

Atelier 10

Matière : ATELIER DEVELOPPEMENT MOBILE NATIF

DSI3

Enseignants : S. Hadhri & M. Hadiji & H. Souissi

L'objectif de ce TP est de manipuler une base de données firebase firestore.

Les interfaces de l'application sont les suivantes



The image displays four screenshots of a mobile application interface, each with a light purple background and rounded rectangular buttons.

- Ajout Screen:** Features a title field with "Android 14" and a "Nb Page" field with "326". Below these are two buttons: "Ajouter" and "Retour".
- Consultation Screen:** Features a "Retour" button at the top, followed by a list of items: "Angular-412", "Flutter-571", "Android 14-326", and "STQB-352".
- Modification Screen:** Features a "Livres" dropdown menu with "Android 14-326" selected, a title field with "Android 14", and a "Nb Page" field with "362". Below these are two buttons: "Modifier" and "Retour".
- Suppression Screen:** Features a "Livres" dropdown menu with "Angular-412" selected, followed by two buttons: "Supprimer" and "Retour".

Etape 1 : Création du projet

1. Créer un nouveau projet avec Android Studio nommé « GestionBibFirestore »
2. Modifier le nom du package à « com.firestore »
3. Créer 4 autres activités : Ajout, Consultation, Modification et Suppression.
4. Télécharger et copier les fichiers squelettes à partir de la plateforme moodle

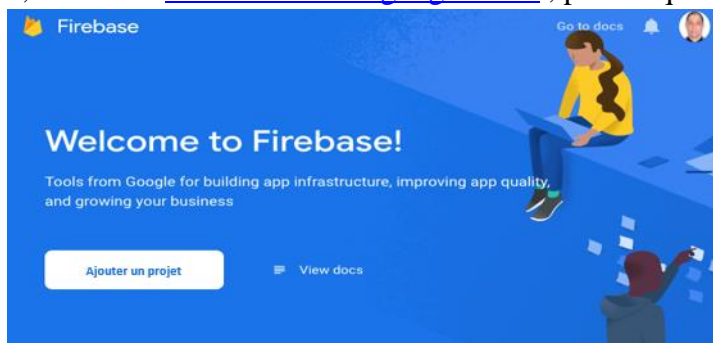
Etape 2 : Présentation & configuration de Firebase

Firebase est une plate-forme fournie par Google pour activer votre application mobile. Firebase regorge de fonctionnalités importantes telles que l'authentification des utilisateurs et la messagerie cloud.

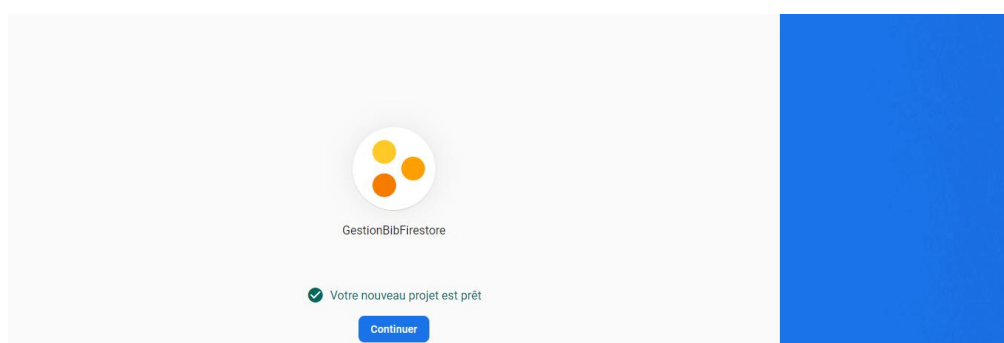
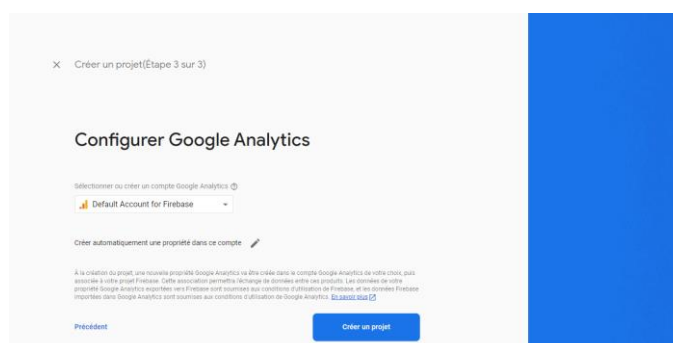
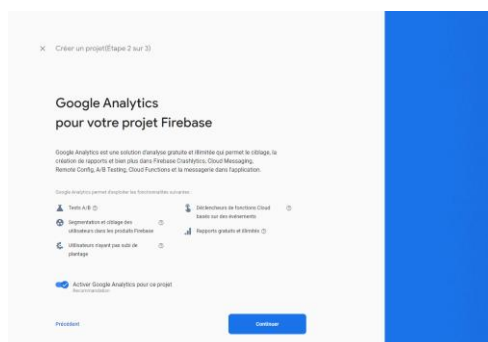
Dans cet atelier, nous allons utiliser le service base de données FirebaseFirestore dans notre application.

