# 📘 AgilePM® v3 – Guide Complet de Révision

*(Mieux structuré, 100% conforme v3, + exemples, + mnémotechniques, + liens logiques)*

# **1. Introduction générale**

AgilePM® v3 est une méthode agile de **gestion de projet** centrée sur :

* **La gouvernance** (rôles, phases, contrôles)
* **La livraison rapide** (timeboxing, increments)
* **Les priorités par valeur métier** (MoSCoW)
* **La combinaison Scrum + DevOps + Management Projet**

Elle répond à un besoin : **être agile sans perdre la maîtrise**, surtout dans les grands groupes ou projets réglementés.

🎯 **Objectif examen** : comprendre **comment AgilePM combine agilité + gouvernance**.

## La Philosophie AgilePM® v3 (Philosophy)\*\*

*“Best business value emerges when projects are aligned to clear business goals, deliver frequently and involve the collaboration of motivated and empowered people.”*

Traduction et explication :

La meilleure valeur métier est obtenue lorsque les projets :

1. **Sont alignés sur des objectifs métier clairs** (vision, bénéfices attendus, priorités).
2. **Livrent fréquemment** des incréments testés et exploitables.
3. **Impliquent des personnes motivées et responsabilisées** (tous les rôles collaborent, empowerment effectif).

Tips : Toujours relier chaque action au résultat métier concret.

Pièges : Livrer beaucoup de fonctionnalités “cool” sans valeur réelle → fail examen.

Parfait ! Voici une version **détaillée 3× plus** de ta section **Introduction, Philosophie et Principes Fondamentaux**, intégrant des explications approfondies, des tips, pièges et ce qu’il faut retenir pour l’examen AgilePM® v3 :

## 

AgilePM® v3 est une méthode agile de **gestion de projet** qui combine **gouvernance**, **livraison rapide**, **priorisation par valeur métier (MoSCoW)** et **intégration Scrum + DevOps + Management projet**. Elle est conçue pour maximiser la **valeur business tout en réduisant le risque**.

### 1.1 La Philosophie AgilePM® v3

*"Best business value emerges when projects are aligned to clear business goals, deliver frequently and involve the collaboration of motivated and empowered people."*

**Décryptage :**

1. **Alignement sur les objectifs métier**
   * Chaque projet doit répondre à une **vision claire**, des **bénéfices mesurables** et des **priorités validées**.
   * **Exemple pratique :** Pour un projet e-commerce, la priorité est le paiement sécurisé avant les options de personnalisation.
2. **Livraison fréquente**
   * Les **Timeboxes** (périodes fixes) permettent des **incréments testables** et exploitables.
   * Les feedbacks rapides réduisent le risque de dérive.
   * **Tip Examen :** La fréquence de livraison et l’itération continue sont des concepts toujours interrogés.
3. **Collaboration et empowerment**
   * Les rôles sont motivés, responsabilisés et collaborent activement.
   * **Exemple :** L’Ambassador guide quotidiennement les développeurs sur les exigences métier.

**À retenir :** La philosophie est le **fil conducteur** de tous les concepts AgilePM : Timeboxes, MoSCoW, Work Products, rôles.

### 1.2 Les 8 Principes Fondamentaux (CRITIQUE pour l’examen)

Les principes sont issus de **DSDM** et définissent le **comportement attendu**.

| **N°** | **Principe** | **Explication détaillée** | **Rôle principal** | **Tips / Pièges** |
| --- | --- | --- | --- | --- |
| 1 | **Focus on the Business Need** | Toute décision ou fonctionnalité doit générer une **valeur tangible**. | Business Sponsor & Visionary | Piège : ne pas inclure des fonctionnalités “nice-to-have” dans Must Have. |
| 2 | **Deliver on Time** | La date est fixe ; le périmètre (scope) est ajustable via MoSCoW. | PM & TL | Astuce : Les Timeboxes respectent la règle du périmètre variable mais qualité constante. |
| 3 | **Collaborate** | L’équipe prend des décisions collectives et partage la responsabilité. | TL & BA | Piège : ne pas impliquer l’Ambassador peut créer des écarts avec le métier. |
| 4 | **Never Compromise Quality** | La qualité minimale est définie en Foundations et doit être respectée. | Tester & TC | Piège critique : sacrifier la qualité = échec garanti à l’examen. |
| 5 | **Build Incrementally from Firm Foundations** | Bases solides : Pre-Project, Feasibility, Foundations avant le développement. | PM & TC | Piège : “bricoler” sans vision ou architecture conduit à l’échec du projet. |
| 6 | **Develop Iteratively** | Développement en cycles : coder, tester, améliorer continuellement. | Dev & Tester | Piège : confondre itératif (amélioration) et incrémentiel (ajout de blocs). |
| 7 | **Communicate Continuously and Clearly** | Communication fréquente et verbale : Daily Stand-ups, ateliers. | TL & Tous | Astuce : documenter l’essentiel, éviter trop de formalisation inutile. |
| 8 | **Demonstrate Control** | Suivi via Work Products, Checkpoints, Risk Logs, révisions. | PM & Change Authority | Tip : chaque Timebox doit générer des artefacts visibles et contrôlables. |

### 1.3 Tips Mnémotechniques et pièges fréquents

| **Concept** | **Mnémotechnique** | **Explication** | **Tips / Pièges / À retenir** |
| --- | --- | --- | --- |
| 8 Principes | **V-T-C-Q-I-I-C-D** | Valeur, Temps, Collaborer, Qualité, Incrémentiel, Itératif, Communication, Démonstration | Ne pas confondre Incrémentiel (construction sur fondations) vs Itératif (amélioration continue) |
| 5 sous-ensembles du Timebox | **K-I-R-C-C** | Kick-off, Investigation, Refinement, Consolidation, Close | Refinement = ajuster MoSCoW avant Consolidation. |
| Rôles Business | **S-V-A** | Sponsor, Visionary, Ambassador | Ambassador = voix quotidienne du métier, Visionary = “pourquoi” |
| Phase Foundations | **V-A-P** | Vision, Architecture, PRL | À définir avant toute Timebox pour garantir “Firm Foundations” |
| MoSCoW | **M-S-C-W** | Must ≤60%, Should, Could, Won’t | Pièges : trop de Must, confondre Should/Must, qualité = Must Always |
| Rôles Support | **SUDeSC** | Security, UX, DevOps, Specialists, Change | Indispensable pour conformité et CI/CD |

### 1.4 Liens clés pour la révision

1. **MoSCoW ≤60% Must** → MVP viable → Pareto 80/20 appliqué : 20% fonctionnalités = 80% valeur.
2. **Timebox = moteur du projet** → 5 sous-ensembles, incréments et artefacts visibles.
3. **Work Products par phase et rôle** → démonstration de contrôle et traçabilité.
4. **Alignement AgilePM ↔ Scrum** → comprendre équivalences : Team Leader = Scrum Master, Timebox = Sprint.
5. **Quality Never Compromised** → règle d’or pour examen et projet réel.

# **2. Les 4 groupes de rôles AgilePM® v3**

*(Business, Solution, Project, Support)*

## 🟦 **A. BUSINESS ROLES**

Orientés **valeur, besoins, justification, priorités**.

| **Rôle** | **Missions clés** | **À retenir pour l’examen** |
| --- | --- | --- |
| **Business Sponsor** | Vision stratégique, financement, arbitrages | C’est *le* décideur final |
| **Business Visionary** | Décrit la vision métier, valide la solution | Porte le “pourquoi” du projet |
| **Business Ambassador** | Clarifie besoins au quotidien, présent en Timebox | C’est le “PO détaillé” |

🎯 Mnémotechnique Business : **“SV-A” = Sponsor – Visionary – Ambassador**

## 🟩 **B. SOLUTION / DELIVERY ROLES**

| **Rôle** | **Missions** | **Spécificités** |
| --- | --- | --- |
| **Team Leader** | Facilite, retire obstacles, synchronise | ≈ Scrum Master |
| **Solution Developer** | Développe increments | Code + design |
| **Solution Tester** | Test continu, validation qualité | Lié au principe *Never Compromise Quality* |
| **Business Analyst** | Relie besoin ↔ solution | Crucial en Investigation |

## 🟧 **C. PROJECT / GOVERNANCE ROLES**

| **Rôle** | **Missions** | **Spécificités** |
| --- | --- | --- |
| **Project Manager** | Pilotage global, risques, reporting | Gouvernance agile (pas command & control) |
| **Technical Coordinator** | Architecture, normes, cohérence technique | ≈ architecte solution |

## 🟪 **D. SUPPORT ROLES** ***(nouveau groupe demandé)***

| **Rôle** | **Missions** | **Pourquoi important** |
| --- | --- | --- |
| **UX/UI Specialist** | Design, prototypes | Essentiel en Investigation |
| **DevOps Engineer** | CI/CD, automatisation, déploiement | Facilite livraison continue |
| **Security Specialist** | Risques, conformité, audits | Important dans secteurs régulés |
| **Data/Integration Engineer** | Interfaces, flux, qualité data | Assure cohérence solution |
| **Change Manager** | Adoption utilisateurs, formation | Évite rejet en Deployment |

🎯 Mnémotechnique Support : **SUDeSC = Security – UX – DevOps – Specialists – Change**

# **3. Cycle de vie AgilePM® v3 — 6 phases**

Chaque phase a **objectifs + acteurs clés + produits d’entrée/sortie**.

