

# CertForge - Document Consolidé des Exigences (Version Finale)

## Table des matières

1. [Introduction](#)
2. [Personas](#)
3. [User Stories](#)
4. [Exigences Fonctionnelles](#)
5. [Exigences Non-Fonctionnelles](#)
6. [Exigences Techniques](#)
7. [Priorisation MoSCoW avec Dépendances](#)
8. [Format du Schéma YAML/JSON](#)
9. [Mapping YAML vers Objets Internes](#)
10. [Workflow de Traitement](#)
11. [Checklist de Validation MVP](#)
12. [Matrice de Test QCM](#)
13. [Glossaire des Types de Questions](#)

## Introduction

CertForge est une plateforme d'apprentissage mobile-first, auto-gérée, qui transforme du contenu structuré généré par IA en un système personnalisé de maîtrise des certifications. Ce document présente les exigences complètes pour le développement de CertForge, incluant les besoins utilisateurs, les fonctionnalités, les contraintes techniques et la priorisation.

## Personas

### 1. Sarah - Chef de Projet en Transition

Sarah est une professionnelle en milieu de carrière qui souhaite obtenir sa certification PMP. Elle a un emploi du temps chargé, des responsabilités familiales, et cherche une solution qui s'intègre à son flux de travail existant avec Obsidian.

## 2. Marcus - Professionnel IT Sécurité

Marcus doit maintenir plusieurs certifications (CompTIA Security+, CISSP) pour répondre aux exigences de son employeur. Il voyage fréquemment et a besoin d'une solution qui fonctionne hors ligne et s'adapte à des sessions d'étude sporadiques.

## 3. Elena - Responsable Qualité

Elena travaille pour une entreprise manufacturière qui met en œuvre les normes ISO 9001. Elle et plusieurs collègues doivent obtenir la certification d'Auditeur Principal ISO 9001 et souhaitent créer une base de connaissances collaborative.

## 4. David - Consultant Indépendant

David est un consultant indépendant qui doit maintenir plusieurs certifications en cybersécurité et plateformes cloud. Sans soutien d'entreprise pour les frais de certification, il est particulièrement attentif aux coûts et frustré par les plateformes d'apprentissage propriétaires.

# User Stories

## Connexion et Configuration

1. **En tant que** nouvel utilisateur,  
**Je veux** pouvoir connecter l'application à mon Google Drive,  
**Afin de** synchroniser mes fichiers d'étude existants sans avoir à les recréer.
2. **En tant que** utilisateur d'Obsidian,  
**Je veux** que l'application reconnaisse mes fichiers YAML/JSON stockés dans Drive,  
**Afin de** maintenir mon système de gestion de connaissances existant.
3. **En tant que** professionnel occupé,  
**Je veux** une configuration initiale simple et rapide,  
**Afin de** commencer à étudier immédiatement sans passer beaucoup de temps sur la configuration.

## Gestion de Contenu

1. **En tant que** candidat à la certification,  
**Je veux** que l'application organise automatiquement mes flashcards par catégories et sous-catégories,  
**Afin de** naviguer facilement dans la structure de la certification.

2. **En tant que** créateur de contenu,  
**Je veux** des instructions claires sur le format YAML/JSON requis,  
**Afin de** créer des flashcards compatibles avec l'application.
3. **En tant que** utilisateur de ChatGPT/Gemini,  
**Je veux** des exemples de prompts pour générer des flashcards au format correct,  
**Afin de** créer rapidement du contenu d'étude pertinent.
4. **En tant que** professionnel en IT,  
**Je veux** pouvoir inclure des extraits de code et des diagrammes dans mes flashcards,  
**Afin de** préparer efficacement les certifications techniques.
5. **En tant que** créateur de contenu,  
**Je veux** pouvoir créer des questions à choix multiples (QCM),  
**Afin de** diversifier les types d'exercices et mieux simuler les examens de certification.

## Apprentissage et Quiz

1. **En tant que** étudiant,  
**Je veux** accéder à des sessions de quiz rapides (3-5 minutes),  
**Afin de** maximiser les petites fenêtres de temps disponibles dans ma journée.
2. **En tant que** candidat à la certification,  
**Je veux** pouvoir indiquer si j'ai répondu correctement ou non à chaque question,  
**Afin de** suivre mes progrès.
3. **En tant que** professionnel en préparation intensive,  
**Je veux** pouvoir choisir des catégories spécifiques pour mes sessions d'étude,  
**Afin de** me concentrer sur mes points faibles.
4. **En tant que** utilisateur mobile,  
**Je veux** une interface optimisée pour une utilisation à une main,  
**Afin de** pouvoir étudier pendant mes déplacements.
5. **En tant que** utilisateur ciblant ses lacunes,  
**Je veux** relancer uniquement mes erreurs passées,  
**Afin de** me concentrer sur les concepts que je maîtrise moins bien.
6. **En tant que** utilisateur avancé,  
**Je veux** filtrer les cartes par tag,  
**Afin de** réaliser un entraînement plus précis sur des concepts spécifiques.

7. **En tant que** candidat à la certification,  
**Je veux** m'entraîner avec des QCM similaires à ceux de l'examen réel,  
**Afin de** me familiariser avec le format d'évaluation officiel.
8. **En tant que** utilisateur,  
**Je veux** pouvoir sélectionner une ou plusieurs réponses dans un QCM,  
**Afin de** répondre correctement aux questions qui ont plusieurs bonnes réponses.
9. **En tant que** utilisateur mobile,  
**Je veux** des options de sélection lisibles et claires pour chaque choix de réponse,  
**Afin de** pouvoir répondre facilement aux QCM sur un petit écran.
10. **En tant que** utilisateur,  
**Je veux** pouvoir marquer une question comme 'à revoir plus tard',  
**Afin de** créer ma propre liste de révision personnalisée.

