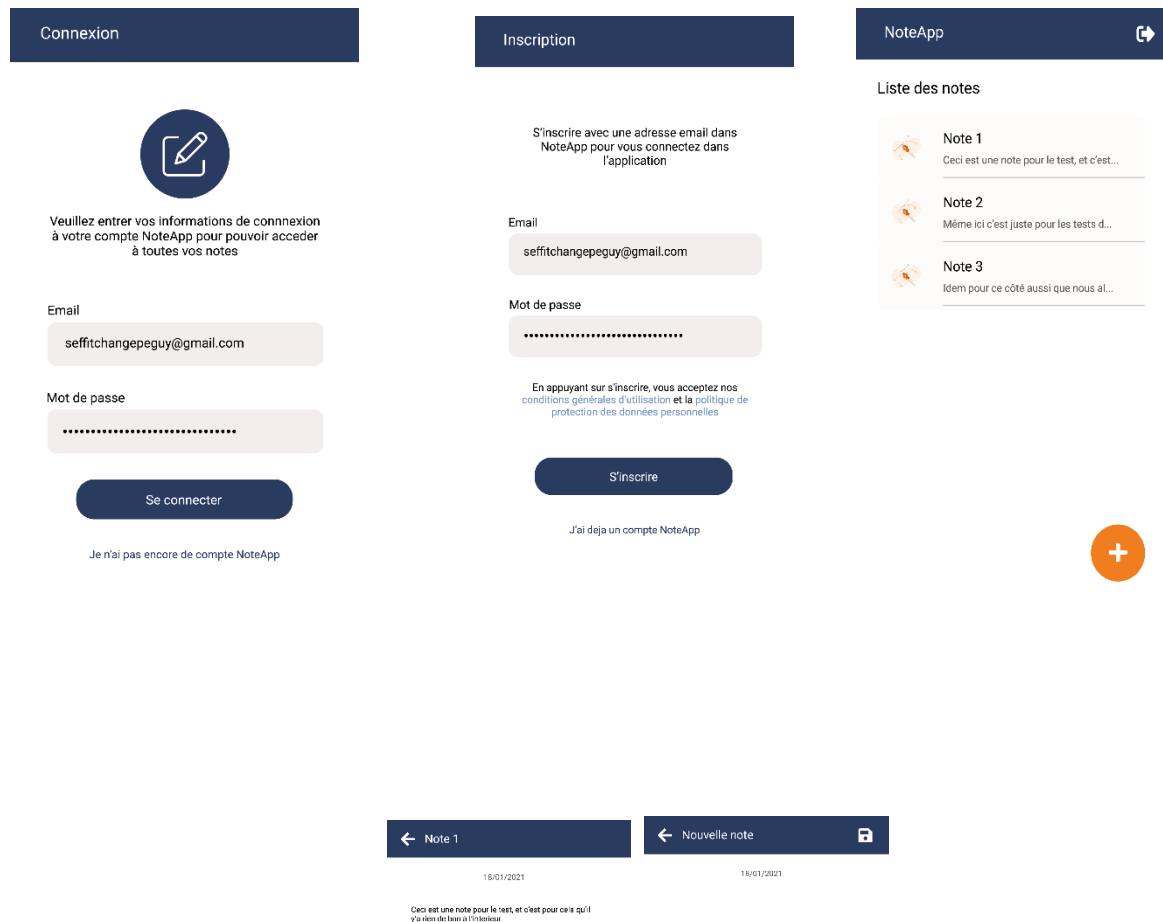
Par SEFFI TCHANGA PEGUY

Et coordonné par Dr Jiomekong Azanzi

Aperçus de l'application :

Maquette disponible à ce lien pour les différents tests

<https://www.figma.com/file/4vrkNT2cHroGjHh5hUEQGk/NoteApp?node-id=0%3A1>



Ionic est un **framework open-source** créé en 2013 par Max Lynch, Ben Sperry, et Adam Bradley. Deux versions distinctes sont disponibles, incompatibles entre elles : la première version, 1.3.3, se base sur AngularJS 1.5.3 tandis que la version 3.5.0 se base sur Angular 4.1.3 et TypeScript.

