

Dossier de spécification BACKEND Instruct'if – GRIGUER Mehdi (B3220)

Instruct'if est un site internet de soutien scolaire où l'on met en relation, des élèves désirant prendre des cours de soutien, et des intervenants bénévoles voulant les aider.

IHM et tableau ICARS associés

Fenêtre Se connecter

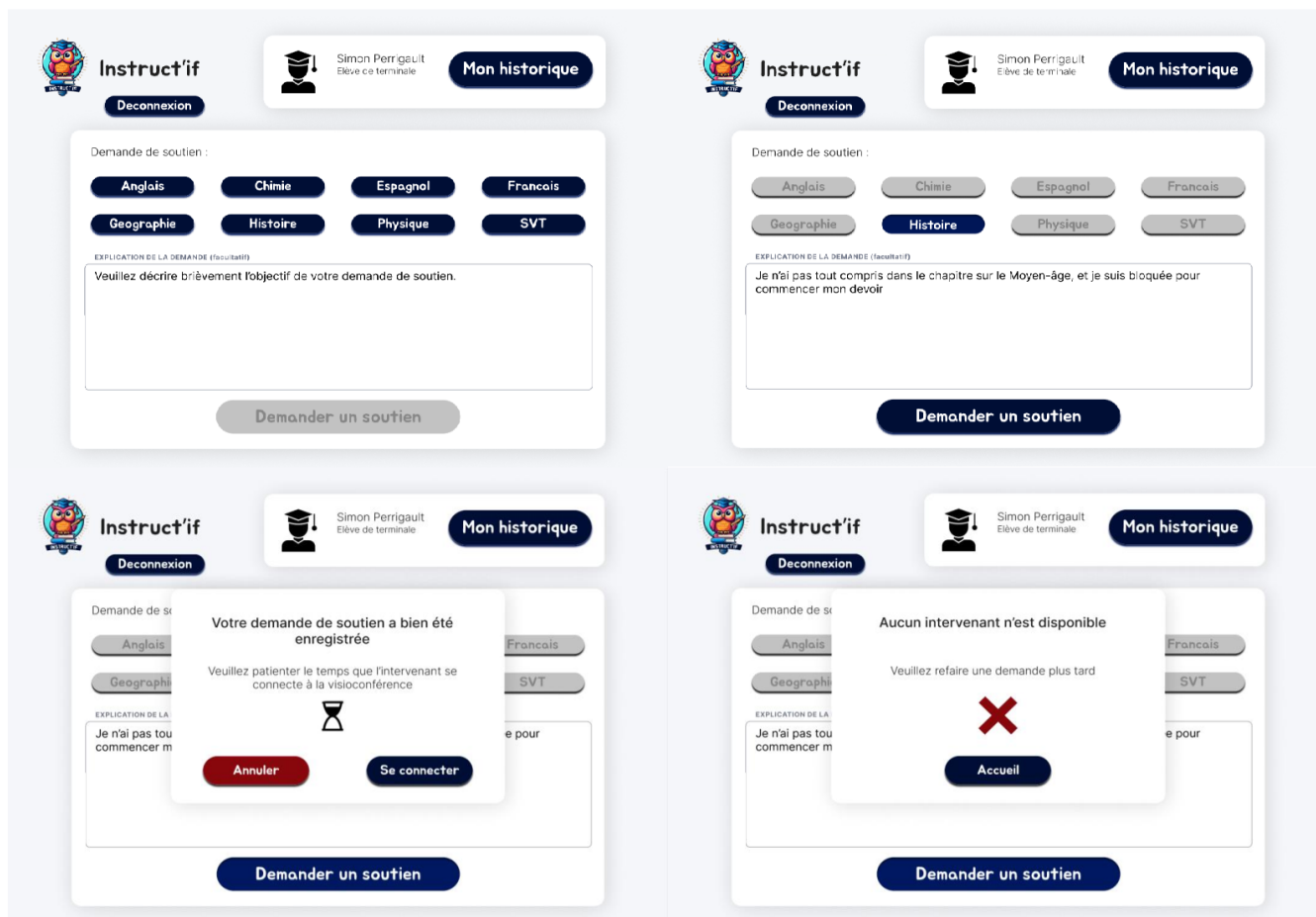
Intention	S'inscrire	Se connecter
Contrôle	Bouton s'inscrire	Bouton continuer
Action	clic	clic
Réponse	Redirection vers fenêtre Inscription	Service authentifier Eleve ou authentifier Intervenant

