

INSA DE LYON
DÉPARTEMENT INFORMATIQUE
DÉVELOPPEMENT D'APPLICATIONS POUR LES SI

TP DASI
INSTRUCT'IF

Auteurs :
Binôme B3423 :
Saad ELGHISSASSI
Simon PERRET

Professeurs :
Mme. TCHOOUNIKINE
Mme. SERVIGNE
M. DUCLOS
M. GRIPAY

Table des matières

Introduction	1
1 Description des cas d'usage	1
2 Modèle du domaine	2
3 Description des IHM	3
4 Spécification des services	9

Introduction

La plateforme INSTRUCT'IF se présente comme une solution numérique innovante destinée à faciliter l'aide aux devoirs pour les élèves de niveau collège et lycée. Ce projet incarne une plateforme dynamique facilitant la connexion entre des élèves en quête de soutien académique et des intervenants qualifiés prêts à offrir leur expertise bénévolement.

L'une des principales forces d'INSTRUCT'IF réside dans sa facilité d'utilisation, offrant aux élèves un moyen rapide et intuitif de demander de l'aide. En quelques clics, une demande de soutien est soumise et traitée aboutissant à l'assignation automatique d'un intervenant approprié. Un système de visioconférence intégré, assurant des sessions de tutorat interactives et adaptées aux besoins spécifiques de chaque élève, renforce l'accessibilité et l'efficacité des soutiens scolaires proposés par la plateforme, ce qui favorise une expérience d'apprentissage immersive et personnalisée, permettant aux élèves de surmonter leurs difficultés scolaires dans un environnement stimulant et sécurisant.

Ce rapport détaillera les fonctionnalités clés d'INSTRUCT'IF en mettant l'accent sur les processus qui ont permis sa conception.

1 Description des cas d'usage

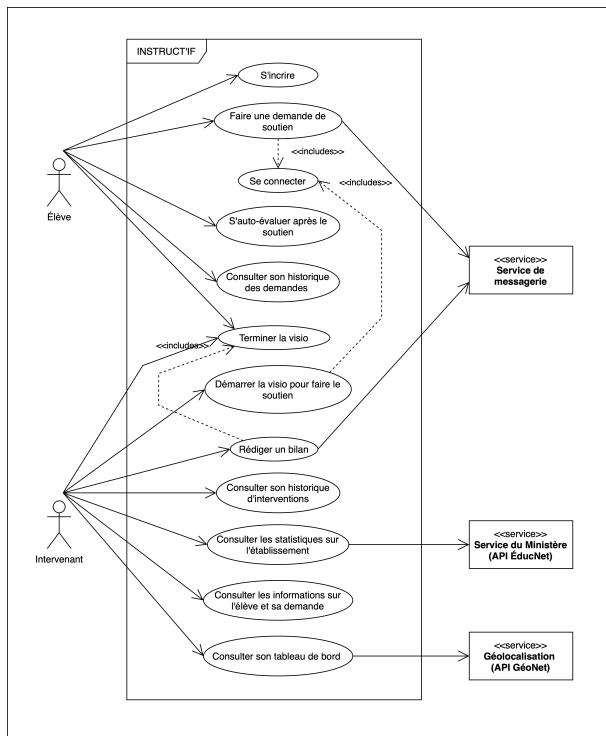


FIGURE 1 – Diagramme des cas d'utilisation de l'application Instruct'IF

1.1 Description du cahier des charges

Elève Sur INSTRUCT'IF, l'élève s'inscrit en indiquant ses données et le code de son établissement pour créer un profil. Après inscription, il se connecte pour une demande de soutien, choisissant une matière et exposant sa difficulté. La plateforme sélectionne alors l'intervenant approprié en fonction du niveau et de la disponibilité. En cas d'indisponibilité, un email invite à renouveler la demande. Sinon, la visioconférence démarre. L'élève évalue à l'issue du soutien la qualité de ce dernier, et reçoit un bilan par mail. Il a aussi accès à l'historique des sessions pour consulter les retours des intervenants.

Intervenant Les intervenants d'INSTRUCT'IF, étudiants, enseignants ou bénévoles, aident collégiens et lycéens. Inscrits sur la plate-forme, ils peuvent recevoir des demandes de soutien par SMS. Les intervenants sont sélectionnés par un algorithme favorisant ceux ayant le moins contribué. Les intervenants sont ouverts à toutes les matières, mais ils indiquent au préalable les classes des élèves qu'ils acceptent d'accompagner.

