

Atelier 2 Mission 5 – Sécurité, qualité et logs

Contexte de la mission

La mission 5 a pour objectif de renforcer la sécurité, améliorer la qualité du code et mettre en place un système de journalisation (logs) afin de tracer les erreurs et comportements de l'application.

Elle se décompose en trois axes :

- Correction des failles de sécurité
- Amélioration de la qualité du code
- Mise en place des logs

Tâche 1 – Corriger des problèmes de sécuritéSécurisation des identifiants API

Problème initial :

Les identifiants d'accès à l'API étaient codés en dur dans l'application, ce qui constitue une faille de sécurité importante.

Solution mise en place :

- Externalisation des identifiants dans le fichier **App.config**
- Lecture dynamique via ConfigurationManager

Modifications réalisées :

- App.config :

```
<appSettings>
  <add key="credentialsApi" value="admin:adminpwd" />
</appSettings>
```

- Access.cs :

```
51 | // <summary>
52 | // Méthode privée pour créer un singleton
53 | // initialise l'accès à l'API
54 | // </summary>
55 | 1 référence
56 | private Access()
57 | {
58 |     String authenticationString;
59 |     try
60 |     {
61 |         authenticationString = ConfigurationManager.AppSettings["credentialsApi"];
62 |         api = ApiRest.GetInstance(uriApi, authenticationString);
63 |     }
64 |     catch (Exception e)
65 |     {
66 |         log.Fatal("Erreur lors de l'initialisation de l'accès à l'API", e);
67 |         Environment.Exit(0);
68 |     }
69 | }
```

Sécurisation de l'accès direct à l'API

Problème initial :

Un accès direct à l'API via navigateur affichait le contenu du dossier :

```
RewriteEngine on|
RewriteCond %{REQUEST_METHOD} =GET
RewriteRule http://localhost/rest_mediatekdocuments/
RewriteCond %{REQUEST_METHOD} =GET
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =GET
RewriteRule ^([a-zA-Z0-9_]+)/({.*})$ src/index.php?table=$1&champs=$2 [B,L]
RewriteCond %{REQUEST_METHOD} =POST
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =PUT
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =PUT
RewriteRule ^([a-zA-Z0-9_]+)/([a-zA-Z0-9_]+)$ src/index.php?table=$1&id=$2 [B,L]
RewriteCond %{REQUEST_METHOD} =DELETE
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =DELETE
RewriteRule ^([a-zA-Z0-9_]+)/({.*})$ src/index.php?table=$1&champs=$2 [B,L]
```

Solution mise en place :

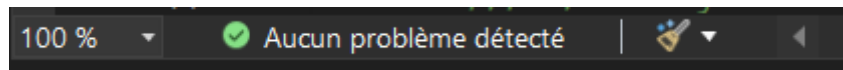
- Ajout d'une règle dans .htaccess pour bloquer l'accès

Modification réalisée :

```
RewriteEngine on
RewriteRule ^$ - [R=400,L]
RewriteCond %{REQUEST_METHOD} =GET
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =GET
RewriteRule ^([a-zA-Z0-9_]+)/({.*})$ src/index.php?table=$1&champs=$2 [B,L]
RewriteCond %{REQUEST_METHOD} =POST
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =PUT
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =PUT
RewriteRule ^([a-zA-Z0-9_]+)/([a-zA-Z0-9_]+)$ src/index.php?table=$1&id=$2 [B,L]
RewriteCond %{REQUEST_METHOD} =DELETE
RewriteRule ^([a-zA-Z0-9_]+)$ src/index.php?table=$1 [B,L]
RewriteCond %{REQUEST_METHOD} =DELETE
RewriteRule ^([a-zA-Z0-9_]+)/({.*})$ src/index.php?table=$1&champs=$2 [B,L]
```

Tâche 2 – Contrôler la qualité

Contrôle de SonarQube



Tâche 3 – Intégrer des logs

Mise en place des logs

Intégration de log4net

Ajouts réalisés :

