

<p>CONCOURS DE L'AGRÉGATION EXTERNE ÉCONOMIE ET GESTION SESSION 2011</p>

**ÉTUDE COMPORTANT L'UTILISATION
DE TECHNIQUES DE GESTION**

Informatique et gestion

Sujet n°2

CAS ENTRAINEUR POUR LES PERMIS PISTE

Durée de préparation : quatre heures

Durée de l'épreuve : une heure

Vous disposez d'une durée maximale de quarante minutes pour présenter oralement la solution de l'étude qui vous est proposée.

PRÉSENTATION DU CONTEXTE

Dans le cadre du plan de relance de 2009, le secrétariat à l'économie numérique a lancé un appel à projet de 10 M€ dédié aux « *serious game* » et a retenu 49 projets innovants, financés à partir de 2010.

Le cas qui est présenté est l'un de ces 49 projets, porté par l'entreprise Technicom qui est une PME de l'agglomération niçoise de 23 personnes et de 2,5 M€ de CA. Il porte sur la définition et la mise au point d'un entraîneur, simulateur dédié à la formation des conducteurs et à la délivrance d'un « permis piste » aux agents et manutentionnaires conduisant des véhicules dans l'enceinte des aéroports.

Les difficiles conditions de circulation sur un aéroport international

La première expérimentation de cet outil de simulation a été faite en 2010 sur l'aéroport de Nice-Côte d'Azur. Avec ses 10 millions de passagers et ses 120 000 atterrissages annuels, cet aéroport est le premier aéroport de province français. Des aéronefs de tailles très différentes (hélicoptère, avion d'affaire, avion de ligne...) se partagent les deux pistes de plus de trois kilomètres et les aires de stationnement et de manœuvre qui se répartissent sur plus de 300 hectares.

La circulation y est difficile et risquée car le trafic est très dense : plus d'un avion atterrit ou décolle toutes les deux minutes et jusqu'à 25 avions circulent au sol simultanément sur les aires de manœuvre aux heures de pointe. De nombreux véhicules de manutention circulent autour des avions. Par exemple, pour un avion comme l'Airbus A380, il ne faut pas moins de 16 véhicules de reconditionnement (camion citerne, *catering*, équipe de ménage, manutention, mécaniciens...). Il y a plus de 325 véhicules agréés (véhicules de gendarmerie, de manutention...) dont environ 120 sont en permanence en train de circuler. Cela occasionne un trafic extrêmement dense et difficile à coordonner, d'autant plus que les avions peuvent être masqués par les nombreux véhicules et installations présents.

Les pilotes d'avion ont eux même une visibilité assez réduite au sol et ne peuvent freiner trop rapidement en raison des risques qu'ils feraient encourir aux passagers. Les accidents sont fréquents sur les pistes (114 accidents mettant en cause un véhicule et un avion pour l'année 2009 en Europe, dont 16 pour les aéroports français). Le souffle des réacteurs à plein régime peut facilement renverser un bus et les angles de visibilité sont assez réduits. La nuit ou lors de mauvaises conditions météorologiques, les problèmes de circulation de piste sont encore bien plus difficiles.

Les conducteurs de véhicules de piste sont en contact radio à la fois avec la tour de contrôle et le service des opérations au sol. Ils doivent donc, bien sûr maîtriser les règles spécifiques à la circulation des aéroports, mais aussi certains éléments de base de la phraséologie aéronautique.

DOSSIER 1 – La formation aux permis piste

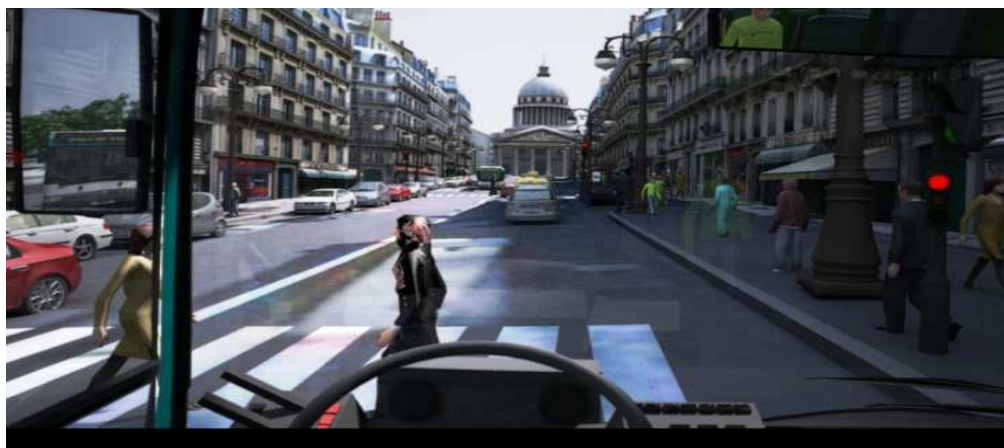
Jusqu'à maintenant, les permis pistes sont délivrés pour 3 ans et ne sont sanctionnés que par un examen théorique sous forme de QCM à l'issue d'une formation d'environ 40 heures. Il n'y a pas à proprement parler de formation pratique. Elle se fait « sur le tas » avec un chef de service qui accompagne pendant quelques jours le détenteur du permis piste et qui décide lorsqu'il pourra le laisser circuler seul dans son véhicule.