Après une session, ils rédigent un bilan et peuvent voir des statistiques sur leur tableau de bord, comme le nombre d'élèves, d'interventions et la satisfaction moyenne.

1.2 Diagramme des cas d'utilisation

Le diagramme de la figure 1 dépeint l'interface de la plateforme INSTRUCT'IF, détaillant les fonctionnalités pour les élèves et les intervenants. Les élèves s'enregistrent, demandent des soutiens, participent à des visioconférences et suivent leurs progrès. Les intervenants gèrent les demandes, historiques et rédigent des bilans. Des API externes enrichissent la plateforme avec des services de messagerie et de géolocalisation.

2 Modèle du domaine

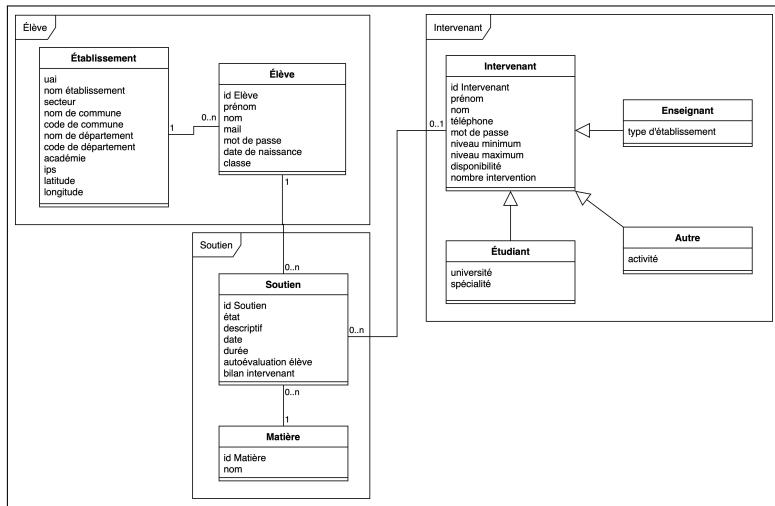


FIGURE 2 – Modèle Conceptuel de Données (MCD) d'Instruct'IF

Le modèle du domaine d'INSTRUCT'IF est conçu pour gérer efficacement les relations entre les élèves, les intervenants et les soutiens scolaires. Il s'appuie sur une structure de base de données relationnelle, utilisant l'héritage et les relations entre entités pour modéliser les interactions du système. Voici une description de son architecture principale et des entités clés :

2.1 Entités principales

Elève Cette entité représente les utilisateurs élèves de l'application. Chaque élève est caractérisé par ses attributs dont un identifiant (unique et généré automatiquement), son mail (unique) ou encore son établissement auquel il est rattaché. L'association avec l'établissement est gérée via une relation ManyToOne.

Intervenant Base de l'héritage pour les intervenants, cette entité contient les attributs communs à tous les types d'intervenants dont un identifiant (unique et généré automatiquement) ou encore son téléphone (unique également). L'héritage est implémenté via l'annotation `@Inheritance(strategy = InheritanceType.JOINED)`, permettant de dériver d'autres types d'intervenants tout en partageant la même table de base : **Etudiant** ayant les attributs université et spécialité, **Enseignant** ayant l'attribut du type d'établissement et **AutreIntervenant** avec un attribut précisant leur activité principale.

Soutien Cette entité représente une session de soutien scolaire. Elle inclut entre autres l'état du soutien : EN_ATTENTE, EN_VISIO, TERMINE et des relations ManyToOne avec les entités **Matiere**, **Eleve** et **Intervenant**.

Matiere Représente les différentes matières enseignées. Chaque matière est caractérisée par un identifiant unique et un nom.

Etablissement Décrit les établissements scolaires. Chaque établissement a entre autres un identifiant unique (UAI) et ses coordonnées géographiques (latitude et longitude) récupérées grâce à l'API.

2.2 Diagramme des classes

Le diagramme de la figure 2 montre l’agencement des données dans INSTRUCT’IF, reliant élèves, établissements et intervenants. Il souligne l’inscription des élèves, la gestion des soutiens et l’assignation des intervenants par spécialité. Ce modèle soutient la coordination des activités éducatives sur la plateforme.

