

X Sprint Planning de 8h pour Sprint 2 semaines

→ Time-box max = 4h pour Sprint 2 semaines

→ Si ça prend plus longtemps = problème de préparation

Solutions :

- ✓ Refinement régulier du Product Backlog
- ✓ Stories claires AVANT le Planning
- ✓ Scrum Master facilite et garde le time-box

Piège 2 : Product Owner Dicte le Sprint Backlog

X "Vous DEVEZ prendre ces 10 stories"

- ✓ PO propose, Developers décident ce qu'ils peuvent faire
 - Developers connaissent mieux leur capacité
 - Auto-organisation

Piège 3 : Pas de Sprint Goal Clair

X Sprint Goal : "Faire des stories"

X Sprint Goal : "Travailler sur le produit"

- ✓ Sprint Goal cohérent qui lie les items
- ✓ Donne direction et permet adaptation

Piège 4 : Micro-Planning Excessif

X Planifier chaque minute des 2 semaines

→ Impossible, perte de temps

→ Plan va changer anyway

- ✓ Décomposition suffisante (tâches \leq 1 jour)
- ✓ Détails émergent pendant le Sprint

Piège 5 : Ignorer la Definition of Done

✗ Sélectionner 50 SP sans considérer la DoD

✓ Capacité doit inclure TOUT le travail pour "Done"

(dev + tests + doc + review + deploy...)

⌚ Questions Type Exam Sprint Planning

Q1 : Qui participe au Sprint Planning ?

<details> <summary>Réponse</summary> ✓ **Toute la Scrum Team** (PO + Developers + SM) - OBLIGATOIRE D'autres peuvent être invités si besoin d'expertise spécifique </details>

Q2 : Durée max Sprint Planning pour Sprint 2 semaines ?

<details> <summary>Réponse</summary> ✓ **4 heures** (8h pour 1 mois, proportionnel) </details>

Q3 : Qui décide quels items du Product Backlog sont pris en Sprint ?

<details> <summary>Réponse</summary> ✓ **Les Developers**, car eux seuls connaissent leur capacité. Le PO peut influencer mais ne peut pas forcer. </details>

Q4 : Peut-on faire Sprint Planning sans Sprint Goal ?

<details> <summary>Réponse</summary> ✗ **Non** - Le Sprint Goal est obligatoire, c'est l'engagement du Sprint Backlog. </details>

Q5 : Si tout n'est pas décomposé en détail à la fin du Sprint Planning, que faire ?

<details> <summary>Réponse</summary> ✓ **C'est OK** - Le plan émerge pendant le Sprint. Décomposition continue au Daily Scrum et pendant le Sprint. L'important est d'avoir assez de détail pour démarrer. </details>

7.3 Daily Scrum

📘 Définition Officielle

"The purpose of the Daily Scrum is to inspect progress toward the Sprint Goal and adapt the Sprint Backlog as necessary, adjusting the upcoming planned work."

Caractéristiques :

- ⏳ **Time-box : 15 minutes** (MAX, pas minimum !)
 - 📅 **Quotidien** : Chaque jour du Sprint, même heure, même lieu
 - 👤 **Pour les Developers** : Event des Developers (SM facilite si demandé)
 - 💬 **Focus : Sprint Goal** et adaptation du plan
-

🗣 Structure du Daily Scrum

✗ ANCIEN Format (Avant 2020) - Ne Plus Utiliser !

Les 3 questions (obsolètes) :

1. Qu'ai-je fait hier ?
2. Que vais-je faire aujourd'hui ?
3. Ai-je des impediments ?

Problèmes :

- Devient un "rapport" au Scrum Master
- Focus sur l'individu, pas l'équipe
- Pas assez d'adaptation

✓ NOUVEAU Format (Depuis 2020) - Flexible !

Scrum Guide 2020 :

"The Developers can select whatever structure and techniques they want, as long as their Daily Scrum focuses on progress toward the Sprint Goal and produces an actionable plan for the next day of work."

Structure Libre :

- ✓ Les Developers **choisissent** la structure
- ✓ Focus sur **Sprint Goal**
- ✓ Résultat = **Plan actionable** pour la journée

Exemples de Formats Possibles :

Format 1 : Walk the Board 📁

1. Regarder le Sprint Backlog (board)

2. Partir de la DROITE (items presque Done)
3. Aller vers la GAUCHE (items To Do)
4. Pour chaque item : "Que faut-il pour avancer ?"

Avantages :

- Focus sur items, pas personnes
- Encourage collaboration
- Finir avant de commencer nouveau

Format 2 : Sprint Goal Focus

Question unique :

"Que devons-nous faire AUJOURD'HUI pour atteindre le Sprint Goal ?"

Avantages :

- Ultra focus sur objectif
- Évite digressions
- Court et efficace

Format 3 : Blocages & Aide

Tour de table :

- Qui a besoin d'aide ?
- Qui peut aider ?
- Quels sont les impediments ?

Avantages :

- Collaboration immédiate
- Résolution rapide problèmes

Format 4 : Classique Adapté

Pour chaque personne (optionnel) :

- Progrès vers Sprint Goal hier ?
- Plan aujourd'hui pour Sprint Goal ?
- Besoin d'aide / impediments ?

(Mais focus = équipe, pas individu)

Time-box : 15 Minutes MAX

Pourquoi 15 minutes ?

- Assez long pour synchroniser
- Assez court pour garder focus
- Force concision et efficacité

Techniques pour respecter 15 min :

1 Stand-up physique

- Littéralement debout
- Inconfort léger = réunions courtes
- Pas de chaises !

2 Timer visible

- 15 min countdown
- Rappel visuel constant

3 "Parking Lot"

- Discussions longues → notées pour après
- Daily = synchronisation, pas résolution complète

4 Pas de digressions techniques

- Détails techniques → discussion séparée après
- Daily = high-level

Si besoin de plus de 15 min :

Daily Scrum : 15 min

↓

Fin du Daily

↓

Ceux concernés restent pour "After-meeting"

↓

Discussions détaillées (pas time-boxed)

👥 Qui Participe ?

OBLIGATOIRE :

- **Developers** (tous !)

OPTIONNEL :

-  **Scrum Master** : Peut faciliter si Developers le demandent
-  **Product Owner** : Peut participer (mais n'est pas obligé)
-  **Autres** : Peuvent observer en silence (pas d'interruption)

⚠ Important :

- Daily Scrum = **pour** les Developers, **par** les Developers
- Pas un "rapport" au SM ou PO
- Pas une "status meeting" pour management

Anti-pattern ✗ :

✗ Daily devient :

"Rapport au Scrum Master qui note tout"

"Justification de son temps au manager"

"Interrogatoire par le PO"

✓ Daily devrait être :

"Équipe se synchronise entre elle"

"Plan adapté collectivement"

"Auto-organisation en action"

⌚ Objectifs du Daily Scrum

3 Objectifs Principaux :

1 Inspecter le progrès vers Sprint Goal 🔎

Questions implicites :

- Sommes-nous on track pour le Sprint Goal ?
- Y a-t-il des risques ?
- Quels items sont bloqués ?

2 Adapter le Sprint Backlog 📈

Adaptations possibles :

- Réorganiser tâches
- Ajouter tâches découvertes
- Retirer tâches non nécessaires
- Changer approche technique

3 Créer plan actionable pour 24h 📋

Résultat attendu :

- Chaque Developer sait quoi faire aujourd'hui
 - Collaborations identifiées
 - Impediments signalés au SM
-

🚫 Ce Que le Daily Scrum N'EST PAS

✗ Pas une Status Meeting

Status Meeting (anti-pattern) Daily Scrum (correct)

Rapport au manager → Synchronisation équipe

Focus passé → Focus futur (plan du jour)

Justification temps → Collaboration

Pas une Résolution de Problèmes

Si problème complexe découvert au Daily :

1. Noter le problème
2. Continuer Daily (time-box !)
3. Après Daily : ceux concernés restent pour résoudre

Daily = identifier problèmes

Après = résoudre

Pas une Planification Détaillée

 45 minutes à re-planifier tout le Sprint

 15 minutes pour ajuster plan des prochaines 24h

Pas Optionnel

 "On se connaît bien, pas besoin de Daily"

 "On est en remote, on s'envoie des messages"

 Daily Scrum est OBLIGATOIRE

(Scrum Guide : tous les événements sont mandatory)

Lieu et Heure

Même Heure, Même Lieu

Pourquoi ?

