Workspace Seat Booking System

A concise technical review of a React + Node.js solution for reserving desks, preventing conflicts, and enforcing attendance policies. Designed for technical interviewers and engineering peers.

Made with GAMMA

Problem Statement

Manual processes

People reserve seats via email/spreadsheet → error-prone and slow.

Double bookings

Simultaneous requests create conflicts and wasted space.

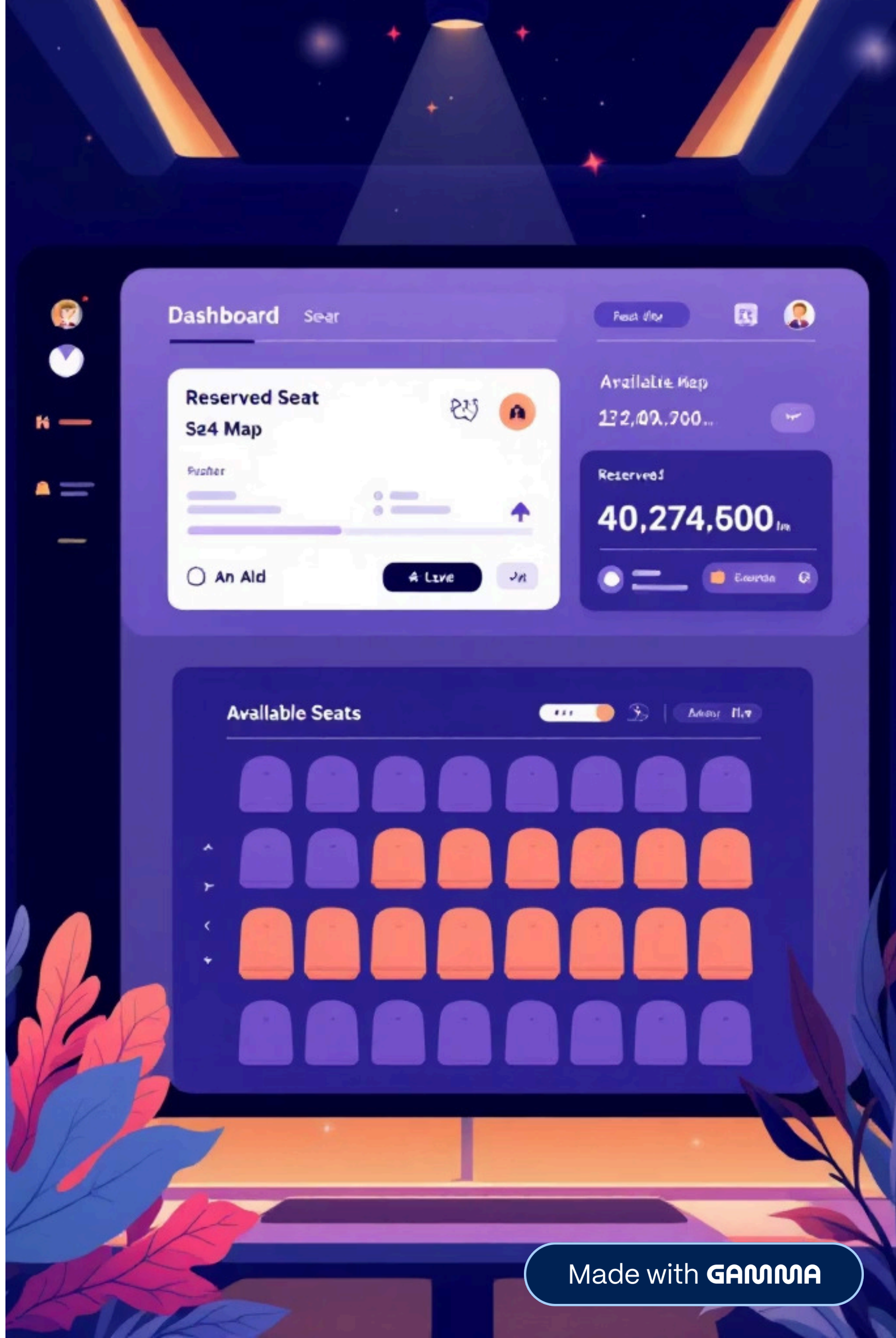
No attendance enforcement

Seats left unused due to no-shows, reducing fairness.

