

|                 |                               |                   |
|-----------------|-------------------------------|-------------------|
| Michel de Broux | <b>Implémentation Android</b> | Charles Momin     |
| Simon Lardinois | LSINF1225                     | Valentin Rombouts |
| Victor Lecomte  | Groupe V                      | Harold Somers     |

## 1 Installation

Le programme se télécharge et s'installe avec Android Studio depuis notre repository github : <https://github.com/vlecomte/bartender>. Il est bien entendu possible d'accéder à la version soumise juste avant la deadline en faisant un checkout sur le commit correspondant.

## 2 Utilisation

Nous avons conçu l'application pour qu'elle soit le plus intuitive possible. La navigation principale se fait via différents onglets :

- Le menu : cet onglet constitue la carte des commandes possibles pour tout utilisateur. Un click permet d'accéder à une description plus détaillée des produits. C'est via cet onglet que les utilisateurs ajoutent et enlèvent des consommations de leurs panier.[PERM : client, serveur, administrateur]
- Le panier : indique le panier actuel ainsi ainsi que le prix total des consommations sélectionnées. C'est via cet interface qu'un utilisateur confirme sa commande. Le panier coté client leur permet de valider leurs [PERM : client, serveur, administrateur]
- L'addition : indique la somme restante à payer en fonction des consommations commandées. [PERM : client, serveur, administrateur]

Pour utiliser l'application, vous pouvez vous créer un compte (client par défaut) ou utiliser un de ces trois logins d'exemple :

- login **tiffany**, password **karateswag22** (client)
- login **jeff**, password **ilovetiffany** (serveur)
- login **joe**, password **youllneverfindit** (admin)

la connexion étant retenue entre deux exécutions de l'application.

Toutefois, une courte explication s'impose pour les données représentées par les différents types de graphes dans l'interface administrateur :

- *Waiter efficiency* : graphique camembert présentant la part des services effectués par chaque serveur sur une période.
- *Turnover evolution* : graphique à deux dimensions présentant l'évolution du chiffre d'affaires.
- *Best customers* : graphique camembert présentant la part du chiffre d'affaires apporté par chaque client.
- *Most popular drinks* : graphique camembert présentant la part des consommations que chaque boisson a apporté.

## 3 Modifications de la base de données

- Afin d'internationaliser la base de données, nous devons ajouter des informations de traduction par rapport aux champs textuels de la base de données qui sont affichés directement

dans l'application. Pour cela, nous avons créé trois nouvelles tables : `ProductDisplayName`, `IngredientDisplayName` et `UnitsDisplayName`. Le champ original est remplacé par un nom de code et la correspondance entre nom de code et nom d'affichage pour chaque langue est donnée par une entrée d'une de ces nouvelles tables.

- Nous avons ajouté une quantité non-négligeable de données au niveau des détails et des commandes afin d'avoir des graphiques plus représentatifs.
- La nouvelle version de la base de données, sous la forme des fichiers `bartender-multilingue.sql` et `bartender-multilingue.sqlite`, est donnée en annexe à ce rapport.

## 4 Modifications par rapport aux schémas UML

Il y a eu relativement peu de modifications par rapport aux schémas UML dans le code. Nous détaillerons celles-ci en actualisant nos schémas dans le rapport complémentaire à la présentation.

## 5 Documentation et propreté

Étant donné qu'il s'agit d'un prototype, la priorité n'a pas été donnée à la documentation ni, dans une certaine mesure, à la propreté et la concision du code. Toutefois, nous estimons nous être débrouillés raisonnablement bien compte tenu des délais.