



Concepteur Développeur d'Applications

Projet Fil Rouge FoodBook

Activité 3 - Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité



SOMMAIRE

I. Présentation	3
II. Cheminement	4
II.1 Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement	4
II.2 Concevoir une application	5
II.3 Développer des composants métier	7
II.4 Construire une application organisée en couches	8
II.5 Développer une application de mobilité numérique	10
II.6 Préparer et exécuter les plans de tests d'une application	12
II.7 Préparer et exécuter le déploiement d'une application	13
II.8 Utiliser l'anglais dans son activité professionnelle en conception et développement d'applications	14
II.9 Actualiser et partager ses compétences en conception et développement d'applications	14
III. Le Cahier des charges	15
III.1 Contexte fonctionnel et technique de l'application	15
III.2 Les données à traiter	15
III.3 Utilisateurs de l'application et gestion de la sécurité	16
III.4 Environnement technique	16
III.5 Architecture retenue	17
III.5.1 Les interfaces	17
III.5.2 La BLL(Business Logic Layer) Client	17
III.5.3 L'API REST et Web Socket	17
III.5.4 BLL(Business Logic Layer) Serveur	17
III.5.5 DAL(Data Access Layer)	17
IV. Livrables et modalités pour l'évaluation des compétences	18

I. PRESENTATION

Ce projet a 2 objectifs principaux :

- Apporter de la cohérence aux apprentissages puisqu'il met en jeu toutes les compétences de l'emploi visé
- Proposer **une mise en situation professionnelle** pour vous permettre de **démontrer vos compétences lors des Evaluations en Cours de Formation**.

Le projet fil rouge se déroule **en parallèle des apprentissages** de la formation Concepteur Développeur d'Applications, il est le support des **Evaluations en Cours de Formation**.

Vous travaillerez à partir du **cahier des charges** fourni dans ce dossier. Vous devrez planifier votre projet et faire un état d'avancement régulier avec vos formateurs.

La production et les résultats que vous réaliserez dans ce cadre seront évalués à la fin de l'activité. Vous trouverez à la fin de ce dossier **les modalités pour l'évaluation des compétences**.

Les solutions mises en œuvre ainsi que leurs réalisations doivent être personnelles et couvrir l'ensemble des compétences attendues du titre CDA.

Le projet fil rouge étant un exemple de votre pratique professionnelle, vous alimenterez votre **Dossier Professionnel**.

II. CHEMINEMENT

Vous travaillerez à partir du cahier des charges fourni. Vous vous organiserez pour vous assurer une bonne compréhension du besoin. Un forum est mis à disposition du groupe sur la plateforme [amio-fit](#) pour centraliser vos questions et les soumettre à l'équipe des formateurs qui jouera le rôle de la Maîtrise d'Ouvrage.

Le projet est à réaliser individuellement, la restitution sera individuelle, pour autant, les échanges entre stagiaires sont encouragés.

Au fur et à mesure de vos apprentissages, vous mettrez en œuvre les compétences et les savoir-faire suivants.

II.1 Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement

À partir des objectifs du projet définis en terme de livrables et de la démarche projet, collaborer à la planification et au suivi des ressources humaines et matérielles allouées au projet, en communiquant à l'écrit ou à l'oral en français ou en anglais, mettre en œuvre les procédures qualité décrites dans le plan qualité projet, définir l'environnement de développement et les outils collaboratifs du travail en équipe, afin d'atteindre les objectifs du projet en termes de coût, de délai et de qualité.

► Contexte professionnel de mise en œuvre

Cette compétence s'exerce au sein d'une équipe projet éventuellement sous la responsabilité d'un chef de projet et dans un environnement de travail organisé selon la méthode choisie.

Le degré de responsabilité du concepteur développeur varie selon la taille et la complexité du projet. L'organisation du travail dépend de la méthode de développement choisie, séquentielle ou itérative. Dans un développement de type séquentiel, cette organisation est structurée de façon hiérarchique, tandis que dans un développement itératif en approche de type Agile, l'organisation est structurée de façon plus collaborative. Selon les projets, la communication écrite et orale au sein des équipes de développement peut s'effectuer en anglais.

► Savoirs, savoir-faire techniques, savoir-faire relationnels, savoir-faire organisationnels

S'assurer que les documents produits en français ou en anglais respectent les règles orthographiques et grammaticales

Mettre en œuvre les procédures de la démarche qualité

Définir l'environnement de développement du projet

Définir un outil collaboratif de partage de ressources

Définir les outils du Cloud à utiliser au sein de l'équipe de projet

Participer à la planification et au suivi du projet au sein de l'équipe de projet

Sécuriser les échanges dans l'équipe de projet

Conduire de façon séquentielle un projet de conception et de développement

Coordonner de façon itérative et en mode collaboratif un projet de conception et de développement

