

Interface Directeur de Thèse - Documentation Complète

Table des matières

1. Rôle du Directeur de Thèse
 2. Modules de l'Interface
 3. Architecture Technique
 4. Guides d'Implémentation
-

1. RÔLE DU DIRECTEUR DE THÈSE {#role}

Définition (selon CDC - page 3)

┆ "Suit le statut du doctorant. Signe les documents nécessaires. Propose les membres du jury."

Responsabilités fonctionnelles

Fonction	Description
Validation des dossiers	Reçoit une notification lorsque le doctorant soumet un dossier d'inscription ou de réinscription. Doit donner son avis (validation ou rejet) dans le circuit de validation en ligne.
Suivi du statut	A accès à un tableau de bord pour suivre l'état d'avancement du doctorant.
Proposition du jury	Dans le module "Processus de soutenance", propose les membres du jury : rapporteurs, examinateurs, président.
Signature électronique	Signe électroniquement les documents nécessaires tout au long du parcours doctoral.
Vérification des prérequis	Participe à la validation des conditions préalables à la soumission d'une demande de soutenance (publications, formations, etc.).
Communication	Interagit via le portail avec les services administratifs pour la planification et l'autorisation de soutenance.

2. MODULES DE L'INTERFACE {#modules}

MODULE 1 : GESTION DES CANDIDATURES

 **Page : "Candidatures en attente de ma validation"**

État ciblé : EN_ATTENTE_DIRECTEUR (déjà validées par l'admin)

Affichage de la liste :




- Nom complet du candidat
- Matricule / CNIE
- Email
- Date de soumission
- État actuel

Vue détaillée (après clic) :



INFORMATIONS PERSONNELLES

- Nom complet
- Matricule / CNIE
- Email
- Téléphone
- Sujet de thèse proposé
- Collaboration (si applicable)

DOCUMENTS JUSTIFICATIFS

-  Curriculum Vitae [Consulter] [Télécharger]
-  Diplôme [Consulter] [Télécharger]
-  Lettre de Motivation [Consulter] [Télécharger]

DÉCISION

- [ Valider la candidature]
- [ Refuser la candidature]
- Si refus → Champ obligatoire "Motif du refus"

Actions disponibles :

1. Valider la candidature

- Le candidat passe de **EN_ATTENTE_DIRECTEUR** → **VALIDE**
- Le candidat devient officiellement **DOCTORANT**
- Notifications envoyées :
 - Au candidat : "Félicitations, votre candidature a été acceptée"
 - À l'administration : "Nouvelle inscription validée"

2. Refuser la candidature

- Le candidat passe à l'état **REFUSE**
- Champ obligatoire :** Motif du refus (minimum 10 caractères)

- Le motif est affiché au candidat dans sa page d'attente
 - Notifications envoyées :
 - Au candidat : "Votre candidature a été refusée" + motif
 - À l'administration : "Candidature refusée par le directeur"
-

MODULE 2 : MES DOCTORANTS

 Page : "Tableau de bord de mes doctorants"

Liste tous les doctorants avec état VALIDE encadrés par le directeur.

Informations affichées :

 MES DOCTORANTS (5 actifs)

 BENNANI Ahmed

 Année d'inscription : 2/6

 Sujet : Intelligence Artificielle et IoT

PROGRESSION DES PRÉREQUIS :

└ Publications (Q1/Q2) : 1/2 

└ Conférences : 2/2 

└ Heures de formation : 150h/200h 

[ Voir le détail] [ Contacter]

 ALAMI Fatima

 Année d'inscription : 5/6  ALERTE

 Sujet : Blockchain et Sécurité

PROGRESSION DES PRÉREQUIS :

└ Publications (Q1/Q2) : 2/2 

└ Conférences : 3/2 

└ Heures de formation : 200h/200h 

 PRÊT POUR SOUTENANCE

[ Initier demande de soutenance]

Alertes automatiques :

- **Alerte jaune** : Doctorant en année 3/6 (fin de durée normale)
- **Alerte rouge** : Doctorant en année 5/6 ou 6/6 (limite maximale approchée/atteinte)
- ⚠ **Alerte prérequis** : Doctorant en retard sur les publications/formations

Actions disponibles :

- Consulter le détail du parcours

- Voir l'historique des réinscriptions
 - Accéder aux rapports de progrès
 - Initier une demande de soutenance
-

MODULE 3 : PROCESSUS DE SOUTENANCE

 **Page : "Demandes de soutenance"**

Étape 1 : Validation des prérequis

Lorsqu'un doctorant soumet une demande de soutenance, le directeur doit vérifier :