3 Description des IHM

3.1 Enchaînement des fenêtres

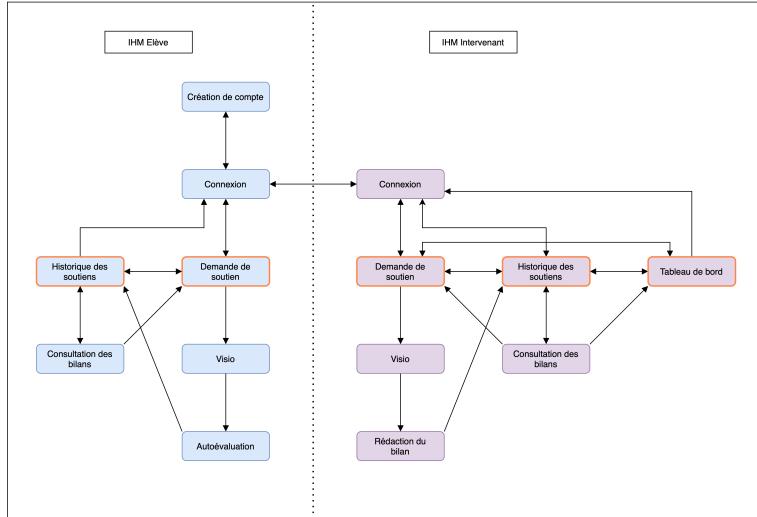


FIGURE 3 – Diagramme des enchaînements de fenêtres l’application Instruct’IF

Nous avons conçu l’interface d’INSTRUCT’IF pour qu’elle soit intuitive. Ainsi, les flux d’actions pour les élèves et pour les intervenants sont clairement définis, permettant une navigation fluide et une expérience utilisateur optimisée. Cela assure que les utilisateurs, qu’ils soient élèves ou intervenants, puissent facilement réaliser les actions clés nécessaires pour l’utilisation de la plateforme INSTRUCT’IF.

L’enchaînement des IHM (Interface Homme-Machine) présenté dans le diagramme ci-dessus montre les flux d’interactions pour les élèves et les intervenants sur la plateforme INSTRUCT’IF.

Les fonctionnalités clés, encadrées en orange, sont intégrées dans le bandeau d’en-tête du site, rendant ces options régulièrement accessibles aux utilisateurs. Pour les élèves, cela comprend la demande de soutien et l’historique des soutiens. Les intervenants ont un accès facile à la demande de soutien, à l’historique des soutiens, ainsi qu’à leur tableau de bord. Ces IHM essentielles sont complétées par des fonctionnalités telles que la visioconférence et l’autoévaluation pour les élèves, ainsi que la rédaction du bilan pour les intervenants.

3.2 IHM Communes

La fenêtre de connexion pour l’élève (Figure 4) présente un formulaire avec les champs "Email" (contenant "alice.cutterman@insa-lyon.fr") et "Mot de passe" (contenant "LaChuteCamus"). Des boutons "Se connecter" et "Je suis intervenant(e)" sont situés au bas du formulaire. En haut à droite, il y a un bouton "Créer un compte".

FIGURE 4 – Fenêtre de connexion (Elève)

La fenêtre de connexion pour l’intervenant (Figure 5) présente un formulaire avec les champs "Téléphone" (contenant "0677558844") et "Mot de passe" (contenant "GuerreDesGaulesCesar"). Des boutons "Se connecter" et "Je suis étudiant(e)" sont situés au bas du formulaire. En haut à droite, il y a un bouton "Créer un compte".

FIGURE 5 – Fenêtre de connexion (Intervenant)

Intension	Contrôle	Action	Réponse
INITIALISATION			<ul style="list-style-type: none"> - Afficher les éléments correspondants à cette fenêtre (Titre, Champs de mail et de mot de passe vides, Bouton Se Connecter, Bouton Je suis intervenant(e), Bouton Créer un Compte) - ModeElève
Se connecter	Bouton: « Se connecter »	Clic	Si (infoRemplies) Si ModeIntervenant OK authentifierIntervenantTelephone(telephone, mdp) Sinon si ModeElève OK authentifierEleveMail(mail, motDePasse) Sinon Message d'erreur
S'inscrire	Bouton: « créer un compte »	Clic	Passer à la fenêtre InscriptionEleve
Passer en mode intervenant	Bouton: « je suis intervenant(e) »	Clic	ModeIntervenant
Passer en mode étudiant	Bouton: « je suis étudiant(e) »	Clic	ModeElève
Modes			
ModeIntervenant			
<ul style="list-style-type: none"> - Vider les champs de remplissage. - Mettre le style de la fenêtre à celui des intervenants (violet) - Cacher le bouton « je suis intervenant(e) » - Afficher le bouton « je suis étudiant(e) » 			
ModeElève			
<ul style="list-style-type: none"> - Vider les champs de remplissage. - Mettre le style de la fenêtre à celui des Elèves (bleu) - Cacher le bouton « je suis étudiant(e) » - Afficher le bouton « je suis intervenant(e) » 			