## Suivi et Statistiques

1. **En tant que** candidat à la certification,  
**Je veux** voir mes statistiques de performance par catégorie,  
**Afin d'** identifier mes forces et faiblesses.
2. **En tant que** étudiant méthodique,  
**Je veux** suivre mon historique de sessions d'étude,  
**Afin de** maintenir une routine d'apprentissage régulière.
3. **En tant que** professionnel avec une date d'examen fixée,  
**Je veux** voir mon taux de progression global,  
**Afin de** savoir si je suis sur la bonne voie pour être prêt à temps.

## Fonctionnalités Hors Ligne

1. **En tant que** voyageur fréquent,  
**Je veux** pouvoir synchroniser mon contenu pour une utilisation hors ligne,  
**Afin d'** étudier pendant les vols ou dans des zones sans connexion.
2. **En tant que** utilisateur en déplacement,  
**Je veux** que mes progrès hors ligne soient synchronisés lors de ma prochaine connexion,  
**Afin de** maintenir des statistiques précises.

# Exigences Fonctionnelles

## Import et Synchronisation

1. L'application doit permettre l'authentification à Google Drive via OAuth.
2. L'application doit scanner les dossiers spécifiés par l'utilisateur pour trouver les fichiers YAML/JSON compatibles.
3. L'application doit parser correctement les fichiers au format défini et signaler les erreurs de format.
4. L'application doit synchroniser automatiquement les modifications de fichiers lors du démarrage de l'application.
5. L'application doit permettre une synchronisation manuelle à la demande de l'utilisateur.

## Organisation du Contenu

1. L'application doit organiser les flashcards selon la structure de dossiers existante dans Google Drive.
2. L'application doit extraire et utiliser les tags/catégories définis dans les fichiers YAML/JSON.
3. L'application doit permettre la navigation dans la hiérarchie des catégories.
4. L'application doit afficher le nombre de flashcards disponibles par catégorie.
5. L'application doit permettre le filtrage des flashcards par tag pour un apprentissage ciblé.

## Quiz et Apprentissage

1. L'application doit proposer des sessions de quiz rapides (3-5 minutes) avec un nombre configurable de questions.
2. L'application doit permettre à l'utilisateur d'indiquer s'il a répondu correctement ou non à chaque question.
3. L'application doit permettre de filtrer les questions par catégorie/tag.
4. L'application doit supporter différents types de contenu dans les flashcards (texte, code, images simples).
5. L'application doit offrir une interface utilisable d'une seule main pour faciliter l'apprentissage mobile.
6. L'application doit permettre de relancer uniquement les questions précédemment manquées.
7. L'application doit supporter les questions à choix multiples (QCM) avec une ou plusieurs réponses correctes.

8. L'application doit permettre à l'utilisateur de sélectionner une ou plusieurs options dans un QCM.
9. L'application doit évaluer correctement les réponses partiellement correctes dans les QCM à réponses multiples.
10. L'application doit afficher clairement toutes les réponses correctes après la soumission d'un QCM.
11. L'application doit permettre de filtrer les sessions de quiz par type de question (flashcard standard, QCM).
12. L'application doit fournir des contrôles de sélection suffisamment grands et espacés pour une utilisation mobile confortable.
13. L'application doit indiquer visuellement si un QCM attend une ou plusieurs réponses.
14. L'application doit permettre à l'utilisateur de marquer des questions comme "à revoir plus tard".
15. L'application doit fournir un retour visuel sur les bonnes/mauvaises options après validation d'un QCM.
16. L'application doit alerter l'utilisateur si aucune réponse n'est sélectionnée lors de la validation d'un QCM.

## **Suivi et Statistiques**

1. L'application doit enregistrer les résultats de chaque session (date, durée, nombre de questions, taux de réussite).
2. L'application doit calculer et afficher le taux de réussite par catégorie.
3. L'application doit afficher l'historique des sessions d'étude.
4. L'application doit calculer un score global de progression.
5. L'application doit distinguer les performances sur les QCM vs les flashcards standard dans les statistiques.
6. L'application doit permettre de rejouer spécifiquement les QCM échoués via une combinaison de filtres par tag et statut d'erreur.
7. L'application doit permettre d'accéder à une liste des questions marquées "à revoir plus tard".

## **Mode Hors Ligne**

1. L'application doit télécharger et stocker localement les flashcards pour une utilisation hors ligne.
2. L'application doit enregistrer les résultats des sessions hors ligne et les synchroniser lors de la reconnexion.
3. L'application doit indiquer clairement quand elle fonctionne en mode hors ligne.

4. L'application doit basculer automatiquement en mode hors ligne en cas d'échec de connexion à Drive.

## **Exigences Non-Fonctionnelles**

### **Performance**

1. L'application doit démarrer en moins de 3 secondes sur un appareil Android de milieu de gamme.
2. L'application doit pouvoir gérer au moins 5000 flashcards sans dégradation notable des performances.
3. La synchronisation avec Google Drive ne doit pas prendre plus de 10 secondes pour 100 fichiers.
4. L'interface utilisateur doit répondre en moins de 200ms aux interactions de l'utilisateur.
5. L'ouverture d'un fichier de 500+ cartes doit se faire en moins de 3 secondes.

