

Réalisation professionnelle 2 :

CCI-PROMO

Compétences mises en œuvre :

- Recenser et identifier les ressources numériques
- Exploiter des référentiels, normes et standards adoptés par le prestataire informatique
- Collecter, suivre et orienter des demandes
- Traiter des demandes concernant les applications
- Planifier les activités
- Réaliser les tests d'intégration et d'acceptation d'un service
- Déployer un service
- Accompagner les utilisateurs dans la mise en place d'un service

Contexte



J'ai travaillé en tant que technicien développeur junior pour l'ESN Badènia Tech qui vient de remporter le marché de développement d'une application mobile android qui doit participer à la promotion de la chambre de commerce et de l'industrie, particulièrement, les formations proposées. J'ai dû faire évoluer l'application existante qui est basique et comporte une seule IHM, la page d'accueil.

Recenser et identifier les ressources numériques

Pour concevoir au mieux l'application web nous avons utilisé de multiples ressources numériques. À savoir :



Android Studio

Android Studio est un environnement de développement pour développer des applications mobiles Android. Il est basé sur IntelliJ IDEA et utilise le moteur de production Gradle. Il est distribué sous Apache 2.0, qui est une licence de logiciel libre et open source.



Mamp

Le logiciel MAMP (abréviation de Macintosh, Apache, Mysql and PHP) permet de lancer un serveur web local sur une machine fonctionnant sur le système d'exploitation Mac OS X.



phpMyAdmin

phpMyAdmin est une application Web de gestion pour les systèmes de gestion de base de données MySQL et MariaDB, réalisée principalement en PHP et distribuée sous licence GNU GPL. C'est un outil qui nous permet de visualiser rapidement l'état de notre base de données et de la modifier, sans avoir à écrire de requêtes SQL.



GitLab

GitLab est un logiciel Open source et collaboratif, distribué sous licence MIT (licence de logiciel pour logiciels libres et open source). Il permet d'héberger les projets web, la gestion de versions des codes sources, permet une collaboration simple entre les collaborateurs sur un même projet. La version de base qui est déjà très complète est gratuite.



Git

Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2.



Eclipse

Eclipse est un environnement de développement intégré libre extensible, universel et polyvalent, permettant de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Il est distribué sous licence Eclipse qui est une licence libre à copyleft faible utilisée à l'origine par l'EDI Eclipse. L'Eclipse Public License est conçue pour être plus favorable aux entreprises souhaitant faire du logiciel propriétaire.



Virtual box

Oracle VM VirtualBox est un logiciel libre de virtualisation d'un ou plusieurs système(s) d'exploitation, publié par Oracle. C'est un outil libre et gratuit, distribué sous Licence publique générale GNU version 2 et licence propriétaire.



Ubuntu

Ubuntu est un système d'exploitation GNU/Linux basé sur Debian. Il est distribué majoritairement sous des licences libres (principalement GNU GPL) et quelques pilotes propriétaires.



FileZilla

FileZilla est un client FTP gratuit, rapide et sécurisé, développé sous la licence publique générale GNU. Il présente l'avantage de pouvoir se connecter à distance sur un serveur donné afin de télécharger des fichiers. Le gestionnaire de site inclus permet de gérer l'ensemble de vos adresses.



Postman

Postman permet de construire et d'exécuter des requêtes HTTP, de les stocker dans un historique afin de pouvoir les rejouer, mais surtout de les organiser en Collections. Cette classification permet notamment de regrouper des requêtes de façon « fonctionnelle ». Postman assure également la gestion des Environnements, qui permet de contextualiser des variables et d'exécuter des requêtes ou des séries de requêtes dans différentes configurations.



Java

La technologie Java définit à la fois un langage de programmation orienté objet et une plateforme informatique. Créée par l'entreprise Sun Microsystems (souvent juste appelée "Sun") en 1995, et reprise depuis par la société Oracle en 2009, la technologie Java est indissociable du domaine de l'informatique et du Web.



SQLite

SQLite est un SGBD (Système de Gestion de Base de Données) relationnel léger conçu spécifiquement pour le stockage local de données. Il ne nécessite pas l'installation d'un serveur de base de données. SQLite fonctionne dans différents environnements, notamment Linux, Windows, macOS, Android et iOS. Ce petit SGBD, créé en 2000 par D. Richard Hipp, est à code source ouvert. La version actuelle, SQLite 3.0, a été publiée en 2004.



DB Browser pour SQLite

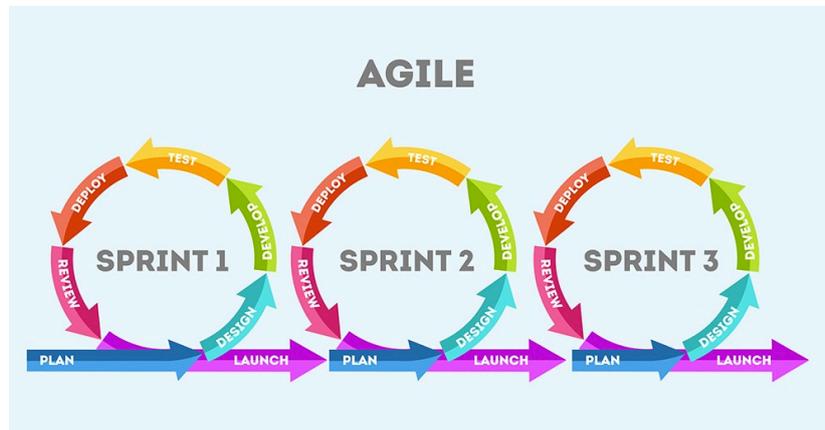
DB Browser pour SQLite est un outil visuel et open source de haute qualité pour créer, concevoir et éditer des fichiers de base de données compatibles avec SQLite.

Il est destiné aux utilisateurs et aux développeurs souhaitant créer des bases de données, rechercher et modifier des données.

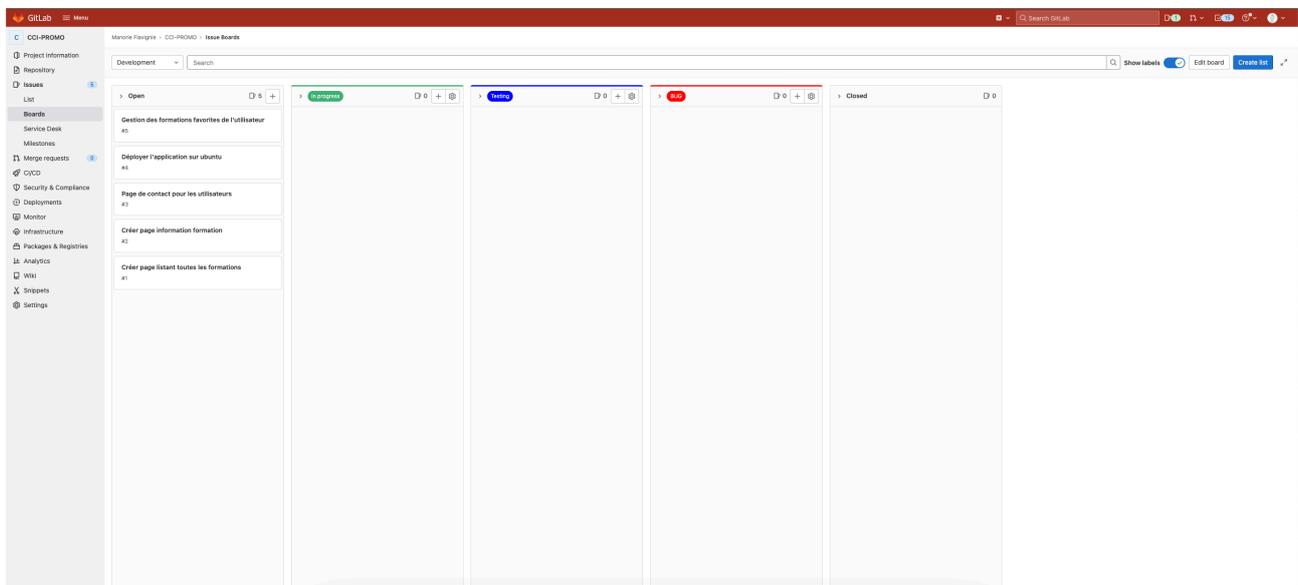
Explications :

Pour faire évoluer au mieux l'application mobile, sur mon poste de travail j'utilise Android Studio pour la réalisation visuelle en Java, je conçois des API en Java, sur Eclipse. Afin de voir tester en local mon code, je lance Mamp, ma base de données est stockée sur PhpMyAdmin. L'organisation des tâches et issues est réalisée sur Gitlab. Enfin je versionne mon code sur Gitlab à l'aide de git. Mes API sont déployées sur ma machine virtuelle Ubuntu que je manipule grâce au logiciel VirtualBox.