-  **Routine** : Pas besoin de réfléchir quand/ où
-  **Discipline** : Habitude forte
-  **Simplicité** : Moins de coordination

Quelle heure choisir ?

Option 1 : Début de journée

Exemple : 9h00 ou 9h30

Avantages :

- Lance la journée
- Plan clair dès le matin
- Tout le monde frais

Inconvénients :

- Flexitime difficile
- Certains commencent avant

Option 2 : Milieu de journée ☀️

Exemple : 11h ou 14h

Avantages :

- Compatible flexitime
- Tout le monde disponible
- Coupure dans la journée

Inconvénients :

- Interruption flow

Option 3 : Fin de journée 🌙

Exemple : 17h

Avantages :

- Review de la journée
- Plan pour lendemain

Inconvénients :

- Fatigue

 Certains partent tôt

 Moins d'énergie adaptation

Recommandation : 9h30 ou 10h00

- Compromis flexitime
 - Énergie du matin
 - Lance la journée productive
-

Daily Scrum Remote

Outils :

-  Visio (Zoom, Teams, Meet)
-  Board virtuel (Jira, Trello, Miro)
-  Timer partagé

Bonnes Pratiques Remote :

1 Caméras ON

- Engagement visuel
- Langage corporel
- Connexion humaine

2 Check-in rapide

- 30 sec : Bonjour, comment allez-vous ?
- Connexion avant business

3 Screen share du board

- Tous voient la même chose
- Focus visuel commun

4 Mute par défaut

- Parle quand ton tour
- Moins de bruit de fond

5 Ponctualité stricte

- Commence à l'heure pile
 - Respecte ceux qui sont à l'heure
-

Questions Type Exam Daily Scrum

Q1 : Durée max du Daily Scrum ?

<details> <summary>Réponse</summary> **15 minutes** maximum </details>

Q2 : Qui facilite le Daily Scrum ?

<details> <summary>Réponse</summary> Les **Developers** le facilitent eux-mêmes. Le SM peut aider s'ils le demandent, mais ce n'est pas son rôle automatique. </details>

Q3 : Le Product Owner doit-il assister au Daily Scrum ?

<details> <summary>Réponse</summary> **Optionnel** - Le PO peut assister mais ce n'est pas obligatoire. Le Daily est pour les Developers. </details>

Q4 : Peut-on sauter le Daily si rien n'a changé depuis hier ?

<details> <summary>Réponse</summary> **Non** - Le Daily Scrum est obligatoire chaque jour du Sprint. Il permet l'inspection et adaptation quotidienne. </details>

Q5 : Que faire si le Daily dépasse régulièrement 15 minutes ?

<details> <summary>Réponse</summary> Réponses possibles : - Utiliser un timer strict - Identifier discussions hors sujet → parking lot - Changer le format (ex: walk the board) - SM coach l'équipe sur concision </details>

Q6 : Les 3 questions (hier/aujourd'hui/impédiments) sont-elles obligatoires ?

<details> <summary>Réponse</summary> **Non** - Depuis Scrum Guide 2020, la structure est **libre**. Les Developers choisissent le format tant qu'il se concentre sur Sprint Goal et produit un plan actionable. </details>

7.4 Sprint Review

Définition Officielle

"The purpose of the Sprint Review is to inspect the outcome of the Sprint and determine future adaptations. The Scrum Team presents the results of their work to key stakeholders and progress toward the Product Goal is discussed."

Caractéristiques :

- ⏳ **Time-box : 4 heures max** pour Sprint 1 mois (proportionnel)
 - 👤 **Participants** : Scrum Team + Key Stakeholders
 - 🎯 **Objectif** : Inspecter Increment et adapter Product Backlog
 - ❌ **Pas une démo PowerPoint !** - Working software demo
-

🎯 Objectifs de la Sprint Review

2 Objectifs Principaux :

1 Inspecter l'Increment 🔎

- Démo des fonctionnalités "Done"
- Valider contre Definition of Done
- Recevoir feedback

2 Adapter le Product Backlog 💡

- Basé sur le feedback
 - Nouvelles découvertes
 - Changements de marché/business
 - Ajuster priorités
-

👤 Qui Participe ?

OBLIGATOIRE :

- ✅ **Toute la Scrum Team** : PO + Developers + SM
- ✅ **Key Stakeholders** : Invités par le Product Owner

Stakeholders peuvent inclure :

- 💼 Management
- 👤 Utilisateurs finaux
- 🏢 Clients
- 💰 Sponsors
- 🔐 Équipes adjacentes (si dépendances)

-  Business analysts
-  Designers

Important :

- Sprint Review = événement **public** et **transparent**
 - Plus il y a de stakeholders, mieux c'est (feedback riche)
 - Pas limité à "senior management"
-

Structure Typique Sprint Review (4h pour 1 mois)

Sprint 2 semaines = 2h typique

Partie 1 : Introduction (15 min)

09:00 - 09:15

- |— Welcome stakeholders
- |— Rappel Product Goal
- |— Rappel Sprint Goal
- └ Agenda de la Review

Partie 2 : Démo de l'Increment (60 min)

09:15 - 10:15

- |— Démo fonctionnalités "Done"
- |— Focus sur valeur business
- |— Live demo (pas slides !)
- |— Questions/feedback au fur et à mesure
- └ Seulement ce qui respecte DoD

Partie 3 : Discussion Product Backlog (30 min)

10:15 - 10:45

- |— PO présente état Product Backlog
- |— Progrès vers Product Goal
- |— Changements depuis dernier Sprint

└– Métriques (vitesse, burndown)

└ Projection timeline

Partie 4 : Feedback & Adaptation (15 min)

10:45 - 11:00

└– Feedback stakeholders

└– Nouvelles idées/features

└– Changements priorités

└– Risques identifiés

└ Décisions PO sur adaptations

La Démo de l'Increment

Bonnes Pratiques Démo :

1 Démo de Working Software

 Bon : Démo live de l'application

 Mauvais : PowerPoint expliquant ce qui a été fait

 Mauvais : Vidéo pré-enregistrée (sauf si impossible)

2 Focus sur la Valeur Business

 Bon : "Maintenant les clients peuvent filtrer par prix, ce qui réduit le temps de recherche de 50%"

 Mauvais : "On a refactoré le code backend en microservices"

3 Raconter une Histoire

Structure :

1. Contexte : "Nos utilisateurs avaient ce problème..."

2. Solution : "On a créé cette fonctionnalité..."

3. Démo : "Voici comment ça marche..." (live)

4. Impact : "Ça permet de..."

4 Seulement ce qui est "Done"

Bon : Démo features 100% Done (selon DoD)

Mauvais : "C'est presque fini, il reste juste les tests..."

Mauvais : "On a fait 80%, on finira au prochain Sprint"

5 Préparer mais Accepter les Bugs

Préparation :

- Test environnement
- Données de démo prêtes
- Scénarios définis

Si bug pendant démo :

"Intéressant, on va investiguer - passons à la prochaine feature"

Pas de panique, c'est du live !

Adaptation du Product Backlog

Résultats Typiques d'une Sprint Review :

1 Nouvelles User Stories Crées

Exemple feedback stakeholder :

"C'est bien, mais ce serait encore mieux si on pouvait trier par date"

PO action :

→ Crée nouvelle story : "En tant qu'utilisateur, je peux trier par date"

→ Priorise dans Product Backlog

2 Stories Existantes Modifiées

Feedback : "Cette fonctionnalité ne correspond pas à ce qu'on voulait"

PO action :

→ Modifie story existante

→ Ajoute clarifications

→ Re-priorise si nécessaire

3 Stories Retirées

Feedback : "Finalement, on n'a plus besoin de cette feature"

PO action :

→ Retire story du Product Backlog

→ Focus sur features à plus haute valeur

4 Priorités Changées

Feedback : "Le marché a changé, il faut prioriser X avant Y"

PO action :

→ Réordonne Product Backlog

→ Explique nouvelles priorités à l'équipe

⚠ Ce Que Sprint Review N'EST PAS

✗ Pas une Session d'Acceptation Formelle

Review ≠ "Go/No-Go" gate

Review = Collaboration et feedback

L'Increment est déjà "Done" (selon DoD)

Review permet d'adapter le futur, pas valider le passé

✗ Pas un Rapport de Status au Management

✗ Mauvais : "Voici notre rapport de productivité"

✓ Bon : "Voici la valeur créée, comment peut-on s'améliorer ?"

