

DASI : Projet PrédicIF

Introduction

Le projet "Predict'IF" est une application Java développée avec NetBeans qui fournit des services métiers pour l'implémentation d'une interface utilisateur. Ce projet est le fruit de la collaboration de plusieurs équipes, chacune ayant une expertise spécifique pour atteindre un objectif commun : développer une application robuste et évolutive qui répond aux besoins métiers de l'entreprise.

L'enjeu principal de ce projet était de fournir des services métiers de qualité pour permettre à l'autre groupe de développer une interface utilisateur intuitive et conviviale pour les utilisateurs finaux. Pour atteindre cet objectif, nous avons développé des classes métiers, de DAO, de vue d'outils et métiers en utilisant un JPA avec la syntaxe JPQL pour simplifier les interactions avec la base de données.

Dans ce contre-rendu, nous présentons un diagramme UML qui représente les différentes classes et relations du projet, ainsi qu'un tableau ICAR qui décrit les différents cas d'utilisation et les interactions avec les acteurs du système. Ces outils ont pour objectif de guider et d'aider le prochain groupe qui développera l'interface utilisateur en leur fournissant une compréhension claire des services métiers disponibles et des besoins métiers qu'ils doivent satisfaire.

Enfin, il est important de se munir d'un esprit critique à la lecture de ce contre-rendu et de manière plus globale, à la lecture du projet. En effet, le projet en lui-même s'articule autour de concepts plus que discutables, celui des médiums et des appels téléphoniques ! De plus, il est important de souligner que c'est la première fois où nous sommes confrontés à un travail en deux temps. Ainsi, nous avons fait au mieux pour que la prise en main du projet soit facile par l'autre groupe mais il se peut que certains détails nous aient échappé.

