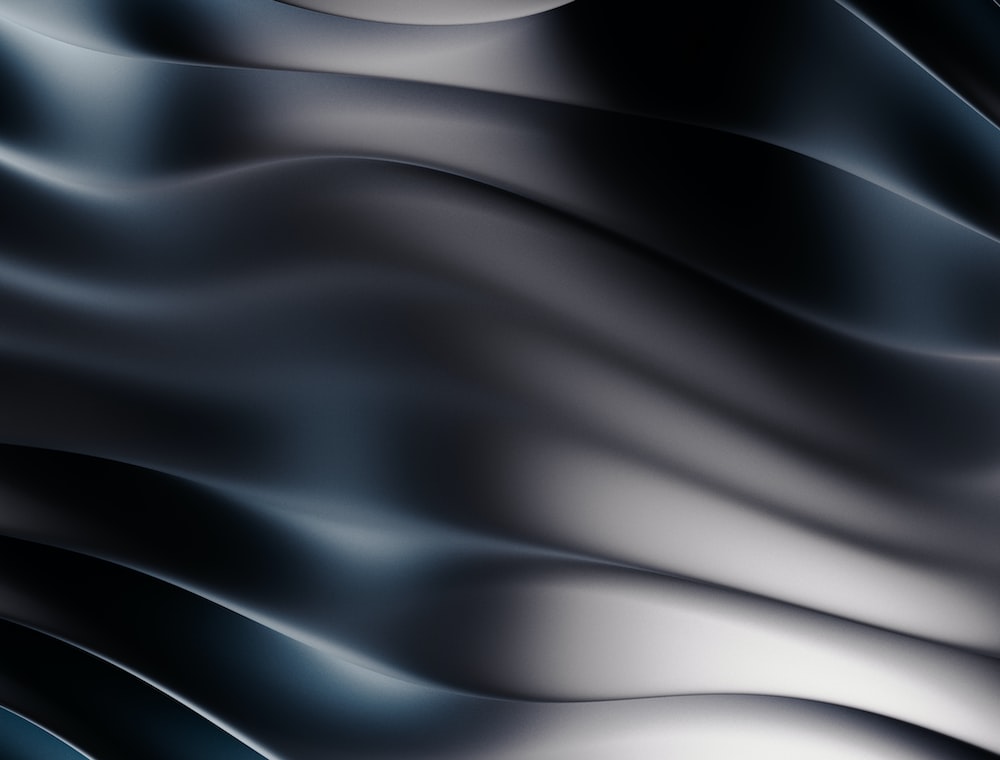


NimRL

Documentation Technique

Marco Rodrigues Marques

****

# Table des matières

[Table des matières 2](#_Toc133492499)

[Introduction 3](#_Toc133492500)

[Description de l’application : 3](#_Toc133492501)

[Étude d’opportunité 3](#_Toc133492502)

[Analyse fonctionnelle 3](#_Toc133492503)

[Analyse organique 3](#_Toc133492504)

[Méthodologie : 3](#_Toc133492505)

[ Exigences 4](#_Toc133492506)

[ Conception 4](#_Toc133492507)

[ Implémentation 4](#_Toc133492508)

[ Validation 4](#_Toc133492509)

[Améliorations possibles 4](#_Toc133492510)

[Conclusion 4](#_Toc133492511)

[Remerciements 4](#_Toc133492512)

[Table des illustrations 4](#_Toc133492513)

[Glossaire 4](#_Toc133492514)

[Sources 4](#_Toc133492515)

[Annexes 4](#_Toc133492516)

# Introduction

Ce document est un rapport des détails techniques de la réalisation du projet NimRL, destiné aux futures personnes impliquées dans ce projet.

## Description de l’application :

NimRL c’est une application qui a pour but la visualisation d’un processus d’apprentissage démontré sur le jeu de Nim.

Le jeu peut fonctionner selon trois modes :

* Joueur Humain contre joueur Humain
* Joueur Humain contre joueur IA
* Joueur IA contre joueur IA

L’application permet à l’utilisateur de jouer contre un autre joueur (Humain ou contrôlé par l’IA).

Elle permet également d’accélérer le processus d’apprentissage en faisant des parties automatiques.

Différentes informations sur l’Intelligence Artificielle et sur le processus d’apprentissage sont affichées dans l’application.

# Étude d’opportunité

Ce projet porte sur le thème de l’Intelligence Artificielle, cette dernière est de plus en plus présente dans nos vies, une meilleure compréhension des algorithmes utilisés par ces intelligences pour apprendre aide à les démystifier,

NimRL aide à comprendre l’apprentissage automatique par renforcement grâce à la visualisation d’un processus d’apprentissage.

# Analyse fonctionnelle

# Analyse organique

## Méthodologie :

Pour ce projet, j'ai décidé d'organiser mes tâches selon une version simplifiée du

Modèle en cascade, qui est composée de 4 phases :

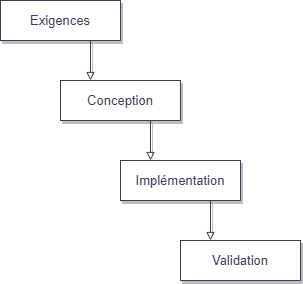


Fig. 1 : Modèle en cascade utilisé

* Exigences : Analyse des besoins du projet selon le cahier des charges.
* Conception : Élaboration de l'architecture logicielle (Diagramme de classes, Conception d'interfaces, Product Backlog).
* Implémentation : Réalisation du logiciel et des différents documents selon l’architecture définie précédemment.
* Validation : Vérification du produit selon les exigences.

Normalement, le modèle en cascade comprend les phases "Analyse" et "Maintenance", mais dans le but d'adapter cette méthodologie au TPI j'ai décidé de simplifier les phases "Exigences" et "Analyse" en une seule phase vu que le cahier de charges du projet nous est déjà fourni, j'ai également exclu la phase "Maintenance" car le projet n'est pas publié.

# Améliorations possibles

# Conclusion

# Remerciements

# Table des illustrations

# Glossaire

# Sources

# Annexes