Projet FlexOffice - Mini-cas 2 - Adam GABRIEL, Louis LEULIETTE, Gauvain BOICHÉ

Gestion de projet - Mini cas 2 - Simulation pratique de projet agile

Projet FlexOffice : application interne de réservation de salles, de matériel informatique et de véhicules de service

Ecole	LiveCampus
Promotion	MCS P 2027.2
Cours	Gestion de projet
Professeur	Yannick AUMAGY
Groupe	Adam GABRIEL - Louis LEULIETTE - Gauvain BOICHÉ
Date de remise	[07/10/2026]

> L'entreprise FlexOffice veut lancer une application interne permettant aux salariés de réserver des salles de réunion, du matériel informatique et des véhicules de service. La direction souhaite disposer rapidement d'une première version afin de tester le service sur un site pilote. Vous travaillez en équipe agile. Votre objectif est de produire un plan de sprint cohérent, suivre l'avancement et préparer une restitution finale.

Backlog de départ

Fonctionnalité	Priorité initiale
Connexion utilisateur	Must
Consulter les ressources disponibles	Must
Réserver une salle	Must
Annuler une réservation	Must
Réserver du matériel informatique	Should
Réserver un véhicule	Should
Recevoir une confirmation par courriel	Should
Afficher l'historique des réservations	Could
Administration des ressources	Must
Statistiques d'utilisation	Could
Synchronisation avec un agenda externe	Could
Notifications mobiles	Could

Tâches demandées

Phase 1 - Préparation du backlog

- Reformuler les besoins en histoires utilisateur.
- Ajouter des **Critères d'acceptation**.
- Prioriser les éléments selon la valeur et le risque.

Phase 2 - Estimation et Sprint Planning

- Estimer les histoires utilisateur en points d'histoire.
- Définir une capacité réaliste pour le sprint.
- Sélectionner les éléments à réaliser.
- Découper les stories retenues en tâches techniques ou fonctionnelles.

Phase 3 - Mise en place du suivi

- Créer un tableau Kanban avec au minimum À faire / En cours / À valider / Terminé.
- Définir 3 indicateurs de suivi.
- Identifier les principaux risques initiaux.

Phase 4 - Qualité et livraison

- Définir la Definition of Done.
- Préciser les tests nécessaires avant validation.
- Décrire comment Git et une chaîne CI/CD seraient utilisés.
- Préparer la liste des livrables de fin de sprint.

Sommaire

1. Synthèse du plan de projet
2. Phase 1 - Préparation du backlog
3. Phase 2 - Estimation et Sprint Planning
4. Phase 3 - Mise en place du suivi
5. Phase 4 - Qualité et livraison
6. Planning et diagramme de Gantt
7. Estimation des ressources et des délais
8. Plan de gestion des risques
9. Hypothèses retenues

1. Synthèse du plan de projet

1.1 Vision et objectifs

Vision. Offrir aux salariés de FlexOffice un point d'entrée unique pour la réservation de salles, de matériel informatique et de véhicules de service, afin d'éviter les doublons de réservation et les échanges informels non-traçables (courriels, tableurs, etc).

Objectif court terme. Livrer rapidement une première version utilisable (MVP) pour la tester sur **un site pilote**, puis décider de la généralisation à partir des retours réels.

Objectifs mesurables du pilote *(à valider avec la direction)* :

- 100 % des réservations de salles du site pilote passent par l'application ;
- aucune double-réservation constatée ;
- satisfaction utilisateur supérieure à 4/5 lors de l'enquête de fin de pilote.

1.2 Périmètre

Dans le périmètre (pilote)	Hors périmètre (reporté après le pilote)
Connexion, rôles employé / administrateur	Synchronisation avec un agenda externe
Administration et consultation des ressources	Notifications mobiles
Réservation et annulation (salles, matériel, véhicules)	Statistiques avancées (selon retours du pilote)
Confirmation par courriel, historique	Déploiement multi-sites

1.3 Équipe et organisation

Rôle	Effectif	Mission
Product Owner (PO)	1 (50 %)	Porte la vision, priorise le backlog, valide les stories
Scrum Master (SM)	1 (50 %)	Facilite les cérémonies, lève les obstacles
Développeurs	3 (100 %)	Front-end, back-end, full-stack
Testeur / QA	1 (50 %)	Stratégie de test, recette, automatisation

Cadre agile : Scrum avec sprints de **2 semaines**, suivi visuel Kanban, Git et CI/CD dès le premier sprint.

1.4 Calendrier et jalons

Jalon	Échéance	Contenu
M1 - Backlog prêt	Fin S1 (semaine 1)	Backlog raffiné, socle et outils prêts
M2 - Démo Sprint 1	Fin semaine 3	Connexion, ressources, réservation de salle
M3 - Version pilote candidate	Fin semaine 5	Toutes les stories Must et Should du pilote
M4 - Version validée (recette)	Fin semaine 7	Recette utilisateur validée, correctifs livrés

Jalon	Échéance	Contenu
M5 - Mise en service du pilote	Semaine 8	Déploiement et formation du site pilote
M6 - Bilan du pilote	Semaine 12	Retours, décision Go / No-Go généralisation

1.5 Chiffres clés

Indicateur	Valeur
Durée jusqu'à la mise en service du pilote	8 semaines
Nombre de sprints de développement	3 (+ 1 semaine de cadrage)
Capacité d'un sprint	24 point d'histoires
Backlog total estimé	68 point d'histoires (13 items), dont 42 pour le périmètre pilote
Charge totale estimée (jusqu'à la fin du pilote)	≈ 173 jours-personnes

1.6 Principaux risques

Les trois risques les plus critiques sont le **double réservations**, l'**intégration de l'authentification** d'entreprise et une **capacité surestimée** en l'absence de vélocité historique. Le détail et les stratégies d'atténuation figurent en section 8.

2. Phase 1 - Préparation du backlog

2.1 Histoires Utilisateurs et **Critères d'acceptation**

HU-00 - Socle technique et chaîne CI/CD

En tant que dev, je veux du versionnage avec une CI afin de livrer des version fiables et identifiées dès le premier sprint.

Critères d'acceptation

- Après un push, quand la CI s'exécute, alors build, lint et tests unitaires sont lancés automatiquement.
- Après un merge sur la branche main, quand la CI passe, alors l'application est déployée.
- Le dépôt contient un README décrivant l'installation locale.

HU-01 - Connexion utilisateur

En tant que salarié, je veux me connecter avec mon compte d'entreprise afin d'accéder à mes réservations.

Critères d'acceptation

- Le salarié disposant d'un compte valide, à saisie de ses identifiants, il accède à la page d'accueil.
- En cas d'identifiants invalides, un message d'erreur générique s'affiche et aucun accès n'est accordé.
- Les rôles « employé » et « administrateur » donnent accès à des fonctions différentes ; la session expire après une période d'inactivité définie.

HU-02 - Consulter les ressources disponibles

En tant que salarié, je veux consulter les ressources disponibles sur un créneau donné afin de choisir celle qui correspond à mon besoin.

Critères d'acceptation

- Avec une date et un créneau, en cas de recherche, seules les ressources libres sur ce créneau sont affichées.
- Je peux filtrer par type (salle, matériel, véhicule), par site et par capacité.
- Le temps d'affichage des résultats est inférieur à 2 secondes pour 200 ressources.