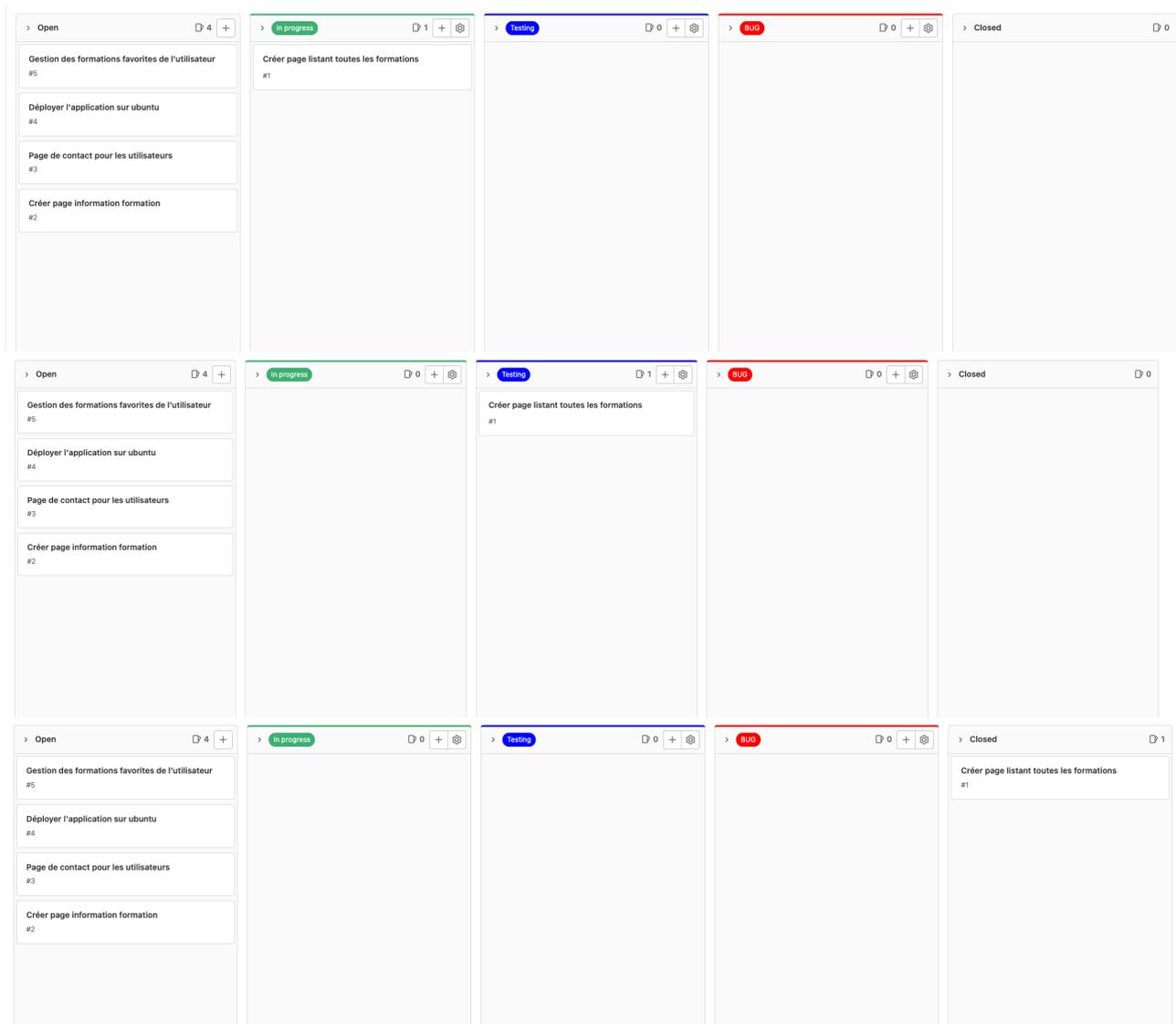
Exploiter des référentiels, normes et standards adoptés par le prestataire informatique
Collecter, suivre et orienter des demandes
Traiter des demandes concernant les applications
Planifier les activités



Pour ce projet j'ai utilisé la méthode agile. La méthode Agile recommande de se fixer des objectifs à court terme. Le projet est donc divisé en plusieurs sous-projets. Une fois l'objectif atteint, on passe au suivant, et ce jusqu'à l'accomplissement de l'objectif final. Cette approche est plus flexible. On entend par sprint des itérations de courtes durées décomposant un processus de développement souvent complexe afin de le rendre plus simple et plus facile à concevoir.



Sprint 1



```
package com.example.cci.vue;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.RecyclerView;
import androidx.recyclerview.widget.RecyclerView.ViewHolder;
import androidx.recyclerview.widget.RecyclerView.Adapter;
import androidx.recyclerview.widget.RecyclerView.ViewHolder;
import androidx.recyclerview.widget.RecyclerView.Adapter;
import androidx.recyclerview.widget.RecyclerView.ViewHolder;

public class FormationActivity extends RacineActivity implements AdapterView.OnItemClickListener {

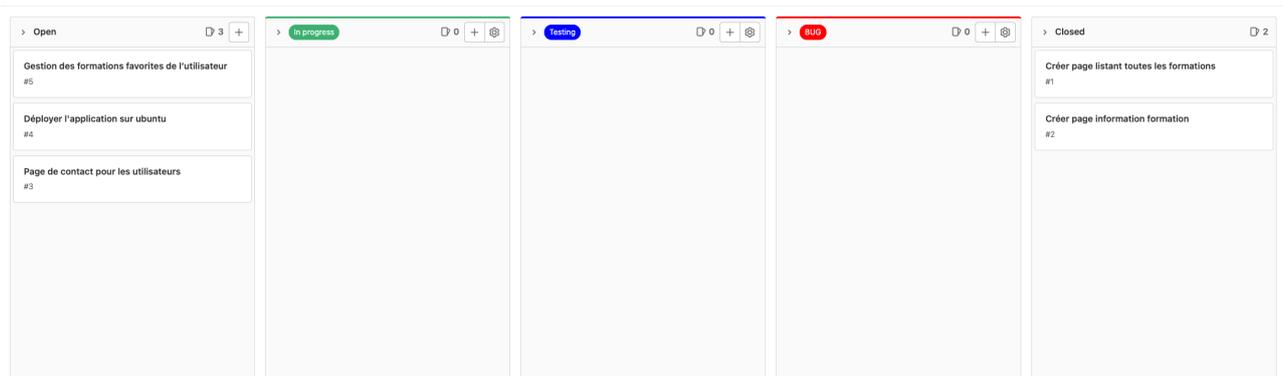
    private static String TAG = "FormationActivity";
    private FormationController formationController;
    private List<Formation> formations;
    private ListView liste;
    private FormationAdapter formationAdapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_formation);
        Log.i(TAG, "onCreate");
        formationController = FormationController.getInstance();
        creerListeFormations();
    }

    private void creerListeFormations(){
        formations = formationController.getFormations();
        Log.i(TAG, "creerListeFormations formations=" + formations);
        formationAdapter = new FormationAdapter(this, formations);
        liste = (ListView)findViewById(R.id.listView);
        liste.setAdapter(formationAdapter);
    }
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:baselineAligned="false"
    android:orientation="vertical" >
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Liste des formations"
        style="@style/titre_1" />
    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        />
</LinearLayout>
```

