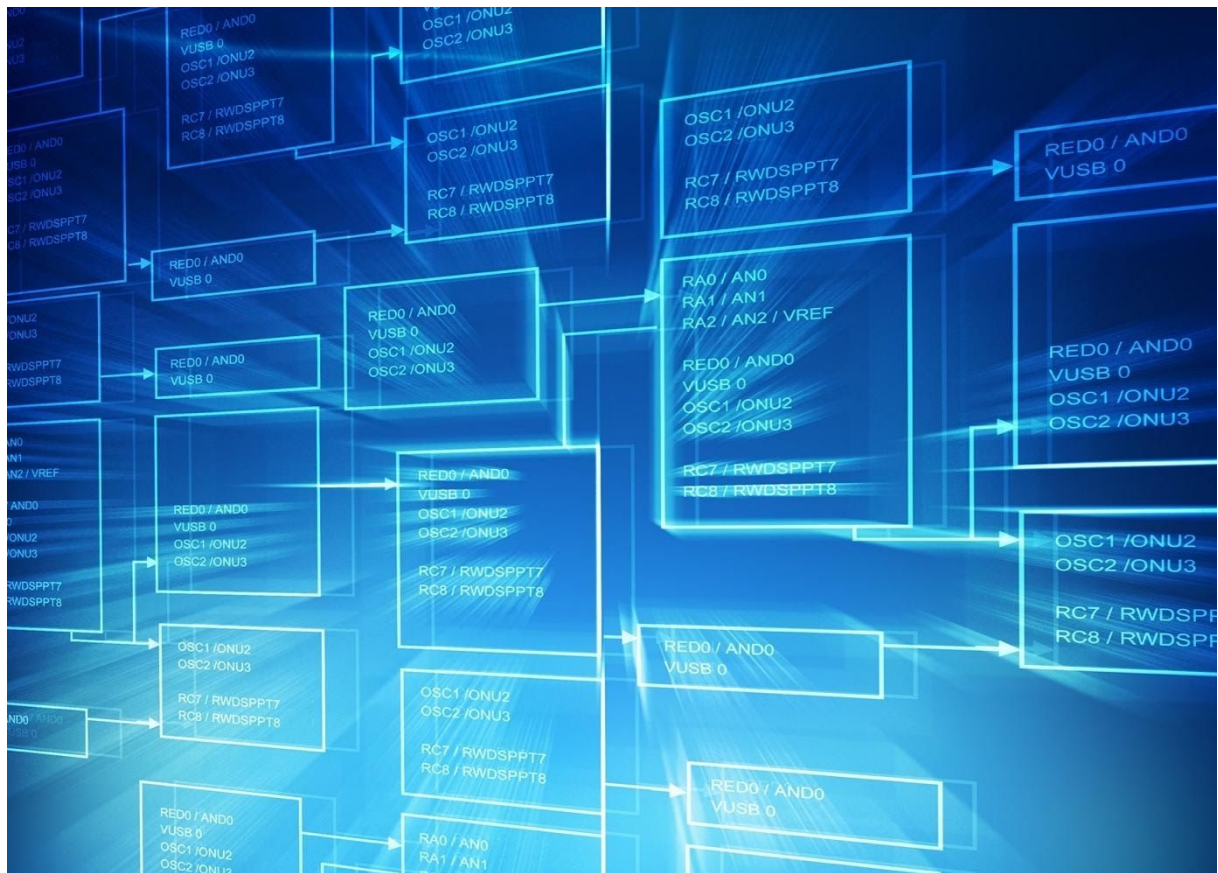


# Application de gestion de RDV

## PsyMeeting

Rapport de Projet – Base De Données 2 – 2019/2020 - L3 P2022

Gabriel Dugny, Thomas Guillaume, Yorane Doare



<b>Introduction</b>	<b>2</b>
<b>Partie I : Description du projet</b>	<b>3</b>
Le MCD (Modèle conceptuel de données)	3
Présentons ce MCD :	3
Le dictionnaire de Données	5
MLD (Modèle Relationnel)	6
Diagramme de classe	7
Choix des Frameworks	8
<b>Partie II : Scénario d'exécution de l'application</b>	<b>9</b>
Scénario	9
Étape 1 :	9
Étape 2 :	9
Étape 3 :	10
Étape 4 :	12
Étape 5 :	13
Étape 6 :	14
Fonctionnalités envisagées et non terminées	16
<b>Partie III : Difficultés rencontrées et les points forts et faibles du projet</b>	<b>17</b>
Difficultés et points faibles du projet	17
Points forts du projet	17
<b>Conclusion :</b>	<b>17</b>

## Introduction

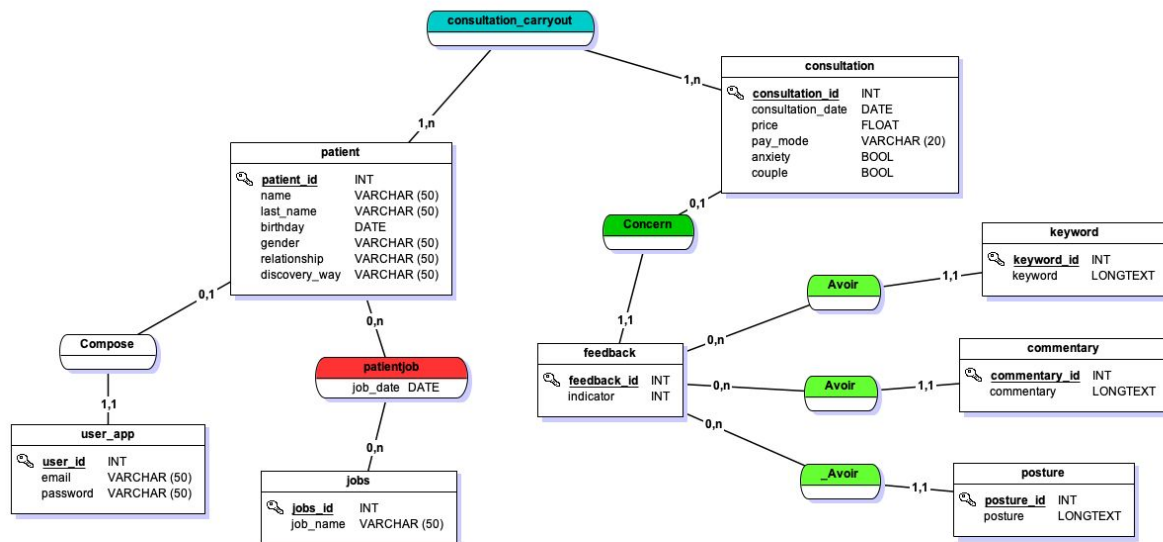
Dans le cadre de notre projet de base de données, nous avons pour but de concevoir une application destinée à une psychologue souhaitant gérer ses propres rendez-vous avec ses patients. Pour les métiers de ce type, un simple cahier pour écrire les informations d'un patient et noter les rendez-vous ne suffit plus. Les possibilités qu'offre le numérique, permettent à la psychologue de simplifier son quotidien. Nous allons donc à travers ce projet permettre à une psychologue de pouvoir enregistrer un rendez-vous de consultation pour un patient à travers notre application. De plus, il sera également possible qu'un patient consulte ses rendez-vous passés et futurs. La psychologue et les patients devront s'authentifier pour se connecter à l'application. Tout au long de ce rapport nous allons donc vous expliquer nos choix et la démarche que nous avons suivi pour la conception de cette base de données.