HU-03 - Réserver une salle

En tant que salarié, je veux réserver une salle de réunion sur un créneau.

Critères d'acceptation

- Avec une salle libre, je valide la réservation, et elle est enregistrée. La salle devient indisponible sur ce créneau.
- Si une salle est déjà réservée, quand je tente de la réserver, la demande est refusée avec un message explicite.
- Une réservation ne peut pas être créée dans le passé ni en dehors des horaires d'ouverture.

HU-04 - Annuler une réservation

En tant que salarié, je veux annuler ma réservation.

Critères d'acceptation

- Quand j'annule une réservation dans le futur et de mon fait, la ressource redevient disponible immédiatement.
- Je ne peux pas annuler la réservation d'un autre salarié (sauf administrateur).
- Une confirmation m'est demandée avant l'annulation définitive.

HU-05 - Réserver du matériel informatique

En tant que salarié, je veux réserver un ordinateur portable (ou autre équipement).

Critères d'acceptation

- Si un équipement est disponible, quand je le réserve, il est bloqué sur la période choisie.
- Le matériel peut être réservé sur plusieurs jours consécutifs.
- Un équipement marqué « hors service » n'est pas proposé à la réservation.

HU-06 - Réserver un véhicule

En tant que salarié, je veux réserver un véhicule de service pour un déplacement professionnel.

Critères d'acceptation

- Si un véhicule est libre, je le réserve en indiquant un motif et une durée.
- Le formulaire demande la destination et la durée estimée du déplacement.
- Un véhicule en maintenance ou déjà réservé n'est pas proposé.

HU-07 - Recevoir une confirmation par courriel

En tant que salarié, je veux recevoir un courriel à chaque réservation ou annulation afin d'avoir une trace écrite et un rappel.

Critères d'acceptation

- Si une réservation est créée, quand elle est validée, un courriel récapitulatif (ressource, date, horaires) est envoyé dans les 2 minutes.
- Un courriel d'annulation est envoyé lors de toute annulation.
- Un échec d'envoi ne bloque pas la réservation et est consigné dans les journaux.

HU-08 - Afficher l'historique des réservations

En tant que salarié, je veux consulter l'historique de mes réservations passées et à venir.

Critères d'acceptation

- Quand j'ouvre « Mes réservations », la liste est affichée du plus récent au plus ancien.
- Je peux filtrer par statut (à venir, passée, annulée) et par type de ressource.
- Je ne vois que mes propres réservations.

HU-09 - Administration des ressources

En tant qu'administrateur, je veux créer, modifier et désactiver des ressources afin de maintenir un catalogue à jour.

Critères d'acceptation

- En tant qu'admin, quand je crée une ressource (type, nom, site, capacité), elle apparaît dans le catalogue.
- La désactivation d'une ressource la retire de la recherche sans supprimer l'historique existant.
- Un salarié non administrateur n'a pas accès aux écrans d'administration.

HU-10 - Statistiques d'utilisation

En tant que responsable des services généraux, je veux visualiser des statistiques d'utilisation.

Critères d'acceptation

- Un tableau de bord affiche le taux d'occupation par type de ressource et par période.
- Les données peuvent être exportées au format CSV.
- Les statistiques sont réservées aux administrateurs.

HU-11 - Synchronisation avec un agenda externe

En tant que salarié, je veux que mes réservations apparaissent dans mon agenda.

Critères d'acceptation

- Avec un compte agenda lié, je crée une réservation, et un événement est créé dans mon agenda.
- L'annulation d'une réservation supprime l'événement associé.
- Le salarié peut dissocier son agenda à tout moment.

HU-12 - Notifications mobiles

En tant que salarié, je veux recevoir des notifications sur mon téléphone afin d'être rappelé de mes réservations.

Critères d'acceptation

- Un rappel est envoyé 30 minutes avant le début d'une réservation.
- L'utilisateur peut activer ou désactiver les notifications.
- Aucune notification n'est envoyée pour une réservation annulée.

2.2 Priorisation selon la valeur et le risque

Méthode. Chaque objet reçoit une note de **valeur métier** (1-5) et une note de **risque** (1-5 : complexité technique, incertitude, dépendance externe). Le **score = valeur + risque** favorise les éléments à forte valeur et ceux dont l'incertitude doit être levée tôt. Le rang final tient aussi compte des **dépendances** (on ne consulte pas des ressources qui n'ont pas pu être créées) et de l'objectif du pilote.

Rang	ID	Fonctionnalité	Priorité initiale	Valeur	Risque	Score	Priorité révisée	Sprint cible
1	HU-00	Socle technique et CI/CD	-	3	3	6	Must	S1
2	HU-01	Connexion utilisateur	Must	5	4	9	Must	S1
3	HU-09	Administration des ressources	Must	4	2	6	Must	S1

Rang	ID	Fonctionnalité	Priorité initiale	Valeur	Risque	Score	Priorité révisée	Sprint cible
4	HU-02	Consulter les ressources	Must	5	2	7	Must	S1
5	HU-03	Réserver une salle	Must	5	4	9	Must	S1
6	HU-04	Annuler une réservation	Must	4	2	6	Must	S2
7	HU-05	Réserver du matériel	Should	3	3	6	Should	S2
8	HU-07	Confirmation par courriel	Should	3	3	6	Should	S2
9	HU-06	Réserver un véhicule	Should	3	4	7	Should	S2
10	HU-08	Historique des réservations	Could	2	1	3	Should *(↑)*	S2
11	HU-10	Statistiques d'utilisation	Could	2	2	4	Could	S3 si capacité
12	HU-11	Synchronisation agenda externe	Could	2	5	7	Won't *(version pilote)*	Après pilote
13	HU-12	Notifications mobiles	Could	2	4	6	Won't *(version pilote)*	Après pilote

Décisions de priorisation justifiées

- **HU-00 ajoutée** : sans socle ni CI/CD, aucun incrément fiable n'est possible ; c'est un prérequis technique.
- **Administration (Must) avancée avant la consultation** : sans ressources dans le catalogue, la consultation et la réservation ne peuvent être testées. Son périmètre est volontairement réduit au minimum (créer / modifier / désactiver).
- **Connexion et réservation de salle traitées dans le premier sprint** : ce sont les deux items les plus risqués (SSO, gestion des conflits). Les traiter tôt réduit l'incertitude.
- **Historique remonté en Should** : forte valeur perçue pour un coût faible (score de risque de 1) ; il complète naturellement l'annulation.
- **Synchronisation agenda et notifications mobiles repoussées** : risque élevé (API externes, canal mobile) pour une valeur faible sur un pilote ; elles seront réévaluées avec les retours du site pilote.

3. Phase 2 - Estimation et Sprint Planning

3.1 Estimation en points d'histoire

Méthode. Planning Poker avec la suite de Fibonacci (1, 2, 3, 5, 8, 13), qui est appliqué pour éviter les hésitation entre 5 et 6 (points d'estimation) pour les tâches dures à estimer. L'histoire de référence est **HU-02 « Consulter les ressources » = 3 points** (moyenne complexité : lecture de données, filtres, un écran). Les autres stories sont estimées par comparaison relative, en intégrant effort, complexité et incertitude.

