

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2023
Épreuve E5 - Conception et développement d'applications (option SLAM)	
ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle (recto)	

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 1		
Nom, prénom : Damien Monchaty		N° candidat : 02148833006		
Épreuve ponctuelle <input checked="" type="checkbox"/> Contrôle en cours de formation <input type="checkbox"/>		Date : 25/04/2023		
Organisation support de la réalisation professionnelle Dans le cadre d'une demande d'une société de service HRManagement, j'ai dû concevoir une application pouvant gérer des projets et leurs missions et leurs employés.				
Intitulé de la réalisation professionnelle HRManagement				
Période de réalisation : Du 20.10.2022 au 07.01.2023 Lieu : Domicile – Candidat Libre Modalité : <input checked="" type="checkbox"/> Seul(e) <input type="checkbox"/> En équipe				
Compétences travaillées <input checked="" type="checkbox"/> Concevoir et développer une solution applicative <input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative <input checked="" type="checkbox"/> Gérer les données				
Conditions de réalisation¹ (ressources fournies, résultats attendus) Ressources fournies : La société HRManagement voulant avoir un outil de gestion de projets personnalisée décide de demander au service développement web de s'occuper de cette tâche permettant de créer des projets en y ajoutant des missions et des employées. Il n'existait jusqu'à lors aucun code au préalable. Résultat attendu : Quelques fonctionnalités tels que : - Création d'un portail Manager - Employé - Création de projet (Ajout/modification/suppression de projet) - Enregistrement d'employé par système d'e-mail - Système de recherche sur des employés & des projets - Suivi des projets par des statuts mis à jour quotidiennement				
Description des ressources documentaires, matérielles et logicielles utilisées² <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; vertical-align: top;"> - Visual Paradigm Online - Microsoft Office - Microsoft Visual Studio Community 2022 - ElasticSearch </td> <td style="width: 50%; vertical-align: top;"> - GitHub - Trello - MySQL Server Community Management </td> </tr> </table>			- Visual Paradigm Online - Microsoft Office - Microsoft Visual Studio Community 2022 - ElasticSearch	- GitHub - Trello - MySQL Server Community Management
- Visual Paradigm Online - Microsoft Office - Microsoft Visual Studio Community 2022 - ElasticSearch	- GitHub - Trello - MySQL Server Community Management			
Modalités d'accès aux productions³ et à leur documentation⁴ La commission aura la possibilité d'accéder à l'ensemble de la solution à l'emplacement suivant : ... L'identifiant et son mot de passe sont : <ul style="list-style-type: none"> ▪ Identifiant : Admin ▪ Mot de passe : Admin12345! Vous pourrez accéder à la documentation technique à l'url suivante				

¹ En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

² Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

³ Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

⁴ Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

Épreuve E5 - Conception et développement d'applications (option SLAM)

ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)

Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs

Etant Candidat libre et seul sur ce projet, voici toutes les étapes de conception du projet **HRManagement** et sa documentation.

- **Github** : <https://github.com/DamienMonchaty/HRManagement>
- **Trello** : <https://trello.com/b/vR9X1UiX/hrmanagement-web-app>
- **Url** : <http://hrmanagementweb.azurewebsites.net/>

- **Etape 1 : Analyse des besoins du Client**
 - Mise en place de la méthodologie **SCRUM**
 - Création de User-Stories via l'outil en ligne **Trello**

- **Etape 2 : Analyse du cahier des charges selon le besoin du client**
 - Expression du besoin
 - Analyse de la demande
 - Analyse du MPD (modèle physique de données)
 - Conception de Diagramme UML – **Unified Modelling Language**

- **Etape 3 : Conception de la base de données hrman_db**
 - Analyse de la demande
 - Conception de tables dans la base de données
 - Insertion de données

- **Etape 4 : Mise en place Mode Projet**
 - Mise en place d'un répertoire **GitHub**
 - Suivi des User-Stories par le biais de **tâches**

- **Etape 5 : Conception du projet HRManagement en C# et ASP NET CORE**
 - Respect du patron architectural **MVC – Model View Controller**
 - Conception Interface graphique **Razor .cshtml**
 - Mise en place d'**Entity Framework Core – ORM Object Relationnal Mapping**
 - Gestion de la sécurité base sur les rôles avec **Identity**
 - Gestion de la base de données indexée **ElasticSearch**
 - Utilisation de **package NuGet**

- **Etape 6 : Ecriture des tests d'intégrations & unitaires du projet HrManagement avec MsTest**
 - Réalisation de test d'intégration des contrôleurs – Utilisation de **Mock**
 - Réalisation de test unitaires sur la couche DAO Data Access Object – Utilisation de base de données **InMemory**

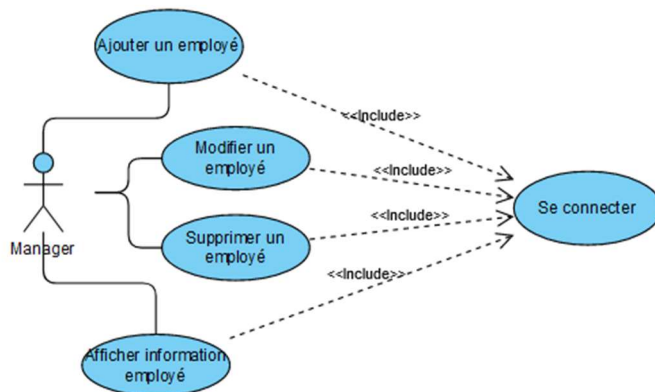
- **Etape 7 : Mise en place de la documentation technique du projet HRManagement**
 - Génération de la documentation technique avec **Doxygen**