Rédiger un compte rendu de réunion

Planifier et suivre les tâches de développement d'une application informatique

Rédiger des rapports d'activité et de reste à faire, éventuellement en anglais

Animer une réunion de travail au sein de l'équipe de projet
Connaissance des règles orthographiques et grammaticales du français et de l'anglais
Connaissance des différents types de démarches de conception de logiciel
Connaissance des outils de planification
Connaissance des méthodologies de découpage de projets en itérations, d'estimation de complexité et de charge, de suivi en temps réel
Connaissance des différents types de Cloud, des outils associés et de leur utilité pour le développement
Connaissance des outils de virtualisation ou de conteneurisation
Connaissance des outils collaboratifs et de leurs vulnérabilités

► **Environnements techniques**

Logiciel de gestion de versions : Git

► **Critères d'évaluation**

Le suivi des activités ou des tâches du projet est mis en œuvre en fonction de la démarche projet adoptée
Les procédures qualité décrites dans le plan qualité projet sont mises en œuvre
L'environnement de développement est défini
Les outils collaboratifs sont choisis
La communication écrite en français ou en anglais est rédigée de façon adaptée à l'interlocuteur et sans faute
La communication technique orale, en français ou en anglais, est réalisée de façon adaptée à l'interlocuteur et sans faute

II.2 Concevoir une application

À partir du cahier des charges de la maîtrise d'ouvrage, concevoir fonctionnellement et techniquement une application informatique sécurisée en suivant une démarche de conception. Constituer le dossier de conception, éventuellement en anglais, modélisant, avec des diagrammes, les cas d'utilisation, les classes d'analyse et de conception, décrivant également l'architecture logicielle multicouche répartie en vue du développement de l'application.
Respecter les bonnes pratiques et les règles du développement sécurisé et compléter ce dossier par la description des exigences de sécurité de l'application.
Pratiquer une veille technologique, y compris en anglais, pour résoudre un problème technique ou mettre en œuvre une nouvelle fonctionnalité ainsi que pour s'informer sur la sécurité informatique et les vulnérabilités connues.
Partager le résultat de sa veille avec ses pairs.

► **Contexte professionnel de mise en œuvre**

La conception de l'application pour la partie fonctionnelle se réalise en relation directe avec le client pour les petits projets, ou bien avec une maîtrise d'ouvrage pour les moyens et gros projets. Pour les projets complexes, la conception de la partie technique se fait en relation avec un architecte logiciel. Pour une application sensible, la conception se fait en relation avec un expert sécurité. La modélisation peut être réalisée à l'aide d'un outil de conception et dans le cadre d'une démarche pilotée par les modèles.

► **Savoirs, savoir-faire techniques, savoir-faire relationnels, savoir-faire organisationnels**

Collecter les besoins des utilisateurs
Prendre en compte une analyse de risque
Analyser un cahier des charges en identifiant les limites du système, les acteurs et les messages
Identifier les besoins de sécurité de l'application
Traduire les besoins en diagrammes UML
Concevoir la solution à partir des diagrammes UML
Adapter l'architecture technique aux besoins des utilisateurs et aux besoins de sécurité
Utiliser les patrons de conception (*design patterns*) et les patrons de sécurité (*security pattern*)
Modéliser l'architecture du système, en respectant les principes et patrons de sécurité
Améliorer à fonctionnalités constantes la conception d'un code existant (*refactoring*)
Pratiquer la conception responsable, en minimisant l'impact de l'application sur la consommation énergétique
Rédiger les comptes rendus de réunion, éventuellement en anglais
Rédiger un dossier de conception, éventuellement en anglais
Planifier les tâches de conception d'une application
Suivre l'avancement des tâches de conception
Établir des rapports d'activité et de reste à faire
Écouter, reformuler et synthétiser les demandes utilisateurs, éventuellement en anglais
Participer à une réunion de travail avec les utilisateurs, éventuellement en anglais
Connaissance des architectures logicielles multicouche réparties y compris avec des microservices
Connaissance du formalisme des diagrammes du langage de modélisation unifié UML
Connaissance du développement objet
Connaissance d'une démarche de développement logiciel pilotée par les modèles
Connaissance des outils principaux de conception et de réalisation du marché tels que logiciels et cadres d'applications
Connaissance des principes de l'analyse de risque
Connaissance des principales vulnérabilités et attaques contre les architectures multicouche réparties
Connaissance des principes et des patrons de sécurité applicables à l'architecture multicouche répartie
Connaissance des principes de la conception responsable de services numériques

► **Environnements techniques**

Langage UML
Outil de modélisation

► **Critères d'évaluation**

Les cas d'utilisation couvrent l'ensemble des exigences utilisateur exprimées dans le cahier des charges
Les besoins de sécurité de l'application sont identifiés
Les besoins d'écoconception de l'application sont identifiés
Les classes d'analyse et de conception sont définies
L'architecture technique est conforme aux bonnes pratiques d'une architecture répartie sécurisée
Le dossier de conception est structuré et documenté en conformité avec la démarche choisie
Le rôle de chaque couche dans la stratégie de sécurité est bien défini

L'objet de la recherche est exprimé de manière précise en langue française ou anglaise
La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité
La documentation technique liée aux technologies associées, en français ou en anglais, est comprise (sans contre-sens, ...)
Le partage du résultat de veille est effectué oralement ou par écrit avec ses pairs

II.3 Développer des composants métier

À partir des diagrammes de classes et des diagrammes de composants du dossier de conception, coder et tester les composants de la couche métier d'une application informatique dans une architecture multicouche répartie conformément aux règles métier et avec les cas d'exception, dans le respect des bonnes pratiques du développement objet et des principes du développement sécurisé. Pratiquer une veille technologique, y compris en anglais, pour résoudre un problème technique ou mettre en œuvre une nouvelle fonctionnalité ainsi que pour s'informer sur la sécurité informatique et les vulnérabilités connues.
Partager le résultat de sa veille avec ses pairs.