FIGURE 6 – ICAR+S associé à l'IHM de connexion

Date	Intervenant	Matière	Autoévaluation
27/01/2024	Camille Martin	Histoire-Géographie	4/5
22/01/2024	Paul Bocuse	Mathématiques	5/5
03/01/2024	Christian Etienne	Histoire-Géographie	2/5

FIGURE 7 – Consultation de l'historique (Elève)

Date	Eleve	Matière	Note de l'élève
27/01/2024	Alice Cutteman	Histoire-Géographie	4/5
22/01/2024	Albert Camus	Français	1/5
03/01/2024	Alan Turing	Mathématiques	4/5

FIGURE 8 – Consultation de l'historique (Intervenant)

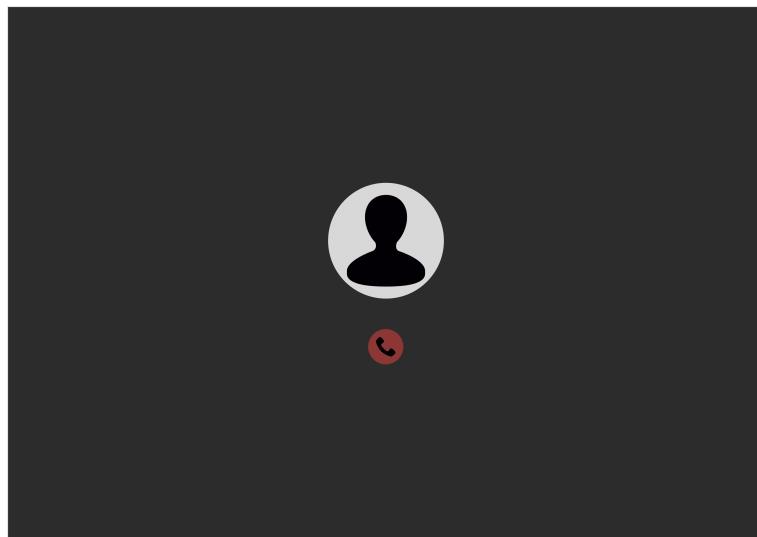


FIGURE 9 – Fenêtre de visioconférence

Intension	Contrôle	Action	Réponse
INITIALISATION			<ul style="list-style-type: none"> - Afficher les éléments correspondants à cette fenêtre (Titre, case des historiques) <p>si ModeEtudiant</p> <ul style="list-style-type: none"> - trouverHistoriqueEleve(eleve) <p>sinon</p> <ul style="list-style-type: none"> - trouverHistoriqueIntervenant(intervenanIdt)
Changer la fenêtre	Menu : Tableau de bord	Clic	- Passer à la fenêtre Tableau de bord
	Menu : Demander un soutien	Clic	- Passer à la fenêtre Demander un soutien
Se déconnecter	Bouton: « Connexion »	Clic	- Passer à la fenêtre Connexion
Modes			
ModeEtudiant			
- Afficher la colonne 'Intervenant'			
ModeIntervenant			
- Modifier la colonne 'Intervenant' en 'Eleve'			
- Afficher le bouton tableau de bord dans la barre des menus			

FIGURE 10 – ICAR+S associé à l'IHM de l'historique

Intension	Contrôle	Action	Réponse
INITIALISATION			<ul style="list-style-type: none"> - Afficher la vidéo - Afficher l' icône micro Raccrocher
Quitter la visio	icône: «Raccrocher »	Clic	<ul style="list-style-type: none"> - Fermeture de la visio : <p>Si ModeEtudiant</p> <ul style="list-style-type: none"> - terminerVisio(soutienEleve) - refreshSoutien(soutienIntervenant) <ul style="list-style-type: none"> -Fermer la visio pour l'Eleve -Fermer la visio pour l'Intervenant -Ouverture de la fenêtre AutoEvaluation <p>sinon</p> <ul style="list-style-type: none"> - terminerVisio(soutienIntervenant) - refreshSoutien(soutienEleve) - Fermer la visio pour l'Eleve Fermer la visio pour l'Intervenant - Ouverture de la fenêtre Bilan
Modes			
ModeEtudiant			
ModeIntervenant			

