

# Stratego : nvs-04

nvs

16 juin 2022

## Stratego : nvs-04 (52075 & 54147)

### dépôt

`https://git.esi-bru.be/54147/dev4_stratego_54147_52075`

`git@git.esi-bru.be:54147/dev4_stratego_54147_52075.git`

### .gitignore

— ko : pas de fichier `.gitignore`, mais un fichier `gitignore...` le dossier `.git` fait 6,27 Mo.

### modélisation

#### remise

`tag / commit` ok.

`retard` (void)

`autre` (void)

#### analyse

retour fait oralement.

## console

remise

tag / commit ok.

## retard

- une version augmentée du rapport a été remise 5 jours après la date de remise 2.

autre (void)

## documentation

- les fichiers ne sont pas documentés, mais comme tout est plongé dans l'espace de nommage `stratego`, la documentation des fonctions et énumérations est bien générée.
- méthodes privées de `Controller` documentées à moitié, pas de documentation des attributs privés des autres classes.
- pas de documentation pour les classes (`class`).
- pas de documentation des méthodes publiques de `class FileReader`, ni de `std::ostream & operator<<(std::ostream & os, const Difficulty& dt);` et `std::ostream & operator<<(std::ostream & os, const Piece& dt);`.

fichier de configuration uniquement ok.

à l'ouverture de celui-ci avec doxywizard (doxygen 1.9.1), j'obtiens :

```
warning: ignoring unsupported tag 'SHOW_HEADERFILE' at line 609, file C:/.../nvs-04_52075_54147/Doxyfile
warning: ignoring unsupported tag 'WARN_IF_INCOMPLETE_DOC' at line 827, file C:/.../nvs-04_52075_54147/Doxyfile
warning: ignoring unsupported tag 'DOCSET_FEEDURL' at line 1408, file C:/.../nvs-04_52075_54147/Doxyfile
warning: ignoring unsupported tag 'FULL_SIDEBAR' at line 1619, file C:/.../nvs-04_52075_54147/Doxyfile
warning: ignoring unsupported tag 'OBFUSCATE_EMAILS' at line 1650, file C:/.../nvs-04_52075_54147/Doxyfile
warning: ignoring unsupported tag 'MATHJAX_VERSION' at line 1709, file C:/.../nvs-04_52075_54147/Doxyfile
warning: ignoring unsupported tag 'DIR_GRAPH_MAX_DEPTH' at line 2566, file C:/.../nvs-04_52075_54147/Doxyfile
```

mais ça compile la documentation html.

rapport

format pdf ok.

*rem.* : une version non recevable du rapport a été déposée dans le dépôt le 30 mars 2022, soit 5 jours après la date de remise 2.

**bogue non signalé** (void)

**écart / ajout non signalé** (void)

**autre** (void)

**rapport / code**

**avertissement restant**

**gcc** signalé

(void)

non signalé

utils/file\_reader.cpp: Dans la fonction membre « std::vector<std::\_\_cxx11::basic\_string<char> > stratego::FileReader::getBluePieces() const »:

utils/file\_reader.cpp:284:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

284 | }

| ^

utils/file\_reader.cpp: Dans la fonction membre « std::vector<std::\_\_cxx11::basic\_string<char> > stratego::FileReader::getRedPieces() const »:

utils/file\_reader.cpp:287:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

287 | }

| ^

utils/file\_reader.cpp: Dans la fonction membre « std::vector<stratego::Piece> stratego::FileReader::getListBluePiece() const »:

utils/file\_reader.cpp:291:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

291 | }

| ^

utils/file\_reader.cpp: Dans la fonction membre « std::vector<stratego::Piece> stratego::FileReader::getListRedPiece() const »:

utils/file\_reader.cpp:294:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

294 | }

| ^

view/view.cpp: Dans la fonction membre « virtual void stratego::View::update(const nvs::Subject\*) »:

view/view.cpp:178:39: attention: paramètre « subject » inutilisé [-Wunused-parameter]

```

178 | void View::update(const nvs::Subject *subject){
    | ~~~~~^~~~~~

```

ces avertissements ont été signalés dans le rapport remis en retard. ils sont à régler pour ce qui concerne les **return** manquants.

