P\_Secu 114



(Une image originale représentant le projet)

Segalen Alban – mid2b

Sébeillon

24p

Helder Costa Lopes

Table des matières

[1 Spécifications 3](#_Toc187215603)

[1.1 Titre 3](#_Toc187215604)

[1.2 Description 3](#_Toc187215605)

[1.3 Matériel et logiciels à disposition 3](#_Toc187215606)

[1.4 Prérequis 3](#_Toc187215607)

[1.5 Cahier des charges 3](#_Toc187215608)

[1.5.1 Objectifs et portée du projet (objectifs SMART) 3](#_Toc187215609)

[1.5.2 Fonctionnalités requises (du point de vue de l’utilisateur) 3](#_Toc187215610)

[1.6 Les points suivants seront évalués 3](#_Toc187215611)

[1.7 Validation et conditions de réussite 4](#_Toc187215612)

[2 Analyse 4](#_Toc187215613)

[2.1 Document d’analyse et conception 4](#_Toc187215614)

[3 Réalisation 4](#_Toc187215615)

[3.1 Dossier de Réalisation 4](#_Toc187215616)

[3.2 Modifications 5](#_Toc187215617)

[4 Conclusion 5](#_Toc187215618)

[4.1 Bilan des fonctionnalités demandées 5](#_Toc187215619)

[4.2 Bilan personnel 5](#_Toc187215620)

[5 Divers 5](#_Toc187215621)

[5.1 Bibliographie 5](#_Toc187215622)

[5.2 Webographie 5](#_Toc187215623)

[6 Annexes 5](#_Toc187215624)

# introduction

## Titre

P\_Secu 114 : Gestionnaire de mot de passe

A compléter,par un titre cours et pertinent. Cela peut être une reprise ou compléter le titre de la première page …

## Description

Réaliser un gestionnaire de mot de passe

A compléter,par une explication du contexte, de la situation, des raisons générales de la mise en route d’un tel projet. Le lecteur doit pouvoir comprendre les motivations du lancement du projet…

## Matériel et logiciels à disposition

- 1 PC ETML standard

- Microsoft Visual Studio Code 2022

A compléter par ce qui est nécessaire pour le démarrage …

## Prérequis

Suivre le module 114

A compléter par une description des compétences, des connaissances et de la formation minimum pour être à même de réaliser le projet …

## Cahier des charges

### Objectifs et portée du projet (objectifs SMART)

Réaliser une application qui permet de gérer ses mots de passes.

### Fonctionnalités requises (du point de vue de l’utilisateur)

L’utilisateur utilise un Master Password pour accéder aux mots de passes. Le Master Password est utilisé comme clé de Vigenère.

On peut ajouter, modifier, supprimer un mot de passe.

Un mot de passe contient : un titre, l’URL du site, le nom d’utilisateur et le mot de passe.

A compléter par une espèce de mode d’emploi du produit. S’il s’agissait d’une montre, décrire qu’à part l’heure, il y aura la possibilité d’utiliser un chronomètre, un réveil, …

## Les points suivants seront évalués

* Le rapport
* Les planifications (initiale et détaillée)
* Le journal de travail
* Le code et les commentaires
* Les documentations de mise en œuvre et d’utilisation

## Validation et conditions de réussite

* Compréhension du travail
* Possibilité de transmettre le travail à une personne extérieure pour le terminer, le corriger ou le compléter
* Etat de fonctionnement du produit livré

# Description générale

## Présentation de l’application

## Fonctionnalités principales

Ajouter, supprimer, modifier et sauvegarder des mots de passes de manières sécurisé.

# algorithme et conception

## Chiffrement de Vigenère

## Schémas des algorithmes

# implémentation

## Structure du code

## Modules et fonctions principales

### Fonction de chiffrement

/// <summary>

/// Chiffre un texte

/// </summary>

/// <param name="text">Le texte à chiffrer</param>

/// <returns>Un texte chiffré</returns>

public static string Encrypt(string text)

{

string encryptedText = ""; //Le texte chiffré

int keyIndex = 0; //L'index de la clé

//Pour chaque charactère du texte

foreach (char c in text)

{

//Si l'index de la clé est plus grand que la longueur de la clé, on revient au début de la clé

if (keyIndex >= MasterPassword.Key.Length)

{

keyIndex = 0;

}

int characterValue = (int)c; //Le code du caractère à chiffrer

characterValue += (int)MasterPassword.Key[keyIndex]; //On ajoute le code du caractère de la clé au code du caratère à chiffrer

encryptedText += (char)characterValue; //On ajoute le caractère chiffré au texte chiffré

keyIndex++; //On incremente l'index de la clé

}

return encryptedText; //On retourne le texte chiffré

}

### Fonction de déchiffrement

/// <summary>

/// Déchiffre un text

/// </summary>

/// <param name="text">Le texte à déchiffrer</param>

/// <returns>Un texte déchiffré</returns>

public static string Decrypt(string text)

{

string decryptedText = ""; //Le texte déchiffré

int keyIndex = 0; //L'index de la clé

//Pour chaque charactère du texte

foreach (char c in text)

{

//Si l'index de la clé est plus grand que la longueur de la clé, on revient au début de la clé

if (keyIndex >= MasterPassword.Key.Length)

{

keyIndex = 0;

}

int characterValue = (int)c; //Le code du caractère à déchiffrer

characterValue -= (int)MasterPassword.Key[keyIndex]; //On soustrait le code du caractère de la clé au code du caratère à déchiffrer

decryptedText += (char)characterValue; //On ajoute le caractère déchiffré au texte déchiffré

keyIndex++; //On incremente l'index de la clé

}

return decryptedText; //On retourne le texte déchiffré

}

### Fonctions de sauvegardes

# sécurité

# tests et validation

# journal de travail

# Analyse

## Document d’analyse et conception

Algorithme

Le message = le texte à encrypter

Récupérer la valeur du premier caractère du message. On y ajoute la valeur du premier caractère de la clé. On y soustrait le nombre de possibilité (fait automatiquement avec c#). On répète l’opération avec la deuxième lettre du message et de la clé. Si on arrive au bout de la clé, on prend la première lettre de la clé.

