

CELINE CHAUGNY  
DAMIEN POINTIN

# Plateforme d'analyse technique pour la finance

## Projet Fin d'Etude

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Introduction . . . . .	2
<b>2</b>	<b>Techniques utilisées</b>	<b>3</b>
2.1	Introduction . . . . .	3
2.2	Introduction . . . . .	3
2.3	Introduction . . . . .	3
2.4	Cryptage du mot de passe . . . . .	3
2.4.1	Fonctions SQL . . . . .	3
2.4.2	Jasypt . . . . .	3
2.4.3	Notre choix . . . . .	4
2.5	Introduction . . . . .	4
2.6	Introduction . . . . .	4
<b>3</b>	<b>Théorie Finance</b>	<b>5</b>
3.1	Introduction . . . . .	5
3.2	Introduction . . . . .	5
3.3	Introduction . . . . .	5
3.4	Introduction . . . . .	5
<b>4</b>	<b>Modélisation</b>	<b>6</b>
4.1	Introduction . . . . .	6
4.2	Introduction . . . . .	6
4.3	Introduction . . . . .	6
4.4	Introduction . . . . .	6
4.5	Introduction . . . . .	6
4.6	Introduction . . . . .	6
<b>5</b>	<b>Utilisation de l'application</b>	<b>7</b>
5.1	Introduction . . . . .	7
<b>6</b>	<b>Gestion de projet</b>	<b>8</b>
6.1	Introduction . . . . .	8
6.2	Introduction . . . . .	8
<b>7</b>	<b>Conclusion</b>	<b>9</b>
7.1	Introduction . . . . .	9
<b>8</b>	<b>Bibliographie</b>	<b>10</b>

# Chapitre 1

## Introduction

### 1.1 Introduction

# Chapitre 2

## Techniques utilisées

### 2.1 Introduction

### 2.2 Introduction

### 2.3 Introduction

### 2.4 Cryptage du mot de passe

Lors du déroulement du jeu nous allons avoir besoin de stocker le mot de passe du joueur. Pour cela, nous avons pensé qu'il était préférable d'encrypter ce mot de passe lors de son stockage. Nous avons ainsi cherché différentes méthodes pour effectuer l'encryptage du mot de passe.

#### 2.4.1 Fonctions SQL

Dans le langage SQL, la fonction MDA5() permet de chiffrer une chaîne de caractère en un entier hexadécimal de 32 caractères.

L'algorithme MDA() est une fonction de hachage cryptographique qui calcule à partir d'une chaîne de caractère son empreinte avec une probabilité très forte que deux empreintes soient différentes. Depuis 2004, une équipe chinoise a découvert des collisions complètes et MD5 n'est donc plus considéré comme sûr au sens cryptographique.

La fonction SQL SHA1() permet de chiffrer une chaîne de caractère sous la forme d'un chaîne de caractères de 40 caractères. SHA1 est également une fonction de hachage cryptographique. Elle a l'avantage d'être considéré comme sûr contrairement à MDA().

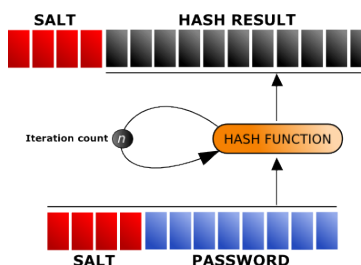
#### 2.4.2 Jasypt

Jasypt est une librairie java qui permet d'encrypter facilement les mots de passe avec une grande sécurité.

Jasypt possède les avantages suivants :

- il permet de choisir la fonction de hachage que nous souhaitons (MDA ou SHA par exemple)
- il ajoute un salage au mot de passe qui permet d'avoir deux mots de passe cryptés différents pour le même mot de passe de départ
- il applique un nombre aléatoire de fois notre fonction de hachage (nombre > 1000 pour rendre plus difficile les attaques)

Nous pouvons résumer l'algorithme de chiffrement de Jasypt comme suit :



Le code pour encrypter le mot de passe est très simple, il nous suffit de créer un objet de type ConfigurablePasswordEncryptor, de définir l'algorithme de chiffrement. La méthode setPlainDigest nous permet avec l'argument false de choisir la méthode la plus sûre avec un salage et un nombre d'itération aléatoire pour notre fonction de hachage. Enfin il ne nous reste plus qu'à appeler la méthode encryptPassword qui nous renvoie notre mot de passe encrypté à partir d'un mot de passe donné en entrée.

```
1 ConfigurablePasswordEncryptor passwordEncryptor = new ConfigurablePasswordEncryptor();
  passwordEncryptor.setAlgorithm( ALGO_CHIFFREMENT );
3 passwordEncryptor.setPlainDigest( false );
  String motDePasseChiffre = passwordEncryptor.encryptPassword( motDePasse );
```

De même que pour vérifier que notre mot de passe correspond au mot de passe chiffré il existe une méthode qui nous renvoie vraie en cas de correspondance :

```
passwordEncryptor.checkPassword(motDePasse, motDePasseChiffre )
```

### 2.4.3 Notre choix

L'inconvénient d'utiliser les fonctions de SQL est que l'on choisit une manière de crypter dépendante de notre base de données. Si nous décidons de changer notre manière de stocker notre base de données nous devons ainsi trouver une nouvelle fonction.

Nous avons donc choisi d'utiliser Jasypt pour encrypter notre mot de passe. Cette librairie a l'avantage de nous permettre d'encrypter une chaîne de caractère de manière relativement sûre sans avoir de grandes compétences en cryptographie. En effet, nous ne connaissons pas en détail le fonctionnement de l'algorithme de cryptage mais avons simplement une idée globale de son fonctionnement.

Nous avons choisi comme fonction de hachage (ALGOCHIFFREMENT) SHA car nous avons que MDA n'est plus sûr. Une fois l'encryptage du mot de passe effectué nous obtenons une chaîne de caractères de taille 56.

## 2.5 Introduction

## 2.6 Introduction

# Chapitre 3

## Théorie Finance

3.1 Introduction

3.2 Introduction

3.3 Introduction

3.4 Introduction

# Chapitre 4

## Modélisation

4.1 Introduction

4.2 Introduction

4.3 Introduction

4.4 Introduction

4.5 Introduction

4.6 Introduction

## Chapitre 5

# Utilisation de l'application

### 5.1 Introduction



## Chapitre 6

# Gestion de projet

### 6.1 Introduction

### 6.2 Introduction

## Chapitre 7

# Conclusion

### 7.1 Introduction

# Chapitre 8

## Bibliographie

<http://www.jasypt.org/howtoencryptuserpasswords.html>  
<http://sql.sh/fonctions/sha1>  
<http://sql.sh/fonctions/md5>

## RENSEIGNEMENTS

Département GM

02.32.95.65.31

gm@insa-rouen.fr

### INSA Rouen

Campus du Madrillet

685 avenue de l'Université – BP 08

76801 SAINT-ÉTIENNE-DU-ROUVRAY cedex

[www.insa-rouen.fr](http://www.insa-rouen.fr)

Membre de



Normandie Université

Financeurs institutionnels



MINISTÈRE  
DE L'ÉDUCATION NATIONALE,  
DE L'ENSEIGNEMENT SUPÉRIEUR  
ET DE LA RECHERCHE