Diagramme UML

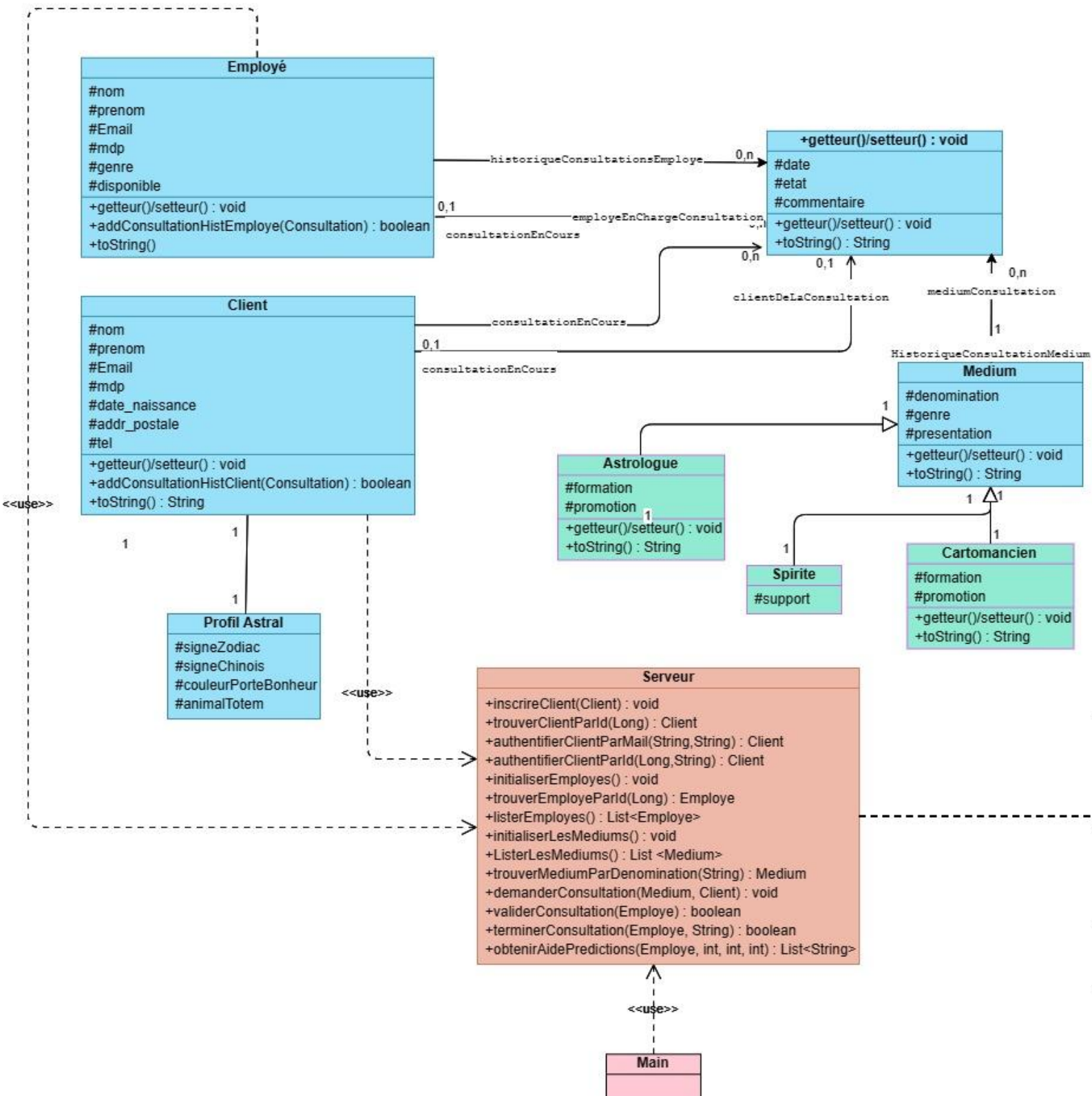


Figure 1 : diagramme UML

Le diagramme de classe UML permet de représenter la structure statique du système logiciel. Il montre les différentes classes du système, les attributs et méthodes, ainsi que les relations entre ces classes.

Description des services métiers

inscrireClient: prend en paramètre un objet de type **Client** pour insérer cet objet dans la base de données. Si l'inscription réussit, un e-mail de confirmation est envoyé au client et son profil astral est récupéré à l'aide de l'API **AstroNetApi**. Si l'inscription échoue, un e-mail d'erreur est envoyé au client.

trouverClientParId: prend en paramètre un identifiant de type **Long** pour récupérer l'objet **Client** correspondant à cet identifiant dans la base de données. Si le client est trouvé, il est renvoyé par la méthode. Sinon, la méthode renvoie **null**.

authentifierClientParMail: prend en paramètres une adresse e-mail et un mot de passe pour authentifier un client en utilisant son adresse e-mail et son mot de passe. Si l'authentification réussit, l'objet **Client** correspondant est renvoyé par la méthode. Sinon, la méthode renvoie **null**.

authentifierEmployeParMail: prend en paramètres une adresse e-mail et un mot de passe pour authentifier un **Employe** en utilisant son adresse e-mail et son mot de passe. Si l'authentification réussit, l'objet **Employe** correspondant est renvoyé par la méthode. Sinon, la méthode renvoie **null**.

authentifierClientParId: prend en paramètres un identifiant de type **Long** et un mot de passe pour authentifier un client en utilisant son identifiant et son mot de passe. Si l'authentification réussit, l'objet **Client** correspondant est renvoyé par la méthode. Sinon, la méthode renvoie **null**. (non utilisé dans l'application)

initialiserEmployes: crée en dur les objets de type **Employe** et insère ces objets dans la base de données.

trouverEmployeParId: prend en paramètre un identifiant de type **Long** pour récupérer l'objet **Employe** correspondant à cet identifiant dans la base de données. Si l'employé est trouvé, il est renvoyé par la méthode. Sinon, la méthode renvoie **null**. (non utilisé dans l'application)

listerEmployes: renvoie la liste de tous les objets **Employe** présents dans la base de données.

initialiserMediums: crée en dur les objets de type **Medium** et insère ces objets dans la base de données.

listerLesMediums : renvoie la liste de tous les objets **Medium** présents dans la base de données.

trouverMediumParDenomination: prend en paramètre une chaîne de caractères représentant la dénomination d'un médium et renvoie le **Medium** correspondant à cette dénomination dans la base de données. Si le médium n'est pas trouvé, la méthode renvoie **null**.