## Partie I : Description du projet

Cette partie assez descriptive va s'intéresser au début de la conception de ce projet.

### I. Le MCD (Modèle conceptuel de données)

Dans un premier temps, nous avons commencé par l'élaboration du MCD, cette étape est primordiale pour la réalisation de la base de données et demande de bien prendre son temps et de réfléchir longuement pour sa conception. En effet, il s'agit d'une étape conceptuelle dans laquelle on recense les interactions logiques entre les données.



### Présentons ce MCD :

Tout d'abord nous avons choisis, pour construire ce modèle conceptuel de données, **7 entités** ainsi que **7 associations**.

Pour ce qui concerne les cardinalités, pour **chaque couple entité-association** les cardinalités représentent le **nombre minimum et maximum d'occurrences de l'association** pouvant exister à un instant quelconque pour une occurrence de l'entité. Ainsi, expliquons ce MCD en détail :

L'entité **patient**, représente le client (le patient) venu pour une consultation avec la psychologue. Cette entité regroupe toutes les informations primaires concernant le patient (son nom, sa date de naissance, son genre, son statut (célibataire/marié/veuf),etc...). Nous avons décidé de mettre deux autres entités pour ce qui concerne l'emploi du patient **jobs** mais également pour les identifiants de connexion à l'application **user\_app**.

Pour les cardinalités, un patient à travers l'association **compose** peut avoir 0 ou 1 compte maximum dans **user\_app**. A l'inverse, un compte ne peut être affecté qu'à une seule personne (1,1).

L'entité **consultation**, regroupe le numéro de consultation, ainsi que toute les information la relative à la consultation (la date, le prix, le moyen de paiement, l'anxiété du patient, ...).

L'entité **feedback** : correspondant aux informations relatives après la séance. L'entité contient un numéro d'identification mais aussi un indicateur ( note de 0 à 10 que le patient s'attribue). De plus chaque entité **feedback** est reliée à une ou plusieurs entités **commentary**, **posture**, **keyword**, désignant les informations reportés par la ou le psychologue.

Pour les cardinalités, il est logique de dire qu'un patient peut prendre au minimum une ou plusieurs consultations, et à l'inverse une consultation peut avoir pris en charge un ou plusieurs patients. De plus, à la fin d'une consultation il peut y avoir zéro ou un seul feedback. Et bien sûr, un feedback concerne une et une seule consultation.

## II. Le dictionnaire de Données

Grâce au logiciel **JMerise**, avec lequel nous avons constitué notre MCD, nous avons pu établir le dictionnaire de nos données. JMerise a récolté nos informations et les a classés.

### Dictionnaire de données :

Num	Nom	Code	Type	Taille	Utilisation
0	id_patient	ID_PATIENT	Int		false
1	nom	NOM	Varchar	50	false
2	patient_id	ID_PATIENT	Int		true
3	prenom	PRENOM	Varchar	50	false
4	date_naissance	DATE_NAISSANCE	Date		false
5	genre	GENRE	Varchar	50	false
6	situation_sociale	SITUATION_SOCIALE	Varchar	50	false
7	consultation_id	CONSULTATION_ID	Int		true
8	date	DATE	Date		false
9	prix	PRIX	Float		false
10	moyen	MOYEN	Varchar	50	false
11	indicateur	INDICATEUR	Int		false
12	profession_id	PROFESSION_ID	Int		false
13	profession_nom	PROFESSION_NOM	Varchar	5	false
14	date_debut	DATE_DEBT	Date		false
15	date_fin	DATE_FIN	Date	5	false
16	comportement_id	COMPORTEMENT_ID	Int		false
17	commentaire	COMMENTAIRE	Varchar	50	false
18	posture_id	POSTURE_ID	Int		true
19	posture_nom	POSTURE_NOM	Varchar	50	false
20	mot_cle_id	MOT_CLE_ID	Int		false
21	mot_cle	MOT_CLE	Varchar	50	false
22	mail	MAIL	Varchar	50	false
23	password	PASSWORD	Varchar	50	true
24	info_id	INFO_ID	Int		false
25	commentaire	COMMENTAIRE	Blob		false
26	commentaire	COMMENTAIRE	Longtext		false
27	name	NAME	Varchar	50	true

Num	Nom	Code	Type	Taille	Utilisation
28	lastname	LASTNAME	Varchar	50	false
29	birthday	BIRTHDAY	Date		true
30	email	EMAIL	Varchar	50	true
31	administrateur_id	ADMINISTRATEUR_ID	Int		false
32	personne_id	PERSONNE_ID	Int		false
33	person_id	PERSONNE_ID	Int		false
34	administrator_id	ADMINISTRATEUR_ID	Int		false
35	jobs_id	JOBS_ID	Int		true
36	job_name	JOB_NAME	Varchar	50	true
37	job_date	DATE	Date		true
38	consultation_date	CONSULTATION_DATE	Date		true
39	price	PRICE	Float		true
40	paymode	PAYMODE	Varchar	50	false
41	indicator	INDICATOR	Int		true
42	feedback_id	FEEDBACK_ID	Int		true
43	commentary	COMMENTARY	Longtext		true
44	keyword	KEYWORD	Longtext		true
45	posture	POSTURE	Longtext		true
46	last_name	LAST_NAME	Varchar	50	true
47	gender	GENDER	Varchar	50	true
48	relationship	RELATIONSHIP	Varchar	50	true
49	hours	HOURS	Time		false
50	user_id	USER_ID	Int		true
51	pay_mode	PAY_MODE	Varchar	20	true
52	discovery_way	DISCOVERY	Varchar	50	true
53	commentary_id	COMMENTARY_ID	Int		true
54	keyword_id	KEYWORD_ID	Int		true
55	anxiety	ANXIETY	Bool		true

