

Projet Java – Propositions de sujets - E22

Sujet I : « Unlock »

Objectif global :

Unlock transpose le concept des *escape-games*, un scénario qu'il faut réaliser en temps limité (*en général 1h*). Le jeu comporte un paquet de cartes virtuelles numérotés.

Détails :

La première carte du paquet sert à lancer le scénario.

La carte comporte de nombreux chiffres, parfois cela peut être des lettres. Chacun correspond à une carte du paquet que vous devez poser face visible sur la table. Cela peut être par exemple un indice, un objet ou un nouvel endroit où aller. C'est avec ces différentes cartes que vous devrez interagir pour avancer dans l'histoire, résoudre les énigmes et aboutir à une solution. Mais pour cela, il faudra les combiner judicieusement.

Rapidement, les cartes s'accumulent. Heureusement, celles-ci sont à usage unique et donc une fois utilisée on peut les défausser. Certaines cartes peuvent être combinées pour en révéler de nouvelles. Attention cependant, il est déconseillé d'essayer toutes les combinaisons les unes après les autres car certaines vous vaudront des pénalités.

Certaines comportent des indices plus ou moins bien cachés dessus. Et certaines, enfin, débloquent des mini-jeux interactifs sur l'application pour obtenir des indices ou le numéro de nouvelles cartes. On découvre le scénario de carte en carte et d'énigme en énigme. Certaines n'ont simplement aucune utilité, sinon vous faire perdre du temps en les analysant.

Les machines, ensuite, correspondent à des mini-jeux plus ou moins simples présents dans l'application.

L'application s'occupe également de tout le côté administratif. Elle gère le chronomètre, elle permet également d'obtenir de l'aide sur une carte ou sur une énigme, d'écouter des messages ou d'entrer des codes pour déverrouiller une porte ou un mécanisme. Il ne faut pas en abuser car ils diminuent votre score final.

Aspects pratiques :

Une classe carte (numéro, type : puzzle/machine/code/pièce), les différents types seront des sous-classes : puzzle (couleur, numéro), machine (selon le scénario), code (plusieurs réels), pièce (plusieurs réels et ensembles de caractères).

Une interface permettra d'afficher les différentes cartes obtenues (avec leur image) au fur et à mesure, de rentrer les codes et éventuellement manipuler une machine.

Sujet II : Application N7

Objectif global :

Le but de ce sujet serait de créer une application qui pourrait être utile à chaque élève de l'n7. En analysant ce qu'il nous manquait ou qui pourrait être amélioré au sein de l'école, nous sommes venus à l'idée de créer une application qui présenterait différentes fonctionnalités.

Les fonctionnalités envisagées :

❖ Partie scolaire :

- Un annuaire regroupant toutes les informations utiles concernant les enseignants/ intervenants/ encadrants.

❖ Partie associative :

- Trouver facilement un ami au sein de l'n7 et indiquer le plus court chemin pour aller le rejoindre.
- Pouvoir « matcher » et faire des demandes d'amis avec des personnes de l'n7.
- Possibilité de commander à distance des produits vendus au foy', avant qu'il n'en reste plus.

Détails :

Chaque élève de l'n7 serait représenté par un objet Elève, par exemple, comprenant comme attributs, le nom, le prénom, la localisation, et sa relation par rapport à l'utilisateur.

Chaque enseignant/ intervenant/ encadrant serait représenté par un objet Sup, par exemple, comprenant le nom, le prénom, l'email, son métier, le numéro de téléphone éventuellement.

Ainsi une classe Personne serait créée en amont, et Elève et Sup seraient des sous-classes de cette super-classe.

La relation d'un élève par rapport à l'utilisateur serait hiérarchisée. Si l'élève se situe au niveau le plus proche de l'utilisateur, il pourra accéder à la localisation de ce dernier au sein de l'n7 et ainsi pouvoir le retrouver facilement, et pourra aussi matcher avec lui. S'il se situe dans un niveau plus éloigné, il ne pourra ni accéder à sa localisation, ni matcher avec lui.

Ayant remarqué qu'au foy' certains produits partent vite aux pauses, on pourrait proposer de pouvoir commander à l'avance, ce qui permettrait aussi de réduire l'attente lors des commandes.

Une classe Produit pourrait être créée et aurait comme attributs le nom du produit, son prix et sa quantité présente.

L'annuaire permettrait aux élèves d'avoir accès rapidement aux informations utiles des intervenants en les regroupant toutes au même endroit.

Les élèves accèderaient à l'application grâce à leurs identifiant et mot de passe INP, un premier menu serait créé.

