

INF1163 : Modélisation et conception orientée objet

Projet

Système de contrôle d'une borne de prêt en libre-service

Travail en équipe (4 membres par équipe)

Le but de ce projet est de vous permettre de mettre en pratique quelques-uns des principes de conception orientée objet vus en classe. L'implémentation se fera en **Java** avec la plateforme de développement **Eclipse**. Le projet vous donne aussi l'occasion d'expérimenter avec quelques outils logiciels performants utilisés dans un environnement de développement professionnel (Git, GitHub, SQLite).

La collaboration entre les membres de chaque équipe est très importante afin de pouvoir respecter les délais impartis. Les étudiants doivent donc apprendre à travailler en équipe. Ils doivent notamment apprendre à se répartir les tâches équitablement, à développer leur sens de la communication et à synchroniser leur travail afin d'être productifs dans leurs efforts. Il faudra nommer un chef d'équipe pour coordonner le travail et veiller à la bonne progression du projet. Le rôle de chef d'équipe pourra être joué, à tour de rôle et pour une période déterminée à l'avance, par chacun des membres de l'équipe. Afin de gérer efficacement le développement de votre code source et de pouvoir rapidement revenir à une version qui fonctionne en cas de problème, il est fortement conseillé d'utiliser l'outil de gestion de version Git (<http://git-scm.com/>). Afin de permettre à plusieurs membres de l'équipe de développer en parallèle et d'intégrer automatiquement leurs différentes portions de code, il est fortement suggéré d'utiliser l'outil de gestion de contrôle GitHub (<https://github.com/>). GitHub est en fait la partie serveur pour l'outil Git. Git est gratuit et peut être installé sur votre PC. GitHub est un serveur sur le nuage qui requiert que vous ouvriez un compte (gratuit).

L'objectif du projet consiste à implémenter le logiciel de contrôle d'une borne de prêt en libre-service de bibliothèque (*Library Self-Checkout System*). En raison des contraintes temporelles, on se limitera au développement d'une version simplifiée qui simule la logique de prêt en libre-service et qui offre des interfaces utilisateur simples. La composante base de données ne faisant pas partie des objectifs de ce cours, on se contentera d'utiliser des fichiers Excel pour stocker les données nécessaires au bon fonctionnement de la borne. Toutefois, pour les équipes qui le voudront, il est conseillé d'utiliser le système de gestion de bases de données simple et pratique suivant : SQLite (<http://www.sqlite.org/>). Il s'agit d'un SGDB léger et gratuit qui s'installe très facilement sur n'importe quel PC.