demandeConsultation: prend en paramètres un objet de type **Medium** et un objet de type **Client** et crée une nouvelle consultation entre ce médium et ce client. La méthode crée un nouvel objet de type **Consultation** avec la date courante et l'insère dans la base de données. Si un employé est disponible (son attribut Disponible doit être "true", doit avoir le moins de consultations de Prédic'IF, et doit être du même genre que le **Medium** qui sera à incarner) pour prendre en charge la consultation, le service met à jour les relations entre les différents objets impliqués dans la consultation (médium, client, employé) pour refléter cette nouvelle consultation, l'employé est alors affecté à celle-ci et une notification lui est envoyée. De plus, l'attribut Disponible de l'employé est passé en "false". Sinon, l'état de la consultation est mis à jour pour indiquer qu'elle a été refusée. Dans tous les cas, un e-mail est envoyé au client pour l'informer du résultat de sa demande de consultation.

validerConsultation : prend en paramètre un objet de type **Employe** pour valider la consultation en cours de cet employé. La méthode met à jour l'état de la consultation pour indiquer qu'elle a été validée et envoie un message sur le téléphone du client pour l'informer que la consultation a été validée. Si la validation réussit, la méthode renvoie **true**. Sinon, elle renvoie **false**.

terminerConsultation : prend en paramètres un objet de type **Employe** et une chaîne de caractères représentant un commentaire. Cette méthode permet de terminer la consultation en cours de cet employé. La méthode met à jour l'état de la consultation pour indiquer qu'elle a été terminée, ajoute le commentaire à la consultation et met à jour les objets **Employe** et **Client** pour indiquer que la consultation est terminée. Si la terminaison réussit, la méthode renvoie **true**. Sinon, elle renvoie **false**.

obtenirAidePredictions : prend en paramètres un objet de type **Employe** et trois entiers représentant les niveaux d'amour, de santé et de travail et utilise l'API **AstroNetApi** pour obtenir des prédictions pour le client associé à la consultation en cours de cet employé. Si des prédictions sont obtenues, elles sont renvoyées sous forme de liste de chaînes de caractères par la méthode.

obtenirConsultationsRefusees : récupère et renvoie la liste de toutes les consultations refusées présentes dans la base de données.

obtenirTopCinqMedium : renvoie la liste des cinq médiums les plus populaires.

obtenirMoyenneParGenre : prend en paramètre une chaîne de caractères représentant un genre et renvoie la moyenne du nombre de consultations par employé de ce genre.

obtenirProfilAstralClient : prend en paramètres un objet de type **Client** et renvoie le profil astral du client sous la forme d'une liste de string.

obtenirHistoriqueConsultationClient : prend en paramètres un objet de type **Client** et renvoie son historique de consultation sous la forme d'une liste de **Consultation**.

obtenirHistoriqueConsultationEmploye : prend en paramètres un objet de type **Employe** et renvoie son historique de consultation sous la forme d'une liste de **Consultation**.

obtenirConsultationCouranteEmploye : Retourne la consultation de l'**Employe** entré en paramètre. Si l'employé saisi n'a pas de consultation courante, la méthode renvoie null.

obtenirInformationConsultationEmploye : Méthode prenant en paramètre un employé et envoyant les informations nécessaires à l'employé pour effectuer sa consultation de la

meilleure des manières. La méthode renvoie un String qui synthétise les informations de la consultation.

Tableau ICAR

Voici le tableau ICAR qui regroupe tous les services métiers avec l'IHM que nous avons conçue.

PAGE D'ACCUEIL :

Intention	Contrôle	Événement	Réponse
Quitter	Page accueil: Quitter	Clic	Fermeture de la page
Inscrire Client	Page accueil : Inscrire client	Clic	<ul style="list-style-type: none"> - Ouverture de la page d'inscription Client - Mise à blanc des champs de saisie (nom, prénom..) - Activation du bouton valider inscription Client
Inscrire Employé (Non utilisée, car les employés sont codés "en dur")	Page accueil : Inscrire employé	Clic	<ul style="list-style-type: none"> - Ouverture de la page d'inscription Employé - Mise à blanc des champs de saisie (nom, prénom..) - Activation du bouton valider inscription Client
Connecter compte Client	Page d'accueil : Connexion Client	Clic	-authentifierClientParMail(String, String) : Client OU -authentifierClientParId(Long,String) : Client Les informations telles que le mail, l'ID et le mot de passe sont récupérées dans des cellules prévues à cet effet. -Si connexion réussie (Client retourné non null), accès à la page dédié au client -Si connexion échouée : gestion des erreurs
Connecter compte Employé	Page d'accueil : Connexion Employé	Clic	-authentifierEmployeParMail(String,String) : Employe Les informations telles que le mail et le mot de passe sont récupérées dans des cellules prévues à cet effet. -Si connexion réussie (Employé retourné non null), accès à la page dédié aux Employés -Si connexion échouée : gestion des erreurs

