

Pendant ce TP vous allez apporter des améliorations à un logiciel existant, parmi les projets Drag'genda, Monopoly tradition et Casino.

1 Sur quel projet je dois travailler ?

Le sujet sur lequel vous allez travailler dépend du sujet sur lequel vous avez travaillé pendant le TP 2 :

Projet du TP 2	Projet pour ce TP
Drag'genda	Monopoly tradition
Monopoly tradition	Casino
Casino	Drag'genda

Si vous tombez sur le projet dont vous êtes l'auteur, prendre le troisième des projets (ni celui sur lequel vous avez travaillé au TP2, ni celui dont vous êtes auteur).

2 Préparation

Pour votre travail de maintenance vous pouvez utiliser le diagramme UML qui a été fait par vos collègues pendant le TP2. Suivre les indications de l'enseignant-e pour savoir quel diagramme UML utiliser.

3 Modifications à apporter

3.1 Drag'genda

Améliorer la saisie La saisie d'un événement est actuellement trop fastidieuse. Nous voulons avoir une saisie où plusieurs éléments sont saisis à la fois, en suivant un format de clé :valeur.

Par exemple, les saisies ci-dessous doivent être possibles :

`ev:anniversaire debut:14/10/2015;20:00 fin:23:00 descr:chez lulu`

`ev:courses Noël debut:23/04/2016;16:00 duree:3:00`

`ev:exam production d'applications debut:14/02/2016;15:30 descr:je dois réviser`

Remarquez que si la date de fin est la même que la date de début, il suffit de donner l'heure de fin (comme dans le premier exemple ci-dessus).

Si un élément requis manque, sa saisie sera demandée. Par exemple, le troisième événement ci-dessus ne spécifie pas de date et heure de fin, il faut demander de le préciser.

Vous êtes libres de proposer un format pour les saisies clé-valeur qui peut être différent que celui ci-dessus. L'exigence principale des utilisateurs est d'avoir une saisie plus succincte, pouvant se faire sur une seule ligne.

Pensez également à afficher une aide pour la saisie.

Ajouter des participants à un événement Lors de la création d'un événement, on veut pouvoir ajouter des participants parmi les autres utilisateurs de l'agenda. Si un participant est ajouté,

l'événement apparaît dans son agenda aussi. Concevoir une interaction avec l'utilisateur qui permettrait de le faire, et l'implémenter.

3.2 Monopoly tradition

Améliorer l'affichage On constate que l'affichage pour le jeu est très illisible, aussi bien pour le plateau de jeu que pour les informations données au utilisateurs.

Actuellement le jeu propose un affichage seulement d'une portion du plateau de jeu. On voudrait avoir une vue globale à chaque tour de jeu. Celle-ci doit indiquer pour chaque case le type de case, à qui elle appartient, combien de maisons ou hôtels s'y trouvent. On doit également voir la position de tous les joueurs. Finalement, chaque joueur doit trouver l'information de la somme en sa possession. Tout ceci doit être présenté d'une manière lisible.

Modifier l'affichage pour répondre à cette exigence. Vous pouvez commencer par une vue linéaire du plateau, mais la version finale devrait afficher le plateau en carré, comme sur un vrai plateau.

Améliorer les saisies Les saisies par menus à numéros ne sont pas très pratiques. Il est préférable de pouvoir saisir une lettre, ou un mot, qui a du sens par rapport à l'option qu'on choisit. Par exemple,

Que voulez-vous faire?

1. Construire une maison
2. Passez votre tour

Au lieu de saisir 1 ou 2, on préfère saisir C pour construire, et P pour passer. Ceci permettrait également d'afficher la question sur une seule ligne :

Que voulez vous faire ? [C]onstruire ou P[asser] ?

3.3 Casino

Finir le jeu Black Jack Actuellement le jeu ne fonctionne pas. Terminez et tester son implémentation.

Faciliter la saisie d'une mise Actuellement, pour faire une mise, le joueur doit ajouter un jeton à la fois, ce qui est assez fastidieux. Vous devez implémenter la possibilité de saisir sa mise en une seule ligne. Par exemple,

pair noir 0 20 D1

indique une mise sur pair, noir, les nombres 0 et 20, et la première douzaine.

Dans une deuxième version, vous allez donner la possibilité de spécifier le nombre de jetons sur chacune des cases. Par exemple,

pair->2 noir->3 0->1

indique qu'on désire placer 2 jetons sur pair, 3 jetons sur noir, et 1 jeton sur le nombre 0.