HexaEditor

C# application

Atelier projet 2017

Serena SADEK – Michael RAMUSI

Table des matières

[1 Introduction 2](#_Toc482692849)

[1.1 Cahier des charges 2](#_Toc482692850)

[2 Analyse fonctionnelle 3](#_Toc482692851)

[2.1 Fonctionnalités principales 3](#_Toc482692852)

[2.2 Interface 4](#_Toc482692853)

[2.2.1 Fenêtre principale 4](#_Toc482692854)

[3 Analyse organique 5](#_Toc482692855)

[3.1 Diagramme de classe 5](#_Toc482692856)

[3.1.1 Classe Reader 5](#_Toc482692857)

[3.1.2 Classe ModelHexaEditor 6](#_Toc482692858)

[3.1.3 Classe ViewHexaEditor 7](#_Toc482692859)

# Introduction

Ce travail est réalisé dans le cadre de l’atelier projet 3ème année accélérée de l’année 2016-2017.

## Cahier des charges

HexaEditor, comme son nom l’indique, est une application d’édition en hexadécimal de fichiers. Elle permet l’édition octet par octet en hexadécimal mais également directement sous la représentation en ASCII. L’application se base sur une application déjà existante : EditHexa version 7.8.

# Analyse fonctionnelle

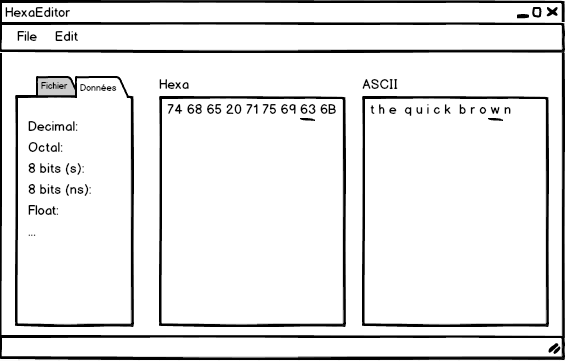
L’application est développée en .NET C# Windows Form.

## Fonctionnalités principales

* Ouverture d’un fichier
* Affichage de son contenu sous forme hexadécimal
* Affichage de son contenu sous forme ASCII
* Edition en hexadécimal
* Edition en ASCII
* Affichage de la valeur sous plusieurs formats (32 bits, décimal, octal, float etc.)
* Annulation de la dernière action d’édition (« ctrl+Z »)
* Sauvegarde des modifications

## Interface

### Fenêtre principale

L’application possède une vue principale qui permet d’ouvrir un fichier puis d’afficher son contenu sous différents formats. L’affichage des octets en hexa et ASCII se fait grâce à une génération dynamique de PictureBox.

**Picturebox :** Contenu du fichier en hexadécimal généré dynamiquement

**TabControl :**

Affichage sous différents formats par rapport à la position actuelle

**MenuStrip :**

Menu pour ouvrir un fichier, sauvegarder et annuler la dernière modification

**Picturebox :** Contenu du fichier en ASCII généré dynamiquement

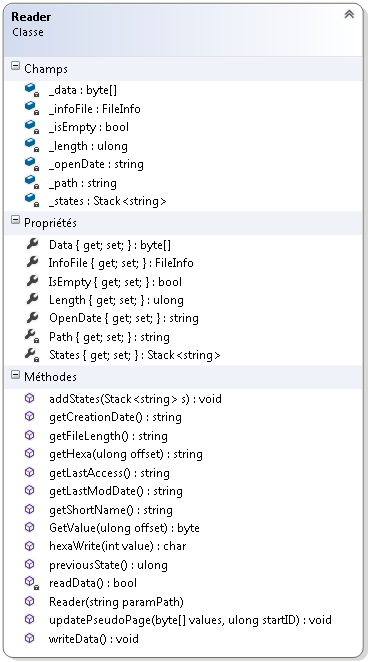
**TabControl :**

Affichage des informations du fichier (date, nom, dernière modification etc.)

