

|  |
| --- |
| Bataille Navale |



*Tiago Santos*

*SI-C1B*

*Tiago.santos@cpnv.ch*

*Date de création : 13.03.2019*



**Table des matières**

[1 Introduction 3](#_Toc5541591)

[1.1 Cadre, description et motivation 3](#_Toc5541592)

[1.2 Organisation 3](#_Toc5541593)

[1.3 Objectifs 3](#_Toc5541594)

[1.4 Planification initiale 3](#_Toc5541595)

[2 Analyse 4](#_Toc5541596)

[2.1 Use cases et scénarios 4](#_Toc5541597)

[Stratégie de test 6](#_Toc5541599)

[3 Implémentation 6](#_Toc5541600)

[3.1 Points techniques spécifiques 6](#_Toc5541601)

[3.1.1 Point 1 6](#_Toc5541602)

[3.1.2 Point 2 6](#_Toc5541603)

[3.1.3 Point … 6](#_Toc5541604)

[3.2 Livraisons 7](#_Toc5541605)

[4 Tests 7](#_Toc5541606)

[4.1 Tests effectués 7](#_Toc5541607)

[4.2 Erreurs restantes 7](#_Toc5541608)

[5 Conclusions 7](#_Toc5541609)

[6 Annexes 7](#_Toc5541610)

[6.1 Sources – Bibliographie 7](#_Toc5541611)

[6.2 Journal de bord du projet 7](#_Toc5541612)

# Introduction

## Cadre, description et motivation

La bataille navale est un projet qui met ensemble deux modules (ICT-431 et MA-20), le ICT-431 consiste à apprendre à s’organiser dans la réalisation d’un projet et le MA-20 consiste à faire une bataille navale où on devrait être capable d’afficher une grille de bataille navale, placer des bateaux et pouvoir jouer.

## Organisation

Organisation générale du projet :

Tiago Santos, [tiago.santos@cpnv.ch](mailto:tiago.santos@cpnv.ch), 0764361002

Responsable de projet :

M. Benzonana, [Pascal.BENZONANA@cpnv.ch](mailto:Pascal.BENZONANA@cpnv.ch) (ICT-431 et MA-20)

Experts :

M. Benzonana, [Pascal.BENZONANA@cpnv.ch](mailto:Pascal.BENZONANA@cpnv.ch) (ICT-431, MA-20 et Maître de classe de la SI-C1B)

M. Carrel, [Xavier.CARREL@cpnv.ch](mailto:Xavier.CARREL@cpnv.ch) (ICT-431, MA-20 et Maître de classe de la SI-C1A)

## Objectifs

Dans le menu, il doit avoir une option « Jouer », « Quitter », « Aide » et

« Score »

L’utilisateur doit pouvoir jouer sur plusieurs cartes et poser les bateaux.

L’utilisateur doit pouvoir tirer et recevoir un message pour dire s’il a touché, loupé ou si le bateau à couler.

## Planification initiale

Date de début : 06.03.2019 Date de fin : 07.04.2019

Étapes principales : Construction des grilles de jeu et rendre le programme jouable.

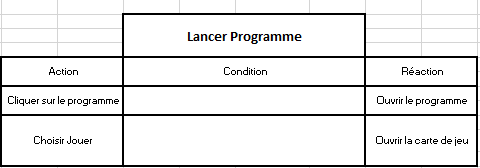
# Analyse

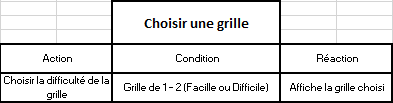
Dans ma bataille vous allez pouvoir choisir un niveau de difficulté et le programme s’adaptera au niveau de difficulté choisi.

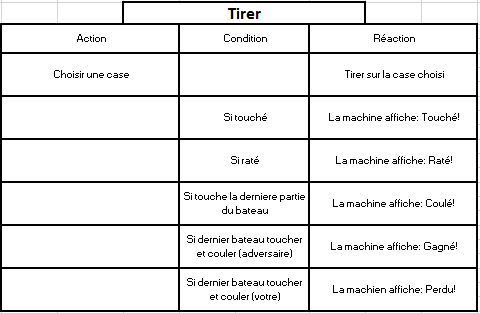
## Use cases et scénarios

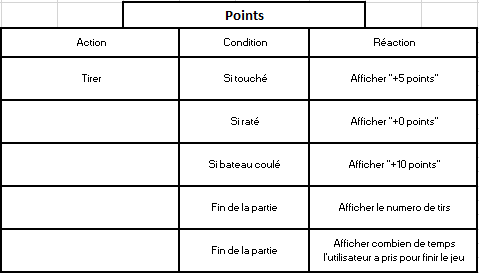
### 



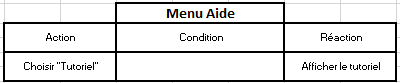








### 



## Stratégie de test :

Test 1 : Choisir « 1 – Jouer » après « 1 – Partie rapide (Sans Score) » et le programme affiche une grille de 10 sur 10 prêt à jouer.