- **Etape 8 : Déploiement dans le cloud Microsoft Azure**
 - Mise en place d'un **Serveur MySQL Cloud Azure** dans le cloud **Azure**.
 - Mise en place d'un **Serveur .net6 Cloud Azure** liant son compte **GitHub**

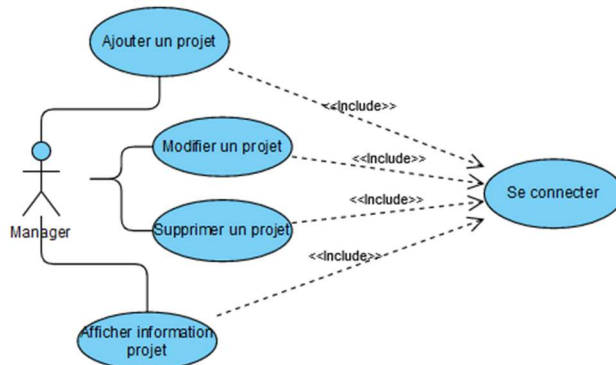
➤ Plusieurs schéma-diagrammes **UML** ont été réalisées :

- Diagrammes de cas d'utilisation

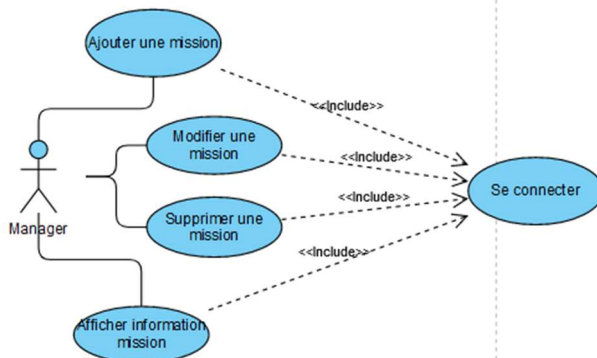
Cas 1 : Ajouter/modifier/supprimer un employé



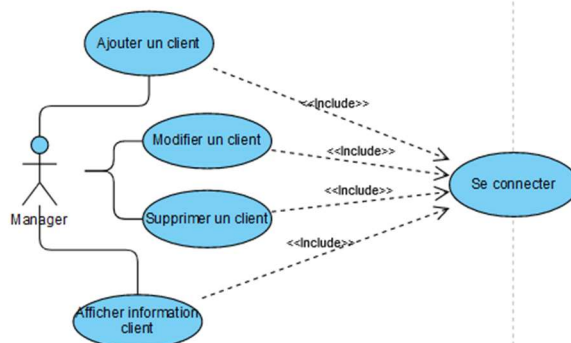
Cas 2 : Ajouter/modifier/supprimer un projet