<div> DEMANDE DE SOUTENANCE - Ahmed BENNANI</div>	
<div><input checked="" type="checkbox"/> VÉRIFICATION DES PRÉREQUIS</div>	
Publications scientifiques :	
<div><input checked="" type="checkbox"/> Article Q1 - "AI in IoT Systems" - IEEE 2024</div>	
<div><input checked="" type="checkbox"/> Article Q2 - "Machine Learning..." - Springer 2024</div>	
<div><input checked="" type="checkbox"/> Condition remplie : 2/2</div>	
Conférences :	
<div><input checked="" type="checkbox"/> ICAI 2024 - Paris</div>	
<div><input checked="" type="checkbox"/> IEEE Conference - Dubai</div>	
<div><input checked="" type="checkbox"/> Condition remplie : 2/2</div>	
Formation doctorale :	
<div><input checked="" type="checkbox"/> 200h complétées</div>	
Documents fournis :	
<div><input checked="" type="checkbox"/> Manuscrit de thèse</div>	
<div><input checked="" type="checkbox"/> Rapport anti-plagiat (< 10%)</div>	
<div><input checked="" type="checkbox"/> Rapport des publications</div>	
<div><input checked="" type="checkbox"/> Attestations de formation</div>	
<div><input checked="" type="checkbox"/> Demande manuscrite</div>	
<div> Commentaire du directeur (optionnel) :</div> <div></div>	
<div><input checked="" type="checkbox"/> Donner un avis favorable</div>	
<div><input checked="" type="checkbox"/> Donner un avis défavorable</div>	

Étape 2 : Proposition du jury

Une fois l'avis favorable donné, le directeur propose la composition du jury :



PROPOSITION DU JURY DE SOUTENANCE

PRÉSIDENT DU JURY :

[+ Ajouter le président]

RAPPORTEURS (Minimum 2) :

Pr. Mohammed TAZI



Grade : Professeur HDR



Établissement : Université de Rabat

Email : m.tazi@um5.ac.ma

[+ Ajouter un rapporteur]

EXAMINATEURS :

Dr. Amina SERRAR



Grade : Maître de conférences HDR



Établissement : ENSAM Casablanca

Email : a.serrar@ensam.ma

[+ Ajouter un examinateur]

[Enregistrer la proposition]

[Soumettre à l'administration]

Formulaire d'ajout de membre du jury :

Nom : [_____]

Prénom : [_____]

Grade : [▼ Professeur / Maître de conférences HDR / ...]

Établissement : [_____]

Email : [_____]

Rôle : [▼ Rapporteur / Examineur / Président]

Étape 3 : Signature électronique

Une fois l'autorisation de soutenance émise par l'administration :



SIGNATURE ÉLECTRONIQUE

Document : Autorisation de soutenance

Doctorant : Ahmed BENNANI

Date de soutenance : 15 Juin 2025

Lieu : Amphithéâtre A - ENSAM Casablanca

[ Consulter le document]

[ Signer électroniquement]

MODULE 4 : RÉINSCRIPTIONS ANNUELLES



Page : "Validation des réinscriptions"

Chaque année, les doctorants doivent se réinscrire. Le directeur valide ces réinscriptions.

DEMANDES DE RÉINSCRIPTION (3)

 BENNANI Ahmed

Réinscription pour l'année : 2025-2026

Année en cours : 2/6

Date de première inscription : Sept 2023

Progression depuis la dernière inscription :

└─ 1 nouvelle publication

└─ 1 conférence internationale

└─ 50h de formation supplémentaires

[ Valider la réinscription]

[ Refuser] (avec motif obligatoire)

 EL AMRANI Youssef

Réinscription pour l'année : 2025-2026

Année en cours : 4/6  DÉROGATION NÉCESSAIRE

Date de première inscription : Sept 2021




 Ce doctorant dépasse la durée normale de 3 ans

Une dérogation du PED (Pôle d'Études Doctorale)

est requise pour poursuivre.