5. Configuration du projet Firebase

- Pour configurer un compte, accéder à console.firebase.google.com/, puis cliquer sur « Ajouter un projet ».



- Saisir le nom du projet et cliquer sur « Créer un projet ».

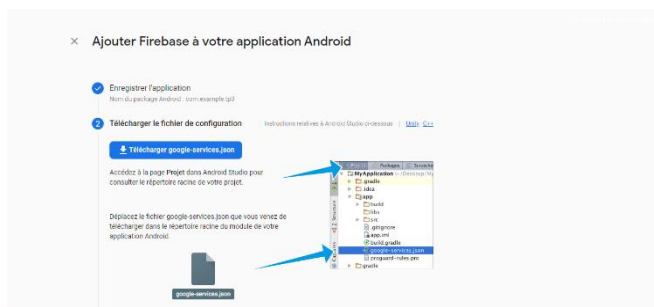


6. Une fois le projet est créé, le tableau de bord Firebase sera affiché permettant d'ajouter Firebase à une application iOS, Android ou bien Web. Dans cet atelier, nous allons voir comment utiliser Firebase sur un terminal Android : Cliquer sur « Ajouter une application » et sélectionner l'icône Android.



- Entrer le nom du package de l'application puis cliquer sur le bouton « Enregistrer l'application ». (Récupérer ce nom de package à partir du fichier « android/app/build.gradle » du module)

- Cliquer sur le bouton « Enregistrer l'application »
- Télécharger le fichier « google-services.json » puis le copier sous le dossier « android/app ». Cliquer ensuite sur « Suivant ».



- Dans votre projet Android, afficher la fenêtre du projet en mode « Project » et vérifier que le fichier « google-services.json » a été ajouté au projet.
7. Dans la console firebase, un code est alors généré :
- Dans le fichier « build.gradle.kts » du projet, ajouter dans dependencies:

Fichier build.gradle du projet

```
...
plugins {
    // ...

    // Add the dependency for the Google services Gradle plugin
    id("com.google.gms.google-services") version "4.4.0" apply false
}
...
```

- Dans le fichier « build.gradle.kts » du module app, ajouter :

Fichier build.gradle du module app

```
plugins {
    id("com.android.application")
    // Add the Google services Gradle plugin
    id("com.google.gms.google-services")
    ...
}

dependencies {
    ...
    implementation(platform("com.google.firebase:firebase-bom:32.6.0"))
}
```

- Cliquer sur « Suivant »
- Cliquer sur « Accéder à la console »

Etape 3 : Intégration de « firebase firestore »

8. Activation du service firestore :

- Dans le menu de gauche, cliquer sur l'icône « Tous les produits »
- Dans la console, cliquer sur « Cloud Firestore » puis sur « Commencer »



- Cliquer sur « Créer une base de données » ➔ « Suivant » ➔ « Démarrer en mode test » ➔ « Activer »
- Remarquer que dans l'onglet « Données » la base de données est vide



9. Dans le fichier « build.gradle.kts » du module app, ajouter :

Fichier build.gradle du module app

```
dependencies {
    ...
    implementation("com.google.firebase:firebase-firestore")
}
```

