

RAPPORT DE STAGE

CHIMÈNE JUNIOR

STAGE EFFECTUÉ

DU 26 AOÛT 2024 AU 18 OCTOBRE 2024



SOMMAIRE

SOMMAIRE	2
INTRODUCTION	3
I. Présentation de l'entreprise	3
II. Principaux services et produits	4
OBJECTIFS du STAGE	7
ENVIRONNEMENT TECHNIQUE	8
MAQUETTAGE	9
I. Ancienne version	9
II. Nouvelle version	10
RÉFLEXION SUR LA COUCHE D'ACCÈS AUX DONNÉES	11
I. Pourquoi j'ai opté pour l'ORM ? (Entity Framework)	11
MODÈLE LOGIQUE DES DONNÉES	11
IMPLÉMENTATION DE LA COUCHE D'ACCÈS AUX DONNÉES	13
TESTS UNITAIRE	14
CONCLUSION	16
REMERCIEMENT	16

Introduction

Lors de ma formation Compétence Plus, j'ai commencé à chercher un stage et j'ai contacté plusieurs entreprises. Ergoss s'est rapidement intéressée à mon profil, d'autant plus qu'ils ont déjà accueilli plusieurs personnes issues de reconversion professionnelle, ce qui a facilité les démarches.

I.Présentation de l'entreprise

Fondée en 2009, Ergoss est une entreprise autofinancée spécialisée dans les solutions de traitement des données aéronautiques.

Avec des bureaux à Toulouse, Miami et Shanghai, l'entreprise est à l'avant-garde de l'innovation dans le domaine de la gestion des données de vol et des systèmes de maintenance.

Ergoss se distingue par son équipe de 24 experts, répartis entre support technique, développement logiciel, recherche et innovation, et gestion de projets spéciaux.

L'entreprise gère plus de 2900 appareils pour 66 opérateurs dans le monde entier, avec des flottes allant de 1 à plus de 400 avions.

Ergoss propose une suite de plateformes et de systèmes intégrés regroupés sous l'enseigne SARA (SARA Suite), qui se compose de trois plateformes et cinq systèmes principaux, conçus pour répondre aux besoins de traitement de données aéronautiques.