### **1. Pre-Project**

* Confirme qu’il y a un besoin réel
* Identifie sponsor, vision initiale  
  Acteurs : Sponsor, Visionary, PM

### **2. Feasibility**

* Étude rapide : valeur, risques, coûts, faisabilité technique
* Décision GO/NO GO  
  Acteurs : PM, Visionary, BA, Tech Coordinator

### **3. Foundations**

* Définition complète :
  + Vision
  + Architecture
  + Priorités
  + PRL (Prioritised Requirements List)
* Création des plans  
  Acteurs : Tous les rôles clés

🎯 Mnémotechnique : **“VAP” = Vision – Architecture – PRL**

### **4. Evolutionary Development**

Cœur du delivery → Timeboxes

Acteurs : Ambassador + Team Leader + Dev/Test + BA + Support

### **5. Deployment**

* Packaging
* Livraison
* Transition utilisateurs  
  Acteurs : Change Manager, DevOps, Visionary

### **6. Realisation**

* Mesure des bénéfices
* REX (retours d’expérience)  
  Acteurs : Sponsor + PM + Visionary

# **4. Tous les Work Products AgilePM® v3 (liste complète)**

## **A. Business Products**

* Business Case
* Business Vision
* **PRL (Prioritised Requirements List)**
* Benefits Assessment

## **B. Solution Products**

* Architecture Definition
* Solution Increments
* User Documentation
* Test Specifications
* Prototypes
* Interface Specifications

## **C. Management Products**

* Project Plan
* Delivery Plan
* **Timebox Plan**
* Risk Log
* Issue Log
* Quality Log
* Checkpoint Reports
* Highlight Reports
* Deployment Plan
* End-of-Phase Report
* Timebox Review Report

🎯 Mnémotechnique management : **“PRIQ-CHD” = Plan – Risk – Issue – Quality – Checkpoint – Highlight – Deployment**

# **5. Le Timebox AgilePM® v3 — OFFICIELLEMENT à 5 sous-ensembles**

Voici la structure exacte v3, **complète et détaillée**.

## 🔵 **1. Kick-off**

Objectif : aligner, planifier, sécuriser le périmètre.

🎯 Activités :

* Comprendre objectifs Timebox
* Finaliser Timebox Plan
* Confirmer Must/Should/Could
* Vérifier dépendances
* Préparer environnements (DevOps)

🎯 Rôles clés : Ambassador + BA + Team Leader + Dev/Test + Support

🎯 Work Products : Timebox Plan

## 🟢 **2. Investigation**

Objectif : comprendre précisément.

🎯 Activités :

* Ateliers facilités (workshops)
* Analyse détaillée des exigences
* Prototypes / UX
* Spécifications testables

🎯 Rôles : Ambassador, BA, UX, Dev/Test

## 🟡 **3. Refinement**

Objectif : développer.

🎯 Activités :

* Développement itératif
* Tests continus
* Daily Stand-ups
* Démonstrations intermédiaires

🎯 Rôles : Dev/Test + Team Leader

## 🟠 **4. Consolidation**

Objectif : stabiliser et préparer la livraison.

🎯 Activités :

* Correction bugs
* Tests complets
* Préparation documentation
* Checkpoint final

🎯 Rôles : Dev/Test + Tech Coordinator + Security

## 🔴 **5. Close**

Objectif : valider et apprendre.

🎯 Activités :

* Démo finale
* Validation Must Have
* Mise à jour logs
* Rétrospective

🎯 Rôles : Visionary + PM + Ambassador + Team Leader

# **6. Lien complet Rôles ↔ Phases ↔ Timeboxes**

| **Phase** | **Acteurs majeurs** | **Pourquoi** |
| --- | --- | --- |
| Pre-Project | Sponsor + PM | Initier projet |
| Feasibility | PM + BA + Tech | Analyser valeur |
| Foundations | Visionary + BA + Tech + PM | Définir bases |
| Evolutionary Development | Ambassador + Dev/Test + Team Leader | Construire solution |
| Deployment | DevOps + Change Manager | Livrer |
| Realisation | Sponsor + PM | Confirmer bénéfices |

# **7. Les 7 concepts “CRITIQUES” pour réussir l’examen**

1. MoSCoW (v3)  
   Must ≤ 60%, critères stricts, validation obligatoire.
2. Quality Never Compromised  
   On coupe le scope, jamais la qualité.
3. Timeboxing (fixe) + variable scope  
   Pilier d’AgilePM.
4. Foundations = Firm Foundations  
   Architecture + vision + PRL avant d’itérer.
5. **Iterative vs Incremental**
   * Incremental = ajouter des blocs
   * Iterative = améliorer ceux déjà créés
6. AgilePM ↔ Scrum alignment  
   Très demandé à l’examen.
7. Gouvernance agile  
   Rôles Business actifs + PM + Team Leader + Technical Coordinator.

# **8. Astuces mnémotechniques (spéciales examen)**

### 🎯 Timebox = **K-I-R-C-C**

Kick-off → Investigation → Refinement → Consolidation → Close

➡️ Phrase : “KIRCC : Kicker, Investiguer, Réfiner, Consolider, Conclure.”

### 🎯 Work Products Management = **PRIQ-CHD**

Plans – Risks – Issues – Quality – Checkpoint – Highlight – Deployment

### 🎯 Rôles Business = **S-V-A**

Sponsor – Visionary – Ambassador

### 🎯 Support = **SUDeSC**

Security – UX – DevOps – Specialists – Change

### 🎯 Foundations = **VAP**

Vision – Architecture – PRL

# **Conclusion – Ce qu’il faut absolument maîtriser**

✔ Timebox à 5 sous-ensembles (super fréquent à l’examen)

✔ MoSCoW v3

✔ Rôles (4 groupes + responsabilités)

✔ Work Products

✔ Cycle de vie complet

✔ Gouvernance agile

✔ Alignement AgilePM/Scrum

# 📘 **AgilePM® v3 – Work Products par Phase et Rôle**

| **Phase** | **Rôles principaux** | **Work Products principaux** | **Détails et rôle de chaque acteur** |
| --- | --- | --- | --- |
| **Pre-Project** | Sponsor, PM, Business Visionary | Business Case | **Sponsor** valide la justification stratégique et le financement, **PM** documente le plan de projet initial, **Visionary** fournit la première vision métier. |
| **Feasibility** | PM, BA, Technical Coordinator, Business Visionary | Business Case (révisé), Feasibility Report, Risk Log | **BA** clarifie les exigences métier, **Technical Coordinator** évalue faisabilité technique, **PM** met à jour le Risk Log, **Visionary** valide les objectifs et les priorités MoSCoW. |
| **Foundations** | PM, BA, Team Leader, Technical Coordinator, Business Visionary, QA | Business Vision, PRL, Solution Architecture Document, Project Plan, Quality Log, Risk Log | **BA** produit PRL détaillée, **TC** définit Architecture et SAD, **PM** complète Project Plan et Risk Log, **Team Leader** planifie Timeboxes, **QA** vérifie conformité, **Visionary** valide vision. |
| **Evolutionary Development – Timebox** | Team Leader, Dev, Tester, BA, Ambassador, DevOps, UX, Security | Timebox Plan, Solution Increments, Test Specifications, Prototypes, Daily Logs, Risk Log | **Team Leader** supervise le Timebox Plan, **Dev** développe increments, **Tester** crée et exécute tests, **BA/Ambassador** clarifie les stories, **DevOps** assure CI/CD, **UX** fournit maquettes, **Security** valide les aspects sécurité. |
| **Deployment** | PM, Team Leader, DevOps, Change Manager, QA | Deployment Plan, End-of-Phase Report, Highlight Reports | **DevOps** exécute la livraison, **PM** supervise, **QA** vérifie qualité finale, **Change Manager** prépare adoption utilisateurs, **Team Leader** coordonne. |
| **Realisation** | Sponsor, PM, Visionary, PMO | Benefits Assessment, End-of-Project Report | **Sponsor** valide la réalisation des bénéfices, **PM** compile l’End-of-Project Report, **Visionary** confirme satisfaction des objectifs métier, **PMO** assure reporting global. |

## 🔹 **Vision synthétique des Work Products par rôle**

| **Rôle** | **Work Products clés** |
| --- | --- |
| **Sponsor** | Business Case, Benefits Assessment |
| **Business Visionary** | Business Vision, Validation PRL, End-of-Project Report |
| **Business Ambassador / BA** | PRL, Stories clarifiées, Feedback Log |
| **Project Manager** | Project Plan, Risk Log, Issue Log, Highlight Reports, End-of-Phase Report |
| **Technical Coordinator** | Solution Architecture Document, Architecture Definition |
| **Team Leader** | Timebox Plan, Daily Logs, Coordination Docs |
| **Solution Developer** | Solution Increments, Code, Test Specs (unit & integration) |
| **Solution Tester** | Test Specifications, Test Logs, Defects Reports |
| **QA** | Quality Log, Conformity Check Reports |
| **DevOps** | CI/CD Pipelines, Deployment Plan, Logs |
| **UX/UI Specialist** | Prototypes, Wireframes, User Documentation |
| **Security Specialist** | Risk Logs, Security Checklists |
| **Change Manager** | Change Plan, Training Materials |

💡 **Conseil examen** :

* Souviens-toi : **Phase → Rôles → Work Products**.
* Chaque Timebox produit au minimum : **Timebox Plan + Solution Increment + Tests + Feedback**.
* Foundations = **fondations solides** → vision + PRL + architecture, jamais négliger.

# **1️⃣ La Case avec en tête : La Philosophie AgilePM® v3 (Philosophy)**

*“Best business value emerges when projects are aligned to clear business goals, deliver frequently and involve the collaboration of motivated and empowered people.”*

Traduction et explication :

La meilleure valeur métier est obtenue lorsque les projets :

1. **Sont alignés sur des objectifs métier clairs** (vision, bénéfices attendus, priorités).
2. **Livrent fréquemment** des incréments testés et exploitables.
3. **Impliquent des personnes motivées et responsabilisées** (tous les rôles collaborent, empowerment effectif).

Tips : Toujours relier chaque action au résultat métier concret.

Pièges : Livrer beaucoup de fonctionnalités “cool” sans valeur réelle → fail examen.