Brève description du système

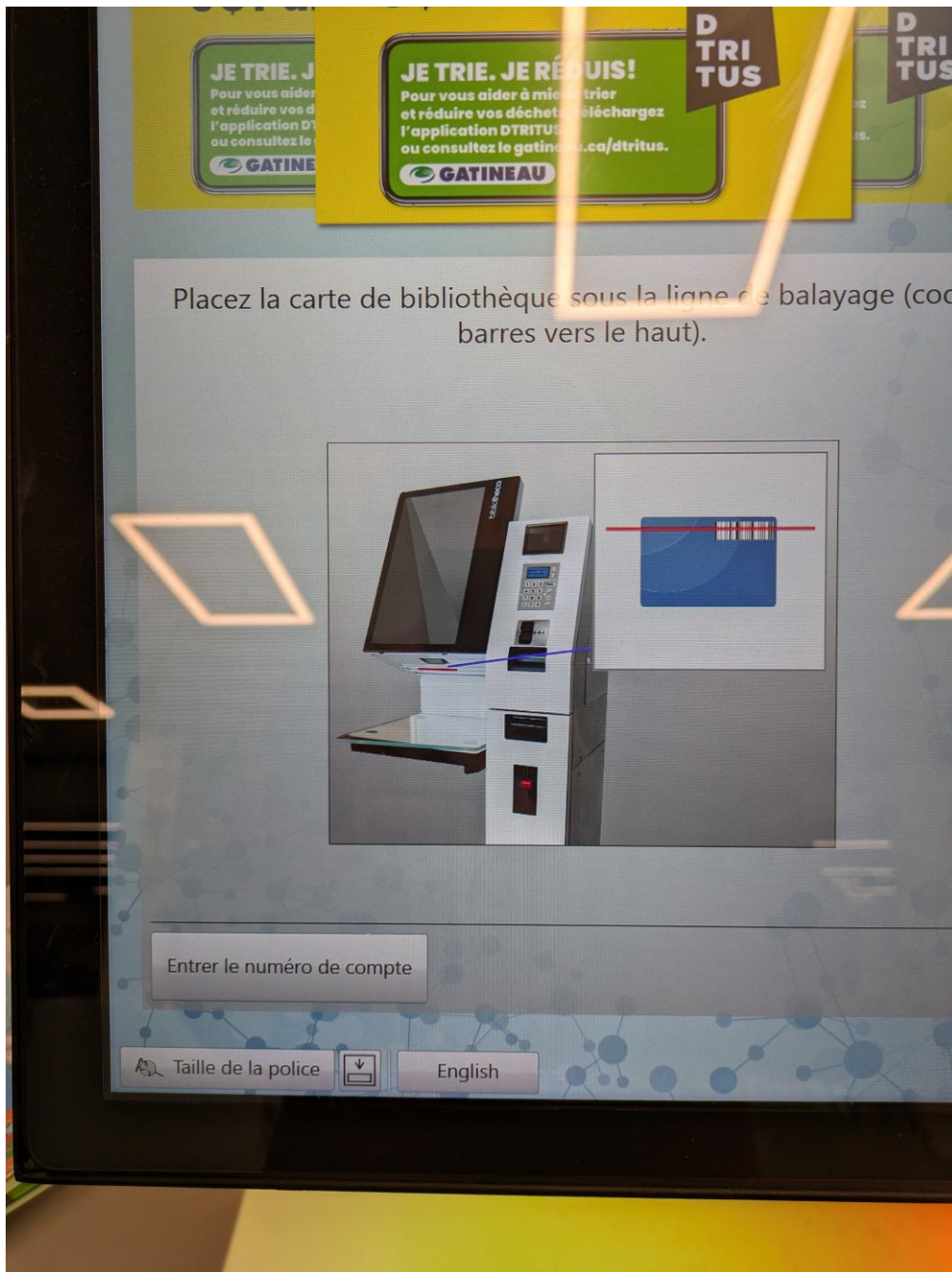
La borne permet aux usagers de la bibliothèque de saisir eux-mêmes les documents qu'ils veulent emprunter, sans avoir à passer au comptoir de prêt. Ci-après le cas d'utilisation « Emprunter document » au format abrégé :

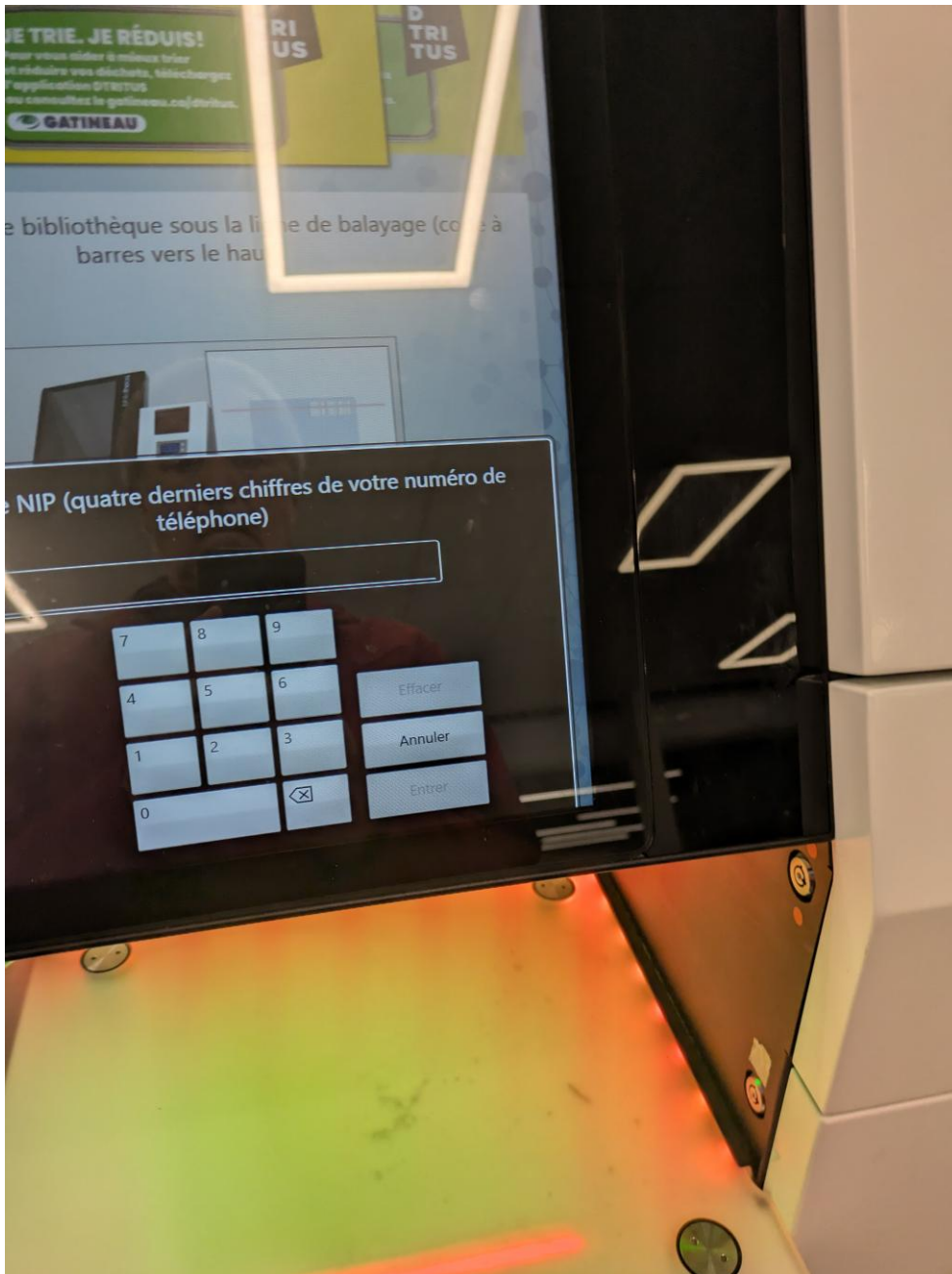
L'utilisateur se présente devant la borne avec le ou les exemplaires de livres qu'il veut emprunter. Il s'identifie à la borne en entrant son numéro de compte et son NIP de bibliothèque. Il dépose ensuite les exemplaires sur la tablette de la borne. La borne identifie chaque exemplaire grâce à la puce RFID incorporée dans l'exemplaire (*Radio Frequency Identity Tag*). Le système demande ensuite si l'utilisateur veut un reçu qui récapitule la liste des exemplaires empruntés avec leur date de retour. L'utilisateur part avec les exemplaires et, le cas échéant, le reçu.