► Contexte professionnel de mise en œuvre

Le développement des composants s'effectue à partir d'un environnement de développement intégré supportant un langage objet ainsi que les cadres d'applications (*framework*) et les bibliothèques associés à l'architecture choisie.

► Savoirs, savoir-faire techniques, savoir-faire relationnels, savoir-faire organisationnels

Utiliser les fonctionnalités de génération de code de l'outil de modélisation UML
Coder des composants dans un langage objet, avec un style de programmation défensif
Valider la sécurité et utiliser des composants issus d'un cadre d'applications (*framework*) ou d'une bibliothèque
Utiliser les moyens de gestion des identités et des certificats numériques
Utiliser un outil collaboratif de partage de ressources
Produire les tests unitaires, avec une double approche fonctionnelle et sécurité
Utiliser les outils d'analyses statiques (par exemple, ceux intégrés au compilateur) et dynamiques (par exemple débogueur) de code
Corriger les erreurs et remédier aux vulnérabilités détectées
Planifier et suivre les tâches de développement
Rechercher une réponse pertinente à une difficulté technique de développement ou à une vulnérabilité
Rechercher des informations sur des sites Internet, des forums et des FAQ francophones ou anglophones
Contribuer à la mise à jour des bases de connaissances francophones ou anglophones accessibles par Internet
Documenter les composants
Connaissance du vocabulaire anglais professionnel associé au développement de composants métier
Connaissance des concepts et d'un langage de développement objet
Connaissance des principes et des règles du développement sécurisé
Connaissance du formalisme des diagrammes du langage de modélisation unifié UML
Connaissance de l'architecture applicative multicouche répartie et des différents types de serveurs

Connaissance des architectures logicielles multicouche réparties
Connaissance du rôle de la couche métier dans la sécurité d'une application multicouche répartie
Connaissance de techniques de communication applicative telles que les services Web
Connaissance des bases de la cryptographie
Connaissance des bonnes pratiques de conception et de sécurisation des composants dans une architecture logicielle multicouche répartie

► **Environnements techniques**

EDI : Visual Studio
Langage objet : C#

► **Critères d'évaluation**

Les bonnes pratiques de conception objet sont appliquées
Les composants métier sont écrits dans un style défensif et toutes les entrées sont validées
Les composants métier sont conformes aux diagrammes de classes et de composants
Les règles métier sont implémentées dans les classes métier avec les cas d'exception
Les tests unitaires des composants métier sont prévus (approche fonctionnelle, structurelle et sécurité)
L'objet de la recherche est exprimé de manière précise en langue française ou anglaise
La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité
La veille sur les vulnérabilités connues permet d'identifier et corriger des failles potentielles
La documentation technique liée aux technologies associées, en français ou en anglais, est comprise (sans contre-sens, ...)
Le partage du résultat de veille est effectué oralement ou par écrit avec ses pairs

II.4 Construire une application organisée en couches

À partir du dossier de conception, de la stratégie de sécurité de l'application, et d'une architecture applicative multicouche répartie, coder et tester les composants des couches présentation et persistance, puis les intégrer avec les composants métiers développés par ailleurs, afin de construire une application informatique dans le respect de la charte graphique, des bonnes pratiques du développement objet et de l'état de l'art de la sécurité informatique.

► **Contexte professionnel de mise en œuvre**

Le développement des composants s'effectue à partir d'un environnement de développement intégré, supportant un langage objet ainsi que les cadres d'applications (framework) et les bibliothèques associées à l'architecture choisie. Les tests d'intégration des composants s'effectuent dans une architecture de serveurs répartie et éventuellement dans un processus d'intégration continue.

► **Savoirs, savoir-faire techniques, savoir-faire relationnels, savoir-faire organisationnels**

