

**Institut Universitaire de Technologie,
Aix-Marseille Université**

ANNEXES

RAPPORT DE STAGE de fin de deuxième année

Bachelor Universitaire de Technologie

Spécialité Réseaux et Télécommunications

Parcours cybersécurité

Développeur d'applications web et
Technicien de maintenance réseau

Xitong LI

IMMERSIZE

Responsable entreprise : Lorenzo Lavaux

Responsable académique : Jean-Luc DAMOISEAUX

2023

Table des matières

Introduction	5
Le travail de la première partie (IMMSERSIZE)	5
Missions du poste occupé	6
Firebase	6
Installation	7
Angular	11
Installation	11
Mettre en ligne	12
Présentation	13
Responsable du débogage	17
Le travail de la deuxième partie (IUT à MARSEILLE LUMINY)	23
Installation du matériel et agencement de la pièce	23
Mise à jour ou réinstallation de la configuration des équipements réseau	27
Conclusion	35
Remerciements	37
Illustration	39
Glossaire	43
Bibliographie	45

1. Introduction

Durant mon stage, j'ai eu l'opportunité de travailler chez Immersize, une entreprise spécialisée dans les jeux d'espace game. En plus de cela, nous avons également collaboré avec des projets gouvernementaux visant à créer des jeux éducatifs divertissants tels que les jeux d'évasion, dans le but d'inviter des personnes au chômage à retrouver confiance en elles et à trouver la motivation pour retourner au travail.

Mon rôle principal était de collaborer sur le développement collaboratif d'une application web en utilisant les langages HTML, CSS et JavaScript, ainsi que les technologies Angular et Firebase. J'ai contribué à l'amélioration, et à l'optimisation.

Au cours des deux dernières semaines, je suis retourné à l'IUT (Institut Universitaire de Technologie) pour effectuer mes dernières tâches. J'ai collaboré avec M. Ali pour aménager une nouvelle salle de classe pour la salle des serveurs. De plus, j'ai pris en charge la maintenance et la mise à niveau des nouveaux commutateurs et routeurs.

Je vais vous présenter les logiciels Angular et Firebase, en expliquant leur utilisation et mes réalisations avec eux. Ensuite, je parlerai de comment j'ai amélioré la visibilité du site web, après laissez-moi vous présenter les travaux de maintenance associés à l'IUT, Enfin, je conclurai en résumant mon expérience et les compétences ce que j'ai appris.

2. Le travail de la première phase (IMMSERSIZE)

2.1 Missions du poste occupé

Dans le cadre de ma responsabilité professionnelle, mon objectif principal est de créer une application web interactive à l'aide d'Angular. Je vais utiliser CSS, HTML et JavaScript pour le développement, et j'aurai besoin d'utiliser Firebase pour stocker les informations des utilisateurs ou des joueurs dans la base de données. Je vais également utiliser le service d'hébergement (Hosting) pour déployer le site web, et le service d'authentification (Authentication)

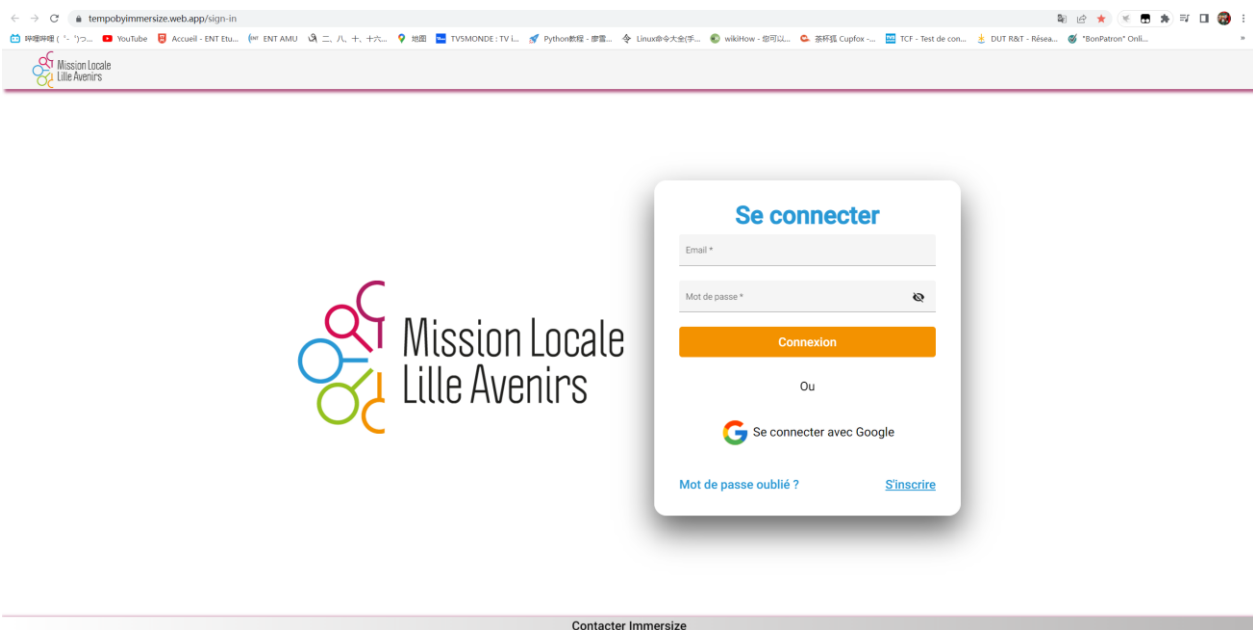


Figure 1 interface de connexion

pour vérifier les informations des utilisateurs, et ainsi de suite. Cela permettra de créer une véritable application web accessible via Internet.

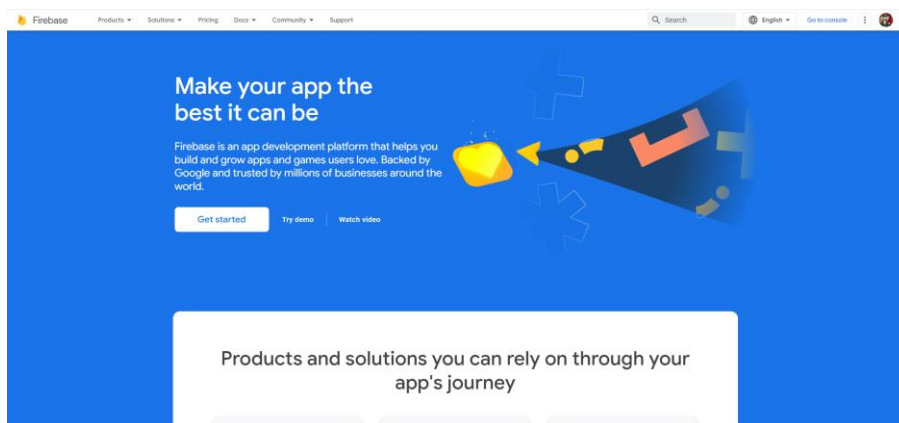


Figure 2 Siteweb Firebase



2.1.1 Firebase

Firebase est une plateforme complète de développement d'applications mobiles et web proposée par Google. Elle offre une gamme d'outils et de services visant à simplifier les tâches et fonctionnalités nécessaires à la création d'applications de haute qualité.

Firebase propose les fonctionnalités et caractéristiques principales suivantes :

1. **Base de données en temps réel (Realtime Database) :** La base de données en temps réel est une base de données cloud synchronisés en temps réel, permettant de stocker et de synchroniser les données de l'application en temps réel. Les développeurs peuvent facilement utiliser la base de données en temps réel pour réaliser des mises à jour et partager des données en temps réel dans leur application.
2. **Authentification (Authentication) :** Firebase offre un service d'authentification prenant en charge plusieurs méthodes d'authentification, telles que l'authentification par e-mail/mot de passe, les connexions avec des réseaux sociaux et les fournisseurs d'identité. Cela permet aux développeurs d'intégrer facilement des fonctionnalités d'authentification utilisateur et de gérer les comptes d'utilisateurs et les autorisations d'accès.
3. **Stockage Cloud (Cloud Storage) :** Le stockage cloud de Firebase permet de stocker et de servir des fichiers tels que des images, des vidéos et d'autres ressources statiques dans le cloud. Il fournit une solution simple et évolutive pour gérer les besoins de stockage de fichiers des applications.
4. **Messagerie Cloud (Cloud Messaging) :** Le service de messagerie cloud de Firebase permet aux développeurs d'envoyer des notifications push ciblées à leurs utilisateurs sur différentes plates-formes, y compris Android, iOS et le Web.
5. **Analytique (Analytics) :** Firebase propose des outils d'analyse avancés pour suivre et mesurer les performances de l'application, le comportement des utilisateurs et les interactions au sein de l'application. Cela aide les développeurs à prendre des décisions éclairées pour améliorer leur application.

2.1.1.1 Installation

Pendant mon travail, j'aurai besoin des services d'Authentification, de Stockage Cloud et de Messagerie Cloud (Hébergement) de Firebase. Voici comment procéder à leur installation :
Étape 1 : Configuration de l'environnement de développement Angular.

- Installez Node.js : Installez Node.js sur votre ordinateur, cela vous fournira npm (le gestionnaire de paquets Node).

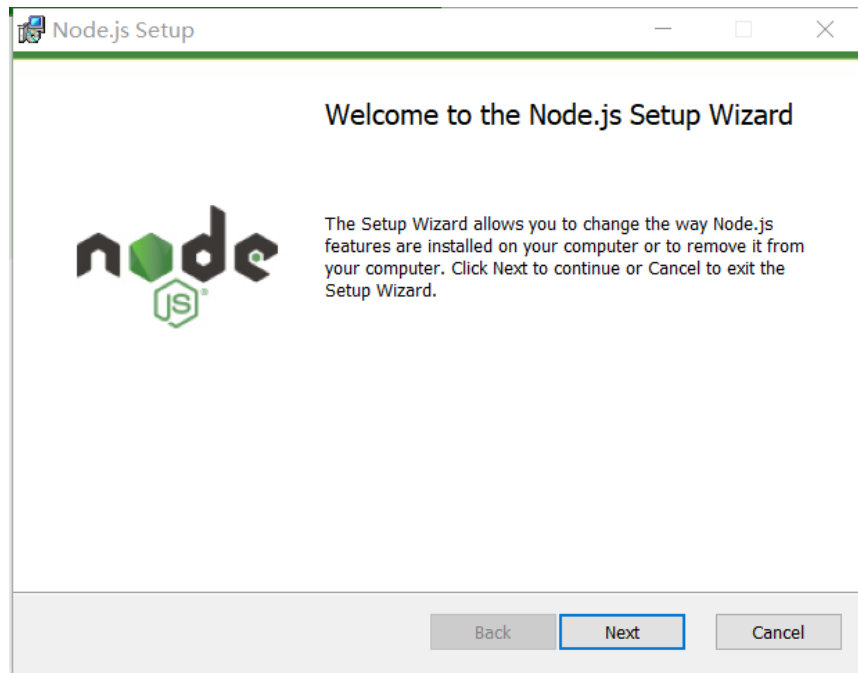


Figure 3 Interface d'installation

Node.js est un environnement d'exécution JavaScript basé sur le moteur Chrome V8, utilisé pour exécuter du code JavaScript côté serveur. Il possède des fonctionnalités non bloquantes et basées sur des événements, ce qui le rend idéal pour créer des applications web performantes et évolutives. Node.js est largement utilisé dans des domaines tels que le développement web, les applications côté serveur, les outils en ligne de commande, l'Internet des objets et le traitement des données massives. Sa simplicité, son efficacité et son écosystème riche en font un choix privilégié des développeurs.

- Installez Angular CLI : Exécutez la commande suivante dans votre terminal pour installer Angular CLI :

```
C:\Users\13241\Desktop\备份\Tempo-main - 副本>npm install -g @angular/cli
added 1 package, removed 21 packages, and changed 238 packages in 19s

36 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New minor version of npm available! 9.5.1 -> 9.7.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.7.1
npm notice Run npm install -g npm@9.7.1 to update!
npm notice
C:\Users\13241\Desktop\备份\Tempo-main - 副本>_
```

Figure 4 Firebase Interface d'installation

Étape 2 : Création d'un nouveau projet Angular

Dans votre terminal, accédez au répertoire dans lequel vous souhaitez créer votre projet, puis exécutez la commande suivante :

```
C:\Users\13241\Desktop\备份\Tempo-main - 副本>ng new Tempo
```

Figure 5 Créer un nouveau projet

Très bien, vous avez maintenant réussi à créer localement le dossier que vous souhaitez héberger sur le serveur Firebase. Nous allons procéder à la configuration du serveur Firebase en ligne.