Cas 3 : Ajouter/modifier/supprimer une mission



Cas 5 : Ajouter/modifier/supprimer un client



Cas 6 : Finaliser une inscription – Valider mission – Rôle Employé

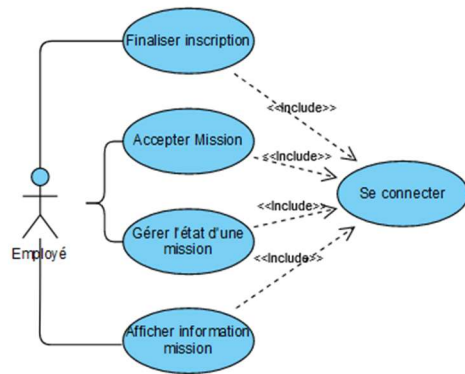


Diagramme de séquence

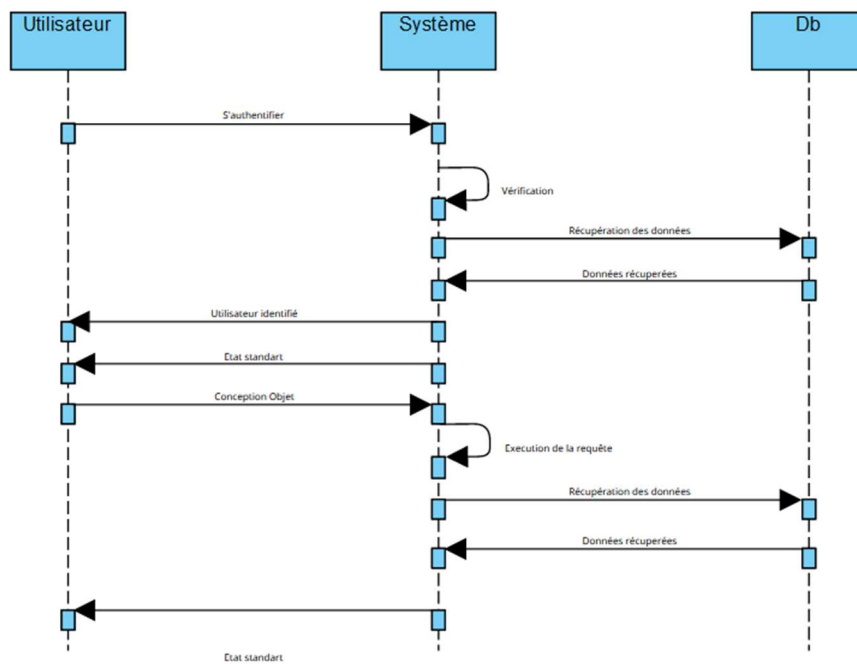
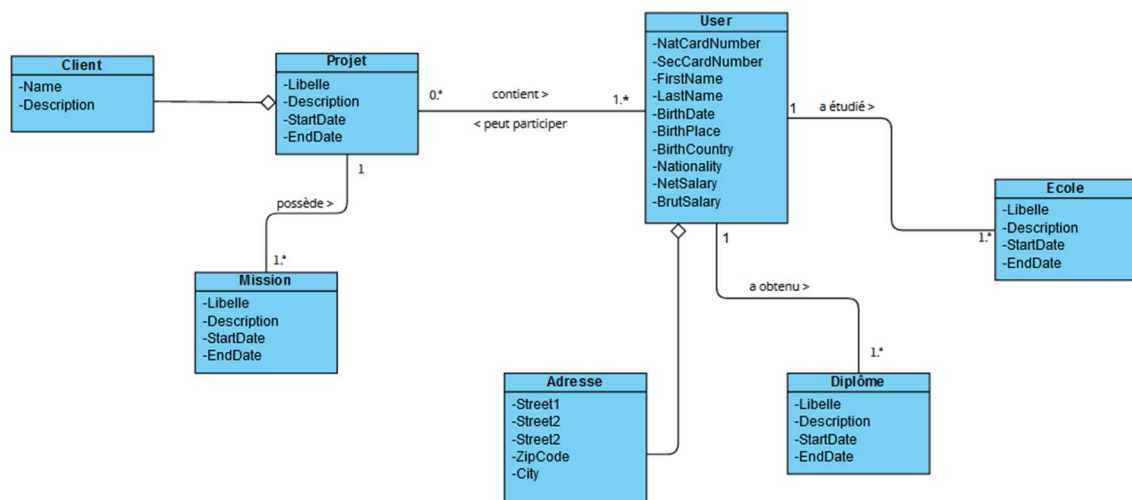
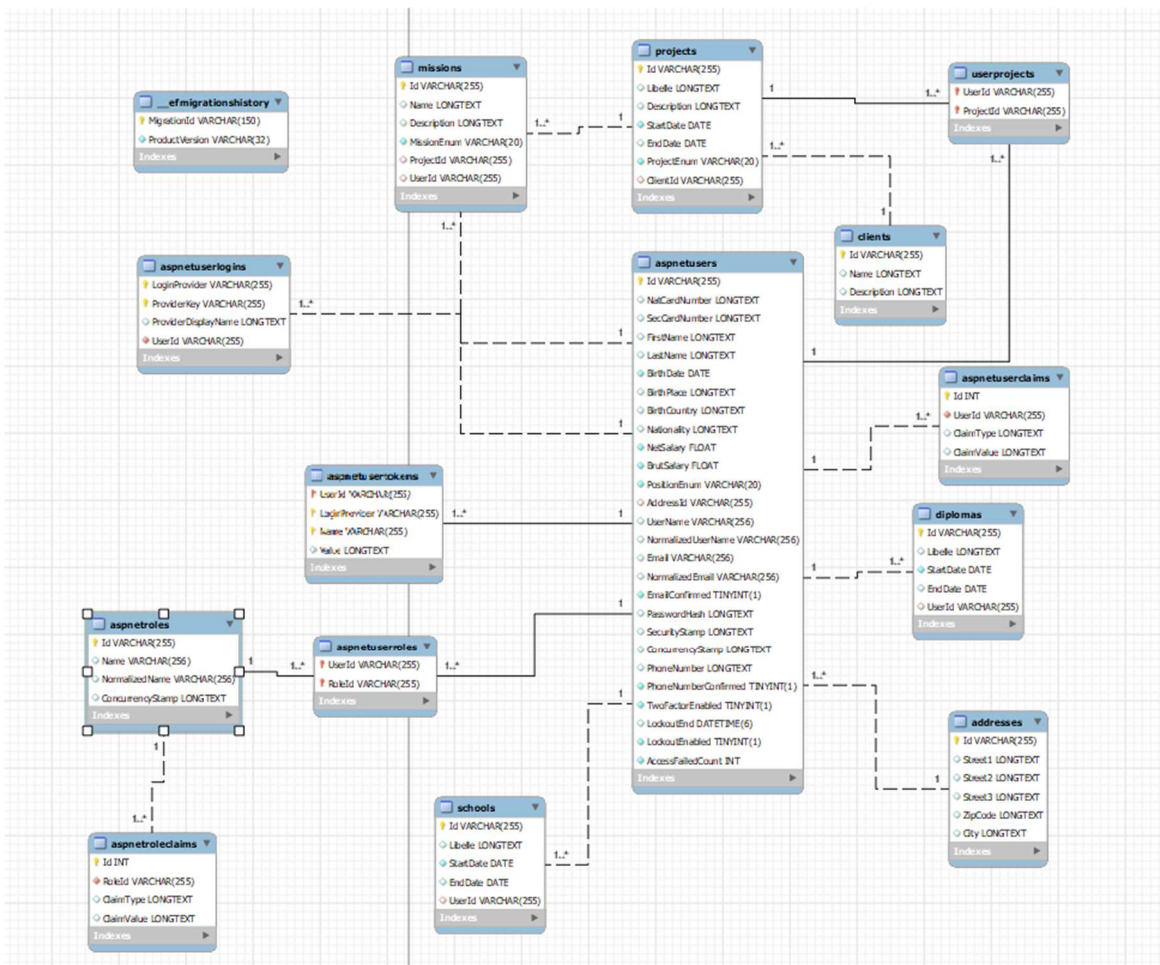