# **2️⃣ Principles (Principes comportementaux)**

| **Principe** | **Explication détaillée** | **Tips & A retenir** | **Pièges courants** |
| --- | --- | --- | --- |
| **Focalisation sur la valeur** | Chaque décision, chaque incrément doit apporter une valeur tangible. | Toujours vérifier la priorité MoSCoW avant chaque livraison. | Livrer des fonctionnalités non indispensables (“Could” ou “Won’t”) comme Must. |
| **Collaboration & Empowerment** | Encourager les décisions rapides, communication continue entre Business, Solution, Project, Support. | Tenir des ateliers réguliers, Daily Stand-ups, revues intermédiaires. | Ignorer l’avis des Business Ambassadors ou du Sponsor. |
| **Adaptabilité / Tailoring** | Adapter la méthode au contexte réel, éviter le copier-coller mécanique. | Tailoring = adapter les pratiques, phases, Work Products selon le projet. | Confondre Tailoring avec simplification excessive → perte de gouvernance. |
| **Livraison fréquente** | Livrer régulièrement des incréments fonctionnels dans la Timebox. | Chaque Timebox doit produire un incrément testable et utilisable. | Accumuler des changements sans livraison intermédiaire. |
| **Qualité non négociable** | Maintenir la qualité, ne jamais la réduire pour tenir les délais. | Tests continus, revues, QA active. | Baisser la qualité → fail examen ou incrément inutilisable. |

# **3️⃣ People (Rôles et responsabilités – 4 groupes)**

AgilePM® v3 définit **4 groupes de rôles**, chacun avec des responsabilités critiques :

### **A. Business Roles**

| **Rôle** | **Responsabilité clé** | **Phase principale** | **Tips & pièges** |
| --- | --- | --- | --- |
| Business Sponsor | Vision stratégique, financement, arbitrage global | Pre-Project / Foundations | Toujours confirmer que le projet apporte un bénéfice métier |
| Business Visionary | Objectifs métier, bénéfices attendus, validation des choix | Foundations / Review | Piège : négliger la validation des Must |
| Business Ambassador | Clarifie quotidiennement les besoins métier | Evolutionary Development / Timeboxes | Assurer feedback constant aux développeurs |
| Business Advisor | Expertise ponctuelle, conseils spécialisés | Toutes phases | Piège : l’ignorer lors des décisions critiques |

**Mnémo : SVAA → Sponsor, Visionary, Ambassador, Advisor**

### **B. Solution / Delivery Roles**

| **Rôle** | **Responsabilité clé** | **Phase principale** | **Tips & pièges** |
| --- | --- | --- | --- |
| Team Leader | Facilitation, coordination, suppression des obstacles | Evolutionary Development | Équivalent Scrum Master : ne pas micro-manager |
| Solution Developer | Construction et intégration de la solution | Dev / Refinement | Tips : intégrer CI/CD dès le début |
| Solution Tester | Validation continue, qualité technique | Dev / Review | Piège : tester trop tard → défauts accumulés |
| Solution Advisor | Expertise technique ponctuelle | Foundations / Dev | Important pour éviter les erreurs d’architecture |

**Mnémo : TL-D-T-SA**

### **C. Project / Governance Roles**

| **Rôle** | **Responsabilité clé** | **Phase principale** | **Tips & pièges** |
| --- | --- | --- | --- |
| Project Manager | Suivi planning, risques, reporting | Feasibility / Foundations | Toujours documenter décisions clés |
| Technical Coordinator | Architecture, normes, qualité technique | Foundations / Refinement | Piège : ne pas vérifier cohérence technique |
| Business Analyst | Traduction besoin ↔ solution | Foundations / Kick-off | Clarifier tous les must before Dev |
| Change Authority | Arbitrage scope rapide | Toutes | Essentiel pour éviter dérives |

**Mnémo : PM-TC-BA**

### **D. Support & Assurance Roles**

| **Rôle** | **Responsabilité clé** | **Phase principale** | **Tips & pièges** |
| --- | --- | --- | --- |
| QA | Vérification conformité & qualité | Foundations / Close | Ne jamais négliger revue QA avant livraison |
| Risk & Compliance Officer | Risques, RGPD, réglementaire | Kick-off / Close | Suivi continu des risques |
| PMO / Delivery Assurance | Reporting, budget, suivi | Foundations / Close | Vérifier reporting Timeboxes |
| DevSecOps / Release Manager | Pipelines CI/CD, releases | Dev / Close | Automatiser tests et déploiement |
| UX/UI Specialist | Maquettes, prototypage | Investigation | Tester UX avec utilisateurs réels |

**Mnémo : Q-R-P-D-U → Quality, Risk, PMO, DevSecOps, UX**

# **4️⃣ Process (Cycle de vie & Phases)**

6 phases principales :

1. **Pre-Project** : cadrage initial, validation besoin, Sponsor et PM.
2. **Feasibility** : étude faisabilité, risques, BV et PM.
3. **Foundations** : vision, architecture, plan global, PRL, PM, TL, TC.
4. **Evolutionary Development** : Timeboxes, incréments, Daily Stand-ups.
5. **Deployment** : mise en production, DevSecOps, QA, RCO.
6. **Realisation** : mesure bénéfices, REX, Sponsor et PM.

Tips : Toujours relier phase → rôle → work product.

Pièges : Confondre Evolutionary Development avec Deployment ou Realisation.

# **5️⃣ Products (Work Products – par phase et Timebox)**

| **Phase / Timebox** | **Business** | **Solution** | **Management** | **Tips / Pitfalls** |
| --- | --- | --- | --- | --- |
| Pre-Project | Business Case | - | Project Plan | Attention : Business Case incomplet → dérive projet |
| Feasibility | Business Vision, Benefits Assessment | - | Risk Log | Vérifier alignement MoSCoW Must/Should |
| Foundations | PRL, Business Vision | Architecture Definition, SAD | Delivery Plan, Quality Log | Documenter décisions architecture |
| Evolutionary Development (Timeboxes) | Updates PRL | Solution Increments, Prototypes, Test Specs | Timebox Plan, Issue Log, Checkpoint Report | Piège : ne pas mettre à jour Risk / Quality Logs |
| Deployment | - | User Documentation, Release Notes | Deployment Plan, Highlight Report | Assurer tests finaux QA et conformité |
| Realisation | Benefits Realization | - | Lessons Learned, REX | Piège : négliger REX = perte apprentissage |

**Tips :** Toujours lier chaque Work Product à un rôle responsable.

# **6️⃣ Timebox AgilePM v3 – 5 sous-ensembles**

Chaque Timebox = **Kick-off → Investigation → Refinement → Consolidation → Close**

| **Sous-ensemble** | **Objectif** | **Rôles principaux** | **Work Products** | **Tips / A retenir** | **Pièges** |
| --- | --- | --- | --- | --- | --- |
| Kick-off | Alignement, planification, assignation | TL, BA, PM, Sponsor, RCO | Timebox Plan, PRL Must/Should | Clarté sur objectifs, rôle de chacun | Confondre Kick-off avec Investigation |
| Investigation | Clarifier exigences, prototypage | BA, Dev, UX | Models, Story Details, Initial Tests | Le prototype = outil de validation | Ne pas impliquer Business Ambassador → risque dérive |
| Refinement | Ajuster solution, priorités MoSCoW | Dev, Testers, TL, BA | Updated Increments, Risk Log | Prioriser Must <60%, ajuster Should | Confondre Should et Must |
| Consolidation | Intégration et tests finaux | Dev, Testers, QA, DevSecOps | Solution Increment, Test Results | Vérifier qualité complète, coverage tests | Sauter tests ou QA partielle = fail examen |
| Close | Démo finale, rétrospective, reporting | TL, PM, BA, Sponsor, RCO | Timebox Report, Updated Risk & Quality Logs | Documenter apprentissages & actions | Négliger Close = perte traçabilité |

**Mnémo : K-I-R-C-C → Kick, Investigate, Refine, Consolidate, Close**

# **7️⃣ Pratiques clés & concepts à maîtriser pour l’examen**

| **Concept** | **Explication** | **Tips / Essentiel** | **Pièges** |
| --- | --- | --- | --- |
| MoSCoW v3 | Must ≤ 60%, critères stricts, validation obligatoire. | Identifier Must absolus | Confondre Must et Should |
| Tailoring | Adapter cadre au projet | Ne jamais copier-coller | Supprimer phases essentielles |
| Build Incrementally from Firm Foundations | Ne jamais bricoler | Foundations = socle solide | Débuter Dev sans vision/archi |
| Iterative vs Incremental | Iterative = amélioration, Incremental = ajout | Savoir différence | Confondre les deux = questions examen |
| Governance Model | Sponsors, Ambassadors, PM, TC = rôles clairs | Respecter reporting & arbitrage | Ignorer Sponsor = fail examen |
| Quality Never Compromised | Maintenir QA sur tout incrément | Tests continus | Livrer incrément non testé |
| AgilePM ↔ Scrum Alignment | Timebox = Sprint, Team Leader = Scrum Master, Ambassador = Product Owner | Savoir mapping | Mélanger rôles → réponse examen fausse |

# **8️⃣ Tips généraux pour réussir la certification**

* Documenter **apprentissages et rétrospectives** à chaque Close.
* Relier **Phase → Timebox → Rôle → Work Product → Valeur**.
* Connaître tous les **21 concepts clés** et 5 sous-ensembles de Timebox.
* Maintenir **qualité constante** et respecter **MoSCoW v3**.
* Savoir aligner AgilePM ↔ Scrum pour chaque rôle et artefact.