Grâce à git et à la commande « git checkout -b issue_1 » j'ai créé ma branche et j'ai ajouté mon code sur celle-ci, j'ai testé et versionné (commit & push sur mon répertoire distant GitLab (<https://gitlab.com/mnrflvgn/cci-promo>)) mon code enfin j'ai mergé ma branche avec la branche master. Le premier sprint est terminé.



```

        liste.setOnItemClickListener(this);
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
        Log.i(TAG, msg: "onItemClick = " + position);
        Log.i(TAG, msg: "formations = " + formations.get(position));
        formationController.setFormation(formations.get(position));
        Intent intent = new Intent( packageContext: FormationActivity.this, FormationSelectedActivity.class);
        startActivity(intent);
    }
}

public class FormationSelectedActivity extends RacineActivity {
    private static String TAG = "FormationSelectedActivity";
    private FormationController formationController;
    private Formation formation;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_formation);
        Log.i(TAG, msg: "onCreate");
        formationController = FormationController.getInstance();
        formationSelected();

        TextView title = findViewById(R.id.formationTitle);
        title.setText("Présentation de la formation : " + formation.getIntitle());

        TextView dateDebut = findViewById(R.id.formationDateDebut);
        dateDebut.setText("Débute le : " + convertStringDate(formation.getDateDebut()));

        TextView duree = findViewById(R.id.formationDuree);
        duree.setText("Durée : " + formation.getDureeMois() + " mois");

        ImageView imageView = findViewById(R.id.formationImage);
        loadImageView(imageView, url: urlImageBase + formation.getAdresseImage());

        imageView.setOnClickListener(new Button.OnClickListener() {...});

        private void formationSelected() {...}

        public void loadImageView (ImageView img, String url) {...}

        public void eventListenerLink(View view) {...}
    }
}

```

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="6sp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/formationTitle"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            style="@style/titre_2"/>

        <ImageView
            android:id="@+id/formationImage"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:adjustViewBounds="true"
            android:background="@color/white"
            android:scaleType="centerCrop" />

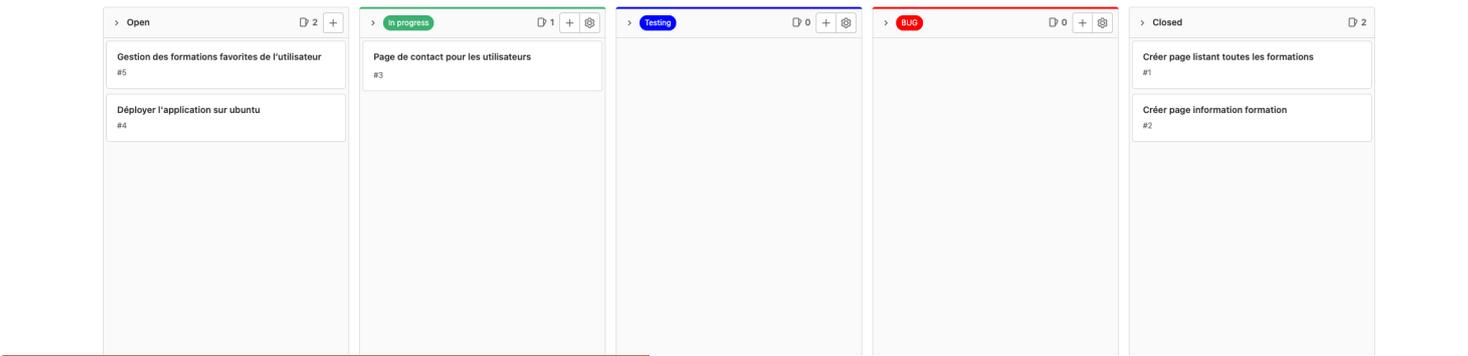
        <TextView
            android:id="@+id/formationDateDebut"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            style="@style/texteListe" />

    <TextView

```

Sur ma branche issue_2, j'ai ajouté mon code, j'ai testé et versionné mon code sur mon répertoire distant Gitlab, enfin j'ai merge ma branche avec la branche master. Le deuxième sprint est terminé.

Sprint 3



Manorie Flavignie > CCI-PROMO > Issues > #3

Open Created 29 minutes ago by **Manorie Flavignie** Owner 0 of 2 tasks completed

Page de contact pour les utilisateurs

Créer un formulaire permettant à l'utilisateur de renseigner son nom, son prénom, son mail et de saisir un message.
 Envoyer le message à la CCI via l'api rest exposée.

Edited just now by Manorie Flavignie

Drag your designs here or [click to upload](#).

Linked issues 0 0 +

Oldest first Show all activity Create merge request

- Manorie Flavignie @mnrflvgn changed the description 25 minutes ago
- Manorie Flavignie @mnrflvgn added **In progress** label 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Créer un formulaire permettant à l'utilisateur de renseigner son nom, son prénom, son mail et de saisir un message**, as completed 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Contrôler les données entrées**, as completed 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Contrôler les données entrées**, as incomplete 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Envoyer le message à la CCI via l'api rest exposée**, as completed 8 minutes ago
- Manorie Flavignie @mnrflvgn changed the description just now

Write a comment or drag your files here...

Markdown and quick actions are supported Attach a file

Comment Close issue

Manorie Flavignie > CCI-PROMO > Issues > #3

Open Created 29 minutes ago by **Manorie Flavignie** Owner 2 of 2 tasks completed

Page de contact pour les utilisateurs

Créer un formulaire permettant à l'utilisateur de renseigner son nom, son prénom, son mail et de saisir un message.
 Envoyer le message à la CCI via l'api rest exposée.

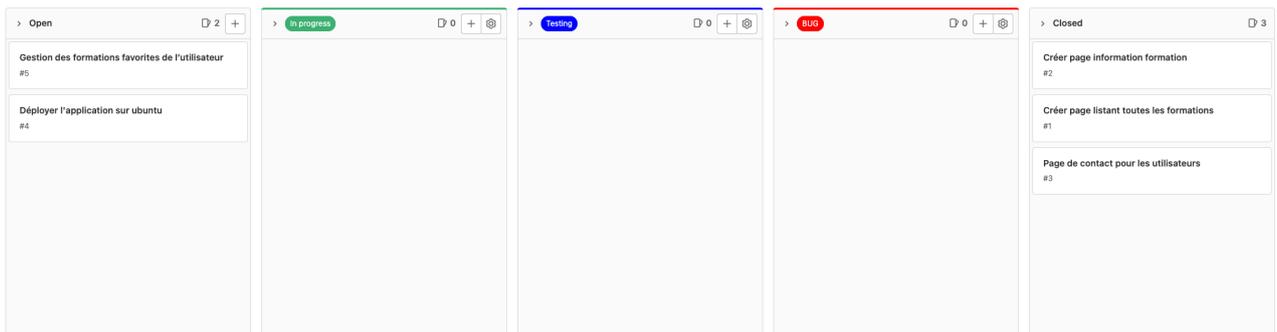
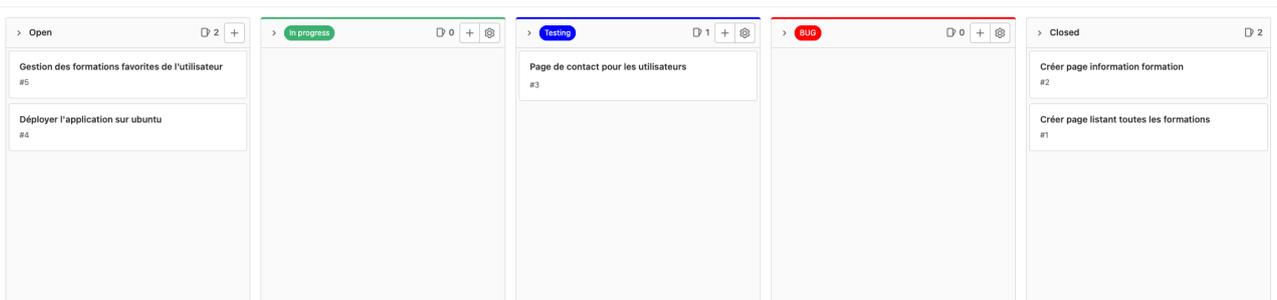
Edited just now by Manorie Flavignie

Drag your designs here or [click to upload](#).

Linked issues 0 0 +

Oldest first Show all activity Create merge request

- Manorie Flavignie @mnrflvgn changed the description 25 minutes ago
- Manorie Flavignie @mnrflvgn added **In progress** label 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Créer un formulaire permettant à l'utilisateur de renseigner son nom, son prénom, son mail et de saisir un message**, as completed 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Contrôler les données entrées**, as completed 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Contrôler les données entrées**, as incomplete 8 minutes ago
- Manorie Flavignie @mnrflvgn marked the task **Envoyer le message à la CCI via l'api rest exposée**, as completed 8 minutes ago
- Manorie Flavignie @mnrflvgn changed the description just now
- Manorie Flavignie @mnrflvgn marked the task **Créer un formulaire permettant à l'utilisateur de renseigner son nom, son prénom, son mail et de saisir un message**, as incomplete just now
- Manorie Flavignie @mnrflvgn marked the task **Envoyer le message à la CCI via l'api rest exposée**, as incomplete just now



