|  |  |
| --- | --- |
|  |  |



**Projet JAVAFX**

**DUT - INFO** PERALDE François

**2A – G5 - G6** PEYRON Hugo

Année : 2020 / 2021

Table des matières

[Documentation 4](#_Toc61971261)

[Je sais concevoir un diagramme UML intégrant des notions de qualité et correspondant exactement à l'application que j'ai à développer : 4](#_Toc61971262)

[Je sais décrire un diagramme UML en mettant en valeur et en justifier les éléments essentiels : 4](#_Toc61971263)

[Je sais documenter mon code et en générer la documentation : 4](#_Toc61971264)

[Je sais décrire le contexte de mon application, pour que n'importe qui soit capable de comprendre à quoi elle sert : 4](#_Toc61971265)

[Je sais faire un diagramme de cas d'utilisation pour mettre en avant les différentes fonctionnalités de mon application : 4](#_Toc61971266)

[Code 4](#_Toc61971267)

[Je maîtrise les règles de nommage Java : 4](#_Toc61971268)

[Je sais binder bidirectionnellement deux propriétés JavaFX.: 4](#_Toc61971269)

[Je sais binder unidirectionnellement deux propriétés JavaFX.: 4](#_Toc61971270)

[Je sais coder une classe Java en respectant des contraintes de qualité de lecture de code : 5](#_Toc61971271)

[Je sais contraindre les éléments de ma vue, avec du binding FXML : 5](#_Toc61971272)

[Je sais définir une CellFactory fabriquant des cellules qui se mettent à jour au changement du modèle : 5](#_Toc61971273)

[Je sais développer une application graphique en JavaFX en utilisant FXML : 5](#_Toc61971274)

[Je sais éviter la duplication de code : 5](#_Toc61971275)

[Je sais hiérarchiser mes classes pour spécialiser leur comportement : 5](#_Toc61971276)

[Je sais intercepter des évènements en provenance de la fenêtre JavaFX : 5](#_Toc61971277)

[Je sais maintenir, dans un projet, une responsabilité unique pour chacune de mes classes : 5](#_Toc61971278)

[Je sais gérer la persistance de mon modèle : 5](#_Toc61971279)

[Je sais surveiller l’élément sélectionné dans un composant affichant un ensemble de données : 6](#_Toc61971280)

[Je sais utiliser à mon avantage le polymorphisme : 6](#_Toc61971281)

[Je sais utiliser certains composants simples que me propose JavaFX : 6](#_Toc61971282)

[Je sais utiliser certains layout que me propose JavaFX : 6](#_Toc61971283)

[Je sais utiliser GIT pour travailler avec mon binôme sur le projet : 6](#_Toc61971284)

[Je sais utiliser le type statique adéquat pour mes attributs ou variables : 6](#_Toc61971285)

[Je sais utiliser les collections : 6](#_Toc61971286)

[Je sais utiliser les différents composants complexes (listes, combo...) que me propose JavaFX : 6](#_Toc61971287)

[Je sais utiliser les lambda-expression : 6](#_Toc61971288)

[Je sais utiliser les listes observables de JavaFX : 6](#_Toc61971289)

[Je sais utiliser un convertisseur lors d’un bind entre deux propriétés JavaFX : 6](#_Toc61971290)

[Je sais utiliser un fichier CSS pour styler mon application JavaFX : 6](#_Toc61971291)

[Je sais utiliser un formateur lors d’un bind entre deux propriétés JavaFX : 7](#_Toc61971292)

# Documentation

## Je sais concevoir un diagramme UML intégrant des notions de qualité et correspondant exactement à l'application que j'ai à développer :

Notre diagramme de classe se situe dans le répertoire documentation/DiagrammeDeClasse.mdj.

## Je sais décrire un diagramme UML en mettant en valeur et en justifier les éléments essentiels :

La description de notre diagramme de classe se trouve dans le fichier documentation/Projet\_JAVAFX.pdf à la page 3 et 4.

## Je sais documenter mon code et en générer la documentation :

La javadoc se situe dans le répertoire documentation/javadoc

## Je sais décrire le contexte de mon application, pour que n'importe qui soit capable de comprendre à quoi elle sert :

Le contexte de notre application se trouve dans le fichier documentation/Projet\_JAVAFX.pdf à la page 3.

## Je sais faire un diagramme de cas d'utilisation pour mettre en avant les différentes fonctionnalités de mon application :

Notre diagramme de cas d'utilisation se situe dans le répertoire documentation/DiagrammeCasUtilisation.mdj.

