

## **Système de gestion de livraison de pizzas**

**Devoir#1, à remettre au plus tard le 17 février**

**Travail en binôme**

---

### **Voici les exigences minimales du système projeté**

PIZZA UML opère un comptoir et un service de livraison de pizzas. Le propriétaire, désire améliorer son système de prise de commande, de cuisine et de livraison. Il vous demande de l'aider à décrire ses besoins en vue d'une informatisation.

Les commandes arriveraient de trois façons : par téléphone, par Internet et par des clients venant directement au comptoir. Pour les commandes téléphoniques, le système téléphonique (via l'afficheur) transmettrait le numéro du téléphone de l'appelant au système informatique. Ce dernier, s'il s'agit d'un numéro de téléphone connu dans le système, afficherait le dossier du client avec son adresse, son numéro de téléphone et ses préférences. Bien entendu, le préposé aux commandes modifierait les informations du dossier, s'il y avait des changements. Pour les nouveaux clients, le préposé aux commandes compléterait une nouvelle fiche. La commande serait prise, le coût calculé incluant les taxes. La date et l'heure de la commande seraient inscrites par le système. La commande saisie, une facture serait imprimée en cuisine, cette facture servirait de bon de travail en cuisine et serait fixée à l'emballage de la commande pour la livraison.

Pour les commandes par Internet, si le client a déjà un dossier, le client entrerait sa commande, le système effectuerait les calculs et dirigerait le tout sur l'écran du préposé aux commandes qui confirmerait la commande par téléphone avant de la faire parvenir en cuisine, le système composant le numéro de téléphone pour lui. Encore ici, le préposé aux commandes pourrait mettre à jour le dossier. Si le client n'a pas de dossier, le client en créerait un et le reste du processus serait semblable. Pour les commandes au comptoir, le préposé aux commandes les saisirait avec une mention « Client comptoir » sans besoin de créer un dossier client. Le reste du processus demeurerait similaire. Une commande saisie recevrait le statut de « en préparation ».

Lorsque la cuisine terminerait la préparation de la commande, celle-ci serait emballée, la facture fixée à l'emballage et le tout transmis au préposé aux livraisons. De plus la cuisine, par un code présent sur la facture, inscrirait dans le système que la préparation de la commande serait terminée. Le système

## Devoir #1, INF5151-40

garderait la date et l'heure de cet événement et changerait le statut de la commande de « en préparation » à « préparation terminée ».

Les commandes au comptoir seraient remises directement aux clients après paiement, le préposé aux livraisons inscrirait ce fait au système. La commande changerait de statut de « préparation terminée » à « commande complétée » avec bien entendu la date et l'heure de l'événement. Les commandes devant être livrées seraient confiées à un livreur par le préposé aux livraisons, celui-ci indiquerait dans le système le numéro du livreur, la date et l'heure du départ du livreur. Le statut de la commande passerait de « préparation terminée » à « en livraison ». Lorsque le livreur aurait livré une commande, par son téléphone mobile, il avertirait le système et celui-ci noterait la date et l'heure et le statut de la commande passerait de « en livraison » à « commande complétée ».

### Questions

- 1) Déterminer les cas d'utilisation du système que l'on veut développer. Pour chacun, proposer une description textuelle (forme vue en cours) (nom, acteur(s), description, préconditions(s) et postconditions(s)).
- 2) Proposer un diagramme de cas d'utilisation de haut niveau, faisant apparaître clairement, acteurs et Use-Cases (concrets).
- 3) Proposer un diagramme de cas d'utilisation plus détaillé, faisant apparaître des relations USE et EXTEND.
- 4) À la vue des Use-Cases proposés, quels sont les concepts importants à retenir pour un futur modèle de classes. Proposer une version préliminaire d'un tel modèle.

---

Tous les diagrammes doivent être conçus à l'aide d'un outil UML