Made with GAMMA

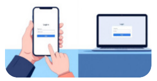
Solution Overview

Web-based booking app with batch allocation, temporary seat locks to prevent races, and eligibility rules enforcing fair use. Lightweight REST API backend and responsive React frontend.



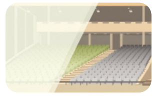
Made with GAMMA

Key Features



User Authentication

Secure sign-in, role-based access, single account per employee.



Seat Availability

Real-time availability state with clear visual indicators.

Locking Mechanism

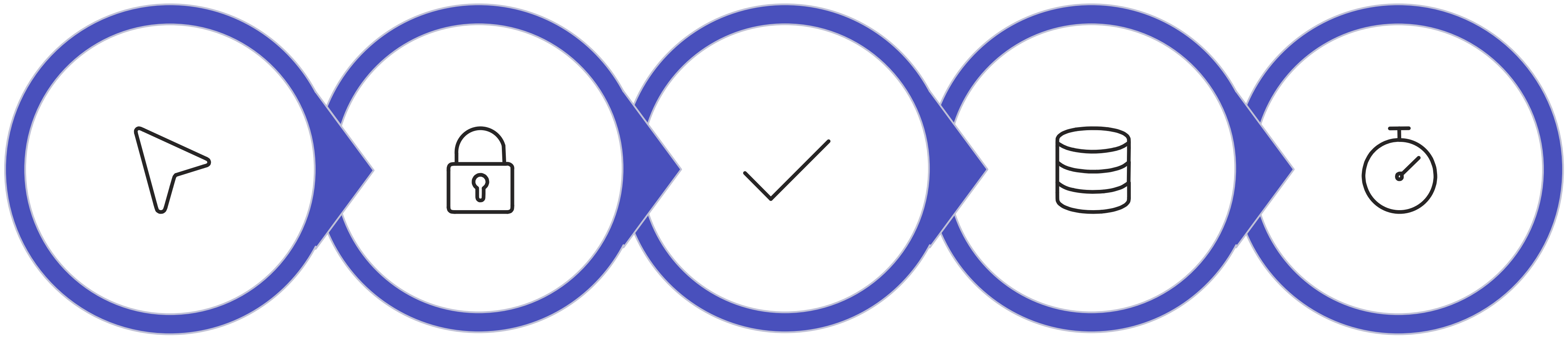
Short-lived locks prevent race conditions during selection.

Attendance Policies

Enforce eligibility and reclaim unused bookings automatically.

Made with GAMMA

Seat Locking Mechanism



Request Lock

Lock Held

Confirm Booking

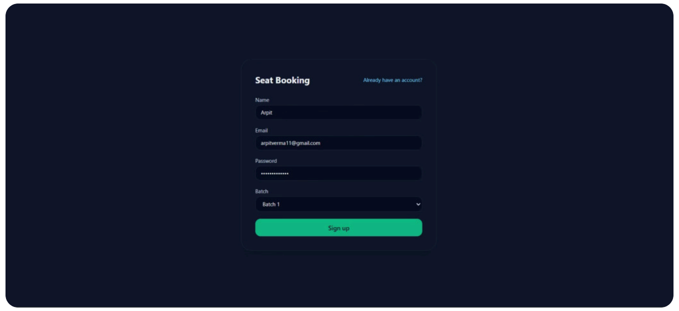
Persist Reservation

Lock Expires

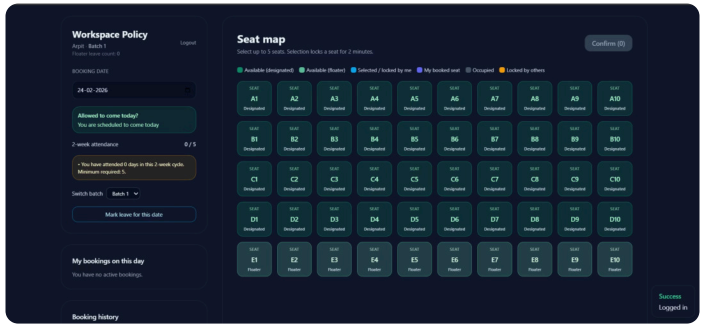
The frontend requests a lock via the API. Backend creates a TTL lock record in the database and returns a lock token. The client confirms within the TTL to convert the lock into a booked reservation. Expired locks are garbage-collected to restore availability.