1- Prérequis :

- **Connaissances**

- HTML5, CSS3, JavaScript et dans la mesure du possible TypeScript
- Le Angular (mais dans ce tutoriel nous expliquerons le plus possible les codes angular utilisés).

- **Outils et matériels**

- Un Ordinateur tournant sur l'un de ces systèmes d'exploitation : Windows, Linux ou MacOS et pouvant supporté Android studio pour la génération de l'apk (mais il n'est pas nécessaire pour la vue de votre application, nous utiliserons un navigateur pour cela).
- Une connexion à internet
- Un navigateur Web (Google chrome, Mozilla Firefox, Brave, Edge, et autres...)
- Avoir un compte Google pour la gestion de la base de donnée.
- Un cerveau hyper motivé sera aussi la bienvenue 😊

Dans le cadre de ce tutoriel, nous utiliserons ionic 5 avec angular. Nous développerons une application de prise de notes simple avec des fonctionnalités de base. Comme base de donnée nous utiliserons **Firestore**. A la fin de ce tutoriel, nous serez capable de créer des pages d'inscription et de connexion avec authentification par email et mot de passe. De créer des formulaires, et d'effectués des opérations de lecture et d'écriture dans la base de donnée.

Dans ce tutoriel nous utiliserons linux, plus particulièrement la distribution Ubuntu 20.04. Mais vous pouvez utilisé d'autres distribution de linux, et les systèmes d'exploitation propriétaire tel que Windows de Microsoft et MacOS de Apple.

Ne vous inquiétez surtout pas si tout parait compliqué au début, ça arrive à tout le monde, mais avec le temps sa sera un jeux d'enfant pour vous 😊.

Bon passons aux choses sérieuses...

2- Installation du nécessaire à l'environnement de développement

Ouvrir votre terminal pour les utilisateurs de Linux, MacOS et l'invite de commande (cmd) pour les utilisateurs de Windows, nous allons installer le CLI de Ionic. Le CLI, ou "Command Line Interface" d'Ionic est un outil permettant d'exécuter des commandes depuis la console, permet d'exécuter des scripts pour la création, la structuration et la production d'une application Ionic. Pour cela nous allons installer quelque outil au préalable.

a) Installation de nodeJs

Rendez-vous sur le lien <https://nodejs.org/en/download/>. Vous télécharger juste le logiciel (.exe pour Windows puis vous l'installez), et les paquets pour Linux et MacOS.

Sur Ubuntu vous pouvez utiliser les commandes :

- **sudo apt-get update**
- **sudo apt-get install nodejs npm**

b) Installation de npm

Npm est le gestionnaire de paquets officiel de Node.js. Il fonctionne avec un terminal et gère les dépendances pour une application. Peu importe le système, vous tapez la commande **npm install -g npm@latest**. (N'oubliez pas de commencer par **sudo** pour les utilisateurs de Linux).

c) Installation de ionic

A présent nous allons installer Ionic et Cordova dans tout le système avec la commande :

npm install -g ionic cordova. (N'oubliez pas de commencer par **sudo** pour les utilisateurs de Linux).

d) Initialisation de notre projet

Placez-vous d'abord dans le dossier où vous souhaitez héberger votre application comme par exemple **Documents**. Ensuite vous tapez la commande :

ionic start NoteApp --capacitor. (N'oubliez pas de commencer par **sudo** pour les utilisateurs de linux).

e) Lancement du serveur

Pour démarrer le serveur, placez-vous dans votre dossier de projet et exécutez la commande **ionic serve**. Et lorsque le serveur est démarré (vous devez voir apparaître le message **Compiled successfully**), allez sur votre navigateur favori et tapez le lien **localhost :8100**.

