

|  |
| --- |
| Bataille Navale |



Eaaswaralingam, Kaarththigan

Kaarththigan.EAASWARALINGAM@cpnv.ch



SI-MI1b

13.03.2019

Table des matières

[1 Introduction 3](#_Toc5570732)

[1.1 Cadre, description et motivation 3](#_Toc5570733)

[1.2 Organisation 3](#_Toc5570734)

[1.3 Objectifs 3](#_Toc5570735)

[1.4 Planification initiale 3](#_Toc5570736)

[2 Analyse 4](#_Toc5570737)

[2.1.1 (Use case 1) 5](#_Toc5570738)

[2.1.2 (Use case 2) 5](#_Toc5570739)

[2.1.3 (Use case 3) 5](#_Toc5570740)

[2.2 Stratégie de test 5](#_Toc5570741)

[2.3 Budget 6](#_Toc5570742)

[3 Implémentation 6](#_Toc5570743)

[3.1 Vue d’ensemble 6](#_Toc5570744)

[3.2 Choix techniques 6](#_Toc5570745)

[3.3 Modèle Logique de données 6](#_Toc5570746)

[3.4 Points techniques spécifiques 8](#_Toc5570747)

[3.4.1 Point 1 : Choisir un menu 8](#_Toc5570748)

[3.4.2 Point 1 : Saisir un numéro valable sinon le programme va demander de choisir une case en boucle. 8](#_Toc5570749)

[3.5 Livraisons 8](#_Toc5570750)

[4 Tests 8](#_Toc5570751)

[4.1 Tests effectués 8](#_Toc5570752)

[4.2 Erreurs restantes 8](#_Toc5570753)

[5 Conclusions 8](#_Toc5570754)

[6 Annexes 10](#_Toc5570755)

[6.1 Sources – Bibliographie 10](#_Toc5570756)

[6.2 Journal de bord du projet 10](#_Toc5570757)

# Introduction

## Cadre, description et motivation

**Motivation** : Cela va me préparer à créer des jeux beaucoup plus gros et beaucoup plus ambitieux par la suite et aussi faire des recherches par moi-même sur internet sur diverses fonctionnalités du « C ».

**Description :** Reproduire le jeu « Bataille Navale » en programmation « C » et insérer toutes les fonctionnalités du jeu.

## Organisation

Élève : Eaaswaralingam Kaarththigan

Responsable du projet : Pascal Benzonana

## Objectifs

* Le but de ce projet est de réaliser un jeu ressemblant à la bataille navale qui devra avoir TOUTES les fonctionnalités puis devoir le finir le 05.04.2019.
* Commencer par les uses cases scénarios, ensuite github, et enfin la maquette. Et quand tout ceux-là sera finit on commence par programmer le menu du jeu, ensuite la grille sur laquelle vont devoir jouer les joueurs. Placer des bateaux sur une griller fixe. Créer une fonction qui va pouvoir, d’abord, regarder la grille (voir à quoi elle ressemble) ensuite créer la fonction pour pouvoir jouer. Et ensuite, pour finir, on créer les fonctions : règle, options, quitter, score. Et finir ceux-là le 05.04.2019.
* Pouvoir finir la documentation avec les uses case, les scénarios et github le lundi 01.04.2019.

## Planification initiale

Date de début : 11.03.2019

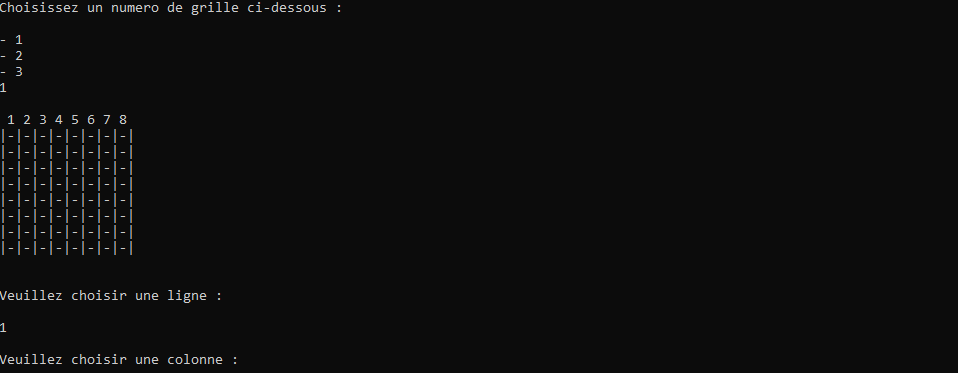
Date de fin : 07.04.2019

But : Finir la documentation ainsi que le programme



# Analyse

Maquette :

