

RENDU

TP1 : Les Bases d'Android

HAI811I - Programmation Mobile



Adam DAIA
Mohammed DAFAOUI
M1 GL



Exercice 3 : Une première application- Interface simple

Dans cette section initiale, un formulaire vierge est élaboré, comprenant les champs suivants :

- Nom
- Prénom
- Âge
- Domaine de compétences
- Téléphone

Voici des images qui montrent le processus.



Exercice 4 : Internationalisation des interfaces

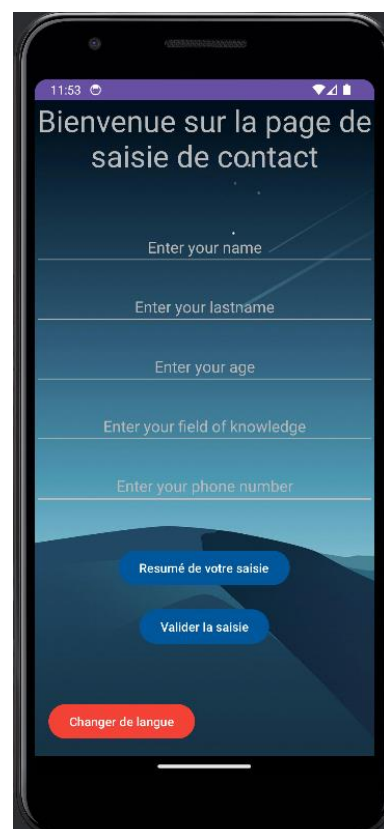
Pour internationaliser l'application, il a fallu tout d'abord écrire les chaînes de caractères dans le fichier strings.xml en français. Ensuite, un nouveau fichier de ressources a été créé en sélectionnant "Nouveau fichier de ressources", où les traductions en anglais des chaînes présentes dans le fichier strings français ont été ajoutées. En résumé :

- Écrire les chaînes dans strings.xml en français.
- Créer un nouveau fichier de ressources pour les traductions en anglais.
- Ajouter les traductions en anglais dans le nouveau fichier de ressources.

Voici des images qui montrant les deux fichiers.

<pre><?xml version="1.0" encoding="utf-8"?> <resources> <string name="nom">Enter your name</string> <string name="prenom">Enter your lastname</string> <string name="age">Enter your age</string> <string name="domaine">Enter your field of knowledge</string> <string name="telephone">Enter your phone number</string> </resources></pre>	<pre>1 <resources> 2 <string name="app_name" translatable="false">CallApplication</string> 3 <string name="nom">Entrez votre nom</string> 4 <string name="prenom">Entrez votre prénom</string> 5 <string name="age">Entrez votre âge</string> 6 <string name="domaine">Entrez votre domaine de compétence</string> 7 <string name="telephone">Entrez votre numéro de téléphone</string> 8 </resources> 9</pre>
--	--

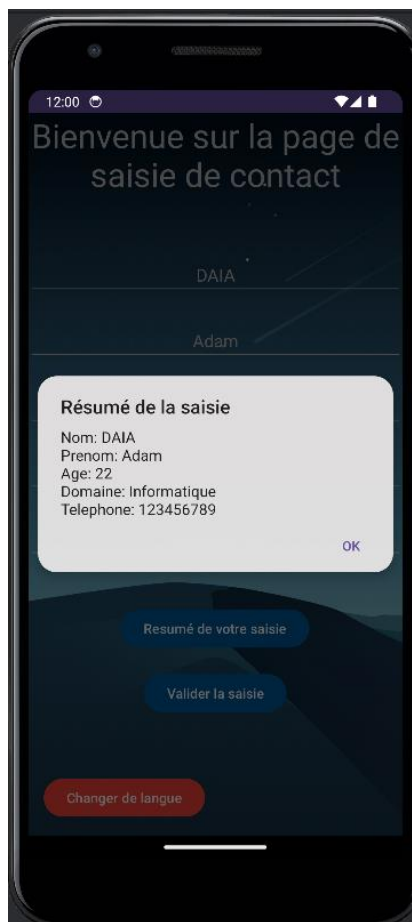
Voici les deux interfaces en français puis en anglais :



