



## TourGuide — Aperçu de l'application

**TourGuide** est une application Spring Boot et une pièce maîtresse du portfolio d'applications TripMaster. Elle permet aux utilisateurs de voir quelles sont les attractions touristiques à proximité et d'obtenir des réductions sur les séjours d'hôtel ainsi que sur les billets de différents spectacles. TourGuide a connu une croissance stable dans le passé, avec plusieurs centaines d'utilisateurs par jour. Ainsi, l'équipe principale de développement a été chargée d'autres projets il y a environ six mois.

Toutefois, grâce à un article de fond publié dans un magazine de voyages de premier plan, 30 000 nouveaux utilisateurs se sont inscrits au cours des deux derniers mois. TourGuide ploie aujourd'hui sous le poids de sa nouvelle base d'utilisateurs. Cette croissance devrait s'accélérer encore plus dans un avenir proche, et nous prévoyons désormais au moins 100 000 utilisateurs par jour !

Dans l'état actuel des choses, TourGuide est tout simplement trop lent pour servir autant de clients. Les utilisateurs l'ont remarqué et se plaignent des mauvaises performances. Auparavant, l'application TourGuide mettait à jour fréquemment leur position géographique, mais elle est maintenant trop lente. Quand ils sont trop sollicités, les composants suivants sont particulièrement lents :

- gpsUtil - ce service collecte l'emplacement du téléphone mobile ou de l'ordinateur portable de l'utilisateur. En cas d'utilisation intensive, gpsUtil connaît des temps de réponse lents ;
- RewardsCentral - ce service fait appel à un réseau de fournisseurs pour regrouper un ensemble de valeurs et déterminer les récompenses offertes pour chaque attraction touristique. La collecte de ces offres prend un temps indéterminé, car TourGuide n'a aucun contrôle sur le taux de réponse des partenaires du réseau. Lors des pics d'utilisation, certains utilisateurs se sont plaint d'avoir reçu les récompenses le lendemain voire le surlendemain.

J'ai mis en place un fichier Excel ([Graphiques et métriques des performances de TourGuide](#)) qui représente graphiquement les temps de réponse actuels en cas de stress. J'ai ajouté un espace réservé pour indiquer les nouveaux temps de réponse une fois que tout le travail sera terminé, et c'est là que tu intervies.

En outre, quelques nouveaux bugs ont été signalés, ce qui a amené certains utilisateurs à abandonner l'application. Pour lutter contre cela, nous avons fait des tests en interne. Voici l'ensemble des défauts signalés, ainsi que les améliorations de fonctionnalités demandées par la direction :

- les testeurs indiquent que certains tests unitaires échouent par intermittence sans qu'on sache pourquoi ;
- les utilisateurs se sont plaints du fait que leurs offres de voyage ne correspondaient pas exactement à leurs préférences, par exemple au niveau du nombre d'enfants ou de la durée du séjour ;
- les utilisateurs ont signalé ne pas recevoir de recommandations d'attractions touristiques. Il faut que les utilisateurs reçoivent des recommandations d'attractions pertinentes, quelle que soit leur distance par rapport à l'emplacement actuel de l'utilisateur ;
- les responsables du produit souhaitent pouvoir observer les déplacements des utilisateurs et ainsi identifier si un schéma logique ou répétitif s'en dégage au fil du temps. Pour ce faire, ils souhaitent ajouter une nouvelle fonctionnalité à Tour Guide : regrouper tous les emplacements de tous les utilisateurs afin de les visualiser.

Les corrections de bugs et les nouvelles fonctionnalités devront être évaluées également avec une suite de tests. L'un de nos ingénieurs Assurance qualité senior a créé une suite de tests et l'a ajoutée à TourGuide. Elle simule les activités de 100 000 utilisateurs et a révélé les problèmes de performance décrits ci-dessus. Si tu as des questions sur nos exigences en matière de programmation et sur nos normes, reporte-toi aux [Consignes de programmation Java de TripMaster](#).

De plus, pour ne pas de nouveau se retrouver dans une situation de ce genre à l'avenir, nous souhaitons améliorer le process qualité pour ce projet. Il a donc été décidé de mettre en place une chaîne de build qui permettra d'exécuter les tests et d'obtenir l'artefact valide du projet. Il faudra que tu la mettes en place avec gitlab-ci.

Toute modification apportée à TourGuide doit être documentée pour le reste de l'équipe. Tu peux utiliser ce [modèle de documentation](#). Le diagramme d'architecture actuel se trouve ici : [Présentation de l'architecture de TourGuide](#).