3- Configuration de votre éditeur de travail

Vous pouvez utiliser n'importe quel éditeur de code (vs code, web Storm, IntelliJ et autre...). Dans ce tutoriel nous utilisons IntelliJ, mais vous pouvez utiliser un autre sans problème. Maintenant vous allez soit ouvrir votre éditeur à partir de votre dossier de projet ou à défaut ouvrir votre éditeur et ouvrir le dossier de votre projet à l'intérieur.

4- Création et configuration de la base de donnée

Nous allons ajouter notre base de donnée firebase à notre projet, pour cela, tapez la commande **npm install firebase -- save**. Après vous devez vous rendre sur le site <https://firebase.google.com/> pour créer votre base de donnée en ligne (un compte Google est nécessaire pour cela). Authentifiez-vous et créez un nouveau projet Firebase. Vous pouvez le domicilier dans un pays de votre choix et lui donner le nom **NoteApp**. Ensuite de dashboard de Firebase vous propose les choix sous la rubrique Overview. Sélectionner **Ajouter Firebase à votre application Web**, mettez comme pseudo nom de l'application **NoteApp** puis copiez-collez la configuration dans le constructeur de votre **AppComponent** (en ajoutant **import * as firebase from 'firebase'**; En haut du fichier mettant à disposition la méthode **initializeApp()**) :

```

import { Component } from '@angular/core';
import { Platform } from '@ionic/angular';
import { SplashScreen } from '@ionic-native/splash-screen/ngx';
import { StatusBar } from '@ionic-native/status-bar/ngx';
import firebase from 'firebase';

@Component({
  selector: 'app-root',
  templateUrl: 'app.component.html',
  styleUrls: ['app.component.scss']
})
export class AppComponent {
  constructor(
    private platform: Platform,
    private splashScreen: SplashScreen,
    private statusBar: StatusBar
  ) {
    this.initializeApp();
  }

  initializeApp() {
    this.platform.ready().then(() => {

      const firebaseConfig = {
        apiKey: 'AIzaSyB-Epxr1bqa0wPQmEPxa_tDe20Naq901HE',
        authDomain: 'noteapp-90fb8.firebaseio.com',
        projectId: 'noteapp-90fb8',
        storageBucket: 'noteapp-90fb8.appspot.com',
        messagingSenderId: '334499608663',
        appId: '1:334499608663:web:a3fbb705e24729c4204b12'
      };
      // Initialize Firebase
      firebase.initializeApp(firebaseConfig);

      this.statusBar.styleDefault();
      this.splashScreen.hide();
    });
  }
}

```









Ensuite rendez-vous dans authentification pour gérer la connexion/inscription des utilisateurs et activer **Adresse e-mail/Mot de passe**.

Authentication

CONFIGURATION WEB ?

UTILISATEURS
MODE DE CONNEXION
MODELES
UTILISATION

Fournisseurs de connexion

Fournisseur	État
 Adresse e-mail/Mot de passe	Activé
 Téléphone	Désactivé
 Google	Désactivé
 Play Jeux	Désactivé
 Facebook	Désactivé
 Twitter	Désactivé
 GitHub	Désactivé
 Anonyme	Désactivé