### **Fiabilité**

1. L'application ne doit pas perdre de données en cas d'interruption pendant une session d'étude.
2. L'application doit gérer correctement les erreurs de connexion à Google Drive sans planter.
3. L'application doit valider l'intégrité des fichiers YAML/JSON avant de les intégrer.
4. En cas d'échec de connexion à Drive, l'application doit utiliser automatiquement le cache local.

### **Compatibilité**

1. L'application doit fonctionner sur Android 10.0 et versions ultérieures.
2. L'application doit s'adapter correctement aux différentes tailles d'écran (smartphones et tablettes).
3. L'application doit fonctionner avec les comptes Google personnels et professionnels (Workspace).
4. L'application doit prendre en charge le mode sombre natif d'Android.

### **Sécurité**

1. L'application doit utiliser des méthodes d'authentification sécurisées pour accéder à Google Drive.

2. L'application doit demander uniquement les permissions minimales nécessaires à son fonctionnement.
3. Les données stockées localement doivent être chiffrées.
4. L'application ne doit conserver aucun contenu en clair sur un serveur tiers (Drive + cache local uniquement).

## Utilisabilité

1. L'application doit être utilisable sans formation préalable par un professionnel familier avec les smartphones.
2. L'interface doit être optimisée pour une utilisation à une main.
3. L'application doit offrir un mode sombre et un mode clair.
4. L'application doit fournir des messages d'erreur clairs et des suggestions de résolution.
5. Les éléments interactifs des QCM doivent être suffisamment grands et espacés pour éviter les erreurs de sélection sur mobile.

## Exigences Techniques

### Architecture

1. L'application doit suivre une architecture modulaire permettant l'ajout futur de fonctionnalités.
2. L'application doit implémenter une séparation claire entre la logique métier et l'interface utilisateur.
3. L'application doit utiliser des patterns de conception modernes (MVVM, Repository, etc.).

### Stockage de Données

1. L'application doit utiliser Room ou une solution équivalente pour le stockage local des données.
2. L'application doit implémenter un système de mise en cache efficace pour les fichiers Google Drive.
3. L'application doit gérer correctement les conflits de synchronisation.

### Parseur YAML/JSON

1. Le parseur doit supporter un schéma flexible mais validé pour les flashcards.
2. Le parseur doit fournir des messages d'erreur précis en cas de problème de format.



3. Le parseur doit être optimisé pour traiter rapidement de grands volumes de fichiers.
4. Le parseur doit supporter les différents types de questions (standard, QCM, images).
5. Le parseur doit lever une erreur explicite si un champ requis (options, correct\_answers) est absent dans un QCM.

## API Google Drive

1. L'intégration Google Drive doit utiliser les API officielles et respecter les bonnes pratiques.
2. L'application doit gérer efficacement les quotas et limitations de l'API Google Drive.
3. L'application doit implémenter une stratégie de retry en cas d'échec temporaire de l'API.

## Priorisation MoSCoW avec Dépendances

### Must Have (Doit avoir - MVP)

#### Connexion et Configuration

- Authentification à Google Drive via OAuth
- Scan des dossiers spécifiés pour trouver les fichiers YAML/JSON
- Parser les fichiers au format défini
- **Dépendances:** Définition complète du schéma YAML/JSON

#### Gestion de Contenu

- Organisation des flashcards selon la structure de dossiers existante
- **Dépendances:** Parseur YAML/JSON fonctionnel
- Navigation dans la hiérarchie des catégories
- **Dépendances:** Organisation des flashcards
- Affichage du nombre de flashcards par catégorie
- **Dépendances:** Organisation des flashcards

#### Quiz et Apprentissage

- Sessions de quiz rapides (3-5 minutes)
- **Dépendances:** Parseur YAML/JSON, organisation des flashcards
- Indication de réponse correcte/incorrecte par l'utilisateur
- **Dépendances:** Sessions de quiz
- Filtrage des questions par catégorie/tag

- **Dépendances:** Organisation des flashcards, extraction des tags
- Interface utilisable d'une seule main
- **Dépendances:** Aucune (peut être développé en parallèle)
- Relancer uniquement les erreurs passées
- **Dépendances:** Suivi des réponses correctes/incorrectes
- Support des QCM basiques (réponse unique)
- **Dépendances:** Parseur YAML/JSON, interface de quiz

## Suivi et Statistiques

- Enregistrement des résultats de chaque session
- **Dépendances:** Sessions de quiz, indication de réponse
- Calcul et affichage du taux de réussite par catégorie
- **Dépendances:** Enregistrement des résultats, organisation par catégorie
- Score global de progression
- **Dépendances:** Enregistrement des résultats

## Mode Hors Ligne

- Téléchargement et stockage local des flashcards
- **Dépendances:** Parseur YAML/JSON, authentification Drive
- Enregistrement des résultats des sessions hors ligne
- **Dépendances:** Stockage local, enregistrement des résultats
- Synchronisation lors de la reconnexion
- **Dépendances:** Stockage local, authentification Drive
- Basculement automatique en mode hors ligne
- **Dépendances:** Détection d'état de connexion

## Should Have (Devrait avoir - Post-MVP proche)

### Gestion de Contenu

- Instructions claires sur le format YAML/JSON requis
- **Dépendances:** Définition complète du schéma
- Exemples de prompts pour générer des flashcards avec ChatGPT/Gemini
- **Dépendances:** Définition complète du schéma
- Support pour les extraits de code et diagrammes simples
- **Dépendances:** Parseur YAML/JSON avancé