```

package com.example.cci.vue;

import ...

public class ContactActivity extends RacineActivity implements View.OnClickListener, IResponseRestCallback{
    private static String TAG = "ContactActivity";
    private MessageController messageController;
    private ServiceRestItf serviceRest;
    private Button btnMsg = null;
    private EditText editNom;
    private EditText editPrenom;
    private EditText editMail;
    private EditText editMsg;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_contact);
        Log.i(TAG, msg: "onCreate");
        messageController = MessageController.getInstance();

        TextView title = findViewById(R.id.contactTitle);
        title.setText("Formulaire de contact");

        editNom = (EditText)findViewById(R.id.editNom);
        editPrenom = (EditText)findViewById(R.id.editPrenom);
        editMail = (EditText)findViewById(R.id.editMail);
        editMsg = (EditText)findViewById(R.id.editMsg);

        btnMsg = (Button) findViewById(R.id.btnMsg);
        btnMsg.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {
        if (v.getId() == R.id.btnMsg) {
            Log.i(TAG, msg: "onClick sur btnMsg");
        }
    }
}

```

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="6sp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/contactTitle"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            style="@style/titre_2"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="24sp"
            android:text="Nom : " />

        <EditText
            android:id="@+id/editNom"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            />

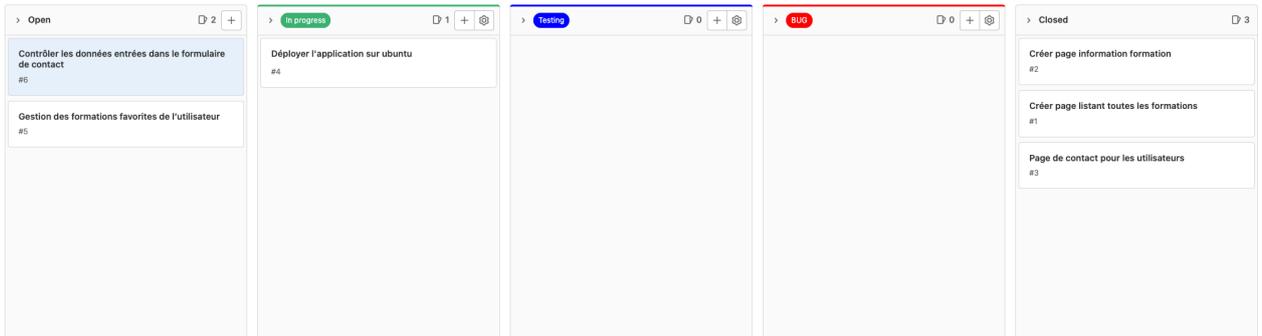
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="24sp"

```

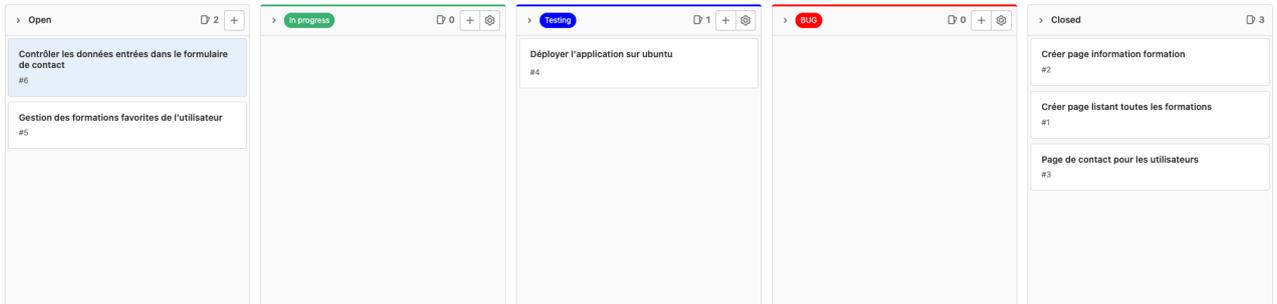
Sur ma branche issue_3, j'ai ajouté mon code, j'ai testé et versionné mon code sur mon répertoire distant Gitlab, enfin j'ai merge ma branche avec la branche master. Le troisième sprint est terminé.

Sprint 4

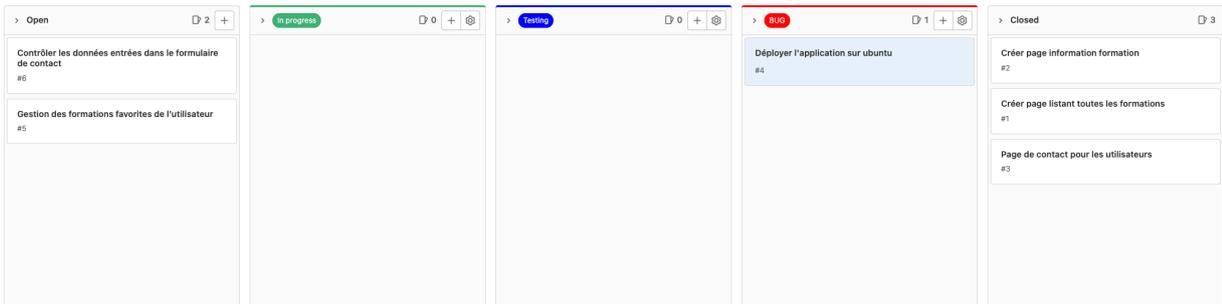
Déployer un service



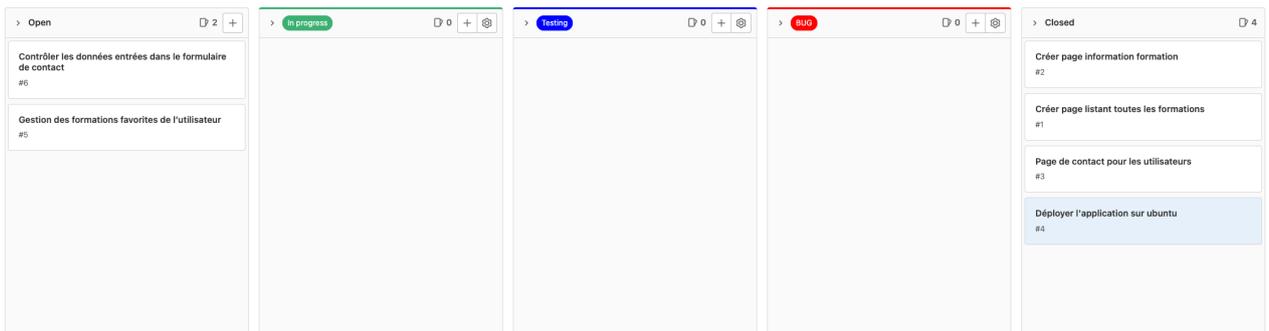
Snapshot 1 of a Kanban board with 5 columns: Open, In progress, Testing, BUG, and Closed. The 'Open' column contains two items: 'Contrôler les données entrées dans le formulaire de contact' (#6) and 'Gestion des formations favorites de l'utilisateur' (#5). The 'In progress' column contains one item: 'Déployer l'application sur ubuntu' (#4). The 'Closed' column contains three items: 'Créer page information formation' (#2), 'Créer page listant toutes les formations' (#1), and 'Page de contact pour les utilisateurs' (#3).



Snapshot 2 of the Kanban board. The 'Open' column remains the same. The 'In progress' column is empty. The 'Testing' column now contains the item 'Déployer l'application sur ubuntu' (#4). The 'Closed' column remains the same.



