

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

Voir fichier « Documentation\diagrammeClasse.svg »

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

Voir fichier « Documentation\description\_diagrammes.pdf »

Je sais documenter mon code et en générer la documentation.

Voir dossier « Documentation\javadoc " ainsi que le code en général

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

Voir fichier « Documentation\ contexte.pdf »

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

Voir fichier « Documentation\UseCaseDiagram.svg »

Je maîtrise les règles de nommage Java.

Voir code en général

Je sais binder bidirectionnellement deux propriétés JavaFX.

```
pauseButton.selectedProperty().bindBidirectional(isPauseProperty);
```

ligne 106 Code\src\view\GameView.java

Je sais binder unidirectionnellement deux propriétés JavaFX.

```
gamePaused.bind(gamePausedProperty);  
stopThread.bind(stopThreadProperty);
```

ligne 33/34 Code\src\model\utils\looper\LooperAbs.java

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

Voir code en général

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

Pas utilisé

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

```
scoresView.itemsProperty().bind(Launcher.scoreHistoryManager.getScoresProperty());
```

```
scoresView.setCellFactory(__ -> new ListCell<Score>() {  
    @Override  
    protected void updateItem(Score item, boolean empty) {  
        super.updateItem(item, empty);
```

```

        if (!empty) {
            var dateScore = item.getDate();
            Text date = new Text();
            date.setText(dateScore.getDayOfMonth() + "/" +
String.format("%02d", dateScore.getMonthValue()) + "/" +
String.valueOf(dateScore.getYear()).substring(2, 4) + " " +
dateScore.getHour() + ":" + dateScore.getMinute());
            date.setFill(Color.WHITESMOKE);

            Text score = new Text();
            score.setText(String.valueOf(item.getScore()));
            score.setFill(Color.WHITESMOKE);

            BorderPane pane = new BorderPane();
            pane.leftProperty().setValue(date);
            pane.rightProperty().setValue(score);
            setGraphic(pane);
        } else {
            setText("");
            setGraphic(null);
        }
    }
});

```

Ligne 46,70 Code\src\view\ScoreView.java

Je sais éviter la duplication de code.

Voir code en général

Je sais hiérarchiser mes classes pour spécialiser leur comportement.

Voir fichier « Documentation\diagrammeClasse.svg »

Je sais intercepter des évènements en provenance de la fenêtre JavaFX.

```

Controller:public void goToScore(ActionEvent actionEvent)
FXML:<Button fx:id="scoreButton" onAction="#goToScore"

```

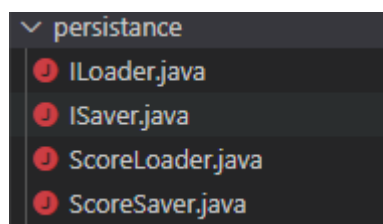
ligne 35 Code\ress\FXML\GameOver.fxml

ligne 37 Code\src\view\GameOver.java

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

Voir code en général

Je sais gérer la persistance de mon modèle.



L'ensemble de classe servant à la persistance

Je sais utiliser à mon avantage le polymorphisme.

Voir code en général

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

Voir gitlab : <https://gitlab.iut-clermont.uca.fr/yogourves/tetris>

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

```
private final ITetrominosRandomFactory tetrominosRandomFactory;
private GridAbs grid;
private ICollider collider;
private Tetrominos tetrominos;
private Tetrominos nextTetrominos;
private final Score score;
private ITetroRotator rotator;
private IMover mover;

public GameBoard(int lines, int columns) {
    grid = new Grid(lines, columns);
    tetrominosRandomFactory = new TetrominosRandomFactory();
    collider = new GridCollider(grid);
    nextTetrominos = tetrominosRandomFactory.create();
    score = new Score();
}
```

On utilise toujours le type le plus haut possible dans l'arborescence de classe

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

```
private final ObservableList<Score> scores;
private final ListProperty<Score> scoresProperty;
```

ligne 22/23 Code\src\model\utils\scoreHistory\ScoreHistoryManager.java

Je sais utiliser les lambda-expression.

```
gamePanel.setOnKeyPressed(keyEvent -> {
    if (isPauseProperty.getValue() == Boolean.FALSE &&
    isGameOverProperty.getValue() == Boolean.FALSE) {
        if (keyEvent.getCode() == KeyCode.LEFT || keyEvent.getCode()
    == KeyCode.Q) {
            for (var listener : eventListeners) {
                listener.onLeftEvent(EventSrc.USER);
            }
            keyEvent.consume();
        }
        if (keyEvent.getCode() == KeyCode.RIGHT || keyEvent.getCode()
    == KeyCode.D) {
            for (var listener : eventListeners) {
```

```

        listener.onRightEvent(EventSrc.USER);
    }
    keyEvent.consume();
}
if (keyEvent.getCode() == KeyCode.UP || keyEvent.getCode() ==
KeyCode.Z) {
    for (var listener : eventListeners) {
        listener.onRotateLeftEvent(EventSrc.USER);
    }
    keyEvent.consume();
}
if (keyEvent.getCode() == KeyCode.DOWN || keyEvent.getCode()
== KeyCode.S) {
    for (var listener : eventListeners) {
        listener.onDownEvent(EventSrc.USER);
    }
    keyEvent.consume();
}
}
if (keyEvent.getCode() == KeyCode.P) {
    pauseButton.selectedProperty().setValue(!pauseButton.selectedP
roperty().getValue());
}
if (keyEvent.getCode() == KeyCode.G) {
    newGame();
}

});

```

Ligne 71/105 Code\src\view\GameView.java

Je sais utiliser les listes observables de JavaFX.

```
private final ObservableList<Score> scores;
```

ligne 22 Code\src\model\utils\scoreHistory\ScoreHistoryManager.java

Je sais utiliser un convertisseur lors d'un bind entre deux propriétés JavaFX.

Pas utilisé

Je sais utiliser un fichier CSS pour styler mon application JavaFX.

Voir fichier Code\ress\FXML\style.css

Je sais utiliser un formateur lors d'un bind entre deux propriétés JavaFX.

Pas utilisé

Je sais développer un jeu en JavaFX en utilisant FXML.

Voir vidéo de présentation

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

```
while (!stopThread.getValue()) {  
    try {  
        Thread.sleep(200);  
        if (!gamePaused.getValue()) for (IObserver o : listObserver)  
            Platform.runLater(o::doAction);  
  
    } catch (InterruptedException e) {  
        e.printStackTrace();  
    }  
  
}
```

Boucle temporelle présente dans le fichier(ligne 23/33) Code\src\model\utils\looper\Looper.java