### Quiz et Apprentissage

- Support des QCM à réponses multiples
- **Dépendances:** Support QCM basique

- Évaluation des réponses partiellement correctes dans les QCM
- **Dépendances:** Support QCM à réponses multiples
- Contrôles de sélection optimisés pour mobile
- **Dépendances:** Interface de base des QCM
- Marquage des questions "à revoir plus tard"
- **Dépendances:** Interface de quiz, stockage local

## Suivi et Statistiques

- Historique détaillé des sessions d'étude
- **Dépendances:** Enregistrement des résultats
- Visualisations graphiques des performances
- **Dépendances:** Calcul des statistiques
- Identification des points faibles par catégorie
- **Dépendances:** Calcul des statistiques par catégorie
- Statistiques distinctes pour QCM et flashcards standard
- **Dépendances:** Support des QCM, enregistrement des résultats

## Interface Utilisateur

- Mode sombre et mode clair
- **Dépendances:** Aucune (peut être développé en parallèle)
- Adaptation aux différentes tailles d'écran
- **Dépendances:** Interface de base
- Messages d'erreur clairs et suggestions de résolution
- **Dépendances:** Gestion des erreurs de base

## Could Have (Pourrait avoir - Futures versions)

### Apprentissage Adaptatif

- Adaptation de la fréquence des questions selon les performances
- **Dépendances:** Statistiques détaillées par flashcard
- Prédiction de la dégradation des connaissances
- **Dépendances:** Historique détaillé des performances
- Recommandations personnalisées de révision
- **Dépendances:** Algorithme d'apprentissage adaptatif

### Collaboration et Partage

- Partage d'ensembles de flashcards avec des collègues
- **Dépendances:** Gestion avancée des permissions Drive
- Statistiques d'équipe basiques

- **Dépendances:** Partage de flashcards, agrégation de statistiques

## Intégrations

- Intégration directe avec Obsidian (pas seulement via Google Drive)
- **Dépendances:** API Obsidian ou solution alternative
- Synchronisation manuelle à la demande de l'utilisateur
- **Dépendances:** Mécanisme de synchronisation de base

## Types de Questions Avancés

- Support pour d'autres types de questions (cloze, vrai/faux)
- **Dépendances:** Parseur YAML/JSON extensible, interface modulaire

## Won't Have (N'aura pas - Hors scope initial)

### Fonctionnalités Premium

- Analyses de performance avancées
- Version iOS (focus initial sur Android uniquement)
- Intégration directe avec ChatGPT/Gemini dans l'application
- Suivi des progrès d'équipe avancé
- Génération automatique de contenu via IA

## Format du Schéma YAML/JSON

Voici un exemple de schéma YAML pour les flashcards CertForge, incluant les différents types de questions (standard, QCM, image) :

```
---
metadata:
  title: "CISSP - Contrôle d'Accès"
  category: "Cybersécurité"
  subcategory: "Contrôle d'Accès"
  tags: ["CISSP", "Sécurité", "IAM"]
  created: "2025-05-18"
  version: "1.0"

flashcards:
  - id: "cissp-001"
    type: "standard"
    question: "Quelle est la différence entre
l'authentification et l'autorisation?"
    answer: "L'authentification est le processus de
vérification de l'identité d'un utilisateur, tandis que
```

l'autorisation détermine ce qu'un utilisateur authentifié est autorisé à faire."

**explanation:** "L'authentification répond à la question 'Qui êtes-vous?' alors que l'autorisation répond à 'Que pouvez-vous faire?'"

**difficulty:** 2

**tags:** ["Authentification", "Autorisation"]

- **id:** "cissp-002"

**type:** "mcq"

**question:** "Quels sont les facteurs d'authentification valides? (Sélectionnez toutes les réponses correctes)"

**options:**

- **option\_id:** "A"

**text:** "Quelque chose que vous savez (mot de passe)"

- **option\_id:** "B"

**text:** "Quelque chose que vous avez (jeton physique)"

- **option\_id:** "C"

**text:** "Quelque chose que vous êtes (biométrie)"

- **option\_id:** "D"

**text:** "Quelque chose que vous préférez (préférences personnelles)"

- **option\_id:** "E"

**text:** "Quelqu'un que vous connaissez (référence sociale)"

**correct\_answers:** ["A", "B", "C"]

**explanation:** "Les trois facteurs d'authentification reconnus sont: quelque chose que vous savez (comme un mot de passe), quelque chose que vous avez (comme une carte à puce), et quelque chose que vous êtes (comme une empreinte digitale). Les préférences personnelles et les références sociales ne sont pas considérées comme des facteurs d'authentification standards."

**difficulty:** 3

**tags:** ["Authentification", "Facteurs d'authentification"]

- **id:** "cissp-003"

**type:** "mcq"

**question:** "Quel modèle de contrôle d'accès utilise des étiquettes pour classifier les informations?"

**options:**

- **option\_id:** "A"

**text:** "DAC (Discretionary Access Control)"

- **option\_id:** "B"

**text:** "MAC (Mandatory Access Control)"

- **option\_id:** "C"

**text:** "RBAC (Role-Based Access Control)"

- **option\_id:** "D"

**text:** "ABAC (Attribute-Based Access Control)"

**correct\_answers:** ["B"]

**explanation:** "Le contrôle d'accès obligatoire (MAC) utilise des étiquettes de sécurité pour classifier les informations et

définir les autorisations d'accès en fonction des habilitations des utilisateurs."

