

TD6 - Diagramme d'activités, scénarios et diag. de séquences

Exercice 1 - Rédiger un bon scénario

Voici 5 variantes de textes décrivant un scénario d'interaction avec un système (tiré du livre d'Alistair COCKBURN):

Version 1 :

1. Le client saisit son numéro de commande. Le système détecte que ce numéro correspond au numéro gagnant, inscrit l'utilisateur et le numéro de commande comme gagnant du mois, envoie un courrier électronique au responsable des ventes, félicite le client et lui donne les instructions nécessaires pour retirer son prix.

Version 2 :

1. Le client saisit son numéro de commande.
2. Le système détecte que ce numéro correspond au numéro gagnant, inscrit l'utilisateur et le numéro de commande comme gagnant du mois, envoie un courrier électronique au responsable des ventes, félicite le client et lui donne les instructions nécessaires pour retirer son prix.

Version 3 :

1. Le client saisit son numéro de commande.
2. Le système détecte que ce numéro correspond au numéro gagnant.
3. Le système inscrit l'utilisateur et le numéro de commande comme gagnant du mois, envoie un courrier électronique au responsable des ventes, félicite le client et lui donne les instructions nécessaires pour retirer son prix.

Version 4 :

1. Le client saisit son numéro de commande.
2. Le système détecte que ce numéro correspond au numéro gagnant.
3. Le système inscrit l'utilisateur et le numéro de commande comme gagnant du mois et envoie un courrier électronique au responsable des ventes.
4. Le système félicite le client et lui donne les instructions nécessaires pour retirer son prix.

Version 5 :

1. Le client saisit son numéro de commande.
2. Le système détecte que ce numéro correspond au numéro gagnant.
3. Le système inscrit l'utilisateur et le numéro de commande comme gagnant du mois.
4. Le système envoie un courrier électronique au responsable des ventes.
5. Le système félicite le client et lui donne les instructions nécessaires pour retirer son prix.

→ Examinez ces différentes versions et les comparer : lesquelles sont les plus adaptées à un bon développement logiciel ? Pourquoi ?

Exercice 2 – Remboursement de dépenses

Voici un fonctionnement du cas d'utilisation « Effectuer le remboursement » d'un système de gestion des frais des salariés :

- Le comptable est informé (notification par email, signal visuel) quand de nouveaux rapports de dépenses arrivent (rapports numériques). Un compte de frais est créé si l'employé n'en possède pas encore ;
- Le rapport est examiné pour approbation, notamment les factures correspondant aux frais doivent être attachées au rapport (scannées). Seuls les frais avec facture sont évalués. La demande est annulée s'il n'y a aucune facture, et un courriel est envoyé au salarié. Les montants inférieurs à 100 € sont automatiquement approuvés, alors que ceux de plus de 100 € doivent être avisés par le supérieur du salarié. En cas de refus, le salarié est avisé par email avec la raison invoquée par le supérieur. Sinon, l'employé reçoit un virement directement sur son compte ce qui clôt la demande.
- Si la demande n'est pas traitée sous 7 jours et dépasse 100€, un mail de rappel est transmis au supérieur. Après 30 jours la demande est annulée : on envoie un courriel d'annulation au salarié lui notifiant qu'il doit redéposer sa demande de remboursement.

1- Écrire le **scénario nominal** (on imagine que dans la majorité des cas, les montants sont > 100€, les factures sont jointes, et le supérieur répond positivement après un rappel).

2- Imaginer tous les **scénarios d'exceptions** possibles.

3- Modéliser le système décrit à l'aide d'un **diagramme d'activité**.

4- Quelles sont les classes impliquées dans ces scénarios ? Construire le **diagramme de séquences d'interaction** synthétisant ces scénarios.

Exercice 3 – Jeu AkinatorAnimal

Le but du jeu est de réaliser un jeu *AkinatorAnimal* permettant de faire deviner au logiciel, un animal pensé par l'utilisateur.