Made with GAMMA

User Workflow



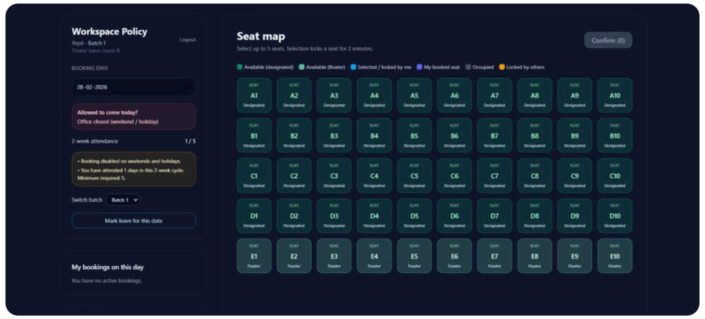
Name: Arpit
Email: arpitverma11@gmail.com
Password: ••••••••••••••
Batch: Batch 1



Arpit · Batch 1
Floater leave count: 0
BOOKING DATE: 24-02-2026
Allowed to come today?
You are scheduled to come today
2-week attendance 0 / 5
• You have attended 0 days in this 2-week cycle. Minimum required: 5.
Switch batch: Batch 1
My bookings on this day
You have no active bookings.
Booking history

Select up to 5 seats. Selection locks a seat for 2 minutes.
Available (designated) · Available (floater) · Selected / locked by me · My booked seat · Occupied · Locked by others

SEAT A1 Designated	SEAT A2 Designated	SEAT A3 Designated	SEAT A4 Designated	SEAT A5 Designated	SEAT A6 Designated	SEAT A7 Designated	SEAT A8 Designated	SEAT A9 Designated	SEAT A10 Designated
SEAT B1 Designated	SEAT B2 Designated	SEAT B3 Designated	SEAT B4 Designated	SEAT B5 Designated	SEAT B6 Designated	SEAT B7 Designated	SEAT B8 Designated	SEAT B9 Designated	SEAT B10 Designated
SEAT C1 Designated	SEAT C2 Designated	SEAT C3 Designated	SEAT C4 Designated	SEAT C5 Designated	SEAT C6 Designated	SEAT C7 Designated	SEAT C8 Designated	SEAT C9 Designated	SEAT C10 Designated
SEAT D1 Designated	SEAT D2 Designated	SEAT D3 Designated	SEAT D4 Designated	SEAT D5 Designated	SEAT D6 Designated	SEAT D7 Designated	SEAT D8 Designated	SEAT D9 Designated	SEAT D10 Designated
SEAT E1 Floater	SEAT E2 Floater	SEAT E3 Floater	SEAT E4 Floater	SEAT E5 Floater	SEAT E6 Floater	SEAT E7 Floater	SEAT E8 Floater	SEAT E9 Floater	SEAT E10 Floater



Arpit · Batch 1
Floater leave count: 0
BOOKING DATE: 28-02-2026
Allowed to come today?
Office closed (weekend / holiday)
2-week attendance 1 / 5
• Booking disabled on weekends and holidays.
• You have attended 1 days in this 2-week cycle. Minimum required: 5.
Switch batch: Batch 1
My bookings on this day
You have no active bookings.

Select up to 5 seats. Selection locks a seat for 2 minutes.
Available (designated) · Available (floater) · Selected / locked by me · My booked seat · Occupied · Locked by others

