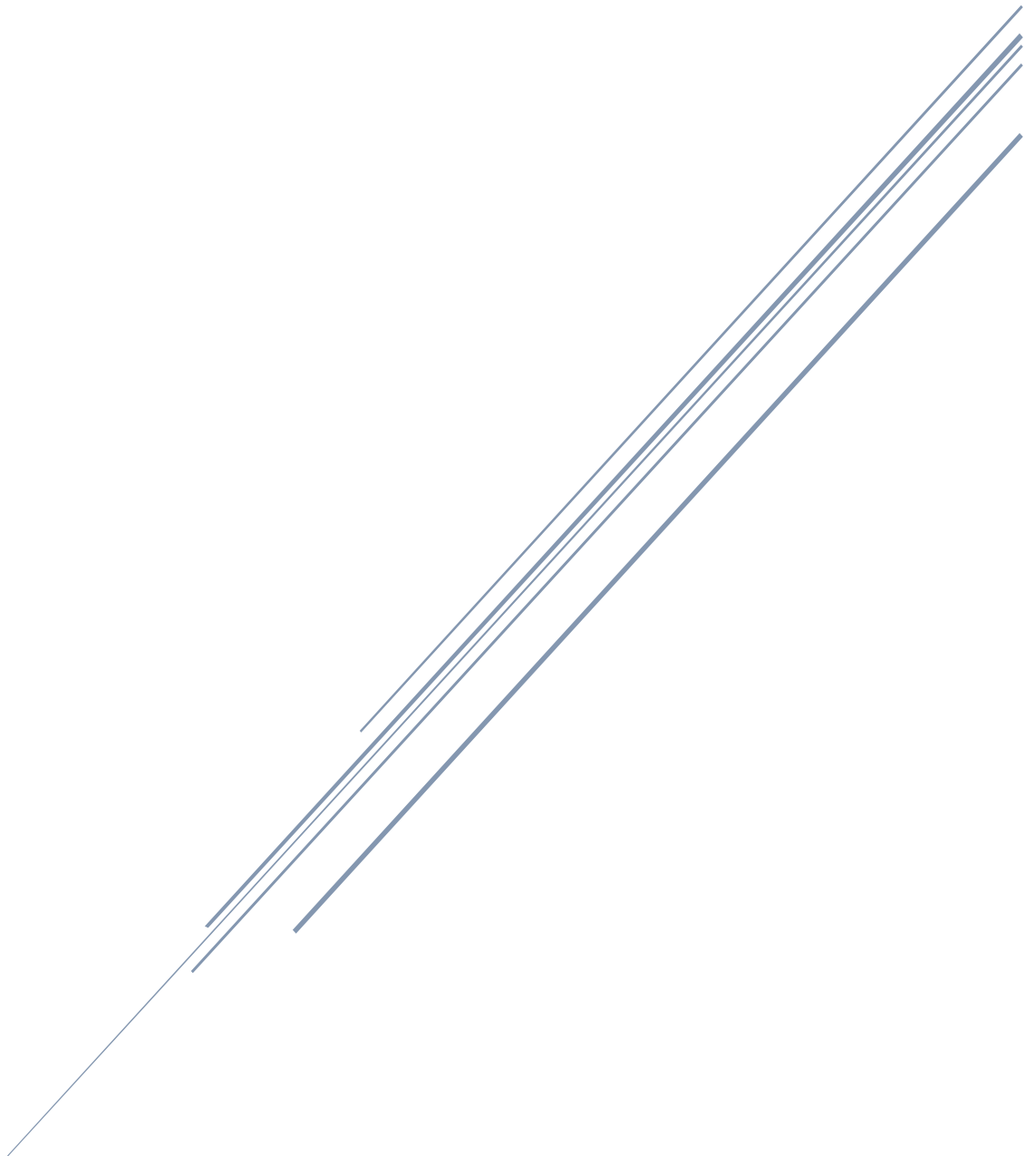


# MASTERCAMP - PROJET

Filière SR : Partage de fichiers chiffrés de bout en bout



Hassan SYLLA – Bilal GUIRRE – Houssein SYLLA – Amine  
GUENFOUD – Faël MOULOU DJ

## Table des matières

<b>Introduction.....</b>	<b>2</b>
<b>Conception de notre site web .....</b>	<b>3</b>
<b>Page d'accueil .....</b>	<b>4</b>
<b>Bcrypt .....</b>	<b>4</b>
<b>Page profil .....</b>	<b>6</b>
<b>Page Sélection des médecins.....</b>	<b>7</b>
<b>Page Boite de réception .....</b>	<b>8</b>
<b>Chiffrement AES .....</b>	<b>9</b>
<b>Conclusion .....</b>	<b>10</b>