#### gcc + clang-analyzer

utils/file\_reader.cpp: Dans la fonction membre « std::vector<std::\_\_cxx11::basic\_string<char> > stratego::FileReader::getBluePieces() const »:

utils/file\_reader.cpp:284:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

```

284 | }
    | ^

```

utils/file\_reader.cpp: Dans la fonction membre « std::vector<std::\_\_cxx11::basic\_string<char> > stratego::FileReader::getRedPieces() const »:

utils/file\_reader.cpp:287:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

```

287 | }
    | ^

```

utils/file\_reader.cpp: Dans la fonction membre « std::vector<stratego::Piece> stratego::FileReader::getListBluePiece() const »:

utils/file\_reader.cpp:291:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

```

291 | }
    | ^

```

utils/file\_reader.cpp: Dans la fonction membre « std::vector<stratego::Piece> stratego::FileReader::getListRedPiece() const »:

utils/file\_reader.cpp:294:1: attention: « return » manquant dans une fonction devant retourner une valeur [-Wreturn-type]

```

294 | }
    | ^

```

view/view.cpp: Dans la fonction membre « virtual void stratego::View::update(const nvs::Subject\*) »:

view/view.cpp:178:39: attention: paramètre « subject » inutilisé [-Wunused-parameter]

```

178 | void View::update(const nvs::Subject *subject){
    | ~~~~~^~~~~~

```

c'est-à-dire idem gcc.

clang++ idem gcc.

clang++ + clang-analyzer idem gcc.

cppcheck signalé

(void)

non signalé

```
view/view.h: 22:5 [view/view.h:22]
style : noExplicitConstructor
      View(Stratego& model);
      ^
```

Class 'View' has a constructor with 1 argument that is not explicit.

à régler éventuellement

cet avertissement a été signalé dans le rapport remis en retard.

```
model/board.h: 37:10 [model/board.cpp:25] -> [model/board.h:37]
performance (inconclusive) : functionStatic
      bool isInside(const Position& pos);
      ^
```

Technically the member function 'stratego::Board::isInside' can be static (but you may consider moving to unnamed namespace).

à régler éventuellement

```
model/piece.h: 78:10 [model/piece.cpp:51] -> [model/piece.h:78]
style (inconclusive) : functionConst
      bool isStronger(const Piece& piece);
      ^
```

Technically the member function 'stratego::Piece::isStronger' can be const.

à régler

```
model/piece.cpp: 55:57 [model/piece.h:81] -> [model/piece.cpp:55]
style (inconclusive) : funcArgNamesDifferent
std::ostream& operator<<(std::ostream& os, const Piece& p){
      ^
```

Function 'operator<<' argument 2 names different: declaration 'dt' definition 'p'.

à régler éventuellement

```
model/position.h: 51:10 [model/position.cpp:25] -> [model/position.h:51]
style (inconclusive) : functionConst
```

```
bool operator==(const Position & pos);  
    ^
```

Technically the member function 'stratego::Position::operator==' can be const.

à régler

```
model/square.cpp: 42:58 [model/square.h:73] -> [model/square.cpp:42]  
style (inconclusive) : funcArgNamesDifferent  
std::ostream& operator<<(std::ostream& os, const Square& sq){  
    ^
```

Function 'operator<<' argument 2 names different: declaration 'dt' definition 'sq'.

à régler éventuellement

```
model/stratego.cpp: 5:11 [model/stratego.cpp:5]  
warning : uninitMemberVar  
Stratego::Stratego() : state_{State::NOT_STARTED},currentPlayer_{0}{  
    ^
```

Member variable 'Stratego::difficulty\_' is not initialized in the constructor.

à régler

cet avertissement a été signalé dans le rapport remis en retard.

```
model/stratego.cpp: 25:13 [model/stratego.cpp:25]  
style : unreachableCode  
    notifyObservers();  
    ^
```

Statements following return, break, continue, goto or throw will never be executed.

