

Rapport du projet Ionic

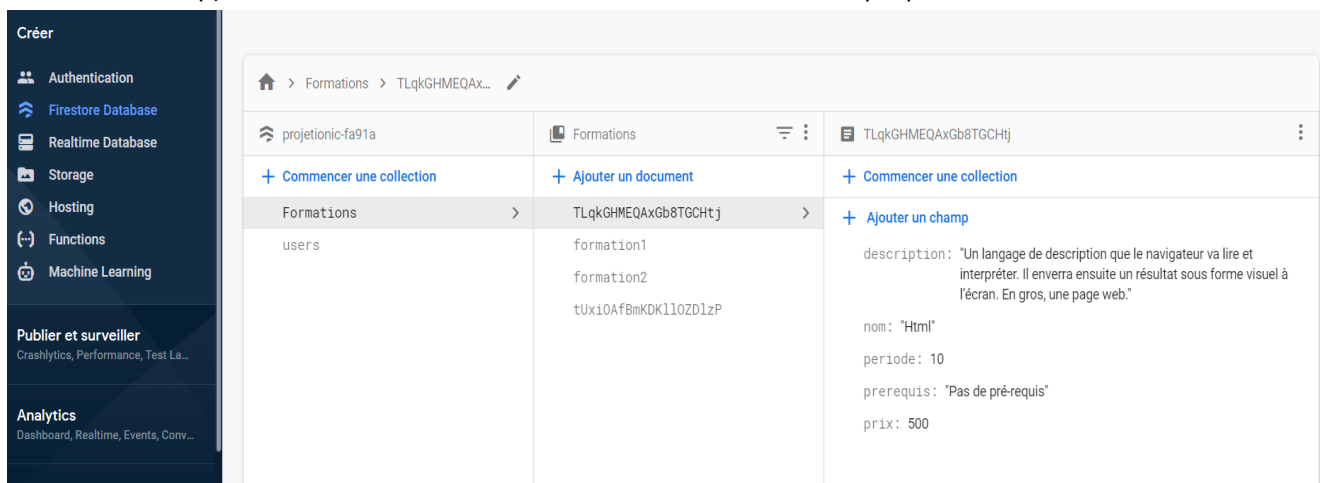
I. Création du projet et liaison avec firebase :

1. Création du projet

On a commencé par créer un nouveau projet Ionic en utilisant la commande « **ionic start Nom-projet** » et on a réglé les paramètres convenables

2. Création de la base de données

Il fallait après créer une nouvelle base de données en firebase (les étapes suivis dans le tp). Et on a crée une collection contenant les formations proposées



3. Liaison de la base de données avec l'application

On a lié ensuite la base de données avec l'application ionic en suivant les étapes convenables

```
export const environment = {
  production: false,
  firebaseConfig: {
    apiKey: "AIzaSyCAzkQvHs99S1NnvUPWfGL8CbYGOxqj_c",
    authDomain: "projeticion-fa91a.firebaseio.com",
    projectId: "projeticion-fa91a",
    storageBucket: "projeticion-fa91a.appspot.com",
    messagingSenderId: "77438804862",
    appId: "1:77438804862:web:f143515d3e098ca88f092f"
  },
};
```

Environnement.ts

App.module.ts

```
// Firebase
import { AngularFireModule } from '@angular/fire/compat';
import { AngularFireAuthModule } from '@angular/fire/compat/auth';
import { environment } from '../environments/environment';
import { CommonModule } from '@angular/common';

@NgModule({
  declarations: [AppComponent],
  entryComponents: [],
  imports: [
    CommonModule,
    BrowserModule,
    IonicModule.forRoot(),
    AppRoutingModule,
    AngularFireModule.initializeApp(environment.firebaseConfig),
    AngularFireAuthModule
  ],
  providers: [
  ],
})
```