✗ Pas Uniquement une Démo

Review inclut :

- Démo (partie importante)
- Discussion Product Backlog
- Feedback
- Adaptation
- Collaboration

Si Review = juste démo → on rate l'adaptation !

Pas une Phase de Test

Testing doit être fait PENDANT le Sprint

Review = démo de ce qui est déjà testé et Done

Métriques Discutées en Review

Métriques Typiques :

1 Vélocité

Sprint 1 : 25 SP

Sprint 2 : 28 SP

Sprint 3 : 30 SP (actuel)

Tendance positive, capacité stable

2 Burndown Chart

Montre progrès quotidien pendant Sprint

Permet de voir si on était on track

3 Release Burndown

Progrès vers Product Goal

Stories restantes vs Sprints estimés

Projection date livraison

4 Business Metrics

- Adoption utilisateurs

- NPS (Net Promoter Score)
 - Revenus générés
 - Bugs en production
 - Performance (temps réponse)
-

🎯 Questions Type Exam Sprint Review

Q1 : Qui peut participer à la Sprint Review ?

<details> <summary>Réponse</summary> **Toute la Scrum Team** (obligatoire) + **Key Stakeholders** invités par le PO (recommandé) C'est un événement **public** et **ouvert**. </details>

Q2 : Durée max Sprint Review pour Sprint 2 semaines ?

<details> <summary>Réponse</summary> **2 heures** (4h pour 1 mois, proportionnel) </details>

Q3 : Peut-on démontrer du travail non "Done" en Sprint Review ?

<details> <summary>Réponse</summary> **Non** - Seulement ce qui respecte la Definition of Done peut être démontré. Le reste reste dans le Product Backlog.
</details>

Q4 : La Sprint Review est-elle obligatoire ?

<details> <summary>Réponse</summary> **Oui** - Tous les événements Scrum sont obligatoires. Pas de Review = pas d'inspection de l'Increment = Scrum incomplet.
</details>

Q5 : Que se passe-t-il si aucun stakeholder ne vient à la Review ?

<details> <summary>Réponse</summary> La Review a quand même lieu (avec au minimum la Scrum Team). Mais c'est un **gros problème** car : - Pas de feedback externe - Pas d'adaptation basée sur stakeholders - Risque de développer dans mauvaise direction

Le SM doit travailler avec le PO pour engager les stakeholders.

</details>

Q6 : Le Product Backlog peut-il être modifié pendant la Sprint Review ?

<details> <summary>Réponse</summary>  **Oui !*** C'est même un des objectifs principaux. Basé sur le feedback et les learnings, le PO adapte le Product Backlog. </details>

7.5 Sprint Retrospective

Définition Officielle

"The purpose of the Sprint Retrospective is to plan ways to increase quality and effectiveness. The Scrum Team inspects how the last Sprint went with regards to individuals, interactions, processes, tools, and their Definition of Done."

Caractéristiques :

-  **Time-box** : 3 heures max pour Sprint 1 mois (proportionnel)
 -  **Participants** : Scrum Team uniquement (PO + Developers + SM)
 -  **Objectif** : Amélioration continue
 -  **Safe space** : Discussions honnêtes et ouvertes
-

Objectifs de la Retrospective

3 Objectifs Principaux :

1 Inspecter le dernier Sprint

- Ce qui s'est bien passé
- Ce qui s'est mal passé
- Problèmes rencontrés

2 Identifier les Améliorations

- Actions concrètes
- Changements de processus
- Ajustements DoD si nécessaire

3 Plan d'Action

- 1-3 améliorations prioritaires
- Assignation responsabilités

- Implémentation dès le prochain Sprint
-

Qui Participe ?

OBLIGATOIRE :

-  **Product Owner** (membre Scrum Team)
-  **Developers** (tous)
-  **Scrum Master** (facilite souvent)

PAS invités :

-  Stakeholders externes
-  Management
-  Autres équipes (sauf exception)

Pourquoi Scrum Team uniquement ?

-  **Safe space** : Parler librement sans jugement
-  **Confiance** : Admettre erreurs sans peur
-  **Honnêteté** : Discussions franches sur problèmes
-  **Focus** : Sur ce que l'équipe peut contrôler

Exception :

Si problème nécessite présence externe :

Exemple : Impediment organisationnel nécessite manager

Options :

1. Inviter personne spécifique pour partie ciblée
 2. OU traiter en dehors de la Retrospective
-

Structure Typique Retrospective (1.5h pour Sprint 2 sem)

Phase 1 : Set the Stage (10 min)

Objectif : Créer atmosphère positive et safe

Activités :

|— Check-in émotionnel

| "Sur une échelle 1-5, comment vous sentez-vous ?"

|— Rappel règles :

| - Vegas Rule (ce qui se dit ici reste ici)

| - No blame

| - Focus sur amélioration, pas critique

└ Revisiter Sprint Goal et résultats

Phase 2 : Gather Data (20 min)

Objectif : Collecter faits et impressions du Sprint

Techniques :

|— Timeline : Événements marquants du Sprint

|— Mad/Sad/Glad : Émotions sur différents événements

|— Start/Stop/Continue : Brainstorm actions

└ What went well / What didn't : Classic

Formats :

- Post-its physiques

- Board virtuel (Miro, Mural)

- Chacun écrit silencieusement puis partage

Phase 3 : Generate Insights (30 min)

Objectif : Comprendre WHY, trouver root causes

Activités :

|— Grouper items similaires

|— Voter sur priorités (dot voting)

- └─ "5 Whys" sur problèmes principaux
- └─ Fishbone diagram si problème complexe
- └ Discussion ouverte

Phase 4 : Decide What to Do (20 min) ✅

Objectif : Actions concrètes et actionables

Critères actions :

- └─ SMART (Specific, Measurable, Achievable, Relevant, Time-bound)
- └─ Qui est responsable ?
- └─ Quand sera-ce fait ?
- └ Comment mesurer succès ?

Règle : Maximum 1-3 actions

(Mieux 1 action bien faite que 10 oubliées)

Phase 5 : Close (10 min) 📄

Objectif : Clôturer positivement

Activités :

- └─ Récap actions décidées
- └─ Appreciation round
 - | "Merci à X pour..."
- └─ Check-out
 - | "Un mot pour décrire comment je me sens maintenant"
- └ Rappel : on implémente dès demain !

Techniques de Facilitation

Technique 1 : Start/Stop/Continue 🚧

Post-its en 3 colonnes :

START STOP CONTINUE

Que devrait-on Que devrait-on Que fait-on bien
commencer à faire ? arrêter de faire ? qu'on doit continuer ?

Exemples :

START STOP CONTINUE

- Pair programming - Interruptions Daily - Code reviews
- Tests auto - Grosses branches - Refinement régulier
- DoD plus stricte - Meetings inutiles - Dailies à 10h

Technique 2 : Sailboat 🛶

Métaphore du bateau :

🌴 Island (Goal)



⛵ Boat (Team)



⾵ Wind | ⚓ Anchor

(Helps) | (Slows)



⽯ Rocks (Risks)

Questions :

- Vent (winds) : Qu'est-ce qui nous pousse vers le goal ?

- Ancre (anchor) : Qu'est-ce qui nous ralentit ?
- Rochers (rocks) : Quels risques à venir ?
- île (island) : Où voulons-nous aller ?

Technique 3 : 4Ls 4□

Liked Learned Lacked Longed For

Qu'ai-je Qu'ai-je Qu'est-ce Qu'est-ce que
aimé ? appris ? qui manquait? je voudrais ?

Exemple :

LIKED LEARNED LACKED LONGED FOR

- Nouvelle feature - Docker skills - Time refinement - Automated tests
- Team spirit - DB optimization - Clear stories - Better tools

Technique 4 : Mad/Sad/Glad 😠 😢 😊

3 colonnes émotionnelles :

MAD (😠) SAD (😢) GLAD (😊)

Qu'est-ce qui Qu'est-ce qui Qu'est-ce qui
m'énerve ? me déçoit ? me rend heureux ?

Permet expression émotions (sain !)