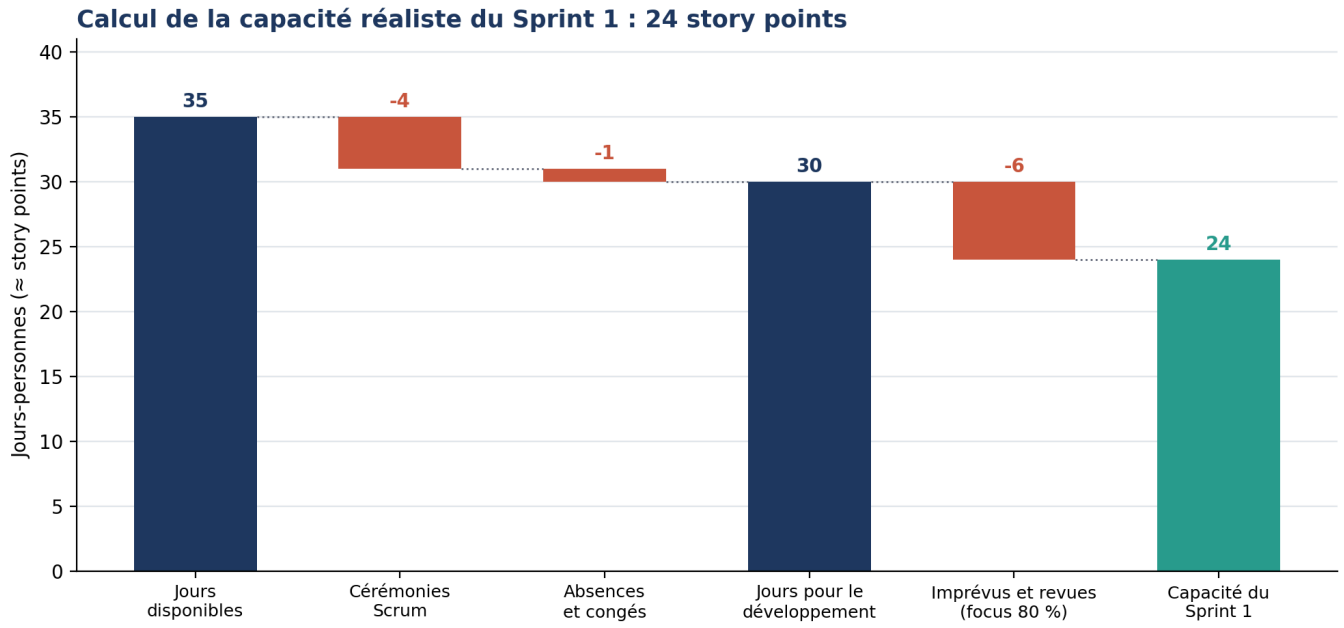
ID	Histoire Utilisateur	Points	Justification
HU-00	Socle technique et CI/CD	3	Mise en place connue, mais nécessite plusieurs briques (dépôt, pipeline, recette)
HU-01	Connexion utilisateur	5	Intégration à l'authentification d'entreprise, gestion des rôles et sessions
HU-02	Consulter les ressources	3	**Référence** : lecture, filtres, un écran
HU-03	Réserver une salle	8	Règles métier, contrôle des conflits et concurrence, interface de sélection de créneau
HU-04	Annuler une réservation	3	Règles de droits simples, remise à disposition de la ressource
HU-05	Réserver du matériel	5	Réutilise la logique de réservation, ajoute la durée multi-jours et les statuts
HU-06	Réserver un véhicule	5	Réutilise la logique de réservation, ajoute motif, destination et maintenance
HU-07	Confirmation par courriel	3	Service d'envoi, modèles de message, gestion des échecs
HU-08	Historique des réservations	2	Liste filtrée sur des données existantes
HU-09	Administration des ressources	5	CRUD complet, droits, désactivation sans perte d'historique
HU-10	Statistiques d'utilisation	5	Agrégations, tableau de bord, export
HU-11	Synchronisation agenda externe	13	API tierce, authentification déléguée, cas d'erreur nombreux (à découper)
HU-12	Notifications mobiles	8	Canal push, gestion des appareils et des préférences
	Total backlog	**68**	

> Le backlog complet représente **68 points** (13 items) ; le périmètre de la version pilote (HU-00 à HU-09) en représente **42**, le reste étant reporté après le pilote.

3.2 Capacité réaliste du sprint

L'équipe n'a pas encore d'historique de vélocité : la capacité est donc **calculée à partir de la disponibilité**, puis recalibrée après le Sprint 1.

Étape	Calcul	Résultat
Jours disponibles (3 développeurs × 10 j + QA 0,5 × 10 j)	30 + 5	35 j-p
Cérémonies Scrum (planning, daily, review, rétro, raffinement)	≈ 11 %	− 4 j-p
Absences et congés prévus		− 1 j-p
Jours disponibles pour le développement		30 j-p
Facteur de focus (imprévus, support, revues de code)	× 0,8	**24 j-p**

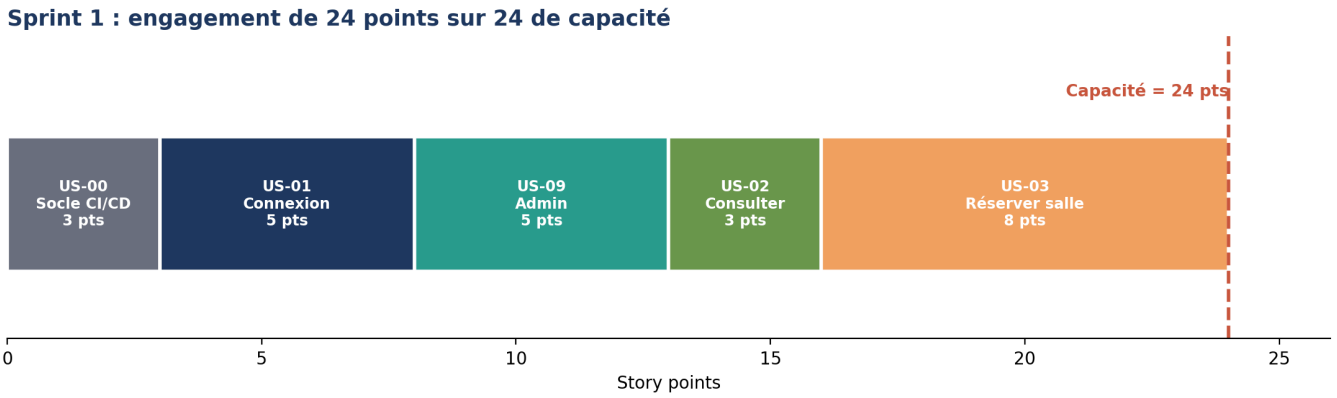


Hypothèse de calibrage : **1 point d'histoire ≈ 1 jour-personne idéal**. **Capacité retenue pour le Sprint 1 : 24 point d'histoires.** Cette valeur sera ajustée selon la vélocité réellement constatée.