PRINCIPAUX SERVICES ET PRODUITS

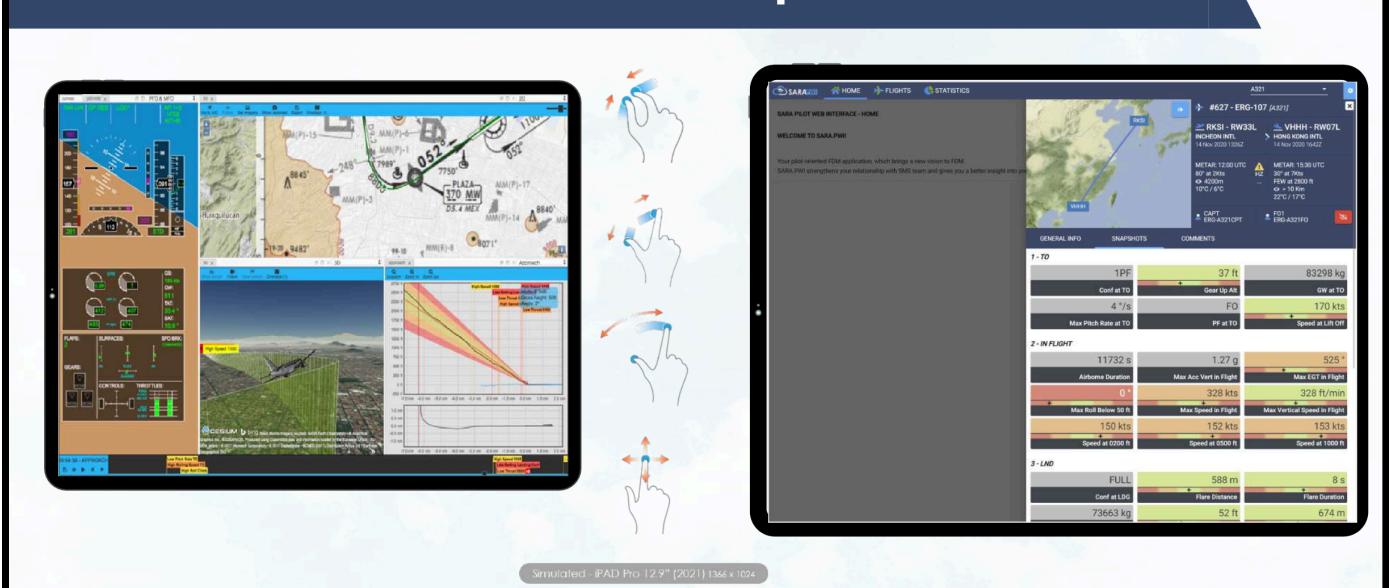
SARA.FDM : Intuitive Chart Mapping



1.SARA.FDM (Flight Data Monitoring & Ops Support) :

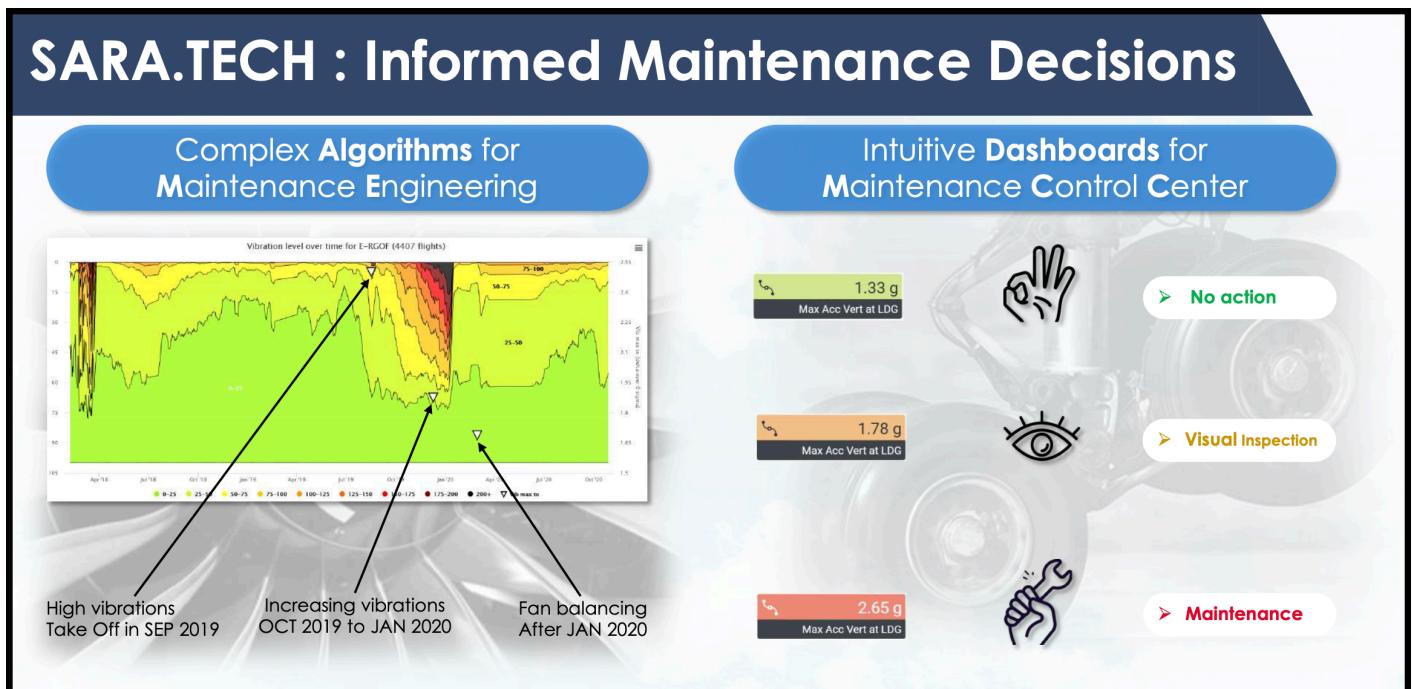
- Surveillance des données de vol avec des fonctionnalités avancées, notamment la relecture 4D des vols et l'analyse intuitive des graphiques.