[https://eduvaud.sharepoint.com/sites/msteams\_3cc8eb/Supports%20de%20cours/Forms/AllItems.aspx?id=%2Fsites%2Fmsteams%5F3cc8eb%2FSupports%20de%20cours%2FICT%2F114%2DCodificationChiffrement%2Fb%2DUnitesEnseignement%2FS%2D114%2DCompendio%2Epdf&parent=%2Fsites%2Fmsteams%5F3cc8eb%2FSupports%20de%20cours%2FICT%2F114%2DCodificationChiffrement%2Fb%2DUnitesEnseignement](https://eduvaud.sharepoint.com/sites/msteams_3cc8eb/Supports%20de%20cours/Forms/AllItems.aspx?id=%2Fsites%2Fmsteams_3cc8eb%2FSupports%20de%20cours%2FICT%2F114-CodificationChiffrement%2Fb-UnitesEnseignement%2FS-114-Compendio.pdf&parent=%2Fsites%2Fmsteams_3cc8eb%2FSupports%20de%20cours%2FICT%2F114-CodificationChiffrement%2Fb-UnitesEnseignement)

Chaque entrée est déjà chiffrée sur le disque.

Le stockage des mots de mot de passe ce fait en Json, car c’est un format largement utilisé et standardisé.

Encodage

Pour stocker la liste de mot de passe, la méthodes est la suivante :

On sérialise la liste de mot de passe en Json (l’entrée est déjà chiffrée).

On crée ensuite une liste de string qui contient le MasterPassword chiffré avec lui-même et le Json chiffré.

On sérialise cette liste en Json.

On écrit le Json ainsi obtenu dans le fichier de sauvegarde.

Ce paragraphe décrit le fonctionnement de manière détaillée.

Autant que possible de manière graphique, imagée, tableaux, etc.

Tous les cas particuliers devraient y être spécifiés…

Il s’agit d’y présenter les fonctionnalités à développer :

Découpage en étapes, en modules, en fonctionnalités, etc.

Formulaires, interfaces graphiques, pages web, etc.

Schémas de navigation, schémas événementiels, structogramme, pseudocode, etc.

Si le projet inclut une base de données :

Dictionnaire des données

Modèle conceptuel des données, modèles logique des données.

# Réalisation

## Dossier de Réalisation

Un seul fichier json.

Windows Forms.

Cette partie permet de reproduire ou reprendre le projet par un tiers.

Pour chaque étape, il faut décrire sa mise en œuvre. Typiquement :

Versions des outils logiciels utilisés (OS, applications, pilotes, librairies, etc.)

Configurations spéciales des outils (Equipements, PC, machines, outillage, etc.)

Code source commenté des éléments logiciels développés.

Modèle physique d’une base de données.

Arborescences des documents produits.

Il faut décrire le parcours de réalisation et justifier les choix.

## Modifications

Historique des modifications demandées (ou nécessaires) aux spécifications détaillées.

Date, raison, description, etc.

# Conclusion

## Bilan des fonctionnalités demandées

Il s’agit de reprendre point par point les fonctionnalités décrites dans les spécifications de départ et de définir si elles sont atteintes ou pas, et pourquoi.

Si ce n’est pas le cas, estimer en « % » ou en « temps supplémentaire » le travail qu’il reste à accomplir pour terminer le tout.

## Bilan personnel

Comme la sécurité des données est primordiale de nos jours, c’est un projet nécessaire.

J’ai bien aimé ce projet car il m’a permis de réaliser une application que je pourrais utiliser pour le restant de ma formation.

Ce projet m’a également permis de perfectionner mes compétences en C#.

Si c’était à refaire:

Qu’est-ce qu’il faudrait garder ? Les plus et les moins ?

Qu’est-ce qu’il faudrait gérer, réaliser ou traiter différemment ?

Qu’est que ce projet m’a appris ?

Suite à donner, améliorations souhaitables, …

Remerciements, signature, etc.

# Tests

|  |  |  |
| --- | --- | --- |
| Tests | Résultats | Validé |
| Entrer un master password erroné | L’application s’arrête | Oui |
| Ajouter une entrée | L’entrée est ajoutées | Oui |
| Quittez l’application et se connecter | Les mots de passes sont toujours disponibles | Oui |
| Supprimer une entrée | L’entrée est supprimée | Oui |
| Modifier le champ titre d’une entrée | Le titre est mis à jour | Oui |
| Quittez l’application et se connecter après avoir mis à jour le titre | Le titre est bien à jour | Oui |

# Divers

## Bibliographie

Références des livres, revues et publications utilisés durant le projet.

## Webographie

Références des sites Internet consultés durant le projet.

# Annexes

Listing du code source (partiel ou, plus rarement complet)

Guide(s) d’utilisation et/ou guide de l’administrateur

Etat ou « dump » de la configuration des équipements (routeur, switch, robot, etc.).

Extraits de catalogue, documentation de fabricant, etc.