PAGE INSCRIPTION EMPLOYÉ : /\ anecdotique car non implémentée

Intention	Contrôle	Événement	Réponse
Valider inscription	Page inscription	Clic	- InscrireEmp(...)

Employé	Employé: valider inscription Employé		- Fermeture page inscription et retour à la page d'accueil - En cas de succès comme de réussite, un popup indiquera l'état de l'inscription
---------	--------------------------------------	--	--

PAGE INSCRIPTION CLIENT :

Intention	Contrôle	Événement	Réponse
Valider inscription Client	Page inscription Client : Valider inscription client	Clic	- inscrireClient(Client) : void -Le client est créer à partir des informations saisies dans différentes cellules - Fermeture page inscription et retour à la page d'accueil - En cas de succès comme de réussite, un popup indiquera l'état de l'inscription

PAGE CLIENT :

Intention	Contrôle	Événement	Réponse
Consulter profil astral	Page Client : Afficher le profil astral du client dans une cellule dédiée	Clic	Appel au service qui fournit le profil astral du client et l'affiche dans la division dédiée : -obtenirProfilAstralClient(Client) : List<String>
Choisir un médium pour effectuer une consultation	Page Client : Charger les profil des médium et les afficher sur une page demande consultation Client	Clic	- obtenirListeDesMediums() : List<Medium>
Consulter historique	Page Client : Consulter l'historique des consultations	Clic	-obtenirHistoriqueConsultationClient(Client) : List<Consultation> Affichage dans une div, la liste des consultations effectuées.

PAGE DEMANDE CONSULTATION CLIENT :

Intention	Contrôle	Événement	Réponse
Validation consultation	page demande consultation Client : Valider le medium sélectionner et envoyer une demande de consultation	Clic	- demanderConsultation(Medium, Client) : boolean -trouverMediumParDenomination(String) : Medium Un menu déroulant est accessible sous la description des médiums où toutes les dénominations sont renseignées. La sélection du médium pour la consultation se fera à l'aide de ce menu déroulant. Si aucune consultation n'est possible, la consultation est refusée et un popup apparaît en l'indiquant. Dans le cas

			<p>contraire, un popup apparaît pour confirmer que la demande de consultation a été prise en compte.</p> <p>Retour à la Page Client</p>
Afficher les médiums	Depuis la Page Client : Charger les profil des médium et les afficher sur une page demande consultation Client	Automatique nt à l'arrivée sur la page	- obtenirListeDesMediums() : List<Medium>

PAGE EMPLOYÉ :

Intention	Contrôle	Événement	Réponse
Consulter historique	Page Employée : Consulter l'historique des consultations	Clic	-obtenirHistoriqueConsultationEmploye(Employe) : List <Consultation> Affichage dans une div, la liste des consultations effectuées.
Afficher demande de consultation	Page Employée : Affichage de la consultation si l'employé a été désigné pour effectuer cette dernière	Automatique	<p>Affichage de la demande de consultation avec son résumé (profil astral, historique de consultations et commentaires des consultations précédentes).</p> <p>-obtenirConsultationCouranteEmploye(Employe) : Employe</p> <p>Apparition du bouton "valider demande de consultation"</p>
Valider demande de consultation	Page Employée : Valider la demande de consultation	Clic	<p>-validerConsultation(Employe) : boolean</p> <p>-Envoyer SMS Client</p> <p>-Ouverture d'une page consultation employée et y affiche les informations relatives à la consultation. Dans cette page, il y a la présence d'une zone de saisie de texte servant à récupérer les commentaires en direct de l'employé</p>
Consulter stat	Page Employée :	Clic	Ouverture d'une page stats et chargement des statistiques qui vont ensuite être affichés sur la page