SARA.PWI : Pilot's Personal Reports & Trends



2. SARA.PWI (Pilot Work Interface) :

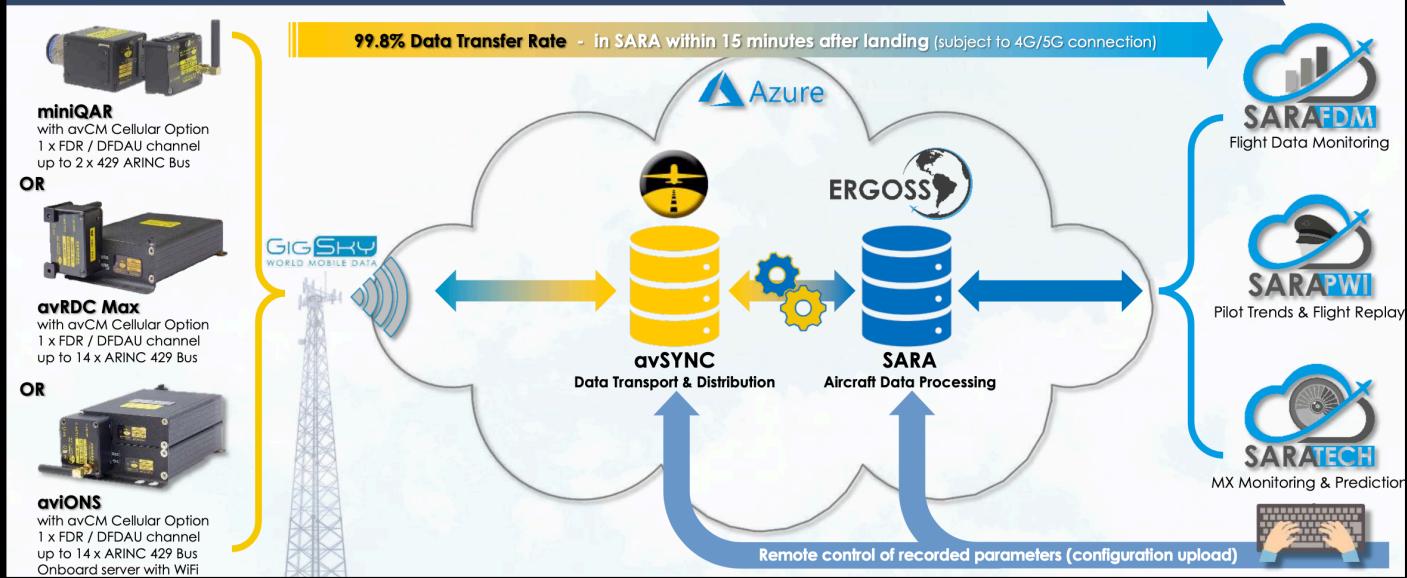
- Fournit aux pilotes des rapports personnels, des tendances de vol et un accès interactif à la relecture de leurs vols. Le système permet une immersion complète dans l'expérience de vol via une simple interface.



3. SARA.TECH :

- Destiné à la maintenance prédictive, ce système analyse les données de maintenance (MX) et permet aux équipes de prendre des décisions éclairées concernant les inspections visuelles ou les interventions t

Connected Aircraft Solutions, and beyond ...



4. Solutions connectées pour les avions :

Ergoss offre des solutions connectées pour un transfert rapide et sécurisé des données de vol, avec un taux de 99,8 % grâce à des dispositifs comme le miniQAR et l'avRDC Max.

OBJECTIFS DU STAGE

Dans le cadre de la mise en conformité avec la norme ISO 27001, qui impose une gestion rigoureuse des risques et des politiques de sécurité (formation, matériel, etc.), mon tuteur a constaté que l'utilisation de fichiers Excel n'était plus adaptée.

Mon objectif, pendant ce stage, est donc de concevoir et développer un outil informatique qui permettra de centraliser et gérer de manière plus efficace :

- L'analyse des risques et les politiques de sécurité (gestion des actifs, rôles et permissions).
- Le suivi des différents processus (formation, revue d'accès).
- Les vérifications régulières ainsi que l'amélioration continue des systèmes de sécurité.

L'outil que je vais développer doit aussi faciliter la gestion des preuves de conformité, les analyses régulières, et la validation des demandes de changement, afin de garantir la conformité et l'amélioration continue selon les exigences de l'ISO 27001.

J'ai travaillé en étroite collaboration avec l'équipe sécurité pour mener à bien ce projet.

Cette application ne devra pas être accessible depuis l'extérieur (réservée à l'intranet).
ques.