3.3 Sélection des éléments : Sprint 1

Objectif du sprint (Sprint Goal) : *« Un salarié authentifié peut consulter les ressources de son site et réserver une salle sans double réservation ; un administrateur peut alimenter le catalogue. »*

Rang	ID	Histoire Utilisateur	Points
1	HU-00	Socle technique et CI/CD	3
2	HU-01	Connexion utilisateur	5
3	HU-09	Administration des ressources	5
4	HU-02	Consulter les ressources	3
5	HU-03	Réserver une salle	8
		Total engagé	**24**



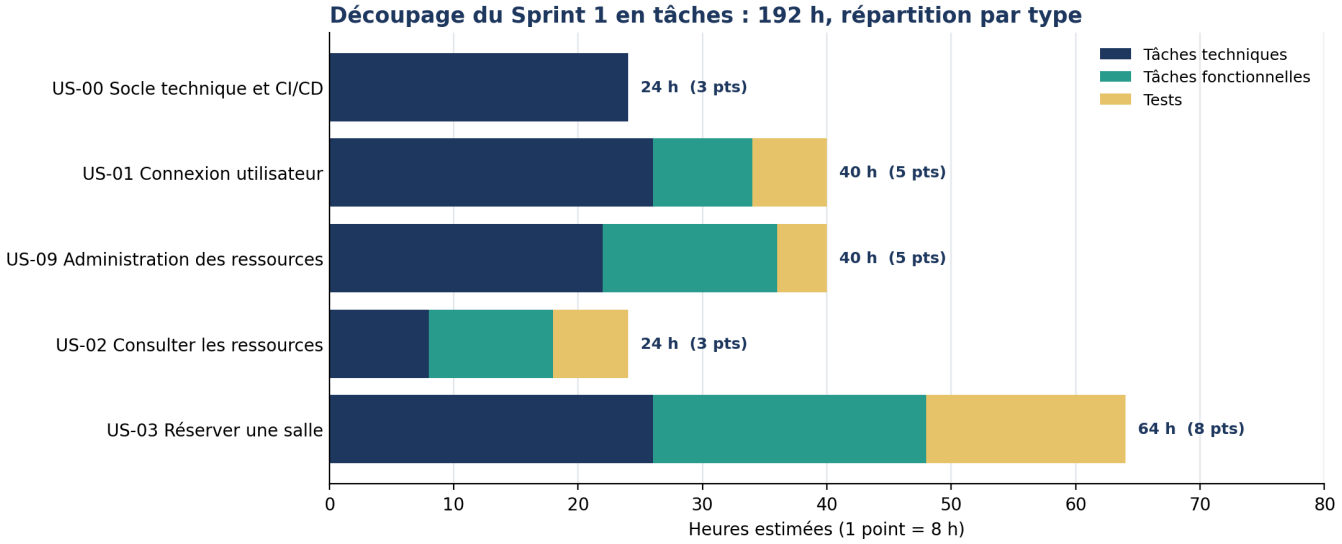
L'engagement égale la capacité calculée (24 points) : le périmètre est ambitieux mais réaliste. Si le Sprint 1 est en retard, **HU-03 est l'histoire à risque** ; le PO accepterait alors de réduire son périmètre (par exemple, sans gestion fine des horaires d'ouverture) plutôt que de dégrader la qualité.

Aperçu des sprints suivants *(prévisionnel, à ajuster après chaque sprint)*

Sprint	Objectif	Contenu	Points
Sprint 2	Compléter la version pilote	HU-04 (3), HU-05 (5), HU-07 (3), HU-06 (5), HU-08 (2)	18 *(+ 6 de réserve pour correctifs du S1)*
Sprint 3	Stabiliser et préparer le pilote	Recette utilisateur, correctifs, tests de charge, documentation, formation, déploiement ; HU-10 si capacité	24 *(dont 12 de stabilisation)*

3.4 Découpage des stories du Sprint 1 en tâches

1 point = 8 heures. Total : **192 heures**, soit 24 jours-personnes.



HU-00 - Socle technique et CI/CD (3 pts = 24 h)

ID tâche	Tâche	Type	Estimation
T-00.1	Créer le dépôt Git, les règles de branches et le README	Technique	2 h

ID tâche	Tâche	Type	Estimation
T-00.2	Initialiser le squelette front-end et back-end	Technique	6 h
T-00.3	Mettre en place la base de données et les migrations	Technique	6 h
T-00.4	Configurer le pipeline CI (build, lint, tests)	Technique	6 h
T-00.5	Provisionner l'environnement de recette et le déploiement automatique	Technique	4 h

HU-01 - Connexion utilisateur (5 pts = 40 h)

ID tâche	Tâche	Type	Estimation
T-01.1	Spike : analyser l'annuaire / SSO de l'entreprise	Technique	4 h
T-01.2	Développer l'API d'authentification (jeton / session)	Technique	12 h
T-01.3	Développer l'écran de connexion	Fonctionnel	8 h
T-01.4	Implémenter la gestion des rôles (employé / administrateur)	Technique	6 h
T-01.5	Écrire les tests unitaires et d'intégration	Test	6 h
T-01.6	Revue de sécurité (mots de passe, expiration de session)	Technique	4 h

HU-09 - Administration des ressources (5 pts = 40 h)

ID tâche	Tâche	Type	Estimation
T-09.1	Concevoir le modèle de données « ressource »	Technique	6 h
T-09.2	Développer l'API CRUD des ressources	Technique	12 h
T-09.3	Développer les écrans d'administration (liste, formulaire)	Fonctionnel	14 h
T-09.4	Gérer la désactivation / l'archivage	Technique	4 h
T-09.5	Écrire les tests	Test	4 h

HU-02 - Consulter les ressources (3 pts = 24 h)

ID tâche	Tâche	Type	Estimation
T-02.1	Développer l'API de recherche et de filtrage par disponibilité	Technique	8 h
T-02.2	Développer l'écran de liste et les filtres	Fonctionnel	10 h
T-02.3	Écrire les tests	Test	6 h

HU-03 - Réserver une salle (8 pts = 64 h)

ID tâche	Tâche	Type	Estimation
T-03.1	Concevoir le modèle de réservation et la règle anti-chevauchement	Technique	10 h

ID tâche	Tâche	Type	Estimation
T-03.2	Développer l'API de création (contrôle des conflits, transaction)	Technique	16 h
T-03.3	Développer l'écran de sélection de créneau et le formulaire	Fonctionnel	16 h
T-03.4	Gérer les horaires d'ouverture et le fuseau horaire	Fonctionnel	6 h
T-03.5	Écrire les tests unitaires, d'intégration et de concurrence	Test	10 h
T-03.6	Recette fonctionnelle et corrections	Test	6 h