PAGE CONSULTATION EMPLOYÉ :

Intention	Contrôle	Événement	Réponse
Obtenir Aides info client	page consultation employée	Clic	<p>-obtenirInformationConsultationEmploye(Employe) : String</p> <p>-obtenirAidePredictions(Employe, int, int, int) : List<String></p> <p>-Affiche l'aide au médium pour la</p>

			consultation
Terminer consultation	page consultation employée	Clic	-terminerConsultation(Employe, String) : boolean -Met à jour automatiquement le commentaire de la consultation avec les informations présentes dans la cellule commentaire -retour à la Page Employée

PAGE STATS :

Intention	Contrôle	Événement	Réponse
Afficher les stats	Action activer depuis la page employé	Automatiquement au chargement de la page	-obtenirConsultationsRefusees() : List <Consultation> -obtenirTopCinqMedium() : List <Medium> -obtenirMoyenneParGenre(String) : double
Retour page employé	Page stat : Retourner à la Page Employée	Clic	Retour à la Page Employée

Figure 2 : tableau

Conception de l'IHM

Voici une conception de l'IHM que nous proposons: les cadres bleus sont des zones de texte à remplir, les cadres verts sont des boutons qui appellent un ou plusieurs service(s) métier(s). Les cadres rouges sont des boutons qui dirigent vers une autre page, et appellent parfois un service métier.

La page d'accueil est commune pour les clients et les employés. C'est sur cette page qu'ils pourront choisir de se connecter en tant que client, ou bien en tant qu'employé. Un nouveau client peut accéder à la page d'inscription depuis cette page.

The wireframe shows a login interface for 'Prédic't'IF'. It is divided into two main sections by a horizontal line. The top section contains three input fields: 'Mail' (blue border), 'Mot de passe' (blue border), and 'Se créer un compte client' (red border). To the right of the 'Mail' and 'Mot de passe' fields are two green buttons labeled 'Connexion client' and 'Connexion employé' respectively. The bottom section contains a welcome message 'Bienvenue sur Prédic't'IF!' and a prompt 'Vous pouvez vous connecter pour accéder aux services'. Below the wireframe, the text 'Page d'accueil' is written in cursive.

Sur la page d'inscription, les clients doivent renseigner toutes les informations nécessaires puis cliquer sur le bouton "Valider inscription" pour s'inscrire.

Une fois connecté, le client peut voir son profil astral et voir son historique de consultations : ces informations apparaissent dans la zone d'affichage lorsqu'un des boutons est appuyé. Il est aussi possible d'accéder à la page de demande de consultation.

Sur la page de demande de consultation, le client a accès à la liste de tous les médiums et peut en choisir un avec qui réaliser sa consultation.

Pour les employés, s'il y a une consultation qui leur a été attribuée, elle s'affiche sur leur page, en plus des informations liées au client (son profil astral et son historique de consultations). Les employés peuvent ensuite valider la consultation depuis cette page, ou bien voir les statistiques de l'application et leur historique de consultations. Lorsque l'employé souhaite terminer la consultation, l'employé peut écrire un commentaire puis terminer la consultation avec le bouton "Terminer consultation".

Nom	Vous y êtes presque ! Plus que quelques informations à fournir pour connaître votre avenir !
Prénom	
Mail	
Mot de passe	
Date de naissance	
Adresse postale	
Téléphone	
Valider inscription	

Page inscription client

Consulter mon profil astral	Consulter mon historique de consultations
Demander une consultation avec un de nos médiums	
Zone d'affichage	

Page client

Demander une consultation		
Spiritistes : zone d'affichage	Cartomanciens : zone d'affichage	Astrologues : zone d'affichage
Menu déroulant pour choisir médium		
Demander consultation		

Page de demande consultation

Affichage de la consultation
s'il y en a une

Valider consultation (s'il y en a une)

Consulter mon historique

Voir les stats

Page employé

Affichage aide prédiction

obtenir aide info client

Commentaire de la consultation

Terminer consultation

Page consultation employé

Affichage des
stats de
Prédiet 'IF

Retour page employé

Page stats