FIGURE 11 – ICAR+S associé à l'IHM de visioconférence

3.3 IHM Elève

Instruct'IF
Espace étudiant

Connexion

Créer un compte

Prénom	Alice
Nom	Cuttermen
Date de naissance	30/02/2003
Email	alice.cuttermen@insa-lyon.fr
Mot de passe	LaChuteCamus
Code Etablissement	JEJAU69100
Classe	5

Créer un compte

FIGURE 12 – Fenêtre de création de compte de (IHM Elève)

Intension	Contrôle	Action	Réponse
INITIALISATION			- Afficher les champs de remplissage vides pour les informations de l'élève - Afficher les éléments correspondants à cette fenêtre (Titre, Champs de Prénom, Nom, Date de Naissance, Email, Mot de Passe, Code Etablissement (UAI), Classe, Boutons Connexion et Créer un Compte)
S'inscrire	Bouton: « Créer un compte »	Clic	Si Infos Remplies et Valides OK - inscrireElève(eleve, uai) - Passer à la fenêtre connexionElève Sinon - Message d'erreur
Se connecter	Bouton: « Connexion »	Clic	Passer à la fenêtre connexionElève
Modes			

FIGURE 13 – ICAR+S associé à l'IHM de création de compte (IHM Elève)

FIGURE 14 – Fenêtre de demande de soutien (IHM Elève)

FIGURE 15 – Fenêtre d'auto évaluation de l'élève (IHM Elève)

Intension	Contrôle	Action	Réponse
INITIALISATION			- Afficher les éléments correspondants à cette fenêtre (Titre, Boutons Demander un soutien, Historique des demandes, Se déconnecter, Valider, Menu déroulant Matière, Zone de texte vide) - ModeDemande
Aller à l'historique des demandes	Bouton: «Historique des demandes »	Clic	- Passer à la fenêtre Historique en mode Eleve
Se déconnecter	Bouton: « Connexion »	Clic	- Passer à la fenêtre Connexion
Choisir la matière pour laquelle un soutien est demandé	ComboBox : "Matière"	Clic Choix d'une matière	- consulterListeMatières() Afficher de la liste des matières pour lesquelles l'élève peut demander un soutien avec un menu déroulant - Changer la matière choisie - Fermer La liste déroulante
Rédiger une demande de soutien	Champ de texte : "Descriptif"	Clic dans la zone de texte puis saisie du texte avec le clavier	- Champ de saisie et descriptif modifiables
Envoyer une demande de soutien	Bouton : "Rechercher un intervenant"	Clic	- soutienEleve = demanderSoutien(eleve, matiere, descriptif) Si TrouverIntervenant OK - ModeAttente si VisioDémarrée OK - Passer à la fenêtre Visio sinon ModeErreur
Modes			
ModeDemande : - Activer la zone de saisie et dégriser le bouton 'Rechercher un intervenant' - Passer les champs en modifiables (matière et description)			
ModeAttente : - Désactiver la zone de saisie et griser le bouton 'Rechercher un intervenant' - Passer les champs en non-modifiables (matière et description) - changer le label du bouton 'Recherche' en 'Attente du lancement de la visio'			
ModeErreur : - Afficher qu'aucun intervenant n'a été trouvé - ModeDemande			

FIGURE 16 – ICAR+S associé à l'IHM de demande de soutien (IHM Elève)

Intension	Contrôle	Action	Réponse
INITIALISATION			-Afficher le titre 'Auto-évaluation' - Afficher les boutons d'autoévaluation - Afficher le bouton "Envoyer"
Choix de l'évaluation	Bouton d'évaluation	Clic	- changer le choix de l'autoévaluation - faire correspondre ce choix à une note comprise entre 1 et 5
Envoyer l'évaluation	Bouton "Envoyer"	Clic	- faireAutoEvaluationEleve(soutienEleve, note) - refreshSoutien(soutienIntervenant) -Ouverture de la fenêtre HistoriqueElèves
Modes			

FIGURE 17 – ICAR+S associé à l'IHM d'auto évaluation de l'élève (IHM Elève)

