



TelecomNancy DirectDealing

“ TelecomNancy DirectDealing est une application d'économie circulaire en ligne permettant à des personnes de prêter/emprunter du matériel (tondeuse, marteau-piqueur, épluche-légumes, etc.) et/ou de proposer/demander des services (réparation d'une fuite d'eau, leçon de piano, déménagement, etc.) ”

Sujet

TelecomNancy DirectDealing est une application d'économie circulaire en ligne permettant à des personnes de prêter/emprunter du matériel (tondeuse, marteau-piqueur, épluche-légumes, etc.) et/ou de proposer/demander des services (réparation d'une fuite d'eau, leçon de piano, déménagement, etc.); les services peuvent se faire de façon ponctuelle (le mardi 20 juin à 13h10) ou récurrente sur une période donnée (tous les mardis 18h et jeudis 19h15 de septembre à décembre). Dans le cas d'un prêt de matériel, la durée du prêt peut être fixée à l'amiable entre le prêteur et l'emprunteur ou fixée dans l'offre de prêt. Les propositions ou les demandes sont géo-localisées.

Tous les utilisateurs de l'application doivent disposer d'un compte.

Un système de récompenses est mis en place pour valoriser les échanges. L'inscription est obligatoire; à l'inscription, toute personne se voit attribuer un certain nombre de florains. Chaque matériel a un coût en florain; quand l'emprunt est terminé, les florains sont transférés de l'emprunteur au prêteur. Il en va de même pour les services.

L'application gère un planning : un prêteur peut ainsi indiquer ses disponibilités. Les prêteurs peuvent mettre leur compte en sommeil pendant une période donnée (pour partir en vacances, par exemple). Il en va de même pour les services.

Un moteur de recherche permet de retrouver aisément un matériel en prêt ou un service, disponible sur une période donnée dans une zone géographique donnée.

L'application assure un moyen de communication (messagerie/chat) entre les personnes concernées par un prêt/service. En cas de litige, une personne peut déposer une réclamation, qui sera traitée par un administrateur. Lorsqu'un emprunt arrive à échéance, l'application notifie l'emprunteur.

Un système d'évaluation est mis en place pour noter les différentes transactions et par conséquent la qualité des objets et services rendus et/ou la disponibilité des personnes. L'évaluation peut être un critère de recherche.

L'application permet de gérer une liste d'attente pour un emprunt ou un service.



Pour vous simplifier la tâche, vous supposerez que les utilisateurs utilisent une seule et même instance de l'application (en basculant de compte). Vous n'avez donc pas de partie serveur à réaliser, ni de communication entre plusieurs instances de l'application. L'application contient toute la logique métier. Par exemple, envoyer une notification à un utilisateur, revient uniquement à ajouter une entrée représentant cette notification dans la file/a boîte de messages de l'utilisateur destinataire.

Organisation

Présence et assiduité

La semaine a été bloquée pour vous permettre de vous concentrer sur cette épreuve. Aucune salle de TD ou de TP n'est réservée spécifiquement, mais vous pouvez vous installer sans gêner les autres cours. Nous vous prions donc de bien vouloir libérer celle-ci lorsque des enseignements y sont prévus dans la semaine. Il vous est demandé d'indiquer à votre encadrant(e) dans quelle salle votre groupe se trouve.

Encadrement

Chaque groupe se verra assigner un enseignant référent. Chaque groupe rencontrera son enseignant à intervalles réguliers. Au cours des journées, plusieurs enseignants seront disponibles pour répondre à vos questions. Ceux-ci seront joignables à travers le forum ou dans une salle dont le numéro vous sera communiqué ultérieurement.

Travail et collaboration

Suite à l'enregistrement de votre groupe, un dépôt privé Git vous a été 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 "commiter" régulièrement afin que l'ensemble des membres du groupe ait 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 "commiter" ses contributions, nous considérerons sa participation comme nulle.

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.



Quand vous utilisez Git faites attention à ne pas publier vos clés d'accès au serveur GitLab ^a. 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 (ce qui ne devrait pas être un problème grâce aux branches et aux tags Git). 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. Il s'agira d'une livraison, donc d'un système déployable et opérationnel. 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 GitLab, 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 ^a.

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 ;
- 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 12 janvier 2024**.



Il ne sera pas possible de faire de démonstration le dernier jour. Vous nous rendrez une démonstration de 10 minutes de votre application sous la forme d'un screencast présentant ses fonctionnalités. Vous pouvez utiliser un outil de capture d'écran tel que OBS Studio <https://obsproject.com/> ou tout autre outil. L'objectif de cette vidéo est de montrer les fonctionnalités de votre application, mais également d'en faire la promotion. Généralement, ce genre de vidéo est accompagné d'un commentaire audio et déroule un ou plusieurs scénarios de démonstration. Le plus simple est de déposer votre vidéo sur une plateforme en ligne telle que YouTube ou Dailymotion 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).

Critères d'évaluation

- Les résultats concrets présentés à la fin de la semaine ;
- le respect des méthodes ;
- l'organisation du travail ;
- la mise en œuvre de tests démontrant le fonctionnement du logiciel ;
- la qualité du développement (principe de conception mis en œuvre, architecture adaptée) ;
- l'invention de nouveaux services et caractère innovant du logiciel.

Communication avec les encadrants

Afin de poser vos questions et de discuter durant la semaine, nous avons créée sur la plateforme de messagerie Microsoft Teams une équipe avec différents canaux (nous créerons un canal pour chaque équipe également). N'hésitez pas à y poser des questions sur le sujet ou sur des points techniques.



Afin de rejoindre l'équipe dédiée à la codingweek, vous devrez utiliser **ce lien d'invitation** à votre première connexion.

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/1mo3fNTjSwDBYj08VujF5QrTmD4rjWrAp-kBYTG386Y0/edit>