ENVIRONNEMENT TECHNIQUE

LANGAGES ET FRAMEWORKS :

- C# (framework standard)
- Angular (pour l'interface web)

OUTILS DE DÉVELOPPEMENT :

- Visual Studio (environnement de développement intégré)
- VS Code (éditeur de code)
- Postman (outil pour tester les API)
- SQL Server Management Studio (outil de gestion de bases de données SQL Server)

MAQUETTAGE

Lors de mon stage, il m'a été demandé de passer directement à la réalisation des mockups, sans passer par les étapes habituelles du zoning et des wireframes, afin de respecter les délais restreints et de maximiser l'efficacité du projet. Il y avait déjà une application existante, mais celle-ci ne me semblait pas suffisamment ergonomique. J'ai donc pris l'initiative de proposer un nouveau mockup, qui a finalement été accepté par mon tuteur et la responsable du Customer Care.

I. ANCIENNE VERSION

Risk Management

ERGOSS

ID	Date	Owner	Group	Asset	Statements	Description	Likelihood	Gravity	Impact	Decision	Treatment
Risk - DF - 220	04/09/2024	DFT	Default	Assets	Enter Statement	Definition	Very Unlikely	Negligible	Default	Accepter	Enter a treatment

Add Save

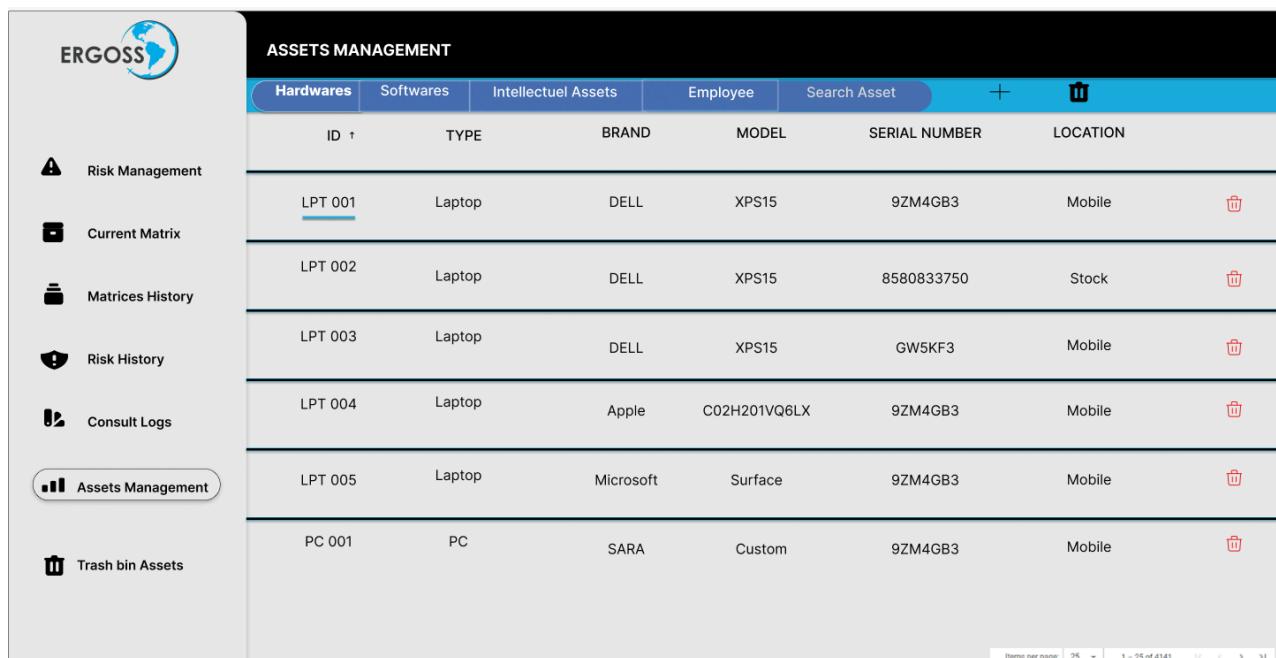
Current Matrix

ERGOSS

		Gravity +			Risk Levels			
		Negligible	Moderate	Severe	0 MEDIUM	Risk should be reduce	trash	
Probability +	Very Unlikely less than once a year	Default	Default	Default	1 LOW	Risk can be accepted	trash	
	Possible More than once a year	Default	Default	Default	2 Default	Default 1	trash	
	Very Likely More than once a month	Default	Default	Default	3 HIGH		trash	
Consequence Types +	Default	No Injury	Serious Injury	Fatality	Clone this matrix			
Close the analysis								

J'ai constaté que l'ancienne version ne respectait pas certaines règles de lisibilité, notamment pour les personnes avec certains handicaps, et que l'ergonomie n'était pas très pratique. C'est pourquoi je me suis permis de proposer une nouvelle version, qui a été acceptée.

