



Développement en environnement mobile : Medical Tourism

Rapport final

Auteurs :

- PONCET Alexandre
- PRADÈRE Théo
- RABASSE Grégoire
- THIBAUD Bastien

Encadrants :

- CHOURABI Olfa
- BEN HAMOUDA Iman



Table des matières

I.	Introduction.....	3
1.	Contexte – le tourisme médical	3
2.	Problèmes actuels	3
3.	Solution.....	3
II.	Cahier des charges.....	4
1.	Cahier des charges fonctionnel	4
a.	Fonctionnalités à instaurer dans ce projet.....	4
b.	Fonctionnalités futures pour la prochaine équipe travaillant sur le projet	4
c.	Remarques.....	4
2.	Contraintes techniques	5
a.	Technologies utilisées.....	5
b.	Stockages et serveurs	5
c.	Expérience utilisateur.....	5
d.	Charte graphique.....	6
e.	Bonnes pratiques.....	6
III.	Présentation de l'application – description des fonctionnalités.....	7
1.	Accueil	7
2.	Services.....	8
3.	Docteurs	9
4.	Interprètes.....	12
5.	Rendez-vous	12
6.	Compte	12
IV.	Développement.....	13
1.	Recherche d'un médecin et utilisation des bases de données	13
2.	Envoi d'un mail depuis l'application.....	14
3.	Prise de rendez-vous	15
V.	Conclusion	16

I. Introduction

1. Contexte – le tourisme médical

De plus en plus de personnes n'hésitent pas à prendre l'avion pour aller dans un autre pays se faire opérer ou réaliser des examens, nourrissant un tourisme médical en plein essor. Et pour cause : les chiffres sont alléchants. Le secteur du tourisme médical, pesant déjà plusieurs milliards de dollars, devrait augmenter de 25% par an sur la prochaine décennie, selon une étude récente de Visa et Oxford Economics.

« La demande grandit en partie à cause du vieillissement de la population, mais aussi à cause d'une classe moyenne croissante dans le monde entier, qui est, par internet, informée qu'il y a des traitements ailleurs pour eux », explique Julie Munro, présidente de la Medical Travel Quality Alliance (MTQUA), qui établit un classement des dix meilleurs hôpitaux pour voyageurs.

Le tourisme médical est loin d'être la chasse gardée de quelques pays ou de se limiter à des habitants de pays riches allant dans des pays moins chers. Etats-Unis, Turquie, Thaïlande, Singapour, mais aussi Espagne ou Allemagne, les cartes montrent des flux dans toutes les directions pour échapper à de longues listes d'attente, accéder à des soins non proposés dans son pays ou trop onéreux.

Le tourisme médical est mondial. Beaucoup d'Européens vont par exemple en Pologne ou en Croatie recevoir des soins dentaires moins chers, mais il y a aussi beaucoup de Russes et d'habitants de la péninsule arabique venant en Europe occidentale afin de bénéficier d'hôpitaux de qualité et de soins qu'ils n'ont pas dans leurs pays.

2. Problèmes actuels

L'organisation d'un voyage médical est très complexe. En effet, il faut :

- Trouver un médecin spécialisé, afin de réaliser les diagnostics/opérations,
- Trouver un interprète afin de faciliter le suivi du patient, et aussi faciliter toutes les démarches administratives,
- Trouver un logement (aux normes pour accueillir des patients),
- Trouver un moyen de transport (par exemple des taxis médicalisés),
- Prendre ses billets d'avion,
- Faire son visa,
- Trouver des activités si le voyage médical est long.

Ceci ressemble à un "parcours du combattant" sachant que la majorité des patients ne parle pas français ce qui alourdit considérablement la charge de travail ainsi que le stress du patient.

3. Solution

La solution proposée dans le cadre de ce projet est la réalisation d'une plateforme mobile pour le tourisme médical. Celle-ci devra comprendre toutes les fonctionnalités possibles afin de faciliter le voyage médical des patients en France.