La figure ci-dessous donne une vue d'ensemble de la borne.

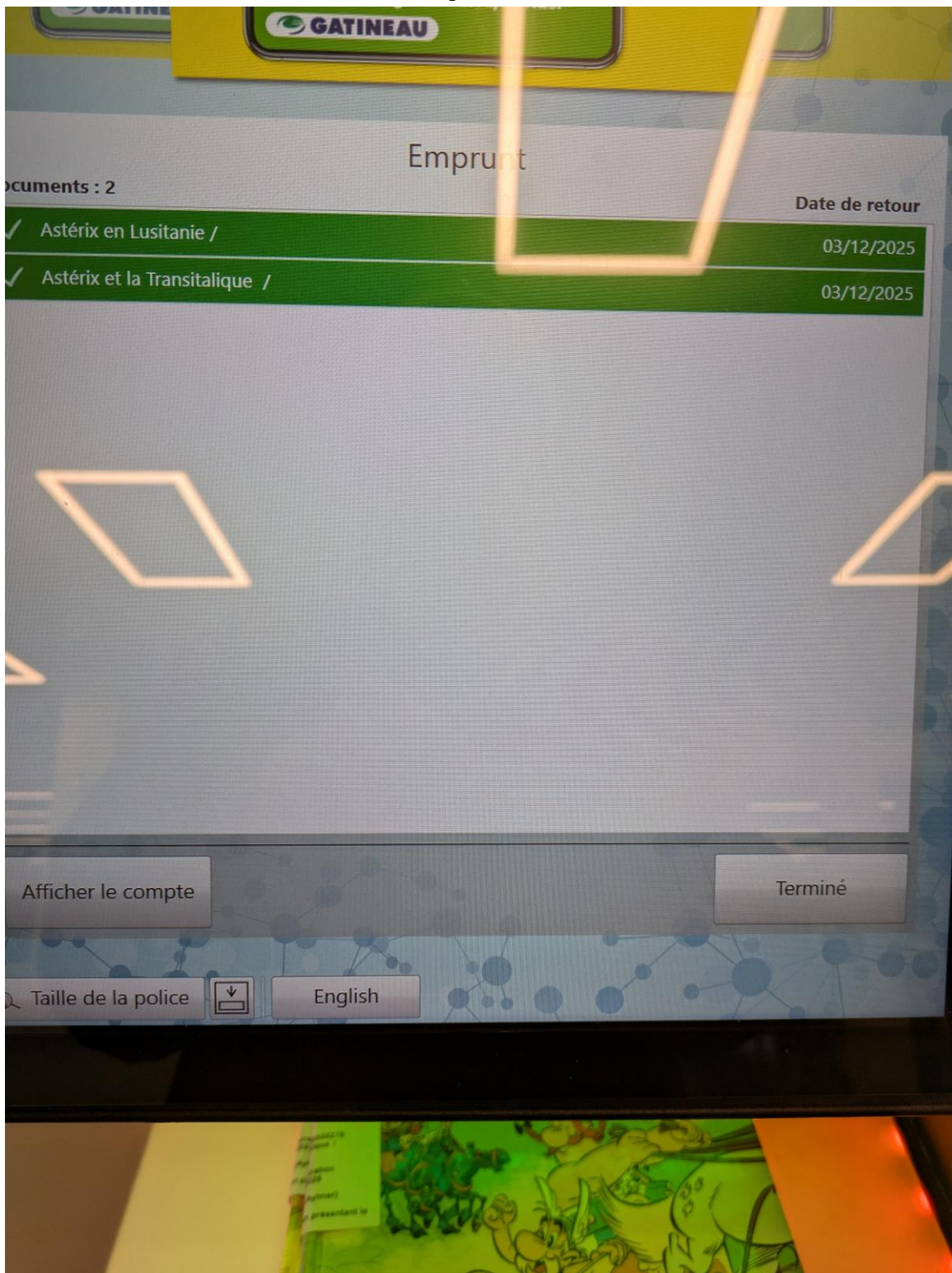


Ci-dessous quelques détails de la borne qui devraient vous aider à mieux comprendre le système. Ces images sont fournies à titre d'illustration, mais il ne s'agit pas pour vous de reproduire à l'identique les interfaces sur ces photos.

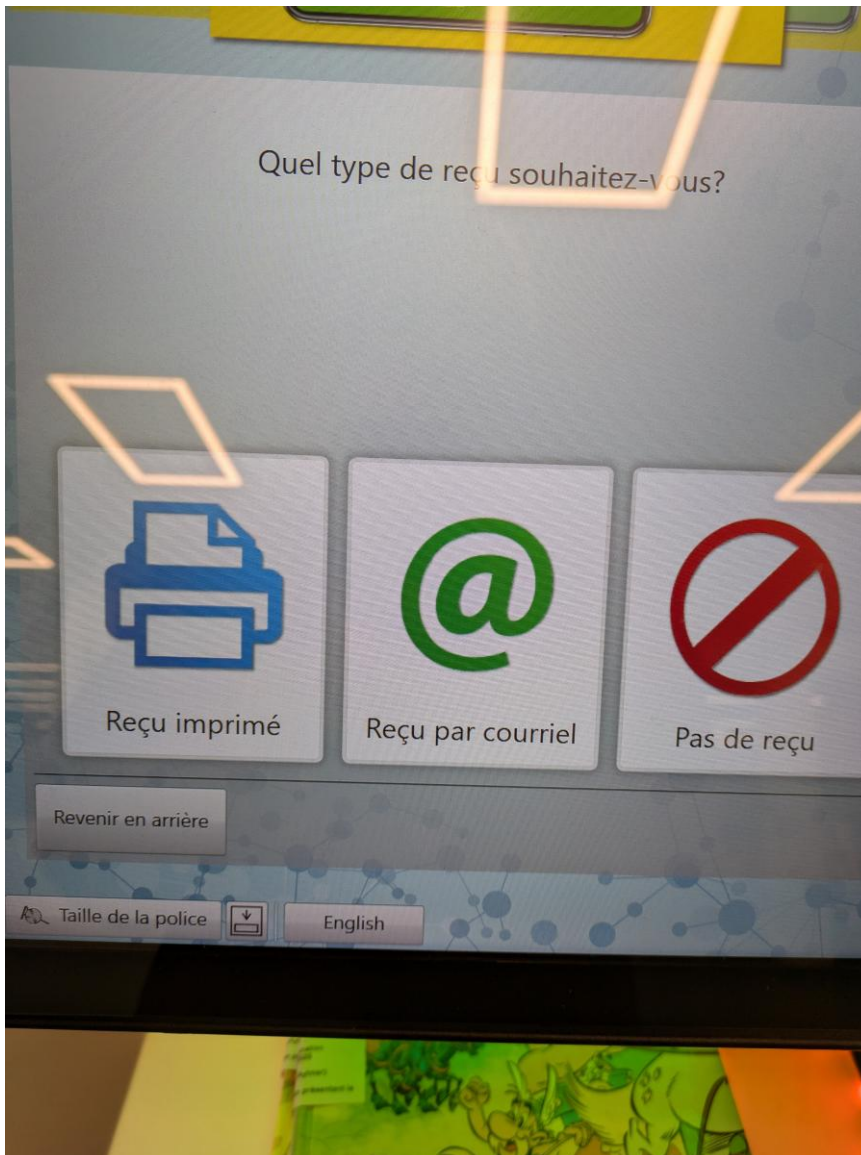




La figure qui suit illustre la situation où l'utilisateur a emprunté 2 exemplaires de livres. La borne affiche le nombre de documents empruntés, les titres et les dates de retour.