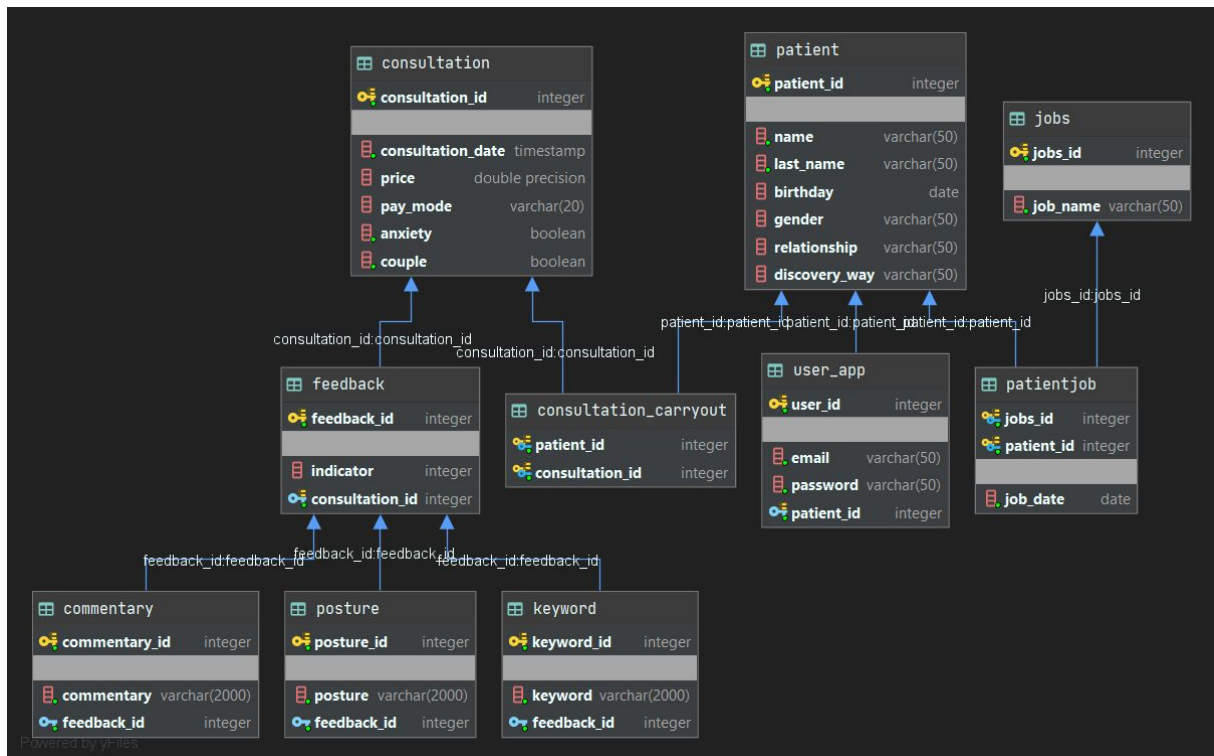
Num	Nom	Code	Type	Taille	Utilisation
56	couple	COUPLE	Bool		true
57	couple	COUPLE	Varchar	50	false

### III. MLD (Modèle Relationnel)

Présentons maintenant notre MLD , ce modèle privilégie les relations.

Pour passer du MCD au MLD la transformation s'appuie avant tout sur les associations, leurs cardinalités ainsi que sur le nombre d'entités concernées

MLD :



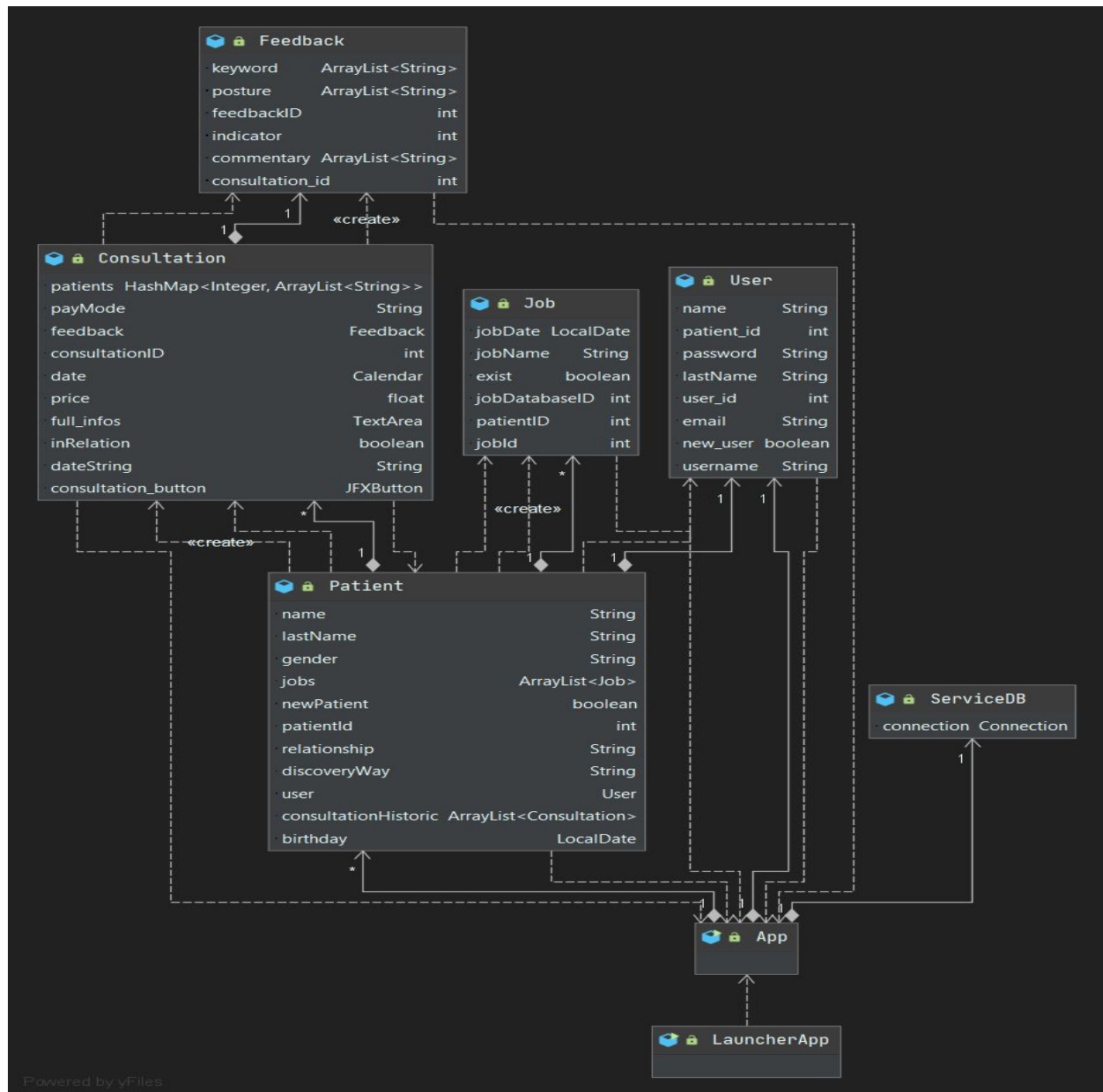
*Le MLD ci-dessus fut généré par l'IDE IntelliJ.*