II. Cahier des charges

1. Cahier des charges fonctionnel

L'application devra à terme contenir toute une série de fonctionnalités. Etant donné la complexité et le nombre de fonctionnalités à instaurer, notre projet s'est concentré sur une partie seulement des fonctionnalités de l'application finale.

a. Fonctionnalités à instaurer dans ce projet

- Trouver un médecin
- Trouver un interprète
- Effectuer une réservation pour un rendez-vous avec un médecin
- Contacter un médecin
- Effectuer une réservation pour un interprète
- Contacter un interprète
- Agenda des disponibilités pour les médecins
- 4 langues disponibles : français, anglais, russe, arabe
- Connexion à l'application par adresse électronique
- Authentification par Facebook ou compte Google+
- Contacter le service client
- Système de rappel des rendez-vous prévus

b. Fonctionnalités futures pour la prochaine équipe travaillant sur le projet

- Trouver un logement
- Réserver un logement
- Trouver un moyen de transport
- Réserver un moyen de transport
- Trouver un(e) infirmier(e)
- Réserver un(e) infirmier(e)
- Paiement en ligne

c. Remarques

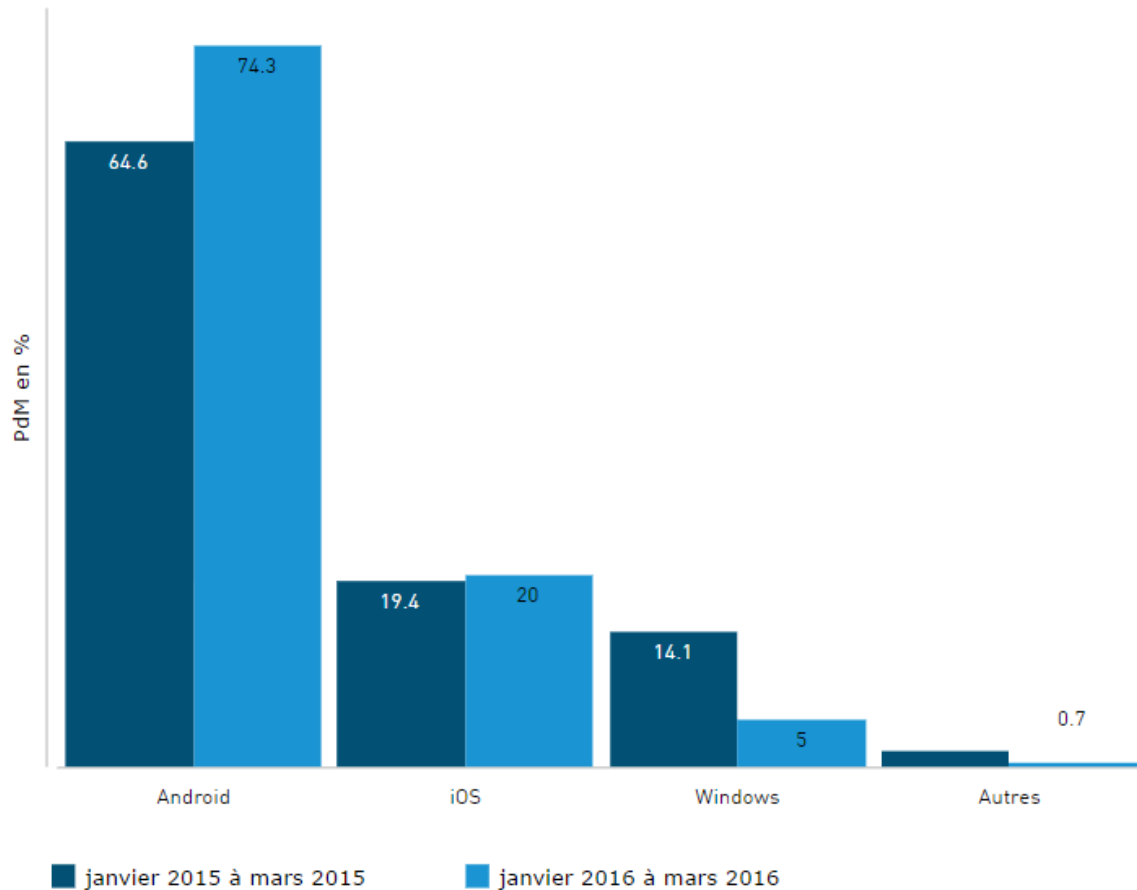
Le dernier point « Paiement en ligne » constituera la plus grande difficulté pour le groupe qui poursuivra le projet. En effet, c'est une partie très complexe, tant sur le plan technique que sur le plan administratif.