Il y a deux types de permis piste : le permis T et le permis TZ. Le premier donne droit à un accès à l'enceinte de l'aéroport mais le conducteur ne doit pas s'approcher des avions en marche et doit se cantonner aux « chemins bleus » (des bandes de peinture bleue matérialisent ces chemins). Le permis TZ permet à son détenteur de traverser les pistes, de s'approcher des avions et de circuler sur les « chemins jaunes ». Enfin, les chemins rouges ne sont accessibles qu'aux services de sécurité. Dans tous les cas, l'obtention du permis piste est subordonnée à celle du permis B (le permis voiture classique).

Conception d'un entraîneur

L'entraîneur développé par la société Technicom est un simulateur ayant une double vocation : d'une part, familiariser les futurs détenteurs du permis piste à un environnement de conduite très spécifique et potentiellement dangereux ; d'autre part, élaborer un démonstrateur en vue de standardiser la formation pratique pour la proposer à d'autres aéroports. Cette standardisation devrait donner à Technicom un avantage concurrentiel et une notoriété pour diffuser son produit à d'autres aéroports, tout en maîtrisant l'accès aux données pendant un temps suffisant pour gêner les entrants potentiels. À terme, il est également prévu que l'entraîneur puisse être décliné pour des métiers spécifiques comme les sapeurs pompiers d'aéroports.

Technicom développe déjà des simulateurs pour l'industrie (manutention des fûts de produits dangereux) ainsi qu'un entraîneur destiné à la sélection et l'entraînement des conducteurs de bus de la RATP (régie autonome des transports parisiens). L'image suivante donne un aperçu de ce qui déjà été développé par Technicom pour la RATP.



Pour construire son entraîneur, Technicom a récupéré les modèles de texture (granularités des matériaux, dégradés de couleurs, ombrages...) qu'elle détenait et elle les a transposés aux infrastructures de l'aéroport de Nice. Elle a du ensuite modéliser les bâtiments, ce qui a représenté un travail de plusieurs mois. La difficulté n'est ici pas réellement technique mais plutôt liée à l'actualisation des bâtiments préfabriqués et provisoires. En effet, ces types constructions ne sont pas statiques et peuvent être déplacés ou changer de forme et de configuration au fil des semaines.

Actuellement, les apprentis conducteurs utilisent l'entraîneur sur un simple ordinateur de bureau muni d'un navigateur avec trois écrans et un kit volant et pédales. La partie graphique est reconstituée par trois vidéo-projecteurs qui projettent les scènes à 60° chacun sur trois écrans blancs, ce qui donne donc une scène à 180°.

Le déroulement de l'apprentissage

Après avoir effectué et validé la formation théorique de 40 heures, le moniteur paramètre l'entraîneur afin de mettre l'apprenant dans des situations de conduite. Le moniteur définit chaque mission *a priori* (type de véhicule, nombre d'avions sur la piste, message de la tour de contrôle...). Chaque type de véhicule est répertorié en fonction des données de leur constructeur (rayon de braquage, performances, dimensions, poids...).

Le paramétrage s'effectue à partir de "scénarios type" dans une base de 11 scénarios déjà paramétrés, en fonction des réactions que le moniteur veut tester sur son apprenant. Cette base de scénarios contient des thèmes prédéfinis (panne, piéton-passager égaré, accident...) qui peuvent se combiner entre eux. Le moniteur a donc la possibilité de créer des scénarios par héritage et d'ajouter des scènes ou des difficultés supplémentaires (nombre et vitesse des objets circulants, pluie...). Les performances et les dimensions des véhicules sont également paramétrables.