Fenêtre Inscription

Intention	Rentrer ses informations	S'inscrire
Contrôle	Champ de texte	Bouton s'inscrire
Action	clic et clavier	clic
Réponse		Service inscrireEleve

Fenêtre Accueil Élève

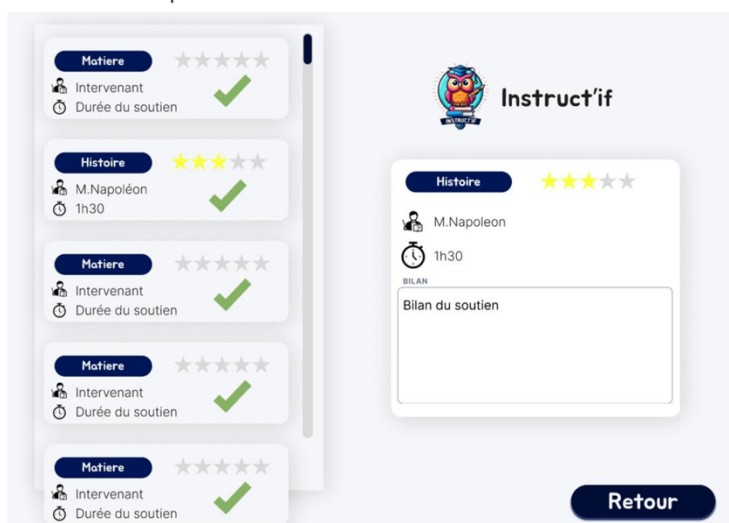
Intention	Contrôle	Action/Event	Réponse
Se déconnecter	Bouton deconnexion	clic	Redirection vers fenêtre Se connecter
Voir historique	Bouton mon historique	clic	Redirection vers fenêtre Historique Élève
Choix matière soutien	Boutons des matières	clic	Passage des autres boutons en gris
Expliquer demande soutien	Champ de texte	clic et clavier	
Soumettre demande soutien	Bouton demander un soutien	clic	Service creerIntervention + popup d'attente ou de refus



Fenêtre Historique Élève

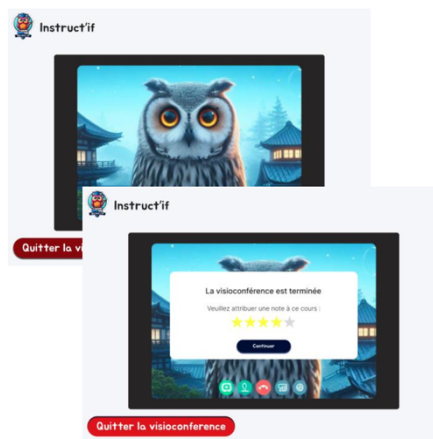
Intention	Retour à l'accueil élève	Voir détail d'un soutien
Contrôle	Bouton retour	Cliquer sur le soutien en question
Action	clic	clic
Réponse	Redirection vers fenêtre Accueil Élève	Affichage des détails avec des getter

Service historiqueInterventionEleve donne la liste des interventions.