Puis chercher solutions aux Mad/Sad

Technique 5 : Speedboat (variation) 🚤

Product Goal



🚢 Team



 Anchors (ce qui ralentit)



 Water (environment)



 Rocks (risques)

Actions de Retrospective

Caractéristiques d'une Bonne Action :

SMART :

- **S**pecific (Spécifique)
- **M**easurable (Mesurable)
- **A**chievable (Atteignable)
- **R**elevant (Pertinent)
- **T**ime-bound (Temporisé)

Exemples :

 Mauvaise action :

"Améliorer la communication"

→ Trop vague, pas mesurable, pas d'owner

 Bonne action :

"David organisera une session 'Lunch & Learn' sur Docker

vendredi prochain de 12h-13h pour partager ses connaissances"

→ Spécifique, owner clair, deadline, measurable (session faite ou non)

 Mauvaise action :

"Faire plus de tests"

→ Vague, pas measurable

Bonne action :

"Ajouter 'Coverage >80%' à notre Definition of Done
et mettre en place outil coverage dès le prochain Sprint"
→ Spécifique, mesurable, temporisé

Suivi des Actions

Problème Courant :

Retrospective :

- └─ Actions décidées 
- └─ Tout le monde motivé 
- └─ Sprint suivant...
- └ Actions oubliées 

Résultat : Démoralisation, Retrospectives inutiles

Solution : Suivi Rigoureux

1. Action Board Visible

Sprint Backlog doit inclure :

- └─ Sprint Backlog Items (stories)
- └ Action Items Retrospective
 - └─ Action 1 : [Description] - Owner: David
 - └─ Action 2 : [Description] - Owner: Marie
 - └ Status visible (To Do, In Progress, Done)

2. Review Actions en Sprint Planning

Début Sprint Planning :

"Avant de planifier, revoyons actions Retro précédente"

- └─ Action 1 :  Done

|— Action 2 : 🟡 En cours → Reporter

└ Action 3 : ❌ Pas faite → Pourquoi ? Encore pertinente ?

3. Reminder au Daily Scrum 💬

SM peut rappeler :

"N'oubliez pas notre action Retro : pair programming sur feature X"

Mais sans être "taskmaster" !

4. Retrospective de la Retrospective 🗂️

À chaque Retro, prendre 5 min :

"Comment vont nos actions des Retros précédentes ?"

Si taux de completion bas :

- Problème à adresser !
 - Soit actions pas réalistes
 - Soit manque commitment
-

⚠ Anti-patterns Retrospective

Anti-pattern 1 : Blame Game 🤚

❌ Mauvais :

"C'est la faute de Jean si le Sprint a raté"

"Le PO n'écoute jamais"

✓ Bon :

"On a eu un problème de clarté des stories"

"Comment peut-on améliorer communication avec PO ?"

Règle : Focus sur le problème, pas la personne

Anti-pattern 2 : Même Format Répété

Si toujours le même format :

- └─ Ennui
- └─ Moins de participation
- └ Insights répétitifs

Solution : Varier techniques (Sailboat, 4Ls, Timeline...)

SM doit renouveler !

Anti-pattern 3 : Trop d'Actions

 15 actions décidées

→ Aucune ne sera faite

 1-3 actions maximum

→ Focus, execution réelle

Anti-pattern 4 : Pas de Safe Space

Si management présent :

- └─ Équipe censure
- └─ Pas de vraies discussions
- └ Retrospective inutile

Solution : Scrum Team UNIQUEMENT

Anti-pattern 5 : Actions Vagues

-  "Améliorer qualité"
-  "Mieux communiquer"
-  "Être plus agile"

Toutes trop vagues, pas actionables

Anti-pattern 6 : Rétrospective Monotone 😴

Si toujours positif ("tout va bien") :

- Pas d'amélioration réelle
- Peur de parler des vrais problèmes

SM doit creuser, poser questions inconfortables

⌚ Questions Type Exam Sprint Retrospective

Q1 : Qui participe à la Sprint Retrospective ?

<details> <summary>Réponse</summary> **La Scrum Team uniquement** :
Product Owner + Developers + Scrum Master Pas de stakeholders externes, pas de management </details>

Q2 : Durée max Retrospective pour Sprint 2 semaines ?

<details> <summary>Réponse</summary> **1.5 heures** (3h pour 1 mois, proportionnel) </details>

Q3 : La Retrospective peut-elle être sautée si tout va bien ?

<details> <summary>Réponse</summary> **Non** - La Retrospective est obligatoire. Même quand tout va bien, il y a toujours des opportunités d'amélioration. </details>

Q4 : Peut-on inviter un manager à la Retrospective pour discuter d'un impediment organisationnel ?

<details> <summary>Réponse</summary> **Possible mais pas idéal**. Mieux vaut :
1. Discuter en Retro (Scrum Team seule)
2. Décider action : SM va parler au manager
3. Meeting séparé avec manager si nécessaire

Garder Retro comme safe space pour l'équipe.

</details>

Q5 : La Definition of Done peut-elle être modifiée en Retrospective ?

<details> <summary>Réponse</summary> **Oui !*** La Retrospective est le moment approprié pour adapter la DoD si l'équipe identifie que les standards de qualité doivent être ajustés. </details>

Q6 : Combien d'actions devrait-on décider en Retrospective ?

<details> <summary>Réponse</summary>  Le Scrum Guide ne prescrit pas de nombre, mais **bonnes pratiques : 1-3 actions maximum** pour assurer leur exécution réelle. </details>

8. Les Artifacts Scrum

8.1 Introduction aux Artifacts

Définition

"Scrum's artifacts represent work or value. They are designed to maximize transparency of key information."

Les 3 Artifacts Scrum :

1.  **Product Backlog**
2.  **Sprint Backlog**
3.  **Increment**

Chaque Artifact a un Engagement (Commitment) associé :

Product Backlog → Product Goal

Sprint Backlog → Sprint Goal

Increment → Definition of Done

Rôle des Engagements :

- Fournir **focus** et **commitment**
 - Améliorer **transparence**
 - Permettre **inspection** et **adaptation**
-

8.2 Product Backlog

Définition Officielle

"The Product Backlog is an emergent, ordered list of what is needed to improve the product. It is the single source of work undertaken by the Scrum Team."

Caractéristiques :

-  **Liste ordonnée** (pas juste priorisée)
-  **Émergente** : Évolue constamment

-  **Source unique** : THE source of work
 -  **Owner** : Product Owner est accountable
-

Product Goal (Engagement du Product Backlog)

Définition :

"The Product Goal describes a future state of the product [...] The Product Goal is the long-term objective for the Scrum Team."

Caractéristiques d'un Bon Product Goal :

-  **Long-terme** : Vision 3-12 mois typiquement
-  **Mesurable** : Critères de succès clairs
-  **Inspirant** : Motive l'équipe
-  **Aligné** : Avec stratégie business

Exemples :

Mauvais Product Goals :

"Avoir un bon produit"

"Satisfaire les clients"

"Augmenter les ventes"

Bons Product Goals :

"Devenir la plateforme #1 de e-learning en France

avec 100K utilisateurs actifs d'ici décembre 2025"

"Réduire le temps de traitement des commandes de 48h à 2h

tout en maintenant 99.9% de précision"

"Permettre aux utilisateurs de créer et publier du contenu
sans intervention technique, avec 10K contenus publiés/mois"

Product Goal vs Company Vision :

Company Vision (5-10 ans)

"Révolutionner l'éducation en ligne"

|

|-- Product Goal 1 (6-12 mois)

| "Platform MVP avec cours vidéo et quiz"

| |

| |-- Sprint Goal 1 (2 weeks)

| |-- Sprint Goal 2 (2 weeks)

| | L Sprint Goal 3 (2 weeks)

|

|-- Product Goal 2 (6-12 mois suivants)

"Social learning avec forums et mentoring"

Structure du Product Backlog

Le Product Backlog contient :

-  **User Stories / Product Backlog Items (PBIs)**
-  **Bugs**
-  **Technical work** (refactoring, infrastructure)
-  **Knowledge acquisition** (spikes, research)
-  **UX improvements**

Format typique d'un PBI (User Story) :

Title : Login avec Google

User Story :

En tant qu'utilisateur

Je veux me connecter avec mon compte Google

Afin de ne pas créer un nouveau mot de passe

Acceptance Criteria :

- Bouton "Se connecter avec Google" visible
- Redirection OAuth Google fonctionne
- Utilisateur créé automatiquement si nouveau
- Session maintenue pendant 30 jours
- Fonctionne sur mobile et desktop