3.4 IHM Intervenant

Intension	Contrôle	Action	Réponse
INITIALISATION			Afficher les éléments correspondants à cette fenêtre (Titre, Bouton Lancer la visio, Historique des interventions, Se déconnecter, Valider, Descriptif) soutienIntervenant = obtenirSoutienEnAttenteParIntervenantId(intervenantId)
Aller à l'historique des demandes	Bouton: "Historique des demandes"	Clic	- Passer à la fenêtre HistoriqueDesDemandes en ModeIntervenant
Se déconnecter	Bouton: "Connexion"	Clic	- Passer à la fenêtre Connexion
Lancer la visio	Bouton : "Lancer la visio"	Clic	lancerVisio(soutienIntervenant) refreshSoutien(soutienEleve) Passer à la fenêtre Visio
Modes			

FIGURE 18 – ICAR+S associé à l'IHM de demande de soutien (IHM Intervenant)

Instruct'IF
Espace intervenant

Tableau de bord Demander un soutien Historique interventions Se déconnecter

Demande de soutien

Matière : Histoire-Géographie
Elève : Alice Cuttermann
Etablissement : Jean Jaures Villeurbanne
Classe : 5

Descriptif
Je n'ai pas très bien compris le chapitre sur le Moyen-Age, j'aimerais un petit peu d'aide avant de commencer mon devoir

Lancer la visio

FIGURE 19 – Fenêtre de demande de soutien (IHM Intervenant)

Instruct'IF
Espace intervenant

Bilan d'intervention

Bilan

Alice à très bien compris ce chapitre en Histoire-Géographie, cependant les événements et leurs chronologies étaient mélangés. Il faut qu'elle apprenne mieux ses dates et les personnages clés de chaque leçon.

Valider

FIGURE 20 – Fenêtre de rédaction du bilan (IHM Intervenant)

Intension	Contrôle	Action	Réponse
INITIALISATION			-Afficher le titre 'Bilan d'Intervention' -Afficher la zone de saisie du bilan - Afficher le bouton "Envoyer"
Rédaction du Bilan	Zone de texte	Saisie	-Saisie du Bilan
Envoi du Bilan	Bouton "Envoyer"	Clic	- redigerBilan(soutienIntervenant, bilan) - refreshSoutien(soutienEleve) -Ouverture de la fenêtre HistoriqueIntervenant
Modes			

FIGURE 21 – ICAR+S associé à l'IHM de rédaction du bilan (IHM Intervenant)



FIGURE 22 – Fenêtre du tableau de bord (IHM Intervenant)

Intension	Contrôle	Action	Réponse
INITIALISATION			<ul style="list-style-type: none"> - Afficher les éléments correspondants à cette fenêtre (Titre, menu, bouton Se déconnecter, statistiques) - Afficher les statistiques relatives au nombre total d'interventions, durée moyenne d'un soutien, satisfaction moyenne des élèves, IPS moyen, nombre d'intervenants actifs et intervenant du mois <pre>statNbTotalIntervention() statDureeMoyenneIntervention() statSatisfactionEleve() statIntervenantMois() statIntervenantActif() statNbEleveInscrit() statIpsMoyenSoutien() - Afficher la répartition des soutiens effectués sur une carte statQuantiteSoutienParCoordonnees()</pre>
Aller à l'historique des demandes	Bouton: «Historique des demandes »	Clic	<ul style="list-style-type: none"> - Passer à la fenêtre Historique en modeIntervenant
Aller au à la demande de soutien	Bouton: «Demander un soutien »	Clic	<ul style="list-style-type: none"> - Passer à la fenêtre DemandeDeSoutienIntervenant
Se déconnecter	Bouton: « Connexion »	Clic	<ul style="list-style-type: none"> - Passer à la fenêtre Connexion
Modes			

FIGURE 23 – ICAR+S associé à l'IHM du tableau de bord (IHM Intervenant)

4 Spécification des services

4.1 Services d'initialisation

```
public void initialisation()
```

Nous avons fait le choix de regrouper la création en "dur" des intervenants ainsi que des matières dans un service d'initialisation. Cependant les services d'inscription des intervenants ainsi que des matières sont dans des services externes. Cela permet notamment d'enrichir les bases de données avec d'autres intervenants ou d'autres matières pour des extensions futures.

```
public Boolean inscrireIntervenant(Intervenant intervenant)
```

```
public Boolean creerMatiere(Matiere matiere)
```

4.2 Services élèves

```
public Boolean inscrireEleve(Eleve eleve, String uai)
```

Ce service d'inscription d'élève réalise la persistance des élèves dans la base de données. Il commence par vérifier que l'établissement de l'élève qui a été renseigné à l'aide de son code uai est valide. Cette validation est

réalisée à l'aide d'une API externe (EducNetApi), puis persiste en base de donnée, si cela n'est pas déjà fait, l'établissement.