## Introduction

Le secret médical est la base de confidentialité à respecter pour un professionnel intervenant dans le système de la santé. Mais malgré la fiabilité et le professionnalisme des médecins, nos données personnelles peuvent être utilisées en cas de faille au niveau des systèmes de sécurité. Le traitement et l'utilisation de ces données personnelles sont soumis à la loi RGPD. Elle permet la stabilité, la confiance ainsi que la transparence quant à l'utilisation des données des utilisateurs.

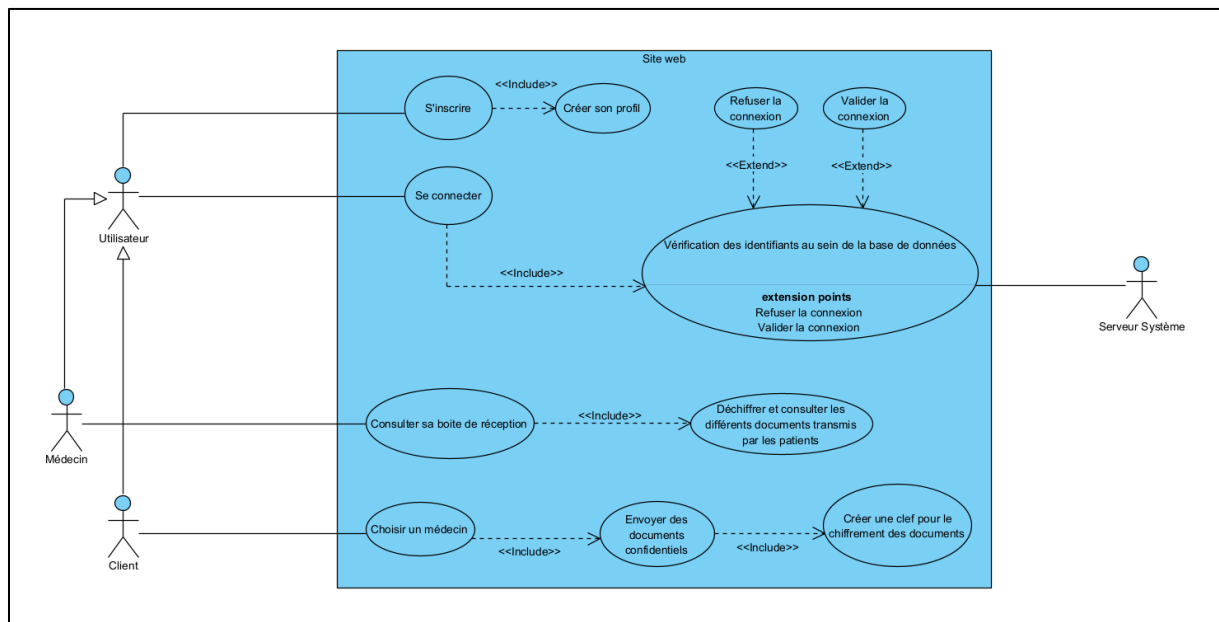
De ce fait, **comment les utilisateurs malades peuvent-ils pouvoir partager leurs documents personnels à leurs médecins sans compromettre leur propre confidentialité ?**

Ainsi, pour pouvoir régler cette problématique, nous avons donc répondu par un site web permettant à des patients, mais aussi à des médecins de pouvoir se connecter pour partager des données sensibles. En cela, le chiffrement de bout en bout des documents semble être une bonne solution pour pouvoir pallier le problème de confidentialité.

Ce site web est entièrement conçu grâce aux framework Vue.js et Express.js qui permettent de construire des applications web basées sur la plateforme de serveur web node.js en utilisant les langages de programmation HTML, CSS et JavaScript. De plus, les serveurs web sont également connectés à une base de données MySQL pour permettre une utilisation efficace des données des utilisateurs.

## Conception de notre site web

Pour la conception de notre site web, nous avons décidé de créer un diagramme UML des cas d'utilisations qui nous permet de mieux représenter les différents acteurs et utilisations de ce site.