La validation du permis piste sur l'aéroport de Nice

La préparation du permis piste nécessite dans un premier temps l'obtention de l'examen théorique. Ensuite des missions sur l'entraîneur sont à effectuer, avec un score acceptable à la clé. Enfin, les candidats au permis piste passent un examen sur l'entraîneur qui sera, comme le permis piste actuel, valable trois ans. Après ce délai de trois ans, 6 heures d'entraînement et l'examen devront être passés ; les conducteurs détenant déjà le permis piste ancienne version devront également se soumettre aux 6 heures d'entraînement, mais ne passeront l'examen qu'à la fin de la période de validité de leur permis.

Il n'est pas prévu d'examen réel dans l'enceinte de l'aéroport pour des raisons de sécurité et d'organisation (le nombre de candidats annuel est trop important par rapport au nombre de moniteurs et aux taux d'occupation des véhicules).

L'aéroport de Nice est donc le premier à établir une formation au permis piste incluant une formation à la fois théorique et pratique ; même si elle ne donne pas dans l'immédiat de diplôme officiel, puisque la formation n'est pas encore standardisée. Cependant, elle est en accord avec la charte de la chambre de commerce et d'industrie qui est cogérante de l'aéroport. Elle a vocation à être proposée au Ministère des Transports comme un standard destiné à

s'étendre à tous les aéroports français, voire européens par l'intermédiaire des normes EASA (*European Aviation Safety Agency*). L'enjeu pour Technicom est donc de jeter les bases d'une standardisation des formations aux permis piste pour que l'EASA édite un règlement et la rende obligatoire. Cela lui permettrait de faire valoir son expérience pour définir la doctrine de formation ainsi que les points clés de l'utilisation des entraîneurs pour les vendre à d'autres aéroports. Le marché est potentiellement important, en raison du fort taux de rotation des emplois des manutentionnaires d'aéroport (280 personnels formés ou renouvelés en 2010 pour le seul aéroport de Nice). De plus, le volume horaire d'entraînement pourrait être augmenté et il est envisageable de délivrer des permis piste spécifique à chaque aéroport. Si cet aspect était validé, le marché des entraîneurs serait encore plus important, puisqu'il faudrait alors modéliser précisément chaque site.



Question 1.1 - Analyser quelles doivent être les qualités d'un simulateur à usage professionnel comme celui conçu par Technicom.

Question 1.2 - Identifier les avantages et les risques pour l'aéroport induits par les nouvelles modalités de préparation et d'obtention des permis piste. Que pourriez-vous proposer pour améliorer encore la formation ?

Question 1.3 – Indiquer dans quelle mesure l'entraîneur développé par Technicom peut être qualifié de « *Serious game* ». Développez votre argumentation après avoir défini ce terme et ses usages.

DOSSIER 2 : Suivi des performances de l'apprenant

Le cahier des charges de réalisation de l'entraîneur prévoit un module d'évaluation des missions effectuées par l'apprenant.

La simulation se découpe en 6 missions qui visent des situations différentes. Au début de chaque mission, l'apprenant se voit fixer des objectifs à accomplir. Il doit donc être capable d'associer à ces différents objectifs les actions correspondantes tout en respectant les contraintes d'antériorité. En effet, certaines actions ne peuvent être exécutées si d'autres n'ont pas été lancées auparavant. Pour chaque action, une évaluation est réalisée : un score de l'apprenant est associé à chacune d'elle. Pour passer d'une action à l'autre, des conditions liées au score sont intégrées à l'entraîneur. Pour certaines, l'apprenant doit avoir réalisé un score minimum ; pour d'autres c'est la combinaison du score obtenu lors de l'action avec le score obtenu aux actions exécutées précédemment dans la mission qui conditionnent le passage à une autre action.

L'ensemble des actions effectuées par l'apprenant est donc mémorisé. Chaque action est associée à des indicateurs de poids variable (parmi ces indicateurs figure le temps passé à la réalisation de l'action). Pour définir le score alloué à l'apprenant pour une action, les valeurs avant et après des différents indicateurs utilisés par le moteur de simulation sont stockées. La comparaison des deux valeurs et le poids relatif de chaque indicateur définissent le score de l'apprenant.

Question 2.1 - Élaborer une modélisation conceptuelle des données liées au suivi des performances de l'apprenant dans l'entraîneur, en explicitant les hypothèses retenues.