# Analyse organique

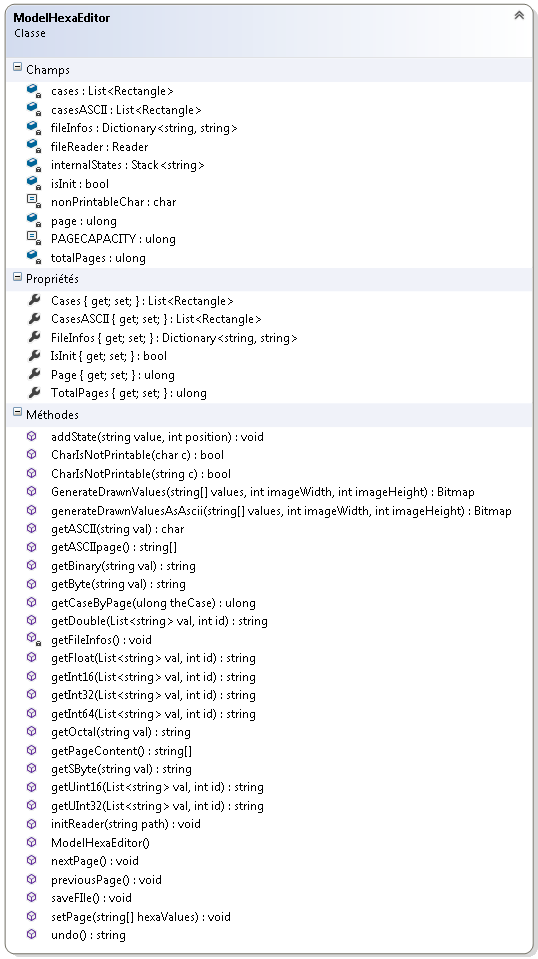
## Diagramme de classe

### Classe Reader



Reader est la classe qui s’occupe de gérer le fichier qui va être traité. Il stocke le contenu du fichier sous forme d’octets dans un tableau de byte (champs \_data). C’est également cette classe qui permet de retourner les informations du fichier comme sa date de dernière modification, son nom etc. Reader permet également de sauvegarder, écraser le fichier avec le contenu (tableau de byte[]) modifié par la classe ModelHexaEditor.

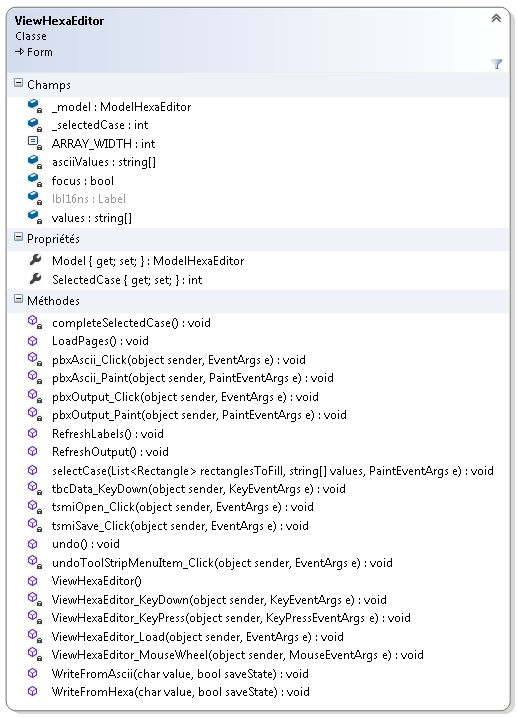
### Classe ModelHexaEditor



La classe ModelHexaEditor est la classe qui s’occupe du traitement principal.

Elle permet d’une part de générer les images Bitmap représentant le contenu du fichier sous forme hexadécimal et ASCII qui seront utilisés par les PictureBox de la vue (GenerateDrawnValues, GenerateDrawnValuesAsASCII). Et d’autre part, ModelHexaEditor contient les différentes méthodes de « conversions » sous différents formats (getInt, getOctal, getDecimal etc.). C’est également dans cette classe qu’est géré le système de pagination.

### Classe ViewHexaEditor



La classe ViewHexaEditor est la form qui gère les contrôleurs et les actions de l’utilisateur, comme le déplacement dans les cases des picturebox.

Elle possède également les rafraichissements de ces derniers avec les méthodes Paint. Cette classe gère aussi l’écriture de l’utilisateur dans les cases, que ce soit sur l’hexadécimal(WriteFromHexa) ou l’ascii (WriteFromAscii).