Tout d'abord, l'utilisateur s'inscrit à la base de données de notre site en précisant son adresse mail, son mot de passe et son statut (patient ou médecin). Le statut de l'utilisateur est très important pour le fonctionnement du site web car le patient et le médecin n'ont absolument pas les mêmes fonctionnalités. Une fois l'inscription accomplie, l'utilisateur peut directement créer son profil lors de sa connexion avec la possibilité de pouvoir mettre en évidence : son nom, sa description et une photo de profil.

L'étape de la connexion permet au serveur de valider les identifiants écrits par l'utilisateur : si ces identifiants ne correspondent pas à des données incluses préalablement dans la base de données alors la connexion est immédiatement refusée.

### Côté Patient

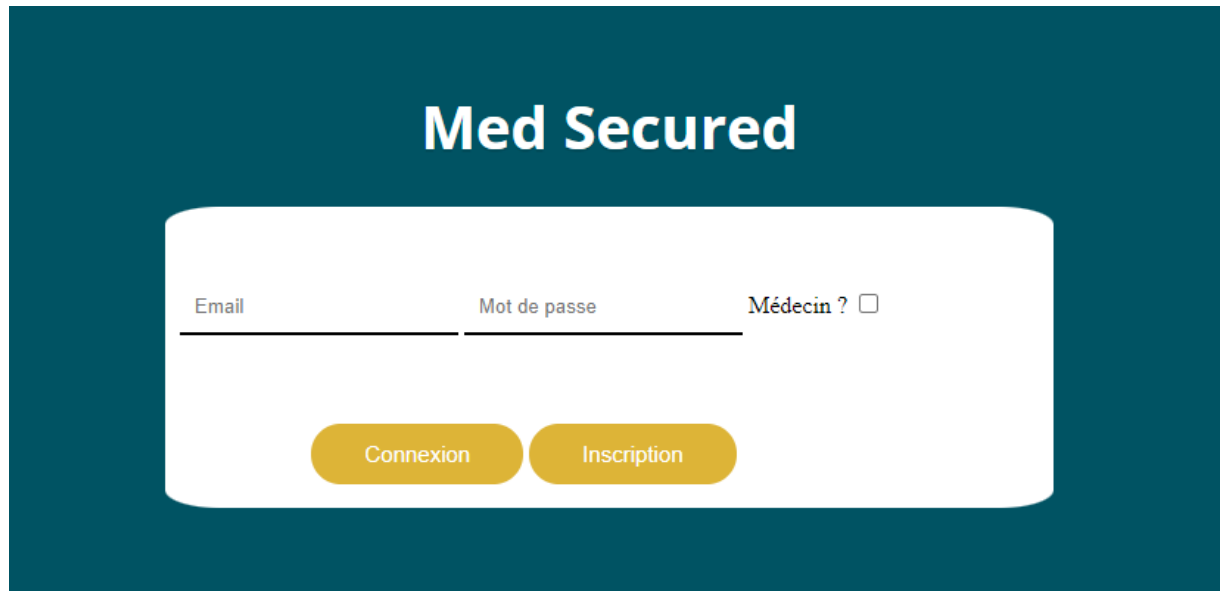
Lors de sa connexion, le patient a tout une liste de médecin qui lui est directement proposé sur le site web. A partir de là, il a donc le choix de sélectionner celui qui correspondrait le plus à ses besoins. Une fois le médecin choisi, le patient doit entrer ses informations confidentielles tout en choisissant une clef pour permettre l'envoi de ces documents chiffrés au médecin voulu.

### Côté Médecin

Lors de sa connexion, le médecin peut directement consulter sa boîte de réception avec toutes les quantités de documents chiffrés envoyés par les patients qui l'auraient choisi sur le site. A partir de là, le médecin doit avoir la clef appartenant à ce patient pour pouvoir lui permettre de déchiffrer ces documents en toute sécurité.

## Page d'accueil

Au démarrage du site, on atterrit tout d'abord sur la page d'accueil qui permet à l'utilisateur de se connecter ou de s'inscrire si ne n'est pas encore fait.

The image shows a web page with a dark teal background. At the top center, the text "Med Secured" is displayed in a large, white, sans-serif font. Below this, there is a white rounded rectangle containing a login and registration form. The form has three input fields: "Email", "Mot de passe", and "Médecin ?" followed by a checkbox. Below these fields are two yellow buttons: "Connexion" and "Inscription".

Med Secured

Email Mot de passe Médecin ? ☐

Connexion Inscription

Lorsqu'un utilisateur s'inscrit, il doit donc préciser une adresse mail, un mot de passe et s'il est médecin ou patient. Ces informations sont ensuite directement inscrites dans la base de données.