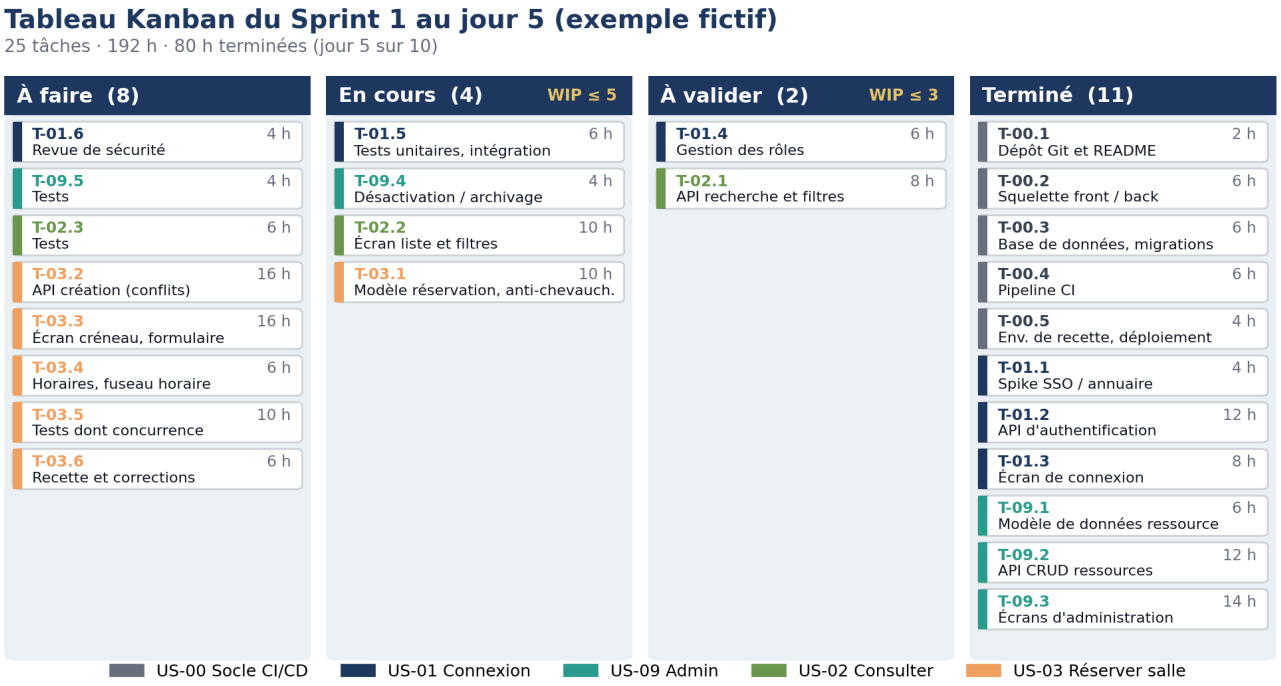
4. Phase 3 - Mise en place du suivi

4.1 Tableau Kanban

Colonnes et règles de gestion

Colonne	Règle d'entrée	Limite de travail en cours (WIP)
À faire	Tâche issue du Sprint Backlog, estimée et comprise	-
En cours	Un développeur prend la tâche et crée une branche Git	5
À valider	Code terminé, merge request ouverte, tests passés en CI	3
Terminé	Revue faite, Definition of Done respectée, mergée	-

Exemple d'état du tableau au jour 5 du Sprint 1 *(situation fictive illustrative)*



Règles d'usage : une tâche ne passe en « Terminé » qu'après validation par une personne autre que son auteur ; la limite WIP est respectée (on termine avant de commencer) ; le tableau est mis à jour avant chaque daily.

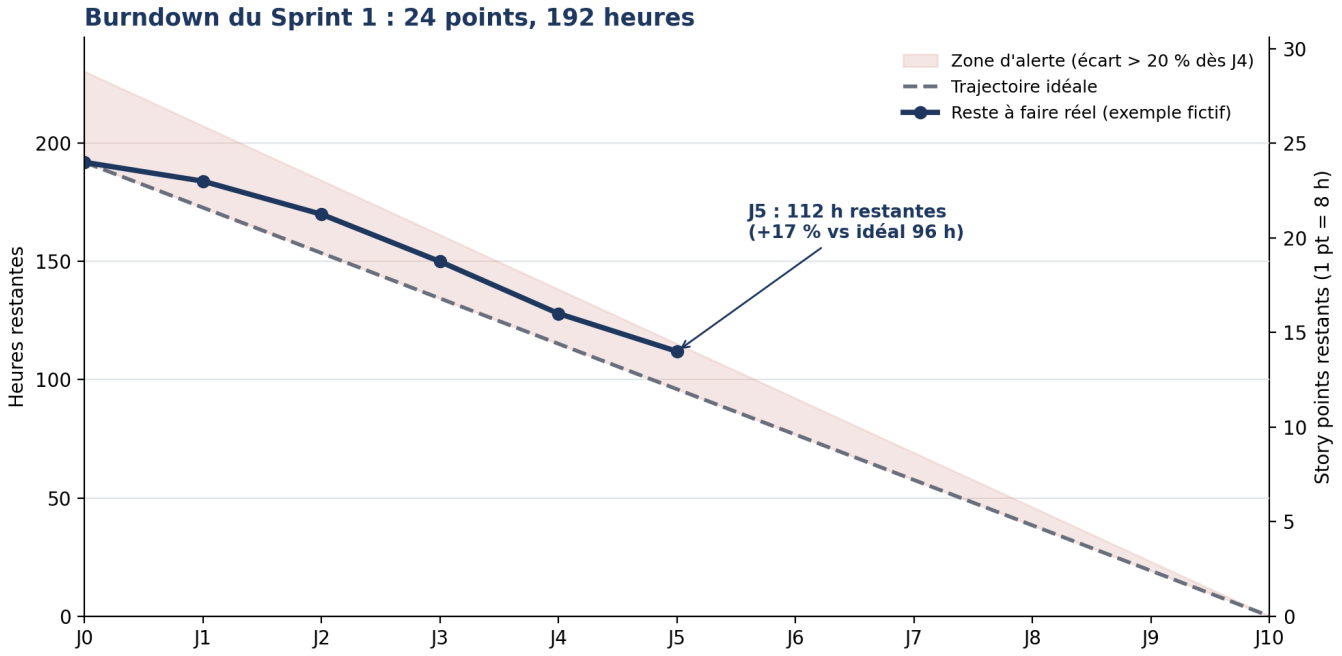
4.2 Trois indicateurs de suivi

#	Indicateur	Définition	Cible	Seuil d'alerte	Fréquence
1	**Burndown du sprint**	Story points restants par jour, comparés à la droite idéale	Suivre la droite idéale	Écart inférieur à 20 % à partir du jour 4	Quotidienne
2	**Vélocité et taux d'atteinte de l'engagement**	Points terminés (selon la DoD) divisés	supérieur ou égal 85 % de l'engagement	supérieur à 70 % sur un sprint	Fin de sprint

#	Indicateur	Définition	Cible	Seuil d'alerte	Fréquence
		par les points engagés			
3	**Cycle time moyen**	Nombre de jours entre « En cours » et « Terminé » pour une histoire	supérieur ou égal à 4 jours par histoire	Inférieur à 6 jours ou colonne « À valider » saturée	Hebdomadaire

Trajectoire idéale du burndown (Sprint 1, 24 points)

Jour	J0	J2	J4	J6	J8	J10
Points restants idéaux	24	19,2	14,4	9,6	4,8	0



4.3 Principaux risques initiaux

Rang	Risque	Criticité
1	Double réservation lors de réservations simultanées	15
2	Authentification d'entreprise plus complexe que prévu	12
3	Capacité surestimée	12
4	Faible disponibilité du PO et des utilisateurs du site pilote	12
5	Dérive du périmètre	9

5. Phase 4 - Qualité et livraison

5.1 Definition of Done (DoD)

Une histoire utilisateur est **terminée** uniquement si **tous** les points suivants sont vérifiés :

Développement

- ☐ Le code est écrit, relu par au moins un autre dev (MR approuvée) et fusionné sur la branche principale.
- ☐ Les standards de code sont respectés.
- ☐ Aucune dette technique critique n'est introduite sans ticket associé.