Exercice 5 : Événements associés aux objets graphiques d'une vue

Dans cette section, nous avons mis à jour l'activité en utilisant les fonctions suivantes :

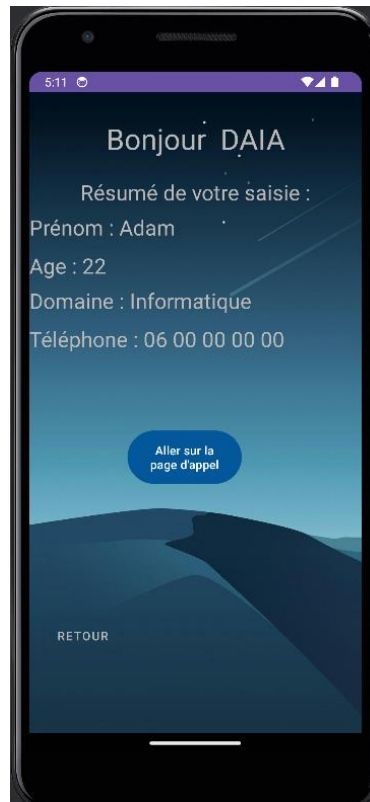
- OnCreate : Pour initialiser l'activité et configurer ses éléments.
- OnClick : Pour définir le comportement du bouton lorsqu'il est cliqué.
- Une méthode showDialog : Pour instancier la classe Dialog et afficher une fenêtre de dialogue invitant l'utilisateur à confirmer ou à annuler la validation.



Exercice 6 : Intent explicite

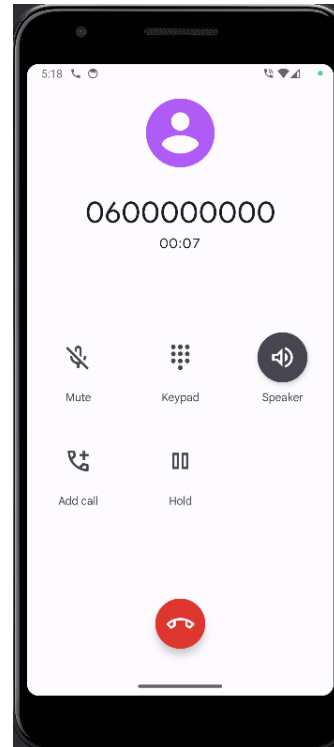
- Un Intent est créé pour récupérer les données saisies dans le formulaire de la première activité, puis pour lancer une deuxième activité.
- La deuxième activité affiche les informations collectées par l'Intent. Elle comprend également deux boutons : "Aller sur la page d'appel", qui ouvrira une troisième activité vide, et "Retour", qui permet de revenir en arrière vers la première activité.