Test 2 : Choisir « 1 – Jouer » après « 2 – Choisir la grille de jeu (Avec Score) vous aurez 2 grilles à choisir « 1 – Facile » « 2 – Difficile »

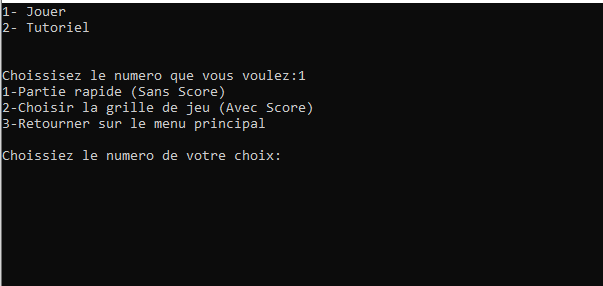
Test 3 : Choisir « 1 – Jouer » après « 2 – Choisir la grille de jeu (Avec Score) vous aurez 2 grilles à choisir « 1 – Facile » « 2 – Difficile » pour tirer « Ou veux-tu tirer en vertical ? » vous devriez choisir un numéro entre 1 et 10 et après quand ce message ce affiche « Ou veux-tu tirer en horizontal ? »

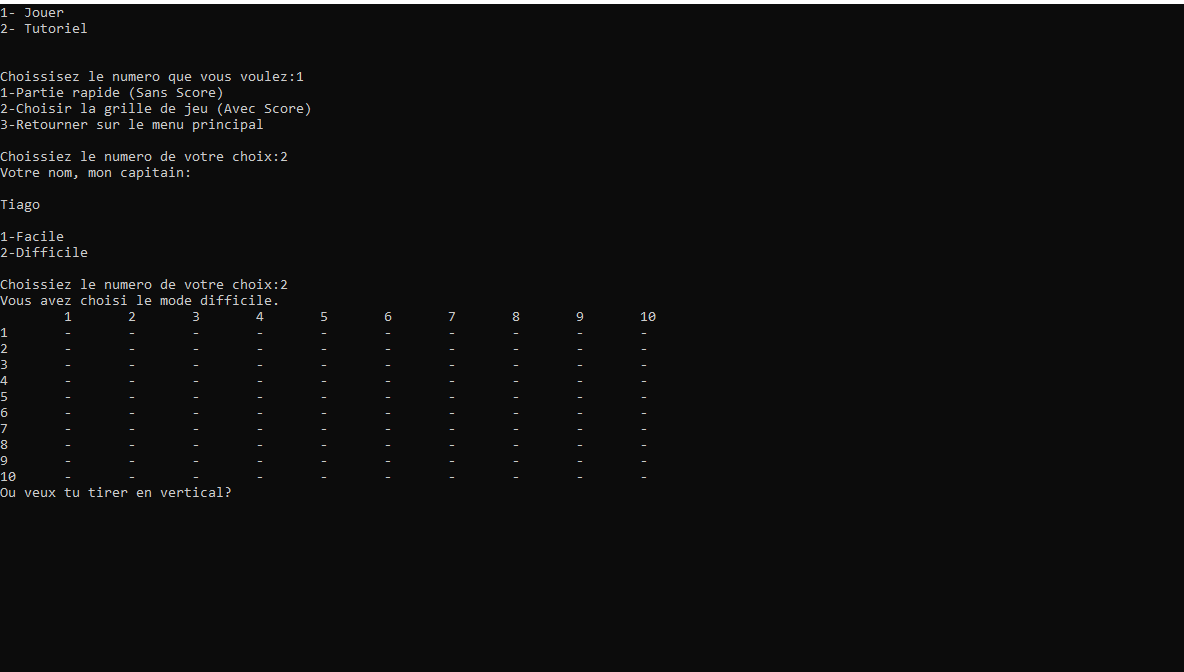
Vous devriez choisir un numéro entre 1 et 10 pour pouvoir tirer. Le programme affichera « T » si touché ou « R » si loupé.

Test 4 : Ouvrir le programme choisir « 2 – Tutoriel » et une page de tutoriel s’ouvre pour vous expliquer le comment jouer.

## Maquettes







# Implémentation

## Points techniques spécifiques

### Pseudo-Code pour afficher « T » :

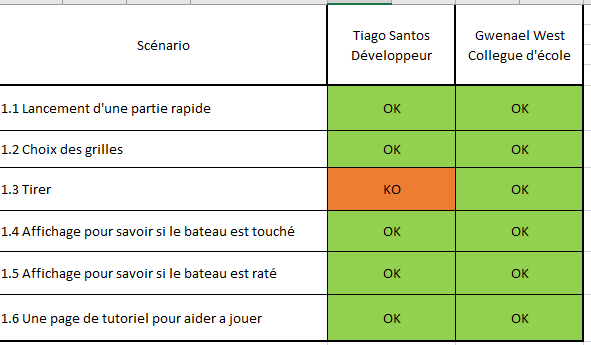
Quand lire réponse d’utilisateur pour le tire en horizontal et lire réponse d’utilisateur pour le tire en vertical est égal aux emplacement des bateaux afficher « T ».

### Pseudo-Code pour afficher « R » :

Quand lire réponse d’utilisateur pour le tire en horizontal et lire réponse d’utilisateur pour le tire en vertical n’est pas égal aux emplacement des bateaux afficher « R ».

# Tests

## Tests effectués



## Erreurs restantes

Je n’ai pas réussi à faire un système de score ni a faire le programme afficher « C » quand le bateau est coulé.

# Conclusions

## Objectifs

### Objectifs atteints :

Affichage des 2 grilles (Facile et Diffcile)

Afficher T si bateau touché.

Afficher R si bateau raté.

Un page de tutoriel.

### Objectifs non atteints :

Afficher C quand un bateau est coulé.

Un système pour le score.

Une grille crée aléatoirement.

## Points Positives et Négatives :

### Positives :

* Première réalisation d’un petit jeu.
* Plusieurs choses appris sur CLion que je ne connaissais pas.

### Négatives :

* Je n’ai pas réussi à finir mon jeu et a remplir le cahier des charges.
* Certaines difficultés que je ne m’attentais pas.

## Améliorations possible sur le projet :

* Implanter un système pour garder et afficher les scores.
* Faire une IA pour faire des duels 1v1.
* Faire une grille aléatoire.

# Annexes

## Sources – Bibliographie

<https://c.developpez.com/cours/20-heures/>

M. Benzonana

## Journal de bord du projet