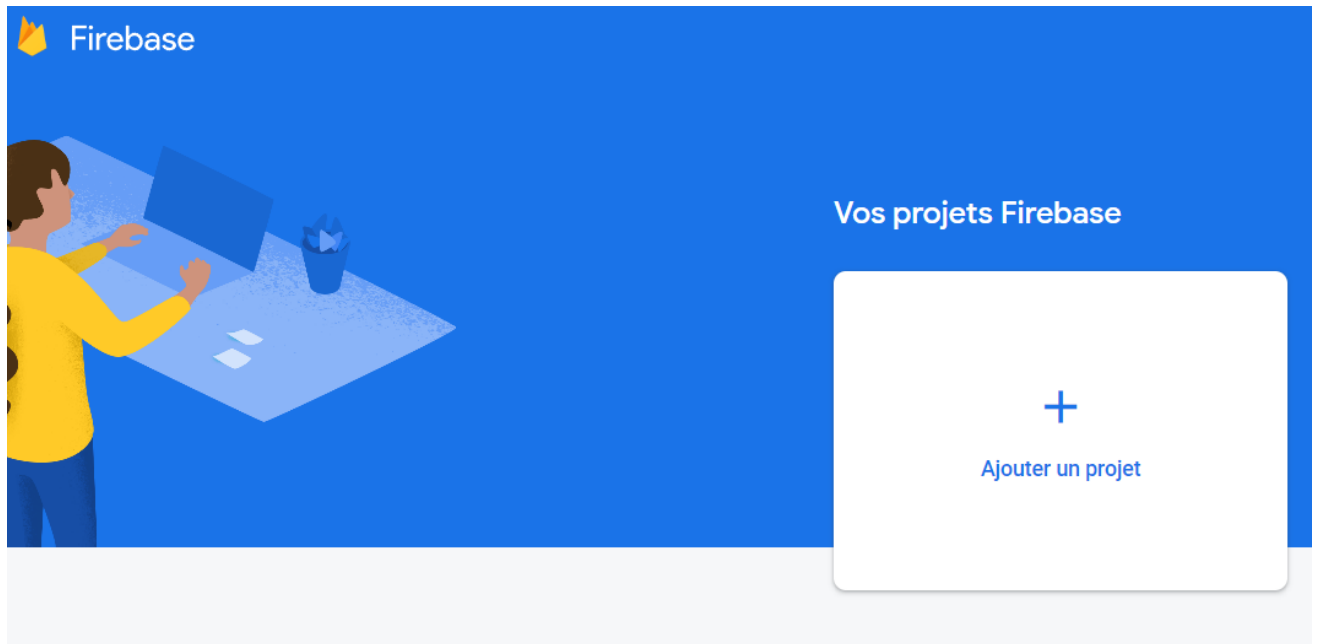


Figure 6 Page Firebase

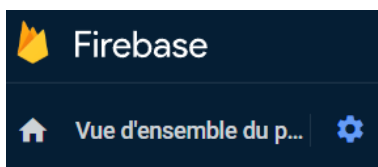


Figure 7 Position du pignon

Nom du projet	TempobyImmersize 
ID du projet 	tempobyimmersize
Numéro du projet 	1063661004815
<u>Emplacement des ressources GCP par défaut </u>	europa-west6
<u>Clé API Web</u>	AlzaSyA8e-fXxC81GEbOAVHlPrcPMao57_q35hA

Une fois dans les paramètres généraux, commencez par créer un nouveau projet. Ensuite, cliquez sur l'icône de la roue dentée pour accéder aux paramètres unifiés. Sur cette page, assurez-vous de définir correctement la région du serveur (non seulement pour le référencement, mais aussi pour l'expérience utilisateur). Et n'oubliez pas de noter le code API pour l'intégrer ultérieurement à votre application locale.

Exécutez la commande [npm](#) pour installer la CLI ou la mettre à jour.

```
$ npm install -g firebase-tools
```

Se connecter à Google

```
$ firebase login
```

Lancez votre projet

Exécutez la commande suivante à partir du répertoire racine de votre application :

```
$ firebase init
```

Figure 8 Étapes de l'installation

Une fois prêt, déployez votre application Web

Placez vos fichiers statiques (par exemple, HTML, CSS, JS) dans le répertoire de déploiement de votre application (le paramètre par défaut est "public"), puis exécutez cette commande depuis le répertoire racine de l'application :

```
$ firebase deploy
```

Figure 9 Étapes de l'installation

Une fois que vous avez configuré l'hébergement sur le serveur, suivez les instructions de code fournies par Firebase pour installer le programme. Il y a quatre instructions dans l'image : la première consiste à installer les composants de base de Firebase, ce que vous avez déjà fait. La deuxième instruction, "firebase login", permet de se connecter à votre compte correspondant via la connexion Google pour la configuration. La troisième instruction est utilisée pour initialiser l'environnement local en vue de la configuration. Enfin, la dernière instruction est utilisée pour téléverser les fichiers locaux sur le serveur et les monter ou les connecter aux services nécessaires.

Le processus d'initialisation avec "**firebase init**" est assez complexe, et si la configuration n'est pas correcte, cela peut entraîner un échec de l'initialisation et nécessiter des frais supplémentaires.

Créez un dossier local et donnez-lui le nom de votre projet. Ouvrez la console CMD et saisissez la commande "**firebase init**" pour commencer le processus d'initialisation des paramètres.

Dans cette étape, nous allons installer les services nécessaires. Dans ce cas, le service d'hébergement (hosting) est essentiel, car il nous permet de téléverser notre application sur le serveur et de la rendre accessible en ligne.

```
> Hosting: Configure and deploy Firebase Hosting sites
  ○ Storage: Deploy Cloud Storage security rules
  ● Emulators: Set up local emulators for Firebase features
```

Figure 10 Étapes de l'installation

Dans Firebase, puisque nous avons déjà créé un projet, nous pouvons simplement sélectionner le nom du projet correspondant lors de la configuration.

```
? Please select an option:  
> Use an existing project
```

Figure 11 Étapes de l'installation

Il est important de noter que cette configuration signifie que nous devons stocker les fichiers que nous souhaitons téléverser dans le dossier "public". Cependant, dans le cas de l'utilisation d'Angular pour notre application, Angular stocke les fichiers optimisés dans le dossier "/dist". Par conséquent, nous devons modifier l'emplacement de stockage des fichiers en le modifiant en **"/dist" (ceci est très important)**.

```
≡ Hosting Setup  
  
Your public directory is the folder (relative to your project directory) that will contain Hosting assets to be uploaded with firebase deploy. If you have a build process for your assets, use your build's output directory.  
  
? What do you want to use as your public directory? public  
? Configure as a single-page app (rewrite all urls to /index.html)?  
? Set up automatic builds and deploys with GitHub? No  
✓ Wrote public/404.html  
✓ Wrote public/index.html
```

Figure 12 Étapes de l'installation

Lors de la configuration de l'application Angular pour Firebase, il est important de spécifier si l'application est une application à page unique (SPA). Pour cela, vous devez indiquer 'yes'.

Pour le reste, vous pouvez laisser les options par défaut. Cela inclut la possibilité d'installer une interface de simulateur, ainsi que la configuration du port local (ce qui vous permettra de tester votre application localement pour vérifier si elle fonctionne correctement avant de la téléverser).

```
≡ Emulators Setup  
? Which Firebase emulators do you want to set up? Press Space to select emulators, then Enter to confirm your choices. Hosting Emulator  
? Which port do you want to use for the hosting emulator? 5000  
? Would you like to enable the Emulator UI? (Y/n)
```

Figure 13 Étapes de l'installation

Enfin, vous recevrez un message indiquant que l'installation a réussi. De plus, dans le dossier de votre projet, vous trouverez quelques fichiers de configuration Firebase.

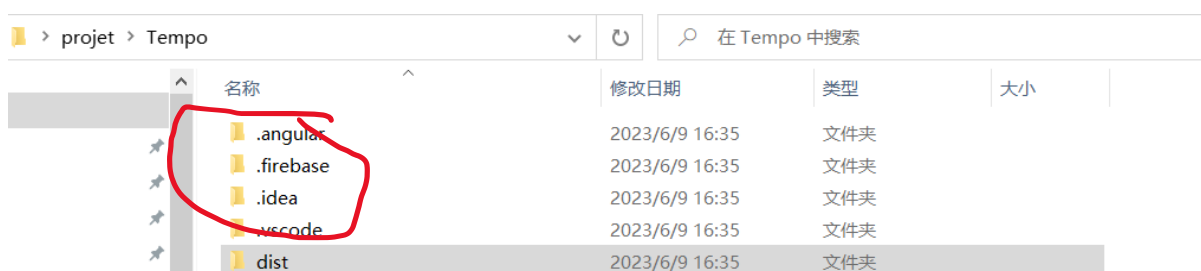


Figure 14 Installation réussie

2.1.2 Angular



Angular est un framework de développement d'applications web développé par Google. Il permet de créer des applications web dynamiques, interactives et évolutives. Voici quelques points importants sur Angular :

1. Architecture basée sur les composants : Angular adopte une architecture basée sur les composants, où chaque composant représente une partie spécifique de l'interface utilisateur. Les composants sont réutilisables, ce qui permet une gestion facile de l'application et une organisation claire du code.
2. Binding de données : Angular offre un puissant système de binding de données bidirectionnel, ce qui signifie que les modifications effectuées dans le modèle se reflètent automatiquement dans la vue et vice versa. Cela facilite la synchronisation des données entre les différentes parties de l'application.
3. Gestion de l'état : Angular propose un mécanisme de gestion de l'état intégré, appelé NgRx, qui permet de gérer efficacement les flux de données et les changements d'état dans l'application. Cela facilite la gestion de la complexité des applications.
4. Routing : Angular propose un système de routage intégré qui permet de gérer la navigation entre les différentes vues de l'application. Il prend en charge la navigation simple, la navigation avec des paramètres et la navigation par onglets, entre autres fonctionnalités.
5. Tests unitaires : Angular intègre des outils et des bibliothèques pour faciliter les tests unitaires de l'application. Cela permet de s'assurer que l'application fonctionne correctement et de détecter rapidement les éventuels problèmes.
6. Performance : Angular est conçu pour être rapide et performant. Il utilise des techniques d'optimisation telles que le lazy loading des modules, la détection du changement et le tree shaking pour améliorer les performances de l'application.
7. Écosystème et communauté active : Angular bénéficie d'un écosystème riche, avec de nombreuses bibliothèques et extensions tierces disponibles pour faciliter le développement. La communauté Angular est également très active, ce qui permet de trouver rapidement de l'aide et des ressources en cas de besoin.
8. Angular est largement utilisé pour développer des applications web modernes et évolutives. Sa structure modulaire, son système de binding de données et ses nombreuses fonctionnalités en font un choix populaire parmi les développeurs pour la création d'applications web robustes et performantes.

2.1.2.1 Installation

1. Installez Angular CLI : Ouvrez ton terminal ou ton invite de commandes et exécutez la commande suivante pour installer Angular CLI de manière globale : `node -v`

2.Installez Angular CLI : Ouvrez ton terminal ou ton invite de commandes et exécutez la commande suivante pour installer Angular CLI de manière globale : `npm install -g @angular/cli`

3.Créez un nouveau projet : Choisissez un répertoire approprié, puis exécutez la commande suivante dans votre terminal ou ton invite de commandes pour créer un nouveau projet Angular : `ng new TEMPO`

4.Lancez le serveur de développement : Exécutez la commande suivante pour démarrer le serveur de développement et prévisualiser votre application dans le navigateur : `ng serve --OPEN`

名称	修改日期	类型	大小
.angular	2023/6/9 16:35	文件夹	
.firebase	2023/6/9 16:35	文件夹	
.idea	2023/6/9 16:35	文件夹	
.vscode	2023/6/9 16:35	文件夹	
dist	2023/6/9 16:35	文件夹	
node_modules	2023/6/9 16:39	文件夹	
public	2023/6/9 16:39	文件夹	
src	2023/6/9 16:39	文件夹	
.browserslistrc	2023/4/21 11:57	BROWSERSLIST...	1 KB
.editorconfig	2023/4/21 11:57	Editor Config 源...	1 KB
.firebaseerc	2023/4/21 11:57	FIREBASERC 文件	1 KB
.gitignore	2023/4/21 11:57	文本文档	1 KB
{	2023/4/21 11:57	文件	0 KB
angular.json	2023/4/21 11:57	JSON 源文件	4 KB
firebase.json	2023/4/21 11:57	JSON 源文件	1 KB
karma.conf.js	2023/4/21 11:57	JetBrains WebSt...	2 KB
package.json	2023/4/21 11:57	JSON 源文件	2 KB
package-lock.json	2023/4/28 15:00	JSON 源文件	1,070 KB
README.md	2023/4/21 11:57	Markdown 源文件	2 KB
tsconfig.app.json	2023/4/21 11:57	JSON 源文件	1 KB
tsconfig.json	2023/4/21 11:57	JSON 源文件	1 KB
tsconfig.spec.json	2023/4/21 11:57	JSON 源文件	1 KB
webpack.config.js	2023/4/21 11:57	JetBrains WebSt...	1 KB

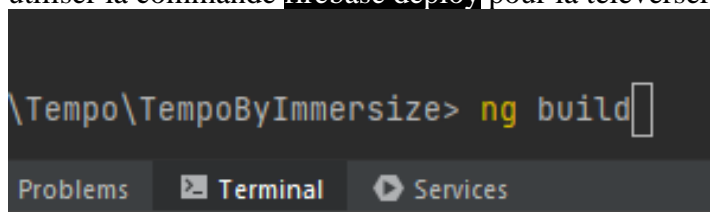
Figure 15 Contenu du Projet

Une fois terminé, votre dossier générera des fichiers de configuration Angular et votre application sera développée dans le dossier src (/src).

2.1.2.2 Mettre en ligne

Comme nous l'avons mentionné précédemment, l'utilisation de la commande `ng serve --open` permet de vérifier si l'application fonctionne correctement localement en détectant d'éventuelles erreurs. Une fois que vous avez réussi à ouvrir l'application et qu'elle fonctionne parfaitement, nous pouvons procéder à la création du package et à son téléversement dans le référentiel de stockage Firebase.

Nous devons utiliser la commande `ng build` pour générer le package de l'application, puis utiliser la commande `firebase deploy` pour la téléverser sur le site web.



```
Tempo\TempoByImmersize> firebase deploy
```

Problems | Terminal | Services

/main (11 minutes ago)

Figure 16 Instructions pour mettre en ligne

2.1.2.3 Présentation

Je vais maintenant vous présenter les détails de notre projet de création, en discutant de chaque dossier dans le programme Angular et de leur rôle respectif, ainsi que des travaux de débogage associés.