## Bcrypt

De nombreuses vulnérabilités peuvent apparaître sur ce site web et des attaques par injection SQL ne sont pas à prendre à la légère. C'est pour ça que nous avons mis en place un système de hachage des données grâce à **Bcrypt**. Cette fonctionnalité a pour but de hacher le mot de passe au sein de la base de données comme on peut le voir ci-dessous.

id_users	email	password
2	bilal.guirre@gmail.com	\$2a\$10\$fQg77V3NMpJty2cT9Kec6utmTRe7loQ...
3	jonathan.patrick@efrei.net	\$2a\$10\$v7MeFRtGza/HuokLw8PjIueeMNivmdxd...
4	jonathan.patrick@efrei.net	\$2a\$10\$/gWSTbtx36R26TIEuey7F.NGcr/0oZ9y...
5	fael.mouloudj@efrei.net	\$2a\$10\$9kGZOHNScITrRBdz3f8Cv.uTM1K5Z.7s...
6	medecin@gmail.com	\$2a\$10\$PToyOSl3TTWXBMUoTKk1cea4eBqdy2...

Pour permettre le hachage des données, Bcrypt utilise l'algorithme Eksblowfish qui est lui-même basée sur l'algorithme de chiffrement symétrique par bloc : Blowfish.

L'algorithme Eksblowfish permet d'établir des sous-clefs grâce à la clef et au sel. Ensuite, l'algorithme se repose sur le même principe que Blowfish en appliquant un certain nombre de tours de l'algorithme standard. Ce nombre de tour doit absolument être une puissance de deux. De plus, cette phase de planification de la clé garantie que tous les états sous-jacents doivent dépendre à la fois du sel et de la clé.

```

EksBlowfishSetup(cost, salt, key)
  state ← InitState()
  state ← ExpandKey(state, salt, key)
  repeat (2cost)
    state ← ExpandKey(state, 0, key)
    state ← ExpandKey(state, 0, salt)
  return state

```

On retrouve donc 3 paramètres pour cet algorithme :

- Cost : Nombre d'itérations
- Salt : Sel utilisé par l'algorithme
- Key : Le mot de passe qu'on souhaite hacher

On peut définir la fonction ExpandKey de cette manière :

```

ExpandKey(state, salt, key)
  for(n = 1..18)
    Pn ← key[32(n-1)..32n-1] ⊕ Pn //treat the key as cyclic
    ctext ← Encrypt(salt[0..63])
    P1 ← ctext[0..31]
    P2 ← ctext[32..63]
  for(n = 2..9)
    ctext ← Encrypt(ctext ⊕ salt[64(n-1)..64n-1]) // Encrypt utilise la clef actuelle et le sel sous forme
cyclique
    P2n-1 ← ctext[0..31]
    P2n ← ctext[32..63]
  for(i = 1..4)
    for(n = 0..127)
      ctext ← Encrypt(ctext ⊕ salt[64(n-1)..64n-1]) // comme au-dessus
      Si[2n] ← ctext[0..31]
      Si[2n+1] ← ctext[32..63]
  return state

```

Grâce au coût, on peut choisir le nombre d'itérations de cet algorithme pour le rendre beaucoup plus long, complexe et robuste, des facteurs de dissuasion pour les attaques par table arc-en-ciel et par force brute. De plus, l'utilisation de sels rend l'algorithme encore plus puissant, si bien qu'il est quasi impossible pour un hackeur de parvenir à déterminer le mot de passe originel. Bcrypt est aujourd'hui considéré comme la méthode de hachage la plus sûre.

## Page Profil

Lorsqu'un utilisateur se connecte, il peut modifier son profil. Il peut alors préciser son nom, sa description et sa photo de profil.


Pour un patient :

### Profil

**Nom : Anne Brigitte**

**Description : Gastro-entérite**

**Photo de Profil :**



Anne Brigitte	Gastro-entérite	<a href="https://aaaestrie.ca/wp-cont">https://aaaestrie.ca/wp-cont</a>
---------------	-----------------	---

Valider

Pour un médecin :

### Profil

**Nom : Jacques Thébault**

**Description : Médecin traitant**

**Photo de Profil :**




Jacques Thébault	Médecin traitant	<a href="https://static.lexpress.fr/mec">https://static.lexpress.fr/mec</a>
------------------	------------------	---