Voici un pseudo code expliquant le déroulé de cet algorithme :

```
fonction inscrireEleve(eleve, uai) {
    creer le contexte de persistance avec JpaUtil

    si EducNetApi confirme l'existence de l'établissement avec uai {
        si l'établissement n'existe pas dans notre base {
            creer un nouvel établissement avec ces informations via EducNetApi
            obtenir latitude et longitude de l'établissement via GeoNetApi
            persister le nouvel établissement
        }
        sinon {
            recuperer l'établissement existant de notre base
        }
        associer l'eleve a l'établissement
        persister l'eleve
        valider la transaction
        envoyer un mail de confirmation a l'eleve
    }
    si echec {
        annuler la transaction
        envoyer un mail d'echec a l'eleve
    }
}

public Eleve authentifierEleveMail(String mail, String motDePasse)
```

Ce service permet aux élèves de s'authentifier à l'aide de leur mail ainsi que leur mot de passe. Il fait appel à la méthode getParMail de la classe eleveDao qui retourne l'élève avec le mail associé et compare les mots de passe. Si le mot de passe est correct, ce service retourne l'élève.

```
public List<Soutien> trouverHistoriqueEleve(Eleve eleve)
```

Ce service retourne une liste d'objets soutien. Ces objets sont tous les soutiens réalisé par l'élève passé en paramètre.

4.3 Services intervenants

```
public Intervenant authentifierIntervenantTelephone(String telephone, String motDePasse)
```

Ce service permet aux intervenants de s'authentifier à l'aide de leur numéro de téléphone ainsi que leur mot de passe. Il va faire appel à la méthode getParTelephone de la classe intervenantDao qui retourne l'intervenant avec le numéro de téléphone associé et compare les mot de passe. Si le mot de passe est correct, ce service retourne l'intervenant.

```
public List<Soutien> trouverHistoriqueIntervenant(Intervenant intervenant)
```

Ce service retourne une liste d'objets soutien. Ces objets sont tous les soutiens supervisés par l'intervenant passé en paramètre.

4.4 Services soutiens

Lorsqu'un élève souhaite demander un soutien, il remplit les informations nécessaires puis demande le soutien avec le service demanderSoutien.

```
public Soutien demanderSoutien(Eleve eleve, Matiere matiere, String descriptif)
```

Ce service permet de persister un soutien d'un élève. Il prend en paramètre l'élève qui à fait la demande, la matière sélectionnée à l'aide du service consulterListeMatiere ainsi que le descriptif renseigné par l'élève. Cette méthode va notamment faire appel aux setter de la classe Soutien pour définir ses attributs. La méthode trouverIntervenantSoutien décrite postérieurement sélectionne un intervenant puis l'associe au soutien. Une notification est finalement envoyée à l'intervenant pour l'informer de son soutien en attente.

```
fonction demanderSoutien(eleve, matiere, descriptif) {
    creer le contexte de persistance avec JpaUtil
    creer un nouvel objet Soutien
```

```

definir l'eleve, le descriptif et la matiere du soutien
trouver un intervenant pour le soutien avec la methode trouverIntervenantSoutien
si aucun intervenant n'a ete trouve {
    envoyer un mail de rejet de la demande et sortir de la methode en envoyant une
        exception }
sinon {
    assigner cet intervenant au soutien}
mettre a jour la disponibilite et le nombre d'interventions de l'intervenant

ouvrir une transaction
mettre a jour l'intervenant dans la base de donnees
persister l'objet Soutien dans la base de donnees
valider la transaction

envoyer une notification a l'intervenant pour lui signaler la prise en charge de la
demande

si ehcec {
    afficher l'erreur
    annuler la transaction
    envoyer un mail a l'eleve pour signaler l'echec de la demande
}
fermer le contexte de persistance
retourner l'objet Soutien
}

```

public List<Matiere> consulterListeMatieres()

Service permettant d'obtenir la liste des matières disponible sur l'application.

public Intervenant trouverIntervenantSoutien(Eleve eleve)