Les autres points seront relativement plus « simples » à intégrer à l'application, étant donné que beaucoup de bouts de code sont réutilisables.

2. Contraintes techniques

a. Technologies utilisées

PdM des OS mobile vendus en France (%)



Bien que le système d'exploitation Android représente environ 75% du marché, nous ne pouvons pas nous permettre de délaissier 25% de part de marché. Ainsi, nous avons dû choisir des langages et supports disponibles à la fois sur Android et iOS.

Deux choix s'offraient à nous de développer l'application sur les deux systèmes d'exploitation à la fois : React Native ou Ionic.

Nous avons fait le choix d'utiliser Ionic, qui s'appuie sur Angular, un framework de JavaScript.

b. Stockages et serveurs

L'application, qui utilise Internet, doit avoir accès à une base de données. Celle-ci doit donc être disposée sur un serveur hébergé par IMTP ou sur un cloud. Comme l'application touche à des données médicales pouvant être sensibles, un serveur privé est la solution recommandée.

c. Expérience utilisateur

Plusieurs études ont été réalisées par des cabinets de conseil et d'audit. Les résultats ont permis de mettre en évidence un parcours utilisateur « type ». Il ressort de ces études plusieurs onglets dans

l'application : une page d'accueil, un onglet regroupant tous les services, un onglet consacré à l'agenda et aux rendez-vous, et un dernier onglet concernant les paramètres et préférences de l'application.

Les médecins et interprètes doivent chacun avoir une page de profil avec un certain nombre de renseignements : leur localisation, les prestations qu'ils proposent, les tarifs, leur CV, et un système de notation.

d. Charte graphique

Pour la partie graphique, nous avons été épaulés par un graphiste. Celui-ci nous a fourni les logos, les couleurs à utiliser, les images, etc.

e. Bonnes pratiques

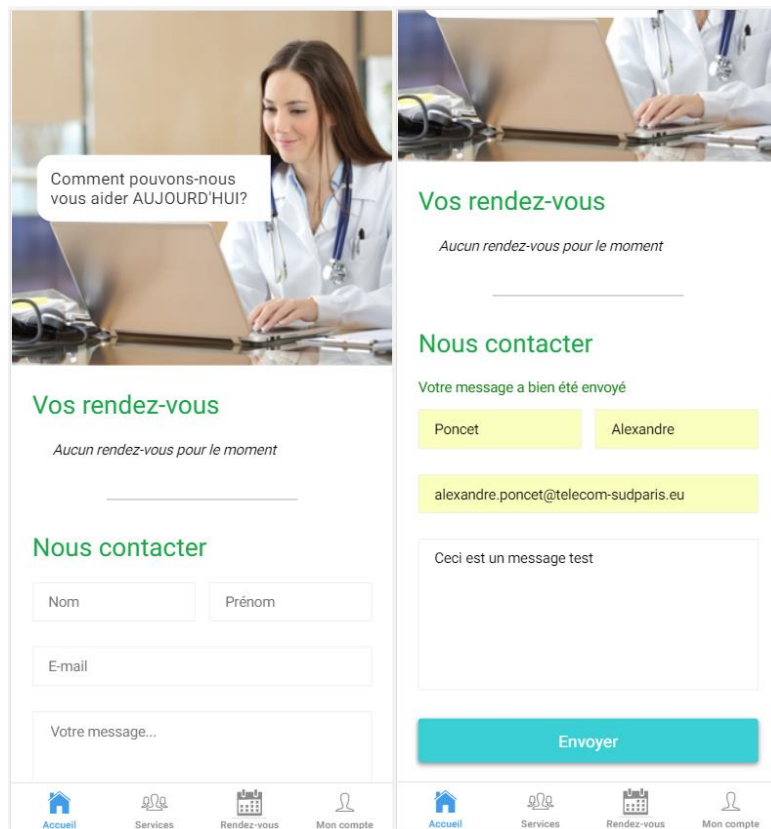
L'application doit être codée proprement afin de permettre sa maintenance, et aussi son évolution par les prochaines équipes travaillant dessus. Cela se matérialise par une manière répétitive de programmation (réutilisation au maximum) du code existant.