- Structure de l'application – code

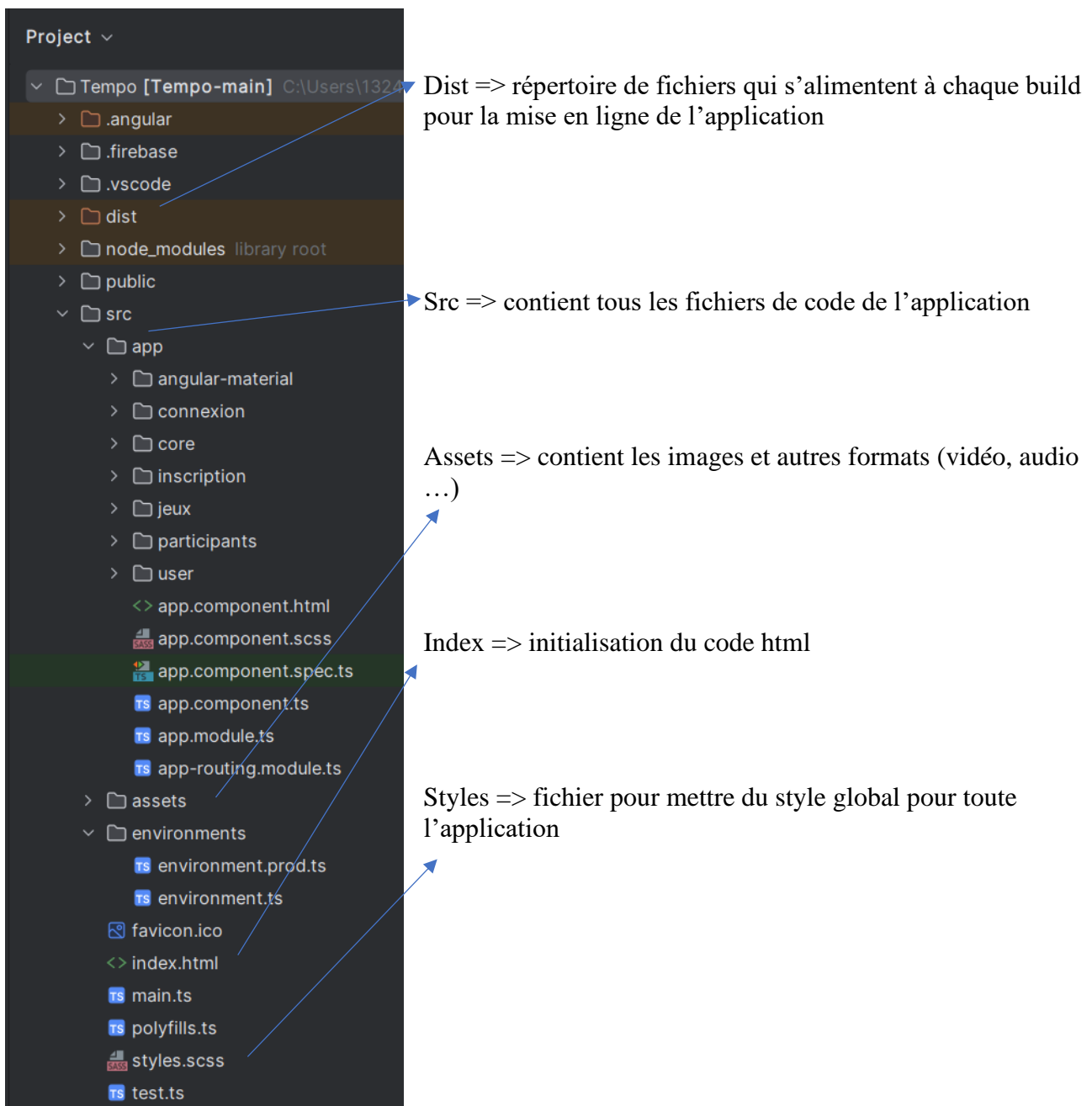


Figure 17 arborescente

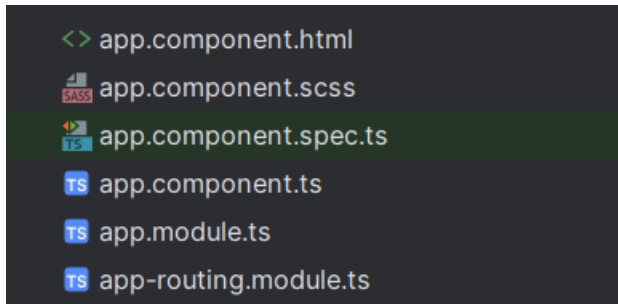


Figure 18 fichiers de base angulaires

App.html : base du routage.

App.component.ts : définition de la tire du site.

App.module.ts : import de tous les modules et composants de l'application

App-routing : définition des routes principales de l'appli et charge les routes enfants en lazy loading.

Le chargement différé (Lazy Loading) est une stratégie de chargement où les ressources ou modules sont chargés uniquement lorsqu'ils sont nécessaires. Dans le développement web, le chargement différé est généralement utilisé pour retarder le chargement de ressources volumineuses ou pour diviser une application en plusieurs modules et les charger dynamiquement lorsque cela est nécessaire.

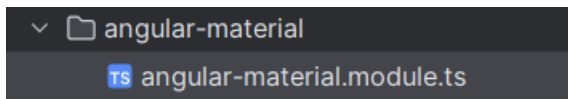


Figure 19 Intégration des fichiers de modules

Angular Material => regroupe tous les imports utilisés pour utiliser les fonctionnalités d'[Angular Matériel](#), solution facile à utiliser et responsive pour les applications angular.

Dans chaque module, il y a généralement quatre fichiers fondamentaux :

1. **Fichier HTML** : Utilisé pour afficher le contenu de la page Web. Il contient la structure de la page et peut inclure des balises HTML pour organiser et afficher les éléments de l'interface utilisateur.
2. **Fichier SCSS** : Il s'agit d'un fichier de style utilisant la syntaxe SCSS (Sass) qui est une extension du CSS. Le fichier SCSS est utilisé pour appliquer des styles, des animations et des mises en forme à la page HTML. Il offre une syntaxe plus avancée et des fonctionnalités supplémentaires par rapport au CSS traditionnel.
3. **Fichier spec.ts** : Il s'agit d'un fichier de configuration spécifique à Angular utilisé pour écrire des tests unitaires pour le module. Il n'est pas généralement utilisé pour le développement de l'application elle-même.
4. **Fichier TS (TypeScript)** : Il s'agit d'un fichier basé sur JavaScript qui contient la logique de l'application. Il est utilisé pour l'interaction avec les utilisateurs, la validation des données, la connexion aux serveurs Firebase, et d'autres fonctionnalités de l'application. TypeScript est un sur-ensemble de JavaScript qui offre des fonctionnalités supplémentaires telles que le typage statique, les classes, les modules, etc.



Figure 20 Module d'enregistrement

Connexion => code pour la partie connexion du user à l'application
 Sign-in.html => code affichage
 Sign-in.scss => code pour le style
 Sign-in.ts => logique en typescript pour les interactions avec la BDD(Base de Données) et les autres composants ou services.

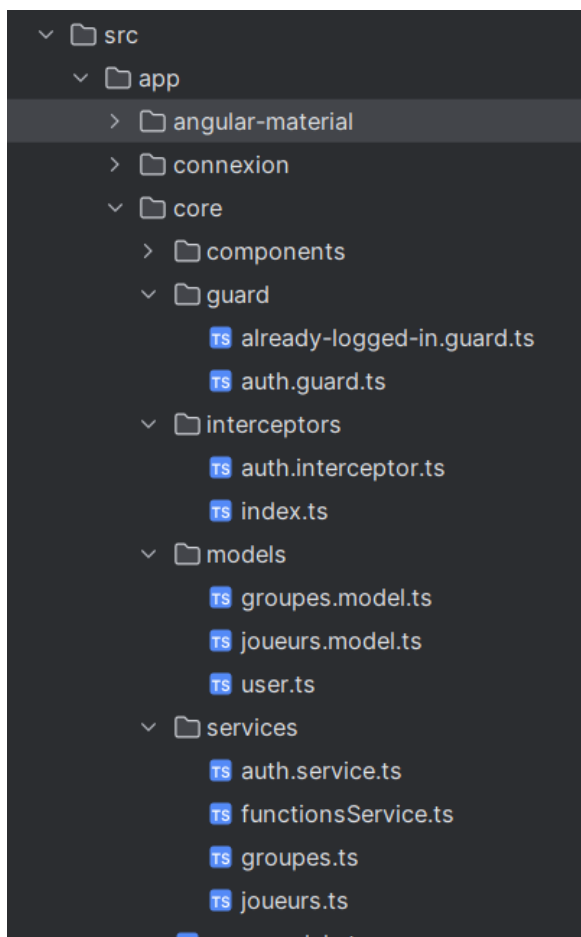


Figure 22 Contenu du projet

Connexion.module.ts => import des composants ou modules utilisés et export des composants présents dans connexion

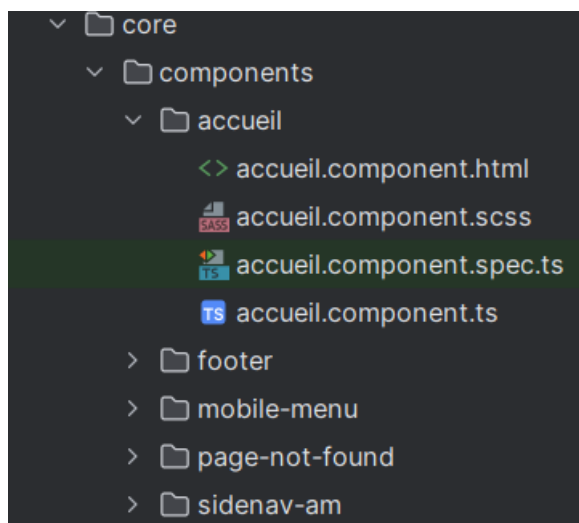


Figure 21 Les fichiers qui composent l'accueil

Core => code qui comporte la base de l'application.
 Accueil => page d'accueil.

Footer => pied de page.
 Mobile menu => code pour le menu quand l'appli est sur mobile.

Page-not-found => page quand l'utilisateur est connecté et ne met pas une route existante.

Sidenav => contient le code pour le header et le menu latéral sur la gauche (PC) et inclus le menu mobile.

Guard => limite l'accès à l'application si user connecter ou pas.

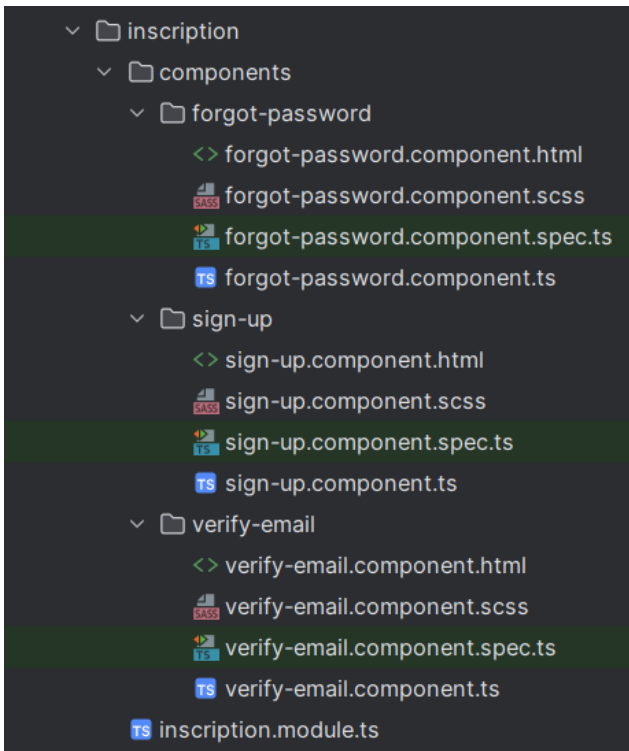
Models => objets utilisés dans l'appli et qui contiennent des informations qui les définissent (id, nom, prénom, courriel ...)

Services => regroupent les fonctions utilisables ensuite dans chaque composant lorsqu'on les importe dans le constructeur.

Auth.service => fonctions pour authentification

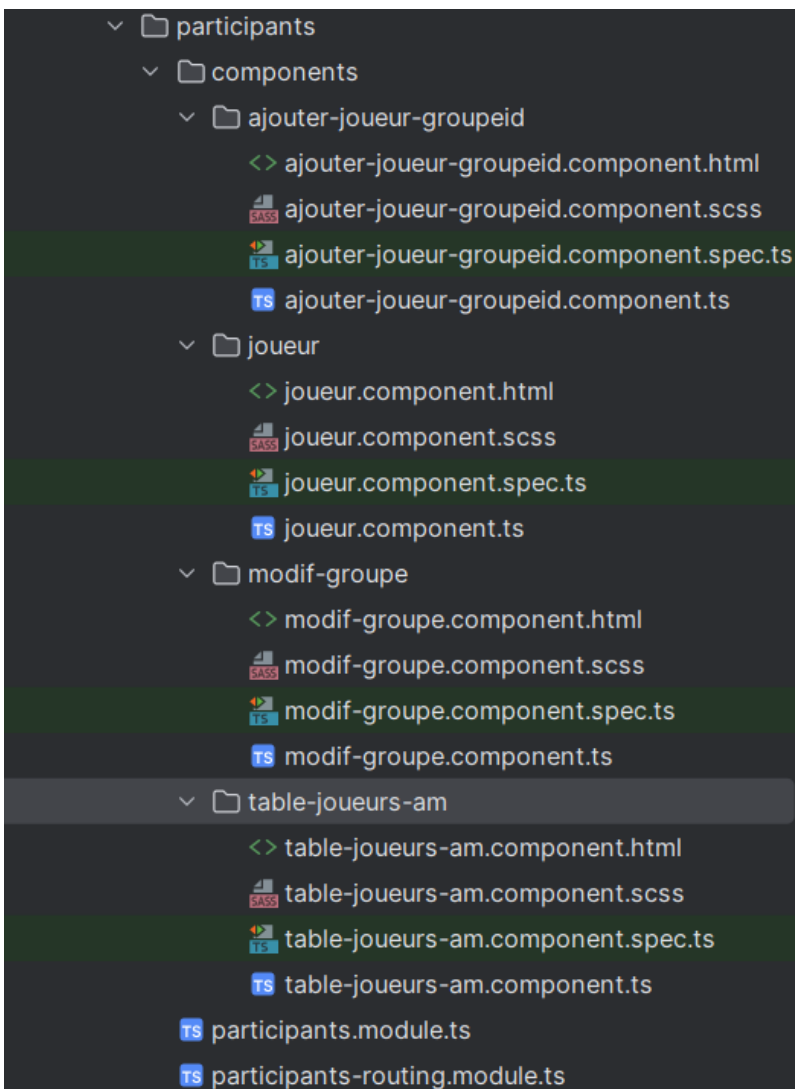
functionsService => fonctions utilisées dans une majeure partie de l'application.

Groupes et joueurs => fonctions qui récupèrent, créent, modifient ou suppriment les données joueurs ou groupes.



Inscriptions => code pour les pages d'inscription du user

Figure 23 Contenu du projet



Participants => contient le code pour afficher ou ajouter des joueurs de l'application unity.

Ajouter-joueur : mutualise le code pour ajouter un joueur avec un groupe prédéfini ou pas.

Joueur : affichage des profils des joueurs
Modif-groupe : code pour modifier le groupe d'un joueur.

Table-joueur-am : tableau de la liste des joueurs, am pour Angular Material.