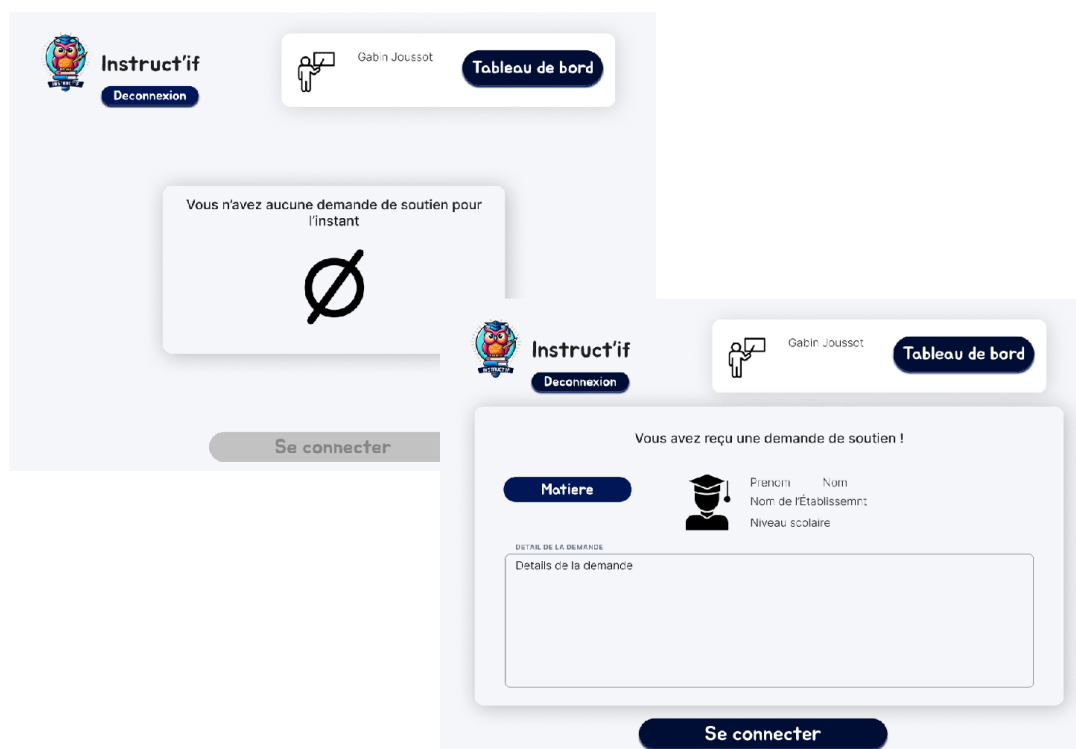
Fenêtre Visioconférence Élève

Intention	Arrêter la visio
Contrôle	Bouton quitter la visio
Action	clic
Réponse	Popup pour entrer note + Service envoiNote



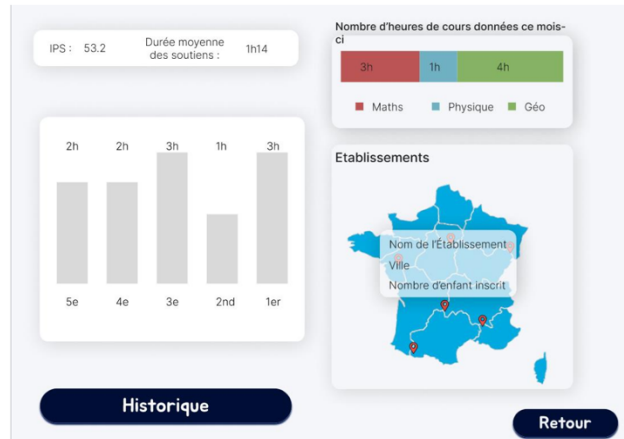
Fenêtre Accueil Intervenant

Intention	Contrôle	Action/Event	Réponse
Se déconnecter	Bouton deconnexion	clic	Redirection vers fenêtre Se connecter
Voir Tableau de bord	Bouton tableau de bord	clic	Redirection vers fenêtre Tableau de bord
Accepter soutien	Bouton se connecter	clic	Redirection vers fenêtre visioconférence intervenant



Fenêtre Tableau de bord

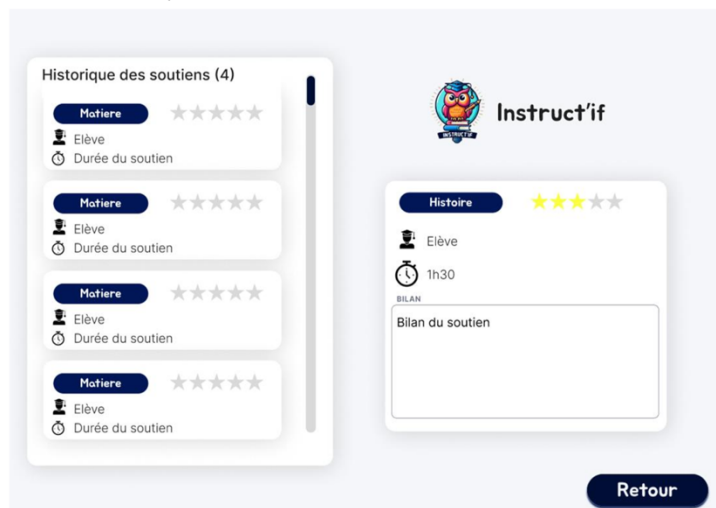
Intention	Retourner à l'accueil intervenant	Voir historique
Contrôle	Bouton retour	Bouton historique
Action	clic	clic
Réponse	Redirection vers fenêtre Accueil intervenant	Redirection vers fenêtre historique