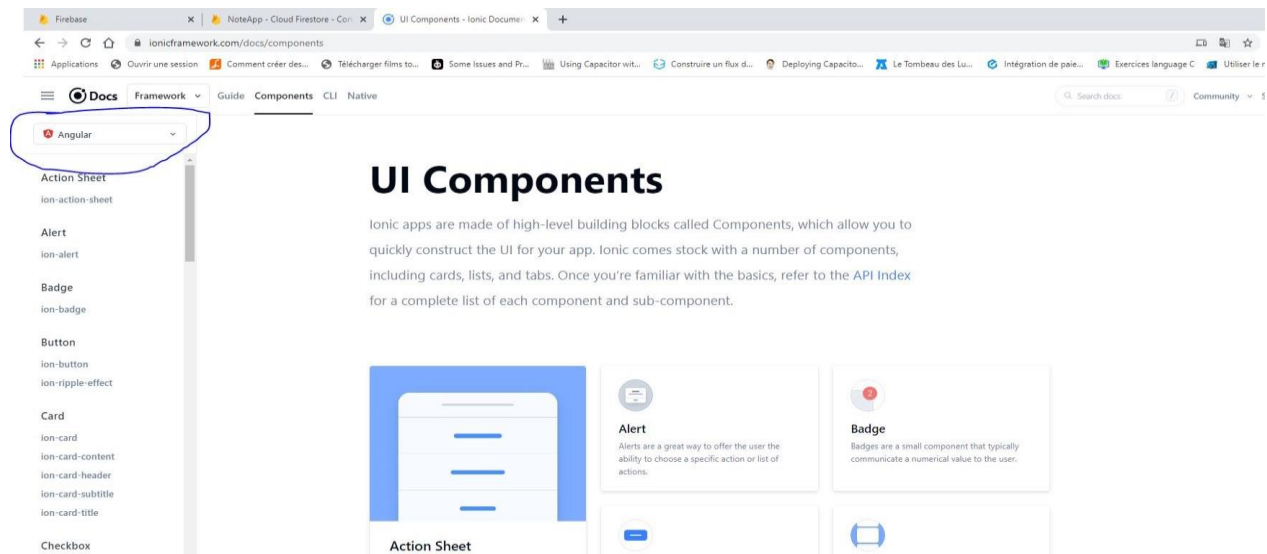
Ensuite rendez-vous dans **Cloud Firestore** et cliquer sur **commencer**. Sous l'onglet **Rules** modifier le **false** en **true**.