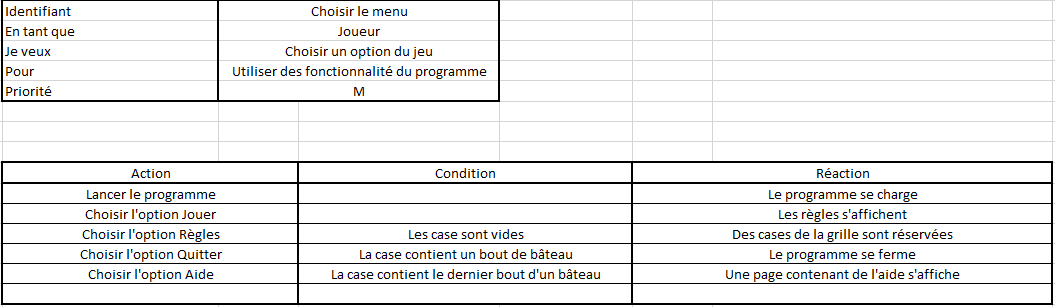


## 

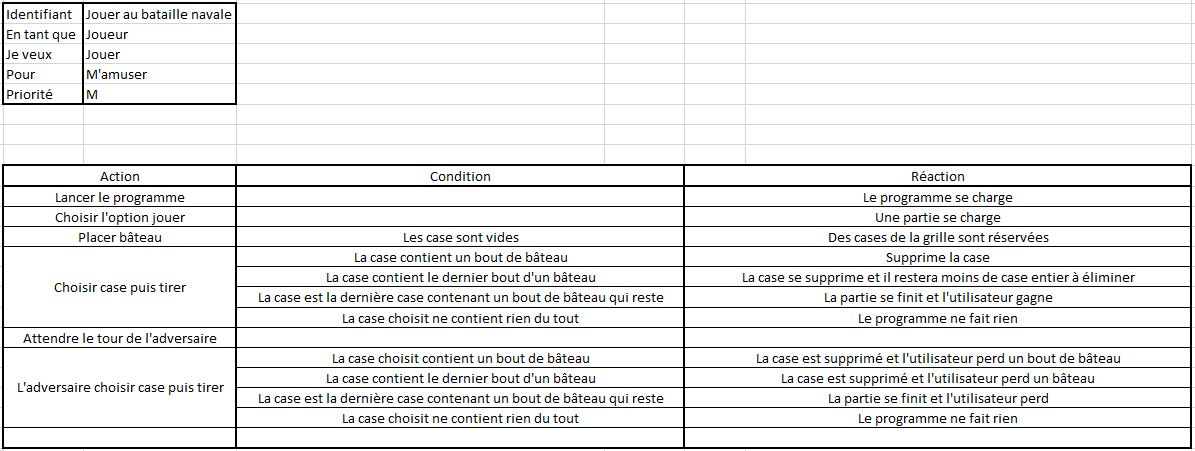
### (Use case 1)

#### 

### (Use case 2)



### (Use case 3)



## Stratégie de test

* Tester le lancement du programme.
* Tester les différents caractères sur notre clavier quand on peut insérer quelque chose.
* Vérifier les différentes fonctions.

## Budget

* On a un temps de huit semaines pour pouvoir finir le projet.
* Le programme doit au moins avoir les fonctions jouer ainsi que le score fonctionnel.
* On dispose de Clion ou CodeBlocks pour pouvoir effectuer ce projet.

# Implémentation

## Vue d’ensemble

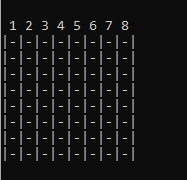
## Choix techniques

Ce projet est réalisé sur :

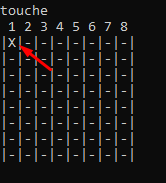
* Les programmes : Clion
* Système d’exploitation : Windows 10
* Programmation en C de base

## Modèle Logique de données

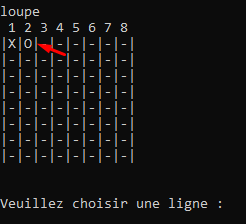
Le programme fonctionne par rapport au niveau. C’est-à-dire que le joueur choisit la difficulté de jeu et cette difficulté correspond à la taille de la map ainsi que le nombre de bateaux.