De plus, les technologies utilisées doivent être récentes et bien documentées, afin d'également favoriser la maintenance et l'évolution de l'application.

III. Présentation de l'application – description des fonctionnalités

Cette partie a pour but de présenter l'application ainsi que ses diverses fonctionnalités. Elle servira aussi de manuel utilisateur. L'application comprend 4 onglets : Accueil, Services, Rendez-vous, Mon compte.

1. Accueil

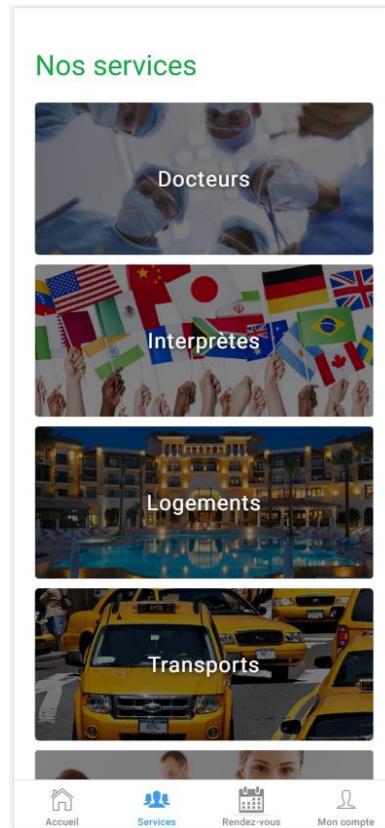


Sur la page d'accueil, il est possible de contacter le service client d'ITMP en envoyant un e-mail. Pour le moment, l'adresse de réception est celle d'un membre du groupe, mais il est possible de la remplacer par une adresse email d'un membre d'ITMP. **Cette fonctionnalité est opérationnelle.**

2. Services

Cet onglet regroupe tous les services proposés par ITMP : Docteurs, Interprètes, Logements, Transports et Infirmier(e)s.

Comme mentionné dans le cahier des charges, seules les parties Docteurs et Interprètes ont été développées et sont fonctionnelles.