Développer des composants dans un langage objet, avec un style défensif
Utiliser des composants tiers et en vérifier l'efficacité et la sécurité
Utiliser une technologie de services distants
Sécuriser chacune des couches logicielles et la communication entre les couches
Utiliser l'environnement de développement, y compris en anglais, et les outils associés pour organiser le développement
Utiliser un outil collaboratif de partage de ressources
Utiliser les outils du Cloud de type PAAS (Platform As A Service)
Planifier et suivre les tâches de développement
Rechercher une réponse pertinente à une difficulté technique de développement ou une vulnérabilité
Rechercher des informations sur des sites Internet, des forums et des FAQ francophones ou anglophones
Contribuer à la mise à jour des bases de connaissances francophones ou anglophones accessibles par Internet
Connaissance du vocabulaire anglais professionnel associé au développement des composants de présentation et de persistance
Connaissance des concepts et d'un langage de développement objet
Connaissance des principes et des règles du développement sécurisé
Connaissance des bases de la cryptographie
Connaissance des différents types de *Cloud* et de leur utilité pour le développement
Connaissance du rôle de l'infrastructure et des protocoles réseau sur la sécurité d'une application multicouche répartie
Connaissance du formalisme des diagrammes du langage de modélisation unifié UML
Connaissance de l'architecture applicative multicouche répartie et des différents types de serveurs
Connaissance des architectures logicielles multicouche réparties
Connaissance des bonnes pratiques de sécurité dans une architecture logicielle multicouche répartie et du rôle de chaque couche dans la stratégie de sécurité.
Connaissance des bonnes pratiques de conception de composants dans une architecture logicielle multicouche répartie
Connaissance de cadres d'applications (framework) ou de bibliothèques de composants de la couche présentation
Connaissance de cadres d'applications (framework) ou de bibliothèques de composants de la couche persistance de données
Connaissance des vulnérabilités spécifiques du cadre d'applications (framework) utilisé
Connaissance du processus et des outils d'intégration continue

► **Environnements techniques**

EDI : Visual Studio
Langage objet : C#
Web Services

► **Critères d'évaluation**

Les bonnes pratiques de conception objet et de développement sécurisé sont appliquées
La répartition des composants de l'application est conforme à l'architecture multicouche répartie

L'interface est conforme à la charte graphique du cahier des charges
Les composants d'interface de la couche présentation répondent aux fonctionnalités décrites dans le dossier de conception
La correspondance entre objets métiers et objets relationnels est opérationnelle
Toutes les couches remplissent le rôle prévu dans la stratégie de sécurité de l'application

II.5 Développer une application de mobilité numérique

À partir du cahier des charges de l'application, des spécificités ergonomiques et fonctionnelles de l'équipement mobile, et en suivant une démarche de développement en approche de type Agile, concevoir la maquette graphique appropriée à l'équipement et en respectant l'expérience utilisateur. Coder, documenter l'application mobile en respectant les contraintes de l'architecture du matériel cible et tester l'application dans l'environnement du matériel cible afin de répondre au besoin fonctionnel de l'utilisateur, en respectant à chaque étape l'état de l'art de la sécurité informatique. Réaliser les développements à partir d'un environnement de développement intégré, éventuellement de langue anglaise.

Pratiquer une veille technologique, y compris en anglais, pour résoudre un problème technique ou mettre en œuvre une nouvelle fonctionnalité ainsi que pour s'informer sur la sécurité informatique et les vulnérabilités connues.

Partager le résultat de sa veille avec ses pairs.

► Contexte professionnel de mise en œuvre

Le développement de l'application mobile se réalise en liaison étroite avec l'utilisateur et à l'aide d'un environnement de développement.

Une application mobile est une application qui est téléchargée sur un store (magasin d'applications) contrairement à une application web (webapp) qui est accessible depuis un navigateur.

Selon le besoin et le niveau de performance attendue, le concepteur-développeur peut développer une application native c'est-à-dire dédiée au matériel cible et à son système d'exploitation. Pour la rendre accessible, le concepteur-développeur doit la publier sur un store dédié au système d'exploitation mobile. Il peut aussi développer une application mobile de type multi plateforme (cross-platform), c'est-à-dire mise à disposition sur plusieurs systèmes d'exploitation. Le concepteur-développeur utilise alors un socle de développement commun. Les applications concernées s'adressent au grand public ou aux salariés d'entreprise et peuvent alimenter des stores privés.

► Savoirs, savoir-faire techniques, savoir-faire relationnels, savoir-faire organisationnels

Maquetter l'application mobile avec un outil : enchaînement et maquette des écrans associés
Adapter le développement de l'interface graphique aux spécificités de l'équipement mobile de façon ergonomique

Mettre en œuvre l'environnement de développement de l'application mobile en tenant compte du système d'exploitation cible

Coder dans le langage approprié les composants de l'application mobile, dans un style défensif
Mettre en œuvre les échanges de données entre l'équipement mobile et un serveur d'entreprise de façon sécurisée

Réaliser un jeu de tests de l'application mobile en fonction des caractéristiques du matériel cible, avec une double approche fonctionnelle et sécurité

Gérer la sécurité de l'application mobile en termes d'authentification, de chiffrement et de permissions