Fenêtre Historique Intervenant

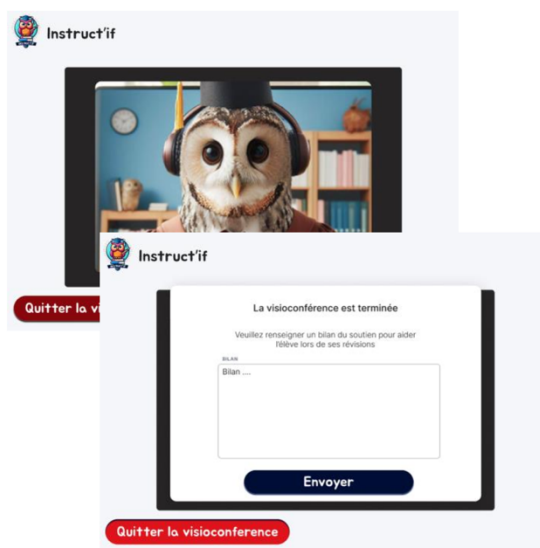
Intention	Retour à l'accueil intervenant	Voir détail d'un soutien
Contrôle	Bouton retour	Cliquer sur le soutien en question
Action	clic	clic
Réponse	Redirection vers fenêtre Accueil Intervenant	Affichage des détails avec des getter

Service historiqueInterventionIntervenant donne la liste des interventions.



Fenêtre Visioconférence Intervenant

Intention	Arrêter la visio
Contrôle	Bouton quitter la visio
Action	clic
Réponse	Popup pour entrer bilan + Service envoiBilan



Spécification des services (signature, courte description des services et algo en pseudo-code pour les services complexes)

void initBD() :

Initialise la base de données en y mettant des matières et des intervenants. Les intervenants peuvent être soit des étudiants, des enseignants, ou autres.

boolean inscrireEleve(Eleve eleve, String para_uai) :

Inscrit un élève dans la bd, si le code UAI de son établissement est valide. L'établissement est soit déjà dans notre bd, soit on le trouve à l'aide d'une API externe avant de le mettre dans la bd. Elle renvoie true si l'inscription est réussie ou false sinon.

Eleve authentifierEleve(String mail, String motDePasse) :

Authentifie un élève en vérifiant ses identifiants de connexion. Elle retourne l'Eleve si l'authentification est réussie ou null en cas d'échec.

Intervenant authentifierIntervenant(String mail, String motDePasse) :

Authentifie un intervenant en vérifiant ses identifiants de connexion. Elle retourne l'Intervenant si l'authentification est réussie ou null en cas d'échec.

Intervenant creerIntervention(String para_matiere, Eleve para_eleve, String para_detail) :

Crée une nouvelle intervention pour un élève dans une matière donnée avec les détails de l'intervention. Cette méthode débute directement l'intervention si elle trouve un intervenant disponible. Elle renvoie l'intervention si on a réussi à la créer et null sinon.

boolean envoiBilan(Intervention intervention, String bilan) :

Enregistrement dans la bd et envoi par mail à l'élève, du bilan d'intervention fait par l'intervenant. Cette méthode arrête l'intervention en cours.

boolean envoiNote(Intervention intervention, int note) :

Enregistrement dans la bd de la note donnée par l'élève à l'intervention.

List<Intervention> historiqueInterventionEleve(Eleve eleve) :

Renvoie la liste des interventions pour un élève donné.

List<Intervention> historiqueInterventionIntervenant(Intervenant intervenant) :

Renvoie la liste des interventions pour un intervenant donné.

List<Matiere> obtenirListeDesMatiere() :

Renvoie la liste de toutes les matières disponibles dans la bd.

HashMap<Integer,Long> obtenirMinutesDeCoursParNiveau(Intervenant intervenant) :

Calcule et retourne le nombre total de minutes d'interventions effectuées par un intervenant en fonction du niveau des élèves. Le résultat est renvoyé sous forme de dictionnaire avec comme clé le niveau et comme attribut les minutes de cours.

HashMap<Matiere,Long> obtenirMinutesDeCoursParMatiere(Intervenant intervenant) :

Calcule et retourne le nombre total de minutes d'interventions effectuées par un intervenant en fonction de la matière enseignée. Le résultat est renvoyé sous forme de dictionnaire avec comme clé la matière et comme attribut les minutes de cours.