- Package log4net 2.0.17
- Configuration dans App.config
- Initialisation dans Program.cs

```
25 </log4net>
26 <startup>
27   <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.7.2" />
28 </startup>
29 <runtime>
30   <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
31     <dependentAssembly>
32       <assemblyIdentity name="Newtonsoft.Json" publicKeyToken="30ad4fe6b2a6aed" culture="neutral" />
33       <bindingRedirect oldVersion="0.0.0.0-13.0.0.0" newVersion="13.0.0.0" />
34     </dependentAssembly>
35     <dependentAssembly>
36       <assemblyIdentity name="Gherkin" publicKeyToken="86496cfa5b4a5851" culture="neutral" />
37       <bindingRedirect oldVersion="0.0.0.0-37.0.1.0" newVersion="37.0.1.0" />
38     </dependentAssembly>
39     <dependentAssembly>
40       <assemblyIdentity name="Cucumber.Messages" publicKeyToken="b10c5988214f940c" culture="neutral" />
41       <bindingRedirect oldVersion="0.0.0.0-31.2.0.0" newVersion="31.2.0.0" />
42     </dependentAssembly>
43     <dependentAssembly>
44       <assemblyIdentity name="System.Runtime.CompilerServices.Unsafe" publicKeyToken="b03f5f7f11d50a3a" culture="neutral" />
45       <bindingRedirect oldVersion="0.0.0.0-6.0.3.0" newVersion="6.0.3.0" />
46     </dependentAssembly>
47     <dependentAssembly>
48       <assemblyIdentity name="System.Buffers" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
49       <bindingRedirect oldVersion="0.0.0.0-4.0.5.0" newVersion="4.0.5.0" />
50     </dependentAssembly>
51     <dependentAssembly>
52       <assemblyIdentity name="System.Threading.Tasks.Extensions" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
53       <bindingRedirect oldVersion="0.0.0.0-4.2.4.0" newVersion="4.2.4.0" />
54     </dependentAssembly>
55     <dependentAssembly>
56       <assemblyIdentity name="ZstdSharp" publicKeyToken="8d151af33a4ad5cf" culture="neutral" />
57       <bindingRedirect oldVersion="0.0.0.0-0.8.7.0" newVersion="0.8.7.0" />
58     </dependentAssembly>
59   </assemblyBinding>
60 </runtime>
61 </configuration>
```


```
1 using MediaTekDocuments.model;
2 using MediaTekDocuments.view;
3 using System;
4 using System.Windows.Forms;
5 using log4net;
6 using log4net.Config;
```

Utilisation dans l'application

Dans Access.cs :

```
using System;  
using System.Collections.Generic;  
using MediaTekDocuments.model;  
using MediaTekDocuments.manager;  
using Newtonsoft.Json;  
using Newtonsoft.Json.Converters;  
using Newtonsoft.Json.Linq;  
using System.Configuration;  
using log4net;
```

Fichier log créé.

| | | | |
|---|------------------|----------------|------|
|  mediatekdocuments.log | 17/03/2026 19:13 | Document texte | 0 Ko |
|---|------------------|----------------|------|

KanBan :

In Progress 1

This is actively being worked on

mediatekdocuments-API-PABAQUE #7

Mission 5 – Assurer la sécurité, la qualité et intégrer des logs

Mission 5 – Assurer la sécurité, la qualité et intégrer des logs #7

Open pab1307/mediatekdocuments-API-PABAQUE **Public**

pab1307 opened on Jan 26 · edited by pab1307 Edits Owner

Tâche 1 – Corriger des problèmes de sécurité

Pour connaître l'intitulé des problèmes, voir le document [missions.pdf](#). Pour chaque problème, créer une "issue" dans le dépôt GitHub correspondant, affecter l'issue à un des développeurs (si vous êtes 2, vous ou votre collègue, sinon vous-même), créer une branche pour proposer une correction de code en expliquant la correction, faire un pull request, accepter le pull request si le résultat correspond bien aux attentes.

Tâche 2 – Contrôler la qualité

Contrôler que SonarQube for IDE est configuré dans Visual Studio.
Corriger les problèmes relevés par SonarQube for IDE dans le code ajouté (excepté les problèmes qui ne doivent pas être corrigés, comme les noms des méthodes événementielles qui commencent par une minuscule).

Tâche 3 – Intégrer des logs

Dans la classe Access, ajouter le code de configuration des logs et des logs au niveau de chaque affichage console (à enregistrer dans un fichier de logs).

Temps estimé : 4h
temps réel : 3h40

Create sub-issue