



“ TelecomNancy Hiking est une application qui permet de préparer et de partager ses plus belles promenades et ses parcours de randonnées. ”

## Sujet

L'objectif de ce projet est de réaliser une application permettant de gérer des parcours de promenades ou de randonnées. Il existe déjà des sites web ou des applications mobiles de ce genre. Si vous n'êtes pas familier avec ce genre d'applications nous vous invitons vivement à étudier le fonctionnement des applications existantes <sup>1</sup>.

Les principales fonctionnalités attendues sont :

- la création/mise à jour d'un parcours en fournissant diverses informations telles qu'un titre, une description courte, une description détaillée, des photos, des informations pouvant constituer une fiche technique – durée, distance, dénivelés, difficulté, etc. –
- la navigation/recherche parmi les différents (fiches) parcours disponibles. La recherche pourra se faire selon différents critères – mot clé, données de la fiche technique, etc. –
- la visualisation d'une fiche parcours. Les différentes informations de la fiche doivent être présentées à l'utilisateur de manière attrayante.
- la création d'un parcours géo-localisé – chemin de coordonnées géographiques –
- l'importation d'une trace GPS au format standard GPX pour servir de parcours géo-localisé
- la visualisation du parcours effectif sur une carte
- l'identification des différentes étapes/points d'intérêts représentés sur la carte – en lien avec des photos éventuellement –
- la visualisation d'un diagramme permettant d'estimer le dénivelé au fur et à mesure du parcours
- l'importation/l'exportation d'un ou de plusieurs parcours dans un format de fichier adapté
- la gestion d'une liste de parcours favoris

Il est possible d'envisager un certain nombre d'extensions aux fonctionnalités basiques présentées ci-dessus.

- le partage des parcours – de leur description, images, etc. – à travers un serveur. Les fonctionnalités d'importation/exportation ouvrent la voie au partage des parcours. Il est possible d'offrir un service de partage en important/exportant vers un serveur de stockage – vers un dépôt Git, sur un drive Google, sur DropBox, sur un serveur web, etc. –.
- la génération de parcours à imprimer. Il serait pertinent de proposer une fonctionnalité d'exportation du parcours sous la forme d'un document imprimable (HTML/PDF) que l'on pourrait emporter lors de son parcours.

---

1. Le site web VisoRando <https://www.visorando.com/> – ou son application mobile VisoRando GPS randonnée <https://play.google.com/store/apps/details?id=org.visorando.android> – illustre parfaitement les fonctionnalités que l'on peut attendre d'une telle application.

# Organisation

## Constitution des groupes

Les groupes sont constitués de 4 élèves qui ne sont pas issus obligatoirement du même groupe de TD. Il vous est donc demandé pour le **vendredi 4 décembre 2020** au plus tard de fournir les informations concernant votre groupe en remplissant le formulaire dédié à l'adresse <https://forms.gle/tX5oeB5TrMKFM6M89>.

Il vous sera demandé de fournir :

- le nom de votre groupe,
- sa composition (pour chaque membre du groupe : son nom, prénom, son adresse email (terminant par **@telecomnancy.eu**).

Ces informations nous permettront de créer un projet sur le serveur GitLab de l'école dédié à votre groupe.

## Présence et assiduité

La semaine est entièrement dédiée au module pour vous permettre de vous concentrer sur ce projet.

Cette année, cette semaine bloquée se déroule entièrement de manière virtuelle. Vous êtes donc libres de gérer votre planning vous-même en coordination avec les autres membres de votre groupe.

## Encadrement

Chaque groupe se verra assigner un enseignant référent. Celle-ci ou celui-ci veillera à la bonne avancée de votre projet au cours de la semaine. Nous vous invitons à prendre contact avec lui pour définir de quelle manière il souhaitera faire le point avec votre groupe.

Au cours des journées, plusieurs enseignants seront disponibles pour répondre à vos questions. Ceux-ci seront joignables à travers la plateforme Microsoft Teams. Vous trouverez un canal CodingWeek dans l'équipe "MOCI - PCD - FISE 2020".

## Travail et collaboration

Suite à l'enregistrement de votre groupe, un dépôt privé Git vous sera créé où vous déposerez le code source et les différentes informations nécessaires à la compilation et à l'exécution de votre application (Java).