Valider

## Page Sélection des médecins

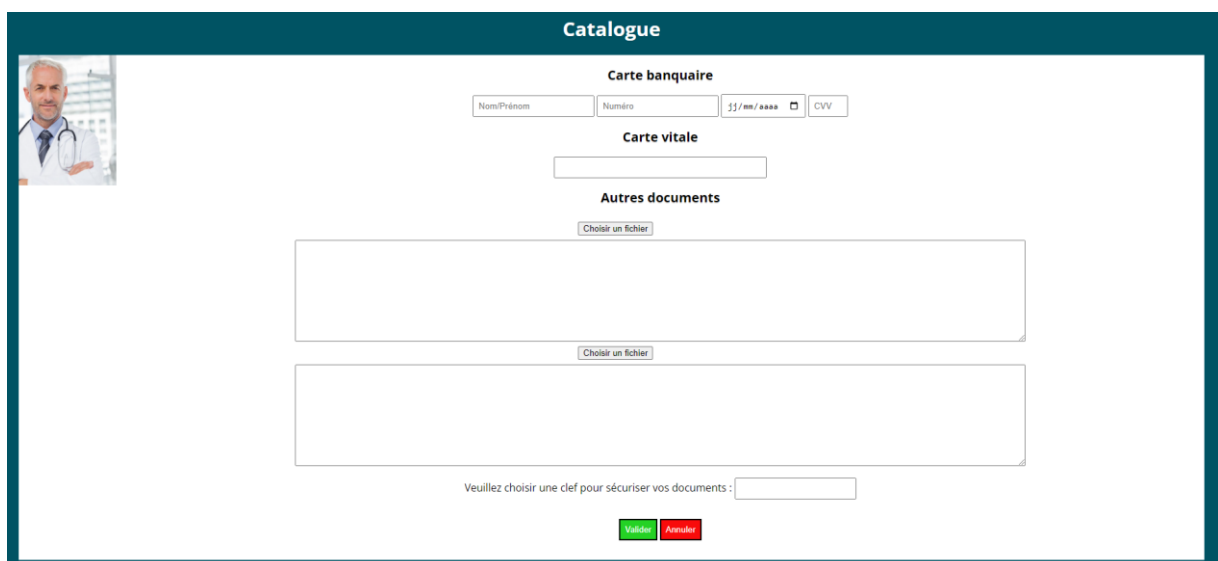
Le patient a la possibilité de se rendre sur un catalogue de médecins où il peut choisir celui qui lui convient le mieux.



The screenshot shows a web interface titled 'Catalogue' with a dark teal header. In the top right corner, there are links for 'Catalogue', 'Profil', and 'Déconnexion'. The main content area lists three doctors, each with a profile picture on the left and their name and specialty in the center, followed by a 'Choisir' button.

- Jonathan medecin - Je suis cardiologue** (Choisir)
- jean - dentiste** (Choisir)
- alexandre - medecin traitant** (Choisir)

Lorsque le patient sélectionne un médecin en cliquant sur le bouton « Choisir » en dessous, on a alors le menu suivant qui s’affiche :



The screenshot shows a form titled 'Catalogue' for document upload. It includes sections for 'Carte bancaire' (Nom/Prénom, Numéro, Expiry date, CVV), 'Carte vitale' (a single input field), and 'Autres documents' (two large file upload areas, each with a 'Choisir un fichier' button). At the bottom, there is a prompt to choose a security key and a 'Valider' button.

**Carte bancaire**

Nom/Prénom Numéro 55/mm/aaaa CVV

**Carte vitale**

**Autres documents**

Choisir un fichier

Choisir un fichier

Veuillez choisir une clef pour sécuriser vos documents :