# 📘 **AgilePM® v3 – Work Products par Phase et Rôle** (Répétition du tableau pour clarté)

| **Phase** | **Rôles principaux** | **Work Products principaux** | **Détails et rôle de chaque acteur** |
| --- | --- | --- | --- |
| **Pre-Project** | Sponsor, PM, Business Visionary | Business Case | **Sponsor** valide la justification stratégique et le financement, **PM** documente le plan de projet initial, **Visionary** fournit la première vision métier. |
| **Feasibility** | PM, BA, Technical Coordinator, Business Visionary | Business Case (révisé), Feasibility Report, Risk Log | **BA** clarifie les exigences métier, **Technical Coordinator** évalue faisabilité technique, **PM** met à jour le Risk Log, **Visionary** valide les objectifs et les priorités MoSCoW. |
| **Foundations** | PM, BA, Team Leader, Technical Coordinator, Business Visionary, QA | Business Vision, PRL, Solution Architecture Document, Project Plan, Quality Log, Risk Log | **BA** produit PRL détaillée, **TC** définit Architecture et SAD, **PM** complète Project Plan et Risk Log, **Team Leader** planifie Timeboxes, **QA** vérifie conformité, **Visionary** valide vision. |
| **Evolutionary Development – Timebox** | Team Leader, Dev, Tester, BA, Ambassador, DevOps, UX, Security | Timebox Plan, Solution Increments, Test Specifications, Prototypes, Daily Logs, Risk Log | **Team Leader** supervise le Timebox Plan, **Dev** développe increments, **Tester** crée et exécute tests, **BA/Ambassador** clarifie les stories, **DevOps** assure CI/CD, **UX** fournit maquettes, **Security** valide les aspects sécurité. |
| **Deployment** | PM, Team Leader, DevOps, Change Manager, QA | Deployment Plan, End-of-Phase Report, Highlight Reports | **DevOps** exécute la livraison, **PM** supervise, **QA** vérifie qualité finale, **Change Manager** prépare adoption utilisateurs, **Team Leader** coordonne. |
| **Realisation** | Sponsor, PM, Visionary, PMO | Benefits Assessment, End-of-Project Report | **Sponsor** valide la réalisation des bénéfices, **PM** compile l’End-of-Project Report, **Visionary** confirme satisfaction des objectifs métier, **PMO** assure reporting global. |

## 🔹 **Vision synthétique des Work Products par rôle** (Répétition du tableau pour clarté)

| **Rôle** | **Work Products clés** |
| --- | --- |
| **Sponsor** | Business Case, Benefits Assessment |
| **Business Visionary** | Business Vision, Validation PRL, End-of-Project Report |
| **Business Ambassador / BA** | PRL, Stories clarifiées, Feedback Log |
| **Project Manager** | Project Plan, Risk Log, Issue Log, Highlight Reports, End-of-Phase Report |
| **Technical Coordinator** | Solution Architecture Document, Architecture Definition |
| **Team Leader** | Timebox Plan, Daily Logs, Coordination Docs |
| **Solution Developer** | Solution Increments, Code, Test Specs (unit & integration) |
| **Solution Tester** | Test Specifications, Test Logs, Defects Reports |
| **QA** | Quality Log, Conformity Check Reports |
| **DevOps** | CI/CD Pipelines, Deployment Plan, Logs |
| **UX/UI Specialist** | Prototypes, Wireframes, User Documentation |
| **Security Specialist** | Risk Logs, Security Checklists |
| **Change Manager** | Change Plan, Training Materials |

# **2️⃣ Concepts clés AgilePM® v3 – 21 éléments essentiels**

Voici les concepts les plus testés à l’examen, avec explications, tips et pièges courants :

| **#** | **Concept** | **Description / Tip** | **Piège courant** |
| --- | --- | --- | --- |
| 1 | **MoSCoW v3 – Must ≤ 60%** | Priorisation stricte : Must (indispensable), Should (important), Could (optionnel), Won’t (pas pour cette TB). | Confondre Should et Must ; dépasser 60% Must |
| 2 | **Tailoring** | Adapter AgilePM à contexte projet avant adoption. | Copier-coller la méthode sans ajustement |
| 3 | **Build Incrementally from Firm Foundations** | Commencer par Foundations solides (Vision + Architecture + PRL). | Bricoler la solution sans base solide |
| 4 | **Iterative vs Incremental** | Incremental = ajouter blocs fonctionnels ; Iterative = améliorer blocs existants. | Confondre pour l’examen |
| 5 | **AgilePM Governance Model** | Sponsors impliqués, Ambassadors présents, PM pilote delivery, TC responsable technique. | Rôles flous = erreurs d’examen |
| 6 | **Quality Never Compromised** | Jamais réduire qualité pour respecter délai. | Négliger tests ou normes → fail examen |
| 7 | **AgilePM ↔ Scrum Alignment** | Timebox = Sprint, Team Leader = Scrum Master, Business Ambassador ≈ Product Owner, Daily Stand-up ≈ Daily Scrum, Review ≈ Sprint Review, Retrospective ≈ Sprint Retrospective. | Confondre rôles et événements |
| 8 | **Timebox 5 sous-ensembles** | Kick-off, Investigation, Refinement, Consolidation, Close. | Ignorer Refinement ou Consolidation |
| 9 | **PRL (Prioritised Requirements List)** | Liste priorisée de fonctionnalités pour la Timebox. | Ne pas mettre à jour chaque TB |
| 10 | **Increment** | Livrable partiel fonctionnel par Timebox. | Livrer incomplet ou non testé |
| 11 | **Risk Log / Issue Log** | Suivi des risques et problèmes. | Oublier de mettre à jour après chaque TB |
| 12 | **Lessons Learned / Retrospective** | Documenter apprentissages pour TB suivante et projet global. | Ne pas capitaliser apprentissages |
| 13 | **Daily Stand-up** | Communication rapide pour lever obstacles. | Micro-management ou longue réunion |
| 14 | **Facilitated Workshops** | Ateliers pour lever ambiguïtés et valider conception. | Ignorer participation métier |
| 15 | **Prototyping** | Maquettes ou proof-of-concept pour clarification. | Tester trop tard ou pas assez |
| 16 | **Business Case** | Document initial : objectifs, budget, bénéfices. | Ne pas vérifier alignement avec sponsor |
| 17 | **Architecture Definition / SAD** | Définition technique et fonctionnelle de la solution. | Trop détaillé ou incomplet |
| 18 | **Deployment Plan** | Planifier mise en prod et release. | Ignorer étapes pré-prod ou rollback |
| 19 | **Iterative Testing** | Tests continus sur incréments. | Ne pas automatiser, ignorer anomalies |
| 20 | **Benefits Assessment** | Mesurer gains post-déploiement. | Ne pas comparer avec objectifs initiaux |
| 21 | **Common Sense & Pragmatism** | Toujours adapter méthode, prioriser valeur et simplicité. | Trop théorique = échec pratique / examen |

💡 **Tips généraux pour l’examen :**

* Connaître les **5 sous-ensembles Timebox** sur le bout des doigts.
* Savoir **qui fait quoi** dans chaque phase / Timebox.
* Toujours appliquer **MoSCoW v3** et respecter les limites de Must.
* Ne jamais **compromettre la qualité**.
* Savoir aligner AgilePM ↔ Scrum pour répondre aux questions de mapping.
* Documenter **Work Products**, **Lessons Learned**, **Risk Log**.
* Maîtriser les **21 concepts clés** avec exemples concrets pour répondre rapidement.

# 📘 **AgilePM® v3 – Guide Complet de Révision Plus ++**

## **1. Introduction générale**

AgilePM® v3 combine **agilité moderne** et **rigueur projet** :

* **Rigueur projet** : gouvernance, rôles clairs, Work Products, reporting
* **Agilité** : Timeboxing, MoSCoW, itération incrémentale, feedback rapide
* **Intégration** : Scrum, DevOps, CI/CD

### 🎯 Objectif du guide

Réviser efficacement, comprendre **la logique du framework**, **anticiper les pièges** et réussir l’examen.

## **2. Les 4 groupes de rôles (avec tips et pièges)**

### 🟦 **A. Business Roles**

| **Rôle** | **Mission** | **Tip** | **Warning / Piège** | **À retenir** |
| --- | --- | --- | --- | --- |
| Business Sponsor | Vision stratégique, finance, décisions | Toujours vérifier disponibilité sponsor dès Pre-Project | Sponsor absent = risque décalage stratégique | "Sponsor = décision + financement" |
| Business Visionary | Objectifs, bénéfices, validation vision | Impliquer dès Feasibility | Confondre avec BA → erreur fréquente | "Visionary = garder le cap sur le besoin" |
| Business Ambassador | Porte voix métier quotidien | Disponible dans toutes Timeboxes | Confondre avec Solution Tester = attention | "Ambassador = voix du métier" |
| Business Advisor | Expertise ponctuelle | Ne pas oublier d’aligner avec PRL | Ajouter trop tôt → surcharge info | "Advisor = expert ponctuel" |

**Mnémotechnique** : **SVAA** → Sponsor, Visionary, Ambassador, Advisor

**Tip examen** : toujours lier chaque rôle à **phase et Timebox**.

### 🟩 **B. Solution / Delivery Roles**

| **Rôle** | **Mission** | **Tip** | **Warning** | **À retenir** |
| --- | --- | --- | --- | --- |
| Team Leader | Facilite, retire obstacles | TL = Scrum Master v3 | Ne pas micro-manager Dev | "TL = facilitateur et coordinateur" |
| Solution Developer | Développe et intègre | Toujours utiliser CI/CD | Code livré sans test = piège critique | "Dev = construire et tester" |
| Solution Tester | QA continue | Tester dès construction | Attendre la fin = bug coûteux | "Tester = qualité intégrée" |
| Business Analyst | Traduction besoin ↔ solution | Clarifier stories avant Dev | Confondre avec BA2 projet = erreur | "BA = pont métier-technique" |

**Tip examen** : associer TL/Dev/Tester à chaque **Timebox sub-phase**.

### 🟧 **C. Project / Governance Roles**

| **Rôle** | **Mission** | **Tip** | **Warning** | **À retenir** |
| --- | --- | --- | --- | --- |
| Project Manager | Plan, suivi, reporting | Toujours maintenir Risk Log | PM micro-manage Dev = anti-agile | "PM = gouvernance & reporting" |
| Technical Coordinator | Architecture, normes | Vérifier conformité technique | Ignorer TC = risque technique | "TC = garde-fou technique" |
| Business Analyst (BA2) | Exigences globales | Soutenir TL et Dev | Confondre avec BA métier | "BA2 = exigences globales" |
| Change Authority | Décisions rapides scope | Clarifier limits | Autorité floue = conflit | "Change Authority = scope rapide" |