[ Demander une dérogation au PED]








Règles de gestion :

-  Années 1-3 : Validation simple
-  Années 4-5 : Dérogation du PED obligatoire
-  Année 6+ : Limite maximale atteinte, réinscription impossible

MODULE 5 : NOTIFICATIONS ET COMMUNICATION

🔔 Centre de notifications

Types de notifications reçues :

Type	Description	Action
 Nouvelle candidature	Un candidat a été validé par l'admin	Consulter et valider
 Demande de réinscription	Un doctorant demande sa réinscription	Valider/Refuser
 Demande de soutenance	Un doctorant soumet sa demande	Vérifier prérequis + Proposer jury
 Rapport de rapporteur reçu	Un rapporteur a soumis son rapport	Consulter le rapport
 Autorisation accordée	L'admin a autorisé la soutenance	Signer le document
 Soutenance planifiée	Date et lieu fixés	Information
 Alerte durée	Un doctorant approche des limites	Attention requise

Interface du centre de notifications :

🔔 NOTIFICATIONS (8)

[Tout marquer lu]

NEW

Nouvelle candidature

[Non lu]

Sarah BENALI a été validée par l'administration

Il y a 2 heures

[Consulter le dossier]

🎓

Demande de soutenance

[Non lu]

Ahmed BENNANI demande à soutenir sa thèse

Il y a 5 heures

[Vérifier les prérequis]

⚠️

Alerte durée

[Lu]

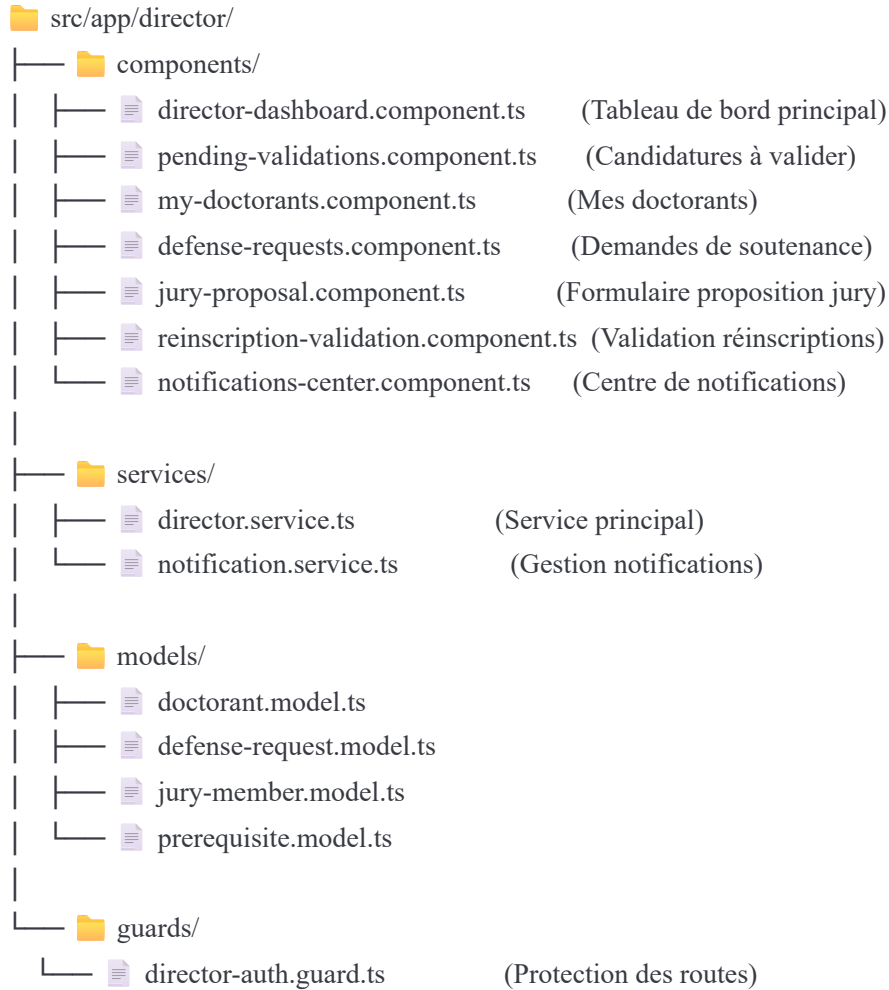
Fatima ALAMI entre en 5ème année (limite proche)

Il y a 1 jour

[Voir le dossier]

3. ARCHITECTURE TECHNIQUE {#architecture}

Structure des composants Angular



4. GUIDES D'IMPLÉMENTATION {#implementation}

A. Models TypeScript

doctorant.model.ts

typescript

```
export interface Doctorant {  
  id: number;  
  nom: string;  
  prenom: string;  
  username: string; // Matricule  
  email: string;  
  telephone: string;  
  
  // Dates importantes  
  dateInscription: Date;  
  anneeEnCours: number; // 1 à 6  
  
  // Sujet de thèse  
  sujetThese: string;  
  collaboration?: string;  
  
  // Prérequis  
  prerequis: Prerequisites;  
  
  // État  
  etat: 'EN_ATTENTE_DIRECTEUR' | 'VALIDE' | 'REFUSE';  
  motifRefus?: string;  
}
```

```
export interface Prerequisites {  
  publications: Publication[];  
  conferences: Conference[];  
  formationHours: number;  
}
```

```
export interface Publication {  
  titre: string;  
  type: 'Q1' | 'Q2' | 'Q3' | 'Q4';  
  annee: number;  
  revue: string;  
}
```

```
export interface Conference {  
  nom: string;  
  lieu: string;  
  annee: number;
```

```
type: 'INTERNATIONALE' | 'NATIONALE';  
}
```