à régler

cet avertissement a été signalé dans le rapport remis en retard, mais pas correctement (stratego.h à la place de stratego.cpp).

```
model/stratego.cpp: 356:71 [model/stratego.h:138] -> [model/stratego.cpp:356]  
style (inconclusive) : funcArgNamesDifferent  
bool Stratego::isMoveScoutExact(const Position& start,const Position& end){  
    ^
```

Function 'isMoveScoutExact' argument 2 names different: declaration 'posEnd' definition 'end'.

à régler éventuellement

```
model/stratego.cpp: 166:24 [model/stratego.cpp:166]
style : unreadVariable
    Square squareStart = this->board_.getSquare(this->selected_);
    ~
```

Variable 'squareStart' is assigned a value that is never used.

à régler éventuellement

cet avertissement a été signalé dans le rapport remis en retard.

```
utils/file_reader.cpp: 284:1 [utils/file_reader.cpp:284]
error : missingReturn
}
~
```

Found a exit path from function with non-void return type that has missing return statement

à régler

cet avertissement a été signalé dans le rapport remis en retard.

```
utils/file_reader.cpp: 287:1 [utils/file_reader.cpp:287]
error : missingReturn
}
~
```

Found a exit path from function with non-void return type that has missing return statement

à régler

cet avertissement a été signalé dans le rapport remis en retard.

```
utils/file_reader.cpp: 291:1 [utils/file_reader.cpp:291]
error : missingReturn
}
~
```

Found a exit path from function with non-void return type that has missing return statement

à régler

cet avertissement a été signalé dans le rapport remis en retard.

```
utils/file_reader.cpp: 294:1 [utils/file_reader.cpp:294]
error : missingReturn
}
~
```

Found a exit path from function with non-void return type that has missing return statement

à régler

cet avertissement a été signalé dans le rapport remis en retard.

```
utils/file_reader.h: 25:10 [utils/file_reader.cpp:277] -> [utils/file_reader.h:25]
performance (inconclusive) : functionStatic
    void readFilePiece();
    ~
```

Technically the member function 'stratego::FileReader::readFilePiece' can be static (but you may consider moving to unnamed namespace).

à régler éventuellement

```
utils/file_reader.h: 27:30 [utils/file_reader.cpp:282] -> [utils/file_reader.h:27]
performance (inconclusive) : functionStatic
    std::vector<std::string> getBluePieces() const;
    ~
```

Technically the member function 'stratego::FileReader::getBluePieces' can be static (but you may consider moving to unnamed namespace).

à régler éventuellement

```
utils/file_reader.h: 28:30 [utils/file_reader.cpp:285] -> [utils/file_reader.h:28]
performance (inconclusive) : functionStatic
    std::vector<std::string> getRedPieces() const;
    ~
```

Technically the member function 'stratego::FileReader::getRedPieces' can be static (but you may consider moving to unnamed namespace).

à régler éventuellement

```
utils/file_reader.h: 30:24 [utils/file_reader.cpp:289] -> [utils/file_reader.h:30]
performance (inconclusive) : functionStatic
    std::vector<Piece> getListBluePiece() const;
    ~
```

Technically the member function 'stratego::FileReader::getListBluePiece' can be static (but you may consider moving to unnamed namespace)



à régler éventuellement

```
utils/file_reader.h: 31:24 [utils/file_reader.cpp:292] -> [utils/file_reader.h:31]
performance (inconclusive) : functionStatic
    std::vector<Piece> getListRedPiece() const;
    ^
```

Technically the member function 'stratego::FileReader::getListRedPiece' can be static (but you may consider moving to unnamed namespace).

à régler éventuellement

```
utils/file_reader.cpp: 19:9 [utils/file_reader.cpp:19]
style : variableScope
    int countRed = 0;
    ^
```

The scope of the variable 'countRed' can be reduced.

à régler éventuellement

cet avertissement a été signalé dans le rapport remis en retard.

```
utils/file_reader.cpp: 20:9 [utils/file_reader.cpp:20]
style : variableScope
    int countBlue = 0;
    ^
```

The scope of the variable 'countBlue' can be reduced.

à régler éventuellement

cet avertissement a été signalé dans le rapport remis en retard.

```
view/view.cpp: 61:45 [view/view.cpp:61]
performance : passedByValue
void View::askPlacementPosition(std::string piece ,Color color){
    ^
```

Function parameter 'piece' should be passed by const reference.