## IV. Diagramme de classes

Pour ce qui est du diagramme de classes, nous avons opté pour le diagramme représentant les classes du modèle.

Diagramme de classes :



Ci-dessus, ce diagramme nous donne une vue des classes du modèle. Ce sont donc la retranscription des entités de la base de données sous forme de classe Java.

La classe **App** et **LauncherApp** servent uniquement pour le lancement de l'application.

Enfin la classe **ServiceDB** nous sert à gérer les connections avec la base de données et centraliser les demandes de connections.



## V. Choix des Frameworks

Pour ce projet, nous avons utilisé **Java**. Après avoir commencé le projet sur **Java 8**, nous sommes passés à **Java 11** car nous avons décidé de transitionner sur un projet utilisant **Maven**, c'est un gestionnaire des dépendances permettant une meilleure persistance du projet avec une gestion des dépendances automatiques.

En effet notre projet utilise donc différentes librairies :

- Pour l'interface graphique utilisant la librairie **JavaFX**. De plus, nous avons consolidé notre application avec des librairies graphiques complémentaires à **JavaFX** comme **JFoenix** (pour le style *Material Design*) et aussi **Ikonli** qui gère les icônes.
- Pour la gestion de la base de données, nous utilisons la dépendance de **PostgreSQL** et **c3p0** pour gérer les pools de connections.

Pour le **SGBD**, nous avons au départ comme demandé, utilisé **Oracle DB**. Dès qu'il a été permis de choisir son moteur de base de données, nous sommes passés sur **PostgreSQL**. Ce SGBD nous a apporté des avantages majeurs : il est open source et il est donc très facile d'intégrer des drivers (là où ceux d'Oracle nécessitent une authentification via Maven et ne sont pas en libre accès), et ses outils d'administrations (PgAdmin) sont là aussi résolument plus modernes. De plus PostgreSQL, nous a permis de mettre notre BDD en ligne grâce au site [ElephantSQL](#), proposant une formule gratuite d'hébergement, largement suffisant pour notre projet.

Remarque :

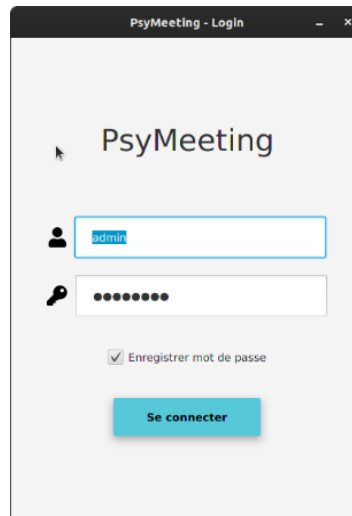
Nous avons déjà bien entamé le projet quand ses modalités ont changé, et nous n'avons pas pu changer de langage pour une technologie avec lesquelles nous étions plus familiers : des technologies Web. En utilisant d'autres technologies, nous aurions été capables d'avoir une interface plus moderne, respectant tous les standards Web modernes, et proposant une interface et une navigation encore plus intuitive.

## Partie II : Scénario d'exécution de l'application

Dans cette partie nous allons vous faire couvrir notre application à travers un scénario d'exécution :

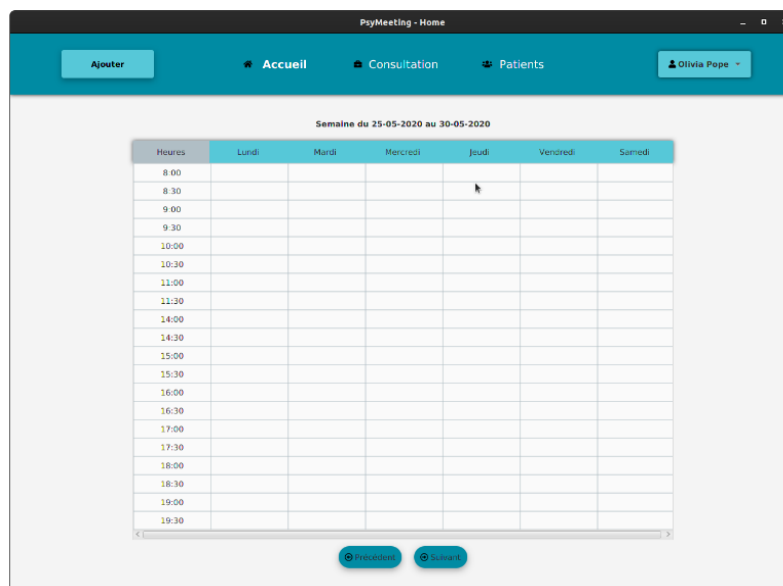
### 1. Scénario

Étape 1 :



La (ou le) Psychologue se connecte à l'application avec ses identifiants (admin et mot de passe). Les identifiants de connections du patient sont créé lors de la création d'un patient avec une authentification par email et mot de passe (généré de façon automatique)

Étape 2 :