Gérer les performances de l'application mobile
Utiliser l'interface graphique, y compris en anglais, d'un environnement de développement d'applications mobiles
Utiliser un outil collaboratif de partage de ressources
Planifier et suivre les tâches de développement
Rechercher une réponse pertinente à une difficulté technique de développement ou à une vulnérabilité
Rechercher des informations sur des sites Internet, des forums et des FAQ francophones ou anglophones
Contribuer à la mise à jour des bases de connaissances francophones ou anglophones accessibles par Internet
Écouter, reformuler et synthétiser les demandes utilisateurs
Participer à une réunion de travail avec les utilisateurs, éventuellement en anglais Rédiger les comptes rendus de réunion, y compris en anglais
Connaissance d'une démarche de développement en approche de type Agile
Connaissance des normes réseaux liées au développement mobile
Connaissance des bases de la cryptographie
Connaissance des architectures d'application mobile : règles ergonomiques, contrôles graphiques et événements, cycle de vie, communication entre applications locale et distante, persistance, sécurité
Connaissance des formats normalisés d'échange de données
Connaissance de l'utilisation des services distants et des méthodes de sécurisation des échanges
Connaissance de l'environnement de développement approprié à l'équipement mobile
Connaissance des principes et des règles du développement sécurisé
Connaissance des vulnérabilités spécifiques au matériel mobile
Connaissance du vocabulaire anglais professionnel associé aux applications mobiles

► **Environnements techniques**

Universal Windows Platform

► **Critères d'évaluation**

Les fonctionnalités de l'application sont conformes au cahier des charges fonctionnel
L'ergonomie respecte l'expérience utilisateur du matériel cible mis en œuvre
Le code source des composants est documenté ou auto-documenté
L'application est testée sur les matériels cibles ou sur des émulateurs
L'application ne possède que les permissions exigées par ses fonctionnalités
La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité
La veille sur les vulnérabilités connues permet d'identifier et corriger des failles potentielles
La documentation technique liée aux technologies associées, en français ou en anglais, est comprise (sans contre-sens, ...)
Le partage du résultat de veille est effectué oralement ou par écrit avec ses pairs

II.6 Préparer et exécuter les plans de tests d'une application

À partir des scénarios des cas d'utilisation de l'application à tester, concevoir le plan de tests et créer l'environnement adéquat, puis réaliser les tests fonctionnels et les rapprocher des résultats attendus, afin de livrer une application répondant aux spécifications fonctionnelles et techniques. Rechercher les vulnérabilités par des tests de sécurité appropriés. Dans le cas d'une application en évolution ou d'un changement de version, identifier les cas de risques de régression et tester l'application à partir du plan de tests approprié.

► **Contexte professionnel de mise en œuvre**

Les tests fonctionnels se réalisent avec les utilisateurs de l'application. Les jeux d'essai sont mis en place sur une base de données de test. Dans le cas d'une démarche de développement en approche de type Agile, ces tests peuvent être intégrés au sein de la démarche de développement dite pilotée par les tests (Test Driven Development). Certains tests de sécurité peuvent être confiés à des experts sécurité (test d'intrusion).

► **Savoirs, savoir-faire techniques, savoir-faire relationnels, savoir-faire organisationnels**

Préparer et exécuter un plan de test de sécurité, en s'appuyant sur un guide de test reconnu, par exemple OWASP pour le web

Rechercher des failles de sécurité par des tests aléatoires (*fuzzing*)

Pratiquer une analyse statique de l'application

Exécuter les tests en manuel, ou en automatique dans le cadre d'un processus d'intégration continue

Analyser les résultats du test de charge et apporter les corrections

Analyser les résultats du test d'intrusion et apporter les corrections

Rédiger le dossier de compte rendu de tests

Faire effectuer un test de charge

Faire effectuer un test d'intrusion

Connaissance des vulnérabilités classiques, des failles de sécurité propres aux bases de données, aux applications web multicouche réparties

Connaissance des différents types de tests fonctionnels, structurels, de non régression et de sécurité Connaissance de la place et de l'impact des tests dans le cycle de vie du projet

Connaissance des outils de tests

► **Environnements techniques**

En fonction de l'architecture retenue

► **Critères d'évaluation**

Le plan de tests couvre l'ensemble des fonctionnalités retenues pour l'informatisation

Un environnement de tests est créé

Les tests, fonctionnels, structurels et de sécurité, exécutés sont conformes au plan de tests définis

Les résultats obtenus sont cohérents avec les résultats attendus

Les tests de non régression sont définis et exécutés conformément au plan de tests définis

II.7 Préparer et exécuter le déploiement d'une application

À partir de l'architecture de l'application répartie et éventuellement d'un processus d'intégration continue et des différents composants assemblés ou indépendants, élaborer le diagramme de déploiement correspondant. Déployer l'exécutable obtenu par assemblage des différents composants ou les exécutables des composants indépendants, sur le ou les environnements de qualification, y compris dans le Cloud, afin d'obtenir une application logicielle opérationnelle et signée selon les exigences de sécurité.

► **Contexte professionnel de mise en œuvre**

La préparation et le déploiement d'une application logicielle répartie se réalise en fin de processus de développement ou tout au long du processus avec un outil d'intégration continue. Dans le cas d'architecture en microservice, les composants sont déployés indépendamment. Le déploiement de l'application s'effectue sur un ou des environnements de qualification interne ou externe en mode Web ou en mode Cloud. Selon le contexte, la taille de la Direction des Systèmes d'information, le concepteur développeur est amené à exécuter ou non le déploiement.