**difficulty:** 3

**tags:** ["Contrôle d'accès", "MAC"]

- **id:** "cissp-004"

**type:** "image"

**question:** "Identifiez les composants manquants dans ce diagramme d'architecture Zero Trust."

**question\_image:** "zero\_trust\_diagram.png"

**answer:** "Les composants manquants sont: 1) Le broker d'identité, 2) Le système d'évaluation continue des risques, 3) La couche de micro-segmentation."

**explanation:** "L'architecture Zero Trust repose sur ces composants pour établir la confiance dynamique plutôt que sur le périmètre réseau traditionnel."

**difficulty:** 4

**tags:** ["Zero Trust", "Architecture de sécurité"]

- **id:** "pmp-001"

**type:** "true\_false"

**question:** "Le diagramme de Gantt est un outil de la gestion du temps dans le PMBOK."

**answer:** true

**explanation:** "Le diagramme de Gantt est effectivement un outil de la gestion du temps (schedule management) dans le PMBOK, utilisé pour représenter graphiquement le calendrier du projet."

**difficulty:** 1

**tags:** ["PMP", "Gantt", "Schedule Management"]

- **id:** "pmp-002"

**type:** "cloze"

**question:** "Dans la gestion de projet agile, les trois rôles principaux de Scrum sont: Product Owner, \_\_\_\_\_ et Development Team."

**answer:** "Scrum Master"

**explanation:** "Les trois rôles principaux dans Scrum sont le Product Owner (qui représente les intérêts des parties prenantes), le Scrum Master (qui facilite le processus) et l'équipe de développement (qui livre le produit)."

**difficulty:** 2

**tags:** ["Agile", "Scrum", "Roles"]

---

Et un exemple équivalent en JSON :

```
{
  "metadata": {
    "title": "CISSP - Contrôle d'Accès",
```

```
"category": "Cybersécurité",
"subcategory": "Contrôle d'Accès",
"tags": ["CISSP", "Sécurité", "IAM"],
"created": "2025-05-18",
"version": "1.0"
},
"flashcards": [
  {
    "id": "cissp-001",
    "type": "standard",
    "question": "Quelle est la différence entre
l'authentification et l'autorisation?",
    "answer": "L'authentification est le processus de
vérification de l'identité d'un utilisateur, tandis que
l'autorisation détermine ce qu'un utilisateur authentifié est
autorisé à faire.",
    "explanation": "L'authentification répond à la question
'Qui êtes-vous?' alors que l'autorisation répond à 'Que pouvez-
vous faire?'",
    "difficulty": 2,
    "tags": ["Authentification", "Autorisation"]
  },
  {
    "id": "cissp-002",
    "type": "mcq",
    "question": "Quels sont les facteurs d'authentification
valides? (Sélectionnez toutes les réponses correctes)",
    "options": [
      {
        "option_id": "A",
        "text": "Quelque chose que vous savez (mot de passe)"
      },
      {
        "option_id": "B",
        "text": "Quelque chose que vous avez (jeton physique)"
      },
      {
        "option_id": "C",
        "text": "Quelque chose que vous êtes (biométrie)"
      },
      {
        "option_id": "D",
        "text": "Quelque chose que vous préférez (préférences
personnelles)"
      },
      {
        "option_id": "E",
        "text": "Quelqu'un que vous connaissez (référence
sociale)"
      }
    ],
    "correct_answers": ["A", "B", "C"],
```

```
    "explanation": "Les trois facteurs d'authentification reconnus sont: quelque chose que vous savez (comme un mot de passe), quelque chose que vous avez (comme une carte à puce), et quelque chose que vous êtes (comme une empreinte digitale). Les préférences personnelles et les références sociales ne sont pas considérées comme des facteurs d'authentification standards.",
    "difficulty": 3,
    "tags": ["Authentification", "Facteurs d'authentification"]
  },
  {
    "id": "cissp-003",
    "type": "mcq",
    "question": "Quel modèle de contrôle d'accès utilise des étiquettes pour classifier les informations?",
    "options": [
      {
        "option_id": "A",
        "text": "DAC (Discretionary Access Control)"
      },
      {
        "option_id": "B",
        "text": "MAC (Mandatory Access Control)"
      },
      {
        "option_id": "C",
        "text": "RBAC (Role-Based Access Control)"
      },
      {
        "option_id": "D",
        "text": "ABAC (Attribute-Based Access Control)"
      }
    ],
    "correct_answers": ["B"],
    "explanation": "Le contrôle d'accès obligatoire (MAC) utilise des étiquettes de sécurité pour classifier les informations et définir les autorisations d'accès en fonction des habilitations des utilisateurs.",
    "difficulty": 3,
    "tags": ["Contrôle d'accès", "MAC"]
  },
  {
    "id": "cissp-004",
    "type": "image",
    "question": "Identifiez les composants manquants dans ce diagramme d'architecture Zero Trust.",
    "question_image": "zero_trust_diagram.png",
    "answer": "Les composants manquants sont: 1) Le broker d'identité, 2) Le système d'évaluation continue des risques, 3) La couche de micro-segmentation.",
    "explanation": "L'architecture Zero Trust repose sur ces composants pour établir la confiance dynamique plutôt que sur le
```



```

périmètre réseau traditionnel.",
  "difficulty": 4,
  "tags": ["Zero Trust", "Architecture de sécurité"]
},
{
  "id": "pmp-001",
  "type": "true_false",
  "question": "Le diagramme de Gantt est un outil de la
gestion du temps dans le PMBOK.",
  "answer": true,
  "explanation":
"Le diagramme de Gantt est effectivement un outil de la gestion
du temps (schedule management) dans le PMBOK, utilisé pour
représenter graphiquement le calendrier du projet.",
  "difficulty": 1,
  "tags": ["PMP", "Gantt", "Schedule Management"]
},
{
  "id": "pmp-002",
  "type": "cloze",
  "question": "Dans la gestion de projet agile, les trois
rôles principaux de Scrum sont: Product Owner, _____ et
Development Team.",
  "answer": "Scrum Master",
  "explanation":
"Les trois rôles principaux dans Scrum sont le Product Owner
(qui représente les intérêts des parties prenantes), le Scrum
Master (qui facilite le processus) et l'équipe de développement
(qui livre le produit).",
  "difficulty": 2,
  "tags": ["Agile", "Scrum", "Roles"]
}
]
}