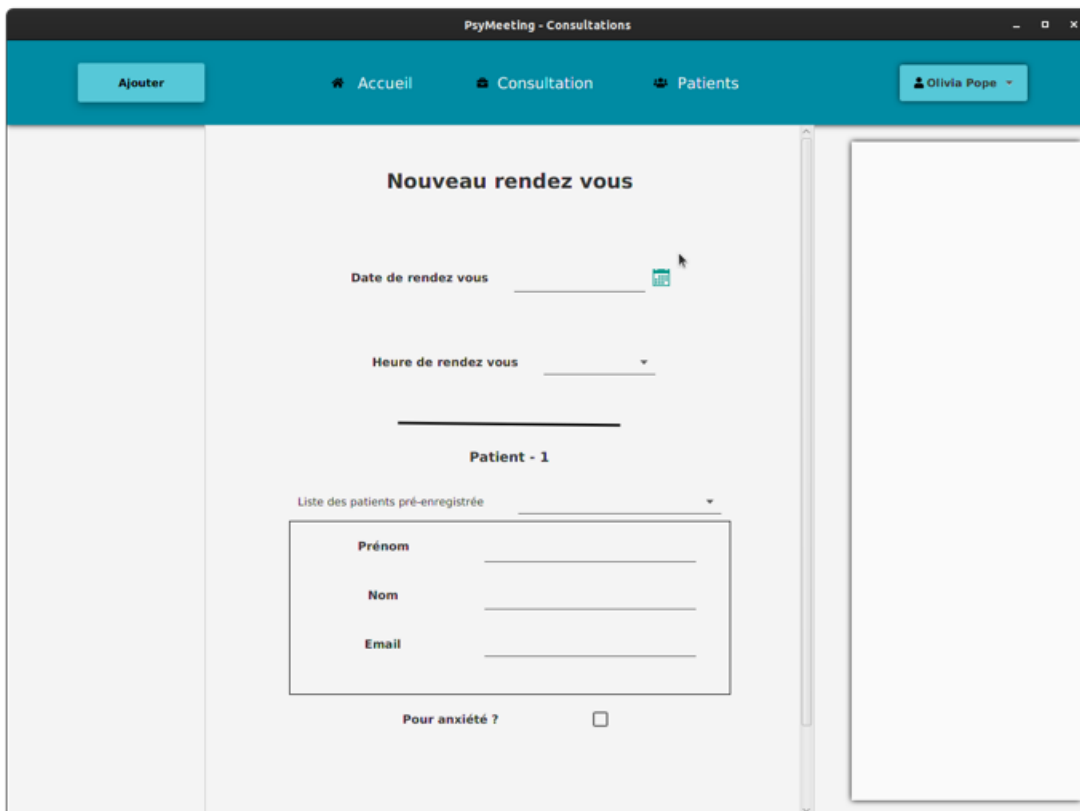


La psychologue accède à son espace, sur sa page d'**Accueil**, un calendrier de la semaine va s'afficher et notre chère psychologue pourra consulter ses consultations présentes, passées ou future. Cette fenêtre d'accueil permet la consultation des rendez-vous sur une semaine choisie par la psychologue.

### Mais comment ajouter une consultation ?

Pour faciliter ce scénario prenons l'exemple d'un couple, M. et Mme Doe, ayant pris un rendez-vous pour une consultation. Ce jeune couple n'a jamais pris de rendez-vous avant.

### Étape 3 :



The screenshot shows a web application window titled "PsyMeeting - Consultations". The interface has a teal header bar with a navigation menu containing "Ajouter", "Accueil", "Consultation", and "Patients". A user profile "Olivia Pope" is visible in the top right. The main content area is titled "Nouveau rendez vous" and contains the following form elements:

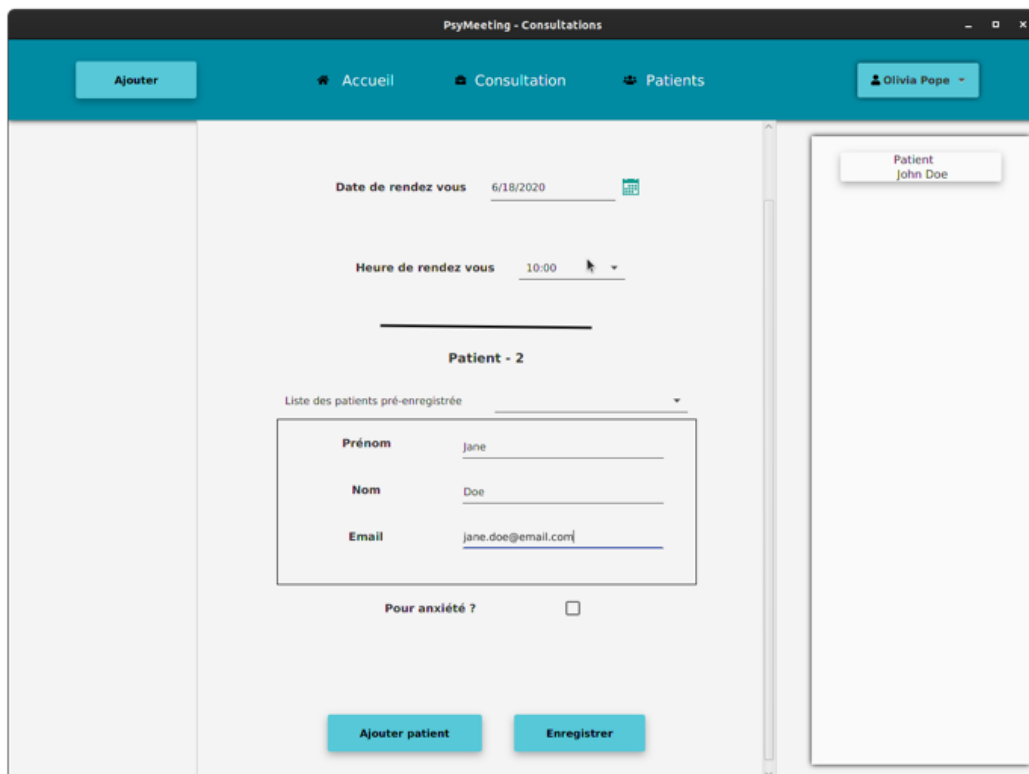
- Date de rendez vous**: A text input field with a calendar icon to its right.
- Heure de rendez vous**: A dropdown menu.
- Patient - 1**: A section header.
- Liste des patients pré-enregistrée**: A dropdown menu.
- Form fields**: A box containing three input fields labeled "Prénom", "Nom", and "Email".
- Pour anxiété ?**: A checkbox.

En cliquant sur l'onglet **Ajouter**, la psychologue pourra ainsi entrer un nouveau rendez-vous, la date pourra être choisie avec l'aide d'un calendrier (c'est toujours plus facile) et elle pourra choisir un créneau d'horaire pour le rendez-vous en fonction des disponibilités.