Figure 24 Contenu du projet

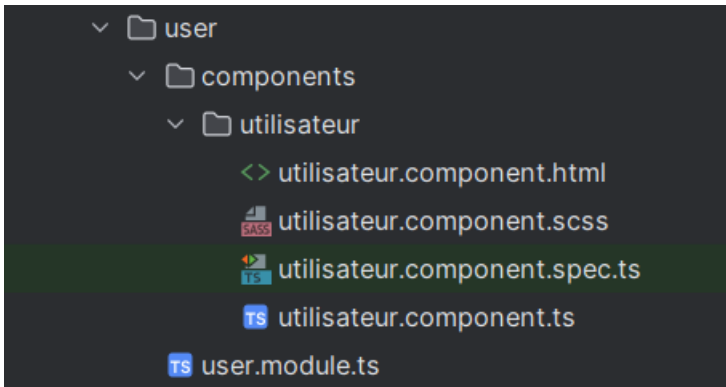


Figure 25 Contenu du user

User => Le code pour l'affichage du profil du user.

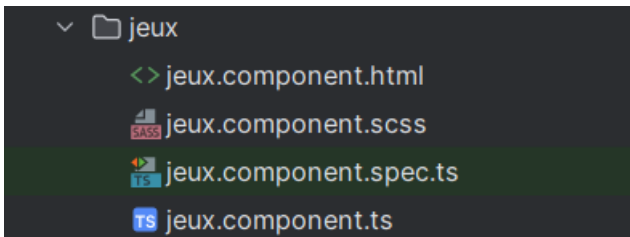


Figure 26 Contenu du jeux

Jeux => Le code suivant permet d'afficher un bouton de démarrage du jeu pour faciliter le contrôle par l'utilisateur.

À l'étape de la présentation du produit, je vais maintenant afficher l'interface finale de l'application Web pour illustrer les différents modules que j'ai décrits lors de ma présentation sur les produits.

Je vais les inclure dans la section des illustrations pour une visualisation plus claire.

2.1.2.4 Responsable du débogage

1. Bloquer inscription mail => domaine se terminer par@lilleavenirs.fr la dernière.

La solution:

```
export class SignUpComponent {
  email = new FormControl("", [Validators.required, Validators.email, Validators.pattern("^[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}$")]);
  // email = new FormControl("", [Validators.required, Validators.email, Validators.pattern("^[a-z0-9._%+-]+@lilleavenirs\\.fr$")]);
  userPwd = new FormControl("", [Validators.required]);
  hide = true;
  userDisplayName = new FormControl("", [Validators.required]);
}
```

(1). email est un contrôleur pour un champ de saisie qui accepte une adresse e-mail. Il a plusieurs validateurs :

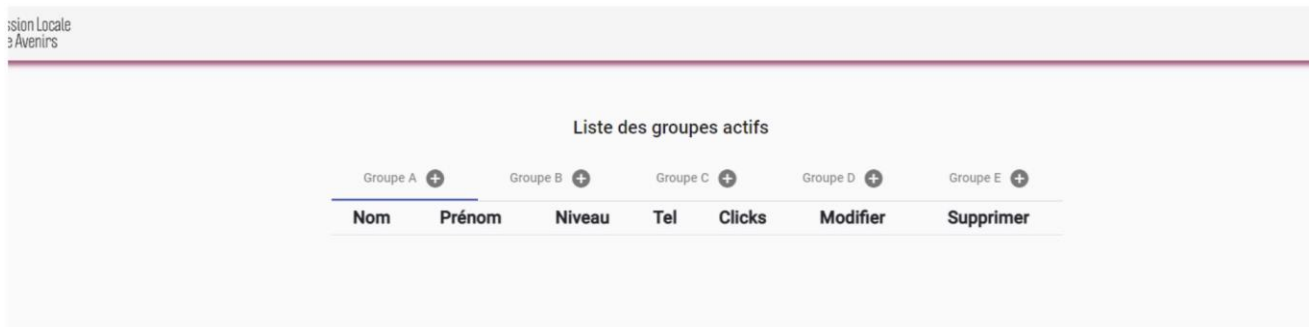
- Validators.required garantit que le champ a été rempli, n'est pas vide.
- Validators.email vérifie que l'entrée est au format e-mail.
- Validators.pattern vérifie que l'entrée correspond à une expression régulière donnée. Dans votre exemple, l'expression régulière est "^[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}\$", ce qui signifie qu'il n'acceptera que les adresses e-mail qui correspondent à ce format spécifique.

(2). userPwd est un contrôleur pour le champ de saisie du mot de passe de l'utilisateur. Son Validators.required garantit que le champ du mot de passe n'est pas vide.

(3). hide est une variable booléenne, principalement utilisée pour contrôler si le champ du mot de passe est affiché ou non.

(4). userDisplayName est un contrôleur pour le champ de saisie du nom d'affichage de l'utilisateur. Comme userPwd, il utilise également Validators.required pour s'assurer que le nom d'utilisateur est rempli.

2.Si la liste des membres est vide, affichez le message 'Aucune liste de membres disponible'.



La solution :

```
<div class="groupe">
  <h2>Liste des groupes actifs</h2>
  <mat-tab-group>
    <mat-tab *ngFor="let groupe of functionsService.groupes" label="{{groupe.id}}">
      <ng-template mat-tab-label>
        {{groupe.id}}
        <mat-icon (click)="functionsService.ajouterJoueurGroupe(groupe.id)"
class="add">add_circle</mat-icon>
      </ng-template>

    <ng-container *ngIf="filteredJoueurs(groupe.groupId).length > 0; else noPlayers">
      <table class="table table-hover">
        <thead>
          <tr>
            <th scope="col">Nom</th>
            <th scope="col">Prénom</th>
            <th scope="col">Niveau</th>
            <th scope="col">Tel</th>
            <th scope="col">Clicks</th>
            <th scope="col">Modifier</th>
            <th scope="col">Supprimer</th>
          </tr>
        </thead>
        <tbody>
          <ng-container *ngFor="let joueur of filteredJoueurs(groupe.groupId)">
            <tr>
              <td (click)="functionsService.showDetailJoueur(joueur.id)"
class="click">{{joueur.name}}</td>
              <td (click)="functionsService.showDetailJoueur(joueur.id)"
class="click">{{joueur.firstName}}</td>
              <td>{{joueur.level}}</td>
              <td>{{joueur.telephone}}</td>
              <td>{{joueur.click_count}}</td>
            </tr>
          </ng-container>
        </tbody>
      </table>
    </ng-container>
  </mat-tab-group>
</div>
```

```

        <td><mat-icon (click)="functionsService.modifGroupeJoueur(joueur.id)">edit</mat-
icon></td>
        <td><mat-icon (click)="functionsService.supprimerJoueur(joueur.id)">delete</mat-
icon></td>
    </tr>
</ng-container>
</tbody>
</table>
</ng-container>

<ng-template #noPlayers>
    <div class="empty-group">
        Le groupe est vide
    </div>
</ng-template>
</mat-tab>
</mat-tab-group>
</div>

```

J'utilise les éléments `<mat-tab-group>` et `<mat-tab>` pour créer un ensemble d'onglets et une série d'onglets. Chaque onglet correspond à un élément du tableau `functionsService.groupes`. Chaque onglet est défini avec l'élément `<ng-template>`, le contenu de l'onglet est l'id de l'élément correspondant, et il y a une icône d'ajout (`add_circle`). En cliquant sur cette icône, la méthode `functionsService.ajouterJoueurGroupe(groupe.id)` sera appelée.

Si `filteredJoueurs(groupe.groupId).length > 0` est faux (Cela signifie que le groupe n'a pas de membres.), un message d'alerte sera affiché, disant "Le groupe est vide".

3. Si une URL incorrecte est entrée pour la page de détails d'un joueur, la page ne renvoie pas une erreur 404 mais reste sur la page de détails d'un joueur inconnu, Les informations du joueur disparaissent après le rafraîchissement de la page.

La troisième question semble avoir deux difficultés, mais en réalité, il n'y a qu'un seul problème : comment garantir que les informations ne seront pas perdues à chaque actualisation de la page. Pour résoudre ce problème, nous utilisons le stockage local pour sauvegarder les informations du joueur.

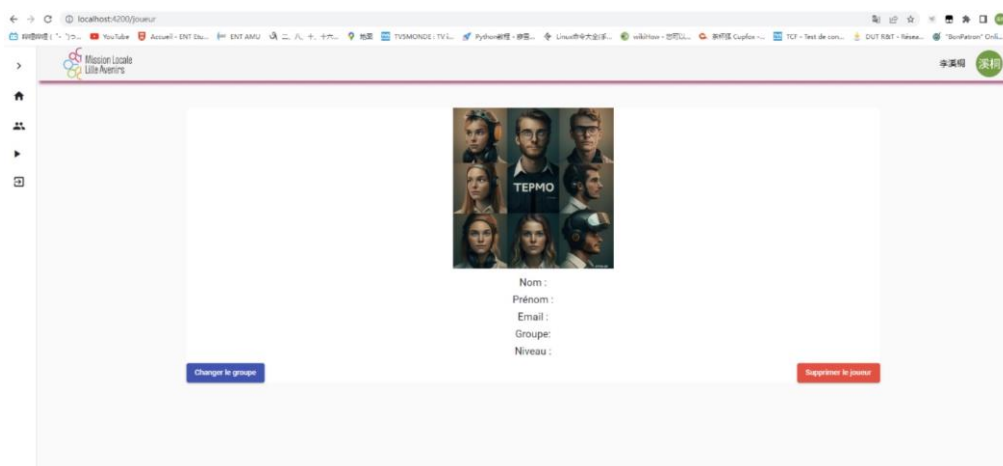


Figure 27 Problème

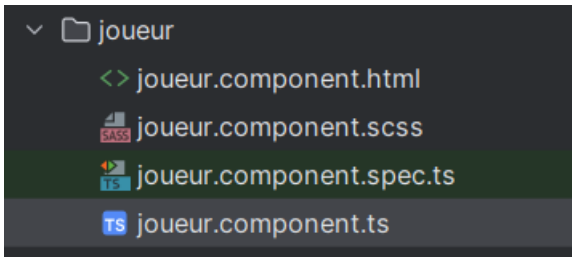


Figure 28 Composants du joue

```

ngOnInit(): void {
  // Récupérer les informations du joueur à partir du stockage local si elles existent
  const storedJoueur = localStorage.getItem('joueur');
  if (storedJoueur) {
    this.functionsService.jr = JSON.parse(storedJoueur);
  } else {
    this.route.queryParams.subscribe(params => {
      this.joueurId = params['joueurId'];
      this.functionsService.jr = {...params};
    });
  }
  const baseUrl = `${window.location.origin}${window.location.pathname}`;
  window.history.replaceState(null, "", baseUrl);
  this.functionsService.showAllGroupes();
}
}

```

La solution :

- (1). Tout d'abord, je tente de récupérer des informations appelées 'joueur' à partir du stockage local (localStorage). Le stockage local est un mécanisme fourni par le navigateur qui permet de stocker des paires clé-valeur sur l'ordinateur de l'utilisateur. Si ces informations existent dans le stockage local, elles sont alors analysées en format JSON et stockées dans functionsService.jr.
- (2). Si ces informations n'existent pas dans le stockage local, elle s'abonne alors aux paramètres de requête de la route. Lorsque ces paramètres de requête changent, la fonction d'abonnement est appelée. Dans cette fonction, elle définit joueurId comme étant le joueurId dans les paramètres de requête, et attribue tous les paramètres de requête à functionsService.jr.
- (3). Ensuite, je crée une URL de base qui est composée de window.location.origin (le domaine et le port du site) et de window.location.pathname (le chemin de la page actuelle).
- (4). Je utilise la méthode window.history.replaceState() pour remplacer l'entrée actuelle de l'historique de navigation par cette URL de base. Cela est pour supprimer les éventuels paramètres de requête dans l'URL afin d'éviter qu'ils soient rechargés lorsque l'utilisateur actualise la page.

Je vais maintenant continuer à présenter functionsService.jr. Ce fichier intègre tous les services, par exemple afficher les membres d'un groupe, montrer les informations du joueur, supprimer un joueur, etc. Il contient les principales fonctionnalités de base.

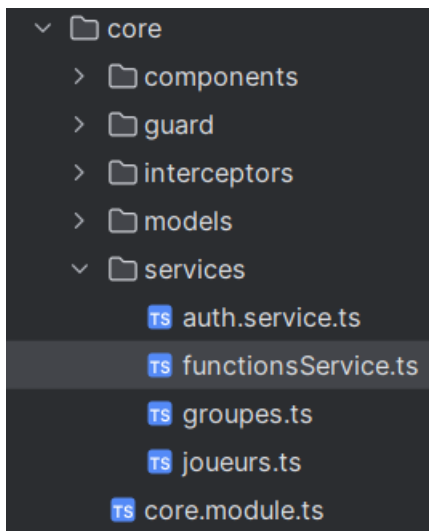


Figure 29 Composants du service

```
showDetailJoueur(joueurId: string | undefined) {
  this.joueursService.getById(joueurId).subscribe((doc:
firebase.firestore.DocumentSnapshot<firebase.firestore.DocumentData> | null) => {
  // Stocker les informations du joueur dans le stockage local
  const joueurData = {id: joueurId, ...doc?.data()};
  localStorage.setItem('joueur', JSON.stringify(joueurData));
  this.router.navigate(['/joueur'], {queryParams: joueurData});
});
}
```

- (1). J'appelle joueursService.getById(joueurId), qui est probablement une méthode qui récupère les informations du joueur avec l'ID joueurId à partir d'un service. Cette méthode retourne un Observable, qui peut être souscrit.
- (2). Je souscris à cet Observable avec la méthode subscribe. Lorsque l'Observable émet une valeur, la fonction dans subscribe est appelée. Cette fonction prend un paramètre doc, qui pourrait être un instantané de document Firebase Firestore ou null.
- (3). Dans la fonction dans subscribe, elle crée d'abord un objet appelé joueurData, qui contient l'ID du joueur et toutes les données récupérées de l'instantané du document.
- (4). J'utilise la méthode localStorage.setItem pour stocker joueurData dans le stockage local. 'joueur' est la clé de stockage et JSON.stringify(joueurData) est la valeur de stockage, qui est le résultat de la conversion de l'objet joueurData en une chaîne.
- (5). J'utilise la méthode this.router.navigate pour naviguer vers le chemin '/joueur', avec joueurData comme paramètres de requête.

