

C6N2_Trvs. Présenter un travail réalisé en synthétisant ses résultats, sa démarche

Une bonne présentation est susceptible de faire valider la compétence sachant qu'il faut assurer les deux types de présentation Client et Technique en français.

Présentation client en français :

Bonjour, permettez-moi de vous présenter ... Grace a cette application, vous pourrez écouter et organiser votre musique locale. Cette solution vous permettra de gérer vos tracks (ajouter, modifier, supprimer), de les organiser par catégorie musicale, et de profiter d'un lecteur audio complet avec contrôles de lecture, volume et progression

Présentation technique en français :

Pour réaliser cette application :

- J'ai utilisé Angular (versionX) avec TypeScript
- J'ai suivi une architecture modulaire avec lazy loading
- J'ai utilisé RxJS/Signals pour la gestion d'état réactive
- Cote persistance, j'ai travaillé avec ... pour stocker les fichiers audio
- J'ai implémenté des services : AudioPlayerService, TrackService, StorageService
- J'ai utilisé Reactive Forms avec validations

C8N2_Trvs. Interagir dans un contexte professionnel de façon respectueuse et constructive

Un bon comportement professionnel est exigé

C4N2_Trvs. Rechercher de façon méthodique une ou des solutions + C3N2_Trvs. Définir le périmètre d'un problème

Validé ou non selon votre manière de résolution d'un blocage lors de la Mise en situation

S'il n'y a pas de blocage rencontré à la base lors de la mise en situation c'est déjà valide

C1N2_Trvs. Planifier le travail à effectuer individuellement

Une bonne planification JIRA avec démarche SCRUM doit être correctement faite (User Stories, Sprints, Backlogs)

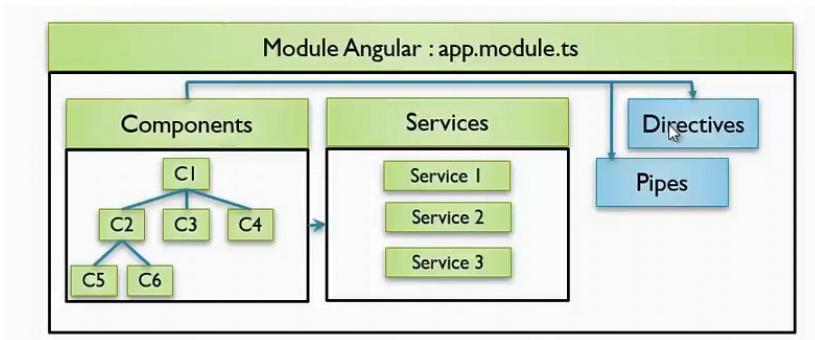
C2N2_Trvs. Contribuer au pilotage de l'organisation du travail individuel et collectif

Une bonne organisation du temps et du travail pour valider cette compétence

C1N2_Tech. Installer et configurer son environnement de travail en fonction du projet

Angular Basics

- Qu'est-ce qu'Angular et comment fonctionne-t-il ?
- Quels sont les prérequis pour travailler avec Angular ? (Node.js → npm → Angular CLI → ng)
- Combien de Framework Angular y en a-t-il ?
- Quels sont les avantages d'utiliser Angular pour le développement d'applications web ?
- Construisez un paragraphe contenant : Angular, TypeScript, Google, SPA, Architecture, JS
- Quelles sont les étapes à suivre pour créer un nouveau projet Angular ?
- C'est quoi la structure d'un projet Angular ?



Component et Cycle de vie

- C'est quoi le cycle de vie d'un component en Angular ?
- Expliquez les hooks : ngOnInit, ngOnChanges, ngOnDestroy, ngAfterViewInit

TypeScript vs Java (POO)

Remplir le tableau comparatif POO Java vs TypeScript :

Concept POO	Java	TypeScript
	public class User { }	class User { }
Constructeur		constructor(name: string) {}
Héritage	class Admin extends User { }	
Interface		interface UserInterface { }
	private, protected, public	private, protected, public
	public getName() { return name; }	get name() { return this._name; }
Classes abstraites	abstract class Shape { }	abstract class Shape { }
Méthodes abstraites		abstract draw(): void;
	class User implements UserInterface	class User implements UserInterface
Encapsulation	private String name;	private name: string;
	Oui, via héritage et interface	Oui, via héritage et interface
Types génériques	List<String>	Array<string>
	Annotations (@Override)	Decorators (@Component)

C2N2_Tech. Développer des interfaces utilisateur

Formulaires

- Quelles sont les deux principales méthodes pour créer des formulaires en Angular ?
- Quelle est la différence entre Template-driven Forms et Reactive Forms ?

C3N2_Tech. Développer des composants métier

La majorité des fonctionnalités métier doivent être fonctionnelles et logiquement correctes

RxJS

- C'est quoi RxJS ?
- Quelles sont ses principales méthodes ? (map, filter, throttleTime, scan, reduce)
- C'est quoi un Observable ? Un Subject ? Un BehaviorSubject ?
- C'est quoi l'Async Pipe ?

Signals

- C'est quoi les Signals en Angular ?
- Différence entre Signals et BehaviorSubject ?

Services

- C'est quoi l'injection de dépendances en Angular ?

C4N2_Tech. Contribuer à la gestion d'un projet

- Il doit y avoir une bonne stratégie de branches Git
- Des commits bien faits
- Un bon README

C5N2_Tech. Analyser les besoins et maquetter une application

Un bon design de l'application est exigé

C6N2_Tech. Définir l'architecture logicielle d'une application

- Il doit y avoir une bonne séparation des modules
- Il doit y avoir une bonne utilisation du lazy loading

Data Binding

- C'est quoi Databinding et quels sont ses types ?
- Différence entre : interpolation {{ }}, property binding [], event binding (), two-way binding [()]

Exemples de Data Binding :

```
about.component.ts
export class AboutComponent {
  info={ nom:"Mohamed",
    email:"med@yousifi.net",
    tel:"0661326837"
  };
  comments=[];
  comment={date:null,message:''};
  newComment=false;
  addComment () {
    if(this.comment.message!=""){
      this.comment.date=new Date();
      this.comments.push({
        date:this.comment.date,
        message:this.comment.message
      });
      this.comment.message='';
    }
  }
}

about.component.html
<ul>
  <li>Nom : {{info.nom}}</li>
  <li>Email : {{info.email}}</li>
  <li>Tel : {{info.tel}}</li>
</ul>
<div>
  <input type="text" [(ngModel)]="comment.message" >
  <button (click)="addComment()" [disabled]="newComment">
    Add comment
  </button>
</div>
<div *ngIf="comments.length>0; else noComments">
  <h3>Liste des messages :</h3>
  <ul>
    <li *ngFor="let c of comments">
      {{c.date}} : {{c.message}}
    </li>
  </ul>
</div>
<ng-template #noComments>
  <p> Liste des commentaires est vide</p>
</ng-template>
```

Routing

- C'est quoi le routage (routing) ? Comment le déclarer ?
- Quels sont les types de routage ?
- C'est quoi la différence entre : <router-outlet>, routerLink, this.router.navigate()

Stockage côté client

- Différence entre localStorage, sessionStorage et IndexedDB ?
- Pourquoi utiliser IndexedDB pour les fichiers audio ?

Abréviations

SPA, RxJS, DOM, CLI, NPM