II. NOUVELLE VERSION



The screenshot displays a modern web-based application for asset management. On the left, a sidebar menu lists navigation options: Risk Management, Current Matrix, Matrices History, Risk History, Consult Logs, Assets Management (which is currently selected), and Trash bin Assets. The main content area is titled "ASSETS MANAGEMENT" and contains a table with the following data:

ID ↑	TYPE	BRAND	MODEL	SERIAL NUMBER	LOCATION	
LPT 001	Laptop	DELL	XPS15	9ZM4GB3	Mobile	
LPT 002	Laptop	DELL	XPS15	8580833750	Stock	
LPT 003	Laptop	DELL	XPS15	GW5KF3	Mobile	
LPT 004	Laptop	Apple	C02H201VQ6LX	9ZM4GB3	Mobile	
LPT 005	Laptop	Microsoft	Surface	9ZM4GB3	Mobile	
PC 001	PC	SARA	Custom	9ZM4GB3	Mobile	

At the bottom right of the table, there are pagination controls: "Items per page: 25", "1 - 25 of 4141", and navigation arrows (left, right, first, last).

RÉFLEXION SUR LA COUCHE D'ACCÈS AUX DONNÉES

On m'a demandé de faire des recherches sur la couche d'accès aux données et de choisir une option. J'en ai identifié trois :

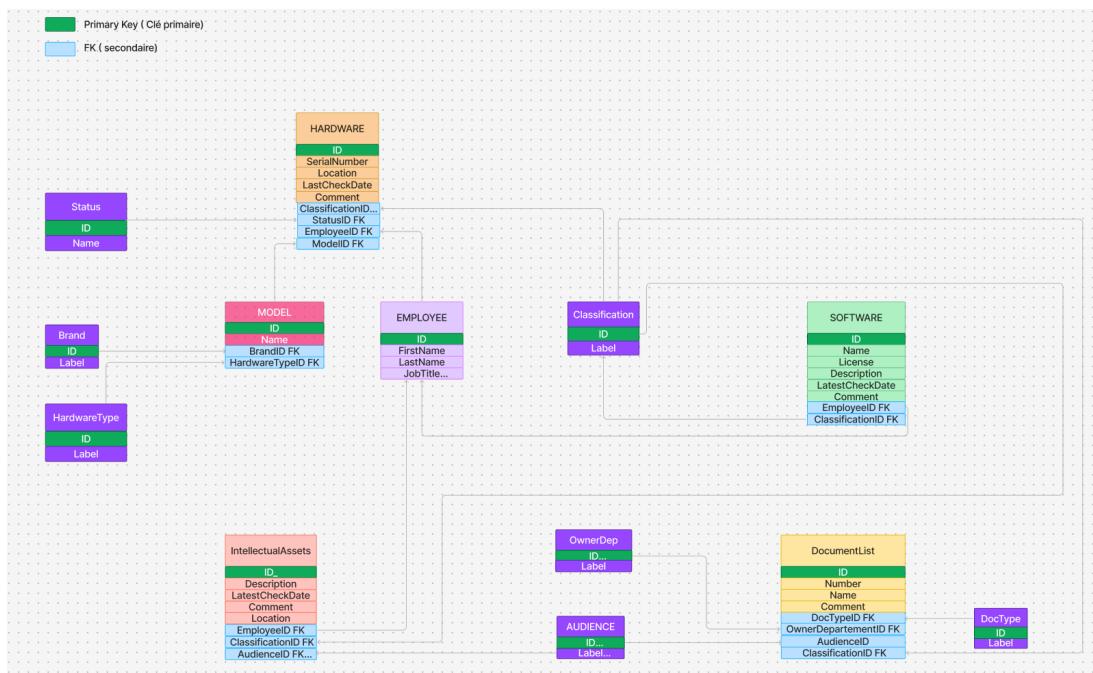
- La couche d'accès aux données directe:
- La couche d'accès aux données avec des objets d'accès aux données (DAO)
- La couche d'accès aux données avec un ORM (Object-Relational Mapper)

I.POURQUOI J'AI OPTÉ POUR L'ORM ? (ENTITY FRAMEWORK)

Afin d'accélérer le développement et d'améliorer la maintenabilité de l'application,

J'ai opté pour un ORM. Ce choix permet de bénéficier d'une abstraction élevée de la base de données, de réduire le risque d'erreurs liées à l'accès aux données et de faciliter l'évolution du modèle de données.