Diagramme de classe














- MLD – Modèle Logique de données



- Les données personnelles sont chiffrées

Id	NatCardNumber	SecCardNumber	FirstName	LastName	BirthDate	BirthPlace
B226988B-42A2-4115-9631-1CD2 IE2AC5F7	IPfWq/4T0c4JqFxmjg99A==	RkNiZnEveo8go...	N6MDRQ5tJg3DV0T5tNw==	JTPS3V0qAyl9L3+KCQ/xsew==	0001-01-01 00:00:00.000000	ly.jyHdCKO...
B226788B-42A2-4115-9631-1CE5 IE2AC5F7	IPfWq/4T0c4JqFxmjg99A==	RkNiZnEveo8go...	5x08t59qSk1t5qL+hF+kQ=	3Q+u+AAcBAcHtBSj3VWw==	0001-01-01 00:00:00.000000	ly.jyHdCKO...

- Ecriture des tests d'intégrations & unitaires du projet HrManagement avec MsTest

Test	Durée
▶  HRManagement.Tests (29)	4.3 s
▶  HRManagement.Tests.Controllers (12)	566 ms
▶  MissionControllerTest (4)	379 ms
 TestAdd	333 ms
 TestDelete	10 ms
 TestEdit	6 ms
 TestGetAllMissions	30 ms
▶  ProjectControllerTest (8)	187 ms
▶  HRManagement.Tests.Repository (17)	3.7 s
▶  MissionRepositoryTest (8)	3.5 s
▶  ProjectRepositoryTest (9)	230 ms

- La classe **ProjectRepositoryTest.cs**

Pour pouvoir tester cette classe, j'ai mis en place une base de données **InMemory** via le **nugget Package EntityFrameworkCore.InMemory**.

De plus, je charge les données avant chaque test, grâce au décorateur **[TestInitialize]** pour pouvoir tester correctement

```
[TestClass]
public class ProjectRepositoryTest
{
    [TestInitialize]
    public void Init()
    {
        var db = GetMemoryContext();
        db.Database.EnsureDeleted();
        db.Projects.Add(new Project { Id = "1", Libelle = "Lib1", Description = "Desc1", StartDate = new DateTime(1996, 01, 31), EndDate = new DateTime(1996, 01, 31) });
        db.Projects.Add(new Project { Id = "2", Libelle = "Lib2", Description = "Desc2", StartDate = new DateTime(1996, 01, 31), EndDate = new DateTime(1996, 01, 31) });
        db.Users.Add(new User { Id = "1", UserName = "User1" });
        db.Users.Add(new User { Id = "2", UserName = "User2" });
        db.UserProjects.Add(new UserProject { UserId = "1", ProjectId = "1" });
        db.UserProjects.Add(new UserProject { UserId = "2", ProjectId = "1" });
        db.SaveChanges();
    }

    [TestMethod]
    public async Task TestAdd()
    {
        // Arrange
        ProjectRepository repository = new ProjectRepository(GetMemoryContext());

        var projectToAdd = new Project()
        {
            Libelle = "Lib3",
            Description = "Desc3",
            StartDate = new DateTime(1996, 01, 31),
            EndDate = new DateTime(1996, 01, 31)
        };
        // Act
        var projectAdded = await repository.Add(projectToAdd);

        // Assert
        Assert.AreEqual(projectToAdd, projectAdded);
    }
}
```

- La classe **ProjectControllerTest.cs**

Dans la méthode de test **TestGetAllProjects()**, j'ai mis en place un **mock**, une **implémentation factice** pour pouvoir tester la méthode **GetAllProjects()** du contrôleur **PersonneController.cs**

GetAllProjects() retourne une liste de projets paginé

```
[TestMethod]
public void TestGetAllProjects()
{
    // Arrange
    var projects = new List<Project> { new Project(), new Project() };
    IPagedList<Project> ipagedListProjects = projects.ToPagedList(1, 5);

    _mockProjectRepo.Setup(x => x.GetAll(1)).Returns(ipagedListProjects);

    var actionResult = _controller.GetAllProjects(1);
    var result = actionResult as PartialViewResult;

    // Assert
    Assert.AreEqual(@"~/Views/Shared/_Projects.cshtml", result.ViewName);
    Assert.IsInstanceOfType(result.ViewData.Model, typeof(IPagedList<Project>));
}

[TestMethod]
public async Task TestGetAllProjectsByUserId()...
```

```
[TestMethod]
public async Task TestGetProject()...
```

➤ Productions réalisées :

▪ Dashboard – Vue **Manager**

The screenshot shows the HRManagement dashboard for a user named 'p1@p.fr'. The interface includes a top navigation bar with a search bar and a 'Rechercher' button. A sidebar on the left contains links for 'Synthèse', 'Employé', 'Projet', 'Client', 'Mission', and 'Profile'. The main content area features a large 'Dashboard' header and four colored boxes representing different categories: 'EMPLOYÉS' (3), 'PROJETS' (2), 'MISSIONS' (2), and 'CLIENTS' (2). Below these boxes is a table titled 'Nos Employés' with columns for 'Nom', 'Prénom', 'Position', 'Projets', 'Missions', and 'Actions'. The table lists three employees: 'prenom2' (MANAGER), 'prenom1' (MANAGER), and an unnamed employee (DEVELOPER).