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /{document=**} {
      allow read, write: if true;
    }
  }
}
```

Voilà, vous êtes prêt à interagir avec votre base de donnée firebase !

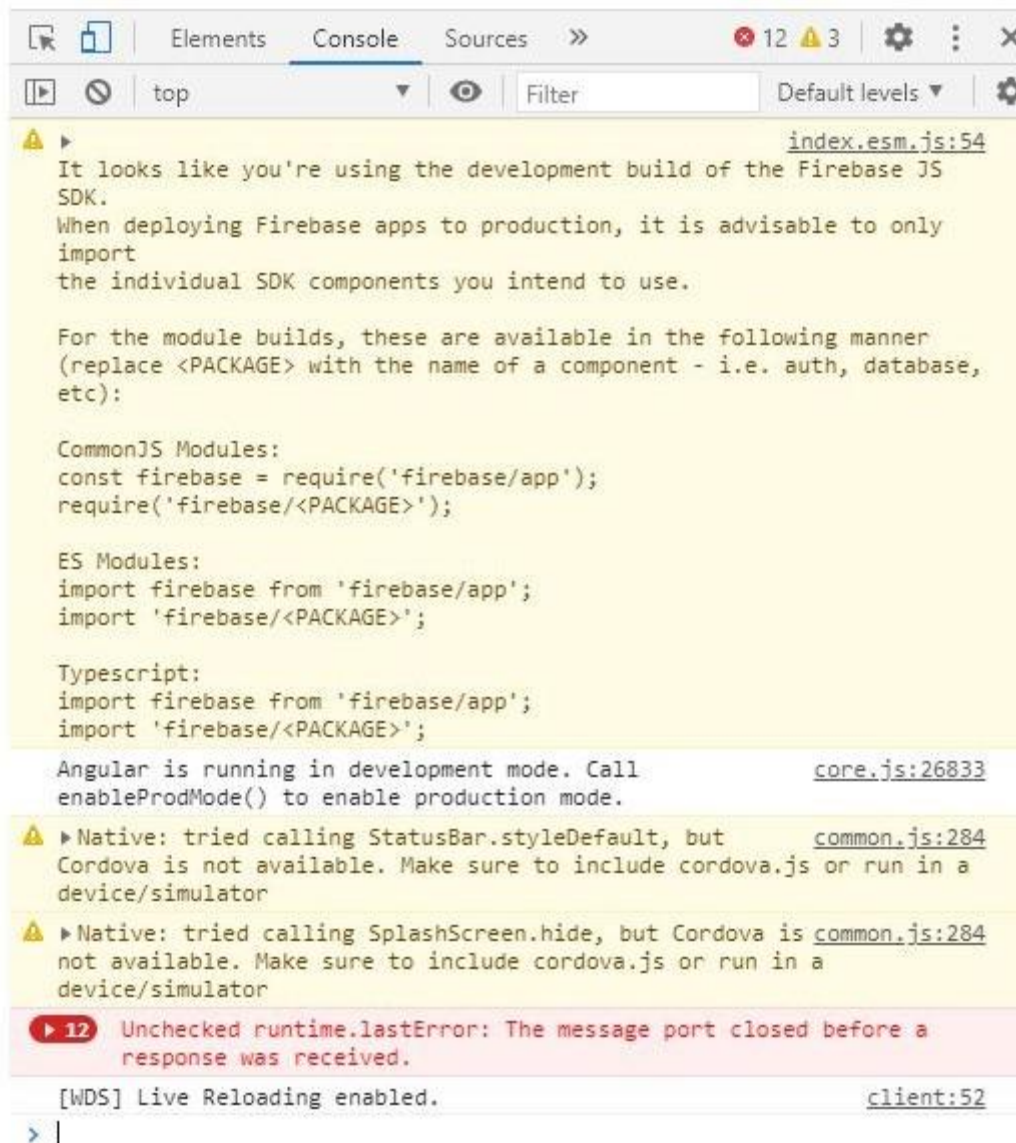
5- Astuces et autres

Notre application ionic est constituer de composant, avec des visuels proches de ceux des applications mobiles. Ces composants sont disponibles à l'adresse <https://ionicframework.com/docs/components>.



Vérifier que la partie entourée sur cette capture est bien positionné sur **Angular**.

Lors du développement de notre application, nous pouvons avoir des erreurs ou affiché des texte pour le débogage, nous devons donc ouvrir l'outil de développement dans notre navigateur sur la page de notre application. Pour ce faire, rendez vous dans les options de notre navigateur -> **Plus d'outils** -> **Outils de développement**.



```

  Elements  Console  Sources  >>
  top
  Filter
  Default levels
  index.esm.js:54
  It looks like you're using the development build of the Firebase JS
  SDK.
  When deploying Firebase apps to production, it is advisable to only
  import
  the individual SDK components you intend to use.

  For the module builds, these are available in the following manner
  (replace <PACKAGE> with the name of a component - i.e. auth, database,
  etc):

  CommonJS Modules:
  const firebase = require('firebase/app');
  require('firebase/<PACKAGE>');

  ES Modules:
  import firebase from 'firebase/app';
  import 'firebase/<PACKAGE>';

  Typescript:
  import firebase from 'firebase/app';
  import 'firebase/<PACKAGE>';

  Angular is running in development mode. Call
  enableProdMode() to enable production mode.
  core.js:26833

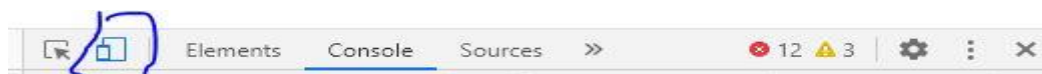
  ▶ Native: tried calling StatusBar.styleDefault, but
  Cordova is not available. Make sure to include cordova.js or run in a
  device/simulator
  common.js:284

  ▶ Native: tried calling SplashScreen.hide, but Cordova is
  not available. Make sure to include cordova.js or run in a
  device/simulator
  common.js:284