► **Savoirs, savoir-faire techniques, savoir-faire relationnels, savoir-faire organisationnels**

Réaliser un diagramme de déploiement
Déployer l'application ou le microservice
Prendre en compte les dépendances vis-à-vis des composants externes du composant à déployer
Prendre en compte les évolutions de versions de l'ensemble des composants externes
Gérer la sécurité de l'application en termes de signature numérique des exécutables, selon les exigences de sécurité identifiées
Planifier et suivre les tâches de déploiement
Connaissance des bases de la cryptographie
Connaissance des concepts liés aux architectures réparties
Connaissance des différents types de serveurs
Connaissance des diagrammes UML concernant les composants et le déploiement
Connaissance du processus d'intégration continue
Connaissance d'un outil de signature de code
Connaissance des règles de mise en production associées à l'application

► **Environnements techniques**

En fonction de l'architecture retenue

► **Critères d'évaluation**

Le déploiement est formalisé à partir d'un diagramme
Les composants assemblés ou indépendants sont déployés sur les environnements de qualification
L'application déployée fournit les services demandés
Le code de l'application est signé, tout ou partie, selon les exigences de sécurité

II.8 Utiliser l'anglais dans son activité professionnelle en conception et développement d'applications

Dans le cadre de son activité professionnelle, pratiquer la langue anglaise afin d'exploiter des documentations techniques, de rédiger des dossiers techniques, de communiquer par mail, au téléphone ou lors de réunion.

Afin d'être opérationnel dans l'emploi, et par rapport au cadre européen commun de référence pour les langues, utiliser l'anglais au niveau B1 en compréhension de l'écrit, au niveau A2 en compréhension de l'oral et expression écrite et orale

► Critères d'évaluation

La documentation technique en anglais est comprise (sans contre-sens, ...)

La communication écrite en langue anglaise est rédigée de façon adaptée à l'interlocuteur et sans faute nuisant à la fiabilité de l'échange

La communication technique orale en langue anglaise est réalisée de façon simple sur des sujets professionnels, en face à face, au téléphone

II.9 Actualiser et partager ses compétences en conception et développement d'applications

À partir d'un besoin de compréhension ou de recherche d'information, se documenter et analyser les informations sur les technologies informatiques récentes. Rechercher des solutions innovantes et pertinentes pour la résolution de problèmes techniques dans le cadre du développement d'applications. Partager ces connaissances et compétences à partir des communautés de développeurs accessibles par internet. Cette compétence s'exerce en permanence et de façon transversale à l'emploi et souvent en anglais.

► Critères d'évaluation

L'objet de la recherche est exprimé de manière précise en langue française ou anglaise

La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité

La veille sur les vulnérabilités connues permet d'identifier et corriger des failles potentielles

Le partage du résultat de recherche et de veille est effectué, oralement ou par écrit, avec ses pairs

III. LE CAHIER DES CHARGES

III.1 Contexte fonctionnel et technique de l'application

La startup « DreamSoftware » est une entreprise active dans le domaine du service numérique. Elle est principalement axée sur le développement de solution logicielle. Elle a pour seul slogan « l'efficacité de ses logiciels ». Les logiciels créés sont simples d'utilisation et permettent d'automatiser certaines tâches d'un secteur d'activité précis.

« DreamSoftware » voudrait élargir sa gamme de logiciels pour cibler une nouvelle clientèle en proposant une solution informatique innovante sur le marché de la restauration. Pour y parvenir, elle doit respecter certaines étapes de son processus de développement.

La mission qui vous est confiée est de développer une solution qui fera office de preuve de concept pour les nouveaux clients afin de pouvoir obtenir un retour pour produire le logiciel commercialisé.

Il vous est demandé de développer une solution pour automatiser les commandes d'un restaurant auprès ses fournisseurs basés sur la réservation et le menu de la semaine. Pour cela, dans la première version, vous devrez mettre en place l'architecture et les fonctionnalités de base suivantes.

Deux niveaux d'utilisation doivent être pris en compte.

- Niveau administrateur – Le restaurateur (sur ordinateur)

Le rôle consiste à :

1. Saisir, modifier et supprimer le menu de la prochaine semaine composé des plats déjà renseignés dans l'application.
2. Ajouter de nouveaux plats avec le nombre d'ingrédients requis pour une personne.
3. Visualiser la commande d'ingrédients à effectuer pour la semaine à venir.

- Niveau utilisateur – Les clients (sur mobile)

Le rôle consiste à consulter le menu de la prochaine semaine et de réserver les jours retenus via smartphone. Les réservations sont clôturées chaque fin de semaine.

L'application mobile sera développée avec une architecture en couches.