### 🟪 **D. Support & Assurance Roles (Nouveau)**

| **Rôle** | **Mission** | **Tip** | **Warning** | **À retenir** |
| --- | --- | --- | --- | --- |
| UX/UI Specialist | Prototypes, tests UX | Toujours impliquer dès Investigation | UX ignoré = adoption faible | "UX = expérience utilisateur" |
| DevOps Engineer | CI/CD, pipelines | Automatiser tout ce qui peut l’être | Ignorer = livraison lente | "DevOps = livraison rapide & fiable" |
| Security Specialist | Risques & conformité | Checker chaque incrément | Ignorer RGPD ou ISO = fail | "Security = non-négociable" |
| Data/Integration Specialist | Connectivité et qualité | Vérifier intégration continue | Mauvaise data = échec reporting | "Data = intégrité et connectivité" |
| Change Manager | Adoption, formation | Préparer utilisateurs tôt | Trop tard → résistance | "Change = adoption + formation" |

**Mnémotechnique** : **SUDeSC** → Security, UX, DevOps, Specialists, Change

## **3. Cycle de vie du projet AgilePM v3**

**Phases clés et tips :**

| **Phase** | **Objectif** | **Rôles principaux** | **Tips** | **Warnings** | **Work Products clés** |
| --- | --- | --- | --- | --- | --- |
| Pre-Project | Décider si le projet doit exister | Sponsor, PM | Clarifier ROI dès le départ | Ne pas initier projet sans Sponsor | Business Case |
| Feasibility | Vérifier valeur et risques | PM, BV, BA2, TC | Focus risques + coûts | Feasibility vague → scope flou | Feasibility Report, Risk Log |
| Foundations | Vision, architecture, plan global | TL, BV, BA2, PM, TC | Préparer PRL clair | Ignorer MoSCoW = dérive | PRL, SAD, Project Plan, Risk/Quality Logs |
| Evolutionary Dev | Livrer incréments via Timeboxes | TL, Dev, Testers, BA | Timeboxes stricts | Ignorer refinements = défauts | Timebox Plan, Solution Increments, Test Specs |
| Deployment | Production | PM, TL, DevOps | Checklist complète | Ignorer QA = incidents | Deployment Plan, Highlight Reports |
| Realisation | Bénéfices | Sponsor, PM, Visionary | Mesurer KPI | Ne pas documenter = leçons perdues | Benefits Assessment, End-of-Project Report |

## **4. Work Products – Vue complète par phase et rôle**

| **Phase** | **Business** | **Solution** | **Project** | **Support** |
| --- | --- | --- | --- | --- |
| Pre-Project | Business Case | – | Project Plan | – |
| Feasibility | Business Case révisé | – | Risk Log, Feasibility Report | QA, Security check |
| Foundations | Business Vision | SAD, Architecture Definition | Project Plan, Timebox Plan | QA, PMO, RCO |
| Evolutionary Dev | PRL | Solution Increments, Test Specs, Prototypes | Timebox Plan, Risk Log | UX, DevOps, Security |
| Deployment | – | – | Deployment Plan, Highlight Report | QA, DevOps, Change Manager |
| Realisation | Benefits Assessment | – | End-of-Project Report | PMO |

## **5. La Timebox AgilePM® v3 – 5 sous-ensembles**

| **Sous-ensemble** | **Objectif** | **Rôles clés** | **Work Products** | **Tip** | **Piège** |
| --- | --- | --- | --- | --- | --- |
| Kick-Off | Clarification PRL, plan Timebox | BA, Ambassador, TL, PM | Timebox Plan v1, Risk Log | Impliquer tous stakeholders | Lancer sans clarifier Must/Should/Could |
| Investigation | Analyse & prototypage | BA, Dev, UX, TC, Security | Models, Stories clarifiées | Détecter risques tôt | Ignorer QA ou UX = adoption faible |
| Refinement | Ajustement MoSCoW, spécifications | BA, Dev, TL, PM | Ajusted PRL, Test Specs | Prioriser correct | Ne pas réviser PRL → erreurs incrément |
| Consolidation | Construction finale, tests | Dev, Testers, TL, DevOps | Solution Increments, CI/CD logs | QA continue | Construire sans consolidation = bug en prod |
| Close | Démo, validation, rétrospective | Sponsor, Visionary, PM, Change Manager | Timebox Report, Risk Log, PRL Update | Documenter apprentissages | Clore sans feedback → perte apprentissage |

**Mnémotechnique** : **KIRCC** → Kick-Off → Investigation → Refinement → Consolidation → Close

**À retenir examen** : chaque Timebox doit **livrer un incrément testé et validé** tout en respectant MoSCoW et qualité.

## **6. 7 concepts essentiels (tips + pièges)**

1. **MoSCoW v3** – Must ≤ 60% → piège courant : confondre Should et Must
2. **Tailoring** – adapter méthode, ne pas copier-coller
3. **Build Incrementally from Firm Foundations** – éviter "bricolage"
4. **Iterative vs Incremental** – comprendre différence pour examen
5. **Governance Model** – Sponsor, Ambassadors, PM, TC → rôle clair
6. **Quality Never Compromised** – jamais réduire qualité, sinon fail examen
7. **AgilePM ↔ Scrum Alignment** – Timebox = Sprint, Team Leader = Scrum Master

## **7. Cas d’usage détaillés avec tips**

### Banque – Module KYC

* **Kick-Off** : BA définit Must (scan pièce, selfie, matching)
* **Investigation** : prototypes OCR, UX test
* **Refinement** : ajustement seuils, test automatisés
* **Consolidation** : microservices, CI/CD
* **Close** : audit RGPD, Timebox Report

Tip : cycles courts = dérisquer réglementaire

Piège : ignorer UX = adoption faible

### Énergie – Dashboard smart-meters

* KO : KPIs et alertes définies
* Investigation : maquettes UX, règles agrégées
* Refinement : ajuster calculs KPIs
* Consolidation : APIs + UI, tests intégrés
* Close : export PDF validé, KPI reporting

**À retenir** : Investigation = clé dérisquer technique

## **8. Schéma Mermaid Timebox**

Extrait de code

flowchart LR  
A[Kick-Off] --> B[Investigation]  
B --> C[Refinement]  
C --> D[Consolidation]  
D --> E[Close]

## **9. Conclusion**

Pour réussir l’examen :

* Maîtriser **Timebox 5 sous-ensembles**, MoSCoW v3, Work Products par phase
* Connaître **rôles + responsabilités**
* Ne jamais compromettre **qualité**
* Savoir aligner AgilePM ↔ Scrum
* Documenter apprentissages et rétrospectives

# 📘 **21 Cas & Tips Examen**

## **1. Règles d’or pour l’examen**

Pour réussir, tu dois absolument maîtriser :

1. **Timebox 5 sous-ensembles** : Kick-Off → Investigation → Refinement → Consolidation → Close
2. **MoSCoW v3** : Must, Should, Could, Won’t (Must ≤ 60% du contenu)
3. **Work Products par phase et rôle**
4. **Rôles + responsabilités (4 groupes)**
5. **Quality Never Compromised** : jamais sacrifier la qualité
6. **Aligner AgilePM ↔ Scrum** : Team Leader = Scrum Master, Timebox = Sprint
7. **Documenter apprentissages / rétrospectives**

**Tip examen** : chaque question pratique testera l’interaction **rôle → phase → Timebox → Work Product → qualité / MoSCoW**.

## **2. Timebox 5 sous-ensembles – Explications & Tips**

| **Sous-ensemble** | **Objectif** | **Rôles clés** | **Work Products** | **Tip** | **Piège** |
| --- | --- | --- | --- | --- | --- |
| Kick-Off | Clarifier PRL & objectifs | BA, Ambassador, TL, PM | Timebox Plan v1, Risk Log | Impliquer tous stakeholders | Lancer sans clarifier Must/Should/Could |
| Investigation | Analyse, prototypage, réduction incertitudes | BA, Dev, UX, TC, Security | Models, Stories clarifiées | Détecter risques tôt | Ignorer QA ou UX = adoption faible |
| Refinement | Ajuster MoSCoW, spécifications | BA, Dev, TL, PM | Ajusted PRL, Test Specs | Prioriser correctement | Ne pas réviser PRL → erreurs incrément |
| Consolidation | Construction finale, tests | Dev, Testers, TL, DevOps | Solution Increments, CI/CD logs | QA continue | Construire sans consolidation = bug prod |
| Close | Démo, validation, rétrospective | Sponsor, Visionary, PM, Change Manager | Timebox Report, Risk Log, PRL Update | Documenter apprentissages | Clore sans feedback → perte apprentissage |

**À retenir examen** : chaque Timebox doit **livrer un incrément testé et validé**, respecter **MoSCoW**, et produire les **Work Products associés**.