La deuxième activité est structurée comme suit



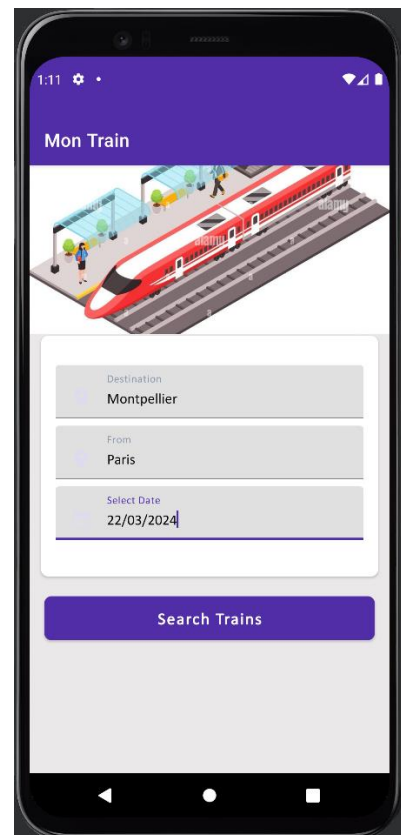
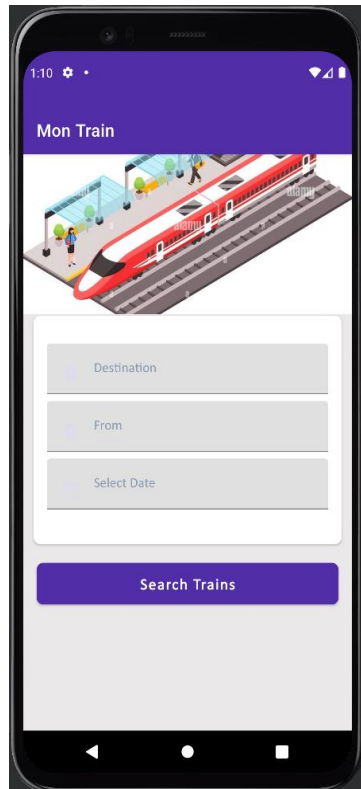
Exercice 7 : Intent implicite

Dans cette partie de l'exercice, la troisième activité est créée avec une image ainsi que le numéro de téléphone saisi dans la première activité, accompagnés d'un bouton "Appeler" qui enverra un appel vers ce numéro.



Exercice 8 : Application simple pour consulter les horaires de trains

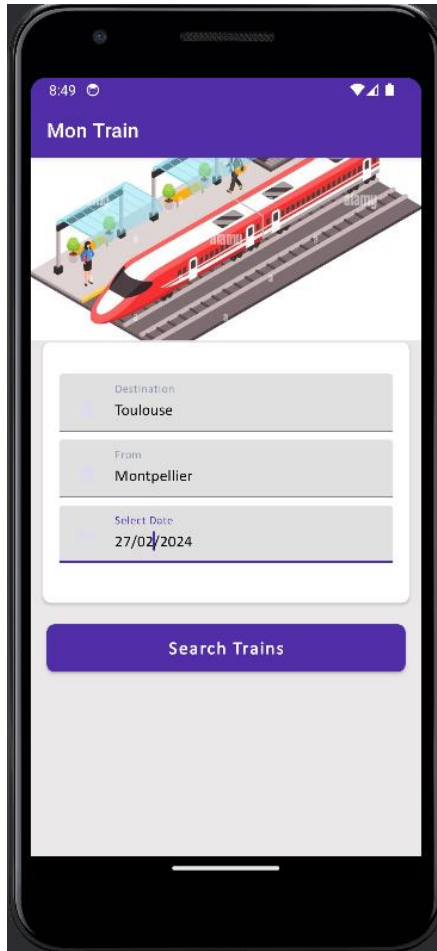
Une interface intuitive est développée pour que les utilisateurs puissent consulter les horaires de trains pour un itinéraire spécifique, en tenant compte de la ville de départ, de la ville d'arrivée et de la date sélectionnée.