Business Value : HIGH (facilite onboarding)

Story Points : 5

Dependencies : Aucune

Notes : Vérifier RGPD pour données Google

Ordre du Product Backlog

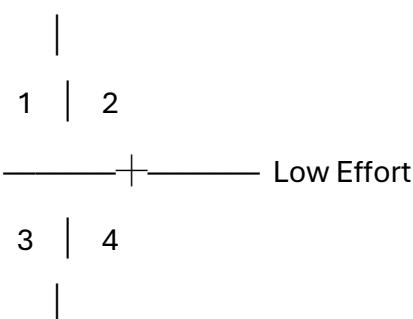
Product Owner ordonne basé sur :

1. **Valeur Business** 💰 (le plus important)
2. **Risque** ⚠️ (items risqués tôt)
3. **Dépendances** 🔗 (ce qui doit être fait avant)
4. **Coût** 💵 (quick wins vs investissements)

Techniques de Priorisation :

1. Value vs Effort Matrix

High Value



Low Value

Ordre : 1 → 2 → 3 → 4

2. MoSCoW 🎯

M - Must have : Critique, sans ça pas de produit

S - Should have : Important mais pas bloquant

C - Could have : Nice to have

W - Won't have : Hors scope pour l'instant

3. WSJF (Weighted Shortest Job First) ⏲

WSJF = (Business Value + Time Criticality + Risk Reduction) / Job Size

Exemple :

Story A : $(8 + 5 + 3) / 5 = 3.2$

Story B : $(9 + 2 + 1) / 8 = 1.5$

→ Story A prioritaire (WSJF plus élevé)

⌚ Refinement du Product Backlog

Définition :

"Product Backlog refinement is the act of breaking down and further defining Product Backlog items into smaller more precise items."

Caractéristiques :

- ⌚ **Continu** : Tout au long du Sprint
- ⌚ **~10% capacité Sprint** : Règle générale
- 👥 **Collaboration** : PO + Developers (+ SM si nécessaire)
- 🎯 **Objectif** : Items "Ready" pour futur Sprint Planning

Activités Refinement :

|– Clarifier stories

- | └ Questions/réponses
- |
- | └ Décomposer gros items
- | └ Epic → Multiple stories
- |
- | └ Estimer
- | └ Story Points, T-shirt sizes
- |
- | └ Définir Acceptance Criteria
- | └ "Done" = quoi exactement ?
- |
- | └ Identifier dépendances
- | └ Techniques, équipes externes
- |
- └ Recherche (Spikes)
 - └ POC si incertitude technique

"Definition of Ready" (Optionnel) :

Un PBI est "Ready" quand :

- User Story claire
- Acceptance Criteria définis
- Estimé par l'équipe
- Dependencies identifiées
- Valeur business comprise
- Taille ≤ 1 Sprint (décomposé sinon)
- Tesable

Note : "Definition of Ready" n'est PAS dans Scrum Guide

mais c'est une pratique courante et utile

Estimation

Pourquoi Estimer ?

-  **Planification** : Combien peut-on prendre en Sprint ?
-  **Prédictibilité** : Forecasting releases
-  **Conversation** : Révèle incompréhensions

Techniques d'Estimation :

1. Story Points

Mesure relative de :

- Complexité
- Effort
- Incertitude

Échelle Fibonacci : 1, 2, 3, 5, 8, 13, 21...

(Grands nombres = plus d'incertitude)

Exemple :

Story A (Login simple) : 3 SP

Story B (Payment gateway) : 13 SP

Story C (Changer couleur bouton) : 1 SP

2. Planning Poker

1. PO présente story
2. Developers posent questions
3. Chacun choisit carte (1,2,3,5,8,13...) en secret
4. Révèle simultanément
5. Si consensus → Done
6. Sinon → Discussion (pourquoi 3 vs 13 ?)

7. Re-vote
8. Répéter jusqu'à consensus

Avantages :

- Engage tout le monde
- Révèle assumptions différentes
- Learning collectif

3. T-Shirt Sizes

XS, S, M, L, XL

Plus rapide que Story Points

Bon pour estimation initiale (epic level)

Puis convertir en Story Points :

XS = 1-2

S = 3-5

M = 8

L = 13

XL = 20+ (à décomposer !)

Questions Type Exam Product Backlog

Q1 : Qui est responsable du Product Backlog ?

<details><summary>Réponse</summary> **Le Product Owner** est accountable pour le Product Backlog, incluant son contenu, ordre et clarté. </details>

Q2 : Combien de Product Backlogs pour une Scrum Team ?

<details><summary>Réponse</summary> **Un seul** Product Backlog par produit (même si plusieurs équipes travaillent dessus). </details>

Q3 : Qui peut ajouter des items au Product Backlog ?

<details> <summary>Réponse</summary> **N'importe qui peut suggérer** des items, mais seul le **Product Owner décide** ce qui entre et dans quel ordre. </details>

Q4 : Le Product Backlog doit-il être complet dès le début ?

<details> <summary>Réponse</summary> **Non** - Le Product Backlog est **émergent**. Il évolue constamment à mesure qu'on apprend. </details>

Q5 : Qu'est-ce qui est plus important dans l'ordre du Product Backlog ?

<details> <summary>Réponse</summary> **La valeur business** est le critère principal, mais le PO peut aussi considérer le risque, les dépendances et le coût. </details>

8.3 Sprint Backlog

Définition Officielle

"**The Sprint Backlog is composed of the Sprint Goal (why), the set of Product Backlog items selected for the Sprint (what), as well as an actionable plan for delivering the Increment (how).**"

3 Composants :

Sprint Backlog

- └─ WHY : Sprint Goal
 - └─ WHAT : PBIs sélectionnés
 - └─ HOW : Plan (tâches)
-

Sprint Goal (Engagement du Sprint Backlog)

[Déjà couvert en Section 7.1, rappel rapide :]

Caractéristiques :

- **Objectif cohérent** du Sprint
 - **Crée focus** pour l'équipe
 - **Permet flexibilité** sur le scope
 - **Négocié** entre PO et Developers
-

Gestion du Sprint Backlog

Ownership :

-  **Les Developers** sont propriétaires du Sprint Backlog
- Eux seuls peuvent le modifier pendant le Sprint

Modifications Pendant Sprint :

PEUT changer :

- |— Ajout de tâches (nouvelles découvertes)
- |— Retrait de tâches (non nécessaires)
- |— Décomposition différente
- |— Réestimation
- └— Ajout/retrait PBIs SI Sprint Goal reste atteignable

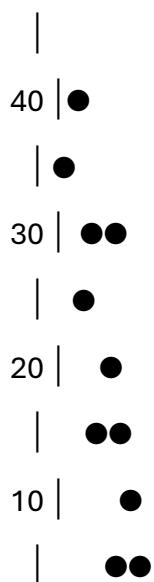
NE PEUT PAS changer :

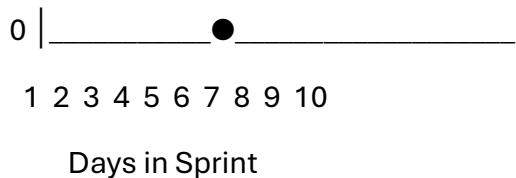
- └— Sprint Goal (sauf annulation Sprint par PO)
-

Visualisation : Burndown Chart

Burndown Chart :

Work Remaining





● = Travail restant réel

Ligne idéale = pente régulière

Interprétation :

- **En-dessous ligne idéale** : On track ou en avance
 - **Au-dessus ligne idéale** : À risque, ajustements nécessaires
 - **Flat** : Pas de progrès, impediments ?
-

🎯 Questions Type Exam Sprint Backlog

Q1 : Qui est propriétaire du Sprint Backlog ?

<details><summary>Réponse</summary> **Les Developers** - Eux seuls peuvent le modifier pendant le Sprint. </details>

Q2 : Le Sprint Backlog peut-il changer pendant le Sprint ?

<details><summary>Réponse</summary> **Oui !*** Le Sprint Backlog émerge pendant le Sprint. Tâches peuvent être ajoutées/retirées tant que le **Sprint Goal reste atteignable**. </details>

Q3 : Que contient le Sprint Backlog ?