Il est nécessaire de bien organiser le contenu de ce dépôt et de "committer" régulièrement afin que l'ensemble des membres du groupe aient accès aux données les plus récentes. Il sera tenu compte des contributions de chacun des membres du groupe. Si un des membres n'a pas su "committer" ses contributions, sa participation sera considérée comme nulle entraînant l'invalidation du module.

Vous devrez :

- documenter la planification et l'avancement du projet ;
- vous assurer des livraisons (releases) quotidiennes et de la bonne utilisation de Git ;
- mettre en œuvre des tests et des procédures assurant la qualité du logiciel ;
- documenter votre architecture et votre conception pour les rendre compréhensibles ;
- documenter l'installation et l'exécution de votre application.

Ces différents points seront pris en compte pour un tiers de la note.

## Travail préparatoire

Vous disposez de quelques jours pour vous mettre à niveau sur les technologies que vous devrez utiliser :

- renforcement de vos connaissances sur l'utilisation de JavaFX et le fonctionnement interne de Git
- mise en place de l'environnement de travail (outils de développement, gestion de versions et de bases de données, etc.) ;
- étude des technologies liées qui pourraient être utilisées.



Quand vous utilisez Git faites attention le cas échéant à ne pas publier vos clés d'accès aux APIs que vous utilisez <sup>a</sup>. Pour cela, il convient de placer ces clés dans un fichier de configuration et de filtrer ce fichier en utilisant le fichier `.gitignore`. Vous devez vous arranger pour que chacun utilise ses propres clés d'accès aux API. Si possible, prévoyez des clés d'accès pour vos démonstrations, sinon documentez la procédure pour obtenir et fournir ces clés.

---

a. Ars Technica. *PSA : Don't upload your important passwords to GitHub*. Janvier 2013

## Évaluations et rendus

Le développement se fait de **façon itérative et incrémentale**. Vous devez produire un **logiciel opérationnel chaque jour**. Nous vous recommandons d'utiliser un outil tel que Gradle pour automatiser la compilation, l'exécution des tests et la construction de votre application. Vous y ajoutez des fonctionnalités en fonction d'une feuille de route (roadmap) que vous aurez définie à l'avance et que vous ferez évoluer en fonction de votre avancée. Il n'est pas question d'avoir plusieurs morceaux de programmes fonctionnant indépendamment les uns des autres. Le programme doit être intégré chaque jour et vous devez à tout moment pouvoir faire une démonstration de la dernière livraison. Le document **README** du dépôt doit indiquer comment exécuter ce programme.

### Rendus intermédiaires

Chaque groupe devra effectuer un rendu journalier sous la forme de commits Git (cela ne signifie pas qu'il faut se limiter à un commit par jour dans le dépôt, au contraire). Ce rendu devra être étiqueté (au moyen de la commande `git tag` dans la branche master) selon la convention de nommage suivante : **RELEASE\_DAY\_X** où X indique le jour de la semaine (lundi : 1, mardi : 2, etc.). Il s'agira d'une livraison, donc d'un système opérationnel que l'on peut déployer. Par défaut, la commande `git push` ne partage pas les tags, il faut explicitement partager ces tags (en utilisant l'option `--tags`).



La contribution de chacun des participants d'un groupe pourra être mesurée à son activité sur GitHub, i.e. le nombre de commit qu'il aura pu faire et leur taille. Il ne sera pas admis que certains d'entre vous n'aient pas commité de code à leur nom. Il est donc important de bien configurer votre client Git afin que vos commits vous soient attribués <sup>a</sup>.

---

a. Vérifiez les variables `user.name` et `user.email` au moyen de la commande `git-config --list`.

### Rendu final

Le rendu final sera étiqueté (toujours la branche master) selon la convention de nommage suivante : **RELEASE\_FINAL**.

Il devra comporter a minima :

- les documents d'analyse et de conception que vous aurez réalisés ;
- les documents de gestion de projet ;
- le code source de l'application client lourd (Java), les instructions textuelles (texte brut ou fichier PDF) indiquant comment compiler, configurer et exécuter cette application (précisant les dépendances externes et comment avoir accès à ces dépendances).

Les derniers commit auront lieu **avant 16h le vendredi 18 décembre 2020**.