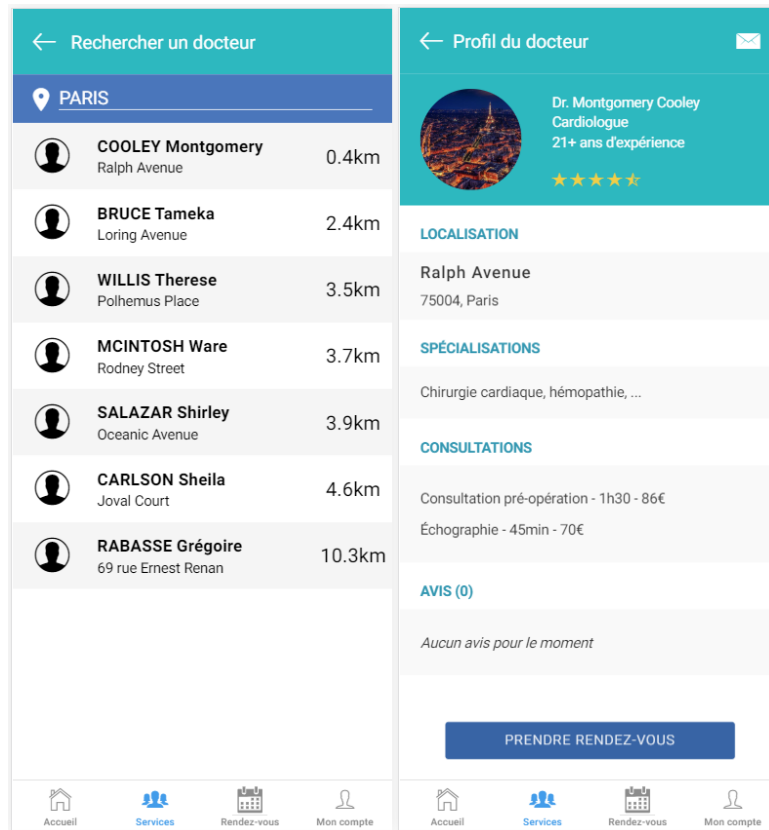
3. Docteurs

Depuis l'onglet Services, lorsque l'on clique sur « Docteurs », l'application propose de rechercher un docteur en fonction de sa spécialité. **Cette fonctionnalité est opérationnelle.**



On peut ensuite choisir la spécialité désirée, et la liste des médecins ayant cette spécialité s'affiche alors. Cette liste est ordonnée en fonction de la distance entre son cabinet et la position souhaitée. Puis, en cliquant sur le nom d'un médecin, son profil s'affiche.

La recherche par géolocalisation comporte encore quelques problèmes.



Sur le profil, la partie Consultations est à remplir par les médecins partenaires.

Le système d'avis n'est pas encore fonctionnel.

Sur le profil d'un médecin, il est possible de le contacter et de faire une demande de rendez-vous en fonction de ses disponibilités.

The image displays three mobile application screens for a medical service, each with a bottom navigation bar containing icons for 'Accueil', 'Services', 'Rendez-vous', and 'Mon compte'.

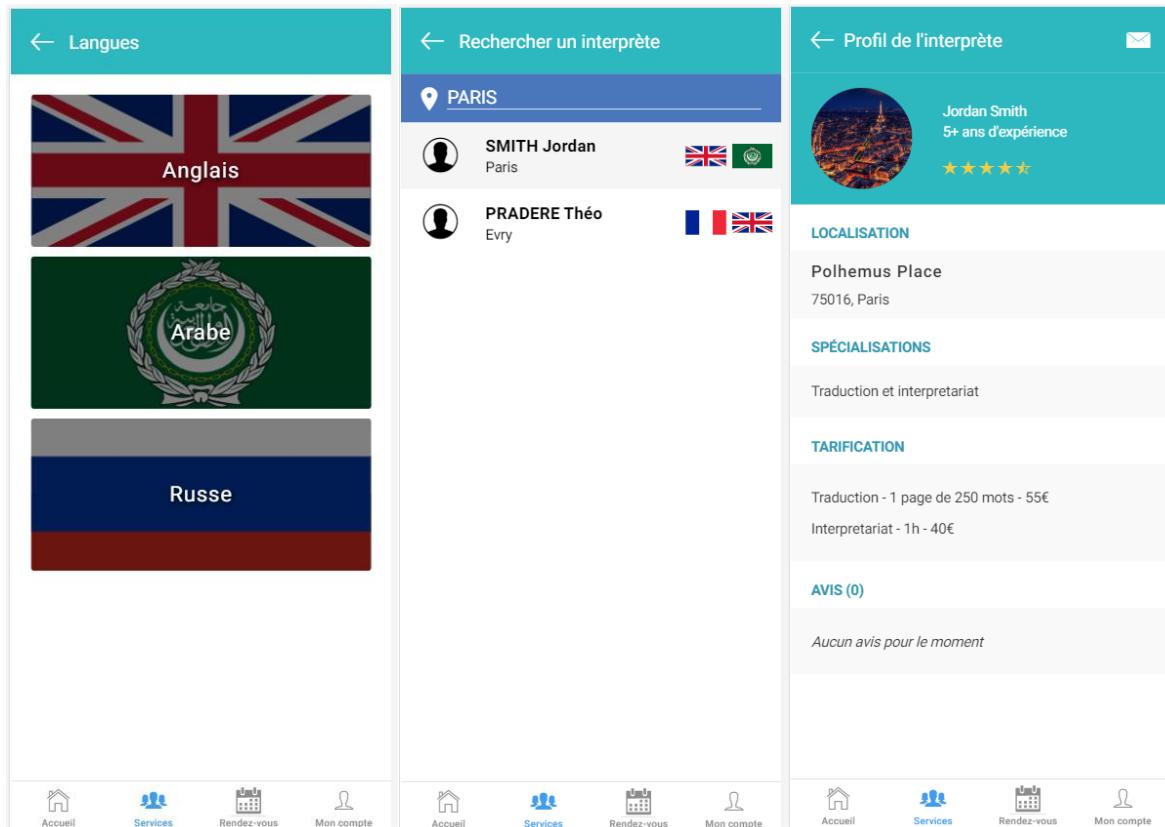
Screen 1: Contacter un docteur
This screen has a teal header. Below the header, it prompts the user to 'ÉCRIREZ VOTRE MESSAGE CI-DESSOUS :'. There are input fields for 'Nom' and 'Prénom', an 'E-mail' field, and a larger text area for 'Votre message...'. A teal 'Envoyer' button is at the bottom.