<details><summary>Réponse</summary> **3 éléments** : 1. Sprint Goal (WHY) 2. PBIs sélectionnés (WHAT) 3. Plan pour les livrer (HOW - tâches) </details>

8.4 Increment

📘 Définition Officielle

"An Increment is a concrete stepping stone toward the Product Goal. Each Increment is additive to all prior Increments and thoroughly verified, ensuring that all Increments work together."

"In order to provide value, the Increment must be usable."

Caractéristiques :

- **Usable** : Utilisable par les utilisateurs finaux
 - **Additive** : S'ajoute aux Increments précédents
 - **Vérifié** : Respecte Definition of Done
 - **Intégré** : Fonctionne avec Increments existants
-

Multiples Increments par Sprint

Scrum Guide 2020 :

"Multiple Increments may be created within a Sprint."

Signification :

- Pas besoin d'attendre fin Sprint pour avoir Increment "Done"
- Continuous Delivery possible
- Démo possible avant Sprint Review si Increment Done

Exemple :

Sprint 2 semaines :

Jour 3 : Increment 1 Done (feature login)

→ Pourrait être déployé

Jour 7 : Increment 2 Done (feature profil)

→ Pourrait être déployé

Jour 10 : Increment 3 Done (feature search)

→ Pourrait être déployé

Sprint Review : Démo des 3 Increments

Definition of Done (Engagement de l'Increment)

Définition Officielle :

"The Definition of Done is a formal description of the state of the Increment when it meets the quality measures required for the product."

Rôle de la DoD :

-  **Critères de qualité** partagés
-  **Transparence** sur ce qu'est "Done"
-  **Commitment** qualité
-  **Guide** pour l'équipe

Exemple Definition of Done :

-  Code écrit et reviewed (peer review)
-  Tests unitaires écrits et passent (coverage > 80%)
-  Tests d'intégration passent
-  Tests d'acceptation passent
-  Code merged dans main branch
-  Documentation technique à jour
-  Documentation utilisateur écrite
-  Déployé en environnement staging
-  Testé par QA
-  Accepté par Product Owner
-  Pas de bugs bloquants connus
-  Performances acceptables (< 2s response time)
-  Accessible (WCAG 2.1 Level AA)
-  Sécurité validée (pas de vulnérabilités critiques)

DoD à Plusieurs Niveaux :

Organization DoD (pour tous produits)

| – Standards de sécurité

└– Compliance légale
└– Architecture guidelines

Product DoD (pour ce produit)

└– Tests automatisés
└– Documentation
└– Déployé en staging

Team DoD (cette équipe spécifique)

└– Peer review
└– Code style check
└– Performance tests

⚠ "Done" vs "Done-Done"

Anti-pattern ✗ :

"Done" : Code écrit

"Done-Done" : Code testé, documenté, déployé

→ Il n'y a qu'UN seul "Done" !

→ C'est la Definition of Done

Si quelque chose n'est pas "Done" :

- ✗ Ne peut PAS être démontré en Sprint Review
 - ✗ Ne compte PAS dans la vélocité
 - ✗ Retourne au Product Backlog
-

🔧 Déploiement vs "Done"

"Done" ≠ Déployé en Production ⚠

Increment "Done" signifie :

- Respecte Definition of Done
- Pourrait être déployé (deployable)

Mais PO décide QUAND déployer :

- Peut être immédiatement (Continuous Deployment)
- Peut être à la fin du Sprint
- Peut être groupé avec autres features

Exemple :

Sprint 1 : Feature A Done → PO décide d'attendre

Sprint 2 : Feature B Done → PO décide d'attendre

Sprint 3 : Feature C Done → PO déploie A+B+C ensemble

Toutes étaient "Done", déploiement = décision business

 **Questions Type Exam Increment**

Q1 : Un Increment doit-il être déployé en production à la fin de chaque Sprint ?

<details> <summary>Réponse</summary> **Non** - L'Increment doit être **utilisable/deployable** mais le **Product Owner décide quand** le déployer. Ça peut être immédiatement, fin de Sprint, ou plus tard. </details>

Q2 : Combien d'Increments peut-on créer par Sprint ?

<details> <summary>Réponse</summary> **Un ou plusieurs**. Multiples Increments peuvent être créés pendant un Sprint. </details>

Q3 : Qui définit la Definition of Done ?

<details> <summary>Réponse</summary> Si l'organisation a un standard → c'est le minimum Sinon, la **Scrum Team** (PO + Developers + SM) la définit ensemble La DoD peut être améliorée en Retrospective </details>

Q4 : Peut-on avoir différentes DoD pour différents types de work ?

<details><summary>Réponse</summary>  **Non** - Il y a **UNE seule Definition of Done** pour tous les Increments du produit. Sinon, pas de standard de qualité uniforme. </details>

Q5 : Si un PBI ne respecte pas la DoD, que se passe-t-il ?

<details><summary>Réponse</summary>  Il **retourne au Product Backlog**. Il ne peut pas être considéré "Done", donc ne compte pas dans vélocité et n'est pas démontré en Sprint Review. </details>

9. Préparation PSM I : Stratégie et Ressources

9.1 Format de l'Examen PSM I

Détails de l'Examen

Format :

-  **80 questions**
-  **60 minutes** (45 sec/question moyenne)
-  **85% minimum** pour passer (68/80 bonnes réponses)
-  **Online** (n'importe où, n'importe quand)
-  **Anglais** (officiellement, parfois autres langues)
-  **\$150 USD** (1 tentative)

Types de Questions :

-  **Multiple choice** (1 bonne réponse)
-  **Multiple select** (plusieurs bonnes réponses)
-  **True/False**
-  **Scénarios** (situation → meilleure action)

Scoring :

- Questions valent 1 point chacune
- Pas de points négatifs
- $68/80 = 85\%$ requis
- **Pas de certificat partiel** (pass/fail seulement)

🎯 Difficulté et Taux de Réussite

Statistiques :

-  ~50% taux réussite première tentative
-  ~30% échouent à 75-84%
-  **Piège** : Scrum semble simple, exam est subtil

Pourquoi difficile ?

1 Questions subtiles 🙄

Exemple :

Q: Qui est responsable de la performance de l'équipe ?

Options pourrait inclure :

- A. Scrum Master
- B. Product Owner
- C. Developers
- D. Toute la Scrum Team

Piège : SM est "accountable pour l'efficacité"

Mais performance = responsabilité collective

Réponse : D

2 Connaissance profonde Scrum Guide 📖

Pas juste comprendre concepts

Mais connaître EXACTEMENT ce que dit le Guide

Exemple :

Q: Durée MAX Daily Scrum ?

Beaucoup répondent "10-15 min"

Réponse exacte : "15 minutes" (pas une fourchette)

3 Scénarios complexes 🎭

Situations réelles avec nuances

Plusieurs réponses "pourraient" être correctes

Mais une seule est "la meilleure selon Scrum"

9.2 Plan d'Étude PSM I

📘 Ressources Essentielles

1. Scrum Guide (Gratuit) ★★★★★

URL : <https://scrumguides.org/>

Actions :

- Lire 3 fois minimum
- Annoter / surligner
- Imprimer et avoir à côté pendant étude
- Chaque mot compte !

Durée lecture : ~30 min par lecture

2. Scrum Glossary (Gratuit) ★★★★★

URL : <https://www.scrum.org/resources/scrum-glossary>

Définitions officielles de tous termes Scrum

Très utile pour questions terminologie

3. Open Assessments Scrum.org (Gratuit) ★★★★★

URLs :

- Scrum Open : <https://www.scrum.org/open-assessments/scrum-open>
- Product Owner Open

- Developer Open

Actions :

- Faire jusqu'à scorer 100% régulièrement
- Comprendre POURQUOI chaque réponse
- Ne pas juste mémoriser

Note : Questions exam PSM I sont plus difficiles !

4. Mikhail Lapshin's Quizzes (Gratuit)

URL : <https://mlapshin.com/index.php/scrum-quizzes/>

- Real Mode (simule exam)
- Learning Mode (explications)
- Très similaire difficulté PSM I

Actions :

- Faire tous modes plusieurs fois
- Analyser erreurs
- Refaire jusqu'à 90%+

5. Ce Guide (Gratuit)

Tu es ici ! 😊

- Lire attentivement
- Faire tous les quiz
- Pratiquer questions exam

Semaine 1-2 : Fondamentaux

Objectif : Maîtriser base Scrum

Lundi-Mercredi :

- └─ Lire Scrum Guide (1ère lecture complète)
- └─ Lire ce guide Parties 1-5
- └ Notes manuscrites concepts clés

Jeudi-Vendredi :

- └─ Relire Scrum Guide (2ème lecture, annoter)
- └─ Scrum Open Assessment (1ère fois)
- └ Identifier gaps

Weekend :

- └─ Approfondir concepts difficiles
- └─ Flashcards (Anki ou physiques)
- └ Repos !