SEAT A1 Designated	SEAT A2 Designated	SEAT A3 Designated	SEAT A4 Designated	SEAT A5 Designated	SEAT A6 Designated	SEAT A7 Designated	SEAT A8 Designated	SEAT A9 Designated	SEAT A10 Designated
SEAT B1 Designated	SEAT B2 Designated	SEAT B3 Designated	SEAT B4 Designated	SEAT B5 Designated	SEAT B6 Designated	SEAT B7 Designated	SEAT B8 Designated	SEAT B9 Designated	SEAT B10 Designated
SEAT C1 Designated	SEAT C2 Designated	SEAT C3 Designated	SEAT C4 Designated	SEAT C5 Designated	SEAT C6 Designated	SEAT C7 Designated	SEAT C8 Designated	SEAT C9 Designated	SEAT C10 Designated
SEAT D1 Designated	SEAT D2 Designated	SEAT D3 Designated	SEAT D4 Designated	SEAT D5 Designated	SEAT D6 Designated	SEAT D7 Designated	SEAT D8 Designated	SEAT D9 Designated	SEAT D10 Designated
SEAT E1 Floater	SEAT E2 Floater	SEAT E3 Floater	SEAT E4 Floater	SEAT E5 Floater	SEAT E6 Floater	SEAT E7 Floater	SEAT E8 Floater	SEAT E9 Floater	SEAT E10 Floater

01

1. Signup

Create account, verify employee eligibility (policy checks).

02

2. Select

Pick a date and seat; UI requests a lock from backend.

03

3. Lock

Seat locked for short TTL to allow confirmation.

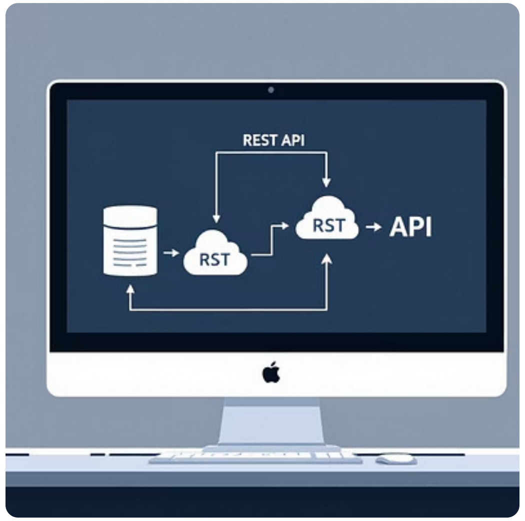
04

4. Confirm

Client confirms; backend persists booking and releases the lock token.