L'utilisateur choisit un animal, et l'application pose des questions auxquelles l'utilisateur peut répondre par une des 5 réponses suivantes : oui, probablement oui, ne sais pas, probablement non, et non. En répondant aux questions, vous permettez au logiciel d'éliminer des ensembles de réponses possibles, mais vous contribuez également à définir votre animal pour les parties suivantes.



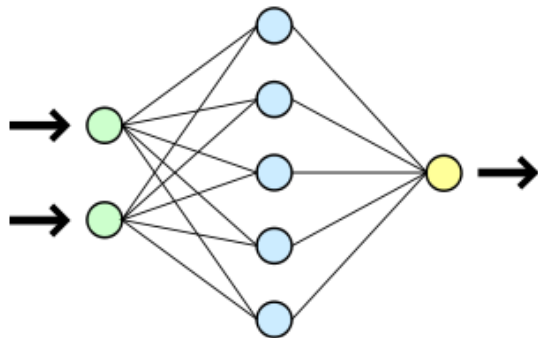
Au bout d'une série de 10 questions, l'ordinateur propose sa réponse. L'utilisateur confirme ou infirme la réponse de l'ordinateur. Si la réponse est fautive, l'ordinateur demande à l'utilisateur s'il souhaite repartir pour une autre série de questions. Au bout de 3 séries, si le jeu n'a pas deviné votre animal, il admet avoir perdu et demande à l'utilisateur la bonne réponse parmi une liste de réponses susceptibles de convenir. Il l'entre alors dans sa base de données, se nourrissant ainsi des réponses que vous avez données.

A la fin d'une partie, qu'elle soit gagnante ou perdante pour l'ordinateur, celui-ci demande à l'utilisateur s'il souhaite ajouter une question pertinente à propos de son animal, et propose ensuite à l'utilisateur d'y répondre pour 10 autres animaux proches, se trouvant déjà actuellement dans sa base

de données. Les propositions des utilisateurs sont évaluées par un modérateur avant d'être ajoutées à la BD. C'est ainsi que les joueurs enrichissent eux-mêmes la base de données. Le modérateur balaie régulièrement le contenu des BD et les nettoie. Ainsi les personnages qui ne sont pas joués suffisamment finissent par être éliminés automatiquement.

En fin de partie gagnée par l'application, le système affiche le nb de fois où l'animal a déjà été joué, et fournit la date et heure de la dernière partie concernée.

Le joueur a aussi la possibilité, une fois la partie achevée, de revoir toutes les questions posées, les réponses qui ont été données par le joueur et celles normalement attendues pour l'animal (rapport de partie). Il peut aussi, à chaque question, corriger la réponse de la question précédente.



*Pour info : Akinator fonctionne sur un moteur créé sur mesure, nommé **Limule**, qui est écrit en C++. Sa base de données MySQL contient environ 100 000 personnages. Le fonctionnement de Limule s'appuie sur un système de réseau de neurones. Le serveur du jeu repose sur 15 serveurs pour permettre aux 250000 joueurs quotidiens d'accéder rapidement à la BD sans problème.*

L'utilisateur n'a pas besoin de créer de compte pour jouer. Dans 85% des cas, le jeu devine en 2 séries (20 questions).

- 1- Élaborer le **DCU**
- 2- Construire le **Diagramme d'activités** du Cas « Lancer une partie »
- 3- Donner le **SCN Nominal** du Cas « Lancer une partie » et imaginer les exceptions
- 4- Construire le **Diagramme de Classes**
- 5- Élaborer le **Diagramme de Séquences** décrivant un aspect intéressant

Exercice 4 – Tamagoshi (suite)

Reprendre l'exercice de la fiche TD5.

Élaborer le **Diagramme de Séquences Objet** décrivant le processus d'approvisionnement quotidien des magasins par l'administrateur.