En somme, cette méthode récupère les informations d'un joueur spécifique, les stocke dans le stockage local, puis navigue vers le chemin '/joueur' avec ces informations.

4. Lors de la connexion avec la touche Entrée, même si le mot de passe est correct, la connexion ne peut pas être effectuée correctement.

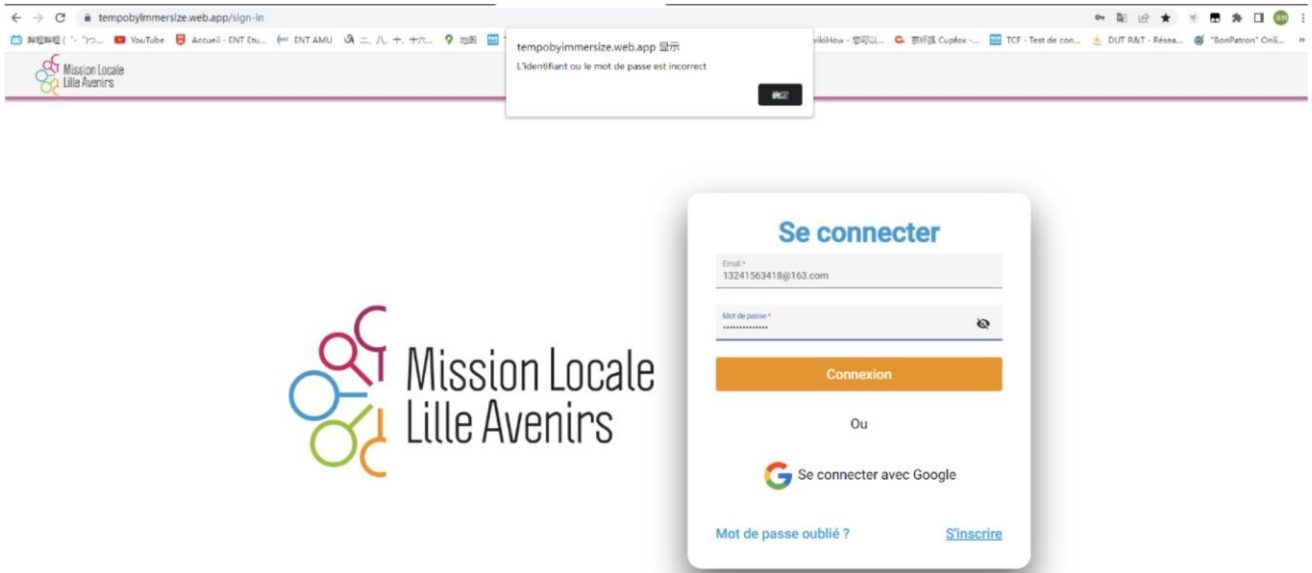


Figure 30 Page de connexion

La solution:

Le problème se situe dans le fichier sign-in.component.ts suivant. Pour une raison inconnue, la fonction onSubmit ne peut pas être utilisée lors de la génération de la page web. Par conséquent, j'ai directement ajouté la fonction de connexion dans le fichier HTML pour garantir que le nom d'utilisateur et le mot de passe peuvent être correctement vérifiés.

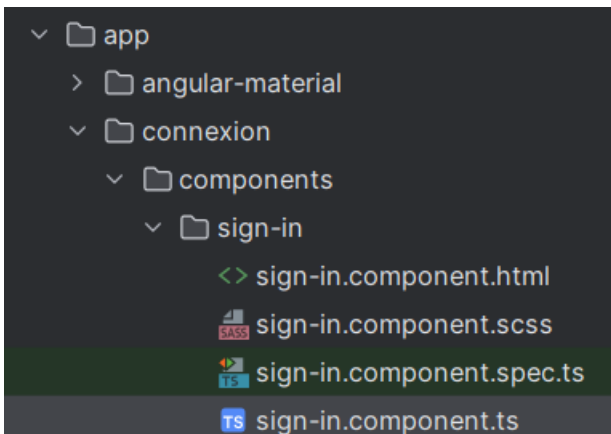


Figure 31 Composants pour sign-in

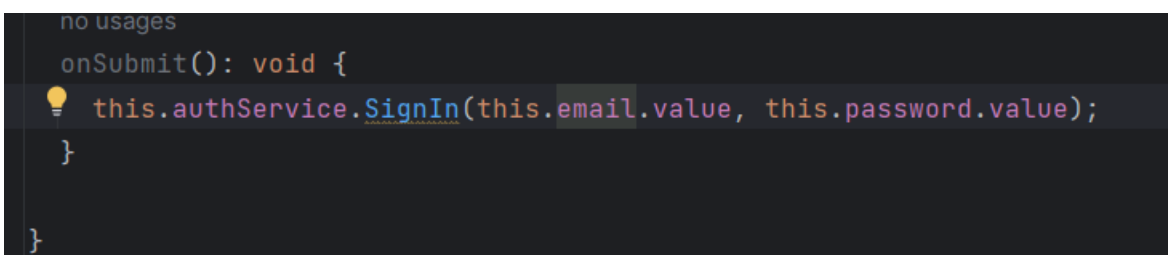
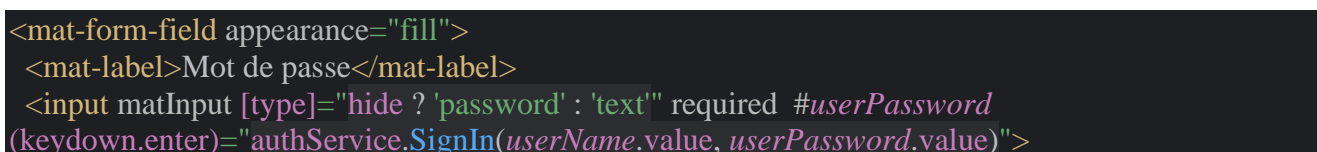


Figure 32 Instructions manquées



3. Le travail de la deuxième partie (Institut Universitaire de Technologie à Aix Marseille)

Dans cette partie, j'ai terminé un stage de développement d'applications web de huit semaines et je suis passé à la deuxième partie, qui consiste à retourner à l'IUT pour effectuer la configuration et la maintenance des équipements réseau. Ensuite, je vais présenter deux parties. Dans la première partie, je vais expliquer comment installer des routeurs et des armoires de serveurs dans une salle des machines. Dans la deuxième partie, je vais expliquer comment mettre à jour les équipements réseau et que faire en cas de situations particulières.

Pendant le stage de deux semaines, il y a deux tâches importantes :

- Installer le matériel, monter les racks et le pont et fixer les commutateurs et routeurs sur le rack.
- Organiser tous les systèmes de commutation et réinstaller les parties défectueuses du système.

3.1 Équiper une salle de classe avec un réseau informatique.

Étape 1 : Lors de l'aménagement de la salle de classe, il est important de prêter attention aux points suivants :

- Installer des plaques murales.

La tâche d'aménagement de la pièce est effectuée en collaboration avec M. Ali. Comme les plaques sont très lourdes, il faut deux personnes pour les manipuler. Une personne soulève la plaque pendant que l'autre se charge de la fixer à l'aide de vis et d'écrous.



Figure 33 Plaque

- Installer des prises électriques multifonctionnelles.

Chaque poste dispose de deux baies, une haute et une basse. Nous équipons chaque baie d'une multiprise fonctionnelle afin de maintenir un aspect esthétique. Cette multiprise a pour fonction de connecter tous les routeurs, commutateurs et autres équipements réseau présents sur ce rack.



Figure 34 prise multifonctionnelle

Parmi eux, chaque poste dispose d'une baie basse. Sur cette baie basse, nous devons installer une autre multiprise fonctionnelle à une hauteur inférieure. Cela permettra de connecter tous les appareils électriques tels que les ordinateurs, les écrans et autres équipements électroniques.

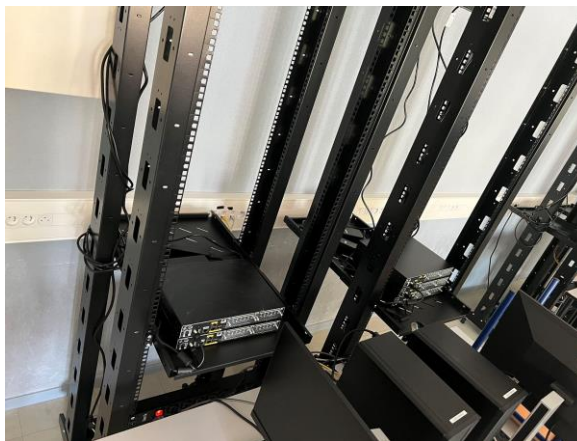


Figure 36 Routeur

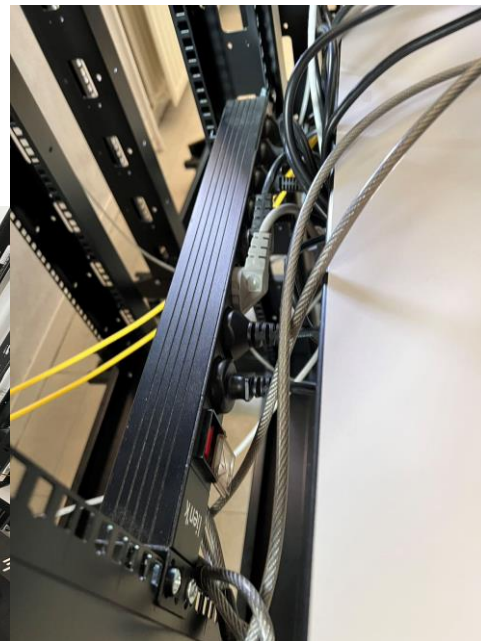


Figure 35 prise multifonctionnelle

- Installer un routeur.

Parfois, les câbles d'alimentation peuvent s'emmêler. Nous devons les dérouler autant que possible ou les dissimuler pour préserver l'esthétique.



Figure 37 Routeur

- Assurer une disposition soignée de l'environnement (agencement des câbles).



Figure 38 Intégrer et protéger

Nous utilisons un matériau en mousse pour envelopper tous les fils qui semblent très désordonnés. Cela permet de leur donner une apparence très soignée.

Voilà, les résultats de notre travail, réalisé en collaboration avec M. Ali

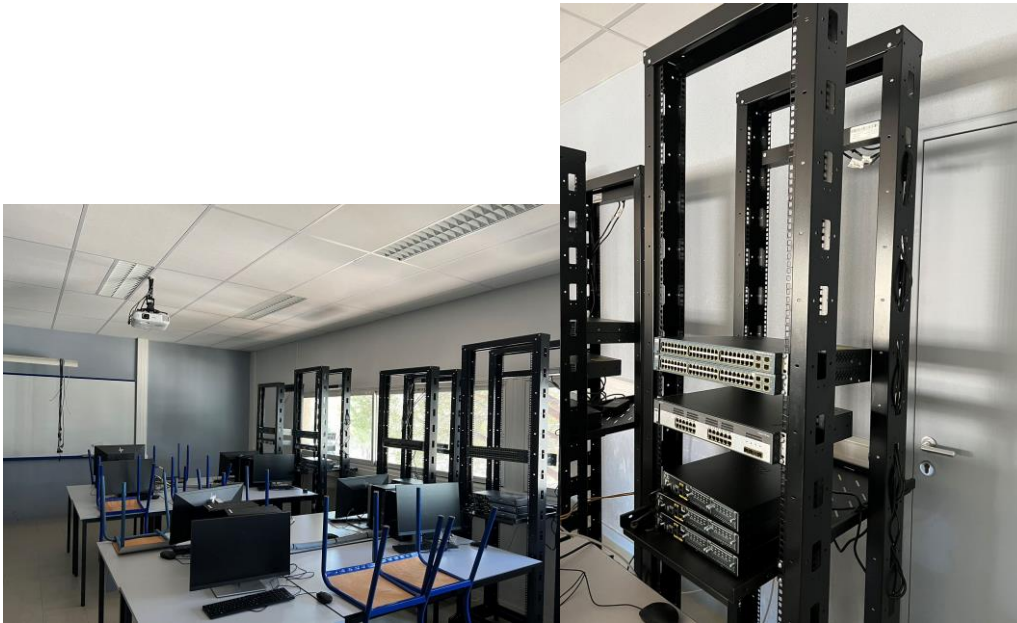


Figure 39 Résultats de l'exposition

Étape 2 : Actuellement, nous devons traiter les commutateurs, mais avant cela, nous avons besoin d'un schéma pour élaborer comment procéder à la réinstallation du système IOS des commutateurs. En supposant que tous les commutateurs fonctionnent correctement, nous devons les déployer dans la nouvelle salle de classe (Salle de Shanghai).