La figure ci-dessous illustre le menu affiché par la borne une fois que tous les exemplaires à emprunter ont été saisis.



On rappelle que ces figures ne sont qu'indicatives et incomplètes, et vous êtes libres de proposer des variations, adaptations et/ou améliorations par rapport à ces vues.

Travail à faire

Il s'agit de développer le logiciel qui permet de procéder aux emprunts via la borne. On simulera la lecture automatique du code RFID en tapant explicitement ce code au clavier. Par souci de simplification, on assumera que chaque code RFID consiste en un nombre unique de 6 chiffres, et que chaque NIP d'utilisateur consiste en un nombre unique de 4 chiffres. Afin de pouvoir tester le système, on créera 2 usagers, Jean et Jeanne avec, respectivement, les numéros de compte suivants : jean123 et jeanne456.

Concrètement, on vous demande d'implémenter 2 cas d'utilisation :

- Le cas d'utilisation « Emprunter documents » qu'on vous a fourni au format abrégé. Vous devez le réécrire au format détaillé, sur 2 colonnes, et dessiner le DSS correspondant qui identifie clairement les événements-systèmes pertinents.
- Un cas d'utilisation « Démarrer le système » en vous inspirant du diagramme UML figurant dans le dernier transparent (#65) des notes de cours 7 (Section 9 sur Moodle). Vous devrez charger dans le système le catalogue comprenant une dizaine de livres (voir format ci-dessous).

RFID	Titre	Auteur	Edition	Date parution	Nombre pages
123456	La peste	Albert Camus	Gallimard	1947	336
789012	Les misérables	Victor Hugo	Folio Junior	2011	291
<i>etc...</i>	<i>etc...</i>	<i>etc...</i>	<i>etc...</i>	<i>etc...</i>	<i>etc...</i>

Bien qu'il y ait de fortes similitudes avec le système *NextGen POS* vu en classe, quelques différences notables sont à mentionner. Par exemple, un livre peut avoir plusieurs exemplaires, mais chaque exemplaire est unique et donc le concept de *lignesDeLivres* n'a pas de sens ici (même si la fantaisie nous prenait de vouloir emprunter 2 exemplaires du même livre, chacun devra être saisi séparément).

Le livrable final du projet doit comprendre :

- Le cas d'utilisation « Emprunter documents » au format détaillé, le DSS correspondant en UML, et le diagramme des classes de la couche d'affaires (inutile de présenter les classes de la couche de présentation).
- Votre catalogue sous forme tabulaire comme montré ci-dessus.
- Le code source en Java, adéquatement documenté et un fichier de distribution de votre application au format jar.
- Toutes les explications nécessaires pour permettre à une tierce personne de compiler, exécuter et utiliser votre prototype. Si vous avez utilisé des bibliothèques non standard, il faut le préciser et donner leur URL et toute autre explication nécessaire à leur intégration dans votre code.
- Une annexe qui synthétise la contribution de chaque membre de l'équipe au projet.

À l'exception du code source et du fichier de distribution **jar**, tous les autres éléments du livrable final doivent être incorporés dans **un fichier unique au format Word ou PDF**. Votre texte (livrable final sauf code source et fichier jar) doit être tapé en utilisant un traitement de texte et vérifié en utilisant un correcteur orthographique.

Ce livrable final constitué (i) d'un fichier unique contenant votre texte et vos diagrammes et (ii) d'un répertoire contenant tous les fichiers de code source et le fichier jar, doit être compressé dans un fichier qui portera **obligatoirement** le nom :

INF1163_Projet_Equipe_X.zip (X étant le numéro de l'équipe) et devra être déposé via Moodle au plus tard le lundi 15 décembre avant minuit.

Si le fichier zip est trop gros pour être accepté par Moodle, vous devrez le déposer sur le nuage (OneDrive, DropBox, GoogleDrive, etc.) et m'envoyer le lien par courriel.

Toute soumission qui ne se conformera pas à ces directives sera refusée.