Toute cette application comprendrait des menus créés par exemple avec la classe JMenuBar (pour le menu principal), JMenuItem (pour les sous-menus), JPanel (avec notamment JButton, JFrame ...).

Sujet III : Un Labyrinthe

Objectif : Incarner un personnage coincé dans un labyrinthe qui doit récupérer pièce et bonus , le labyrinthe serait semé de piège . Il existerait plusieurs niveaux et le personnage aurait une barre de vie.

Détail: Le personnage serait représenté par ses coordonnées, sa barre de vie et le nombre de pièces qu'il a ramassé

On créerait une classe labyrinthe.

Sujet IV : Application automatisée d'assistance aux juges arbitres de tennis

Objectif global :

Les tournois de tennis amateurs se déroulent généralement sur une durée de 2 semaines. Le nombre de participant pour un seul tournoi peut parfois atteindre 200 personnes et cela nécessite une grande organisation.

Cette application aurait pour but de générer un tableau échelonné adéquat (respect du niveau de chacun), fournir un calendrier respectant toutes les contraintes (respect du temps de repos de chacun, terrains disponibles, respect des disponibilités des joueurs ...) et de faciliter la vie du juge arbitre sur les deux semaines de tournoi (saisie des résultats rapide et simple, actualisation automatique du calendrier de programmation à chaque résultat,

Fonctionnalités envisagées :

1.

Création du tournoi. Le JA entre les dates du tournoi, la fiche de chaque participant (infos, classement, dispos, inscription réglée ou non..), les terrains à disposition pour générer un calendrier. De cette façon, dès qu'un joueur s'inscrit, l'application peut envoyer un sms automatique pour avertir le joueur quel jour il risque de rentrer dans le tournoi

2. Génération d'un tableau échelonné. Les inscriptions sont ouvertes à tous les niveaux et donc les entrées des joueurs dans le tournoi doivent se faire de façon échelonnée de façon à avoir des matchs de même niveau tout au long du tournoi. Les joueurs issus d'un même club doivent être positionnés dans le tableau de façon à se rencontrer que le plus tard possible dans le tournoi. Le tableau est actualisée à chaque saisie de résultat.
3. Génération automatique de la programmation des matchs. L'application fournit un programme de la journée suivant le nombre de terrains disponibles et selon les disponibilités des joueurs. Le programme des jours suivants s'actualisent dès qu'un résultat est saisi ou dès que les disponibilités d'un joueur changent. Il sera également important de respecter le temps de repos de chacun et qu'il soit équitable pour chacun des joueurs.
4. Convocation par sms automatique. Les joueurs doivent être prévenus idéalement 48h avant le début de leur match. Un sms sera alors envoyé pour les en informer.
5. D'autres fonctionnalités peuvent être envisagées...

Sujet V : Un « Killer » statistique

Objectif global :

Ce sujet se baserait sur le jeu du « Killer » : chaque participant reçoit une mission à effectuer envers une personne et, si il l'accomplie, il élimine la personne du jeu et reçoit à nouveau une mission à réaliser envers une autre personne encore en jeu. La partie se termine lorsqu'il ne reste plus qu'un joueur.

Les fonctionnalités envisagées :

❖ Partie avec mort subite :

Une ou des personnes jouent. Elles peuvent ajouter des joueurs virtuels s'ils souhaitent. Pour tenter de réussir une mission, chaque joueur doit grâce à une interface graphique se déplacer jusqu'à sa cible (qu'il peut distinguer facilement graphiquement) et la toucher sans se faire atteindre par son chasseur. Une fois qu'il touche une personne, il aurait une certaine probabilité de le tuer. Cette probabilité serait initialisée à une valeur choisie par l'utilisateur. A chaque fois qu'il réussit il augmente sa probabilité de réussir son prochain défi. S'il échoue, la partie s'arrête et un classement s'affiche, déterminé en fonction des probabilités des joueurs lorsqu'ils sont morts.

❖ Partie sans mort subite :

Le principe du jeu reste inchangé à part que lorsqu'une personne meurt, il réintègre directement la partie avec une probabilité de réussir son défi plus faible que celle qu'il avait avant de mourir. La partie s'arrête lorsqu'un joueur arrive à une probabilité de 100% de réussir son défi.

Détails :

Chaque joueur serait représenté par l'objet Killer comprenant un pseudo, une couleur d'affichage, une représentation de personnage, le nombre de personnes tuées (et le nombre de fois où il est mort dans le cas de la partie sans mort subite) et la probabilité qu'il a de tuer une personne.