```

## Mapping YAML vers Objets Internes

Le tableau suivant présente le mapping entre les champs YAML/JSON et les classes/objets internes de l'application Android :

| Champ YAML/JSON | Classe/Objet Android | Description   |
|-----------------|----------------------|---|
| metadata        | DeckMetadata         | Classe contenant le métadonnée du jeu de flashcards |

| Champ YAML/JSON                    | Classe/Objet Android                               | Description  |
|------------------------------------|--|--|
| <code>metadata.title</code>        | <code>DeckMetadata.title: String</code>            | Titre du jeu de flashcard                                      |
| <code>metadata.category</code>     | <code>DeckMetadata.category: String</code>         | Catégorie principale du jeu                                    |
| <code>metadata.subcategory</code>  | <code>DeckMetadata.subcategory: String</code>      | Sous-catégorie du jeu  |
| <code>metadata.tags</code>         | <code>DeckMetadata.tags: List&lt;String&gt;</code> | Liste des tags associés au jeu                                 |
| <code>metadata.created</code>      | <code>DeckMetadata.createdDate: LocalDate</code>   | Date de création du jeu  |
| <code>metadata.version</code>      | <code>DeckMetadata.version: String</code>          | Version du jeu   |
| <code>flashcards</code>            | <code>List&lt;Flashcard&gt;</code>                 | Liste des flashcards du jeu                                    |
| <code>flashcards[].id</code>       | <code>Flashcard.id: String</code>                  | Identifiant unique de la flashcard                             |
| <code>flashcards[].type</code>     | <code>Flashcard.type: FlashcardType (enum)</code>  | Type de la flashcard (standard, mcq, image, true_false, cloze) |
| <code>flashcards[].question</code> | <code>Flashcard.question: String</code>            | Question de la flashcard                                       |
| <code>flashcards[].answer</code>   | <code>Flashcard.answer: String</code>              | Réponse de la flashcard  |

| Champ YAML/JSON                               | Classe/Objet Android   | Description   |
|---|--|---|
|   |  | (pour les types standard, image, true_false, cloze) |
| <code>flashcards[].explanation</code>         | <code>Flashcard.explanation: String?</code>                  | Explication optionnelle de la réponse               |
| <code>flashcards[].difficulty</code>          | <code>Flashcard.difficulty: Int</code>                       | Niveau de difficulté (1-5)                          |
| <code>flashcards[].tags</code>                | <code>Flashcard.tags: List&lt;String&gt;</code>              | Tags spécifiques à la flashcard                     |
| <code>flashcards[].question_image</code>      | <code>Flashcard.questionImage: String?</code>                | Chemin de l'image pour les flashcards de type image |
| <code>flashcards[].options</code>             | <code>McqFlashcard.options: List&lt;McqOption&gt;</code>     | Options pour les QCM (sous-classe de Flashcard)     |
| <code>flashcards[].options[].option_id</code> | <code>McqOption.id: String</code>                            | Identifiant de l'option                             |
| <code>flashcards[].options[].text</code>      | <code>McqOption.text: String</code>                          | Texte de l'option                                   |
| <code>flashcards[].correct_answers</code>     | <code>McqFlashcard.correctAnswers: List&lt;String&gt;</code> | Liste des identifiants des réponses correctes       |

| Champ YAML/JSON                               | Classe/Objet Android  | Description                                   |
|---|---|---|
| <code>flashcards[].answer</code> (true_false) | <code>TrueFalseFlashcard.isTrue:</code><br><code>Boolean</code> | Valeur booléenne pour les questions vrai/faux |

## Classes supplémentaires pour la gestion des données

| Classe                       | Description                               | Champs principaux   |
|------------------------------|---|---|
| <code>FlashcardDeck</code>   | Représente un jeu complet de flashcards   | <code>metadata: DeckMetadata</code> ,<br><code>flashcards: List&lt;Flashcard&gt;</code> ,<br><code>sourceFile: String</code>                                      |
| <code>StudySession</code>    | Représente une session d'étude            | <code>id: UUID</code> , <code>startTime: LocalDateTime</code> , <code>endTime: LocalDateTime?</code> , <code>flashcardResults: List&lt;FlashcardResult&gt;</code> |
| <code>FlashcardResult</code> | Résultat d'une flashcard dans une session | <code>flashcardId: String</code> , <code>isCorrect: Boolean</code> , <code>responseTime: Long</code> , <code>markedForReview: Boolean</code>                      |
| <code>UserProgress</code>    | Progression globale de l'utilisateur      | <code>categoryStats: Map&lt;String, CategoryStats&gt;</code> , <code>totalCards: Int</code> , <code>masteredCards: Int</code>                                     |
| <code>CategoryStats</code>   | Statistiques par catégorie                | <code>category: String</code> , <code>correctCount: Int</code> , <code>totalCount: Int</code> , <code>lastStudied: LocalDateTime?</code>                          |