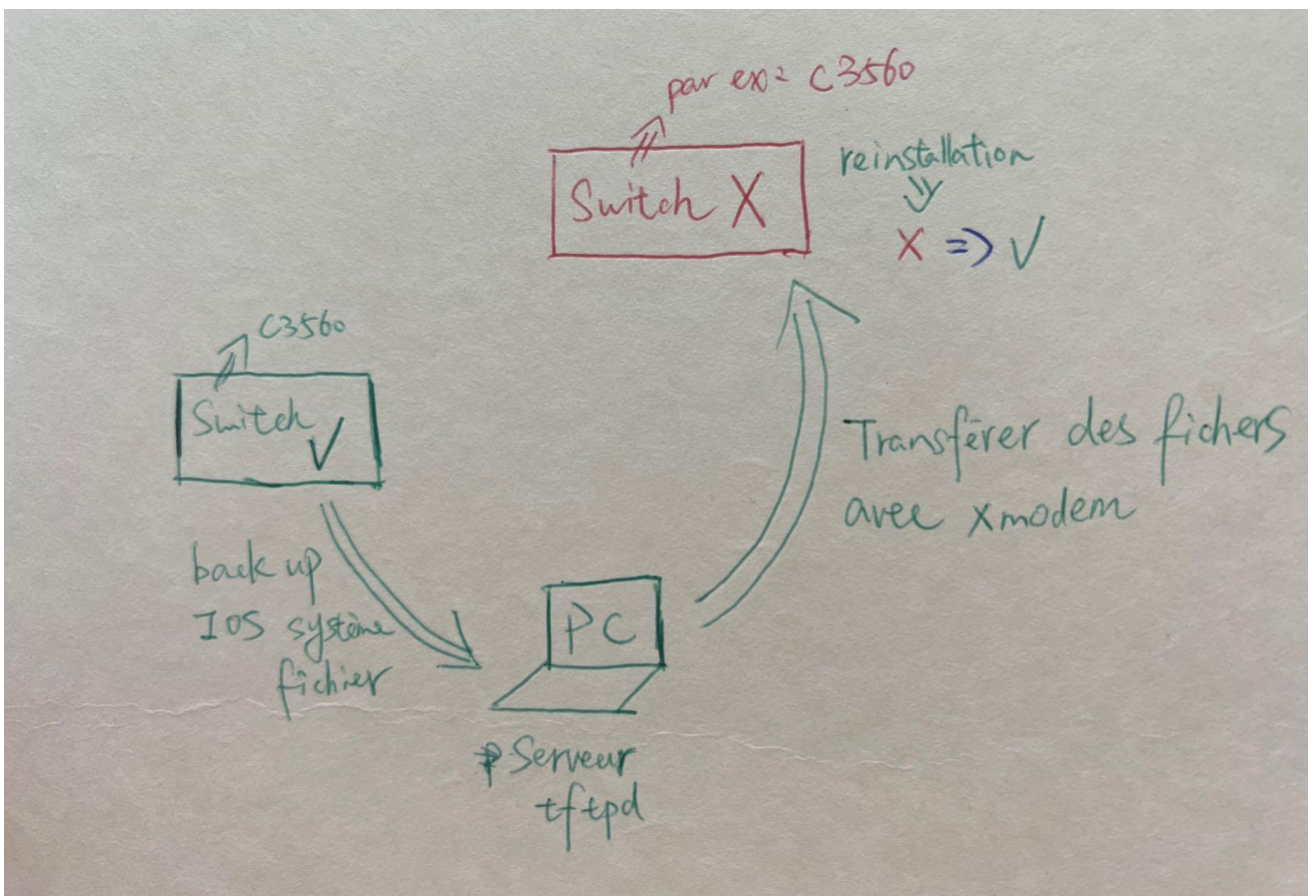


Figure 40 schéma

OK, nous avons élaboré une stratégie. Maintenant, nous devons mettre en place un moyen de détecter les serveurs et routeurs défectueux. J'ai créé un dossier où j'ai enregistré tous les relevés du "show version". Si ces machines présentent des problèmes, je les marquerai directement dans leur nom en utilisant des soulignements (_).

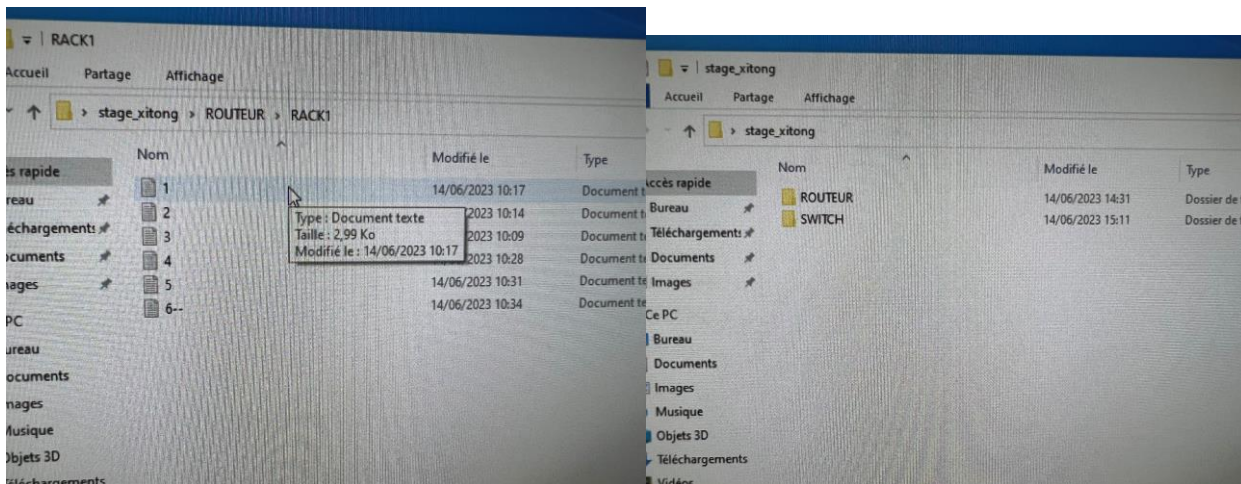


Figure 41 Documents enregistrés



Figure 42 Tags

Nous avons classé les racks, routeurs et commutateurs par catégories, et nous les avons étiquetés afin de pouvoir les retrouver rapidement.

La prochaine étape consiste à vous montrer quelques exemples typiques et à vous présenter leurs solutions respectives.

3.2 Maintenance des équipements réseau.

(1).

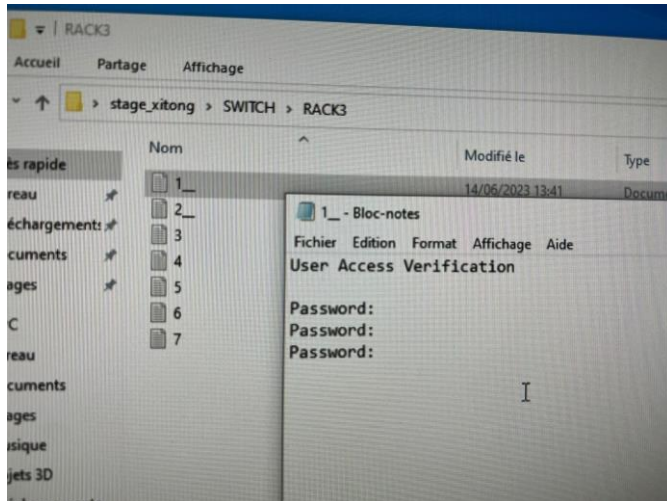


Figure 43 Ne connaît pas le mot de passe

La solution:

Étape 1 : Connectez votre ordinateur portable au port COM du commutateur à l'aide d'un câble de contrôle. Ouvrez la fenêtre de débogage du terminal superutilisateur (Super Terminal).

Étape 2 : Rallumez le commutateur. Pendant le processus de démarrage, maintenez enfoncé le bouton "mode" situé à l'avant du commutateur jusqu'à ce que le voyant "sys" cesse de clignoter (c'est-à-dire qu'il reste allumé en continu). Cela vous permettra d'accéder au mode de bas niveau du commutateur, indiqué par "switch :".

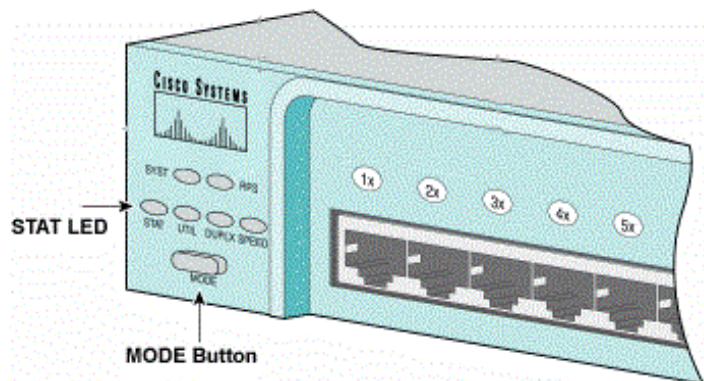
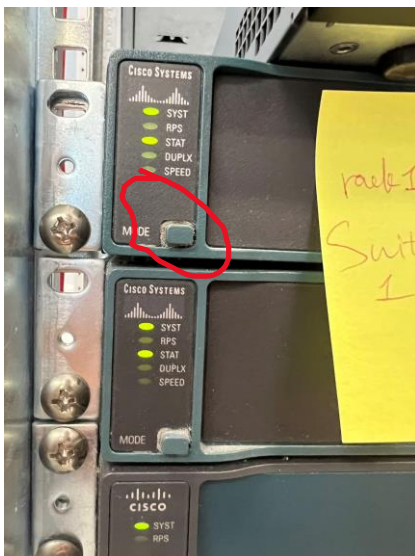
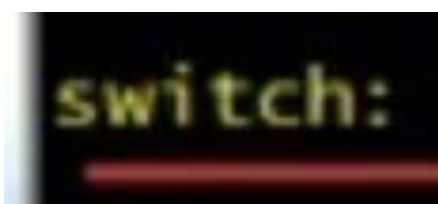


Figure 44 Bouton de changement de mode bios



Étape 3 : Entrez la commande "<flash_init>", puis appuyez sur Entrée pour initialiser le système de fichiers Flash. Ensuite, entrez la commande "<dir flash>" et appuyez sur Entrée pour vérifier les

(2).

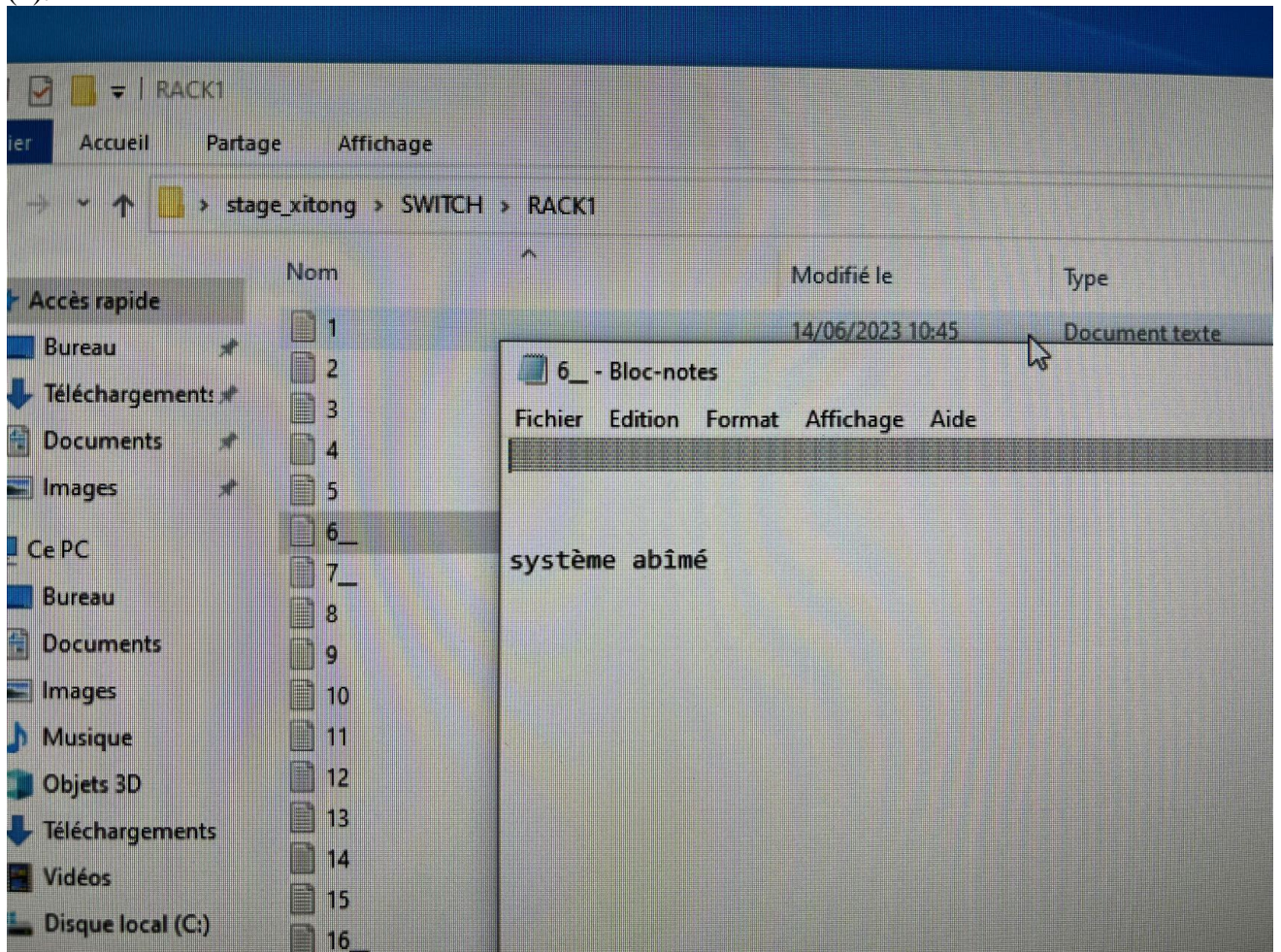


Figure 49 Commutateurs défectueux

La solution :

Si la deuxième image présente simplement une différence de vitesse, il suffit de modifier la vitesse des ports pour rétablir le bon fonctionnement, ou bien pour les faire correspondre au schéma de la troisième image. Nous pouvons discuter des détails plus en détail par la suite.

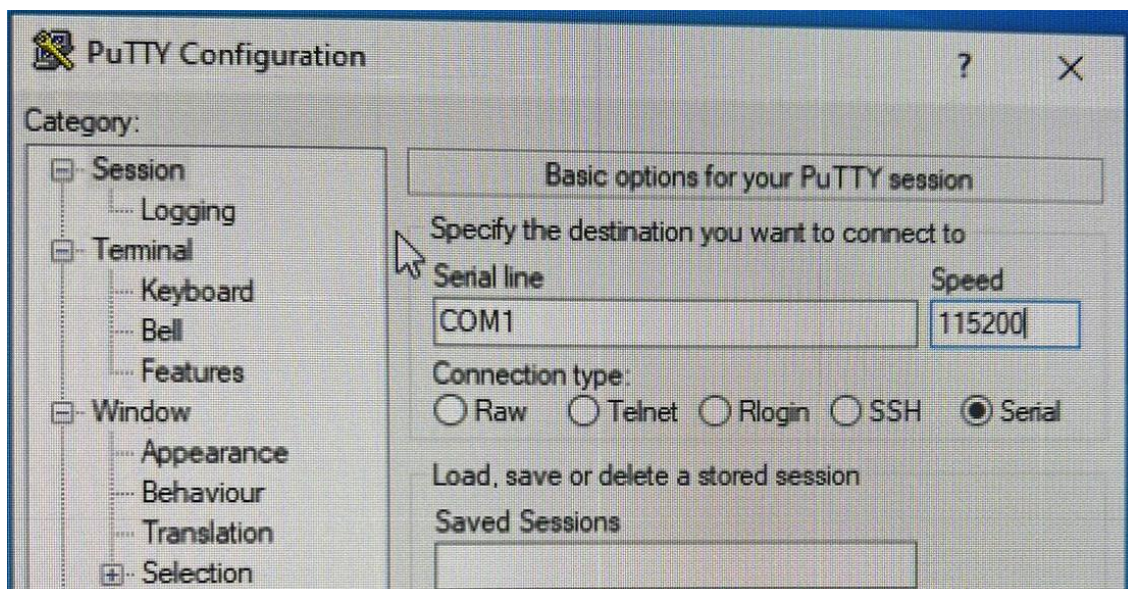


Figure 50 interface de putty

(3).