## **3. 21 Cas Pratiques par Secteur**

### **Banque**

1. **Onboarding KYC**
   * Kick-Off : BA définit Must (scan pièce, selfie, matching)
   * Investigation : prototypes OCR + tests UX
   * Refinement : seuils sécurité ajustés
   * Consolidation : microservices, tests unitaires & intégration
   * Close : audit QA + Timebox Report

Tip : cycles courts = contrôle réglementaire

Piège : ignorer UX = adoption faible

1. **Paiement mobile**
   * Must : authentification + transaction sécurisée
   * Could : notifications push
   * Won’t : reporting avancé immédiat
   * Test sur sandbox
   * Démo finale validée par Business

**À retenir** : bien distinguer Must/Should/Could

1. **Crédit en ligne**
   * Investigation : analyse risque crédit
   * Refinement : scoring ajusté
   * Consolidation : API scoring + tests automatisés
   * Close : rétrospective lessons learned

**Tip** : QA et Security = priorité

### **Énergie**

1. **Dashboard smart-meters**
   * Must : courbe + alertes
   * Investigation : maquettes UX + KPIs
   * Refinement : ajuster formats
   * Consolidation : intégration API + tests
   * Close : export PDF et KPI reporting

**À retenir** : Investigation = clé dérisquer technique

1. **Facturation automatisée**
   * Kick-Off : définir SLA
   * Investigation : prototype calcul factures
   * Refinement : ajustement MoSCoW
   * Consolidation : automatisation factures
   * Close : QA, rétrospective
2. **Prévision consommation**
   * Must : calcul consommation
   * Could : prédiction avancée
   * Refinement : algorithmes améliorés
   * Close : Timebox Report

### **Gouvernement / Public**

1. **Formulaire e-administration**
   * Must : accessibilité RGAA, sécurité ISO 27001
   * Investigation : tests UX population cible
   * Refinement : OTP, signature électronique
   * Consolidation : tests fonctionnels
   * Close : préproduction + lessons learned
2. **Carte nationale d’identité**
   * Kick-Off : définir Must/Should
   * Investigation : capture données
   * Refinement : correction workflow
   * Consolidation : intégration CI/CD
   * Close : audit conformité
3. **Permis de conduire en ligne**
   * Investigation : prototype UX
   * Refinement : validation business
   * Consolidation : intégration API
   * Close : démo finale

### **Télécom / IT**

1. **Portail client**
   * Must : login sécurisé + profil
   * Investigation : maquettes UI
   * Refinement : validation fonctionnalités
   * Consolidation : tests, CI/CD pipelines
   * Close : audit sécurité
2. **Facture électronique**
   * Kick-Off : SLA défini
   * Investigation : prototype PDF
   * Refinement : règles taxes
   * Consolidation : intégration systèmes
   * Close : Timebox Report
3. **Service SMS / USSD**
   * Must : workflow transactionnel
   * Could : notifications
   * Close : rétrospective

### **Retail / E-commerce**

1. **Module panier**
   * Kick-Off : Must = ajout, suppression produits
   * Investigation : UX prototypes
   * Refinement : règles panier
   * Consolidation : tests intégrés
   * Close : rétrospective
2. **Paiement en ligne**
   * Must : sécuriser transactions
   * Investigation : intégration API bancaire
   * Consolidation : tests end-to-end
   * Close : QA + lessons learned
3. **Gestion comptes client**
   * Kick-Off : Must = création compte
   * Investigation : stories
   * Consolidation : CI/CD
   * Close : Timebox Report

### **Santé / Pharma**

1. **Téléconsultation**
   * Must : vidéo + chat sécurisé
   * Could : messagerie interne
   * Consolidation : tests sécurité
   * Close : audit compliance
2. **Prescription électronique**
   * Investigation : maquettes UX
   * Refinement : règles validation prescriptions
   * Consolidation : tests intégrés
   * Close : Timebox Report
3. **Gestion stock médicaments**
   * Kick-Off : définir SLA
   * Consolidation : intégration ERP
   * Close : rétrospective

### **Transport / Logistique**

1. **Suivi colis**
   * Kick-Off : définir tracking Must
   * Investigation : UX maps
   * Refinement : ajustement API
   * Close : QA + Timebox Report
2. **Planification flotte**
   * Must : calcul routes
   * Could : optimisation avancée
   * Consolidation : tests & intégration
   * Close : lessons learned
3. **Application mobilité**
   * Kick-Off : user login
   * Investigation : prototype
   * Refinement : ajustement UX
   * Consolidation : tests mobiles
   * Close : démo finale

## **4. Tips & Pièges pour l’examen**

1. **Timebox = cœur du delivery** → respecter 5 sous-ensembles
2. **MoSCoW v3** : Must ≤ 60%, Should / Could / Won’t bien compris
3. **Work Products** : connaître par phase + rôle
4. **Rôles** : Business / Solution / Project / Support → identifier qui fait quoi
5. **Quality Never Compromised** → Timebox sans QA = fail
6. **Aligner AgilePM ↔ Scrum** → Timebox = Sprint, Team Leader = Scrum Master
7. **Documenter apprentissages** → Close Timebox & Rétrospective

# 📘 **Aligner AgilePM® v3 ↔ Scrum**

AgilePM et Scrum sont tous deux des approches agiles, mais **AgilePM v3 est plus gouverné et orienté projets** alors que Scrum est plus léger et centré sur l’équipe. Comprendre leur correspondance est **essentiel pour l’examen**, notamment dans les questions de mapping rôles, artefacts et événements.

## **1️⃣ Correspondance des rôles**

| **AgilePM® v3** | **Rôle Scrum** | **Explication / Tip examen** |
| --- | --- | --- |
| **Team Leader (TL)** | Scrum Master | Facilite le travail de l’équipe, retire les obstacles, organise les cérémonies. Tip : TL gère aussi la gouvernance delivery, pas juste l’équipe. |
| **Business Ambassador (BA)** | Product Owner | Porte la voix du métier, priorise les fonctionnalités via MoSCoW. Tip : ne jamais confondre BA et Business Sponsor. |
| **Solution Developer / Tester (SD/ST)** | Development Team | Construit, teste et livre les incréments. Tip : DevOps + QA souvent inclus dans Dev Team en Scrum. |
| **Business Sponsor / Visionary** | Stakeholders | Fournissent vision et direction. Tip : Scrum ne formalise pas le Sponsor mais AgilePM le fait, attention à l’examen. |
| **Support & Assurance Roles (QA, RCO, DevSecOps, UX)** | Non formalisé | Ces rôles support dans AgilePM n’ont pas d’équivalent direct en Scrum mais sont essentiels pour gouvernance et conformité. |

**⚠️ Pitfall examen** : Ne pas aligner TL avec Scrum Master = mauvaise interprétation des responsabilités.

## **2️⃣ Correspondance des artefacts / Work Products**

| **AgilePM® v3** | **Scrum** | **Explication** |
| --- | --- | --- |
| **Timebox Plan** | Sprint Backlog | Contient tâches / stories prévues pour la Timebox. Tip : Timebox Plan est plus formel, validé par PM et BA. |
| **Solution Increments** | Increment / Done | Les fonctionnalités livrées à la fin de la Timebox / Sprint. Tip : vérifier “Definition of Done” = QA, tests, documentation. |
| **Prioritised Requirements List (PRL)** | Product Backlog | Liste priorisée des exigences. Tip : AgilePM utilise MoSCoW pour priorisation, Scrum Product Backlog est plus flexible. |
| **Timebox Report** | Sprint Review Report | Résume les résultats et décisions. Tip : inclure feedback, issues, risques, actions corrective. |

**⚠️ Pitfall examen** : Confondre Timebox Plan et PRL = erreur classique.

## **3️⃣ Correspondance des événements / cérémonies**

| **AgilePM® v3** | **Scrum** | **Explication / Tip examen** |
| --- | --- | --- |
| **Timebox Kick-Off** | Sprint Planning | Définir objectifs, PRL à traiter, assigner rôles. Tip : inclure risques, contraintes, rôle du Sponsor. |
| **Investigation / Refinement** | Backlog Refinement / Daily Scrum | Clarification, prototypage, priorisation, ajustements continus. Tip : Daily stand-up est implicite dans Dev / TL. |
| **Consolidation** | Sprint Work / Daily Scrum | Construction, tests, intégration continue. Tip : inclure QA et DevOps pour alignement qualité. |
| **Close / Review** | Sprint Review & Retrospective | Démo au métier, feedback, leçons apprises. Tip : Close inclut Risk Log et Timebox Report, Scrum ne formalise pas ces documents. |

**À retenir examen** : AgilePM = Scrum + gouvernance formelle + Work Products supplémentaires.

## **4️⃣ Tips pour l’examen**

1. **Timebox = Sprint** mais AgilePM ajoute 5 sous-ensembles → comprendre leur séquence.
2. **Team Leader ≠ Dev** : rôle de facilitation + reporting.
3. **MoSCoW priorisation** = clé → Must/Should/Could/Won’t. Scrum n’utilise pas cette granularité.
4. **Support & Assurance Roles** : Scrum n’en parle pas → inclure QA, RCO, UX dans réponses examen.
5. **Align Work Products** : PRL → Product Backlog, Timebox Plan → Sprint Backlog, Increments → Done.
6. **Documentation obligatoire** : AgilePM exige Timebox Report et Risk Log → Scrum non.
7. **Ne jamais sacrifier la qualité** : QA, tests, CI/CD intégration.

## **5️⃣ Pièges classiques à éviter**

* Confondre BA (Business Ambassador) avec Business Sponsor.
* Croire que la Timebox = Sprint Scrum simple (oublier Kick-Off, Investigation, Refinement, Consolidation, Close).
* Ignorer les Work Products et les documents de gouvernance dans AgilePM.
* Ne pas inclure les rôles de support pour conformité, sécurité et UX.

## **6️⃣ À retenir**

* **AgilePM® v3** = Scrum + gouvernance projet + 5 sous-ensembles Timebox + Work Products formels.
* Alignement clair **rôles → artefacts → cérémonies**.
* **MoSCoW v3** et **Quality Never Compromised** : essentiels.
* **Timebox Report et Risk Log** = différence clé avec Scrum.
* Examen = souvent scenario pratique → identifier **qui fait quoi dans quelle Timebox et quel Work Product**.