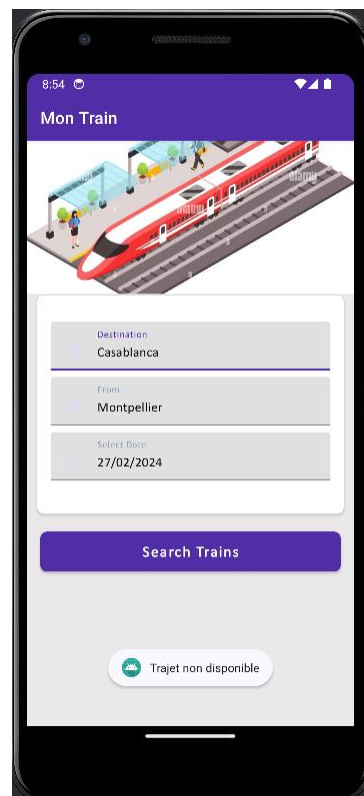
Résultat



Un exemple d'un 2ème trajet



Un toast s'affiche quand le trajet saisi n'existe pas



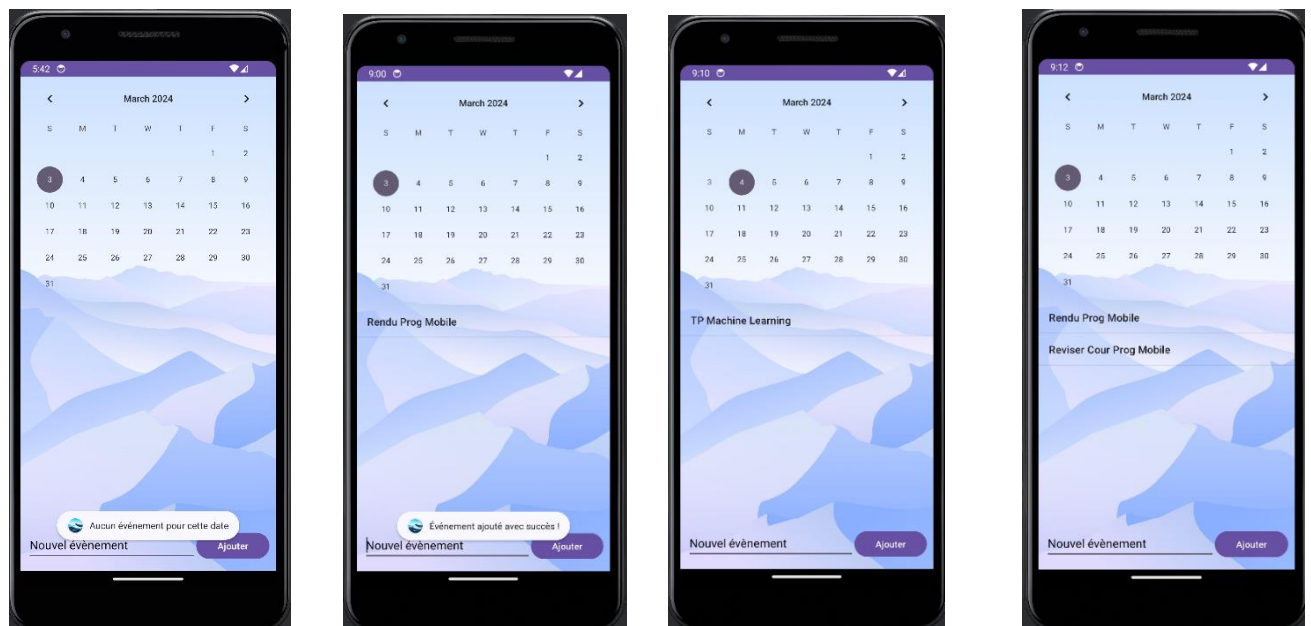
Exercice 9 : Application simple d'agenda

Nous avons intégré un système d'agenda permettant aux utilisateurs de consulter et de gérer les événements planifiés pour une date spécifique. L'interface de l'agenda comprend un calendrier interactif et une liste dynamique des événements liés à la date sélectionnée.

Les utilisateurs peuvent ajouter de nouveaux événements en saisissant un texte dans un champ dédié et en appuyant sur un bouton. Une fois ajoutés, les événements sont enregistrés et associés à la date correspondante dans l'agenda.

Cette fonctionnalité repose sur l'utilisation d'un `CalendarView` pour la sélection des dates et d'une `ListView` pour l'affichage des événements. Les événements sont stockés dans une structure de données de type `Map`, utilisant la date en millisecondes comme clé. Ainsi, chaque événement est attribué à sa date respective dans l'agenda.

En outre, lorsqu'une date est sélectionnée dans le calendrier, les événements associés sont automatiquement actualisés pour offrir une expérience utilisateur fluide et intuitive.



Aucun évènement
ajouté

1 évènement ajouté au
3/03

1 évènement ajouté au
4/03

2 -ème évènement
ajouté au 3/03