Screen 2: Prendre rendez-vous
This screen has a teal header. It shows a calendar for 'Juin 2018'. The date '18' is highlighted in red. Below the calendar, there are green buttons for appointment times: '09:00 - 10:30', '10:30 - 12:00', '12:00 - 13:30', '13:30 - 15:00', and '15:00 - 16:30'.

Screen 3: Prendre rendez-vous (Confirmation)
This screen has a blue header. It displays a confirmation dialog box with the title 'Réserver le rendez-vous :'. The dialog shows the selected date and time: 'Lundi 18 juin 2018 12:00' and 'Lundi 18 juin 2018 13:30'. There are two buttons in the dialog: a green 'Confirmer' button and a red 'Refuser' button.

4. Interprètes

De la même manière, on peut rechercher, contacter, et prendre un rendez-vous avec un interprète.



5. Rendez-vous

Cet onglet affiche tous les rendez-vous pris par un patient.

6. Compte

Dans cet onglet, on peut notamment se connecter à son compte par **adresse e-mail (fonctionnel)**. L'identification par **Facebook et Google+ n'est pas encore fonctionnelle**.

On peut aussi choisir la langue de l'application : français, anglais, russe ou arabe. L'application détecte automatiquement la langue du téléphone mais c'est via cette page que l'on peut changer la langue manuellement.