à régler

cet avertissement a été signalé dans le rapport remis en retard.

```
view/view.cpp: 61:0 [view/view.cpp:61]
```

style : unusedFunction

~

The function 'askPlacementPosition' is never used.

à régler éventuellement

utils/file\_reader.cpp: 282:0 [utils/file\_reader.cpp:282]

style : unusedFunction

~

The function 'getBluePieces' is never used.

à régler éventuellement

model/stratego.cpp: 344:0 [model/stratego.cpp:344]

style : unusedFunction

~

The function 'getInteractiveBlue' is never used.

à régler éventuellement

model/stratego.cpp: 347:0 [model/stratego.cpp:347]

style : unusedFunction

~

The function 'getInteractiveRed' is never used.

à régler éventuellement

utils/file\_reader.cpp: 289:0 [utils/file\_reader.cpp:289]

style : unusedFunction

~

The function 'getListBluePiece' is never used.

à régler éventuellement

utils/file\_reader.cpp: 292:0 [utils/file\_reader.cpp:292]

```
style : unusedFunction
~
The function 'getListRedPiece' is never used.
à régler éventuellement
utils/file_reader.cpp: 285:0 [utils/file_reader.cpp:285]
style : unusedFunction
~
The function 'getRedPieces' is never used.
à régler éventuellement
model/stratego.cpp: 276:0 [model/stratego.cpp:276]
style : unusedFunction
~
The function 'isFullCorrectly' is never used.
à régler éventuellement
utils/subject.cpp: 11:0 [utils/subject.cpp:11]
style : unusedFunction
~
The function 'unregisterObserver' is never used.
à régler éventuellement
nofile: 0:0
information : missingInclude

Cppcheck cannot find all the include files (use --check-config for details)
?
```

code source

portabilité

casse noms fichiers ok.

séparateur / ok.

c++ standard ok.

si pas std : portabilité (void)

bonnes pratiques

déclarations anticipées si possible

```
— dans difficulty.h, piece.h et player.h :
    // #include <iostream>    // rnvs : include / déclaration anticipée
    #include <ostream>        // rnvs : include / déclaration anticipée
— dans position.h :
    // #include <string>        // rnvs : include / déclaration anticipée
    // #include <ostream>        // rnvs : include / déclaration anticipée
— dans board.h :
    // #include "position.h"    // rnvs : include / déclaration anticipée
    namespace stratego
    {

        class Position;          // rnvs : include / déclaration anticipée
— dans stratego.h :
    #include <array>            // rnvs : include / déclaration anticipée
— dans file_reader.h :
    #include <string>
    // #include <fstream>        // rnvs : include / déclaration anticipée
    #include <vector>
    // #include "model/stratego.h" // rnvs : include / déclaration anticipée
```

```

#include "model/piece.h"          // rnvs : include / déclaration anticipée
namespace stratego
{

class Stratego;                  // rnvs : include / déclaration anticipée
— dans view.h :
#include "utils/observer.h"
// #include "model/stratego.h"    // rnvs : include / déclaration anticipée
#include <string>                  // rnvs : include / déclaration anticipée

namespace stratego
{

enum class Color;                // rnvs : include / déclaration anticipée
class Stratego;                  // rnvs : include / déclaration anticipée

// class Subject;                // rnvs : include / déclaration anticipée
— dans controller.h :
// #include "model/stratego.h"    // rnvs : include / déclaration anticipée
// #include "view/view.h"         // rnvs : include / déclaration anticipée
// #include "utils/file_reader.h" // rnvs : include / déclaration anticipée
// #include "utils/keyboard.hpp"  // rnvs : include / déclaration anticipée
// #include "utils/string_convert.hpp" // rnvs : include / déclaration anticipée

namespace stratego
{

class Stratego;    // rnvs : include / déclaration anticipée
class View;        // rnvs : include / déclaration anticipée
class Position;    // rnvs : include / déclaration anticipée
enum class Difficulty; // rnvs : include / déclaration anticipée

```

using namespace dans .h ok.

