Building the I.S. Oral Defense Scheduler: A Full-Stack Web Application using Next.js, Supabase, and Vercel

Pem Tsering Gurung
Department of Mathematical and Computational Sciences
Advised by Dr. Sofia Visa

The Problem

- Every CS senior gives an oral defense.
- Scheduling a single oral defense requires a one-hour availability for: (1) a room; (2) student; (3) First Reader; (4) Secondary Reader.
- Laborious manual scheduling relies on scattered email chains, prone to deadlocks, double-bookings and frustration.

The Solution

- A centralized, custom-built web application. Instead of relying on humans to cross-reference emails, the system acts as a single source of truth.
- It offloads the cognitive burden onto a relational PostgreSQL database, automatically enforcing department scheduling rules to guarantee mathematically conflict-free bookings.

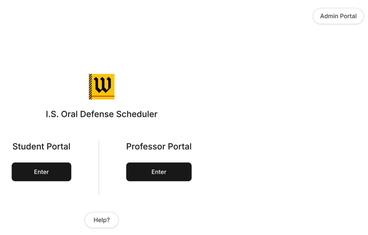

Figure 1. I.S. Oral Defense Scheduler

The Triple-Constraint Algorithm

To determine bookable slots, the backend evaluates every 60-minute interval against three simultaneous database constraints:

- **Faculty Availability:** Both selected readers must explicitly mark themselves as available.
- **No Double-Booking:** Neither professor can be assigned to another defense during that specific hour.
- **Room Availability:** At least one physical room must be open and unoccupied.

System Architecture

- **Built on Next.js 15 and Tailwind CSS, deployed on Vercel's serverless Infrastructure.**
- Utilizes a **PostgreSQL database** through **Supabase** to enforce strict referential integrit and Row-Level Security (RLS).

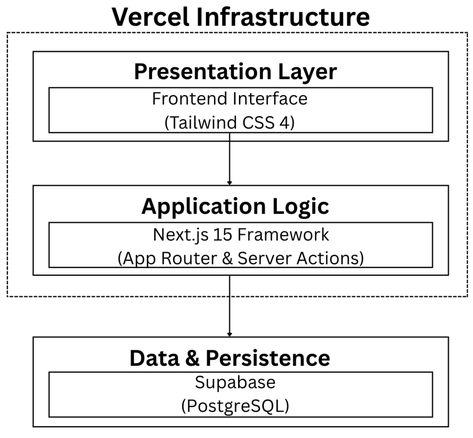

Figure 2 : High Level System Architecture

The Administrator Workflow

Admin is master controller: initializes the database; sets defense period and passwords; enters the faculty roster; and enters room availabilities.

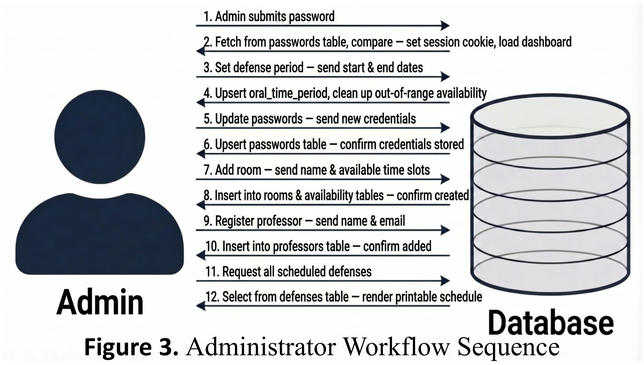

Figure 3. Administrator Workflow Sequence

The Professor Workflow

Faculty select their name from a roster and enter their available hours.

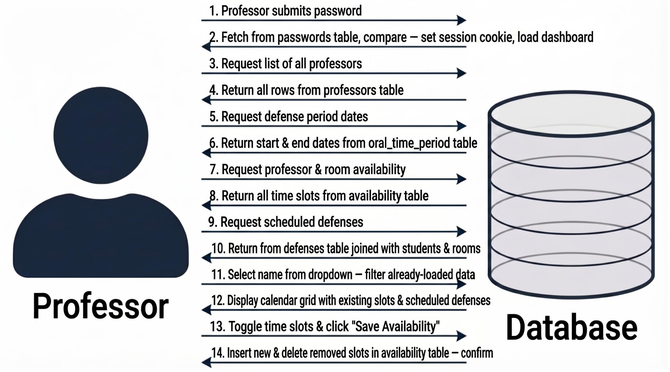

Figure 4. Professor Workflow Sequence

The Student Workflow

The student workflow is a guided booking wizard. The system filters the database in real-time, rendering a calendar that displays only mathematically valid, conflict-free time slots.

