

# Rapport DASI

## Introduction :

Le but du logiciel présenté dans ce rapport est de permettre le bon fonctionnement d'une entreprise permettant l'envoi régulier de prédiction à ses clients via une interface web. L'utilisateur peut ainsi traiter les réceptions des inscriptions et les traiter ensuite à l'aide des IHM que nous avons conçues.

Dans le cadre de PredictIf, chaque client est attribué à un employé, qui devra ensuite sélectionner les prédictions qu'il destine à son client parmi une base de données chargée dans le logiciel au préalable, en fonction du signe astrologique du client. Notons que la dimension scientifique est ici mise de côté, bien que l'offre élude ce détail, les prédictions sont choisies arbitrairement par l'employé et ne sont en aucun cas relatives à une conjoncture spatiale planétaire ou autre critère attrayant à l'astronomie.

## Conception du logiciel :

La conception a été réalisée dans le but de faciliter la persistance des objets dans une base de données, voici tout d'abord la liste de ces derniers :

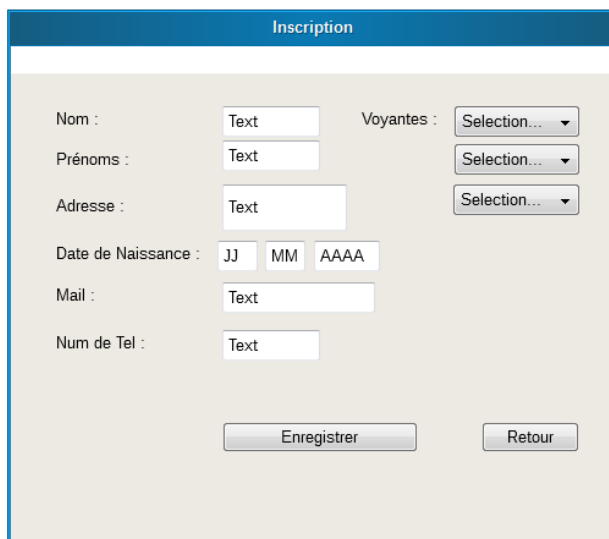
- Client : La liste de clients est mise à jour au fil des inscriptions
- Employé : Chaque employé est enregistré dans la base de données, l'attribution des clients est automatique lors de l'inscription.
- Horoscope : C'est le produit final « livré » au client, il contient les prédictions choisies par l'employé pour un client particulier.
- Medium : Egalement enregistrés au préalable dans la base de données, les mediums sont totalement virtuels, leur nom est en fait signataire de chaque horoscope afin de légitimer ceux-ci auprès des clients.
- Prédiction : Déclinées en trois catégories, amour, travail et santé, ce sont elles qui constituent un horoscope. Elles ont également une « force » qui leur est affectée, mesure de leur bonne augure.

-Zodiac : Il est composé des douze signes bien connus, pour plus de simplicité au niveau de l'implémentation, chaque signe est associé à un mois.

L'architecture de l'application est fournie en annexe.

## Présentation des IHM :

La première IHM présentée est celle conçue pour que le client puisse s'abonner à PredictIf et ainsi bénéficier des services décrits ci-dessus. Elle lui permet d'indiquer son Nom, Prénom, Adresse, Date de Naissance, son mail et son numéro de téléphone, mais également de choisir trois voyants favoris qui seront les rédacteurs (présumés) de ses horoscopes.



Une fois toutes les cases remplies, il peut valider son inscription à l'aide du bouton « enregistrer ».

Une autre interface est celle qu'utilisent les employés, afin de confectionner les horoscopes destinés aux clients. La première étape consiste à choisir le client auquel est dédié l'horoscope parmi la liste des clients affectés à l'employé, et qui sont à même de recevoir un horoscope.

L'interface se présente comme suit :

The screenshot shows a window titled "Choix client". Inside, there is a label "Liste Clients:" followed by a list box containing three items: "Item1", "Item2", and "Item3". Below the list box is a button labeled "Realiser horoscope pour \*nomclient\*".

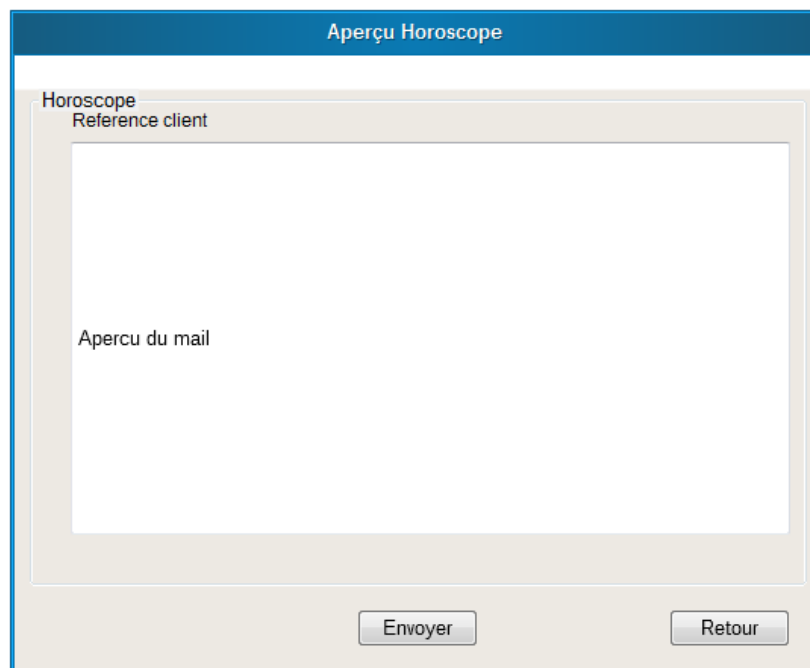
L'employé sélectionne alors le client de son choix, puis clique sur le bouton « Realiser horoscope pour « nomclient » » afin d'accéder à l'interface ci-dessous :