Nom	Prénom	Position	Projets	Missions	Actions
prenom2	nom2	MANAGER	• Projet Web • Projet SQL	• M1 - Architecture & Besoins • M2 - Sécurité	✎
prenom1	nom1	MANAGER	Sans Projets	Sans Missions	✎
		DEVELOPER	Sans Projets	Sans Missions	✎

▪ Formulaire inscription Employé

The screenshot shows the 'Nouvel employé' registration form. The form includes fields for 'Email', 'BrutSalary', 'NetSalary', 'Position', and 'Role'. The 'Email' field is filled with 'damienmonchaty@hotmail.fr'. The 'BrutSalary' field is filled with '2500'. The 'NetSalary' field is filled with '1875'. The 'Position' dropdown is set to 'DEVELOPER'. The 'Role' dropdown is set to 'Visitor'. At the bottom of the form are two buttons: 'Envoyer' (green) and 'Annuler' (red).

▪ Email reçu pour inscription

Bienvenue chez HRManagement

[Commencer votre inscription](#)

Votre Mot de passe : Test1234!

[← Répondre](#) [→ Transférer](#)

▪ Page de confirmation

The screenshot shows the confirmation page after registration. The interface includes a top navigation bar with a search bar and a 'Rechercher' button. A sidebar on the left contains links for 'Synthèse' and 'Profile'. The main content area features a large 'Bienvenue damienmonchaty@hotmail.fr' header and a button labeled 'Commencer votre inscription'. A green notification bar at the top states 'Félicitations Email confirmé avec succès !'.

■ Formulaire inscription Employé

Partie **Informations** :

Inscription ?

1

2

3

InformationsDiplomesEntreprise

Vos informations

Email

damienmonchaty@hotmail.fr

Mot de passe

••••••••

Confirmer mot de passe

Nom

Prénom

Date d'anniversaire

0001-01-01T00:00:00.000

Lieu de naissance

Votre adresse

Rue 1

Rue 2

Rue 3

Code postal

Ville

Next

Partie **Diplômes** :

Inscription ?

1

2

3

InformationsDiplomesEntreprise

Poste

Poste

DEVELOPER

CNI / Sécurité Sociale

Numéro de carte d'identité

Numéro de carte de sécurité sociale

Previous

Parcours

+

Libelle

Date de début

Date de fin

Mes diplômes

+

Libelle

Date de début

Date de fin

Next

Partie **Entreprise** :

Inscription ?

1

2

3

InformationsDiplomesEntreprise

Votre salaire

Salaire Brut

2500

Salaire Net

1875

Autres ...

Previous

Envoyer ➕

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2023
Épreuve E5 - Conception et développement d'applications (option SLAM)	
ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle (recto)	

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 2								
Nom, prénom : Damien Monchaty		N° candidat : 02148833006								
Épreuve ponctuelle <input checked="" type="checkbox"/>	Contrôle en cours de formation <input type="checkbox"/>	Date : 25/04/2023								
Organisation support de la réalisation professionnelle Dans le cadre d'une demande d'un site d'actualité de sport automobile FormulaNews, j'ai dû concevoir une application mobile diffusant les résultats de toutes les courses de Formule 1 depuis 1950. Elle permettra de plus d'avoir des informations sur le championnat en cours en termes de pilotes, constructeurs & classement										
Intitulé de la réalisation professionnelle FormulaNews										
Période de réalisation : Du 24.01.2022 au 07.04.2023 Lieu : Domicile – Candidat Libre Modalité : <input checked="" type="checkbox"/> Seul(e) <input type="checkbox"/> En équipe										
Compétences travaillées <input checked="" type="checkbox"/> Concevoir et développer une solution applicative <input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative <input checked="" type="checkbox"/> Gérer les données										
Conditions de réalisation⁵ (ressources fournies, résultats attendus) Ressources fournies : Le site d'actualité FormulaNews aimerait avoir une application mobile diffusant plusieurs actualités concernant la formule 1 pour gagner en audience. Il n'existait jusqu'à lors aucun code au préalable. Résultat attendu : Quelques fonctionnalités tels que : - Information sur la prochaine course & dernière course - Consultation d'article – Mise en place d'un blog - Consultation des résultats de toutes les courses de Formule 1 depuis 1950 : - Classement Pilotes & Constructeurs - Toutes les courses et leurs résultats - Tous les pilotes et leurs résultats - Une utilisation facile & intuitive de l'application										
Description des ressources documentaires, matérielles et logicielles utilisées⁶ <table border="0"> <tr> <td>- Visual Paradigm Online</td> <td>- GitHub</td> </tr> <tr> <td>- Microsoft Office</td> <td>- Trello</td> </tr> <tr> <td>- Microsoft Visual Studio Community 2022</td> <td>- Visual Studio Code</td> </tr> <tr> <td>- SQLite</td> <td></td> </tr> </table>			- Visual Paradigm Online	- GitHub	- Microsoft Office	- Trello	- Microsoft Visual Studio Community 2022	- Visual Studio Code	- SQLite	
- Visual Paradigm Online	- GitHub									
- Microsoft Office	- Trello									
- Microsoft Visual Studio Community 2022	- Visual Studio Code									
- SQLite										
Modalités d'accès aux productions⁷ et à leur documentation⁸ La commission aura la possibilité d'accéder à l'ensemble de la solution à l'emplacement suivant : ... Vous pourrez accéder à la documentation technique à l'url suivante										

⁵ En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

⁶ Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

⁷ Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

⁸ Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

Épreuve E5 - Conception et développement d'applications (option SLAM)

ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)**Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

Etant Candidat libre et seul sur ce projet, voici toutes les étapes de conception du projet FormulaNews et sa documentation.