10. Coder la fonction ajouterLivres() permettant d'ajouter le livre saisi à la base de données. Pour cela, utiliser la fonction add() :

https://firebase.google.com/docs/firestore/manage-data/add-data?hl=fr#add_a_document

Fichier Ajout.java

```
protected void ajouterLivre() {
    Map<String, Object> data = new HashMap<>();
    data.put("titre", edTitre.getText().toString());
    data.put("nbpage", Integer.parseInt(edNbPage.getText().toString()));

    db.collection("livres")
        .add(data)
        .addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
            @Override
            public void onSuccess(DocumentReference documentReference) {
                edTitre.setText("");
                edNbPage.setText("");
                edTitre.requestFocus();
                Toast.makeText(getApplicationContext(), "Livre ajouté avec succès",
                    Toast.LENGTH_SHORT).show();
            }
        })
        .addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Toast.makeText(getApplicationContext(), "Erreur lors de l'ajout du
livre", Toast.LENGTH_SHORT).show();
            }
        });
}
```

11. Coder la fonction remplir() permettant de récupérer tous les livres de la base de données et de les ajouter au ListView. Pour cela, utiliser la fonction get() :

https://firebase.google.com/docs/firestore/query-data/get-data?hl=fr#get_multiple_documents_from_a_collection

Fichier Consultation.java

```
private void remplir() {
    db.collection("livres")
        .get()
        .addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
            @Override
            public void onComplete(@NonNull Task<QuerySnapshot> task) {
                if (task.isSuccessful()) {
                    for (QueryDocumentSnapshot document : task.getResult()) {
                        String id = document.getId();
                        String titre = document.getString("titre");
                        int nbpage = document.getDouble("nbpage").intValue();

                        Livre l = new Livre(id, titre, nbpage);
                        adpLivre.add(l);
                    }
                } else {
                    // Erreur lors de la récupération des livres
                    Toast.makeText(getApplicationContext(), "Erreur lors de la
récupération de la liste des livres", Toast.LENGTH_SHORT).show();
                }
            }
        });
}
```

12. Coder la fonction modifierLivre() permettant de modifier le livre sélectionné dans le Spinner par les nouvelles valeurs saisies dans les EditText dans la base de données. Pour cela, utiliser la fonction update() :

<https://firebase.google.com/docs/firestore/manage-data/add-data?hl=fr#update-data>

Fichier Modification.java

```
protected void modifierLivre() {
    Livre livre= (Livre) spLivre.getSelectedItem();

    Map<String, Object> map = new HashMap<>();
```

```

map.put("titre", edTitre.getText().toString());
map.put("nbpage", Integer.parseInt(edNbPage.getText().toString()));

db.collection("livres").document(livre.getId())
    .update(map)
    .addOnSuccessListener(new OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            // Le livre a été modifié avec succès
            remplir();
            Toast.makeText(getApplicationContext(), "Livre modifié avec succès",
Toast.LENGTH_SHORT).show();
        }
    })
    .addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            // Erreur lors de la modification du livre
            Toast.makeText(getApplicationContext(), "Erreur lors de la
modification du livre", Toast.LENGTH_SHORT).show();
        }
    });
}

```

13. Coder la fonction supprimerLivre() permettant de supprimer le livre livre sélectionné dans le Spinner de la base de données. Pour cela, utiliser la fonction delete() :

https://firebase.google.com/docs/firestore/manage-data/delete-data?hl=fr#delete_documents

Fichier Suppression.java

```

protected void supprimerLivre() {
    if(spLivre.getSelectedItemPosition() >= 0) {
        Livre livre = (Livre) spLivre.getSelectedItem();

        db.collection("livres").document(livre.getId())
            .delete()
            .addOnSuccessListener(new OnSuccessListener<Void>() {
                @Override
                public void onSuccess(Void aVoid) {
                    // Le livre a été supprimé avec succès
                    remplir();
                    Toast.makeText(getApplicationContext(), "Livre supprimé avec succès",
Toast.LENGTH_SHORT).show();
                }
            })
            .addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    // Erreur lors de la suppression du livre
                    Toast.makeText(getApplicationContext(), "Erreur lors de la suppression
du livre", Toast.LENGTH_SHORT).show();
                }
            });
    }
}

```