defense-request.model.ts

typescript

```
export interface DefenseRequest {  
  id: number;  
  doctorant: Doctorant;  
  dateSubmission: Date;  
  
  // Documents  
  manuscrit: string; // URL  
  rapportAntiPlagiat: string; // URL  
  rapportPublications: string; // URL  
  attestationsFormation: string[]; // URLs  
  demandeManuscrite: string; // URL  
  
  // Statut  
  status: 'EN_ATTENTE_DIRECTEUR' | 'JURY_PROPOSE' | 'EN_ATTENTE_RAPPORTEURS' | 'AUTORISE' | 'REFUSE';  
  
  // Avis du directeur  
  avisDirecteur?: 'FAVORABLE' | 'DEFAVORABLE';  
  commentaireDirecteur?: string;  
  
  // Jury proposé  
  jury?: JuryMember[];  
}
```

jury-member.model.ts

typescript

```
export interface JuryMember {  
  id?: number;  
  nom: string;  
  prenom: string;  
  grade: 'PROFESSEUR' | 'MCF_HDR' | 'MCF' | 'DIRECTEUR_RECHERCHE';  
  etablissement: string;  
  email: string;  
  role: 'PRESIDENT' | 'RAPPORTEUR' | 'EXAMINATEUR';  
  
  // Statut du rapport (pour rapporteurs)  
  rapportSoumis?: boolean;  
  avisRapport?: 'FAVORABLE' | 'DEFAVORABLE' | 'RESERVE';  
}
```

B. Service Principal

director.service.ts

typescript

```

import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';
import { Observable } from 'rxjs';

@Injectable({
  providedIn: 'root'
})
export class DirectorService {
  private apiUrl = 'http://localhost:8080/api/director';

  constructor(private http: HttpClient) {}

  // ===== CANDIDATURES =====
  getCandidaturesEnAttente(): Observable<Doctorant[]> {
    return this.http.get<Doctorant[]>(`${this.apiUrl}/candidatures/pending`);
  }

  validerCandidature(userId: number): Observable<any> {
    return this.http.post(`${this.apiUrl}/candidatures/${userId}/validate`, {});
  }

  refuserCandidature(userId: number, motif: string): Observable<any> {
    return this.http.post(`${this.apiUrl}/candidatures/${userId}/reject`, { motif });
  }

  // ===== MES DOCTORANTS =====
  getMesDoctorants(): Observable<Doctorant[]> {
    return this.http.get<Doctorant[]>(`${this.apiUrl}/doctorants`);
  }

  getDoctorantDetail(id: number): Observable<Doctorant> {
    return this.http.get<Doctorant>(`${this.apiUrl}/doctorants/${id}`);
  }

  // ===== RÉINSCRIPTIONS =====
  getReinscriptionsEnAttente(): Observable<any[]> {
    return this.http.get<any[]>(`${this.apiUrl}/reinscriptions/pending`);
  }

  validerReinscription(reinscriptionId: number): Observable<any> {
    return this.http.post(`${this.apiUrl}/reinscriptions/${reinscriptionId}/validate`, {});
  }
}

```



```

refuserReinscription(reinscriptionId: number, motif: string): Observable<any> {
    return this.http.post(`${this.apiUrl}/reinscriptions/${reinscriptionId}/reject`, { motif });
}

dementerDerogation(reinscriptionId: number, justification: string): Observable<any> {
    return this.http.post(`${this.apiUrl}/reinscriptions/${reinscriptionId}/derogation`, { justification });
}

// ===== SOUTENANCES =====
getDemandesSoutenance(): Observable<DefenseRequest[]> {
    return this.http.get<DefenseRequest[]>(`${this.apiUrl}/defenses/pending`);
}

donnerAvisSoutenance(defenseId: number, avis: 'FAVORABLE' | 'DEFAVORABLE', commentaire?: string): Observable<any> {
    return this.http.post(`${this.apiUrl}/defenses/${defenseId}/avis`, { avis, commentaire });
}

proposerJury(defenseId: number, juryMembers: JuryMember[]): Observable<any> {
    return this.http.post(`${this.apiUrl}/defenses/${defenseId}/jury`, { members: juryMembers });
}

signerDocument(documentId: number): Observable<any> {
    return this.http.post(`${this.apiUrl}/documents/${documentId}/sign`, {});
}

// ===== STATISTIQUES =====
getStatistiques(): Observable<any> {
    return this.http.get(`${this.apiUrl}/statistics`);
}
}