Les rendez-vous peuvent être pris du lundi au samedi et chaque séance dure 30 minutes. Notre psychologue ne peut pas travailler plus de dix heures par jour ainsi elle ne pourra pas avoir plus de 20 consultations par jour.

Une liste des patients déjà enregistrés dans la base donnée est affichée et permet de simplifier la saisie des patients. Ainsi si un patient récurrent revient, elle aura juste à sélectionner son nom parmi la liste.

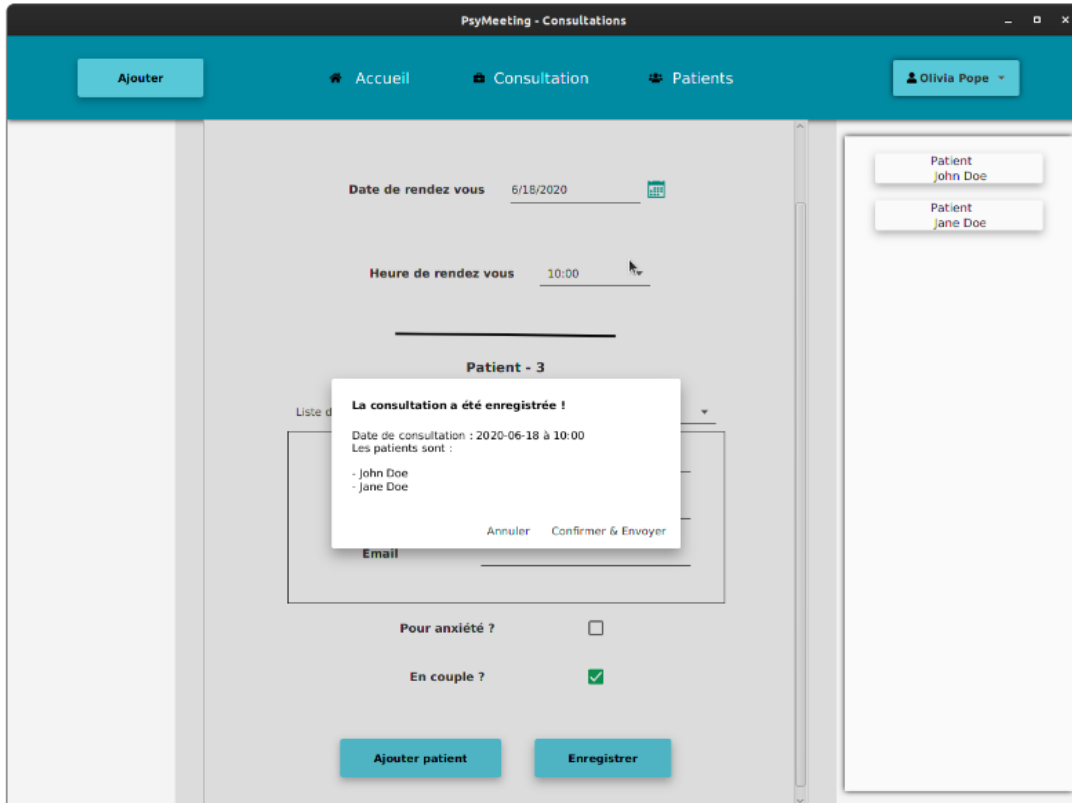
Mais dans notre exemple il est question de M et Mme Doe qui sont nouveaux. La psychologue devra donc rentrer les noms des deux patients dans l'encadré ci-dessus. Un onglet concernant l'anxiété du patient est également disponible. Il lui suffira juste de cocher la case « anxiété ? » relative au patient si celui-ci présente des symptômes anxieux lors de la prise de rendez-vous.



The screenshot shows the 'PsyMeeting - Consultations' web application. The interface has a teal header with a navigation bar containing 'Ajouter', 'Accueil', 'Consultation', and 'Patients' tabs. A user profile 'Olivia Pope' is visible in the top right. The main content area is divided into two sections. The left section contains a form for scheduling a consultation, with fields for 'Date de rendez vous' (6/18/2020), 'Heure de rendez vous' (10:00), and a dropdown for 'Patient - 2'. Below this is a section titled 'Liste des patients pré-enregistrée' with a search bar. A form for adding a new patient is displayed, with fields for 'Prénom' (Jane), 'Nom' (Doe), and 'Email' (jane.doe@email.com). There is also a checkbox labeled 'Pour anxiété ?'. The right section shows a list of patients, with 'John Doe' currently selected and highlighted. At the bottom of the form, there are two buttons: 'Ajouter patient' and 'Enregistrer'.

Une fois les informations de M Doe rentrées, son nom s'affiche dans l'encadré à droite. Comme M et Mme Doe sont deux, la psychologue devra cliquer sur l'onglet « Ajouter patient ». De là, elle pourra remplir les informations de Mme Doe et une case « en couple ? » apparaîtra.