**autre** (void)

**gestion de la mémoire** ok : pas de `new` dans les classes métier ni celles de l'application console.

**tests unitaires** (ceci n'est pas demandé)

(void)

**classes métier**

**initialisation** plateau

ok : plateau de jeu construit vide avec les pièces d'eau bien positionnées.

interactive

ko :

- `Controller::interactivePlacement()` semble s'en occuper mais `Stratego::interactivePlacement()` ne place aucun pion sur le plateau de jeu. ceci est signalé dans le rapport.

fichiers

plutôt ko :

- `FileReader::readFilePiece(Stratego & game)` s'en charge : pas ok car c'est `Stratego` qui devrait plutôt le faire... d'autant plus que le placement se fait avec `Stratego::addPiece(const Position & pos, Piece piece)` qui peut être appelé pendant la partie => possibilité de tricher...

validation du nombre de pièces de chaque type

- interactif : ko.
- fichiers : ok dans `FileReader::readFilePiece(Stratego & game)` mais code avec beaucoup de duplication et de variables, moche et non modulaire.

possibilité de mode débutant

ok : `Stratego::startStratego(Difficulty difficulty)` s'en occupe avant le début de la bataille.

joueurs (éventuellement)

il y a une classe **Player**. elle enveloppe juste une `enum class Color` et n'apporte aucune plus value par rapport à cette énumération.

**déplacement** obligatoire

ko : `Stratego::turnPlayed()` est public et ne vérifie pas qu'un déplacement ou combat a eu lieu : on peut donc sélectionner une pièce puis passer son tour, sans se déplacer.

pièces d'eau inaccessibles

ok : `Stratego::isSelectionExact(const Position & pos)` s'en occupe.

éclaireur

latéral uniquement

ok : voir `Stratego::movePieceScout(const Position & pos)`.

illimité sauf obstacle

ok : voir `Stratego::movePieceScout(const Position & pos)`.

bombes et drapeau immobiles

ok : `Stratego::isSelectionExact(const Position & pos)` s'en occupe.

autres pièces

latéral uniquement

ko :

— `Stratego::movePieceNormal(const Position & pos)` vérifie si déplacement de 1 latéral mais pas exclusivement => diagonale possible.

une seule position

ok : voir `Stratego::movePieceNormal(const Position & pos)`.

limite des 3 allers - retours

— ko : pas implémenté.

**combat**    détection

ok : dans `Stratego::movePieceNormal(const Position & pos)` et `Stratego::movePieceScout(const Position & pos)`.

résolution

ok : voir `Stratego::attack(const Position & pos)`.

cas de l'espionne

ok : voir `Stratego::attack(const Position & pos)`.

**fin de partie**    ko :

— la méthode `Stratego::isFinish()` n'est invoquée par aucune méthode publique de `Stratego`. c'est le contrôleur qui s'en charge.

par prise de drapeau

ok : voir `Stratego::isFinish()`.

par impossibilité de déplacement

ok : voir `Stratego::isFinish()`.

ko :

— dans `Stratego::isFinish()` on vérifie s'il reste des pions déplaçables *en théorie*, mais il n'y a pas de vérification si effectivement ils sont déplaçables.

**méthodes**    complètes : 1 méthode / 1 action de jeu

semble ok : le déplacement se fait en 2 méthodes distinctes : `Stratego::selectPiece(const Position & pos)` puis `Stratego::movePiece(const Position & pos)` qui invoque `Stratego::movePieceScout(const Position & pos)` ou `Stratego::movePieceNormal(const Position & pos)` qui vérifient bien que le jeu est dans l'état `State::PIECE_SELECTED`.

mais en fait ko :

— la détection de fin de jeu ne se fait pas automatiquement après / avant un déplacement.

impossibilité de tricher (bibliothèque)

ko :



- il y a une `enum class State`, mais elle ne verrouille pas tout : par exemple il n'y a pas d'état pour indiquer si le plateau n'a pas ou a été construit.
- `Stratego::startStratego(Difficulty difficulty)` ne vérifie pas si le plateau est ok.
- `Stratego::addPiece(const Position & pos, Piece piece)` est public et peut être appelé si `state_ == State::PIECE_SELECTED` mais ne vérifie rien d'autre.
- `Stratego::selectPiece(const Position & pos)` ne vérifie pas l'état du jeu : on peut l'appeler n'importe quand, même quand le plateau n'a pas été construit.
- `Stratego::turnPlayed()` est public et ne vérifie pas qu'un déplacement ou combat a eu lieu : on peut donc sélectionner une pièce puis passer son tour, sans se déplacer.