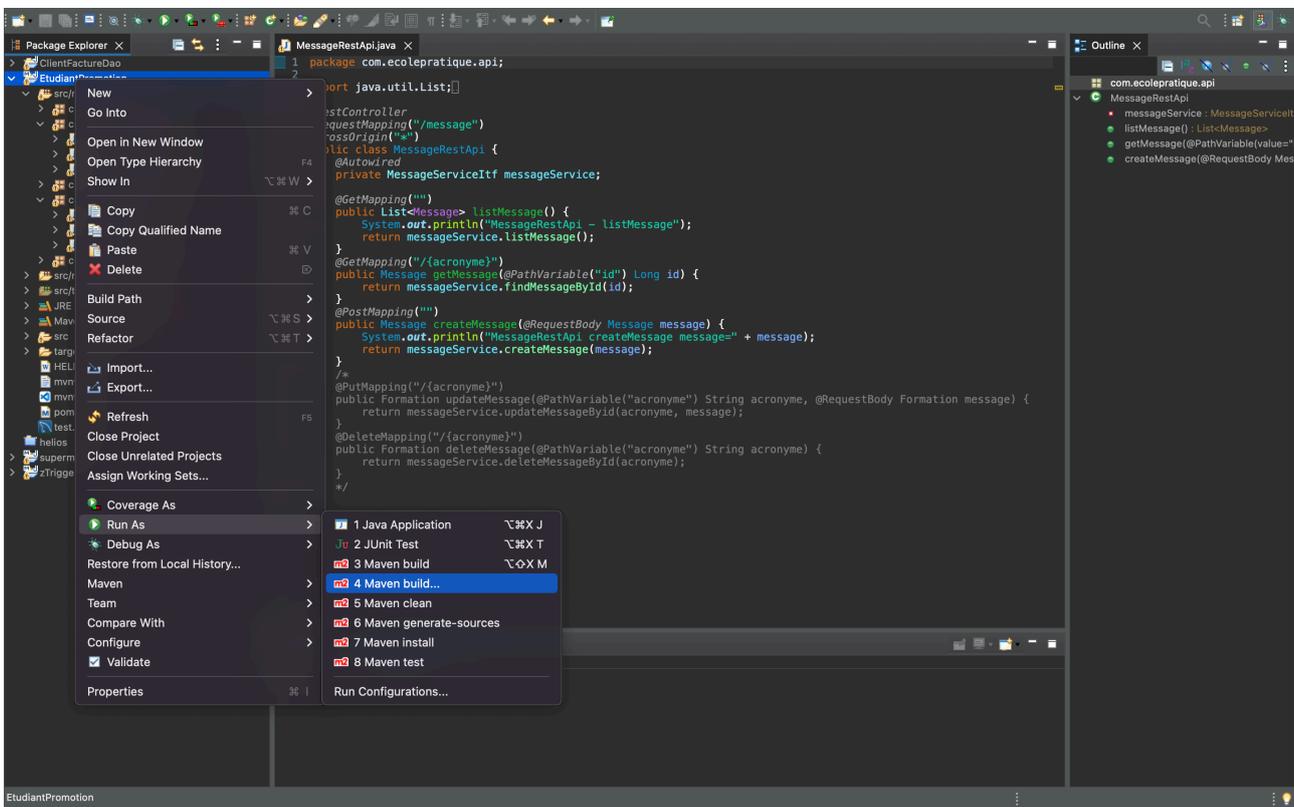
Snapshot 3 of the Kanban board. The 'Open' column remains the same. The 'In progress' and 'Testing' columns are empty. The 'BUG' column now contains the item 'Déployer l'application sur ubuntu' (#4). The 'Closed' column remains the same.



Snapshot 4 of the Kanban board. The 'Open' column remains the same. The 'In progress' and 'Testing' columns are empty. The 'BUG' column is empty. The 'Closed' column now contains four items: 'Créer page information formation' (#2), 'Créer page listant toutes les formations' (#1), 'Page de contact pour les utilisateurs' (#3), and 'Déployer l'application sur ubuntu' (#4).

J'ai réalisé les API de l'application sur Eclipse

```
application.properties X
1 server.port=8089
2 server.servlet.context-path=/ecole
3
4 spring.datasource.url = jdbc:mysql://localhost/ccipromotion?serverTimezone=Europe/Paris
5 spring.datasource.username = btssio
6 spring.datasource.password = btssio
7 spring.datasource.driverClassName = com.mysql.cj.jdbc.Driver
8 spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL5Dialect
9
10 #spring.h2.console.enabled=true
11 #spring.datasource.url=jdbc:h2:mem:testdb
12 #spring.datasource.driverClassName=org.h2.Driver
13 #spring.datasource.username=sa
14 #spring.datasource.password=
15
16 #spring.jpa.generate-ddl=true
17 spring.jpa.show-sql = true
18 spring.jpa.properties.hibernate.format_sql=true
19 # Hibernate ddl auto (create, create-drop, update)
20 spring.jpa.hibernate.ddl-auto = create-drop
21
```



Grâce au paquet openssh-server, que j'ai installé sur ma machine virtuelle ubuntu, à l'aide de la commande « sudo apt-get install openssh-server », j'ai pu transformer ma machine en serveur.

Connexion via SSH :

```

manorie@manorie:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.1.111 netmask 255.255.255.0 broadcast 192.168.1.255
inet6 2a01:e34:ec2e:35a0:146b:def8:fad0:3c5a prefixlen 64 scopeid 0x0<global>
inet6 2a01:e0a:131:8e20:6667:55e5:d286:392 prefixlen 64 scopeid 0x0<global>
inet6 fe80::d2b5:282b:b317:dbad prefixlen 64 scopeid 0x20<link>
inet6 2a01:e34:ec2e:35a0:ac6e:5c24:963c:ea19 prefixlen 64 scopeid 0x0<global>
inet6 2a01:e0a:131:8e20:9ff1:9991:9d71:2480 prefixlen 64 scopeid 0x0<global>
ether 08:00:27:a9:75:df txqueuelen 1000 (Ethernet)
RX packets 3045 bytes 1648210 (1.6 MB)
RX errors 0 dropped 28 overruns 0 frame 0
TX packets 1033 bytes 252189 (252.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Boucle locale)
RX packets 2078 bytes 429359 (429.3 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 2078 bytes 429359 (429.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

```

dev@mbp-server [18:08:42] [~]
-> % ssh manorie@192.168.1.111
manorie@192.168.1.111's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.11.0-41-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

43 mises à jour peuvent être appliquées immédiatement.
Pour afficher ces mises à jour supplémentaires, exécuter : apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Sat Apr  2 18:15:30 2022 from 192.168.1.26

```

FileZilla m'a permis de transférer mon script sql (cci-promotion2.sql) et mon application (cci-promo.jar) sur ma machine ubuntu.

The screenshot shows the FileZilla interface with the following details:

- Host:** sftp://192.168.1.111
- User:** manorie
- Status:** Connected to 192.168.1.111
- Local Site:** /Users/dev/Desktop/BTS SIO/A2/Conception et développement d'applications/Exa...

Nom de fichier	Taille de fichier	Type de fichier	Dernière modification
..			
favorino.png	2929	png-fichier	23.03.2022 08:...
favori.jpg	2721	jpg-fichier	23.03.2022 08:...
ccipromotion2.sql	16344	SQL File	15.03.2022 08:2...
cci5.jar	42939281	Java JAR file	17.03.2022 01:3...
cci-promo.jar	42945323	Java JAR file	31.03.2022 13:3...
Infos techniques.pdf	1311651	pdf-fichier	17.03.2022 01:17...
Dossier documentair...	417965	pdf-fichier	31.03.2022 11:3...
CCIPROMO consigne...	156916	pdf-fichier	17.03.2022 01:0...
._DS_Store	6148	Fichier	31.03.2022 13:5...
CCI-PROMO		Dossier	04.04.2022 10:2...
- Remote Site:** /home/manorie/cci

Nom de fichier	Taille de fichier	Type de fichier	Dernière modif	Droits d'accès	Propriétaire	Grc
..						
cci-promo.jar	42945323	Java JAR ...	31.03.2022 11:...	-rw-rw-r--	manorie	m...
ccipromotion2.sql	16344	SQL File	17.03.2022 11:...	-rw-rw-r--	manorie	m...

Ensuite afin de lancer mon application, je me positionne dans mon dossier cci contenant mon script et mon application.

```
manorie@manorie:~$ cd cci
manorie@manorie:~/cci$ ls
cci-promo.jar  ccipromotion2.sql
manorie@manorie:~/cci$
```

Puis, je lance mysql je crée mon user, ma base de données, enfin j'exécute le script sql importé.

