

## Exercice 1 : Distributeur automatique de produits

Le client d'un distributeur automatique de produits courants peut y trouver des produits alimentaires (pains, conserves, boissons, etc.). Une fois qu'il ait choisi les produits qu'il désirait acheter, il paye ses achats. Il existe deux façons de payer les produits : soit en espèces, soit par carte de crédit. Lors de l'achat d'un produit alimentaire, et uniquement dans ce cas, le client vérifie sa date limite de consommation.

Modéliser cette application à l'aide d'un **diagramme des cas d'utilisations**.

## Exercice 2 : Robot Ménager

On souhaite modéliser avec UML le fonctionnement d'une application *RobotMénager* commandant un robot ménager. On s'intéresse à l'**état du contrôleur de moteur**.

Une commande séparée détermine quand le moteur doit être en marche et envoie en permanence le message « *en marche* » (on) au contrôleur du moteur lorsque celui-ci doit tourner.

Quand le message « *en marche* » est émis, la commande du moteur doit démarrer celui-ci et le faire tourner. Pour cela, un courant est appliqué à la fois aux bobines *démarrer* et *tourner*. Un capteur nommé *relais de démarrage* détermine le moment où le moteur a démarré.

À ce moment-là, la bobine « *démarrer* » est coupée et seule la bobine « *tourner* » reste en tension. Les deux bobines sont coupées lorsque « *en marche* » n'est pas émis.

Ce type de moteur peut être endommagé par une surchauffe s'il y a une surcharge ou un échec au démarrage. Pour le protéger des dommages dus à la chaleur, on inclut souvent un capteur de température. Si le moteur est trop chaud, le système de contrôle coupe le courant aux 2 bobines, et ignore tout message « *en marche* » jusqu'à ce que l'on presse un bouton de réinitialisation et que le moteur ait refroidi.

Dessiner le **diagramme d'états-transitions**.

## Exercice 3 : Hôpital

Parmi les tâches critiques de l'hôpital : transfusions sanguines et les opérations de transplantation d'organes qui font l'objet de processus hautement sécurisés. Lors d'une demande de transfusion sanguine, l'infirmière chargée de cette tâche vérifie l'ordre du médecin et l'approbation du patient. Si ces deux éléments sont valides elle enregistre l'ordre de transfusion et informe la banque du sang de la demande en y précisant les éléments nécessaires (date de transfusion prévue, lieu de la transfusion, groupe sanguin etc.), elle imprime aussi un document de traitement avant transfusion selon les indications du prescripteur (un médecin), ce traitement est adressé au service où le patient sera hospitalisé. Dans le cas où la prescription médicale est incomplète et/ou l'approbation du patient est absente, l'ordre de transfusion est mis en attente et le service qui a envoyé l'ordre en est informé.

Lorsque le sang est livré, le bon de livraison est enregistré et un ordre de transfusion exécutif est envoyé au service concerné. A la fin de la transfusion on enregistre les informations sur l'exécution de la transfusion (conditions d'exécution de l'acte, éventuels changements de date ou de personnel etc.). Dans le cas où l'on n'a pas reçu le document

d'exécution dans les 24h suivant la date prévue de l'acte, une réclamation est envoyée au service concerné et le service de suivi de qualité est alerté de l'incident. Cette procédure vise à sécuriser les actes de transfusion sanguine en garantissant leur traçabilité.

On souhaite développer un logiciel pour **gérer les transfusions sanguines et les opérations de transplantation d'organes** dans un Hôpital.

- 1/ A l'aide d'un diagramme des cas d'utilisations, représenter les fonctions qui devraient figurer dans le système.
- 2/ Choisissez 2 cas d'utilisations et décrivez-les avec des diagrammes de séquences.
- 3/ A l'aide d'un diagramme de classes, représentez la structure interne des données manipulées par le système.