## Étape 4 :

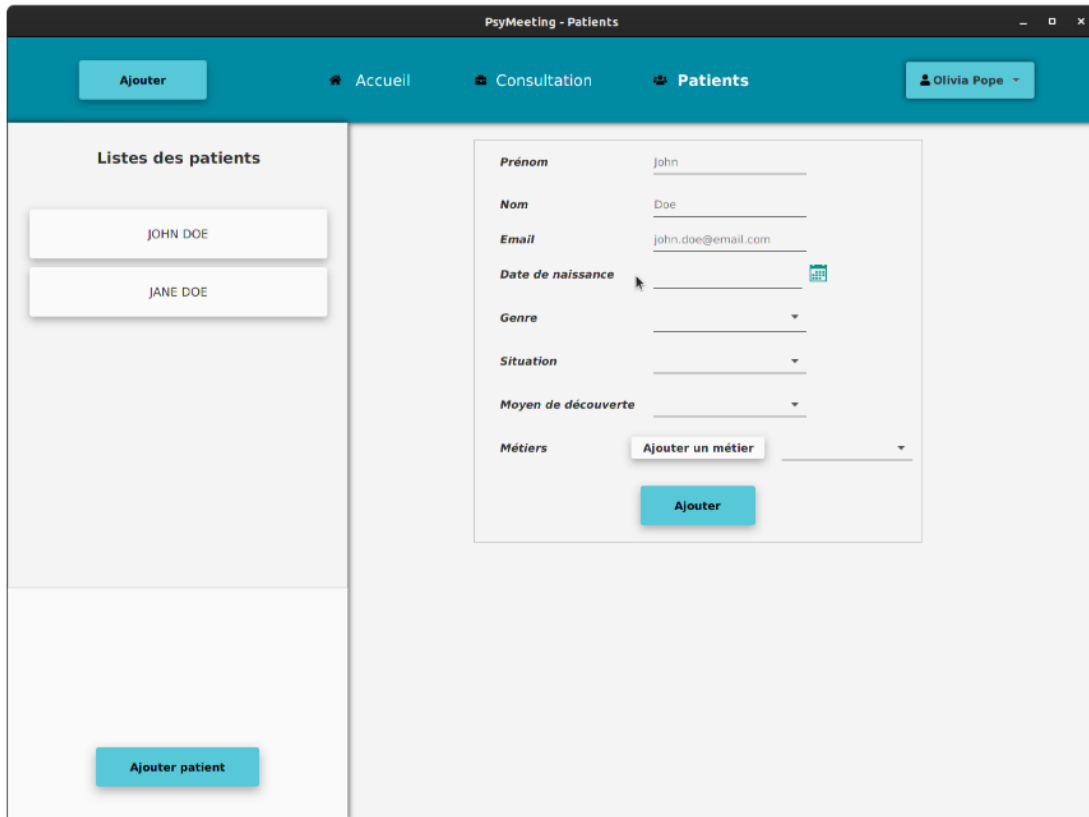


The screenshot shows the 'PsyMeeting - Consultations' web application. The interface includes a top navigation bar with 'Ajouter', 'Accueil', 'Consultation', and 'Patients' tabs, and a user profile 'Olivia Pope'. The main form is for registering a consultation, with fields for 'Date de rendez vous' (6/18/2020) and 'Heure de rendez vous' (10:00). A modal window titled 'La consultation a été enregistrée !' (The consultation has been registered!) is displayed, showing the consultation date and time, and listing the patients: 'John Doe' and 'Jane Doe'. The modal also includes 'Annuler' and 'Confirmer & Envoyer' buttons. Below the modal, there are checkboxes for 'Pour anxiété ?' (unchecked) and 'En couple ?' (checked). At the bottom, there are 'Ajouter patient' and 'Enregistrer' buttons.

La consultation a été enregistré, les patients sont M et Mme Doe et leur consultation est prévue le 18 Juin à 10h00. Ces deux patients vont effectuer une consultation en couple (la psychologue a donc coché la case).

Il n'est en revanche pas possible qu'une consultation concerne 4 personnes. Chaque consultation doit pouvoir accueillir au maximum trois personnes.

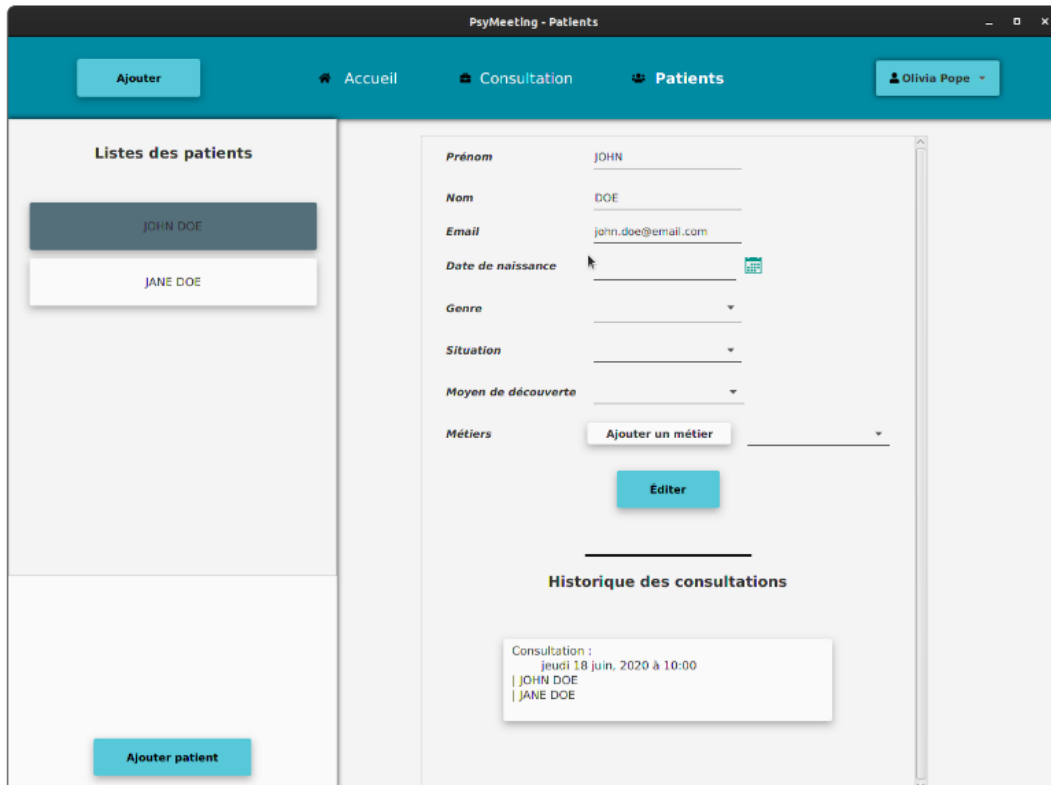
## Étape 5 :



Cliquons maintenant sur l'onglet patient, pour compléter le profil du jeune couple.

En effet, dans cette interface, la psychologue pourra rentrer toutes les informations concernant le couple (son sexe, sa situation, ...).

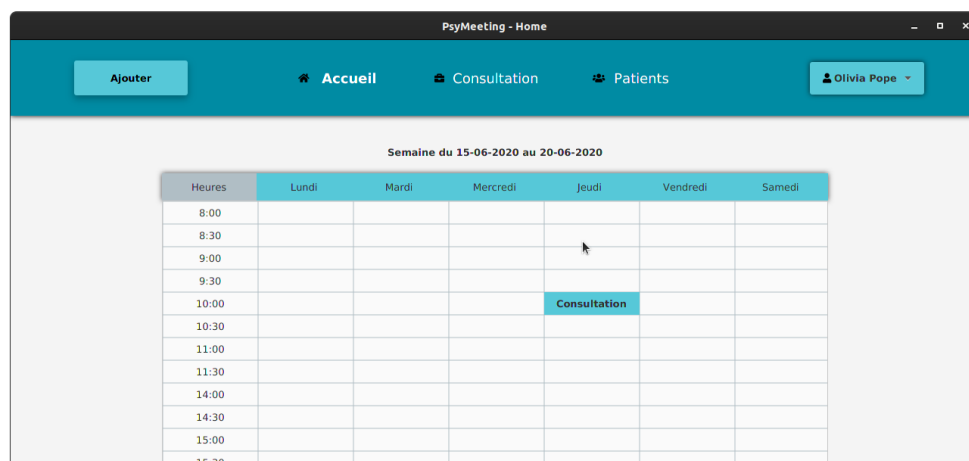
Comme on peut le remarquer, une liste complète de tous les patients s'affiche à gauche de la photo ci-dessus. Ceci permettra également à la psychologue d'avoir accès, en deux clics (depuis la page d'accueil), aux informations des patients de son choix.