MODÈLE LOGIQUE DE DONNÉES



Grâce à l'utilisation du modèle conceptuel MLD (Modèle Logique des Données), j'ai pu concevoir et créer ma base de données :

```

USE _ISO27001;

CREATE TABLE [dbo].[Brand](
    id INT PRIMARY KEY IDENTITY(1,1),
    name VARCHAR(50) NOT NULL
);

CREATE TABLE [dbo].[hardwareType] (
    id INT PRIMARY KEY IDENTITY(1,1),
    name NVARCHAR(100) NOT NULL
);

CREATE TABLE [dbo].[Model] (
    id INT PRIMARY KEY IDENTITY(1,1),
    name NVARCHAR(100) NOT NULL,
    brandId INT,
    hardwareTypeId INT,
    FOREIGN KEY (BrandId) REFERENCES Brand(id),
    FOREIGN KEY (hardwareTypeId) REFERENCES hardwareType(id)
);

CREATE TABLE [dbo].[Employee] (
    id INT PRIMARY KEY IDENTITY(1,1),
    firstName NVARCHAR(50) NOT NULL,
    lastName NVARCHAR(50) NOT NULL,
    jobTitle VARCHAR(100) NULL
);

CREATE TABLE [dbo].[classification] (
    id INT PRIMARY KEY IDENTITY(1,1),
    name NVARCHAR(100) NOT NULL,
    description NVARCHAR(MAX)
);

```

Bases de données

- Bases de données système
- _ISO27001**
 - Diagrammes de base de données
 - Tables
 - Tables système
 - Tables externes
 - Tables de graphe
 - dbo.Analyse
 - dbo.Audience
 - dbo.Brand
 - dbo.classification
 - dbo.ConsequenceGravity
 - dbo.ConsequenceType
 - dbo.Decision
 - dbo.Department
 - dbo.DocumentList
 - dbo.DocumentType
 - dbo.Employee
 - dbo.Gravity
 - dbo.Groups
 - dbo.Hardware
 - dbo.hardwareType
 - dbo.Likelihood
 - dbo.Logs
 - dbo.Matrix
 - dbo.MatrixCell
 - dbo.Model
 - dbo.ResidualRisk
 - dbo.ResidualRiskTreatment
 - dbo.Risk
 - dbo.RiskGroup
 - dbo.RiskLevel
 - dbo.RiskList
 - dbo.RiskTreatment
 - dbo.status
 - dbo.Treatment
 - dbo.Users

J'ai rédigé mes requêtes pour créer les tables dans ma base de données (BDD)

Le résultat de la création de mes tables dans la base de données

IMPLEMENTATION DE LA COUCHE D'ACCÉS AUX DONNÉES

```
1  using Microsoft.Data.SqlClient;
2  using Dapper;
3  using System.Data;
4
5  namespace ApiRiskManagement.services.Asset
6  {
7      public class BrandRepository
8      {
9          private readonly IDbConnection _connection;
10
11         public BrandRepository (IDbConnection connection)
12         {
13             _connection = connection;
14         }
15
16
17         /// <summary>
18         /// Add a new Brand.
19         /// </summary>
20         /// <param name="brand">The Brand object to add.</param>
21         public int Post(Brand brand)
22         {
23             string sql = @"INSERT INTO brand (Name) VALUES (@Name) ;
24                 SELECT SCOPE_IDENTITY() AS 'int'";
25             return _connection.Query<int>(sql, brand).FirstOrDefault();
26         }
27
28         /// <summary>
29         /// Update an existing Brand.
30         /// </summary>
31         /// <param name="brand">The Brand object with updated information.</param>
32         public void Update(Brand brand)
33         {
34             string sql = "UPDATE brand SET Name = @Name WHERE Id = @Id";
35             _connection.Execute(sql, brand);
36         }
37     }
38 }
```

J'ai créé une classe appelée BrandRepository. Elle sert à gérer les opérations CRUD (Créer, Lire, Mettre à jour, Supprimer) pour une entité qu'on appelle Brand.

