

Cadrage UX — Système de vote musical en salle de cours

Groupe : Bastien & Ikram
MyDigitalSchool



Projet: Design Music queuing

Date: 29/09/2025

Contexte du projet

Ce projet s'inscrit dans le cadre du développement d'une plateforme de vote musical utilisée dans des salles de cours universitaires. L'objectif est de permettre aux étudiants de :

- proposer des morceaux musicaux pour une session donnée,
- voter une seule fois par jour pour leur morceau préféré,
- consulter les résultats en quasi temps réel.

Le backend repose sur une API REST Node.js/Prisma intégrant :

- le scraping automatique des emplois du temps,
- une authentification simple par e-mail,
- la soumission et le vote uniques,
- l'affichage en temps réel des résultats.

Notre mission est de concevoir l'expérience front (UX/UI) qui permettra d'exploiter cette API de façon accessible, fluide et équitable.

Problématique

Les outils classiques (tableurs, sondages en ligne, partages informels) ne permettent pas :

- de limiter efficacement les votes à un par jour et par utilisateur,
- d'assurer une accessibilité conforme WCAG AA,
- de neutraliser les biais cognitifs (ancrage, preuve sociale, etc.),
- de proposer une expérience fluide sur mobile dans un temps limité (avant le début du cours).

Objectifs UX

- Offrir une expérience simple et rapide : soumettre ou voter en moins de 10 secondes.
- Garantir un vote unique par jour avec feedback explicite.
- Afficher les résultats en temps réel de manière neutre et compréhensible.
- Concevoir une interface accessible WCAG AA : contrastes, navigation clavier, aria-live.
- Proposer un parcours utilisateur robuste aux biais cognitifs et sans dark patterns.

Personae

Persona 1 : Leila , 20 ans — Étudiante pressée (mobile-first)

Contexte : étudiante en B3 Dev, souvent en retard le matin, se connecte sur son téléphone dans les couloirs.

Besoins :

- Accéder rapidement à la plateforme sans étapes inutiles,
- Voter en 1 à 2 clics,
- Avoir des boutons visibles et clairs.

Douleurs :

- Petites cibles tactiles,
- Messages peu lisibles,
- Temps de chargement non expliqué.

Accessibilité :

- Préfère des boutons larges, focus visible, contraste élevé.

Persona 2 : Lucas , 23 ans — Étudiant mélomane (desktop/tablette)

Contexte : très impliqué dans la vie étudiante, aime proposer des titres et suivre les résultats pendant le cours.

Besoins :

- Formulaire de soumission simple,
- Feedback clair après l'ajout,
- Classement fiable et lisible.

Douleurs :

- Classements biaisés,
- Erreurs API sans explication,
- Interfaces mal structurées.

Accessibilité :

- Navigation clavier complète, titres hiérarchisés, aria-live sur le classement.

Scénarios, User flows & Hypothèses

Scénarios utilisateurs :

1- Soumission d'un morceau

- Connexion
- Sélection de la session
- Formulaire "Artiste + Titre"
- Validation
- Confirmation visuelle ("Morceau ajouté !")

États prévus : vide, envoi, succès, erreur, validation manquante.

2- Vote (1 fois par jour)

- Accès à la liste
- Tri neutre par défaut (aléatoire)
- Sélection d'un morceau
- Feedback positif
- Bouton désactivé + message "Vous avez déjà voté aujourd'hui"

États prévus : normal, désactivé (quota), loading, erreur réseau.

3- Consultation des résultats

1. Accès à la page Résultats
2. Classement sobre (barres + labels)
3. Mise à jour douce en arrière-plan via polling/socket
4. Aria-live polite pour annonces non intrusives.

États prévus : vide, chargement (squelettes), erreur, données partielles.

Hypothèses UX :

Hypothèse	Impact attendu	Mesure
Tri neutre au chargement	Réduction biais d'ancrage	Dispersion votes premiers items
Bouton désactivé + tooltip	Diminution tentatives multiples	Nombre tentatives répétées
Feedback clair après soumission	Meilleure adoption	% réitération soumission
Focus visible & tab logique	Meilleure inclusion clavier	Temps moyen vote clavier
Aria-live sur résultats	Meilleure compréhension dynamique	% utilisateurs satisfaits

Conclusion

Ce cadrage UX pose les fondations d'une interface :

- mobile-first, rapide et accessible,
- conçue pour limiter les biais cognitifs,
- robuste face aux différents états d'usage,
- et parfaitement alignée avec les endpoints de l'API backend.

Prochaine étape : Design tokens et Maquettes Figma.