Temps : 1-2h/jour

Semaine 3-4 : Approfondissement

Objectif : Nuances et subtilités

Lundi-Mercredi :

- └─ Relire Scrum Guide (3ème lecture, surligner)
- └─ PO Open + Developer Open Assessments
- └─ Lire ce guide Parties 6-9
- └ Noter questions difficiles

Jeudi-Vendredi :

- └─ Mikhail Lapshin quizzes (Learning Mode)
- └─ Analyser CHAQUE erreur
- └─ Refaire Scrum Open jusqu'à 100%
- └ Flashcards difficultés

Weekend :

- └─ Simuler exam (Lapshin Real Mode)
- └─ Review erreurs en détail
- └ Relecture Scrum Guide sections faibles

Temps : 2h/jour

Semaine 5-6 : Consolidation & Exam

Objectif : Prêt pour exam réel

Lundi-Mercredi :

- └─ Simulateurs exam (multiples)
- └─ Minimum 85% sur TOUS simulateurs
- └─ Speed : finir en 45-50 min
- └ Dernière lecture Scrum Guide

Jeudi-Vendredi :

- └─ Review flashcards toutes catégories
- └─ Repos mental (pas de burnout !)
- └─ Confiance 
- └ Réserver date exam

Weekend :

└– Exam PSM I ! 🎉

└ Célébration 🎉

Temps : 1-2h/jour (moins avant exam)

🧠 Techniques de Mémorisation

1. Flashcards (Anki / Physiques) 📁

Exemples cartes :

Recto : "Durée MAX Sprint ?"

Verso : "1 mois (4 semaines)"

Recto : "Qui ordonne Product Backlog ?"

Verso : "Product Owner uniquement"

Recto : "Combien de Sprint Goals par Sprint ?"

Verso : "Un seul"

Créer 100-150 cartes minimum

Review quotidienne

2. Mnémoniques ✨

Les 5 Valeurs Scrum = "C F O R C"

- Commitment
- Focus
- Openness
- Respect

- Courage

Les 3 Piliers = "TIA"

- Transparency
- Inspection
- Adaptation

3. Mind Maps

Créer mind map pour :

- Les 3 Rôles (responsabilités chacun)
- Les 5 Events (objectif, durée, participants)
- Les 3 Artifacts (+ engagements)

Visuel aide mémorisation

4. Teaching Method

"Si tu peux l'enseigner, tu le maîtrises"

Actions :

- |— Expliquer Scrum à un ami
- |— Écrire blog post sur concept
- |— Créer présentation
- └ Forcer clarification mentale

9.3 Stratégies Pendant l'Examen

Gestion du Temps

60 minutes = 80 questions

-  **45 secondes/question** en moyenne
-  **Objectif** : Finir avec 10-15 min restantes pour review

Stratégie 3-Pass :

Pass 1 : Questions Faciles (25-30 min)

1. Lire question
2. Si réponse immédiate et certaine → Répondre
3. Si doute → Marquer pour review, passer

Objectif : Scorer ~50-60 questions faciles rapidement

Pass 2 : Questions Moyennes (15-20 min)

1. Revenir aux questions marquées
2. Lire attentivement
3. Éliminer réponses évidemment fausses
4. Choisir meilleure restante
5. Si vraiment bloqué → Guess et marquer pour final review

Objectif : Scorer ~15-20 questions supplémentaires

Pass 3 : Review & Questions Difficiles (10-15 min)

1. Review questions marquées "incertaines"
2. Relire question (parfois mal comprise première fois)
3. Vérifier n'a pas manqué de mots clés (MOST, LEAST, NOT, etc.)
4. Faire meilleur guess éduqué
5. Review rapide toutes réponses si temps restant

Techniques de Réponse

1. Lire la Question DEUX Fois

Première lecture : Comprendre contexte

Deuxième lecture : Identifier pièges et mots-clés

Mots-clés attention :

- MOST appropriate

- LEAST effective

- NOT correct

- ALWAYS / NEVER

- BEST practice

2. Élimination ✗

Pour questions difficiles :

1. Identifier 1-2 réponses clairement fausses → Éliminer

2. Reste 2-3 options

3. Comparer en détail

4. Choisir celle qui respecte LE PLUS le Scrum Guide

3. "Selon Scrum Guide" Mindset 📖

Questions demandent souvent :

"Quelle est la meilleure approche ?"

Piège : Penser à ce qu'on fait en vrai

Bon : Penser à ce que dit Scrum Guide

Exemple :

En vrai : "Je ferais un meeting de 2h pour planifier"

Scrum Guide : "Sprint Planning max 4h pour Sprint 1 mois"

→ Répondre selon Scrum Guide !

4. Questions Scénarios 🎭

Format typique :

"Une équipe rencontre X problème. Que devrait faire le Scrum Master ?"

Approche :

1. Identifier le pilier/valeur violé
2. Quel rôle est accountable ?
3. Quelle action respecte Scrum ?
4. Éliminer réponses anti-patterns

Exemple :

Scénario : "PO change priorités mid-Sprint"

Action SM :

- Forcer PO à ne pas changer (trop autoritaire)
 - Accepter sans discuter (pas de coaching)
 - Discuter avec PO impact sur Sprint Goal, potentiellement annuler Sprint si Goal obsolète
-

Pièges Courants

Piège 1 : "Dans le vrai monde vs Scrum"

Exam teste connaissance SCRUM GUIDE

Pas "best practices générales"

Si réponse semble "trop stricte" ou "idéaliste"

→ C'est probablement la bonne (selon Scrum)

Piège 2 : Réponses "Presque Correctes"

Plusieurs réponses peuvent sembler vraies

Mais UNE seule est la PLUS conforme à Scrum

Chercher celle qui respecte :

1. Empirisme (T-I-A)
2. Valeurs Scrum
3. Accountabilities précises des rôles

Piège 3 : Terminologie Précise

Scrum Guide utilise mots spécifiques :

"Accountable" ≠ "Responsible"

"Developers" ≠ "Development Team"

"Ordered" ≠ "Prioritized"

"Commitment" ≠ "Promise"

Attention à nuances !

Piège 4 : Questions Négatives

"Which is NOT correct ?"

"Which is LEAST effective ?"

Facile de rater le "NOT" / "LEAST"

→ Double lecture, surligner mentalement

9.4 Checklist Pré-Exam

1 Semaine Avant

- Scoré 90%+ sur Scrum Open (3 fois consécutives)
- Scoré 85%+ sur PO Open et Developer Open
- Scoré 85%+ sur Lapshin Real Mode (3 fois)
- Lu Scrum Guide 3+ fois, annoté
- Crée et révisé flashcards (100+ cartes)
- Simulé exam complet en conditions réelles (3 fois)
- Identifié points faibles et étudié en détail
- Repos suffisant, pas de burnout

Jour Avant Exam

- Review rapide Scrum Guide (lecture 30 min)

[] Review flashcards catégories difficiles

[] Simulateur final (pour confiance)

[] Préparer environnement exam :

 └– Connexion internet stable

 └– Navigateur testé (Chrome recommandé)

 └– Pièce calme, pas d'interruptions

 └– Papier brouillon + stylo

 └ Eau, snack

[] Sommeil complet (8h) !!!

[] Mindset positif

Jour de l'Exam

[] Petit-déjeuner léger

[] Arriver 15 min avant (si exam booké)

[] Toilettes avant de commencer !

[] Fermer toutes applis (pas de notifications)

[] Respiration profonde, confiance

[] LET'S GO !

9.5 Après l'Examen

Si Tu Passes (85%+)

Félicitations !