Il ne sera pas possible de faire de démonstration le dernier jour. Vous nous rendrez une démo de 10 mn de votre application sous la forme d'un screencast/vidéo présentant ses fonctionnalités. Vous pouvez utiliser un outil de capture d'écran tel que OBS Studio <https://obsproject.com/> ou tout autre outil. Le plus simple est de déposer votre vidéo sur une plateforme en ligne telle que YouTube ou autres et de publier ensuite le lien dans votre fichier **README** de votre dépôt Git (ce n'est pas un problème si le commit publiant ce lien est réalisé après l'heure de rendu limite de votre projet).

Cette vidéo devra être rendu au plus tard **samedi 19 décembre 2020** midi.

## Critères d'évaluation

- résultats concrets présentés à la fin de la semaine ;
- respect des méthodes ;
- organisation du travail ;
- mise en œuvre de tests démontrant le fonctionnement du logiciel ;
- qualité du développement (principe de conception mis en œuvre, architecture adaptée).

## Communication avec les encadrants

Afin de poser vos questions et de discuter durant la semaine, nous avons créé sur la plateforme Microsoft Teams un canal CodingWeek. N'hésitez pas à y poser des questions sur le sujet ou sur des points techniques. Nous avons également créé un canal par groupe de projet si vous voulez vous y réunir.

<https://teams.microsoft.com/l/channel/19%3a3f035441e0334a1a8bd9dde10859f17e%40thread.tacv2/CodingWeek?groupId=b12b6c62-4947-4cde-99f9-0bae25017df8&tenantId=158716cf-46b9-48ca-8c49-c7bb67e575f3>

Afin de centraliser les différentes réponses, nous avons mis également en place un document partagé. N'hésitez pas à consulter/éditer ce document régulièrement au cours de la semaine.

[https://docs.google.com/document/d/1J8UfEwU3E4gU5-4V7JeqZY0Yn5ZPbyCSGg\\_mEE1oDzs/edit?usp=sharing](https://docs.google.com/document/d/1J8UfEwU3E4gU5-4V7JeqZY0Yn5ZPbyCSGg_mEE1oDzs/edit?usp=sharing)

## Quelques pointeurs

Vous trouverez dans cette section divers pointeurs sur des bibliothèques que vous pouvez utiliser. Vous pouvez bien entendu utiliser d'autres bibliothèques que celles présentées.

**MapJFX** (<https://github.com/sothawo/mapjfx>) fournit un composant graphique JavaFX permettant de visualiser et de naviguer sur une carte. Il repose sur la technologie OpenLayers et vous permet donc de choisir parmi plusieurs sources de cartes (OpenStreetMap, Bing, etc.). En alternatives, il existe également le projet Gluon Maps (<https://gluonhq.com/labs/maps/>) ou encore le composant LeafletMap (<https://github.com/ssaring/sportstracker/tree/master/leafletmap>).

**Java JPX Library** (<https://github.com/jenetics/jpx>) est une bibliothèque Java permettant de lire et de générer des traces GPS au format GPX.

**HttpClient**. Depuis Java 11, une API HttpClient est intégrée à l'API Java pour réaliser des requêtes HTTP et permettre à vos programmes de converser avec des serveurs Web. Un tutoriel d'introduction de cette API est disponible en ligne (<https://openjdk.java.net/groups/net/httpclient/intro.html>). Vous pouvez également utiliser la bibliothèque Apache HTTPComponents (<https://hc.apache.org/>).

**DerbyDB**. Si vous le souhaitez, vous pouvez utiliser une base de données SQL embarquée tels que Apache Derby (<https://db.apache.org/derby/>), H2 (<https://www.h2database.com/>) ou HSQLDB (<http://hsqldb.org/>) pour gérer la persistance des données de votre application. Elles sont facilement utilisables (<https://dzone.com/articles/3-java-embedded-databases>) et supportent toutes la spécification JDBC (Java Database Connectivity). Une introduction à cette technologie est disponible à l'adresse : <https://www.jmdoudoux.fr/java/dej/chap-jdbc.htm>.

**Gradle / JavaFX**. Le projet disponible à l'adresse (<https://gitlab.telecomnancy.univ-lorraine.fr/Gerald.Oster/boilerplate-gradle-jdk15>) peut vous servir de point de départ pour configurer un projet Gradle support Java (ici la version 15, mais facilement adaptable à la version 11) et JavaFX.