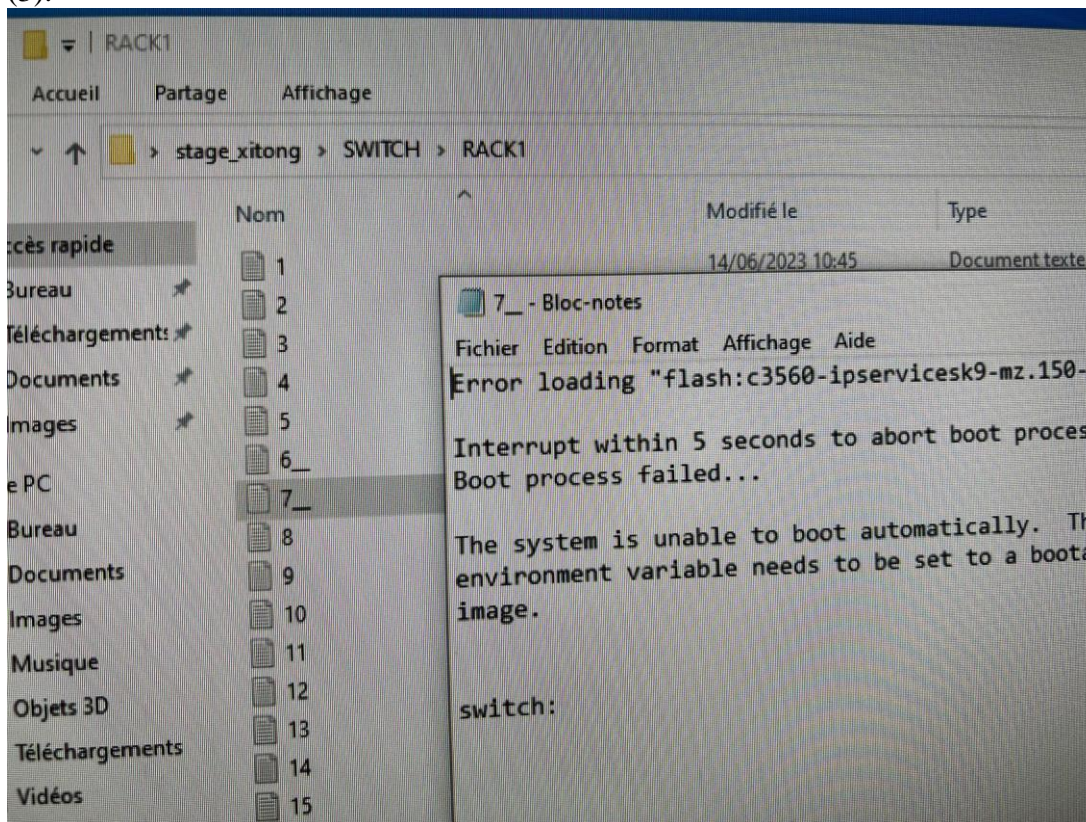


Figure 51 Commutateurs dont les fichiers système sont corrompus

Le dernier cas est également le plus typique. Lorsque vous voyez "switch:" s'afficher, cela indique généralement que vous êtes entré dans le BIOS du commutateur, ce qui signifie que le système IOS de Cisco lui-même est endommagé et ne peut pas démarrer, il entre donc automatiquement en mode BIOS.

Pour cette méthode, selon notre schéma, nous devons d'abord obtenir une copie intacte du fichier du système IOS, qui doit être compatible avec le modèle du commutateur. Nous avons deux options : la première consiste à utiliser un service TFTP pour obtenir le fichier du système IOS à partir d'un commutateur en bon état ayant le même fichier ou à partir d'un fichier IOS identique. L'autre option consiste à utiliser la commande "show version" pour vérifier le nom du fichier, puis à effectuer une recherche sur Internet pour télécharger le même système.

Index of /Downloads/Cisco/IOS/Catalyst 3560V2/

../		
c3560-ipbasek9-mz.122-55.SE12.bin	01-Feb-2022 19:55	11775204
c3560-ipbasek9-tar.122-55.SE12.tar	01-Feb-2022 19:55	15656960
c3560-ipservicesk9-mz.122-55.SE12.bin	01-Feb-2022 19:55	12745998
c3560-ipservicesk9-tar.122-55.SE12.tar	01-Feb-2022 19:55	16629760

Figure 52 Sites où vous pouvez télécharger IOS

Une fois que vous êtes dans l'interface BIOS du commutateur, la première étape consiste à utiliser la commande "dir flash:" pour afficher les fichiers présents dans la mémoire flash du commutateur. Ensuite, vous pouvez utiliser la commande "copy flash:nom_du_fichier tftp:adresse_IP_du_serveur" pour transférer le bon fichier système (IOS) vers un serveur (PC) via TFTP.

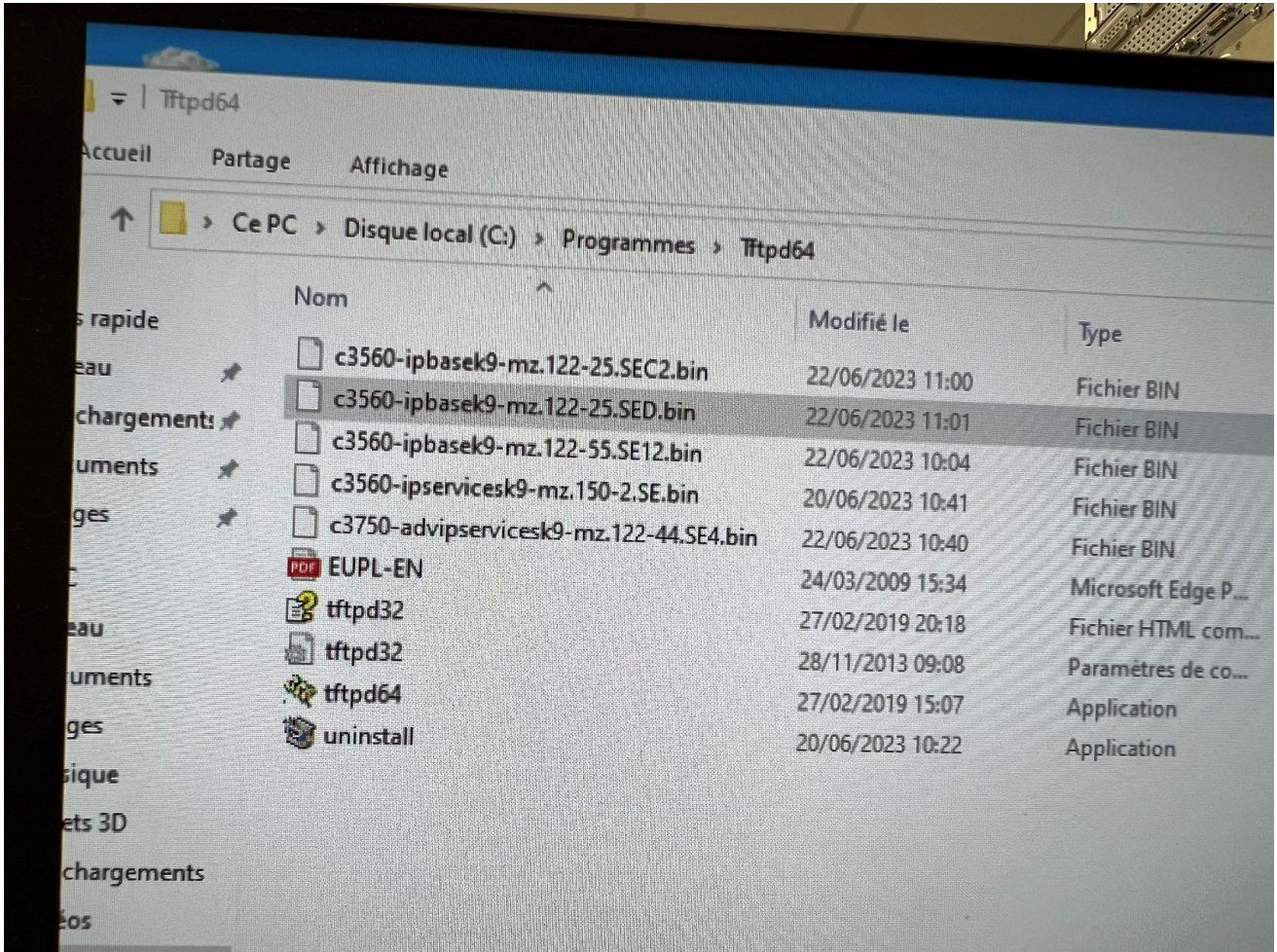


Figure 53 Bibliothèque système pour les serveurs TFTP

Une fois que vous avez téléchargé le bon fichier système, vous pouvez le stocker dans un répertoire dédié du serveur TFTP pour faciliter les étapes ultérieures de téléchargement.

```
switch: flash_init
Initializing Flash...
flashfs[0]: 1 files, 1 directories
flashfs[0]: 0 orphaned files, 0 orphaned directories
flashfs[0]: Total bytes: 15998976
flashfs[0]: Bytes used: 12474880
flashfs[0]: Bytes available: 3524096
flashfs[0]: flashfs fsck took 10 seconds.
...done Initializing Flash.

switch:
```

```
switch: set BAUD 115200
```

```
switch: copy xmodem: flash:c3560-ip-servicesk9-mz.122-53.SE.bin
Begin the Xmodem or Xmodem-1K transfer now...
CCC
Starting xmodem transfer. Press Ctrl+C to cancel.
Transferring c3560-ip-servicesk9-mz.122-53.SE.bin...
 100% 12181 KB 6 KB/s 00:31:56 0 Errors
```


Conclusion

En conclusion, bien que mon stage de dix semaines ait été court, il m'a apporté une expérience précieuse. Tout d'abord, j'ai eu l'opportunité d'apprendre de nouvelles connaissances et compétences. Ensuite, j'ai acquis une première expérience professionnelle dans mon domaine. Troisièmement, cela m'a permis de me familiariser avec l'environnement de travail, ce qui me semblait auparavant lointain et flou.

À l'IUT, j'ai acquis des connaissances en CISCO, SQL, HTML et CSS, et lors de mon stage, j'ai pu les appliquer et renforcer ces compétences. J'ai également approfondi ma compréhension du langage de programmation JavaScript, qui s'est avéré plus difficile que les exercices pratiques en classe. Cependant, cela m'a donné l'occasion de faire mes propres recherches et de poser des questions à mon tuteur de stage pour en apprendre davantage.

Dans l'ensemble, je suis satisfait de mes réalisations et de l'environnement de mon stage. J'ai participé à divers projets tels que le développement d'applications web, l'installation de nouvelles salles informatiques et la maintenance des équipements réseau. Pendant mon stage, j'ai pu développer des bases solides dans le domaine de l'informatique.

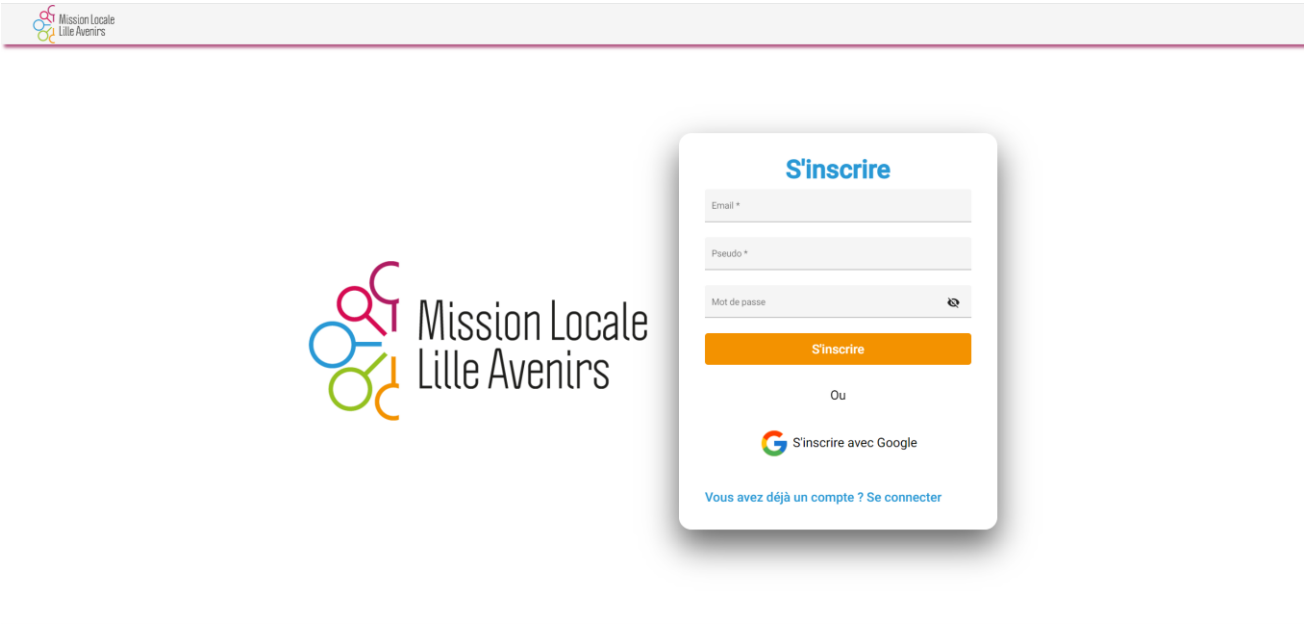
Remerciements

Je tiens à remercier toutes les personnes qui ont contribué à la réussite de mon stage et qui m'ont aidé. Tout d'abord, je tiens à remercier tous les membres de l'équipe d'immersize, Nainia Anas et mon chef Lorenzo Lavaux, de m'avoir accueilli dans l'entreprise. Je les remercie également pour leur patience, leur confiance et leur expertise au quotidien.