III.2 Les données à traiter

« DreamSoftware » souhaite gérer les données suivantes :

- Pour les plats :
 - Nom du plat
 - Liste des ingrédients (ingrédient et quantité)
 - Type (entrée, plat ou dessert)
- Pour les ingrédients :
 - Nom de l'ingrédient
 - Prix moyen
- Menu :
 - Date de début de la semaine
 - Liste des plats de la semaine (entrée, plat, dessert) midi et soir
- Réservation :
 - Date
 - Nom et prénom

- Numéro de téléphone
- Midi et/ou Soir
- Quantités et formules au choix (entrée, plat, dessert), (entrée et plat), (plat et dessert)

« DreamSoftware » attends pouvoir trier les données suivantes dans la partie application de bureau :
Les plats :

- Par popularité (nombre de réservations)
- Par type (entrée, plat ou dessert)
- Par ingrédient particulier

III.3 Utilisateurs de l'application et gestion de la sécurité

1^{re} phase :

Pour la partie administration, il n'y a pas de contrôle particulier. À partir du moment où le logiciel est installé sur un poste toute personne ayant accès à ce poste peut l'utiliser.

Tous les utilisateurs qui installeront l'application mobile ont accès à la partie consultation et réservation.

2^e phase (Bonus : Système d'authentification et d'autorisation) :

La partie administration et la partie application mobile utilisent un système d'authentification basé sur les « **tokens JWT** ».

Les comptes d'administration seront créés dans la base de données.

Les utilisateurs de l'application mobile pourront s'enregistrer sur l'application et utiliser leur compte client.

III.4 Environnement technique

Les développements seront réalisés à partir de l'IDE Microsoft Visual Studio. Nous nous appuierons sur la plateforme .NET de Microsoft autant pour le client bureau, le serveur que pour la partie mobile. Nous utiliserons une base de données SQL Server.

Pour ce projet, chaque stagiaire aura accès à :

- Un serveur SQL Server chez OVH pour héberger la base de données.
- Un serveur Web IIS chez OVH pour héberger les services Web.

Les Url et les comptes vous seront communiqués individuellement par e-mail.

III.5 Architecture retenue

L'architecture retenue sera une architecture multicouche qui s'articulera ainsi :

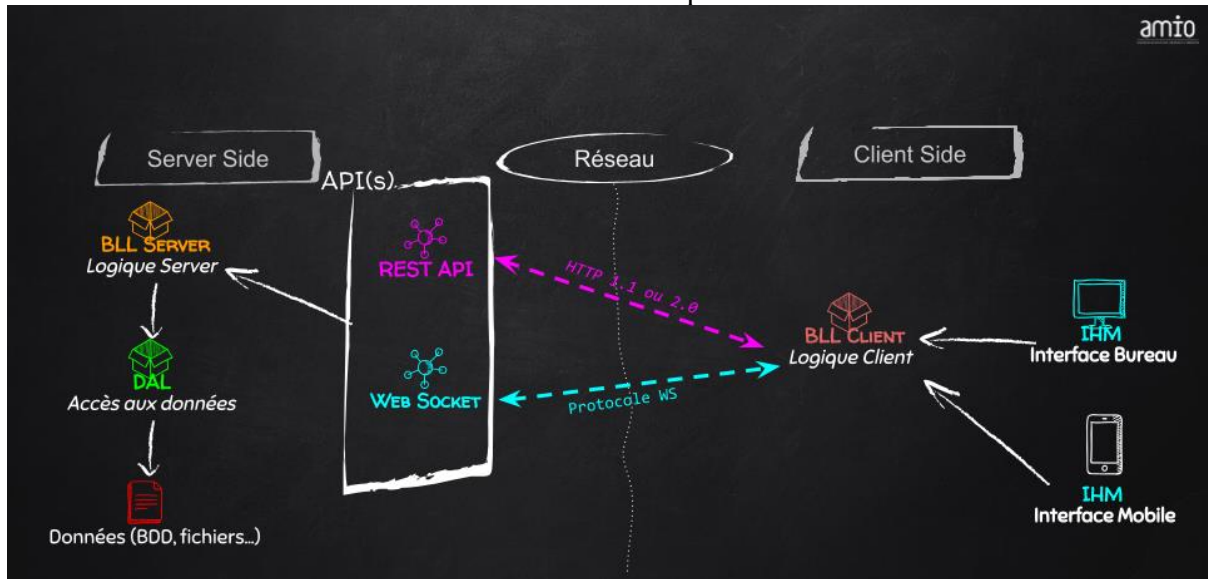


Figure 1 Architecture FoodBook

III.5.1 Les interfaces

L'interface de bureau sera développée sous la technologie **WinForm de microsoft**.

L'interface mobile quant à elle sera sous la technologie **UWP avec une architecture MVVM (Model vue, vue model)**.

III.5.2 La BLL(Business Logic Layer) Client

La BLL client, devra utiliser la librairie **standard .NET**, elle contiendra toute la logique client et elle seule communiquera avec les API(s).

III.5.3 L'API REST et Web Socket

Les composants serveur comme les API(s) devront être sous la **technologie .Net 5 (Net Core v5)**. Elles permettront de faire une abstraction de notre code serveur, et de fournir aux clients un choix exhaustif de fonctionnalités.