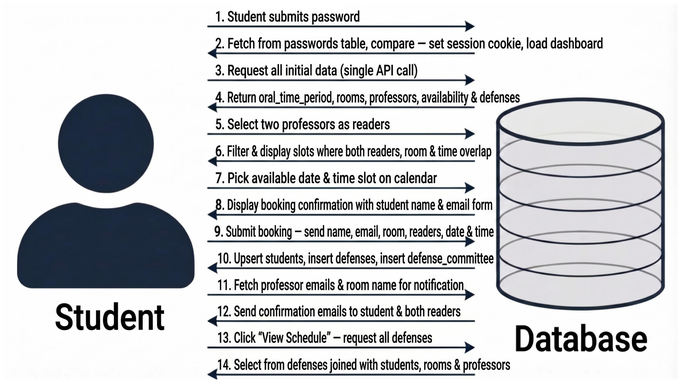

Figure 5. Student Workflow Sequence

Booking a Defense Example

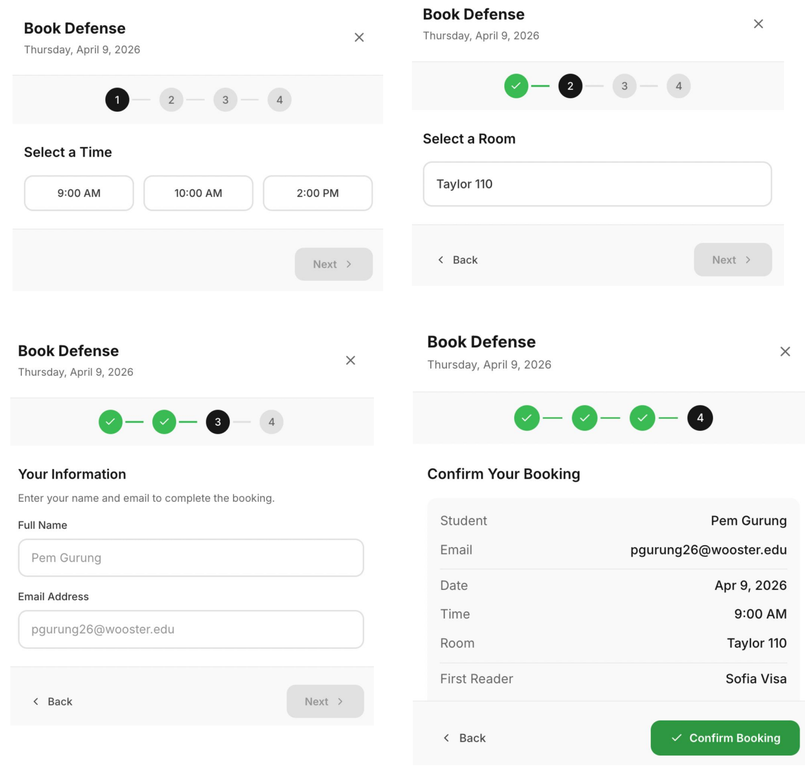

Figure 6. Example of a Student Booking Sequence

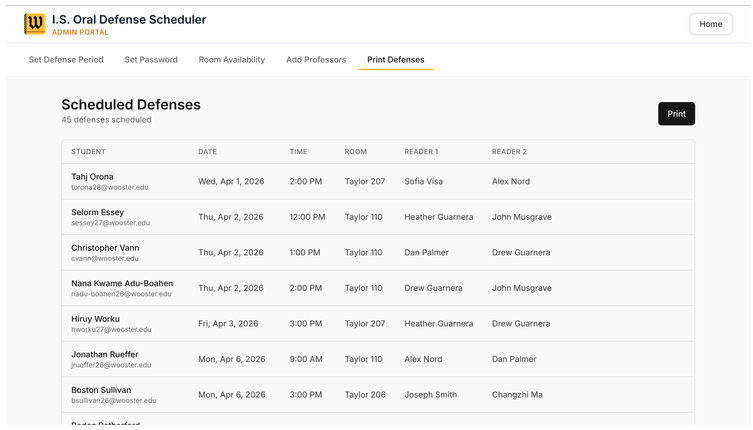

Figure 7. Example of an Oral Defense Schedule

Result & Impact

- **Successful Departmental Deployment**: deployed to schedule approximately 50 oral defense presentations for the Math, Data Science, and CS class of 2026, replacing the inefficient manual email process.
- **Zero Deadlocks Achieved**: the system strictly enforces referential constraints, with no double bookings.