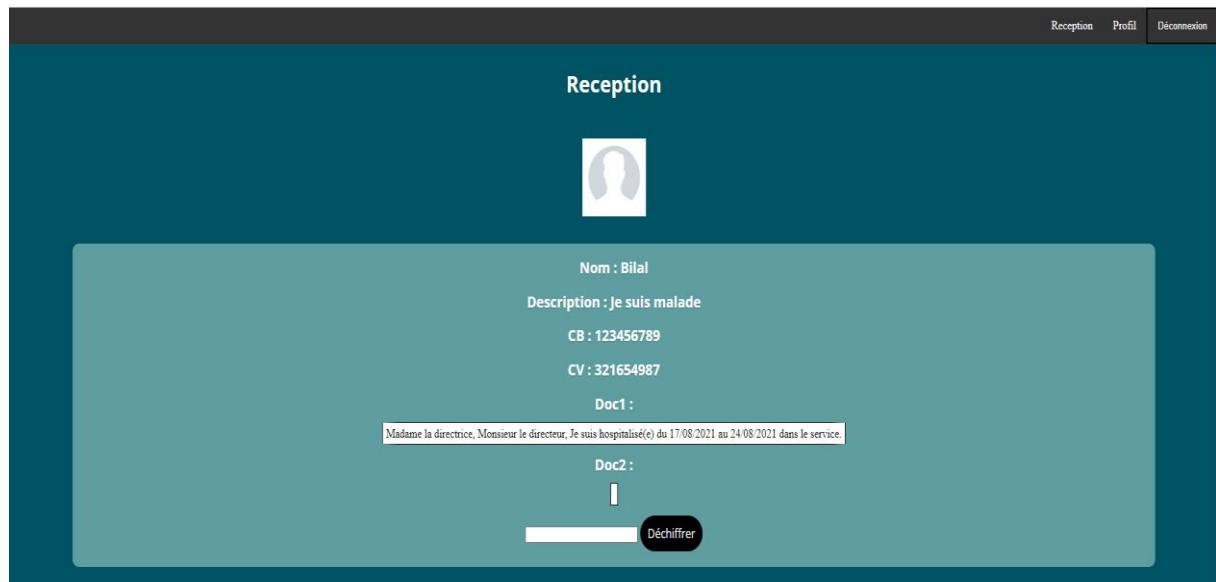
Valider Annuler

Sur ce menu, le patient peut charger des documents confidentiels, délivrer ses données bancaires et également écrire son numéro de sécurité sociale. Lorsqu’il finit, il peut ainsi créer une clef qui est connu par lui seul pour permettre le chiffrement des documents. Le bouton « Valider » envoie directement ces données à la boîte de réception du médecin.



## Page Boite de réception

Lorsqu'un médecin se connecte, il peut regarder sa boîte de réception pour consulter tous les documents qui lui sont envoyés par les patients. Il faut juste qu'il entre la clef correspondante à chaque client pour pouvoir déchiffrer le contenu des documents.



## Chiffrement AES

Pour le chiffrement des documents, nous avons utilisé le chiffrement AES aussi connu sous le nom de Rijndael, un algorithme de chiffrement symétrique.

Cet algorithme fonctionne par bloc : les données à chiffrer sont découpées par blocs. La taille de la clef peut varier pour permettre une plus grande quantité de combinaisons possible.

```
procedure Rijndael(State,Cipherkey)
  KeyExpansion(CipherKey,ExpandedKey)
  AddRoundKey(State,ExpandedKey[0])
  for i = 1 to Nr - 1 do
    Round(State,ExpandedKey[i])
  end for
  FinalRound(State,ExpandedKey[Nr])
end procedure
```

```
procedure Round(State,ExpandedKey[i])
  SubBytes(State);
  ShiftRows(State);
  MixColumns(State);
  AddRoundKey(State,ExpandedKey[i]);
end procedure
procedure FinalRound(State,ExpandedKey[Nr])
  SubBytes(State);
  ShiftRows(State);
  AddRoundKey(State,ExpandedKey[Nr]);
end procedure
```

Aujourd'hui AES est considéré comme fiable et sûr pour contrer les différentes attaques informatiques.

## Conclusion

Pour réaliser ce projet, nous avons dû nous organiser de sorte que le projet puisse avancer dans les meilleures conditions possibles tout en respectant les délais imposés. La charge de travail était conséquente, étant donné qu'il s'agissait d'une expérience différente. Nous avons uniquement 5 semaines pour nous consacrer entièrement à un projet, en respectant un cahier des charges, une problématique et un sujet clair. Cela ne fut pas facile, il s'agissait aussi de la découverte d'un domaine que nous n'avions pas traité sous cet angle. Nous avons donc dû nous documenter pour pouvoir mener à bien ce projet. Nous avons utilisé toutes les ressources dont nous disposions, ainsi que des différents éléments appris durant les séances de TP pour pouvoir avancer au mieux. Malgré tous les efforts fournis, nous sommes tout de même conscients que nous aurions pu mieux optimiser notre temps. En cela, nous aurions pu améliorer certains aspects du projet. Nous restons tout de même fiers du travail que nous avons produit et nous nous servons de tout ce que nous avons appris durant ces 5 semaines intenses pour pouvoir rebondir et effectuer un meilleur travail lors de nos projets futurs. Il est aussi important de souligner qu'il s'agissait d'une introduction au choix de notre majeur pour l'année suivante. Il s'agissait aussi d'améliorer notre bagage technique pour mener au mieux nos recherches de stage.