```
CREATE DATABASE ccipromotion CHARACTER SET 'utf8';
CREATE USER btssio@'%' IDENTIFIED BY 'btssio';
GRANT ALL PRIVILEGES ON ccipromotion.* TO btssio@'%';
```

```
manorie@manorie:~/cci$ sudo mysql
[sudo] Mot de passe de manorie :
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 31
Server version: 8.0.28-0ubuntu0.20.04.3 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SOURCE ccipromotion2.sql
Query OK, 0 rows affected (0,00 sec)

Query OK, 0 rows affected (0,00 sec)

Query OK, 0 rows affected (0,01 sec)

Query OK, 0 rows affected (0,00 sec)

Query OK, 0 rows affected (0,00 sec)

Query OK, 0 rows affected (0,00 sec)
```

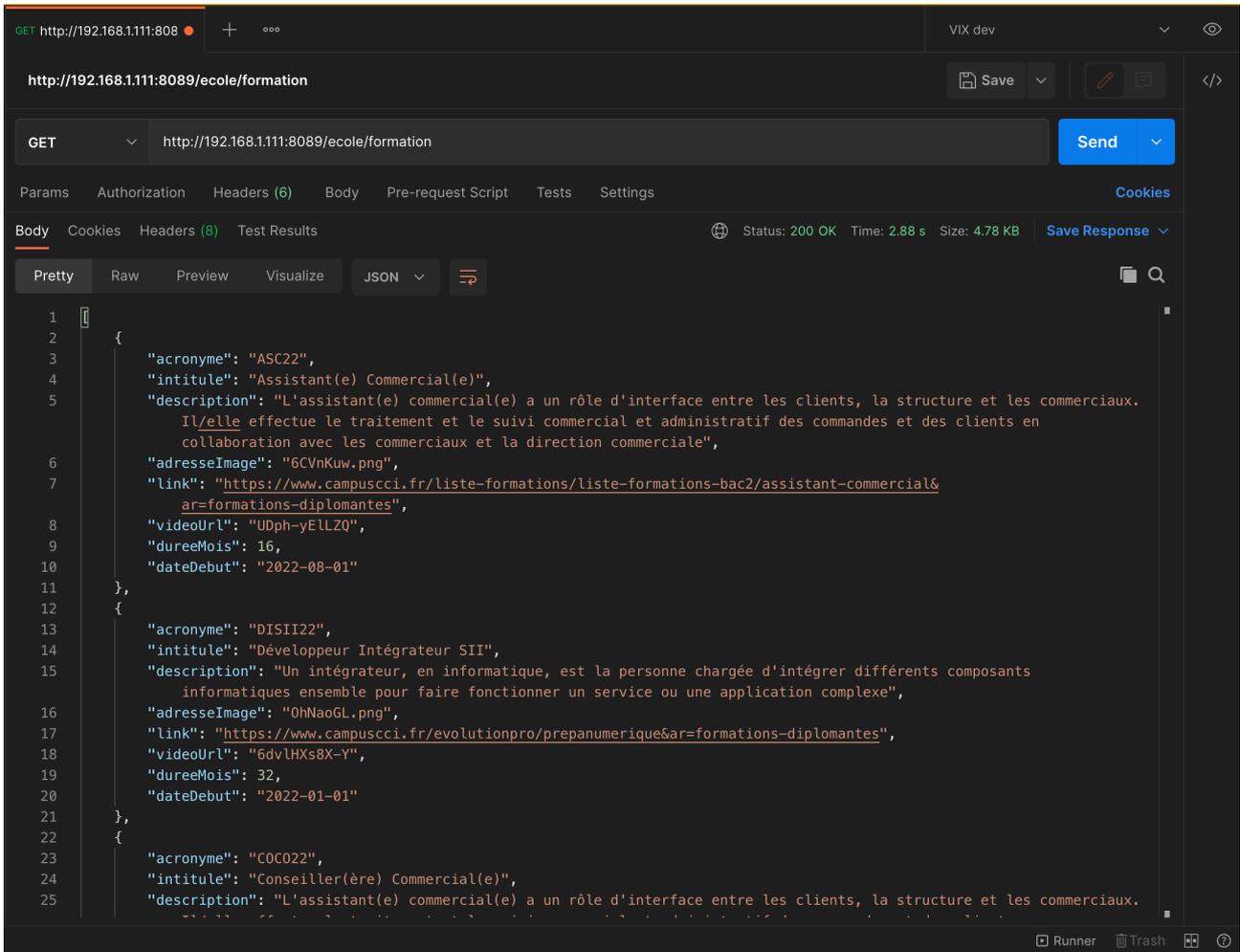
Enfin j'ouvre le port 8089 et je lance mon application.

```
manorie@manorie:~$ sudo ufw allow 8089
[sudo] Mot de passe de manorie :
manorie@manorie:~/cci$ java -jar ccipromotion.jar

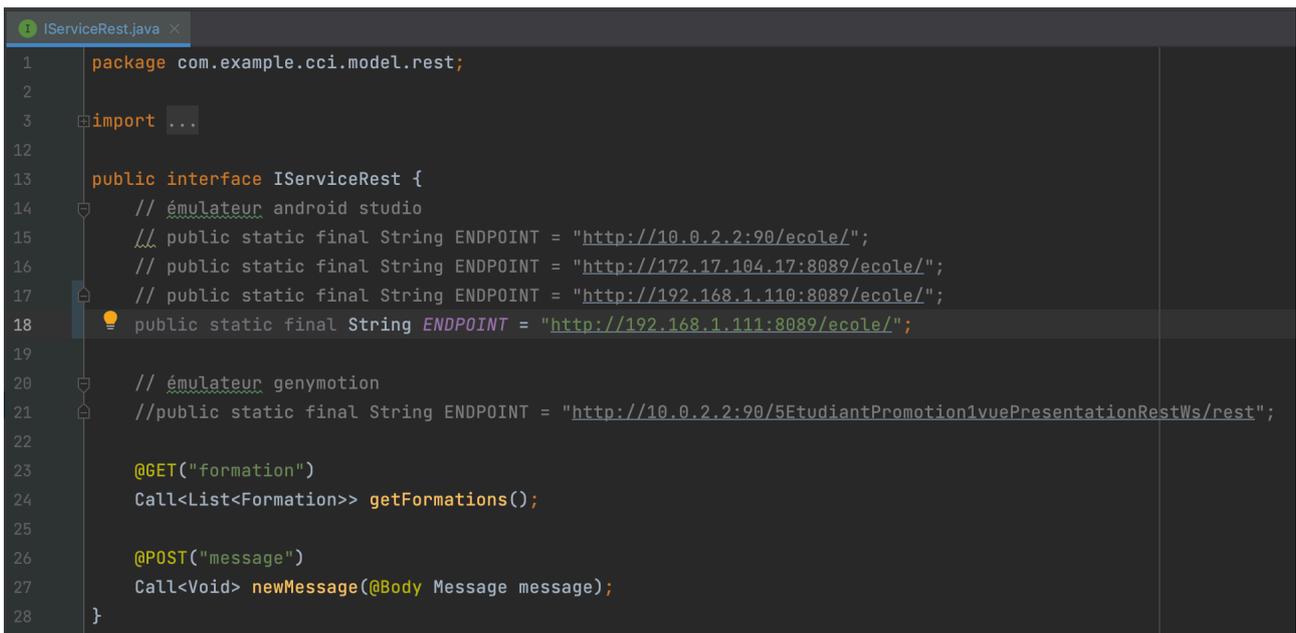
:: Spring Boot :: (v2.5.6)