## Workflow de Traitement

### Étapes du Processus

1. **Détection du Fichier**
2. L'application scanne les dossiers spécifiés dans Google Drive

3. Identification des nouveaux fichiers ou des fichiers modifiés avec extensions .yaml, .yml, ou .json

#### **4. Validation du Format**

5. Vérification de la structure générale (metadata + flashcards)

6. Validation des champs obligatoires (title, category, question, answer)

7. Vérification de la cohérence des types de données

8. Validation spécifique selon le type de question (standard, mcq, image, true\_false)

9. Vérification de la présence des champs requis pour chaque type (ex: options et correct\_answers pour les QCM)

#### **10. Traitement des Erreurs**

11. En cas d'erreur de format, génération d'un log détaillé

12. Affichage d'une notification à l'utilisateur avec la nature de l'erreur

13. Suggestion de correction (exemple: "Champ 'options' manquant dans la flashcard QCM avec ID 'risk-003'")

#### **14. Intégration dans la Base de Données Locale**

15. Extraction des métadonnées (titre, catégorie, sous-catégorie, tags)

16. Création/mise à jour des entrées dans la base de données locale

17. Indexation pour la recherche et le filtrage

#### **18. Organisation Hiérarchique**

19. Création/mise à jour de la structure de navigation basée sur les catégories et sous-catégories

20. Association des flashcards aux nœuds appropriés dans la hiérarchie

21. Mise à jour des compteurs de flashcards par catégorie

#### **22. Mise en Cache pour Utilisation Hors Ligne**

23. Stockage des données structurées pour un accès rapide

24. Enregistrement des références aux images (si présentes)

25. Téléchargement des ressources externes pour utilisation hors ligne

#### **26. Confirmation à l'Utilisateur**

27. Notification de succès avec statistiques (nombre de flashcards importées, catégories)

## 28. Proposition de démarrer une session d'étude avec le nouveau contenu

### Logs d'Erreur Potentiels

#### Erreurs de Format

```
ERREUR DE FORMAT - fichier: pmp_risk.yaml
- Ligne 25: Champ obligatoire 'question' manquant dans la
  flashcard avec ID 'risk-003'
- Ligne 42: Type de données incorrect pour 'difficulty' -
  attendu: nombre, trouvé: chaîne de caractères
- Ligne 78: Champ 'options' manquant dans la flashcard QCM avec
  ID 'risk-007'
```

#### Erreurs de Contenu

```
AVERTISSEMENT - fichier: cissp_security.yaml
- Flashcard ID 'sec-001' dupliquée (première occurrence ligne
  15, seconde occurrence ligne 87)
- Image référencée 'network_diagram.png' introuvable dans Google
  Drive
- QCM ID 'sec-005' n'a aucune réponse correcte définie dans
  correct_answers
```

#### Erreurs de Synchronisation

```
ERREUR DE SYNCHRONISATION - fichier: iso9001_audit.yaml
- Impossible d'accéder au fichier: vérifiez les permissions dans
  Google Drive
- Quota d'API Google Drive dépassé, nouvelle tentative dans 65
  secondes
```

## Checklist de Validation MVP

### Fonctionnalités Essentielles

#### Connexion et Configuration

- [ ] Authentification Google Drive fonctionnelle
- [ ] Sélection de dossiers Drive intuitive
- [ ] Parseur YAML/JSON robuste avec gestion des erreurs
- [ ] Synchronisation automatique au démarrage

## Organisation et Navigation

- [ ] Structure hiérarchique reflétant l'organisation des dossiers
- [ ] Navigation fluide entre catégories et sous-catégories
- [ ] Filtrage par tag fonctionnel
- [ ] Compteurs de flashcards précis par catégorie

## Quiz et Apprentissage

- [ ] Sessions de quiz rapides (3-5 minutes) configurables
- [ ] Interface d'évaluation (correct/incorrect) intuitive
- [ ] Option de relancer uniquement les erreurs passées
- [ ] Filtrage par catégorie/tag dans les sessions
- [ ] Support des QCM avec sélection unique ou multiple
- [ ] Évaluation correcte des réponses aux QCM
- [ ] Rendu visuel clair des QCM (espacement, contraste)
- [ ] Indication visuelle du type de sélection attendu (unique/multiple)
- [ ] Retour visuel sur les bonnes/mauvaises options après validation
- [ ] Comportement approprié en cas de sélection incomplète ou absence de sélection

## Suivi et Statistiques

- [ ] Enregistrement fiable des résultats de session
- [ ] Statistiques par catégorie précises et utiles
- [ ] Score global de progression clair
- [ ] Historique des sessions accessible
- [ ] Capacité à rejouer uniquement les QCM échoués

## Mode Hors Ligne

- [ ] Téléchargement complet du contenu pour usage hors ligne
- [ ] Fonctionnement sans dégradation en mode avion
- [ ] Synchronisation des résultats à la reconnexion
- [ ] Basculement automatique en cas de perte de connexion

## Validation UX

### Performance

- [ ] Démarrage de l'application < 3 secondes
- [ ] Ouverture d'un fichier de 500+ cartes < 3 secondes
- [ ] Navigation sans latence perceptible