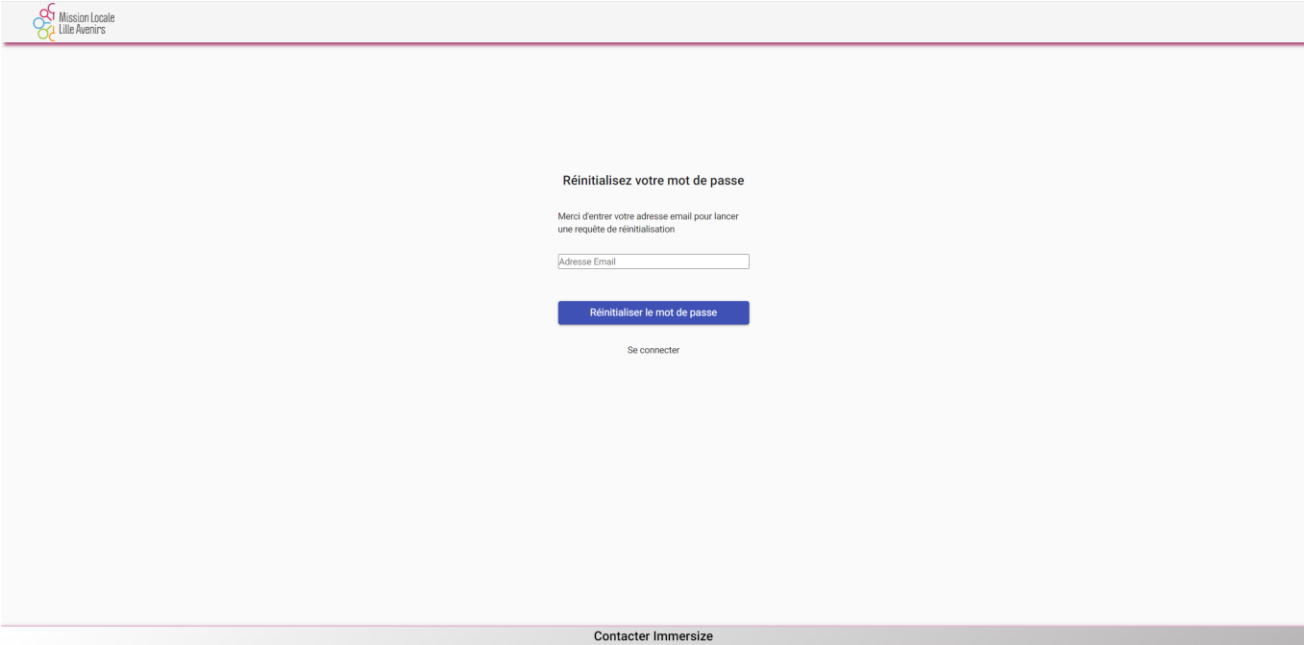
Ensuite, j'adresse mes remerciements à Mme Corinne HOUSSAIN et M. Jean Luc Damoiseaux de l'IUT de MARSEILLE LUMINY, qui m'ont beaucoup aidé dans la recherche et la candidature de stage. Leurs conseils m'ont permis de trouver un stage qui correspondait parfaitement à mes attentes, et ils ont fourni des conseils précieux pour l'avenir.

Enfin, je tiens à exprimer ma gratitude envers mon tuteur, M. Nguyen cong Tin, qui est mon enseignant à l'IUT de Marseille et responsable de la supervision de mes missions pendant mon stage.

Illustration



Page de la Connexion



Forgot-password



Se connecter


Email *

Merci de renseigner votre adresse email

Mot de passe *

Connexion

Ou

 Se connecter avec Google

[Mot de passe oublié ?](#) [S'inscrire](#)

Contacteur Immersize

Se connecter

Mission Locale Lille Avenir

李溪桐

Lancer le jeu

Distribution Distribution Distribution des joueurs dans les groupes	Lancer le jeu Lancer le jeu Démarrer le jeu
Réinitialiser un groupe Vider Sélectionnez le groupe que vous souhaitez réinitialiser	Réinitialiser le jeu Réinitialiser le jeu Cliquez pour réinitialiser le jeu

Contacteur Immersize

Jeux

Mission Locale Lille Avenir

李溪桐

Ajouter Joueur

Filtre

Prénom	Nom	Email	Groupe	Niveau	Modifier	Supprimer
LORENZO	Prince	immersize@gmail.com	A	1		
	testgroup	test@testt.com	A			
Loki	Laufeyson	loki@asgard.com	A	2		
Nainia	Allen	anas.nainia1@gmail.com	A	1		
Eddie	Brock	vnom@gmail.com	A			
XITONG	LI	lxitong1209@gmail.com	A	1		
Bruce	Wayne	notBatman@alfred.com	A	1		
Clark	Kent	supe@dailypianet.com	D	1		
Martin	Luther King	ihaveadream@gmail.com	D	1		
Thor	Odinson	mjobtir@midgard.com	D	1		
	swann	swanntest@hotmail.fr	D			
Swann	Swann	swannanas@gmail.com	B			
John	Carmack	johnc@gmail.com	B			
XITONG	LI	13241563418@123.com	B	1		
Michael	Brock	michael@asgard.com	D			

Contacteur Immersize

Table des users

Mission Locale Lille Avenir

李溪桐

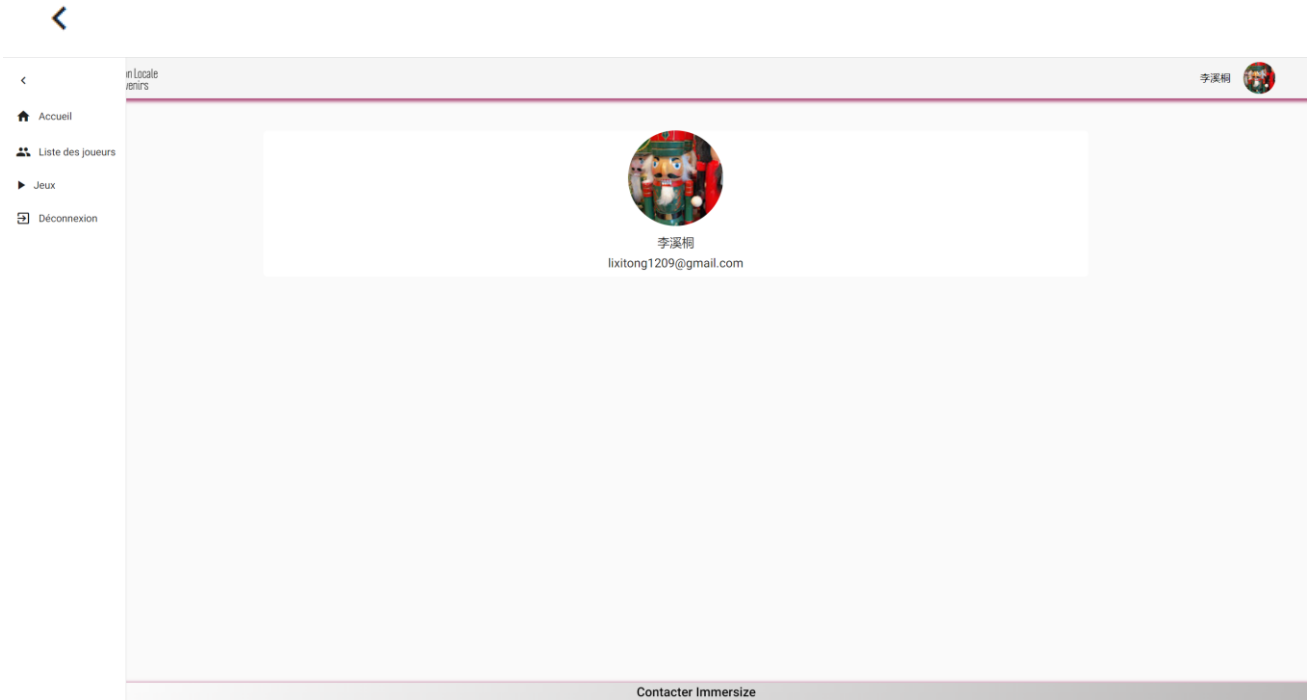
lundi 19 juin 2023

Liste des groupes actifs

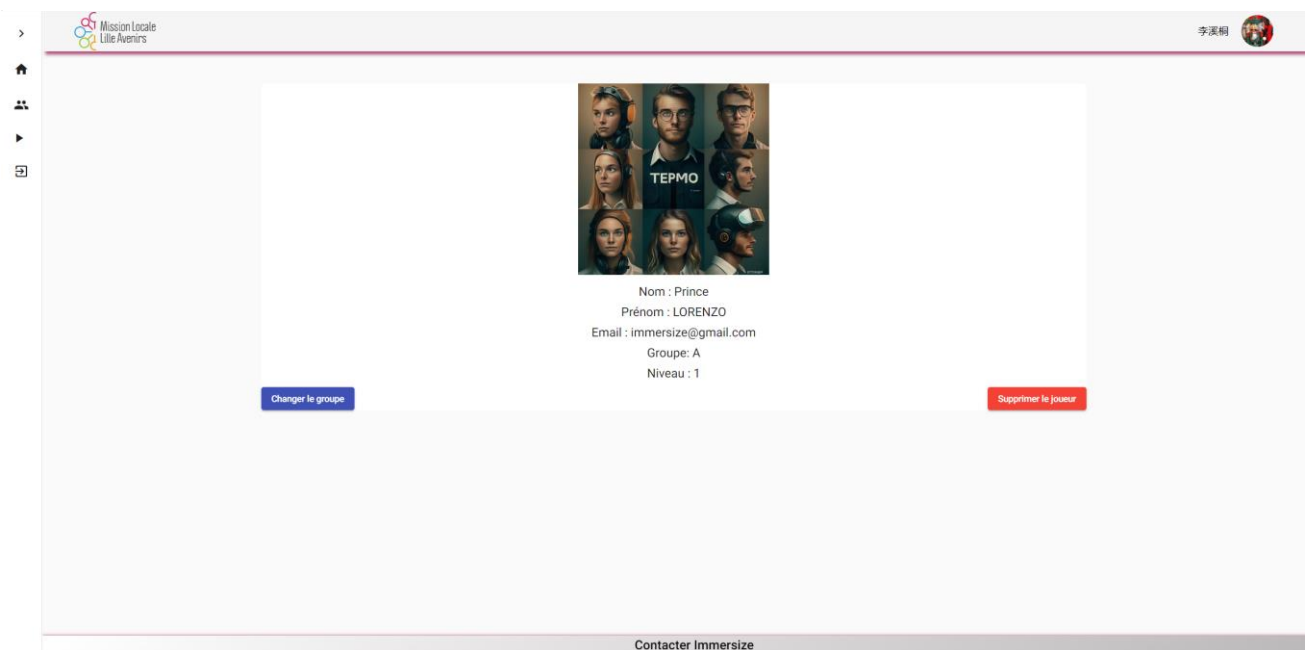
A	B	C	D	E			
Nom	Prénom	Niveau	Tel	Clicks	Modifier	Supprimer	
Prince	LORENZO	1	0691094404				
testgroup							
Laufeyson	Loki	2	0601090379				
Allen	Nainia	1	0719781874				
Brock	Eddie		0601011301				
LI	XITONG	1					
Wayne	Bruce	1	0606060606				

Contacteur Immersize

Table des participants



Page de profil



Information sur les joueurs

Glossaire

HTML (HyperText Markup Language) - Langage de balisage utilisé pour créer et structurer le contenu des pages Web.

CSS (Cascading Style Sheets) - Langage de feuille de style utilisé pour styliser et mettre en forme les éléments HTML d'une page Web.

SQL (Structured Query Language) - Langage de requête structuré utilisé pour gérer et manipuler les bases de données relationnelles.

SPA (Single Page Application) - Application monopage, qui est une application Web qui se charge une seule fois et met à jour dynamiquement son contenu sans recharger la page entière.

BDD (Base de Données) - Système de gestion de données permettant de stocker, organiser et gérer des informations de manière structurée et accessible.

PC (Personal Computer) - Ordinateur personnel, se réfère généralement à un ordinateur de bureau ou portable utilisé par un individu.

IOS - Système d'exploitation mobile développé par Apple pour ses appareils tels que l'iPhone et l'iPad.

TFTP (Trivial File Transfer Protocol) - Protocole de transfert de fichiers simple utilisé pour transférer des fichiers entre des appareils sur un réseau.

Bibliographies

OpenClassrooms - Débutez avec Angular (<https://openclassrooms.com/en/courses/7471261-debutez-avec-angular>) Dernière mise à jour le 13 juin 2022 par Will Alexander.

MDN Web Docs - Angular Getting Started ([https://developer.mozilla.org/zh-CN/docs/Learn/Tools_and_testing/Client-side JavaScript frameworks/Angular getting started](https://developer.mozilla.org/zh-CN/docs/Learn/Tools_and_testing/Client-side_JavaScript_frameworks/Angular_getting_started)) Consulté depuis le 20 mai.

Runoob - AngularJS Tutorial (<https://www.runoob.com/angularjs/angularjs-tutorial.html>) Consulté depuis le 29 avril.

Wikipédia - Angular (<https://fr.wikipedia.org/wiki/Angular>) Consulté depuis le 18 avril.

Firebase Documentation - Hosting and Functions (<https://firebase.google.com/docs/hosting/functions>) Dernière mise à jour le 16 juin 2023, UTC.

Cisco Support Documentation - Catalyst 2950 Series Switches (<https://www.cisco.com/c/en/us/support/docs/switches/catalyst-2950-series-switches/41845-192.html>) Dernière mise à jour le 18 juillet 2022.

Free CCNA Workbook - Recovering a Corrupt Cisco IOS Image on a Catalyst Switch (<https://www.freeccnaworkbook.com/workbooks/ccna/recovering-a-corrupt-cisco-ios-image-on-a-catalyst-switch>) Consulté depuis le 21 juin.

Réinitialiser le mot de passe du commutateur (<https://blog.51cto.com/woyaoxuelinux/1972612>) Consulté depuis le 20 avril.

MDN Web Docs - JavaScript (<https://developer.mozilla.org/fr/docs/Web/JavaScript>) Consulté depuis le 25 avril.