Question 2.2 - Présenter le schéma relationnel d'une base de données qui permettrait de stocker les données liées au suivi des performances de l'apprenant.

Question 2.3 - Proposer une requête SQL permettant d'éprouver le schéma proposé et répondant aux besoins d'évaluation du score d'un apprenant.

L'entraîneur a été développé avec le langage *Flash ActionScript 3* (voir la présentation du langage en *annexe 1*).

Question 2.4 - Présenter une architecture pour l'application de suivi des performances de l'apprenant.

Question 2.5 - Proposer une architecture technique pour l'exploitation de cette application, en intégrant les problématiques de l'hébergement et de l'administration de l'application (notamment l'identification des apprenants, agents de l'aéroport).

À la fin de chaque mission, un bilan est affiché à l'apprenant qui tient compte du score des actions qu'il a réalisées. Ce bilan est construit dynamiquement par comparaison entre les scores obtenus lors des actions par l'apprenant et les règles stockées dans le logiciel.

Le bilan de l'apprenant présente le score de la mission, l'accomplissement ou non des objectifs de la mission et le score obtenu pour les actions réalisées. Pour chaque action, une évaluation sous forme textuelle est présentée avec, le cas échéant, une analyse de son échec.

Question 2.6 - Compléter l'analyse par une modélisation des données relatives au bilan de la mission, en explicitant les hypothèses retenues.

DOSSIER 3 : intégration de l'entraîneur dans un système de gestion de l'apprentissage

Si jusqu'à présent l'entraîneur permet à l'apprenant de se préparer à l'examen, il pourrait être intégré dans un système de gestion de l'apprentissage (*learning management system, LMS*) respectant le standard SCORM (*Sharable Content Object Reference Model*). L'annexe 2 présente les enjeux du standard SCORM.

Question 3.1 – Présenter les avantages et les inconvénients, pour les différentes parties prenantes, de migrer l'entraîneur dans un système de gestion de l'apprentissage.

L'intégration de l'entraîneur dans le LMS nécessite d'enregistrer les apprenants dans cet outil. Une extraction des coordonnées des apprenants au format XML a été effectuée selon une structure conforme à la base de données du LMS. Un extrait du fichier obtenu est présenté en *annexe 3.1*. La structure de la table de la base de données du LMS qui stocke les coordonnées des apprenants est décrite en *annexe 3.2*.

Question 3.2 - Écrire un programme permettant le transfert des coordonnées des apprenants dans la base de données du LMS, en mobilisant les classes définies en *annexe 4* et en définissant éventuellement le-s composant-s supplémentaire-s nécessaire-s à l'implémentation de la solution.

ANNEXE 1 : Définition du langage Action Script

Extrait de Wikipédia : <http://fr.wikipedia.org/wiki/ActionScript>

ActionScript est le langage de programmation utilisé au sein d'applications clientes (Adobe Flash, Adobe Flex) et serveur (Flash media server, JRun, Macromedia Generator).

ActionScript est un langage de script, orienté objet et prototype, basé sur ECMAScript (conforme à 100% avec la norme ECMA-262, révision 3, depuis ActionScript 3.0). ECMAScript étant une version standardisée de JavaScript, ActionScript et JavaScript partagent une syntaxe semblable.

Ce langage permet d'ajouter de l'interactivité aux animations Flash, en répondant aux actions de l'utilisateur, et en pilotant les movie clip (conteneurs graphiques permettant de hiérarchiser les animations), et les différents objets multimédias (images, son, vidéo...). Il permet également la communication de l'application avec le serveur, notamment par le chargement de fichiers ou la communication avec un langage serveur comme le PHP.

ANNEXE 2 : Ce que vise le standard SCORM

Extrait de l'article du 12 septembre 2008 du site <http://www.scorm.fr/?p=44>

Les pionniers de l'EAO¹ puis de la FAD² ont reconnu très tôt le besoin de disposer d'un modèle de référence qui précise les contenus d'apprentissage, de même que leur identification, leur stockage et leur présentation dans l'apprentissage à distance. Le SCORM représente un modèle de coordination conçu de façon à fournir à l'apprentissage en ligne un ensemble de pratiques normalisées qui peuvent être acceptées généralement et être mises en œuvre sur une vaste échelle.

[...]