De plus, chaque patient possède un historique de consultation. M et Mme Doe également.

Ceci a été fait pour que la psychologue puisse accéder à toutes les informations nécessaires de façon rapide et efficace.

Étape 6 :



Heures	Lundi	Mardi	Mercredi	Jeudi	Vendredi	Samedi
8:00						
8:30						
9:00						
9:30						
10:00				Consultation		
10:30						
11:00						
11:30						
14:00						
14:30						
15:00						
15:30						

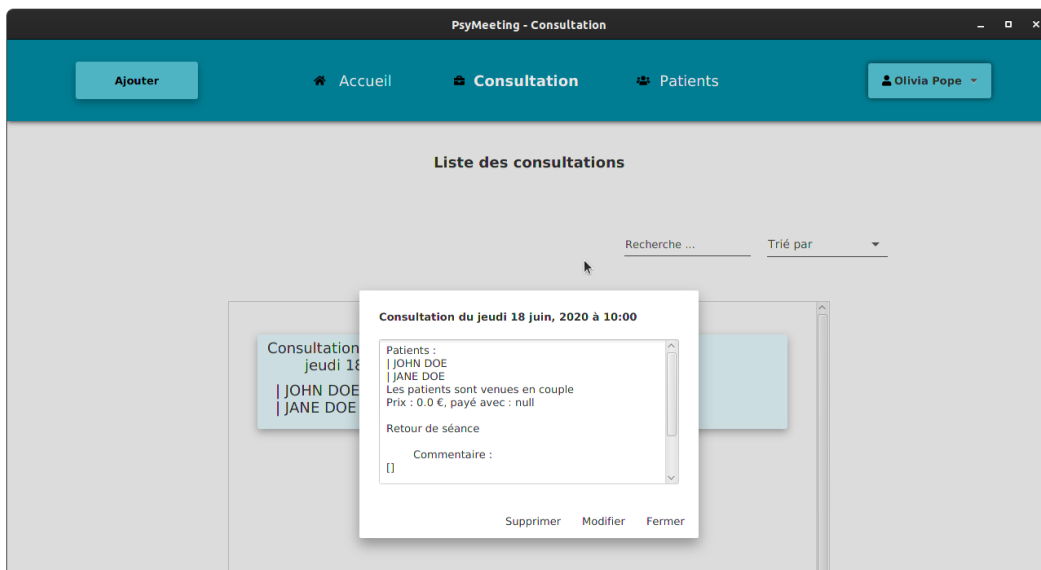
Retournons sur la page d'accueil. On peut voir que la consultation apparaît bien dans le planning de notre psychologue.



Passons maintenant dans l'onglet consultation :



Notre psychologue pourra avoir accès à l'ensemble de ses consultations. De plus, une barre de recherche ainsi que la possibilité de trier ses consultations permettront encore une fois à la psychologue de trouver rapidement les informations qu'elle cherche.



En cliquant sur n'importe quelle consultation. La psychologue aura accès à un résumé de la séance avec :

- Le nom des patients
- Les modalités de leur venu
- Le prix et le moyen de paiement
- Un retour de séance

De plus , la psychologue a la possibilité d'ajouter et annuler un rendez-vous.

## 2. Fonctionnalités envisagées et non terminées

Malgré notre avancé dans le projet, nous n'avons pas pu tout mettre au point. Il reste encore quelques fonctionnalités que nous avons envie d'implanter

Les fonctionnalité envisagé et non terminé sont :

- La modification d'un rendez-vous d'un patient par la psy
- Login avec authentification par email et mot de passe pour le patient. \*
- Incorporation de vue pour l'utilisation de la base de données
- Mise à jour des commentaires du rendez-vous ( nous avons l'interface graphique seulement qui est implantée )

\* Lorsque la psychologue ajoute un patient, son adresse email est stocker dans « user » et un mot de passe de connexion lui est attribué. Cependant nous n'avons pas établi une vue utilisateur sur l'application pour restreindre ses fonctionnalités.

## Partie III : Difficultés rencontrées et les points forts et faibles du projet

### I. Difficultés et points faibles du projet

Lors du projet, nous avons rencontré de multiples difficultés. Déjà d'un point de vue technique, nous avons dû apprendre à utiliser la librairie JavaFX et comprendre son fonctionnement. De plus, dès lors que le projet avançait, des nouveaux problèmes de structure de données sont apparues. C'est pourquoi, nous avons pris du temps à tout structurer en MVC et créer un environnement de travail optimal et simple d'accès en utilisant Maven.

Un autre problème fut aussi de gérer les requêtes SQL tout en maintenant une expérience utilisateur correcte. C'est pourquoi nous avons opté pour le principe de Service qui offre JavaFX, permettant de scinder les requêtes SQL et les interactions utilisateurs en deux Threads distincts et donc ne pas bloquer l'expérience utilisateur.

Enfin un dernier problème fut le déploiement de l'application sous un exécutable .jar, il fallu passer par Maven et faire des configurations pour arriver à obtenir un résultat satisfaisant.

### II. Points forts du projet

Au niveau des points forts, nous sommes d'abord plutôt content d'avoir réussi à construire un projet structuré, répondant aux normes MVC et ayant un support moderne avec Maven.

De plus ce projet fut une bonne expérience pour découvrir le design et comment penser une interface utilisateur.

Enfin l'utilisation de PostgreSQL et d'une base de données en ligne est plutôt satisfaisant car nous avons un produit quasi fini, prêt à être utilisé.

## Conclusion :

Pour conclure, le projet fut pour nous une bonne expérience, nous avons appris beaucoup de nouvelles technologies et nous avons pu créer quelque chose de moderne et dynamique. Nous sommes fiers de notre rendu car nous avons pris beaucoup de temps à peaufiner les détails et donner le plus possible une expérience utilisateur agréable.