# 📊 **Tableau AgilePM® v3 ↔ Scrum – Rôles, Artefacts, Timeboxes, Work Products, Tips et Pièges**

| **AgilePM® v3** | **Scrum** | **Work Products / Artefacts** | **Événement / Timebox** | **Tip examen** | **Piège / A éviter** | **Mini cas d’usage** |
| --- | --- | --- | --- | --- | --- | --- |
| **Team Leader (TL)** | Scrum Master | Timebox Plan, Checkpoint Report | Kick-Off, Daily Stand-up | Facilite, retire obstacles, gère flux | Confondre avec Dev | TL organise Kick-Off pour définir PRL à traiter |
| **Business Ambassador (BA)** | Product Owner | PRL, Benefits Assessment | Kick-Off, Review | Priorise via MoSCoW, clarifie besoins métier | Confondre avec Business Sponsor | BA valide Must Have pendant Review |
| **Business Sponsor (BS)** | Stakeholder | Business Case, Benefits Assessment | Pre-Project, Review | Fournit vision stratégique | Ignorer rôle décisionnel | Sponsor approuve démarrage du projet |
| **Business Visionary (BV)** | Stakeholder | Business Vision, Benefits Assessment | Foundations, Review | Définit vision cible, valide décisions | Oublier validation vision globale | BV confirme objectifs lors de Foundations |
| **Solution Developer (SD)** | Dev Team | Solution Increment, Test Specs | Investigation, Refinement, Consolidation | Code + test + intégration continue | Oublier tests continus | Dev produit micro-service et tests unitaires |
| **Solution Tester (ST)** | Dev Team | Test Specs, QA Log | Development, Consolidation | Valide incréments | Ignorer critères qualité | Teste chaque incrément avant Review |
| **Business Analyst (BA2)** | - | Stories, Models, PRL updates | Kick-Off, Investigation | Traduit besoin → solution | Mélanger avec BA | BA2 crée modèles et clarifie PRL pour Dev |
| **Project Manager (PM)** | - | Project Plan, Highlight Reports, Risk Log | Kick-Off, Close | Suit risques, planning, reporting | Micro-management | PM valide Timebox Plan et suit risques |
| **Technical Coordinator (TC)** | - | Architecture Definition, SAD | Foundations, Investigation | Supervise architecture et normes | Ignorer impact technique | TC vérifie conformité architecture |
| **Quality Assurance (QA)** | - | Quality Log | Review, Close | Vérifie conformité et qualité | Compromettre qualité pour temps | QA valide les incréments avant Review |
| **Risk & Compliance Officer (RCO)** | - | Risk Log | Kick-Off, Close | Surveille risques et conformité | Oublier risques réglementaires | RCO met à jour log risques RGPD |
| **UX/UI Specialist** | - | Prototypes, UI Models | Investigation, Refinement | Support UX et maquettes | Ignorer ergonomie | UX produit maquette interactive pour Review |
| **DevSecOps / Release Manager** | - | CI/CD Pipeline Logs | Development, Close | Automatisation, intégration continue | Ignorer release process | DevSecOps configure pipeline et tests automatisés |
| **Change Manager** | - | Deployment Plan, Training Docs | Close | Accompagnement changement | Omettre formation utilisateurs | Prépare formation et documentation Release |

## 🔹 **Correspondance événements / Timebox AgilePM ↔ Scrum**

| **Timebox AgilePM** | **Scrum** | **Explication / Tip** |
| --- | --- | --- |
| **Kick-Off** | Sprint Planning | Définir PRL, assigner tâches, aligner sur MoSCoW. Tip : inclure risques et contraintes. |
| **Investigation** | Backlog Refinement | Clarification des stories, prototypage rapide. Tip : inclure Business Ambassador pour validation. |
| **Refinement** | Backlog Refinement / Daily Scrum | Ajustements continus, tests unitaires et techniques. Tip : Dev et Testers doivent collaborer. |
| **Consolidation** | Sprint Work | Construction et tests continus. Tip : inclure QA et DevOps pour intégration CI/CD. |
| **Close** | Sprint Review & Retrospective | Démo, feedback, Timebox Report, Risk Log, actions correctives. Tip : documenter apprentissages pour réutilisation. |

## 🔹 **Tips et pièges essentiels à retenir**

1. **TL ≠ Dev** → rôle de facilitation, pas seulement coder.
2. **MoSCoW v3** : Must/Should/Could/Won’t → must ≤ 60% d’une Timebox.
3. **Support Roles** = QA, RCO, UX, DevOps → essentiels, Scrum ne les formalise pas.
4. **Timebox Report et Risk Log** = documents clés d’AgilePM.
5. **Ne jamais compromettre qualité** → principe “Quality Never Compromised”.
6. **Aligner Work Products** → PRL = Product Backlog, Timebox Plan = Sprint Backlog, Solution Increment = Done.

## 🔹 **Mini-cas d’usage rapide (examen)**

1. **Banque** : module KYC → BA clarifie Must Have, Dev crée micro-service, QA valide tests, Close = Timebox Report + audit.
2. **Énergie** : dashboard smart-meters → Investigation = prototypage UX, Refinement = ajustement KPIs, Consolidation = API + UI intégration, Close = export PDF validé.
3. **Gouvernement** : formulaire e-gov → Kick-Off = contraintes ISO 27001, Investigation = tests utilisateurs, Refinement = OTP + signature électronique, Close = déploiement préprod.

# **Annexes (15 tableaux complémentaires)**

| **N°** | **Thème de l’annexe** | **Contenu / Objectif** |
| --- | --- | --- |
| 1 | **Rôles détaillés par phase** | Tableau avec toutes les phases, chaque rôle impliqué, responsabilités, timebox associés. |
| 2 | **Work Products par phase** | Pour chaque phase (Pre-Project → Realisation), liste des artefacts à produire, importance et destinataires. |
| 3 | **Timebox complet (5 sous-ensembles)** | Kick-Off → Investigation → Refinement → Consolidation → Close ; pour chaque sous-ensemble : rôles, artefacts, tips, pièges. |
| 4 | **MoSCoW v3 par type de Timebox** | Must / Should / Could / Won’t → exemples, pièges, cas pratiques, % d’allocation. |
| 5 | **Alignement AgilePM ↔ Scrum (étendu)** | Rôles, artefacts, événements, tips, pièges, mini-cas d’usage. |
| 6 | **Checklist qualité “Quality Never Compromised”** | Points à vérifier par sous-ensemble de Timebox, rôles responsables, pièges fréquents. |
| 7 | **Risques et conformité (RCO & QA)** | Risques types, impact, mitigation, work products associés, alertes pour examen. |
| 8 | **Rôles de support (QA / DevOps / UX / Change)** | Missions détaillées, phase et timebox impactées, tip d’efficacité, pièges à éviter. |
| 9 | **Planification et estimation AgilePM v3** | Estimation par Timebox, points, story sizing, relation avec PRL et prioritisation MoSCoW. |
| 10 | **Rétrospective et apprentissage** | Points à documenter, Work Products générés (Timebox Report, Risk Log, Lessons Learned), tips et pièges. |
| 11 | **Communication & collaboration** | Tableau rôles ↔ artefacts ↔ points de contact, fréquence, meetings, tips pour réussite en équipe. |
| 12 | **Décisions critiques & points de contrôle** | Tableau décisionnel (BS, BV, PM, CA), artefacts requis, moment Timebox / phase, pièges classiques. |
| 13 | **Mapping Work Products ↔ Certification** | Chaque Work Product à connaître pour l’examen, phase associée, importance relative, pièges fréquents. |
| 14 | **Exemples pratiques / mini-cas** | 21 mini-cas couvrant banque, énergie, administration, télécom, avec Timebox, MoSCoW et rôles. |
| 15 | **Résumé mnémotechnique complet** | Tableaux récapitulatifs : Timebox → Kick-Off → Close, MoSCoW, Rôles, Work Products, Tips à retenir pour examen. |

# **Annexe 1 – Rôles détaillés par phase**

| **Phase** | **Business Roles** | **Solution Roles** | **Project Roles** | **Support Roles** | **Tips** | **Pièges** |
| --- | --- | --- | --- | --- | --- | --- |
| Pre-Project | BS | – | PM | PMO / RCO | Clarifier objectifs, aligner sponsor | Ignorer early risks |
| Feasibility | BS, BV | – | PM, TC, BA2 | QA, RCO | Valider valeur & risques | Définir scope trop vague |
| Foundations | BS, BV, BA2 | TL, SD, ST | PM, TC | QA, PMO, RCO | Créer PRL initiale, architecture solide | Sauter étapes de documentation |
| Evolutionary Dev | BA, BV | TL, SD, ST | PM, TC | DevSecOps, QA, UX | Daily stand-up, prototyping | Ne pas réajuster MoSCoW |
| Deployment | BS, BV | TL, SD, ST | PM, TC | QA, DevOps | Vérifier readiness | Déployer sans test final |
| Realisation | BS, BV | – | PM | PMO, RCO | Mesurer bénéfices, documenter REX | Oublier lessons learned |

# **Annexe 2 – Work Products par phase**

| **Phase** | **Work Products** | **Rôles clés** | **Tips** | **Pièges** |
| --- | --- | --- | --- | --- |
| Pre-Project | Business Case, Outline Business Vision | BS, PM | Définir objectifs clairs | Ignorer sponsors secondaires |
| Feasibility | Feasibility Report, Risk Log, PRL draft | BV, PM, BA2, QA | Early MoSCoW | Trop de détails inutiles |
| Foundations | PRL, Architecture Definition, Solution Architecture Document | TL, SD, ST, TC, BA2 | Prioriser must-have | Architecture trop complexe |
| Evolutionary Dev | Timebox Plan, Increment, Test Specs, Prototypes | TL, Dev, Testers, BA, UX | Documentation légère mais claire | Sauter tests continus |
| Deployment | Deployment Plan, User Docs, Training Material | PM, DevOps, QA | Vérifier readiness | Pas de checklist finale |
| Realisation | Benefits Assessment, Lessons Learned, Highlight Reports | PM, BS, PMO | Documenter succès & erreurs | Oublier feedback stakeholders |

# **Annexe 3 – Timebox AgilePM® v3 (5 sous-ensembles)**

| **Sous-ensemble** | **Objectif** | **Rôles impliqués** | **Work Products** | **Tips** | **Pièges** |
| --- | --- | --- | --- | --- | --- |
| Kick-Off | Clarifier scope, aligner équipe | BA, TL, PM, TC, QA, RCO | Timebox Plan v1, PRL Must/Should | Confirmer ressources, objectifs clairs | Sauter briefing, rôle non assigné |
| Investigation | Analyse, prototypage, clarification | BA, Dev, ST, UX, Advisor | Models, Stories, Prototypes | Réunions courtes, documenter decisions | Confondre Investigation et Development |
| Refinement | Détailler, ajuster MoSCoW | BA, Dev, TL, QA | Updated Stories, Test Specs | Reprioriser MoSCoW, feedback rapide | Trop de documentation inutile |
| Consolidation | Construire, tester, intégrer | Dev, ST, TL, DevOps | Increment, Tests unitaires/fonctionnels, CI/CD report | Continuer tests, intégrer feedback | Ne pas faire daily stand-up |
| Close | Valider incrément, rétro, reporting | TL, PM, BA, Dev, QA, RCO | Timebox Report, Risk Log, Lessons Learned | Documenter apprentissages, préparer next TB | Livrer incomplet, ignorer lessons |