- Github – FormulaNews-Web : <https://github.com/DamienMonchaty/FormulaNews-Web>
- Github : – FormulaNews-Mobile : <https://github.com/DamienMonchaty/FormulaNews>
- Trello : <https://trello.com/b/Y39APC9A/formulanews-web-app-mobile>
- Url – FormulaNews-Web :
- Url – FormulaNews-Mobile (téléchargement) :

- Etape 1 : **Analyse des besoins du Client**
 - Mise en place de la méthodologie **SCRUM**
 - Création de User-Stories via l'outil en ligne **Trello**

- Etape 2 : **Analyse du cahier des charges selon le besoin du client**
 - Expression du besoin
 - Analyse de la demande
 - Analyse du **MPD - Modèle Physique de Données**
 - Conception de Diagramme **UML - Unified Modelling Language**

- Etape 3 : Mise en place **Mode Projet**
 - Mise en place d'un répertoire **GitHub**

- Etape 3 : Conception du projet **FormulaNews Mobile en C# et Xamarin Forms**
 - Respect du patron architectural **MVVM – Model View ViewModel**
 - Conception Interface graphique **XAML**
 - Utilisation de **divers packages NuGet**
 - Mise en place d'**Entity Framework Core – ORM Object Relational Mapping**
 - Mise en place d'**Entity Framework SQLite**

- Etape 3 : Conception du projet **FormulaNews Web en Angular NodeJs**
 - Mise en place de **Composants Web**
 - Conception Interface graphique **HTML5, CSS3**

- Etape 4 : Ecriture des tests UI avec **Xamarin.UITest**
 - Ecriture de tests UI des interfaces graphiques de l'application mobile **FormulaNews**

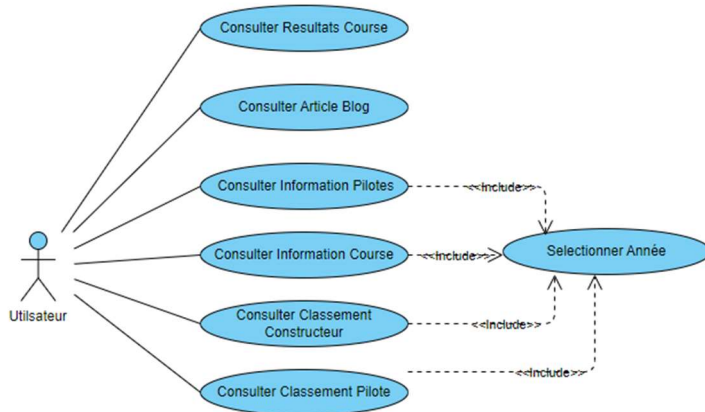
- Etape 5 : Mise en place de la documentation technique du projet **FormulaNews**
 - Génération de la documentation technique avec **Doxygen**

- Etape 6 : Mise en production de l'application Mobile Android **FormulaNews**
 - Packaging & Versioning de l'application
 - Protection de l'application
 - Compilation, Archive & distribution

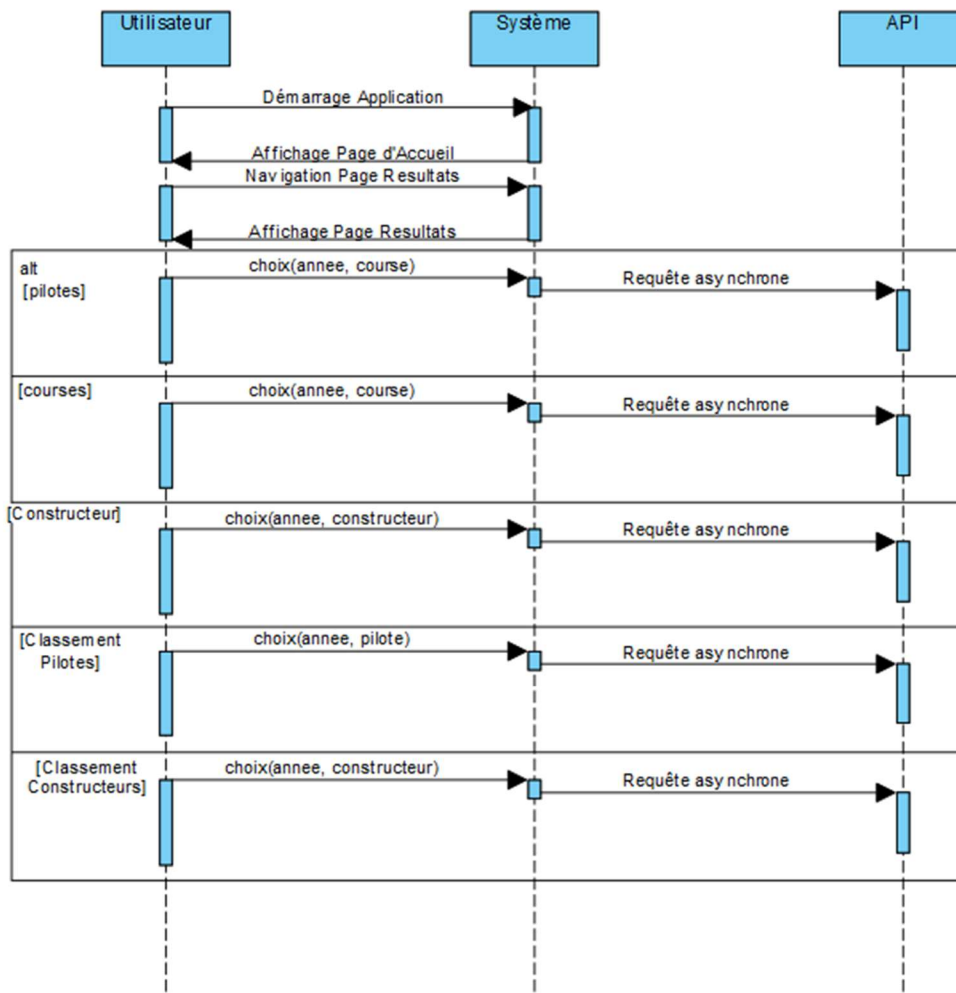
- Etape 7 : Mise en production de l'application Web **FormulaNews**
 - Mise en place d'un **Serveur NodeJs Cloud Azure** en liant son compte **GitHub**