J'ai utilisé Dapper, une bibliothèque super pratique qui simplifie la communication avec la base de données SQL.

Dans ma classe BrandRepository, j'ai ajouté un constructeur qui prend un paramètre de type IDbConnection. Cela me permet d'initialiser la connexion à la base de données quand je crée une instance de la classe.

TESTS UNITAIRE

```
1  |  using ApiRiskManagement.services.Asset;
2  |  using System.Data.SqlClient;
3  |
4  |  namespace UnitTests
5  |  {
6  |      [TestClass]
7  |      0 références
8  |      public class UnitTest1
9  |      {
10 |          // This class defines unit tests for the 'BrandRepository' class.
11 |          private BrandRepository repository;
12 |          private int TestId;
13 |
14 |          // Creates a new connection string
15 |          [TestInitialize]
16 |          0 références
17 |          public void Init()
18 |          {
19 |              var connection = new SqlConnection("Server=tcp:saradbwe.database.windows.net");
20 |              repository = new BrandRepository(connection);
21 |
22 |              var obj = new Brand()
23 |              {
24 |                  Name = "Test",
25 |              };
26 |
27 |              TestId = repository.Post(obj);
28 |          }
29 |
30 |          // Tests the Post method of the BrandRepository
31 |          [TestMethod]
32 |          0 références
33 |          public void Test001_PostBrand()
34 |          {
35 |              var obj = new Brand()
36 |              {
37 |                  Name = "Test",
38 |              };
39 |
40 |              var Id = repository.Post(obj);
41 |
42 |              Assert.IsNotNull(Id);
43 |
44 |              repository.Delete(Id);
45 |              repository.Get(Id);
46 |              var ObjSupposeToDelete = repository.Get(Id);
47 |              Assert.IsNull(ObjSupposeToDelete);
48 |
49 |          // Tests the Get method of the BrandRepository
50 |          [TestMethod]
51 |          0 références
52 |          public void Test002_getBrand()
53 |          {
54 |
55 |              var objSaved = repository.Get(TestId);
56 |
57 |              Assert.IsNotNull(objSaved);
58 |
59 |              Assert.AreEqual("Test", objSaved.Name);
60 |          }
61 |      }
62 |  }
```

J'ai également effectué mes tests unitaires pour m'assurer que tout fonctionne correctement. Cela m'a permis de vérifier que chaque méthode, comme celles pour ajouter, récupérer, mettre à jour et supprimer des marques, se comporte comme prévu. Les tests garantissent que les fonctionnalités sont robustes et fiables.

The screenshot shows the Visual Studio Test Explorer interface. At the top, there are navigation buttons and a status bar indicating "Série de tests achevée : 3 tests (3 réussi(s), 0 non réussi(s), 0 ignoré(s)) exécutés en 789 ms". Below this is a table with columns: Test, Durée, Caractéris..., and Message d'erreur. The table lists the following test results:

Test	Durée	Caractéris...	Message d'erreur
UnitTests (3)	727 ms		
UnitTest1 (3)	727 ms		
Test001_PostBrand	165 ms		
Test002_getBrand	453 ms		
Test003_updateBrand	109 ms		

On the right side of the interface, there are summary sections: "Récapitulatif du groupe UnitTests" showing "Tests dans le groupe: 3" and "Durée totale: 727 ms"; and "Résultats" showing "3 Réussite".

Tous mes tests unitaires ont réussi, ce qui confirme que chaque méthode fonctionne correctement. Je suis donc rassuré quant à la stabilité et la fiabilité de mon code.

CONCLUSION

Grâce à ce stage, j'ai pu acquérir une expérience en développement web et découvrir les méthodes de travail agiles. J'ai été particulièrement impressionné par la dynamique de l'équipe. C'est pourquoi je suis très heureux de poursuivre mon apprentissage en alternance au sein de cette entreprise.

REMERCIEMENT

Je suis très reconnaissant envers l'équipe d'Ergoos Logicel pour m'avoir offert cette opportunité d'acquérir une expérience pratique . Grâce à eux , j'ai pu développer de nouvelles compétences et consolider mes connaissances, ce qui m'a définitivement convaincu de poursuivre mon parcours professionnel en tant que Concepteur développeur d'applications.