Gaillard Victor

Challet Tristan

Preuves Jeu Java FX

***Je maîtrise les règles de nommage Java.***

Dans Dev/src/model/metier nous avons fait la classe Entite qui respecte les règles de nommage.

***Je sais binder bi directionnellement deux propriétés Java FX.***  
Dans Dev/src/views/codeBehind le code behind de GameWindow contient un attribute FXML binder bidirectionnellement ligne 79.

***Je sais binder uni directionnellement deux propriétés Java FX.***  
Dans Dev/src/model/metier nous avons la classe Monde qui contient des attributs DoubleProperty sur la largeur et la hauteur de la fenêtre lignes 30 et 41.

Ces attributs sont bind dans Dev/src/views/codeBehind dans le code behind de la fenetre FXML GameWindow sur la widthProperty et heightProperty du mainPane lignes 70 et 71.

***Je sais coder une classe Java en respectant des contraintes de qualité de lecture de code.***  
Dans Dev/src/model/metier nous avons fait la classe Entite qui respecte les contraintes avec par exemple les attributs DoubleProperty qui ont leurs getteur/setteur rassemblés.

***Je sais contraindre les éléments de ma vue, avec du binding FXML.***

***Je sais définir une CellFactory fabriquant des cellules qui se mettent à jour au changement du modèle.***

***Je sais éviter la duplication de code.***  
Dans Dec/src/views/codeBehind nous avons le code Behind GameWindow qui contient une méthode createBinding ligne 45. Elle permet de créer un binding sur une Entite.

***Je sais hiérarchiser mes classes pour spécialiser leur comportement.***

Dans Dev/src/model/deplacement nous avons des interfaces pour les déplacements dans chaque direction. DeplaceurNormal les implémentes et DeplaceurNormalVitesse 1 ou 2 héritent de DeplaceurNormal.

***Je sais intercepter des évènements en provenance de la fenêtre Java FX.***

Dans Dev/src/model/controls nous avons une classe KeyListener qui permet de récupérer les touches du clavier qui sont appuyées.

***Je sais maintenir, dans un projet, une responsabilité unique pour chacune de mes classes.***

Nous avons séparés la gestion de la collisions et le déplacement dans deux classes différentes.

***Je sais gérer la persistance de mon modèle.***

***Je sais utiliser à mon avantage le polymorphisme.***

***Je sais utiliser GIT pour travailler avec mon binôme sur le projet.***  
Nous avons utilisé le GitLab de l’université pour travailler sur le projet.

***Je sais utiliser le type statique adéquat pour mes attributs ou variables.***

Dans le manager, nous utilisons aux ligne 131 & 132, nous utilisons le type statique Observer pour deux instance d’objets de classes différentes.

***Je sais utiliser les différents composants complexes (listes, combo...) que me propose Java FX.***

***Je sais utiliser les lambda-expression.***

Le constructeur de ManagerVue possède une lambda-expression.

***Je sais utiliser les listes observables de Java FX.***  
Dans Dev/src/model/metier nous avons une classe Monde qui contient en attribut ligne 19 une ListProperty des entités.

***Je sais utiliser un convertisseur lors d’un bind entre deux propriétés Java FX.***

Dans GameWindow, de la ligne 79 à 89, on utilise un StringConverter pour convertir un Integer en String durant le binding.

***Je sais utiliser un fichier CSS pour styler mon application Java FX.***  
Dans Dev/rsrc/fxml/css nous utilisons un fichier style.css.

***Je sais utiliser un formateur lors d’un bind entre deux propriétés Java FX.***

J***e sais développer un jeu en Java FX en utilisant FXML.***

Notre programme se lance et on peut jouer au jeu.

***Je sais intégrer, à bon escient, dans mon jeu, une boucle temporelle observable.***

Nous lançons une nouvelle boucle temporelle à chaque lancement de partie dans la méthode startGame() de Manager, ligne 180.