Tests

- ☐ Les tests unitaires et d'intégration sont écrits et passent en CI.
- ☐ La couverture de nouveau code est supérieure ou égale à 70 %.
- ☐ Tous les **Critères d'acceptation** sont vérifiés.
- ☐ Aucun bug bloquant ou majeur ouvert.

Livraison

- ☐ La fonctionnalité est déployée sur l'env `staging` par la CI/CD.
- ☐ La documentation est mise à jour (README, API, notes de version).
- ☐ L'histoire est démontrée et acceptée par le PO.

Sécurité et conformité

- ☐ Aucune faille critique détectée par l'analyse des dépendances.
- ☐ Les données manipulées respectent les règles RGPD.

5.2 Tests nécessaires avant validation

Type de test	Objectif	Responsable	Moment	Exemple pour FlexOffice
Tests unitaires	Vérifier la logique métier isolée	Développeurs	À chaque commit (CI)	Règle de détection de chevauchement de créneaux
Tests d'intégration	Vérifier l'interaction entre composants (API, base de données)	Développeurs	À chaque merge request (CI)	Création d'une réservation => enregistrement en base
Tests de concurrence	Garantir l'absence de double réservation	Développeurs / QA	Sprint 1 puis à chaque évolution	Deux utilisateurs réservent la même salle au même instant

Type de test	Objectif	Responsable	Moment	Exemple pour FlexOffice
Tests de bout en bout (E2E)	Valider les parcours complets	QA	Avant la fin de chaque sprint	Connexion => recherche => réservation => annulation
Tests de non-régression	Vérifier que l'existant fonctionne toujours	QA (automatisés en CI)	À chaque livraison	Rejouer les parcours clés après chaque histoire
Tests de sécurité	Détecter les failles courantes	Développeurs / QA	À chaque build + revue avant pilote	Contrôle des droits, injection, gestion de session
Tests de performance	Vérifier la tenue en charge du site pilote	QA	Sprint 3	100 utilisateurs simultanés, recherche en moins de 2 s
Tests de compatibilité / accessibilité	Garantir l'usage sur les navigateurs et écrans courants	QA	Sprint 2 et 3	Navigateurs principaux, affichage tablette
Recette utilisateur (UAT)	Valider que le produit répond au besoin réel	PO + utilisateurs du site pilote	Sprint 3 (semaine 7)	Scénarios métier réalisés par de vrais salariés

5.3 Utilisation de Git et d'une chaîne CI/CD

Stratégie Git

- **Dépôt central unique** hébergé sur une plateforme de gestion de code (GitLab, GitHub ou équivalent).
- **Modèle de branches simplifié** (trunk-based léger) :
  - `main` : branche stable, toujours déployable ;
  - `HU-xx` : une branche courte par histoire ou par tâche ;
  - `HU-xx-fix` : correctifs urgents sur la version pilote.
- **Conventions de commit** : `feat:`, `fix:`, `test:`, `docs:`, `refactor:` (par exemple `feat: contrôle des conflits de réservation`).
- **Merge Request obligatoire** : aucun commit direct sur `main` ; au moins une approbation d'un pair et une CI verte sont exigées.
- **Versionnement** : les versions livrées sont taguées (`v0.1.0` à la fin du Sprint 1, `v0.2.0` à la fin du Sprint 2, `v1.0.0-pilote` pour la mise en service).

Chaîne CI/CD

Étape	Déclencheur	Actions	Résultat attendu
1. Build	Chaque push	Installation des dépendances, compilation	Application construite sans erreur
2. Qualité	Chaque push	Lint, analyse statique du code	Aucune erreur bloquante

Étape	Déclencheur	Actions	Résultat attendu
3. Tests	Chaque push	Tests unitaires et d'intégration, mesure de couverture	Tous les tests passent, couverture supérieur ou égal à 70 %
4. Sécurité	Chaque merge request	Analyse des dépendances et du code	Aucune faille critique
5. Déploiement en recette	Merge sur `main`	Déploiement automatique, migrations de base, tests de fumée	Version disponible pour le PO et le QA
6. Tests E2E	Après déploiement en recette	Parcours automatisés	Parcours clés validés
7. Déploiement pilote	Création d'un tag de version, **validation manuelle** du PO	Déploiement sur l'environnement du site pilote, possibilité de retour arrière	Version en production pilote

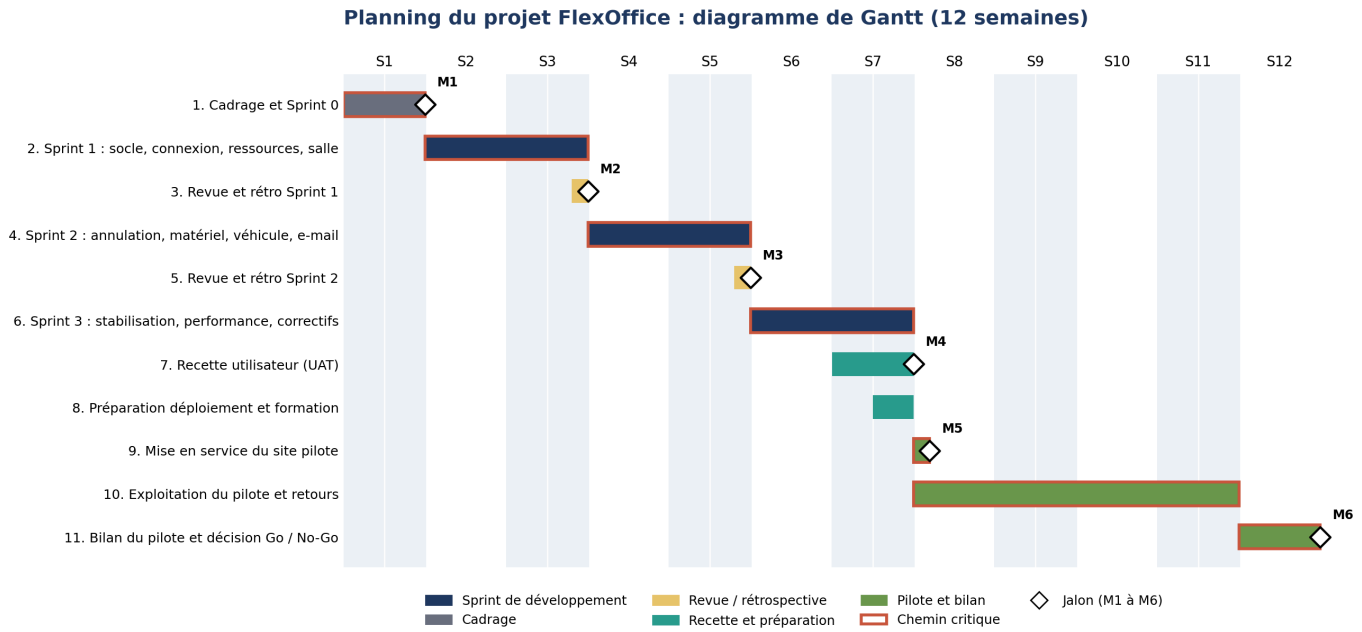
Bénéfices : retour d'information en quelques minutes pour les développeurs, livraisons reproductibles, réduction des erreurs humaines, incrément disponible en continu pour la démonstration.