- ☐ Synchronisation Drive efficace (< 10s pour 100 fichiers)

## Interface

- ☐ Utilisable d'une seule main
- ☐ Lisibilité optimale (contraste, taille de texte)
- ☐ Mode sombre/clair fonctionnel
- ☐ Adaptation correcte aux différentes tailles d'écran
- ☐ Éléments interactifs des QCM suffisamment grands et espacés

## Expérience Utilisateur

- ☐ Parcours d'onboarding clair pour nouveaux utilisateurs
- ☐ Messages d'erreur compréhensibles avec solutions
- ☐ Feedback visuel approprié pour les actions
- ☐ Cohérence visuelle à travers l'application
- ☐ Comportement approprié en cas de sélection incomplète ou partiellement correcte dans les QCM

## Fiabilité

- ☐ Aucune perte de données en cas d'interruption
- ☐ Gestion robuste des erreurs de connexion
- ☐ Validation de l'intégrité des fichiers
- ☐ Récupération automatique après crash

## Matrice de Test QCM

Le tableau suivant présente les scénarios de test spécifiques pour la fonctionnalité QCM :

| ID      | Scénario                                  | Entrée                               | Sortie Attendue                                     | Critère de Réussite                                |
|---------|---|--------------------------------------|---|--|
| QCM-T01 | QCM à réponse unique - réponse correcte   | Sélection de l'option B (correcte)   | Feedback positif, marqué comme correct              | Statistiques mises à jour, feedback visuel correct |
| QCM-T02 | QCM à réponse unique - réponse incorrecte | Sélection de l'option A (incorrecte) | Feedback négatif, affichage de la bonne réponse (B) | Statistiques mises à jour, feedback visuel correct |



| ID      | Scénario   | Entrée   | Sortie Attendue                                     | Critère de Réussite                                   |
|---------|--|--|---|---|
| QCM-T03 | QCM à réponses multiples - toutes correctes        | Sélection des options A, B, C (toutes correctes)           | Feedback positif, marqué comme correct              | Statistiques mises à jour, feedback visuel correct    |
| QCM-T04 | QCM à réponses multiples - partiellement correctes | Sélection des options A, B (C manquante)                   | Feedback mitigé, marqué comme partiellement correct | Score partiel enregistré, feedback visuel différencié |
| QCM-T05 | QCM à réponses multiples - incorrectes             | Sélection des options A, D (B, C manquantes, D incorrecte) | Feedback négatif, affichage des bonnes réponses     | Statistiques mises à jour, feedback visuel correct    |
| QCM-T06 | QCM - aucune sélection                             | Tentative de validation sans sélection                     | Message d'alerte, empêche la validation             | L'utilisateur est informé, aucune progression         |
| QCM-T07 | QCM - marquage pour révision                       | Marquer un QCM "à revoir plus tard"                        | QCM ajouté à la liste de révision                   | QCM accessible via la liste de révision               |
| QCM-T08 | QCM - filtrage par type                            | Sélection du filtre "QCM uniquement"                       | Seuls les QCM sont affichés                         | Filtrage correct, compteur précis                     |
| QCM-T09 | QCM - relecture des erreurs                        | Sélection "Revoir les erreurs" après une session           | QCM incorrects/ partiellement corrects affichés     | Seuls les QCM échoués sont présentés                  |
| QCM-T10 | QCM - accessibilité mobile                         | Test sur petit écran (5")                                  | Options clairement visibles et sélectionnables      | Pas d'erreurs de sélection involontaires              |

# Glossaire des Types de Questions

| Type              | Description   | Champs Requis   | Interaction Utilisateur   |
|-------------------|---|---|---|
| <b>standard</b>   | Question classique avec réponse textuelle                           | <code>question</code> , <code>answer</code>                                 | L'utilisateur lit la question, réfléchit à la réponse, puis révèle la réponse correcte et s'auto-évalue |
| <b>mcq</b>        | Question à choix multiples avec une ou plusieurs réponses correctes | <code>question</code> , <code>options</code> , <code>correct_answers</code> | L'utilisateur sélectionne une ou plusieurs options, puis valide sa réponse                              |
| <b>image</b>      | Question basée sur une image avec réponse textuelle                 | <code>question</code> , <code>question_image</code> , <code>answer</code>   | L'utilisateur observe l'image, répond à la question, puis révèle la réponse correcte                    |
| <b>true_false</b> | Question à laquelle on répond par vrai ou faux                      | <code>question</code> , <code>answer</code> (booléen)                       | L'utilisateur sélectionne "Vrai" ou "Faux", puis valide sa réponse                                      |
| <b>cloze</b>      | Phrase à compléter avec un ou plusieurs mots manquants              | <code>question</code> (avec emplacement marqué), <code>answer</code>        | L'utilisateur remplit le(s) blanc(s), puis valide sa réponse  |
| <b>matching</b>   | Association d'éléments entre deux colonnes                          | <code>question</code> , <code>pairs</code>                                  | L'utilisateur associe les éléments de la colonne de gauche avec ceux de la colonne de droite            |
| <b>ordering</b>   | Mise en ordre d'éléments selon un critère                           | <code>question</code> , <code>items</code> , <code>correct_order</code>     | L'utilisateur réorganise les éléments dans l'ordre correct  |

Note: Les types "matching" et "ordering" sont mentionnés à titre indicatif pour l'évolutivité future et ne font pas partie du MVP ou du Should Have.