

# Sprint 4 - État, Objectifs et Répartition des tâches

## Bilan depuis Sprint 4 (réalisations)

- **UI complète et fonctionnelle** : MainMenu, GameView, LeaderboardView, SettingsView.
  - **Logique de jeu avancée** : modes Solo/Duel, timer, scoring, indices, transitions entre manches.
  - **AudioService** : lecture des extraits via API iTunes avec fallback vers playlist locale, gestion du volume.
  - **GameController** : génération automatique de playlists par genre, gestion des erreurs réseau.
  - **Leaderboard** : affichage des scores, tri et formatage, export CSV/JSON partiel.
  - **Validation des inputs utilisateur** renforcée (normalisation, trim, suppression caractères spéciaux).
  - **Tests unitaires et manuels** réalisés pour AudioService, GameController, ScoreService, PersistenceService, Player, Track.
  - **Mode Duel** : implémentation alternée des tours, scores séparés, affichage du gagnant.
- 

## Répartition des points (réalisations depuis Sprint 4)

- **Iskander — 36 points**
    - Intégration audio complète (API iTunes, fallback, gestion volume).
    - Sélection de genres.
    - Tests unitaires AudioService.
  - **Léo — 32 points**
    - Persistance complète : scores, paramètres, playlists.
    - Leaderboard : filtres, statistiques détaillées.
    - Tests PersistenceService et ScoreService.
  - **Achraf — 32 points**
    - UI complète : MainMenu, GameView, LeaderboardView, SettingsView.
    - Logique de jeu : modes Solo/Duel, timer, scoring, indices.
    - Tests manuels de l'UI.
- 

## Objectifs du Sprint 5

- Finaliser l'application pour la présentation finale dans 3 semaines.
- Audio : récupérer tous les extraits directement en ligne depuis l'API ; fallback complet si nécessaire.
- UI : polir animations, transitions, feedback visuel et sonore, jauge bonus de rapidité dynamique.
- Vérifier le fonctionnement complet de toutes les fonctionnalités : Solo/Duel, scoring, indices, leaderboard, paramètres.

- README : compléter avec instructions d'installation et d'utilisation.
  - Mettre à jour les diagrammes UML et le diagramme Gantt.
  - Vérifier la couverture des tests unitaires et manuels (>80%).
- 

## Répartition des tâches par personne (Sprint 5)

### Iskander

- Intégrer complètement les extraits de musique en ligne via l'API.
- Gérer le volume et les erreurs réseau dans AudioService.
- Implémenter le fallback complet vers playlists locales si l'API échoue.
- Finaliser les tests unitaires AudioService et vérifier la robustesse.
- Optimiser le cache pour les requêtes API fréquentes.

### Léo

- Compléter la persistance des scores, paramètres et playlists.
- Finaliser le leaderboard : filtres par mode/genre, statistiques détaillées, export CSV/JSON.
- Vérifier et compléter les playlists locales si l'API échoue.
- Tests unitaires PersistenceService et ScoreService.
- Vérification finale de la sauvegarde et du chargement des données.

### Achraf

- Finaliser l'UI : animations, transitions entre vues, jauge de rapidité dynamique.
  - Vérifier et polir les fonctionnalités Solo/Duel : tours alternés, scoring, indices.
  - Implémenter les feedbacks visuels et sonores pour les réponses et les bonus.
  - Tests manuels complets de l'UI et vérification des écrans récapitulatifs.
  - Finaliser les écrans de transition entre manches avec récapitulatif des scores.
- 

## Réalisations prévues / tâches principales

- Mode Solo : finalisation scoring, bonus de rapidité, transitions, affichage récapitulatif.
- Mode Duel : tours alternés, musiques distinctes, scores séparés, affichage du gagnant.
- Leaderboard : filtres par mode, statistiques détaillées, export CSV/JSON complet.
- AudioService : récupération en ligne des extraits, fallback complet vers playlists locales.
- UI : animations de transition entre vues, jauge bonus de rapidité dynamique, écrans récapitulatifs, feedback visuel et sonore.
- Validation des inputs utilisateur : robustesse maximale.
- Tests unitaires et manuels : couverture étendue et vérification de toutes les interactions.

---

## Tests

- Unitaires pour AudioService, GameController, ScoreService, PersistenceService, Player, Track.
  - Manuels : validation complète de l'UI et interactions utilisateurs.
  - Vérification du fallback audio et des erreurs réseau.
- 

## Documentation

- README finalisé avec instructions complètes.
  - Diagrammes UML mis à jour (classes et architecture).
  - Diagramme Gantt mis à jour.
- 

## Livrable attendu

- Application BlindTest complète et prête pour la présentation finale.
- UI et UX finalisées avec animations et feedback.
- Fonctionnalités Solo/Duel, scoring, indices, leaderboard opérationnelles.
- Audio fonctionnel en ligne ou playlists locales complètes.
- Documentation complète (README et diagrammes UML/Gantt).
- Couverture de tests étendue et vérifiée.