5.4 Livrables de fin de sprint

#	Livrable	Responsable	Format
1	**Incrément logiciel** fonctionnel déployé en recette (version taguée)	Équipe de développement	Application + tag Git
2	**Sprint Backlog** à jour (stories terminées / non terminées)	PO / SM	Tableau Kanban
3	**Product Backlog** raffiné et repriorisé	PO	Backlog
4	**Notes de version** (fonctionnalités, correctifs, limites connues)	Équipe	Document
5	**Rapport de tests** (résultats, couverture, anomalies ouvertes)	QA	Rapport
6	**Indicateurs du sprint** (burndown, vélocité, cycle time)	SM	Tableau de bord
7	**Compte rendu de la Sprint Review** (démo, retours des parties prenantes)	PO / SM	Compte rendu
8	**Compte rendu de la rétrospective** et plan d'actions d'amélioration	SM	Compte rendu
9	**Documentation technique** à jour (installation, API, architecture)	Développeurs	README / wiki
10	**Registre des risques** mis à jour	SM	Registre

6. Planning et diagramme de Gantt

6.1 Diagramme de Gantt



6.2 Données de planification

Les semaines sont numérotées à partir du démarrage du projet (S1 = première semaine).

ID	Activité	Début	Fin	Durée	Prédécesseur
1	Cadrage et Sprint 0 (backlog, architecture, outils)	S1	S1	1 sem.	-
2	**Sprint 1** : socle, connexion, ressources, réservation de salle	S2	S3	2 sem.	1
3	Revue et rétrospective du Sprint 1	S3	S3	1 j	2
4	**Sprint 2** : annulation, matériel, véhicule, courriel, historique	S4	S5	2 sem.	2
5	Revue et rétrospective du Sprint 2	S5	S5	1 j	4
6	**Sprint 3** : stabilisation, tests de performance, correctifs	S6	S7	2 sem.	4
7	Recette utilisateur (UAT)	S7	S7	1 sem.	4
8	Préparation du déploiement et formation du site pilote	S7	S8	1,5 sem.	6
9	Mise en service du site pilote	S8	S8	1 j	7, 8
10	Exploitation du pilote et suivi des retours	S8	S11	4 sem.	9
11	Bilan du pilote et décision Go / No-Go	S12	S12	1 sem.	10

Jalons

Jalon	Semaine	Prédécesseur
M1 - Backlog prêt, socle défini	S1	Activité 1

Jalon	Semaine	Prédécesseur
M2 - Démo Sprint 1	S3	Activité 3
M3 - Version pilote candidate	S5	Activité 5
M4 - Version validée en recette	S7	Activité 7
M5 - Mise en service du pilote	S8	Activité 9
M6 - Bilan et décision	S12	Activité 11

Chemin critique : Cadrage => Sprint 1 => Sprint 2 => Sprint 3 / UAT => Mise en service => Exploitation du pilote => Bilan.

Cérémonies par sprint (2 semaines)

Cérémonie	Durée	Moment
Sprint Planning	4 h	Jour 1
Daily Scrum	15 min	Chaque jour
Raffinement du backlog	2 h	Milieu de sprint
Sprint Review (démo)	2 h	Jour 10
Rétrospective	1 h 30	Jour 10

7. Estimation des ressources et des délais

7.1 Ressources humaines

Les charges sont exprimées en **jours-personnes (j-p)**.

Rôle	Effectif	Allocation	Cadrage (S1)	Sprint 1	Sprint 2	Sprint 3	Pilote (S8-S11)	Total
Product Owner	1	50 %	3	5	5	5	5	**23**
Scrum Master	1	50 %	3	5	5	5	2	**20**
Développeurs	3	100 %	9	30	30	30	12	**111**
Testeur / QA	1	50 %	1	5	5	5	3	**19**
Total	**6**		**16**	**45**	**45**	**45**	**22**	**173**

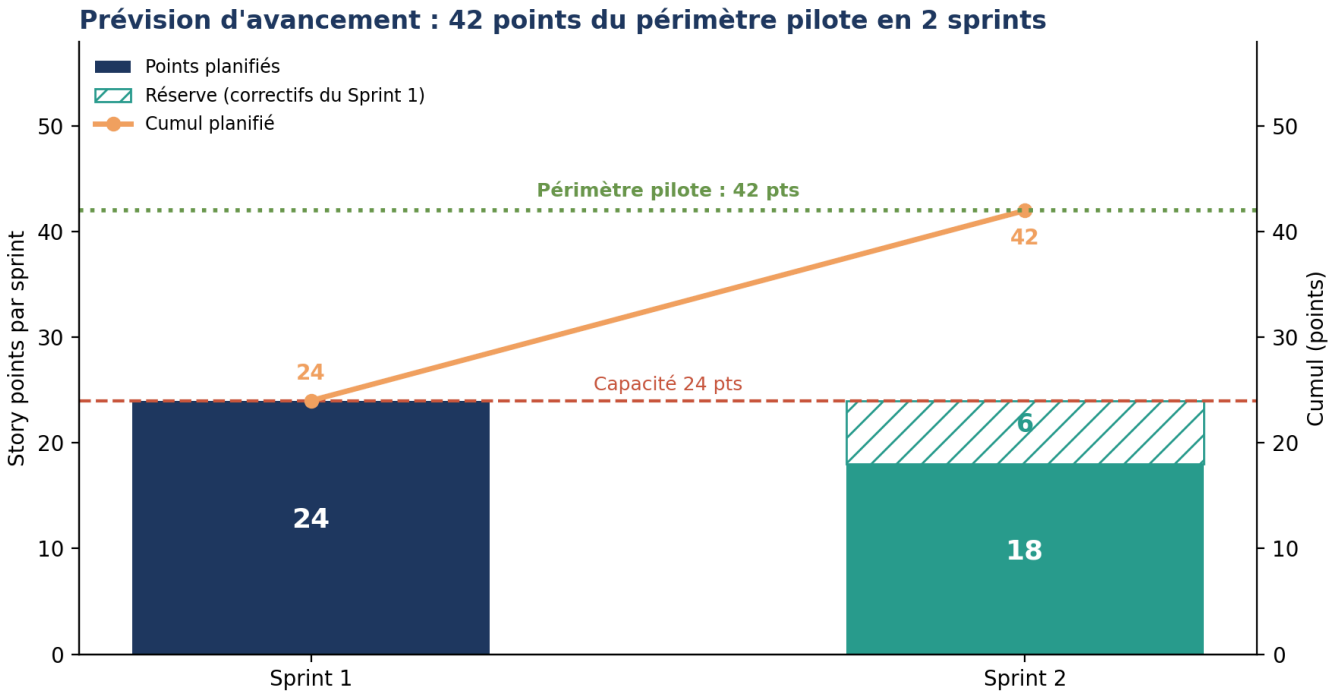
Pendant le pilote, les développeurs sont mobilisés à environ 20 % pour le support et les correctifs.