2022-04-05 18:39:50.943 INFO 41104 --- [main] c.e.EtudiantPromotionApplication : Starting EtudiantPromotionApplication v0.0.1-SNAPSHOT using
ng Java 17.0.2 on manorie with PID 41104 (/home/manorie/cci/ccipromotion.jar started by manorie in /home/manorie/cci)
2022-04-05 18:39:50.964 INFO 41104 --- [main] c.e.EtudiantPromotionApplication : No active profile set, falling back to default profiles:
default
2022-04-05 18:39:54.242 INFO 41104 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mod
e.
2022-04-05 18:39:54.494 INFO 41104 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 226 ms. Found
3 JPA repository interfaces.
2022-04-05 18:39:56.912 INFO 41104 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8089 (http)
2022-04-05 18:39:56.940 INFO 41104 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-04-05 18:39:56.941 INFO 41104 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.54]
2022-04-05 18:39:57.193 INFO 41104 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/ecole] : Initializing Spring embedded WebApplicationContext
2022-04-05 18:39:57.194 INFO 41104 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 6
008 ms
2022-04-05 18:39:58.671 INFO 41104 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2022-04-05 18:39:58.969 INFO 41104 --- [main] org.hibernate.Version : HHH000412: Hibernate ORM core version 5.4.32.Final
2022-04-05 18:39:59.343 INFO 41104 --- [main] org.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
2022-04-05 18:39:59.671 INFO 41104 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2022-04-05 18:40:01.086 INFO 41104 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
```

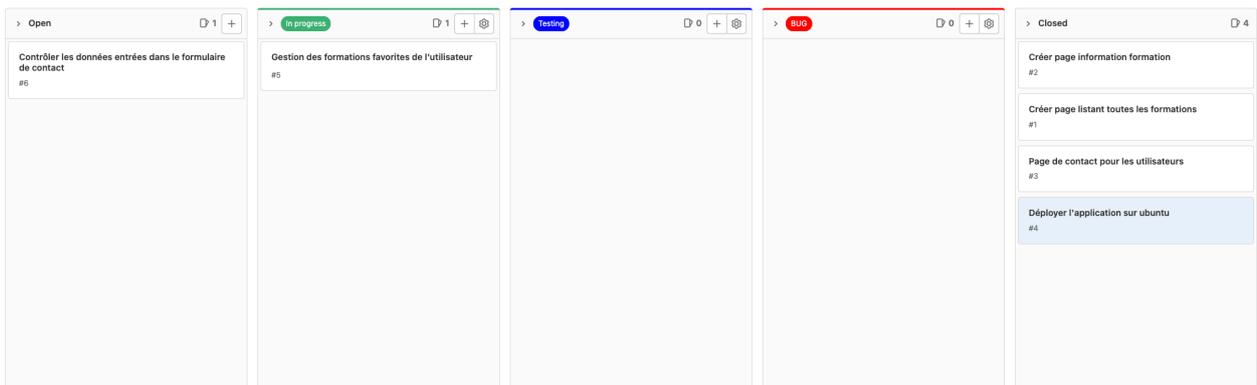
Pour finir, il est nécessaire de tester la connexion établie grâce à Postman, il suffit de rentrer la route souhaitée.



Afin de configurer l'application android, il faut lui ajouter un nouvel ENDPOINT.



Sprint 5



Manorie Flavignie > CCI-PROMO > Issues > #5

Closed Created 1 week ago by **Manorie Flavignie** Owner 0 of 2 tasks completed

Reopen issue

Gestion des formations favorites de l'utilisateur

- Ajouter la fonctionnalité : ajouter/supprimer la formation courante de la liste des formations
- Ajouter une entrée dans le menu, « favoris » permet d'afficher la liste des formations favorites et d'accéder au détail d'une formation, comme précédemment

Edited 1 week ago by Manorie Flavignie

Drag your designs here or [click to upload](#).

Manorie Flavignie > CCI-PROMO > Issues > #5

Closed Created 1 week ago by **Manorie Flavignie** Owner 2 of 2 tasks completed

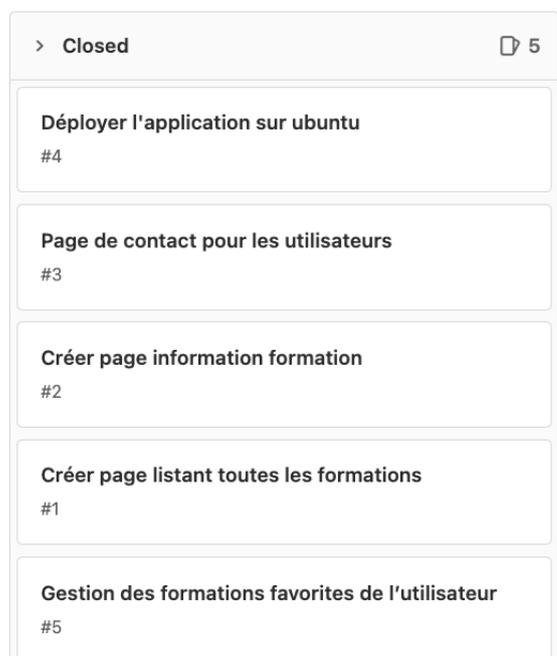
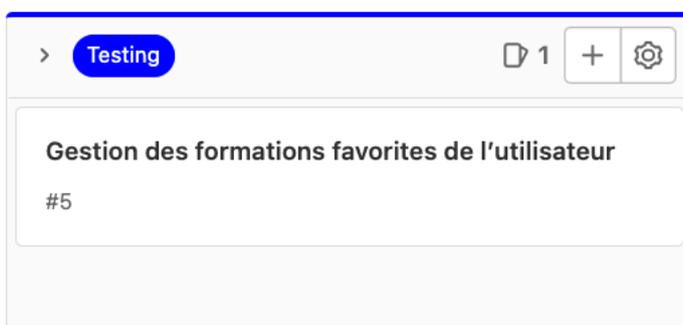
Reopen issue

Gestion des formations favorites de l'utilisateur

- Ajouter la fonctionnalité : ajouter/supprimer la formation courante de la liste des formations
- Ajouter une entrée dans le menu, « favoris » permet d'afficher la liste des formations favorites et d'accéder au détail d'une formation, comme précédemment

Edited just now by Manorie Flavignie

Drag your designs here or [click to upload](#).



Sur ma branche issue_5, j'ai ajouté mon code, j'ai testé et versionné mon code sur mon répertoire distant Gitlab, enfin j'ai merge ma branche avec la branche master. Le cinquième sprint est terminé.

```

package com.example.cci.SQLite;

import ...

public class DatabaseOpenHelper extends SQLiteOpenHelper {

    public final static String TABLE_NAME = "favorite";
    public final static String FORMATION_ACR = "acronyme";
    public final static String IS_LIKED = "isLiked";
    public final static String[] columns = { FORMATION_ACR, IS_LIKED };

    final private static String CREATE_CMD =

        "CREATE TABLE favorite (" + FORMATION_ACR
        + " VARCHAR(8) PRIMARY KEY, "
        + IS_LIKED + " VARCHAR(8) NOT NULL)";

    final private static String NAME = "favorite_db";
    final private static Integer VERSION = 1;
    final private Context mContext;

    public DatabaseOpenHelper(Context context) {
        super(context, NAME, factory: null, VERSION);
        this.mContext = context;
    }

    @Override
    public void onCreate(SQLiteDatabase db) { db.execSQL(CREATE_CMD); }
}

```

```

if (cursor != null) {
    if (cursor.moveToFirst()) {
        do {
            try {
                int n = cursor.getColumnIndex(mDbHelper.FORMATION_ACR);
                acronyme = cursor.getString(n);

                n = cursor.getColumnIndex(mDbHelper.IS_LIKED);
                is_liked = cursor.getString(n);

                if (is_liked.equals("true")){
                    i = findViewById(R.id.heart);
                    i.setImageResource(R.drawable.coeurplein);
                }
            } catch (SQLException e) {
                Log.i(TAG, msg: "Erreur lecture formation");
            }
        } while (cursor.moveToNext());
    }
    cursor.close();
}
}

```

```

public void onClick(View v) {
    Log.i(TAG, msg: "Bouton heart");

    if (is_liked.equals("true")){
        ImageView i = findViewById(R.id.heart);
        i.setImageResource(R.drawable.coeurvide);

        ContentValues values = new ContentValues();
        values.put(DatabaseOpenHelper.IS_LIKED, "false");

        mDbHelper.getWritableDatabase().update(DatabaseOpenHelper.TABLE_NAME, values,
            whereClause: DatabaseOpenHelper.FORMATION_ACR + "=?",
            new String[] { formation.getAcronyme() });

        is_liked = "false";
    }else {
        ImageView i = findViewById(R.id.heart);
        i.setImageResource(R.drawable.coeurplein);

        ContentValues values = new ContentValues();
        values.put(DatabaseOpenHelper.IS_LIKED, "true");

        mDbHelper.getWritableDatabase().update(DatabaseOpenHelper.TABLE_NAME, values,
            whereClause: DatabaseOpenHelper.FORMATION_ACR + "=?",
            new String[] { formation.getAcronyme() });

        is_liked = "true";
    }
}
}

```

```

com.example.cci.Entity.Formation
public String getAcronyme()
CCI-PROMO.app

```

```

@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
    Log.i(TAG, msg: "onItemClick = " + position);

    serviceRest.getFormation( objetReponse: this, formations.get(position));
    Log.i(TAG, msg: "serviceRest.getFormation = " + formations.get(position));

    formationsList = formationController.getFormations();
    Log.i(TAG, msg: "formationController.getFormation = " + formationsList);

    for (Formation f : formationsList) {
        if (f.getAcronyme().equals(formations.get(position))){
            formationController.setFormation(f);
        }
    }

    Intent intent = new Intent( packageContext: FavoriteActivity.this, FormationSelectedActivity.class);
    startActivity(intent);
}
}

```

Sprint 6

> **In progress** 1 + ⚙

Contrôler les données entrées dans le formulaire de contact
#6

> **Testing** 1 + ⚙

Contrôler les données entrées dans le formulaire de contact
#6

> **Closed** 6

Contrôler les données entrées dans le formulaire de contact
#6

Gestion des formations favorites de l'utilisateur
#5

Déployer l'application sur ubuntu
#4

Page de contact pour les utilisateurs
#3

Créer page information formation
#2

Créer page listant toutes les formations
#1

Sur ma branche issue_6, j'ai ajouté mon code, j'ai testé et versionné mon code sur mon répertoire distant Gitlab, enfin j'ai merge ma branche avec la branche master. Le troisième sprint est terminé.