III.5.4 BLL(Business Logic Layer) Serveur

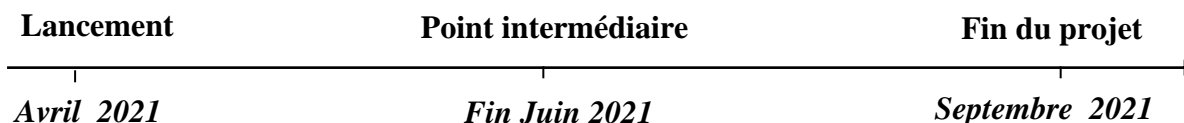
La BLL Serveur contiendra la logique du côté serveur et la cohérence des données.

III.5.5 DAL(Data Access Layer)

La DAL quant à elle communiquera avec la base de données, elle pourra être développée avec le **micro ORM « DAPPER »** ou avec **ADO.net et le framework .Net 5**.

IV. LIVRABLES ET MODALITES POUR L'EVALUATION DES COMPETENCES

Calendrier du projet :



Un **point d'avancement intermédiaire** est prévu fin juin. Sous forme d'une démo, vous présenterez votre travail à ce stade et rendrez compte de votre avancement :

- Votre reformulation et votre analyse du besoin
- Résumé du projet en Français (200 à 250 mots)
- Une explication de l'architecture retenue
- La structure de la base de données mise en place
- La conception technique : les classes métier, les classes d'accès aux données.
- Les maquettes d'écrans de l'application selon votre avancement.
- Le planning prévisionnel du projet, l'état d'avancement, les prochaines étapes.

Pour satisfaire à l'**évaluation finale des compétences**, vous rédigerez un **dossier projet** et réaliserez une **présentation orale** à partir d'un **support de présentation** de type Powerpoint.

La date de remise des résumés anglais et en français du projet de 200 à 250 mots est fixée **au lundi 30 aout 2021**.

La date de remise **des autres livrables** est fixée au **Vendredi 10 septembre 2021**. Les **livrables à fournir** seront :

- Les applications (client/serveur, web service et mobile)
- Un **dossier projet**
- Une présentation de type PowerPoint de votre projet.
- Votre **Dossier Professionnel** que vous aurez pris soin de compléter.

Contenu du dossier de projet :

Le dossier de projet respecte ce plan type :

1. Liste des compétences du référentiel qui sont couvertes par le projet
2. Résumé du projet en anglais d'une longueur d'environ 20 lignes soit 200 à 250 mots, ou environ 1200 caractères espaces non compris
3. Cahier des charges, expression des besoins
4. Spécifications fonctionnelles du projet
5. Spécifications techniques du projet, élaborées par le candidat, y compris pour la sécurité
6. Réalisations du candidat comportant les extraits de code les plus significatifs, et en les argumentant, y compris pour la sécurité
7. Plan de tests et de déploiement
8. Description de la veille, effectuée par le candidat durant le projet, sur les vulnérabilités de sécurité
9. Description d'une situation de travail ayant nécessité une recherche, effectuée par le candidat durant le projet, à partir de sites francophones ou anglophones

Pour la description de la veille sur les vulnérabilités de sécurité. Décrire comment vous avez effectué la veille de sécurité: les sites et les mots clés utilisés et indiquer les vulnérabilités trouvées et éventuellement les failles potentielles corrigées.

Pour la description de la situation de travail, ayant nécessité une recherche basée sur un ou des sites anglophones. La description doit concerner un problème technique ou une nouvelle fonctionnalité mise en œuvre, dans le cadre du projet FoodBook.

Décrire le besoin d'information, et indiquer comment vous avez effectué la recherche : les mots clés de recherche utilisés et la liste des sites retournés. Pour finir précisez les critères de sélection du (ou des) site(s) et indiquer la solution trouvée et si elle a pu être mise en œuvre.

Attention la longueur du dossier de projet hors annexes est de 30 à 35 pages, soit environ 48750 caractères espaces non compris.

Tous les livrables devront être fournis au format numérique (PDF et Word) et déposés sur la plateforme AMIO-Fit.

Les évaluations individuelles se dérouleront **du 14 au 16 septembre 2021.**

Vous démarrerez votre présentation **par un résumé en anglais du projet FoodBook.** Vous veillerez à montrer votre appropriation du sujet en reformulant les exigences fonctionnelles.

La présentation orale se déroulera en deux parties avec **40 minutes pour la présentation de la réalisation du projet FoodBook** et **30 minutes d'entretien technique avec le jury.**

Pour **les présentations orales**, vous préparez un support de présentation. Vous vous attacherez à produire une présentation professionnelle de votre travail.

- Vous rappellerez les objectifs, les jalons et annoncerez les compétences visées.
- Vous situerez le contexte. Vous reformulerez les contours du projet et les objectifs fonctionnels par type d'utilisateurs (vous pourrez vous appuyer sur votre modélisation UML). Vous montrerez que vous vous êtes approprié l'architecture.
- Vous explicitez votre démarche en mettant en évidence les principales étapes de votre travail
- Vous présentez les résultats les plus pertinents pour démontrer les compétences en jeu.
- Vous argumenterez vos choix. Vous exposerez les difficultés rencontrées et les solutions que vous avez apportées.
- Vous présentez votre organisation de la gestion du projet.
- En conclusion, vous ferez le point sur votre état d'avancement, vous dresserez votre bilan à cette étape.