II. Création des pages et des services :

On a créé après les pages et services nécessaires pour notre application

ionic g create page registration

ionic g create page login

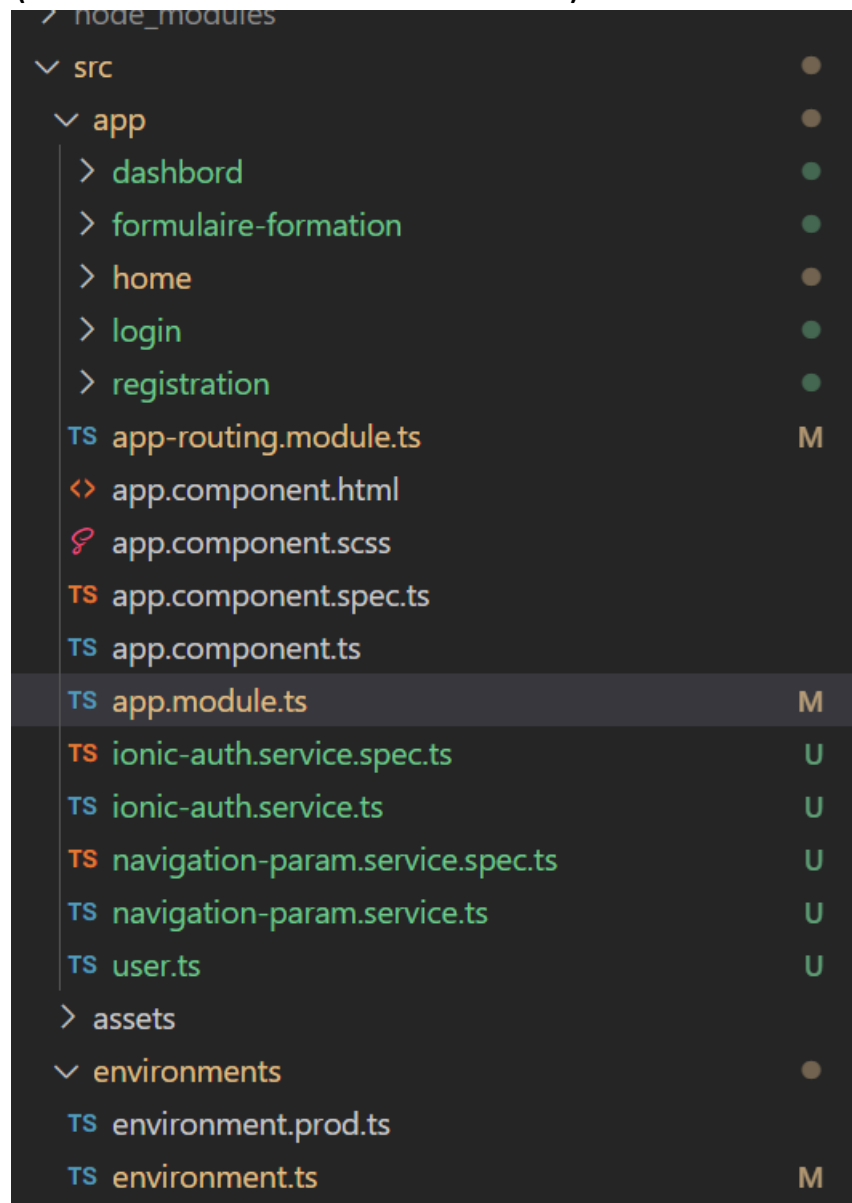
ionic g create page dashboard

ionic g create page formulaire-formation

ionic g service ionic-auth (service d'authentification)

ionic g service navigation-param (pour passer les données entre les pages)

User.ts (créé manuellement et contient la classe user)



III. Contenu des pages et services

1. Service authentification

Le service authentification contient les méthodes de création utilisateur

```
createUser(value) {  
  return new Promise<any>((resolve, reject) => {  
    this.angularFireAuth.createUserWithEmailAndPassword(value.email, value.password)  
      .then(  
        res => resolve(res),  
        err => reject(err))  
  })  
}
```

une méthode de connexion

```
signinUser(value) {  
  return new Promise<any>((resolve, reject) => {  
    this.angularFireAuth.signInWithEmailAndPassword(value.email, value.password)  
      .then(  
        res => resolve(res),  
        err => reject(err),  
      )  
  })  
}
```

Une méthode de deconnexion

```
signoutUser() {  
  return this.angularFireAuth.signOut().then(() => {  
    console.log('Logged out!');  
  });  
}
```

Une méthode pour enregistrer les données

```
saveDetails(data) {  
  return this.firestore.collection("users").doc(data.uid).set(data);  
}
```

Deux méthode pour récupérer les données

```
userDetails() {  
  return this.angularFireAuth.user  
}  
getDetails(uid : string) {  
  return this.firestore.collection('users').doc(uid).valueChanges();  
}
```

2. Page registration

Elle contient

Registration.page.html : contient un Reactive form, l'utilisateur doit remplir les champs et cliquer sur enregistrer pour s'inscrire

Registration.page.ts : contient une initiation du formulaire, une méthode d'inscription et une méthode pour naviguer vers le login

3. Page login

Elle contient

login.page.html : contient un Reactive form, l'utilisateur doit remplir les champs et cliquer sur se connecter pour le faire

login.page.ts : contient une initiation du formulaire, une méthode de connexion et une méthode pour naviguer vers le dashboard

4. Page dashboard

Elle contient

dashboard.page.html : contient une liste de formations proposées importées de la base de données et une liste des cours dans lesquelles l'utilisateur est inscrit

dashboard.page.ts : contient une initiation des formations ainsi qu'une initiation de l'objet user passé en paramètres de la page login, et une méthode de déconnexion

5. Page formulaire-formation

Elle contient

Formulaire-formation.page.html : contient un Reactive form, les champs nom et email sont déjà remplis par les données de l'utilisateur importés de la base de données, dès que l'utilisateur ajoute son numéro de téléphone il sera lui aussi ajouté automatiquement, en cliquant sur le bouton valiser il sera redirigé vers la page dashboard où il va visualiser la formation dans laquelle il s'est inscrit

Formulaire-formation.page.ts : contient une initiation du formulaire, une méthode d'ajouter les infos rentrés