7.2 Répartition de l'effort de développement par histoire (Sprint 1)

Story	Points	Heures	Jours-personnes
HU-00 Socle technique et CI/CD	3	24	3
HU-01 Connexion utilisateur	5	40	5
HU-09 Administration des ressources	5	40	5
HU-02 Consulter les ressources	3	24	3
HU-03 Réserver une salle	8	64	8
Total	**24**	**192**	**24**

7.3 Prévision d'avancement par sprint

Sprint	Points planifiés	Vélocité cumulée prévue	Reste du périmètre pilote (42 pts)
Sprint 1	24	24	18
Sprint 2	18 *(+ 6 de réserve)*	42	0
Sprint 3	12 de stabilisation + 12 selon retours	-	-



7.4 Ressources techniques et matérielles

Ressource	Usage	Disponibilité requise
Dépôt Git et plateforme CI/CD	Versionnement, intégration et déploiement continus	Sprint 0
Environnement de développement (local)	Développement individuel	Sprint 0
Environnement de recette	Démonstrations, tests QA et UAT	Fin du Sprint 1
Environnement du site pilote (production)	Utilisation réelle	Semaine 7
Base de données	Stockage des ressources et réservations	Sprint 0
Accès à l'annuaire / SSO d'entreprise	Authentification (HU-01)	Début du Sprint 1
Service d'envoi d'courriels (SMTP)	Confirmations (HU-07)	Début du Sprint 2
Outil de suivi (tableau Kanban)	Gestion du sprint	Sprint 0
Données initiales des ressources du site pilote	Alimentation du catalogue	Fin du Sprint 1

7.5 Synthèse des délais

Étape	Durée	Cumul
Cadrage (Sprint 0)	1 semaine	S1
Développement (3 sprints)	6 semaines	S7

Étape	Durée	Cumul
Déploiement et formation	1 semaine	S8
Pilote	4 semaines	S11
Bilan	1 semaine	S12

Délai pour disposer d'une première version utilisable sur le site pilote : 8 semaines. Une première démo (réservation de salle) est disponible dès la fin de la **semaine 3**.

Marges de sécurité : réserve de 6 points au Sprint 2 et 12 points de capacité libre au Sprint 3, qui absorbent les aléas sans remettre en cause la date de mise en service.

8. Plan de gestion des risques

8.1 Méthode d'évaluation

- **Probabilité (P)** : 1 = très improbable, 3 = possible, 5 = quasi certain.
- **Impact (I)** : 1 = négligeable, 3 = modéré, 5 = bloquant pour le pilote.
- **Criticité = P × I** : supérieure ou égale à 12 = **élevée**, 6 à 11 = **moyenne**, supérieur ou égal à 5 = **faible**.
- **Stratégies :** Éviter · Réduire · Transférer · Accepter.

8.2 Registre des risques

ID	Risque	P	I	Criticité	Stratégie	Actions d'atténuation	Plan de secours (si le risque survient)	Responsable
R1	**Double réservation** lors de réservations simultanées	3	5	**15**	Éviter	Verrou ou contrainte d'unicité en base ; tests de concurrence automatisés dès le Sprint 1	Verrouiller temporairement la création côté serveur ; correctif prioritaire	Développeurs
R2	**Authentification d'entreprise** plus complexe que prévu	3	4	**12**	Réduire	Spike dès le début du Sprint 1 ; contact avec l'équipe SI identifié en amont	Authentification locale temporaire avec comptes importés	Tech lead
R3	**Capacité surestimée** (pas de vélocité historique)	4	3	**12**	Réduire	Marge incluse dans la capacité ; recalibrage après le Sprint 1 ; réserve de 6 pts au Sprint 2	Retirer les stories Should du périmètre pilote	Scrum Master
R4	**Faible disponibilité du PO** et des utilisateurs du site pilote	3	4	**12**	Réduire	Créneaux de validation réservés à l'agenda ; désignation d'un suppléant du PO ; utilisateurs relais sur site	Décisions déléguées au suppléant ; démos enregistrées	Product Owner

ID	Risque	P	I	Criticité	Stratégie	Actions d'atténuation	Plan de secours (si le risque survient)	Responsable
R5	**Dérive du périmètre** (ajout de fonctionnalités Could)	3	3	**9**	Éviter	Périmètre pilote figé ; toute nouvelle demande passe par le backlog et la priorisation du PO	Report systématique au backlog post-pilote	Product Owner
R6	**Données de ressources** incomplètes ou inexactes	3	3	**9**	Réduire	Modèle de données d'import fourni au site pilote ; responsable des données désigné ; contrôle de qualité avant la mise en service	Saisie manuelle assistée par l'équipe	Product Owner
R7	**Faible adoption** par les salariés du site pilote	2	4	**8**	Réduire	Communication et formation avant la mise en service ; interface simple ; ambassadeurs sur site	Sessions de support supplémentaires, ajustement de l'ergonomie	Product Owner
R8	**Sécurité et conformité RGPD** (données personnelles des salariés)	2	4	**8**	Réduire	Minimisation des données ; contrôle des accès par rôle ; analyse automatique des failles ; validation du DPO	Correctif de sécurité prioritaire ; suspension du service si nécessaire	Tech lead
R9	**Dette technique** et baisse de la qualité sous pression	2	3	**6**	Réduire	Definition of Done stricte ; revues de code ; temps de refactoring dans chaque sprint	Sprint dédié à la dette avant l'extension	Équipe

ID	Risque	P	I	Criticité	Stratégie	Actions d'atténuation	Plan de secours (si le risque survient)	Responsable
R10	**Absence** d'un membre clé	2	3	**6**	Réduire	Polyvalence des développeurs ; revues croisées ; documentation à jour	Réaffectation des tâches, réduction de l'engagement	Scrum Master
R11	**Indisponibilité du service d'courriel**	2	2	**4**	Accepter	Envoi asynchrone avec reprise ; échec journalisé sans bloquer la réservation	Notification dans l'application en attendant	Développeurs

8.3 Matrice de criticité (probabilité × impact)

Probabilité / Impact =>	1	2	3	4	5
5					
4			R3		
3			R5, R6	R2, R4	R1
2		R11	R9, R10	R7, R8	
1					

8.4 Suivi des risques

- Le registre est **revu à chaque Sprint Planning** et lors de chaque rétro par le Scrum Master.
- Les risques de criticité supérieurs ou égaux à 12 sont **suivis chaque semaine** ; le dépassement d'un seuil d'alerte déclenche immédiatement le plan de secours.
- Tout nouveau risque identifié est ajouté au registre, évalué (P × I) et rattaché à un responsable.
- Un risque clos ou sans objet est archivé avec date et raison.

9. Hypothèses retenues

Le cas ne précisant pas certaines données, les hypothèses suivantes ont été retenues. Elles devront être confirmées avec les parties prenantes.

#	Hypothèse
H1	Sprints de **2 semaines** (10 jours ouvrés).
H2	Équipe de 6 personnes : PO (50 %), Scrum Master (50 %), 3 développeurs (100 %), 1 testeur (50 %).
H3	**1 point d'histoire ≈ 1 jour-personne idéal** ; ce ratio sera recalibré après le Sprint 1.
H4	L'entreprise dispose d'un annuaire d'entreprise (type SSO) exploitable pour la connexion.
H5	Le pilote concerne **un seul site**, avec quelques centaines de salariés au plus.
H6	Le pilote dure **4 semaines** avant le bilan et la décision de généralisation.
H7	Aucune vélocité historique n'existe : la capacité du Sprint 1 est estimée à partir de la disponibilité.
H8	Les stories HU-11 (agenda externe) et HU-12 (notifications mobiles) sont exclues de la version pilote.
H9	Les environnements (recette, pilote) et le service d'courriel sont fournis par le service informatique de l'entreprise.