IV. Développement

1. Recherche d'un médecin et utilisation des bases de données

Lorsque l'on veut rechercher un médecin (par exemple un cardiologue), on va cliquer sur la tuile correspondante à sa spécialité.

```

10 <div id="departments">
11   <ng-container *ngFor="let d of departments">
12     <div class="department" *ngIf="d[0]===1" style="background-image: url('assets/imgs/departments/cardio.png')" (click)="goToDoctorsList(d[0])">
13       <label>{{ d[1] }}</label>
14     </div>
15     <div class="department" *ngIf="d[0]===2" style="background-image: url('assets/imgs/departments/neurologie.jpg')" (click)="goToDoctorsList(d[0])">
16       <label>{{ d[1] }}</label>
17     </div>
18     <div class="department" *ngIf="d[0]===3" style="background-image: url('assets/imgs/departments/oncologie.jpg')" (click)="goToDoctorsList(d[0])">
19       <label>{{ d[1] }}</label>
20     </div>
21     <div class="department" *ngIf="d[0]===4" style="background-image: url('assets/imgs/departments/allergologie.jpg')" (click)="goToDoctorsList(d[0])">
22       <label>{{ d[1] }}</label>
23     </div>
24     <div class="department" *ngIf="d[0]===5" style="background-image: url('assets/imgs/departments/dermato.png')" (click)="goToDoctorsList(d[0])">
25       <label>{{ d[1] }}</label>
26     </div>
27     <div class="department" *ngIf="d[0]===6" style="background-image: url('assets/imgs/departments/gastro.png')" (click)="goToDoctorsList(d[0])">
28       <label>{{ d[1] }}</label>
29     </div>
30     <div class="department" *ngIf="d[0]===7" style="background-image: url('assets/imgs/departments/geriatrie.jpg')" (click)="goToDoctorsList(d[0])">
31       <label>{{ d[1] }}</label>
32     </div>
33     <div class="department" *ngIf="d[0]===8" style="background-image: url('assets/imgs/departments/gyneco.png')" (click)="goToDoctorsList(d[0])">
34       <label>{{ d[1] }}</label>
35     </div>
36     <div class="department" *ngIf="d[0]===9" style="background-image: url('assets/imgs/departments/hematologie.jpg')" (click)="goToDoctorsList(d[0])">
37       <label>{{ d[1] }}</label>
38     </div>
39     <div class="department" *ngIf="d[0]===10" style="background-image: url('assets/imgs/departments/hepatologie.jpg')" (click)="goToDoctorsList(d[0])">
40       <label>{{ d[1] }}</label>
41     </div>
42   </ng-container>
43 </div>

```

L'évènement `goToDoctorsList(d[i])` (avec `i` correspondant à une certaine instance dans la base de données) va alors être déclenché. Cet évènement va appeler la fonction ci-dessous :

```

44 goToDoctorsList(specialityId: number) {
45   this.navCtrl.push(ListDoctorsPage, {'specialityId': specialityId});
46 }

```

Cette fonction va alors afficher la page contenant la liste de tous les médecins de cette spécialité.

2. Envoi d'un mail depuis l'application

Dans l'application, il est possible à plusieurs endroits d'envoyer un mail (que ce soit à un docteur, un interprète, l'équipe ITMP, etc.). Le code ci-dessus correspond à l'envoi de mail à l'équipe ITMP (pour les autres destinataires, les champs entre apostrophes sont différents) :

```

10 <section id="contact">
11 <form (submit)="onSubmit(nom.value, prenom.value, mail.value, message.value)">
12 <p *ngIf="sendingError" class="error">{{ sendingError }}</p>
13 <p *ngIf="sendingSuccess" class="success">{{ sendingSuccess }}</p>
14 <div>
15 <input type="text" placeholder="{{ 'HOME.FORM.LAST_NAME' | translate }}" spellcheck="false" required #nom>
16 <input type="text" placeholder="{{ 'HOME.FORM.FIRST_NAME' | translate }}" spellcheck="false" required #prenom>
17 </div>
18 <input type="email" placeholder="{{ 'HOME.FORM.EMAIL' | translate }}" spellcheck="false" required #mail>
19 <textarea placeholder="{{ 'HOME.FORM.MESSAGE' | translate }}" spellcheck="false" required #message></textarea>
20 <button type="submit">{{ 'HOME.FORM.SEND' | translate }}</button>
21 </form>
22 </section>

```

Pour envoyer un mail, on fait donc appel à la fonction `onSubmit()` ci-dessous :

```

29 onSubmit(last, first, email, message) {
30   if (last === "" || first === "" || email === "" || message === "") {
31     this.sendingError = "Veuillez renseigner tous les champs.";
32     return;
33   }
34   if (!this.sending) {
35     this.sendingSuccess = "";
36     this.sendingError = "";
37     this.sending = true;
38     this.http.post('http://localhost:8000/clients/send-contact-email', {nom: last, prenom: first, email: email, message: message}).subscribe(() => {
39       this.sendingSuccess = "Votre message a bien été envoyé";
40       this.sending = false;
41     }, err => {
42       this.sendingError = err.message;
43     });
44   }
45 }