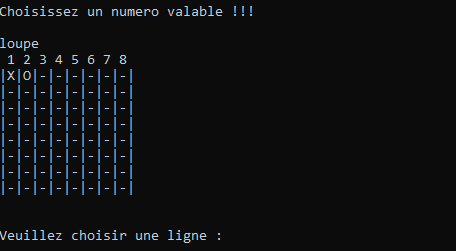
Voici à quoi ressemble la grille de base.



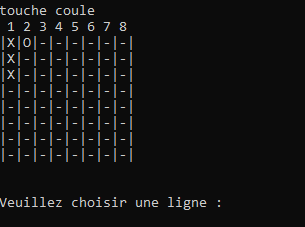
Voici à quoi elle ressemble quand elle est touchée.



Voici à quoi elle ressemble quand elle est loupée.



Quand elle n’est pas valable



Quand le bateau est coulé.

## Points techniques spécifiques

### Point 1 : Choisir un menu

Point 2 : Choisir une grille

### Point 1 : Saisir un numéro valable sinon le programme va demander de choisir une case en boucle.

Le programme va vérifier si la case a déjà été touché ou pas.

Point 2  : Saisir un numéro entre 1 et 3 (sinon la question sera reposée) qui définira une grille sur laquelle le joueur devra jouer.

## Livraisons

03.04.2019 : Première envoie du dossier « Bataille Navale » complet.

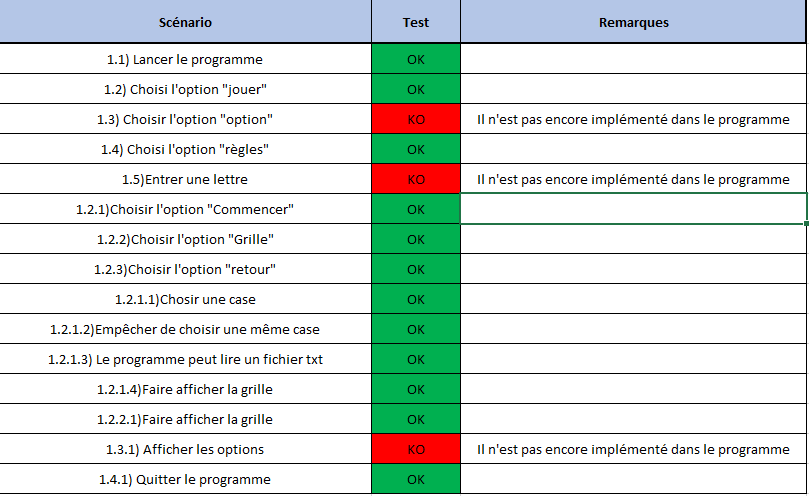
03.04.2019 : Mise à jour du dossier avec toutes les modifications faites.

04.04.2019 : Mise à jour du dossier avec le score.

<https://github.com/EaaswaralingamKaarththigan/Bataille-Navale>

# Tests

## Tests effectués



## Erreurs restantes

Problème au niveau de l’insertion de la case de la grille.

* + Si l’on met une lettre de l’alphabet, le programme crash.

# Conclusions

* Les objectif atteints :
  + Pouvoir jouer à la bataille navale.
  + Avoir nos propres règles.
  + Pouvoir avoir un score à la fin.
* Les objectif non-atteints :
  + IA non programmé.
  + Il n’y a pas d’interface personnalisé.
* Point positifs :
  + Le programme est fonctionnel et nous pouvons aisément y jouer dessus.
* Points Négatifs :
  + L’interface est simplette, il n’y a pas beaucoup de personnalisation.
* Difficultés :
  + Programmer le score
  + Créer l’option « coulé »

# Annexes

## Sources – Bibliographie

Aides externes :

* Jessy Borcard
* Adam Gruber
* Esteban Giorgis
* Pascal Benzonana
* Dylan Berney

Sites Webs :

* OpenClassroom

## Journal de bord du projet

Journal de travail

Jour Semaine Temps [h] Type Description

15.mars 1911 0.1 h Analyse Placement des issues dans les

sprints

15.mars 1911 0.016h Analyse Création des issues

15.mars 1911 0.016h Analyse Création des sprints

08.mars 1910 0.1 h Analyse Création du projet sur github

20.mars 1912 1h Implémentation Fonction Jouer

20.mars 1912 0.1 h Implémentation Afficher la grille

20.mars 1912 0.016h Implémentation Création de la grille

20.mars 1912 0.1 h Implémentation Afficher le menu

20.mars 1912 0.016h Implémentation Création du menu

20.mars 1912 0.1 h Documentation Création des objectifs

20.mars 1912 0.1 h Implémentation Création des uses cases