# Code

## Je maîtrise les règles de nommage Java :

-> Tout nos packages sont écrits en minuscules

-> Toutes nos classes on leurs première lettre en majuscule

-> Les variables ont leurs premières lettre en minuscule

Tout cela montre que nous maitrisons les règles de nommage en Java.

## Je sais binder bidirectionnellement deux propriétés JavaFX.:

Nous avons bindé bidirectionnellement le pseudo du joueur et le nom des entités.

## Je sais binder unidirectionnellement deux propriétés JavaFX.:

Nous avons bindé unidirectionnellement les position x, y et z des entités.

## Je sais coder une classe Java en respectant des contraintes de qualité de lecture de code :

-> Nous avons documenté et commenté notre code

-> Nous mettons les variables en premier puis les méthodes ensuite

Cela montre que nous savons coder un classe en respectant des contraintes de qualité de lecture de code.

## Je sais contraindre les éléments de ma vue, avec du binding FXML :

Dans la plupart de nos contrôleur de vues, nous avons des éléments bindé @FXML. Exemple pour la vue accueilJeu, nous avec le TextField pseudo.

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

Dans notre classe Monde, nous utilisons pour lesEntites une FXCollections.observableArrayList().

## Je sais développer une application graphique en JavaFX en utilisant FXML :

Notre application utilise 4 vue FXML.

## Je sais éviter la duplication de code :

Dans le controlleurJeu, notre fonction initialize utilise deux fois updatePanneau pour empêcher une duplication de code.

## Je sais hiérarchiser mes classes pour spécialiser leur comportement :

Nous avons différences classes hiérarchisées : entites, puis deplaceurs, puis createurs, puis collisionneur, puis les controlleurs et le boucleur.

## Je sais intercepter des évènements en provenance de la fenêtre JavaFX :

Dans notre controlleurAccueilJeu, nous avons 3 addEventFilter avec des KeyEvent permettant de déclencher une fonction en pressant une touche dans la nouvelle vue créée.

## Je sais maintenir, dans un projet, une responsabilité unique pour chacune de mes classes :

Chaque classe a une utilité précise et il n'y a pas une classe servant à faire la même chose qu'une autre.

## Je sais gérer la persistance de mon modèle :

Nous utilisons la sérialisation pour sauvegarder la position x et y de nos entités.

## Je sais surveiller l’élément sélectionné dans un composant affichant un ensemble de données :

Dans notre jeu, quand nous interagissons avec un objet / un prof, nous affichons le nom et le message de l'objet ou du prof.

## Je sais utiliser à mon avantage le polymorphisme :

Nous avons une classe entite et plusieurs classe héritant d'entite. Avec chacune des variables personnelles.

## Je sais utiliser certains composants simples que me propose JavaFX :

Nous utilisons la méthode start et stop pour démarrer et arrêter l'application dans le Launcher. Nous utilisons aussi des labels et toutes sortes de choses.

## Je sais utiliser certains layout que me propose JavaFX :

Nous utilisons dans notre vue jeu.fxml un Pane, dans notre ControlleurJeu des VBox, etc.

## Je sais utiliser GIT pour travailler avec mon binôme sur le projet :

Nous avons uniquement utilisé GIT avec des branches de créé, des merges, etc.

## Je sais utiliser le type statique adéquat pour mes attributs ou variables :

## Je sais utiliser les collections :

Nous utilisons pas exemple, pour nos quetes, des ArrayList.

## Je sais utiliser les différents composants complexes (listes, combo...) que me propose JavaFX :

Nous utilisons de nombreuses ArrayList comme par exemple pour les quêtes.

## Je sais utiliser les lambda-expression :

Dans notre controlleurjeu, nous utilisons dans le initialize une lambda-expression.

## Je sais utiliser les listes observables de JavaFX :

Dans notre classe Monde, nous utilisons une ObservableList.

## Je sais utiliser un convertisseur lors d’un bind entre deux propriétés JavaFX :

Pour récupérer le pseudo de notre joueur sur la page jeu, nous utilisons un convertisseur.

## Je sais utiliser un fichier CSS pour styler mon application JavaFX :

Nous utilisons un fichier css dans nos recourses/css/tout.css que nous utilisons par exemple dans notre vue accueilJeu.fxml.

## Je sais utiliser un formateur lors d’un bind entre deux propriétés JavaFX :

Nous n'avons pas utilisé de formateur lors de nos bind.