The screenshot shows a window titled "Realisation horoscope". Inside, there is a section labeled "Horoscope" with a sub-label "Reference client". Below this, there are three input fields: "Numéro Prédiction: travail" with a text box containing "n°", "Numéro Prédiction: santé" with a text box containing "n°", and "Numéro Prédiction: amour" with a text box containing "n°". Below these is a label "Voyante :" followed by a dropdown menu showing "Voyante". To the right of these fields is a label "Prédictions disponibles:". Below this label, there are three list boxes: "Travail" containing "Item1", "Item2", and "Item3"; "Santé" containing "Item1", "Item2", and "Item3"; and "Amour" containing "Item1", "Item2", and "Item3". At the bottom of the window are two buttons: "Valider" and "Retour".

---

A ce stade, il peut sélectionner une prédiction de chaque type parmi celles contenues dans les listes Travail, Santé et Amour en entrant le numéro correspondant, ainsi que la Voyante signataire de l'horoscope, prise parmi les Voyantes favorites du client. Il peut enfin valider pour avoir un aperçu du mail, généré automatiquement à partir des données apportées par

la validation :



The screenshot shows a software window titled "Aperçu Horoscope". Inside the window, there is a section labeled "Horoscope" with a sub-label "Reference client". Below this is a large, empty rectangular box labeled "Apercu du mail". At the bottom of the window, there are two buttons: "Envoyer" on the left and "Retour" on the right.

Si le mail lui convient, il peut alors cliquer sur « Envoyer » afin de valider son travail, et permettre au client de recevoir son horoscope.

Note : le bouton retour permet de retourner directement au choix du client, en annulant tout choix de prédiction et tout envoi de mail.

## Présentation des services :

Lors de l'implémentation du logiciel, nous avons inclus une classe « service voyance », pouvant être appelée par les autres classes et fournissant des informations utiles. Ces différents services sont listés ci-dessous :

- public static String horoscope(int numClient, int numPa, int numPs, int numPt):  
Renvoie un horoscope formaté à partir des informations passées en paramètre.
- public static void creerZodiac(Zodiac z): permet de sauvegarder le zodiac passé en paramètre dans la base de données.
- public static void creerMedium(Medium m): permet de sauvegarder le medium passé en paramètre dans la base de données.
- public static void creerPrediction (Prediction p): permet de sauvegarder la prédiction passée en paramètre dans la base de données.
- public static void creerEmploye (Employe e): permet de sauvegarder l'employé passé en paramètre dans la base de données.
- public static void creerClient(Client c): permet d'assurer la persistance d'un client dans la base de données après son inscription.
- public static Client idClient(int id) : renvoie le client référencé à l'id passée en paramètre
- public static PredictionAmour idPredAmour(int id): renvoie la prédiction Amour référencée à l'id passée en paramètre.
- public static PredictionSante idPredSante(int id): renvoie la prédiction Sante référencée à l'id passée en paramètre
- public static PredictionTravail idPredTravail(int id): renvoie la prédiction Travail référencée à l'id passée en paramètre.
- public static List<Client> listeClient(): Renvoie la liste de tous les clients présents dans la base de données.
- public static List<Client> listeClientNom(String nom): Renvoie la liste de tous les clients portant le nom passé en paramètre.
- public static List<Zodiac> listeZodiac(): Renvoie la liste de tous les signes du zodiac présents dans la base de données.

- public static List<Medium> listeMedium(): Renvoie la liste de tous les médiums présents dans la base de données.
- public static List<Medium> listeMediumClient(int numclient) : Renvoie la liste des médiums favoris du client référencé au numéro passé en paramètre.
- public static List<Prediction> listePrediction () : Renvoie la liste de toutes les prédictions présents dans la base de données.
- public static List<Prediction> listePredictionforce (int force): renvoie la liste de toutes les prédictions dont la force est égale au numéro passé en paramètre.
- public static List<Prediction> listePredictionAmour () : Renvoie la liste de toutes les prédictions Amour présents dans la base de données.
- public static List<Prediction> listePredictionTravail () : Renvoie la liste de toutes les prédictions Travail présents dans la base de données.
- public static List<Prediction> listePredictionSante () : Renvoie la liste de toutes les prédictions Santé présents dans la base de données.
- public static List<Employe> listeEmploye(): Renvoie la liste de tous les employés présents dans la base de données.
- public static Zodiac compatible (int mois) : renvoie le zodiac correspondant au numéro du mois passé en paramètre.
- public static List<Medium> creationListe() : crée une liste de médiums aléatoirement, cette liste peut contenir de 2 à 5 médiums (destinés à être affectés à un client).
- public static Employe affectationEmploye() : Renvoie un employé destiné à être affecté à un client, en fonction du nombre de client que doit gérer chaque employé.