Le seul service complexe est le service inscrireEleve.

```
Algorithme INSCRIRE_ELEVE
Entrée: Eleve eleve, String para_uai

Initialiser resultat à VRAI

Essayer
    Créer un contexte de persistance
    Essayer de trouver etab dans la bd par son code_uai

    Si etab est non nul Alors
        Afficher "Etablissement trouvé dans BD"
    Sinon
        Tenter de trouver etab en ligne via EducNetApi

        Si aucun résultat en ligne Alors
            Afficher "Etablissement non trouvé"
            resultat devient FAUX
        Sinon
            On remplit les attributs de l'etab avec les informations trouvées sur
EducNetApi
        Fin Si
    Fin Si

    Si resultat est VRAI Alors
        Ouvrir une transaction
        Incrémenter le nombre d'élèves inscrits dans l'etab
        Si etab n'est pas dans BD Alors
            On créer l'etab dans la BD
        Fin Si
        Associer etab à eleve
        On créer l'eleve dans la BD
        Valider la transaction
    Fin Si

Attraper Exception
    Afficher "Erreur d'inscription"
    Annuler la transaction
    resultat devient FAUX

Finalement
    Fermer le contexte de persistance

Retourner resultat
Fin Algorithme
```

Modèles UML du domaine

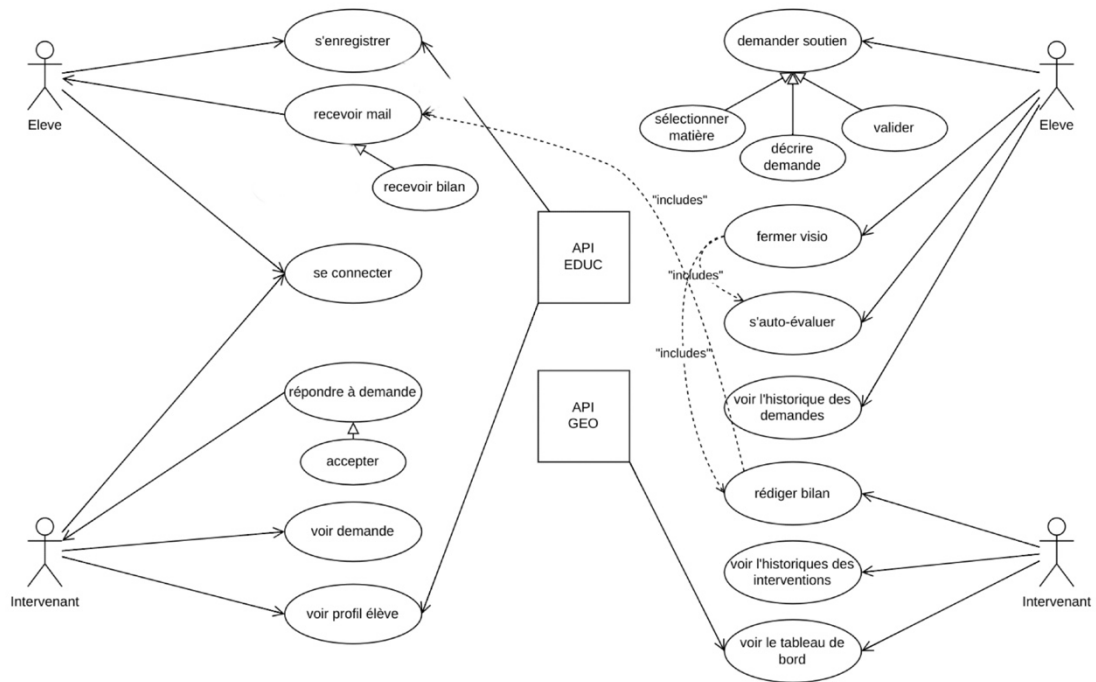


Figure 1: Diagramme de cas

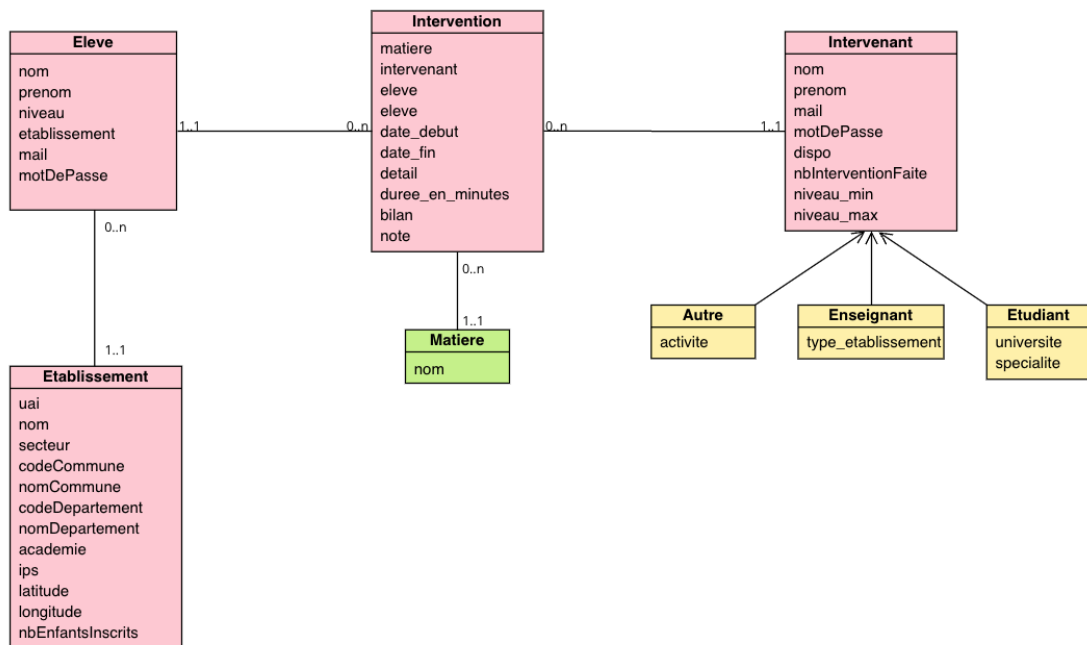


Figure 2 : Diagramme de classe

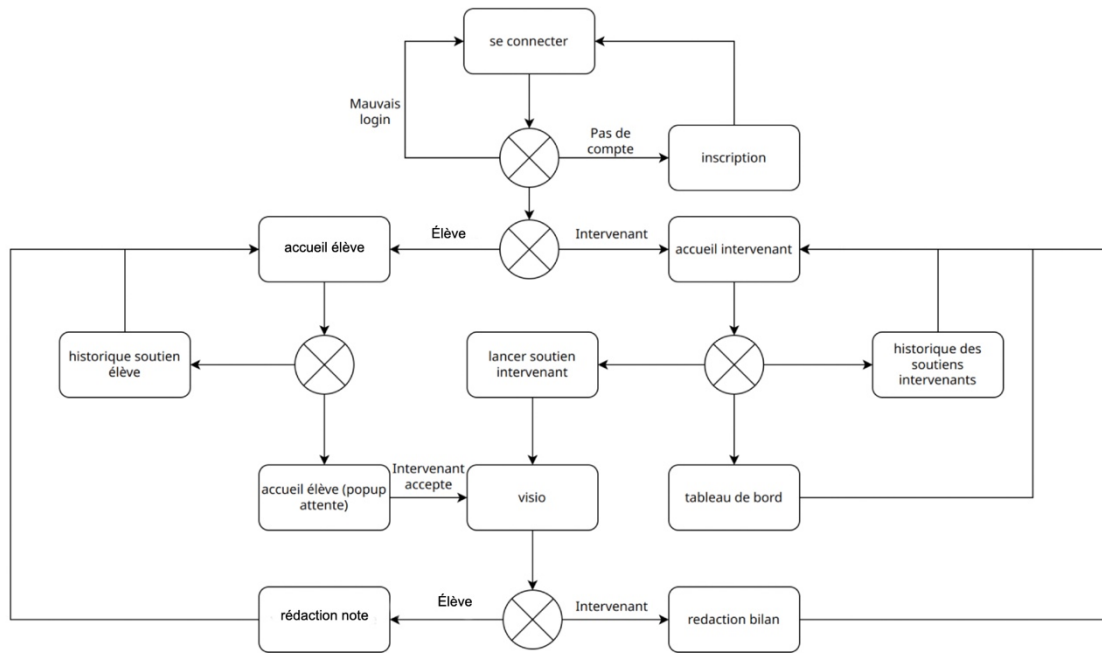


Figure 3 : Diagramme d'activité