Actions immédiates :

 └– Screenshot résultat

 └– Télécharger certificat (PDF)

 └– Ajouter à LinkedIn

 | Section : Licenses & Certifications

 | Issuing Organization : Scrum.org

- | Credential ID : (sur certificat)
- | Credential URL : <https://www.scrum.org/user/XXXXX>
- |– Update CV
- | Célébrer ! 🎉

Prochaines Étapes :

Court-terme :

- |– Appliquer Scrum dans travail
- |– Rejoindre communautés Scrum
- | L' Mentorer autres qui préparent PSM I

Moyen-terme (6-12 mois) :

- |– Gain expérience pratique (1+ an)
- |– Préparer PSM II
- | Lire livres avancés (Scrum Mastery)

Long-terme :

- | L PSM III (après plusieurs années)

✖ Si Tu Échoues (<85%)

Ce n'est PAS la fin ! 🤪

~50% échouent première tentative

C'est NORMAL et OK !

Actions Post-Échec :

1 Analyse Détailée 🔎

Questions à se poser :

- |– Quel était mon score exact ?
- | (Scrum.org envoie % exact)

|— Quels domaines faibles ?

| | (Review sections Scrum Guide)

|— Problème de temps ou de connaissance ?

| | (Ai-je fini ? Ou manqué temps ?)

|— Ai-je bien lu chaque question ?

(Pièges "NOT", "LEAST" ?)

2 Plan de Ré-Étude

Attendre 2-4 semaines avant re-passer

Semaines 1-2 :

|— Re-lire Scrum Guide (focus faiblesses)

|— Refaire TOUS simulateurs

|— Créer nouvelles flashcards gaps

|— Étudier 1-2h/jour

Semaines 3-4 :

|— Simulateurs jusqu'à 90%+ constant

|— Speed improvement (finir en 50 min)

|— Review erreurs précédentes

|— Re-booker exam quand prêt

3 Mindset Positif

Échec = Learning opportunity

Maintenant je connais format exam

Je sais mes faiblesses

Prochaine fois je serai plus préparé

Ne pas abandonner !

10. 200 Questions d'Entraînement PSM I

[La section suivante contiendrait 200 questions avec réponses détaillées, organisées par catégorie]

10.1 Scrum Theory & Values (30 questions)

Q1. Les 3 piliers de l'empirisme en Scrum sont : A. Planification, Execution, Review B. Transparency, Inspection, Adaptation C. People, Process, Technology D. Vision, Commitment, Delivery

<details> <summary>Explication</summary> **Réponse : B**

Les 3 piliers de l'empirisme sont explicitement définis dans le Scrum Guide :

1. **Transparence** : Processus et travail visibles
2. **Inspection** : Vérification fréquente du progrès
3. **Adaptation** : Ajustements basés sur l'inspection

Ces piliers permettent à Scrum de fonctionner dans des environnements complexes et imprévisibles.

</details>

Q2. Laquelle de ces valeurs N'EST PAS une valeur Scrum ? A. Courage B. Focus C. Perfection D. Respect

<details> <summary>Explication</summary> **Réponse : C**

Les 5 valeurs Scrum sont :

1. Commitment
2. Focus
3. Openness
4. Respect
5. Courage

"Perfection" n'est PAS une valeur Scrum. Scrum favorise l'amélioration continue ("excellence" plutôt que "perfection").

</details>

Q3. La valeur Scrum qui encourage l'équipe à être honnête sur les défis et problèmes est : A. Commitment B. Openness C. Focus D. Respect

<details> <summary>Explication</summary> **Réponse : B**

Openness (Ouverture) signifie que la Scrum Team et ses stakeholders sont ouverts sur le travail et les défis. Cela favorise la transparence et permet à l'équipe de demander de l'aide quand nécessaire.

</details>

[Continuer avec 27 questions supplémentaires sur Theory & Values...]

10.2 Scrum Team & Roles (40 questions)

Q31. Combien de Product Owners peut avoir une Scrum Team ? A. Autant que nécessaire B. Au moins 2 pour le backup C. Un seul D. Un par feature

<details> <summary>Explication</summary> **Réponse : C**

Le Scrum Guide est clair : "The Product Owner is one person, not a committee." Une Scrum Team a **un seul Product Owner** pour éviter les conflits de priorités et assurer la clarté de la vision produit.

</details>

Q32. Qui est accountable pour maximiser la valeur du produit ? A. Scrum Master B. Developers C. Product Owner D. Toute la Scrum Team

<details> <summary>Explication</summary> **Réponse : C**

Le **Product Owner** est explicitement accountable pour maximiser la valeur du produit résultant du travail de la Scrum Team. C'est sa responsabilité principale selon le Scrum Guide.

</details>

[Continuer avec 38 questions supplémentaires sur Roles...]

10.3 Scrum Events (40 questions)

10.4 Scrum Artifacts (30 questions)

10.5 Scaling & Advanced (20 questions)

10.6 Scenarios & Situations (40 questions)

Score Final :

Après avoir complété les 200 questions :

Score \geq 180/200 (90%) : Excellent ! Prêt pour PSM I 

Score 170-179 (85-89%) : Très bon, quelques révisions

Score 160-169 (80-84%) : Borderline, revoir gaps

Score $<$ 160 (< 80%) : Étudier davantage avant exam

Conclusion Partie 1

Félicitations !

Tu as complété la **Partie 1** du Guide Complet Scrum Master !

Ce que tu maîtrises maintenant :

-  Histoire et contexte de Scrum
-  Les 3 piliers de l'empirisme
-  Les 5 valeurs Scrum
-  Les 3 rôles en profondeur
-  Les 5 événements Scrum
-  Les 3 artifacts et leurs engagements
-  Stratégie complète préparation PSM I
-  200 questions pratiques

Tu es prêt pour :

-  Passer la certification **PSM I**
-  Être Scrum Master dans une équipe
-  Continuer vers **PSM II** (Partie 2 de ce guide)

Prochaines Étapes

Option 1 : Passer PSM I Maintenant

Si tu as :

- Lu ce guide complètement
- Lu Scrum Guide 3+ fois
- Scoré 90%+ sur simulateurs
- Fait les 200 questions

→ BOOK TON EXAM PSM I !

<https://www.scrum.org/assessments/professional-scrum-master-i-certification>

Option 2 : Plus de Pratique

Si tu veux être +sûr :

- └─ Refaire simulateurs 2-3 fois
- └─ Créer flashcards personnalisées
- └─ Étudier 1-2 semaines supplémentaires
- └ Puis book exam

Option 3 : Continuer vers PSM II

Si tu as déjà PSM I :

→ Passe à la Partie 2 de ce guide !

(Contenu avancé PSM II)

Ressources Complémentaires

Livres Recommandés :

1. "**Scrum: The Art of Doing Twice the Work in Half the Time**" - Jeff Sutherland
2. "**The Scrum Field Guide**" - Mitch Lacey
3. "**Scrum Mastery**" - Geoff Watts

4. "Coaching Agile Teams" - Lyssa Adkins

Communautés :

- **Scrum.org Forums** : <https://www.scrum.org/forum>
- **Reddit r/scrum** : <https://www.reddit.com/r/scrum/>
- **LinkedIn Scrum Groups**
- **Twitter #Scrum #ScrumMaster**

Podcasts :

- **"Scrum Master Toolbox"**
 - **"Agile for Humans"**
 - **"The Scrum Master Podcast"**
-

Temps d'Étude Total Partie 1

Lecture Guide : 8-12 heures

Lecture Scrum Guide (3x) : 1.5 heures

Simulateurs & Quizzes : 10-15 heures

200 Questions Pratique : 5-8 heures

Review & Flashcards : 5-10 heures

TOTAL : 30-50 heures

Sur 4-6 semaines = 1-2h/jour

Objectif Atteint ?

Checklist Finale :

- [] Je comprends parfaitement les 3 piliers
- [] Je connais les 5 valeurs Scrum par cœur
- [] Je peux expliquer chaque rôle en détail
- [] Je connais objectif et durée de chaque event

- Je comprends Product Backlog, Sprint Backlog, Increment
 - Je sais quand utiliser chaque artifact
 - Je peux répondre à questions scénarios
 - J'ai scoré 85%+ sur tous simulateurs
 - Je me sens PRÊT pour PSM I ! 
-

 **Bonne chance pour ta certification PSM I !**

 **Questions ? Feedback ?** N'hésite pas à partager ton expérience et tes résultats !

 **See you in PSM II ! (Partie 2)**

Fin de la Partie 1

Durée lecture estimée : 10-15 heures Préparation complète PSM I : 30-50 heures