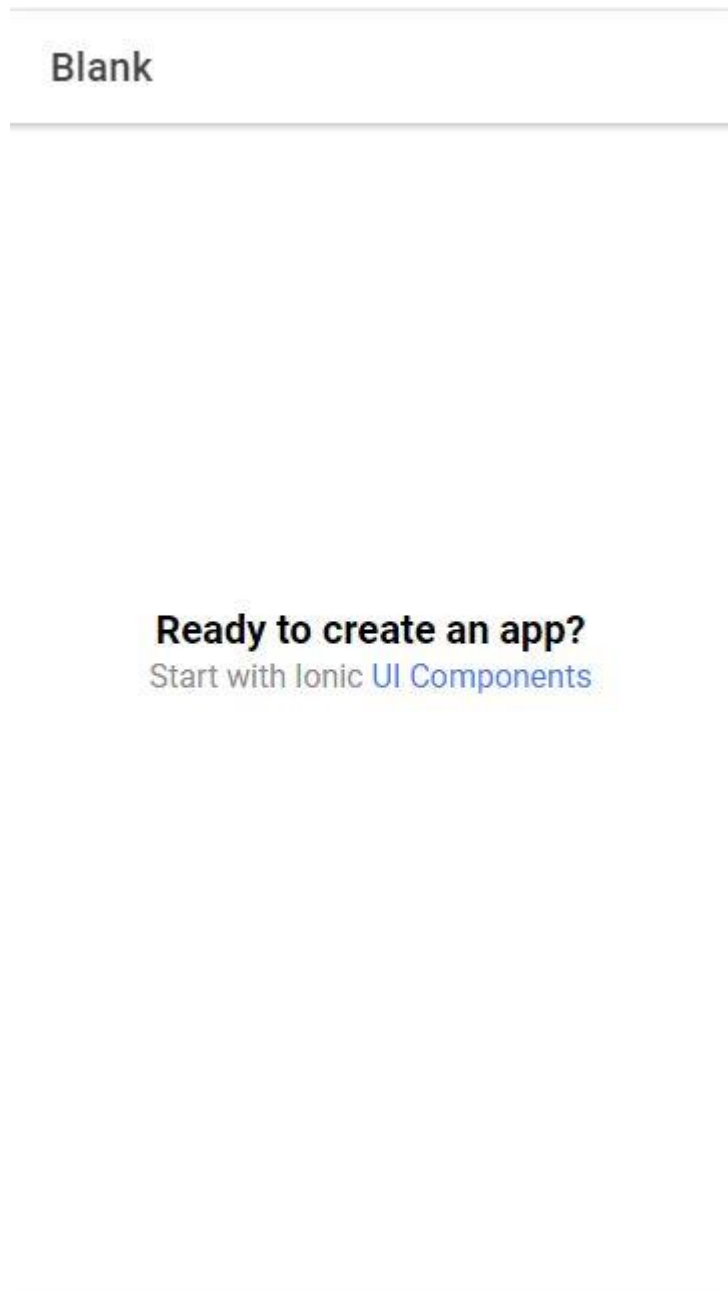
  ▶ 12 Unchecked runtime.lastError: The message port closed before a
  response was received.

  [WDS] Live Reloading enabled.
  client:52
  > |
```

Et pour avoir un aperçu comme sur mobile, nous allons cliquer sur ce petit bouton :



Et nous avons donc notre application vierge :



6- Développement de l'application

La suite du programme sera fait sur le tutoriel vidéo. Suivez ces étapes pas à pas pour la creation des composants, services, model et autre de notre application. Ainsi que le build et l'ajout de notre projet dans android studio et la génération de l'application.

Code source disponible ici : <https://github.com/paulmexa/projet-noteapp.git>

Bibliographie

- <https://wikipedia.org/>
- <https://openclassrooms.com/>
- <https://www.figma.com/>
- <https://github.com/>