```

package com.example.cci.util;

import ...

public class InputValidatorHelper {
    public boolean isValidEmail(String string){
        final String EMAIL_PATTERN = "^[_A-Za-z0-9-\\+]+(\\.[_A-Za-z0-9-]+)*@[A-Za-z0-9-]+(\\.[A-Za-";
        Pattern pattern = Pattern.compile(EMAIL_PATTERN);
        Matcher matcher = pattern.matcher(string);
        return matcher.matches();
    }

    public boolean isOnlyCharater(String string){
        final String LETTER_PATTERN = "[a-zA-Zéèçàêêî]+";
        Pattern pattern = Pattern.compile(LETTER_PATTERN);
        Matcher matcher = pattern.matcher(string);
        return matcher.matches();
    }

    public boolean isNullOrEmpty(String string){
        if(string == null) return true;
        else if(string.isEmpty()) return true;
        else return false;
    }

    public boolean isNumeric(String string){
        final String LETTER_PATTERN = "[0-9.]+";
        Pattern pattern = Pattern.compile(LETTER_PATTERN);
        Matcher matcher = pattern.matcher(string);
        return matcher.matches();
    }
}

```

```

package com.example.cci.util;

import junit.framework.TestCase;

public class InputValidatorHelperTest extends TestCase {
    private static String TAG = "InputValidatorHelperTest";
    private InputValidatorHelper inputValidatorHelper;

    public InputValidatorHelperTest() { inputValidatorHelper = new InputValidatorHelper(); }

    public void testIsValidEmail1() {
        String mail = "3azerty@test.a";
        assertEquals( expected: false, inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail2() {
        String mail = "2azertArotest.com";
        assertFalse(inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail3() {
        String mail = "nom@prenom.fr";
        assertTrue(inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail4() {
        String mail = "nom@pre!nom.fr";
        assertFalse(inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail5() {
        String mail = "no#m@prenom.fr";
        assertFalse(inputValidatorHelper.isValidEmail(mail));
    }
}

```

```

if (editNom.getText().toString().isEmpty() || editPrenom.getText().toString().isEmpty()
    || editMail.getText().toString().isEmpty()
    || editMsg.getText().toString().isEmpty()){

    LayoutInflater inflater = this.getLayoutInflater();
    View layout = inflater.inflate(R.layout.activity_contact, root: null);

    TextView textView = (TextView)findViewById(R.id.erreur);
    textView.setText("Attention vous devez remplir tous les champs !");
    textView.setTextColor(Color.RED);
    // Create Toast
    Toast toast = new Toast(context: this);
    toast.setDuration(Toast.LENGTH_LONG);
    toast.setGravity(gravity: Gravity.TOP|Gravity.LEFT, xOffset: 0, yOffset: 0);
    toast.setView(layout);
    toast.show();

}else if(!inputValidatorHelper.isOnlyCharater(prenom)) {
    Log.i(TAG, msg: "Toast - Prénom invalide");
    Toast.makeText(
        context: ContactActivity.this, text: "Prénom invalide !", Toast.LENGTH_SHORT
    ).show();

}else if(!inputValidatorHelper.isOnlyCharater(nom)) {
    Log.i(TAG, msg: "Toast - Nom invalide");
    Toast.makeText(
        context: ContactActivity.this, text: "Nom invalide !", Toast.LENGTH_SHORT
    ).show();

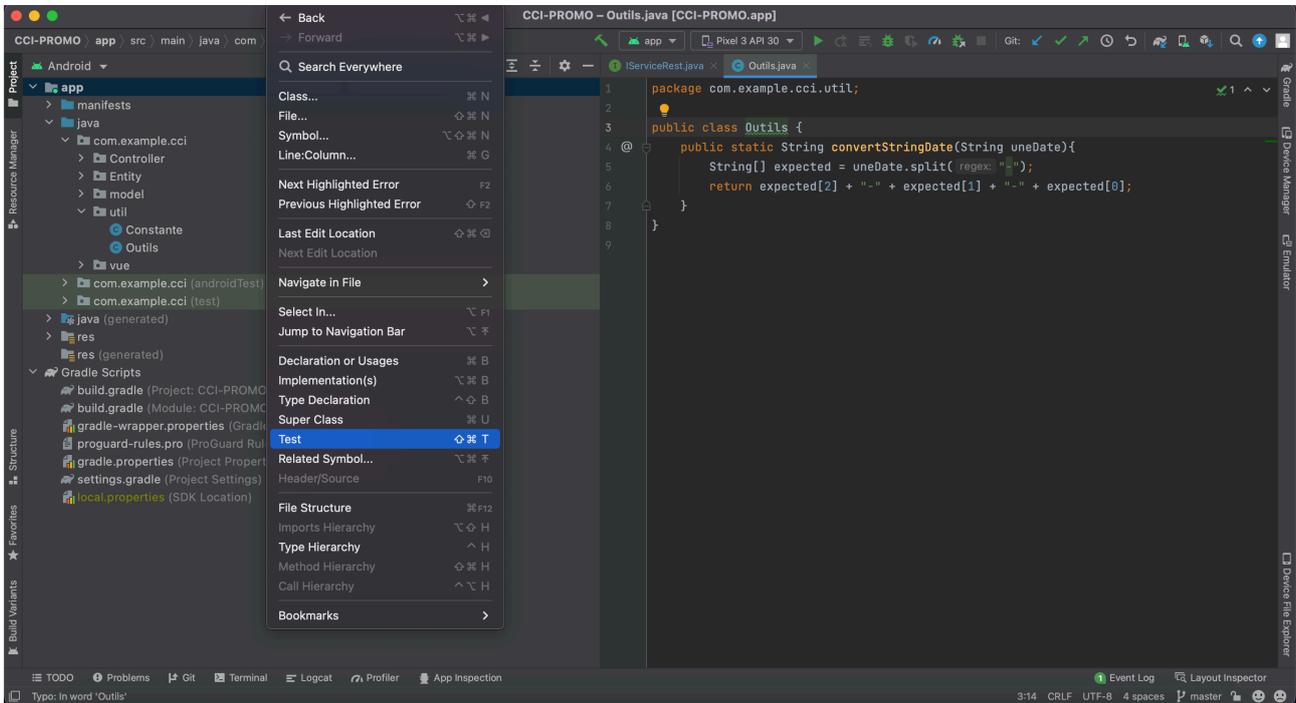
}else if (!inputValidatorHelper.isValidEmail(mail)){

```

Réaliser les tests d'intégration et d'acceptation d'un service

Grâce à Junit, je fais des tests unitaires afin de tester mes fonctions.

Fonction de conversion de dates :



Fonctions de validations de données entrantes :

```
package com.example.cci.util;

import junit.framework.TestCase;

public class InputValidatorHelperTest extends TestCase {
    private static String TAG = "InputValidatorHelperTest";
    private InputValidatorHelper inputValidatorHelper;

    public InputValidatorHelperTest() { inputValidatorHelper = new InputValidatorHelper(); }

    public void testIsValidEmail1() {
        String mail = "3azerty@test.a";
        assertEquals("expected: false, inputValidatorHelper.isValidEmail(mail));", false, inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail2() {
        String mail = "2azertArotest.com";
        assertFalse(inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail3() {
        String mail = "nom@prenom.fr";
        assertTrue(inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail4() {
        String mail = "nom@pre!nom.fr";
        assertFalse(inputValidatorHelper.isValidEmail(mail));
    }

    public void testIsValidEmail5() {
        String mail = "no#m@prenom.fr";
        assertFalse(inputValidatorHelper.isValidEmail(mail));
    }
}
```

De plus, je réalise des tests fonctionnels.