Le SCORM décrit un « modèle d'agrégation du contenu » et un « environnement d'exécution » pour les objets d'apprentissage afin d'appuyer un enseignement adaptatif fondé sur les objectifs, les préférences, le rendement et d'autres facteurs liés à l'apprenant (notamment les techniques d'instruction). Le SCORM décrit également un modèle de « séquençement et navigation » pour la présentation dynamique, et selon les besoins de l'apprenant, des objets d'apprentissage.

Les 6 capacités définies par SCORM

Pour favoriser l'obtention d'une entente avec l'industrie et concrétiser un tel modèle, ADL³ a défini des exigences élevées pour tous les environnements d'apprentissage basés sur le SCORM. Ces exigences sont appelées les « capacités » d'ADL et constituent la base de tous les changements et les ajouts concernant le SCORM. Ces « capacités » sont les suivantes :

- Abordabilité : capacité à augmenter l'efficacité et la productivité en réduisant le temps et les coûts nécessaires pour dispenser la formation.
- Accessibilité : capacité de repérer des composants d'enseignement à partir d'un site distant, d'y accéder et de les distribuer à beaucoup d'autres sites.
- Adaptabilité : capacité à personnaliser la formation en fonction des besoins des personnes et organisations.
- Durabilité : capacité de résister à l'évolution et aux changements de la technologie sans avoir recours de nouveau à la conception, à la configuration et au codage, qui sont des processus coûteux.
- Interopérabilité : capacité d'utiliser, dans un autre emplacement et avec un autre ensemble d'outils ou sur une autre plate-forme, des composants d'enseignement développés dans un site avec un certain ensemble d'outils ou sur une certaine plate-forme.
- Réutilisabilité : souplesse permettant d'intégrer des composants d'enseignement dans des contextes et des applications multiples.

¹ Enseignement assisté par ordinateur

² Formation à distance

³ Advanced Distributed Learning

ANNEXE 3 : Description des apprenants

Annexe 3.1 : Flux XML de description d'apprenants

```
<?xml version="1.0" encoding="utf-8" ?>
<eleves>
  <eleve code="e001" >
    <nom> Dupont </nom>
    <prenom> Pierre </prenom>
    <dateNaissance> 02/02/1968 </dateNaissance>
  </ eleve >
  <eleve code="e002" >
    <nom> Yayaoui </nom>
    <prenom> Mehdi </prenom>
    <dateNaissance> 06/06/1988 </dateNaissance>
  </ eleve >
  <eleve code="e003" >
    <nom> Marceau </nom>
    <prenom> Adeline </prenom>
    <dateNaissance> 04/03/1978 </dateNaissance>
  </ eleve >
</eleves>
```

Annexe 3.2 : Table ELEVE de la base de données du LMS

ELEVE (id, nom, prenom, dateNaissance)

id : clef primaire

ANNEXE 4 : Classes pour le parcours d'un document XML

Classe NoeudXml

Cette classe représente un élément ou un attribut XML.

Classe NoeudXml

Privé

nom, valeur : chaîne // nom et valeur du nœud.

...

Public

fonction getNom() : chaîne // retourne le nom du nœud XML.

fonction getValeur() : chaîne // retourne la valeur du nœud XML.

fonction nbFils() : entier

// Retourne le nombre d'éléments fils du nœud courant s'il s'agit d'un élément XML.

// Retourne -1 s'il s'agit d'un attribut XML.

fonction getFils(unIndex : entier) : NoeudXml

// Retourne l'élément fils d'index unIndex si le nœud courant est un élément XML.

// Le premier élément fils est à l'index 0.

// Retourne null si le nœud courant est un attribut XML.

fonction nbAttributs() : entier

// Retourne le nombre d'attributs XML du nœud courant s'il s'agit d'un élément XML.

// Retourne -1 s'il s'agit d'un attribut XML.

fonction getAttribut(unIndex : entier) : NoeudXml

// Retourne l'attribut XML d'index unIndex si le nœud courant est un élément XML.

// Le premier attribut est à l'index 0.

// Retourne null si le nœud courant est un attribut XML.

...

Fin Classe

Classe DocXml

Cette classe permet de parcourir un document XML après l'avoir chargé en mémoire.

Classe DocXml

Privé

...

Public

DocXml()

// constructeur

procédure charger(nomFichier : chaîne)

// charge l'objet DocXml à partir du fichier nomFichier.

fonction racine() : NoeudXml

// Retourne la racine du document XML.

...

Fin Classe