Made with GAMMA

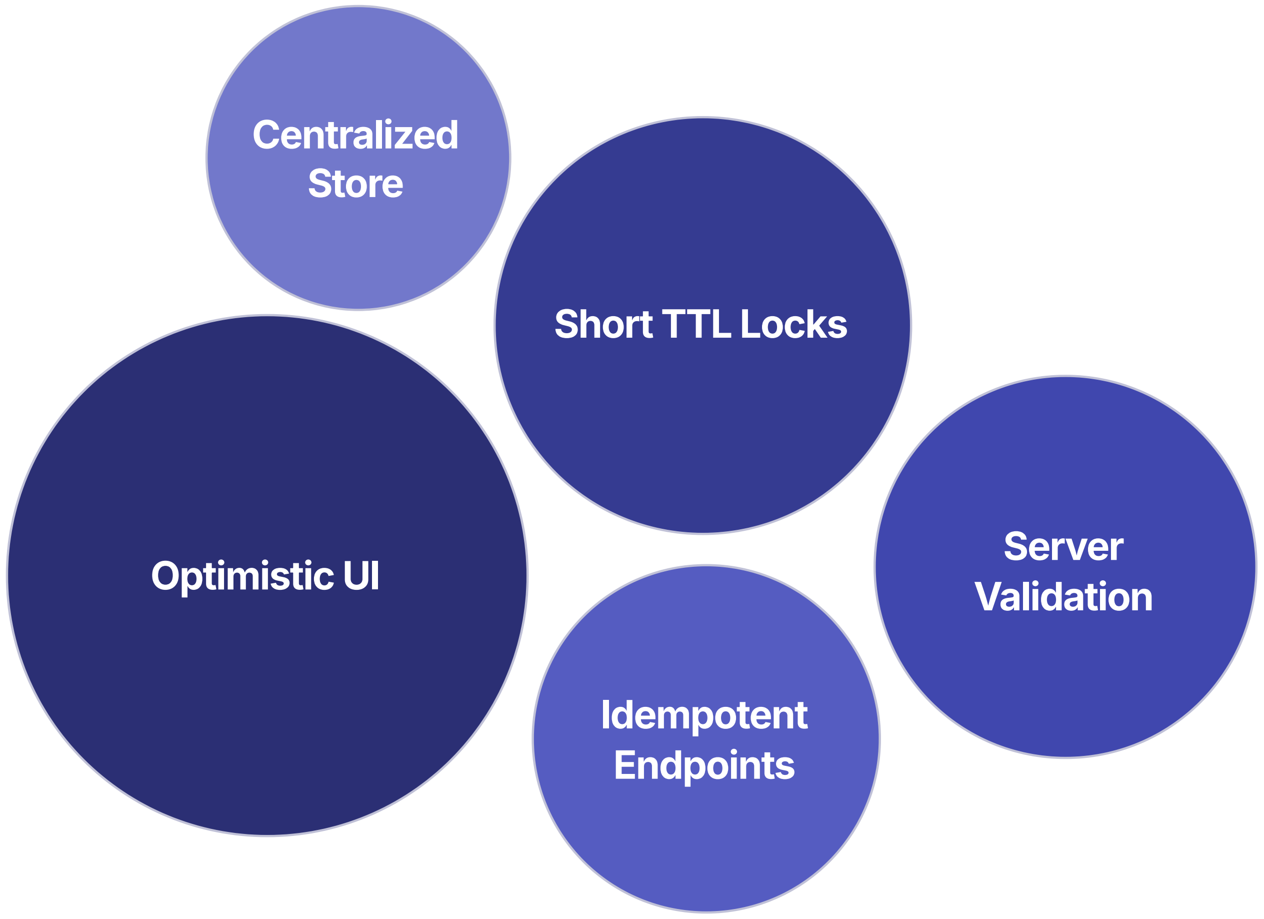
Technical Stack



Frontend: React (hooks, context, optimistic UI). Backend: Node.js + Express. API: REST endpoints for locks, bookings, and policies. DB: transactional store (Postgres) with advisory locks.

Made with GAMMA

System Design Thinking



1 **Fairness**

Batch allocation & eligibility rules to prioritize access.

2 **Consistency**

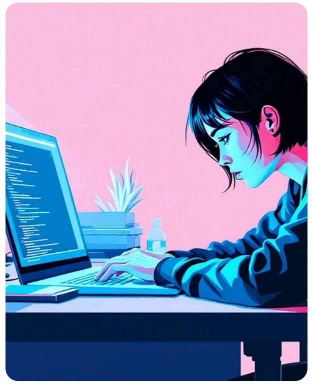
Authoritative server validation prevents stale client state.

3 **Scalability**

Stateless API servers + central DB for locks support horizontal scaling.

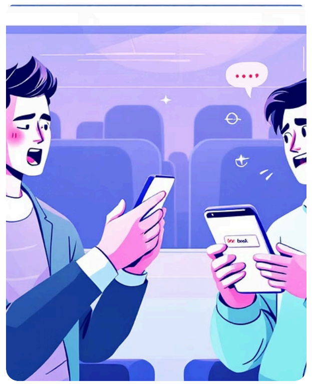
Made with GAMMA

Challenges Faced



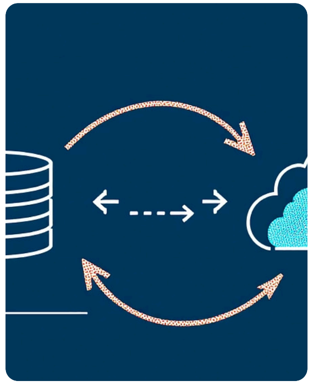
Managing Seat State

Ensuring availability status is accurate across many clients and rapid changes.



Preventing Conflicts

Race conditions required tight lock TTLs, retry strategies, and server-side checks.

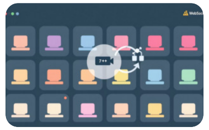


Synchronization

Balancing speed with correctness (DB transactions) and eventual consistency.

Made with GAMMA

Future Improvements & Conclusion



Real-time UX

Upgrade to WebSockets for push updates and zero-latency seat changes.

Admin Analytics

Add reports for utilization, no-shows, and policy compliance to optimize capacity.

Automation

Auto-release unused bookings, smart reallocation, and adaptive TTLs for fairness.

Impact: reduced double bookings, fairer seat distribution, and a scalable foundation. Key learnings: design for concurrency, validate on the server, and make UX forgiving. Ready for production-grade scaling.

Made with GAMMA