```

Lorsque l'utilisateur a rempli les champs « Nom », « Prénom », « E-mail » et « Votre message... » (s'il ne remplit pas un de ces champs, le message « Veuillez renseigner tous les champs. » est affiché) et cliqué sur le bouton « Envoyer », le serveur est requêté (à l'adresse `http://localhost:8000/clients/send-contact-email`). La requête étant une requête `post`, cela va provoquer l'envoi de données sur le serveur et donc par la suite l'envoi d'un mail au docteur, à l'interprète ou à l'équipe ITMP (selon le formulaire de contact utilisé).

3. Prise de rendez-vous

Lorsque l'on a trouvé le docteur qui nous convient, il est possible de prendre de rendez-vous avec lui. Pour ceci, l'application utilise le code ci-dessous :

```

8   <app-calendar
9   [events]="events"
10  [locale]="locale"
11  [showFreeTime]="true"
12  [startHour]="9"
13  [endHour]="17"
14  [interval]="90"
15  [freeDays]="[0]"
16  (freeTimeClicked)="onFreeTimeClick($event)"
17 >>/app-calendar>
18
19 <div id="modal" *ngIf="isModalOpened && dates">
20   <div id="confirmation" *ngIf="!loading">
21     <h2>Réserver le rendez-vous :</h2>
22     <p>{{ m(dates[0]).tz('Europe/Paris').format('LLLL') | capitalize }}</p>
23     <hr>
24     <p>{{ m(dates[1]).tz('Europe/Paris').format('LLLL') | capitalize }}</p>
25     <button (click)="book()">Confirmer</button>
26     <button class="deny" (click)="closeModal()">Refuser</button>
27   </div>
28
29   <ion-spinner *ngIf="loading"></ion-spinner>
30 </div>

```

Le code entre dans la balise `app-calendar` sert à afficher le calendrier. La balise `div` ensuite sert à afficher les plages horaires possibles pour prendre les rendez-vous. Lorsque l'on clique sur une plage horaire pour la réserver, une pop-up s'ouvre en demandant si l'on souhaite confirmer ou non ce rendez-vous. Si l'on clique sur « Confirmer », l'application fait appel à la fonction `book()` ci-dessous :

```

59 book() {
60   if (!this.navParams.get('doctorId')) return;
61   if (!this.dates) return;
62
63   this.loading = true;
64   const dates = this.dates;
65
66   this.storage.get('access_token').then(at => {
67     this.docProvider.setAppointment('Doc', dates[0], dates[1], at.userId, this.navParams.get('doctorId'))
68       .then(appointment => {
69         alert('RDV réservé');
70         this.navCtrl.popToRoot();
71       });
72   });
73 }

```

Cette fonction enregistre alors le rendez-vous dans la base de données, et affiche alors une pop-up « RDV réservé ». Le rendez-vous sera alors affiché dans l'onglet « Rendez-vous ».

V. Conclusion

Une application mobile pour le tourisme médical est un projet intéressant, car le marché est en pleine expansion. Nous nous sommes concentrés sur les fonctionnalités clés que l'application doit fournir, à savoir la prise de rendez-vous avec les médecins et les interprètes.

Le projet a ainsi besoin d'être poursuivi, pour les logements, les transports, les infirmier(e)s, mais plus particulièrement sur le paiement via l'application.

Le thème du projet s'inscrit dans une tendance très actuelle, ce qui a facilité l'expression et la compréhension des attentes liées à l'application.

Enfin, ce projet a permis une montée en compétences de l'équipe, tant sur le plan du développement web/mobile que sur la conduite d'un projet.