### **contrôleur**

**fiabilisation lectures clavier** ko : les lignes vont de 0 à 9 et non de 1 à 10 comme demandé.

**respect de la convention d'identification des cases** ok.

légende

ok.

### **convivialité**

- il semble possible de désélectionner la pièce fournie, mais j'ai beau dire 0, cela ne fonctionne pas.
- détection de sélection d'une pièce non déplaçable : ok.

**alternance des joueurs** plateau caché entre les joueurs

pas mis en œuvre.

information si un combat a eu lieu

pas implémenté : indiqué dans le rapport.

### **vue**

**design pattern observer** ok.

**absence de flux (cout) dans classes métier** ko! `std::cout` dans `+ Stratego::switchPlayer() + FileReader::readFilePiece(Stratego& game)`

**affichage** masqué en mode normal

bug non signalé en mode normal :

- pas d’affichage du plateau après sa construction car `break` avant notification dans `Stratego::startStratego(Difficulty difficulty)`.

éventuellement non masqué en mode débutant

ko :

- les pièces adverses sont bien visibles... mais rien n’indique que ce sont les pièces de l’adversaire : seule la force des pièces est indiquée, pas leur propriétaire.

cimetière

il y a un attribut `deadPieces_` dans `Stratego`, mais pas de *getter*.

**autre** (void)

**gui**

**remise**

**tag / commit**

- je ne vois pas le *tag* `gui` dans le dépôt utilisé pour les 2 remises précédentes. si je reclone, je le vois...

**retard** (void)

**autre** (void)

**documentation**

- il y a un fichier pour Doxygen. quand je l’ouvre j’obtiens :

```
warning: ignoring unsupported tag 'TCL_SUBST' at line 248, file C:/.../nvs-04_52075_54147_gui/GUI/Stratego_Gui/Doxyfile
warning: ignoring unsupported tag 'COLS_IN_ALPHA_INDEX' at line 1090, file C:/.../nvs-04_52075_54147_gui/GUI/Stratego_Gui/Doxyfile
warning: ignoring unsupported tag 'PERL_PATH' at line 2163, file C:/.../nvs-04_52075_54147_gui/GUI/Stratego_Gui/Doxyfile
warning: ignoring unsupported tag 'MSCGEN_PATH' at line 2185, file C:/.../nvs-04_52075_54147_gui/GUI/Stratego_Gui/Doxyfile
mais ça compile sans souci.
```

— fichiers et classes graphiques pas documentés, mais bien les méthodes publiques et privées de celles-ci, sauf `View::messageSelection()`.

**rapport**

**format pdf** ok.

**bogue non signalé**

— il faut changer le nom du dossier en `Stratego_FinalV` pour que les ressources (fond de plateau, eau) soient bien chargées.

**écart / ajout non signalé** (void)

**autre** (void)

**code source**

**portabilité**

**casse noms fichiers** ok.

**séparateur /** ok.

**c++ standard + qt** ok.

**si pas std + qt : portabilité** (void)

**gestion de la mémoire** ok.

**contrôleur**

**respect des règles** ok.

**convivialité**

- placement interactif propre et facile : ok.
- choix des fichiers avec `QFileDialog` : ok.
- pour désélectionner un pion : réaliser un mouvement interdit.
- sélection origine / destination et ça marche : ok.

**vue**

**design pattern observer** ok.

**convivialité**

- image et valeurs des pions : ok.
- pas de rapport de combat, mais cimetière avec dernière pièce prise : on connaît donc l'identité du (ou des) perdant(s), pas du vainqueur.
- pas de masquage du plateau entre les 2 joueurs.

**autre**

- menu d'aide.
- menu *nouvelle partie*.

**examen**

retours faits le jour même.