Ce service sélectionne un intervenant selon les critères exigés dans le cahier des charges. En premier lieu nous créons une liste de tous les intervenants puis les trions par ordre croissant de nombre d'interventions réalisées. Nous parcourons ensuite cette liste et sélectionnons le premier qui vérifie les conditions de disponibilité ainsi que de niveau académique compatible avec l'élève.

```

fonction trouverIntervenantSoutien(Eleve eleve) {
    cree une liste de tous les intervenants
    trie cette liste par ordre croissant de nombre d'intervention
    pour chaque intervenant de la liste {
        si disponibilite == true && niveauMin < classeEleve < niveauMax {
            selectionne cet intervenant
            break
        }
    }
    retourne intervenant selectionne
}

```

public Soutien obtenirSoutienEnAttenteParEleveId(Long eleveId)

public Soutien obtenirSoutienEnAttenteParIntervenantId(Long intervenantId)

Ces deux services renvoient les soutiens en attente d'un intervenant et d'un élève dont l'id est rentré en paramètre.

public Boolean lancerVisio(Soutien soutien)

Ce service permet de lancer la visioconférence. Elle vérifie tout d'abord qu'un intervenant à bien été sélectionné puis met à jour l'état pour qu'il passe à " EN VISIO " et renseigne également la date de début de la visioconférence.

public Boolean terminerVisio(Soutien soutien)

Ce service permet de mettre fin à la visioconférence. Elle met à jour l'état pour qu'il passe à " TERMINE " et renseigne également la durée de la visioconférence. Ce service repasse également la disponibilité de l'intervenant à true.

public Boolean faireAutoEvaluationEleve(Soutien soutien, Integer note)

Service permettant à l'élève de s'autoévaluer sur sa compréhension au terme de la visio. Cette note est ensuite utilisée pour calculer la satisfaction des élèves.

```
public Boolean redigerBilan(Soutien soutien, String bilan)
```

Service permettant aux intervenant de réaliser un bilan de l'intervention à la suite de la visio. Ce bilan est ensuite envoyé par mail à l'élève mais aussi disponible dans l'historique des soutiens.

```
public Soutien refreshSoutien(Soutien soutien)
```

Service permettant de rafraîchir l'état des entités Soutien avec leur représentation actuelle dans la base de données. Cela signifie que si des modifications ont été apportées à l'entité Soutien dans la base de données par d'autres transactions depuis le moment où il a été récupéré ou la dernière fois qu'il a été rafraîchié, ces modifications seront reflétées dans l'instance de l'entité après l'appel de refreshSoutien(soutien). Par exemple, si c'est l'intervenant qui a lancé la visio, la méthode lancerVisio(soutien_intervenant) va changer l'état du soutien dans l'instance soutien_intervenant mais pas dans l'objet soutien_eleve. Il faut alors appeler refreshSoutien(soutien_eleve) pour que l'objet soutien_eleve soit à jour et conforme à l'entité Soutien de la base de données.

4.5 Services statistiques

Ces services sont utilisés pour réaliser les statistiques du tableau de bord visible dans l'IHM intervenant. Ils permettent notamment d'obtenir le nombre total d'intervention sur la plateforme tout intervenant confondus, la durée moyenne d'une intervention, la satisfaction des élèves, l'intervenant du mois, le nombre d'intervenant actifs et d'élèves inscrits. Nous pouvons également obtenir la satisfaction moyenne des élèves et enfin la répartition géographique des interventions.

```
public Long statNbTotalIntervention()

public Double statDureeMoyenneIntervention()

public Double statSatisfactionEleve()

public String statIntervenantMois()

public Long statIntervenantActif()

public Long statNbEleveInscrit()

public Double statIpsMoyenSoutien()

public List<List<Object>> statQuantiteSoutienParCoordonnees()
```

Ce dernier service retourne une List de List contenant des Object. Chaque List interne représente un ensemble de données où chaque élément correspond à :

- La longitude de l'établissement (convertie en Double),
- La latitude de l'établissement (convertie en Double),
- Le nombre de soutiens associés à ces coordonnées géographiques (converti en Integer).

La fonction utilise une requête JPQL pour grouper les soutiens par les coordonnées géographiques des établissements des élèves et compter le nombre de soutiens pour chaque groupe de coordonnées. Les résultats de la requête sont ensuite organisés dans une structure de liste de listes pour être retournés.