# **Annexe 4 – MoSCoW v3 par type de Timebox**

| **Timebox** | **Must** | **Should** | **Could** | **Won’t** | **Tips** | **Pièges** |
| --- | --- | --- | --- | --- | --- | --- |
| Kick-Off | Définir scope minimal | Clarifier besoins secondaires | Idées complémentaires | Non prioritaire | Toujours valider avec BA | Trop de Must → risque overload |
| Investigation | Stories clés | Prototypes utiles | Options futures | Non priorisé | Prototypage rapide | Confondre Should & Could |
| Refinement | Détails essentiels | Ajustements importants | Optimisations | Non inclus | Vérifier dépendances | Ignorer dépendances |
| Consolidation | Fonctionnalité construite | Correctifs | Améliorations cosmétiques | Pas maintenant | Test continu | Oublier tests unitaires |
| Close | Incrément complet | Feedback important | Améliorations futures | Non livré | Documenter lessons | Livrer incomplet |

# **Annexe 5 – Alignement AgilePM ↔ Scrum**

| **AgilePM** | **Scrum** | **Rôles correspondants** | **Artefacts / Work Products** | **Événements / Timebox** | **Tips** | **Pièges** |
| --- | --- | --- | --- | --- | --- | --- |
| Team Leader | Scrum Master | Facilitation, supprime obstacles | Timebox Plan | Daily Stand-up | Toujours retirer blockers | Micro-manage Devs |
| Business Ambassador | Product Owner | Clarification requirements, Prioritisation MoSCoW | PRL, Business Vision | Timebox Review | Focus sur Must-have | Ignorer feedback métier |
| Timebox | Sprint | Développement itératif | Increment, Test Specs | Daily, Review, Retrospective | Planifier réalistement | Trop long/short |
| Timebox Review | Sprint Review | Démo aux stakeholders | Feedback Log, QA validation | Review | Récolter feedback | Ignorer anomalies |
| Work Products AgilePM | Artefacts Scrum | PRL = Product Backlog, Solution Increments = Sprint Backlog | - | - | Maintenir traceabilité | Confondre documents |
| MoSCoW | Priorisation Product Backlog | Must = Sprint Goal | Must / Should / Could | - | Reprioriser | Ignorer Won’t |

# **Annexe 6 – Checklist qualité et pièges fréquents par Timebox**

| **Timebox** | **Checklist Qualité** | **Tips** | **Pièges** |
| --- | --- | --- | --- |
| Kick-Off | Objectifs clairs, PRL validée, ressources confirmées | Documenter décisions | Ignorer dépendances |
| Investigation | Prototypes testés, exigences clarifiées | Réunions courtes | Confondre Investigation & Development |
| Refinement | Stories détaillées, MoSCoW validée | Reprioriser Must | Trop de Should/Too many Could |
| Consolidation | Tests unitaires & intégration continue | CI/CD report | Ignorer anomalies mineures |
| Close | Timebox Report, Lessons Learned, Risk Log à jour | Documenter apprentissage | Livrer incrément incomplet |

# **Annexe 7 – Risques et gestion des problèmes**

| **Type de risque** | **Phase** | **Timebox** | **Rôle clé** | **Mitigation** | **Mini-cas** |
| --- | --- | --- | --- | --- | --- |
| Scope creep | Foundations | Kick-Off | PM, BA | MoSCoW strict | Ajouter un Must après Kick-Off → revoir PRL |
| Dépendances techniques | Evolutionary Dev | Consolidation | TC, Dev | Daily Stand-up, blockers log | API non disponible → réplanifier test |
| Non-conformité | Deployment | Close | QA, RCO | Checklist ISO, RGPD | Documentation incomplète → audit échoue |
| Délais | Evolutionary Dev | Refinement | PM, TL | Ajuster backlog, prioriser Must | Dev bloqué → timebox dépasse |

# **Annexe 8 – Support Roles et leurs interactions**

| **Rôle Support** | **Phase / Timebox** | **Mission** | **Tip** | **Piège** |
| --- | --- | --- | --- | --- |
| UX/UI Specialist | Investigation, Refinement | Prototypes, feedback utilisateur | Impliquer tôt | Ignorer contraintes techniques |
| DevOps Engineer | Consolidation, Deployment | CI/CD pipelines | Automatiser tests | Ignorer rollback |
| Security Specialist | All | Audit sécurité | Checklist RGPD, ISO 27001 | Ignorer alertes critiques |
| Data/Integration Specialist | Foundations, Evolutionary Dev | Connectivité, qualité données | Test early | Data mismatch |
| Change Manager | Deployment, Realisation | Adoption utilisateurs | Formation ciblée | Communication tardive |

# **Annexe 9 – Communication et gouvernance**

| **Phase** | **Rôles** | **Artefacts** | **Tips** | **Pièges** |
| --- | --- | --- | --- | --- |
| Pre-Project | BS, PM | Project Brief | Aligner sponsors | Trop de détails |
| Feasibility | BV, PM | Feasibility Report | Réunions courtes | Ignorer feedback |
| Foundations | BA, TL, PMO | PRL, SAD | Transparence | Omettre risques techniques |
| Evolutionary Dev | Dev, BA | Timebox Plan | Daily Stand-up | Surcharge info |
| Deployment | PM, DevOps | Deployment Plan | Checklist | Ignorer validation métier |
| Realisation | PM, PMO | Lessons Learned | Documenter succès | Oublier KPI post-déploiement |

# **Annexe 10 – Décisions critiques et escalation**

| **Décision** | **Phase** | **Rôle** | **Tip** | **Piège** |
| --- | --- | --- | --- | --- |
| Changement de scope | Foundations | Change Authority | Toujours MoSCoW | Ignorer impact sur delivery |
| Validation architecture | Foundations | TC | Aligné vision | Non vérifié par PM |
| Ajustement priorité | Evolutionary Dev | BA, TL | Feedback immédiat | Ne pas documenter |
| Go/No-Go Déploiement | Deployment | PM, Sponsor | Checklist complète | Sauter tests finaux |

# **Annexe 11 – Mapping Work Products → Phases → Timebox**

| **Work Product** | **Phase** | **Timebox** | **Rôle responsable** | **Tips** | **Piège** |
| --- | --- | --- | --- | --- | --- |
| PRL | Foundations | Kick-Off, Investigation | BA, BV | Mise à jour régulière | Ignorer dependencies |
| Architecture Definition | Foundations | Kick-Off | TC | Réviser avec dev | Trop détaillé pour TB |
| Increment | Evolutionary Dev | Refinement, Consolidation | Dev | Tests continus | Livrer incomplet |
| Risk Log | All | Toutes | PM, RCO | Mettre à jour chaque TB | Oublier anomalies |
| Lessons Learned | Realisation | Close | PM, PMO | Documenter | Ne pas partager |

# **Annexe 12 – Rétrospective & apprentissages**

| **Phase / Timebox** | **Activité** | **Rôle** | **Tips** | **Pièges** |
| --- | --- | --- | --- | --- |
| Evolutionary Dev | Daily Stand-up | TL, Dev, BA | Focus 15 min | Micro-manage |
| Close | Rétrospective | TL, Dev, BA, PM | Documenter améliorations | Ignorer feedback |
| Realisation | Lessons Learned | PM, PMO, BS | Synthétiser succès & erreurs | Ne pas partager |

# **Annexe 13 – Mini-cas d’usage par secteur**

| **Secteur** | **Cas** | **Timebox clé** | **Tips** |
| --- | --- | --- | --- |
| Banque | Module KYC | Kick-Off → Close | Focus réglementaire, MoSCoW Must |
| Energie | Dashboard smart-meters | Investigation → Close | Prototypage early, tests data |
| Gouvernement | Formulaire e-administration | All | Sécurité ISO + accessibilité |
| E-commerce | Checkout rapide | Refinement → Consolidation | Increment rapide, tests UX |
| Santé | Patient record | Evolutionary Dev | Prioriser sécurité et validation |

# **Annexe 14 – Règles “Quality Never Compromised”**

| **Principe** | **Application** | **Timebox** | **Tip** | **Piège** |
| --- | --- | --- | --- | --- |
| Testing continu | Dev + Testers | Refinement → Consolidation | Automatisation | Ignorer anomalies mineures |
| Revue métier | BA, BV | Review | Feedback immédiat | Ignorer issues backlog |
| Conformité | QA, RCO | All | Checklist | Baisser qualité pour respecter deadline |

# **Annexe 15 – Résumé mnémotechnique et tips pour examen**

| **Concept** | **Astuce mnémotechnique** | **Tips** | **Pièges** |
| --- | --- | --- | --- |
| Timebox 5 sous-ensembles | **KIRC-C** = Kick-Off, Investigation, Refinement, Consolidation, Close | Réviser chaque étape + rôle | Confondre Refinement et Consolidation |
| MoSCoW v3 | Must-Should-Could-Won’t | Vérifier 60% Must | Ajouter trop de Must |
| Roles 4 groupes | SVAA / TL-D-T-SA / PM-TC-BA / SUDeSC | Identifier rôle par phase | Oublier Support |
| Align AgilePM ↔ Scrum | Team Leader = Scrum Master | Rétrospective ≈ Sprint Retrospective | Ignorer feedback |
| Work Products par phase | PRL / SAD / Increments | Tracer tous les artefacts | Livrer incomplet |