➤ Plusieurs schéma-diagrammes UML ont été réalisées :

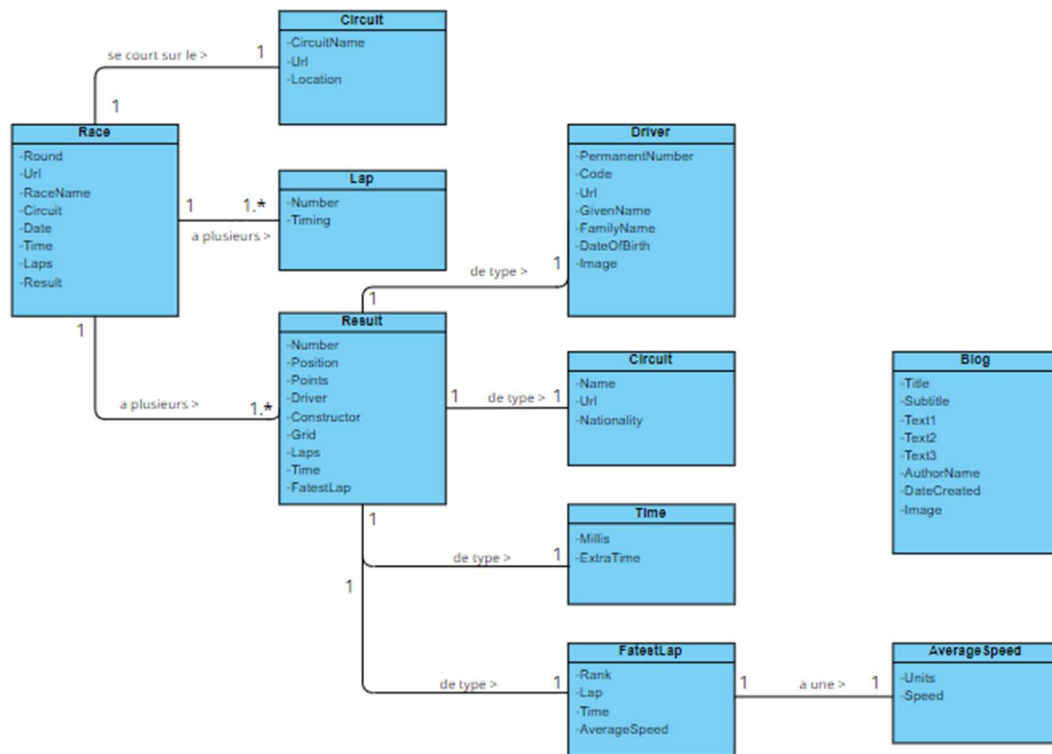
- Diagramme de cas d'utilisation



- Diagramme de séquence



- Diagramme de classes



➤ Gestion des données :

Pour gérer des données en **Xamarin Forms**, j'ai mis en place une base de données locale **SQLite**

Voici les étapes :

- Pour utiliser SQLite dans XF, il est nécessaire d'installer le paquet **NuGet sqlite-net-pcl**.
- Ecrire un code spécifique pour chaque plateforme iOS & Android

```

public interface ISQLiteDb
{
    SQLiteAsyncConnection GetConnection();
}

```

- Cette interface expose une méthode appelée **GetConnection()** qui sera implémentée dans chaque projet spécifique à la plate-forme et renverra la chaîne de connexion appropriée.

```

using Environment = System.Environment;
[assembly: Dependency(typeof(SQLiteDb))]
namespace FOne_MobileApp.Droid.Persistence
{
    public class SQLiteDb : ISQLiteDb
    {
        public SQLiteAsyncConnection GetConnection()
        {
            var documentPath = Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments);
            var path = Path.Combine(documentPath, "MySQLite.db");
            return new SQLiteAsyncConnection(path);
        }
    }
}

```

- La classe BlogDao qui permet de récupérer des informations en base de données

```
public class BlogDao : IDao<Blog>
{
    private SQLiteAsyncConnection _connection;

    public BlogDao(ISQLiteDb db)
    {
        _connection = db.GetConnection();
    }

    public async Task<Blog> GetFirst()
    {
        return await _connection.Table<Blog>().FirstAsync();
    }

    public async Task<IEnumerable<Blog>> GetListAsync()
    {
        return await _connection.Table<Blog>().ToListAsync();
    }
}
```

