**✅Chapitre 1 :. CONTEXTE GÉNÉRAL**

**1.1 Introduction  
1.2 Présentation de l’organisme d’accueil  
 1.2.1 Présentation générale  
 1.2.2 Organigramme de la société  
 1.2.3 Les services de la société  
1.3 Aperçu du projet  
 1.3.1 Contexte du projet  
 1.3.2 Problématique  
 1.3.3 Objectifs du projet**

**1.5 Choix de la méthodologie de travail  
 1.5.1 Différences entre méthodes classiques et agiles  
 1.5.2 Méthodologie agile adoptée : SCRUM  
 1.5.3 Pilotage du projet avec SCRUM  
1.6 Conclusion**

**✅ Chapitre 2 : Analyse et Spécification des Besoins**

**2.1. Introduction  
Courte présentation du chapitre et de son rôle dans le projet.**

**2.2. Analyse des besoins fonctionnels  
Ce que le système doit faire du point de vue de l'utilisateur ou de l’administrateur.  
Exemples : détecter une attaque, classifier le type d'attaque, envoyer une alerte, bloquer automatiquement...**

**2.3. Analyse des besoins non fonctionnels  
Les contraintes techniques ou qualitatives que le système doit respecter.  
Exemples : rapidité, fiabilité, sécurité, scalabilité, accessibilité cloud...**

**2.4. Environnement de travail  
*Présentation de l’environnement utilisé pour développer le projet.***

* **2.4.1. Environnement matériel  
  (Exemple : Processeur, RAM, etc.)**
* **2.4.2. Environnement logiciel  
  (Exemple : OS, éditeurs de code, bibliothèques...)**

**2.5. Spécification des objectifs du système  
Définition claire des objectifs globaux et spécifiques du projet.  
Exemples : intégrer un modèle IA, assurer la détection en temps réel, notifier par mail, etc.**

**2.6. Choix des outils et des technologies  
Présentation des technologies choisies pour le développement :**

* **Langages (ex : Python)**
* **Librairies (ex : Scikit-learn, TensorFlow)**
* **Base de données (ex : MongoDB, MySQL)**
* **Environnement cloud (ex : AWS, Azure)**
* **Outils d'alerte (ex : SMTP, SendGrid…)**

**2.7. Conception fonctionnelle du système (modélisation UML)  
Utilisation de diagrammes pour représenter le système :**

* **Diagramme de cas d’utilisation**
* **Diagramme de classes (si pertinent)**
* **Diagramme de séquence (exécution de scénarios)**

**2.8. Conclusion du chapitre  
Résumé des points importants abordés et transition vers le chapitre suivant (collecte et traitement des données).**

**✅ Chapitre 3 : Collecte et traitement des données – Sprint 1**

***(Correspond à l’ancien Sprint 2)***

* **3.1 Introduction**
* **3.2 Présentation des sources de données (KDD99, NSL-KDD, CICIDS2017, etc.)**
* **3.3 Prétraitement des données  
    • Encodage des variables  
    • Normalisation  
    • Gestion des valeurs manquantes**
* **3.4 Conception du schéma de la base de données**
* **3.5 Implémentation de la base d’attaques**
* **3.6 Conclusion du chapitre**

**✅ Chapitre 4 : Développement du modèle d’Intelligence Artificielle – Sprint 2**

* **4.1 Introduction**
* **4.2 Choix de l’algorithme de classification (Random Forest, SVM, CNN...)**
* **4.3 Entraînement du modèle sur la base de données**
* **4.4 Évaluation des performances (Accuracy, Recall, F1-score…)**
* **4.5 Optimisation du modèle**
* **4.6 Sauvegarde du modèle entraîné**
* **4.7 Conclusion du chapitre**

**✅ Chapitre 5 : Intégration du système d’alerte et détection d’attaques – Sprint 3**

* **5.1 Introduction**
* **5.2 Intégration du modèle IA dans le système**
* **5.3 Détection et classification automatique des attaques**
* **5.4 Blocage ou neutralisation des attaques**
* **5.5 Mise en place du système de notifications (email, alerte système…)**
* **5.6 Tests fonctionnels et scénarios d’attaque**
* **5.7 Conclusion du chapitre**

**✅ Chapitre 6 : Déploiement sur le cloud et documentation – Sprint 4**

* **6.1 Introduction**
* **6.2 Choix de la plateforme cloud (AWS, Azure, GCP…)**
* **6.3 Configuration du serveur cloud et hébergement de l’application**
* **6.4 Test de bon fonctionnement en environnement distant**
* **6.5 Documentation technique du système**
* **6.6 Conclusion générale**