```

C. Composant Principal : Candidatures en Attente

pending-validations.component.ts

typescript

```
import { Component, OnInit, signal } from '@angular/core';
import { CommonModule } from '@angular/common';
import { FormsModule } from '@angular/forms';
import { DirectorService } from '@director/services/director.service';
import { Doctorant } from '@director/models/doctorant.model';
```

```
@Component({
  selector: 'app-pending-validations',
  standalone: true,
  imports: [CommonModule, FormsModule],
  templateUrl: './pending-validations.component.html',
  styleUrls: ['./pending-validations.component.css']
})
```

```
export class PendingValidationsComponent implements OnInit {
```

```
  candidatures = signal<Doctorant[]>([]);
  expandedCandidatureId = signal<number | null>(null);
  showRefusalInputId = signal<number | null>(null);
  motifRefus = "";
  isLoading = signal(false);
```

```
  constructor(private directorService: DirectorService) {}
```

```
  ngOnInit() {
    this.loadCandidatures();
  }
```

```
  loadCandidatures() {
    this.isLoading.set(true);
    this.directorService.getCandidaturesEnAttente().subscribe({
      next: (data) => {
        this.candidatures.set(data);
        this.isLoading.set(false);
      },
      error: (err) => {
        console.error('Erreur lors du chargement des candidatures:', err);
        this.isLoading.set(false);
      }
    });
  }
```

```
  toggleExpand(id: number) {
    if (this.expandedCandidatureId() === id) {
      this.expandedCandidatureId.set(null);
    }
  }
```

```
    } else {  
        this.expandedCandidatureId.set(id);  
        this.showRefusalInputId.set(null);  
        this.motifRefus = "";  
    }  
}
```

```
viewDocument(filename: string, event: Event) {  
    event.stopPropagation();  
    if (filename) {  
        window.open(`http://localhost:8080/api/documents/${filename}`, '_blank');  
    }  
}
```

```
confirmValidate(candidat: Doctorant, event: Event) {  
    event.stopPropagation();  
    if (confirm(`Confirmer la validation de ${candidat.nom} ${candidat.prenom} ?`)) {  
        this.directorService.validerCandidature(candidat.id).subscribe({  
            next: () => {  
                alert('Candidature validée avec succès !');  
                this.loadCandidatures();  
            },  
            error: (err) => console.error('Erreur:', err)  
        });  
    }  
}
```

```
initiateRefusal(id: number, event: Event) {  
    event.stopPropagation();  
    this.showRefusalInputId.set(id);  
    this.motifRefus = "";  
}
```

```
cancelRefusal(event: Event) {  
    event.stopPropagation();  
    this.showRefusalInputId.set(null);  
    this.motifRefus = "";  
}
```

```
confirmRefusal(candidat: Doctorant, event: Event) {  
    event.stopPropagation();  
    if (!this.motifRefus.trim()) {  
        alert('Veuillez saisir un motif de refus.');        return;  
    }  
}
```

```
}

if (confirm(`Refuser définitivement la candidature de ${candidat.nom} ?`)) {
  this.directorService.refuserCandidature(candidat.id, this.motifRefus.trim()).subscribe({
    next: () => {
      alert('Candidature refusée.');
      this.loadCandidatures();
      this.showRefusalInputId.set(null);
    },
    error: (err) => console.error('Erreur:', err)
  });
}
}
```

D. Routes Angular

app.routes.ts (extrait)

typescript

```
import { Routes } from '@angular/router';
import { DirectorAuthGuard } from '@director/guards/director-auth.guard';

export const routes: Routes = [
  // ... autres routes

  {
    path: 'director',
    canActivate: [DirectorAuthGuard],
    children: [
      {
        path: 'dashboard',
        loadComponent: () => import('@director/components/director-dashboard.component')
          .then(m => m.DirectorDashboardComponent)
      },
      {
        path: 'candidatures',
        loadComponent: () => import('@director/components/pending-validations.component')
          .then(m => m.PendingValidationsComponent)
      },
      {
        path: 'doctorants',
        loadComponent: () => import('@director/components/my-
```