- Pour insérer des données au démarrage de l'application, j'ai mis en place dans le projet **Android** une méthode **CreateDatabase()** dans la classe **MainActivity.cs**

```
[Activity(Label = "FormulaNews", Icon = "@drawable/logoicon", Theme = "@style/splashscreen", MainLauncher = true,
public class MainActivity : global::Xamarin.Forms.Platform.Android.FormsAppCompatActivity
{
    SQLiteConnection dataBase;

    protected override void OnCreate(Bundle savedInstanceState)
    {
        base.OnCreate(savedInstanceState);
        this.CreateDatabase();
        Xamarin.Essentials.Platform.Init(this, savedInstanceState);
        global::Xamarin.Forms.Forms.Init(this, savedInstanceState);
        LoadApplication(new App());
    }

    public override void OnRequestPermissionsResult(int requestCode, string[] permissions, [GeneratedEnum] Android
    {
        Xamarin.Essentials.Platform.OnRequestPermissionsResult(requestCode, permissions, grantResults);

        base.OnRequestPermissionsResult(requestCode, permissions, grantResults);
    }

    private void CreateDatabase()
    {
        dataBase = this.GetConnection();
        dataBase.CreateTable<Blog>();

        dataBase.Query<Blog>("DELETE From Blog");
    }
}
```

➤ Ecriture des tests UI avec **Xamarin.UITest**

Xamarin.UITest est un framework de test C# utilisant **NUnit** pour les tests d'acceptation de l'interface utilisateur sur les applications iOS et Android.

Les tests interagissent avec l'interface utilisateur comme le ferait un utilisateur - en saisissant du texte, en appuyant sur des boutons et des gestes - tels que des balayages.

Test	Durée	C.	Me...
▲ FOne_MobileApp.UITest (3)	1.5 min		
▲ FOne_MobileApp.UITest (3)	1.5 min		
▲ Tests(Android) (3)	1.5 min		
✔ BlogsDisplayed	33.5 s		
✔ BlogsDisplayedFromHomeView	23.5 s		
✔ ResultsDisplayed	31 s		

- La classe **Tests.cs**

```
[TestFixture(Platform.Android)]
public class Tests
{
    IApp app;
    Platform platform;

    public Tests(Platform platform)
    {
        this.platform = platform;
    }

    [SetUp]
    public void BeforeEachTest()
    {
        app = AppInitializer.StartApp(platform);
        app.Repl();
    }

    [Test]
    public void BlogIsDisplayed()
    {
        app.ScrollDown();
        app.WaitForElement(c => c.Marked("frame_blogDetails"));
        app.Tap(c => c.Marked("frame_blogDetails"));
        app.Screenshot("On " + this.GetType().Name);
    }

    [Test]
    public void BlogIsDisplayedFromHomeView()...

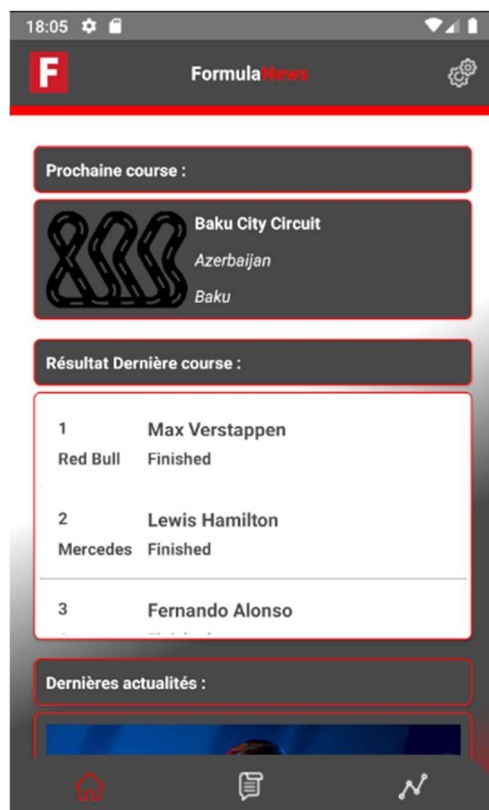
    [Test]
    public void ResultIsDisplayed()...
}
```

➤ Productions réalisées :

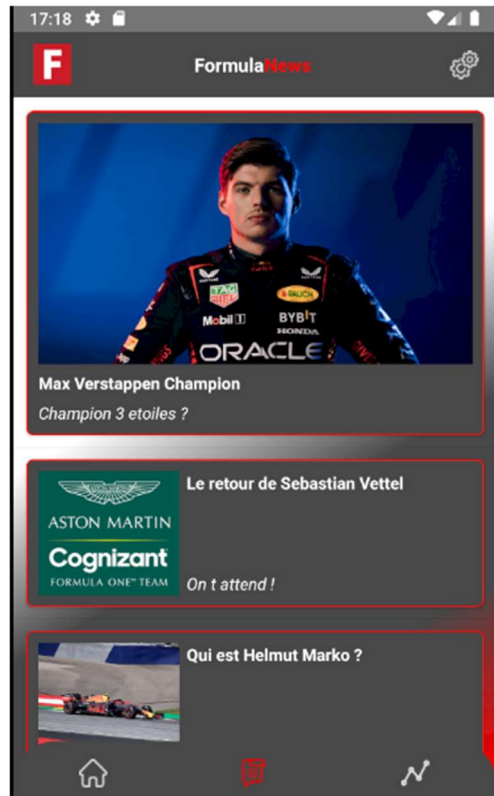
Les vues ont été conçues en **XAML - Extensible Markup Language**.

Il s'agit d'un langage de balisage déclaratif utilisé par les développeurs pour définir les interfaces utilisateur des applications **WPF & Xamarin Forms**.

- Home



▪ Blog - Blog Détails



▪ Resultats Année 2023 - Pilote Détails

