system overview

Signox LogX Team

2025-07-09

Table of Contents

# 1 Smart Roster System - Comprehensive Overview

## 1.1 Introduction

The Smart Roster system is an integrated platform for managing academic operations, attendance, scheduling, user management, notifications, and analytics. It is designed for educational institutions to streamline workflows for administrators, HODs, trainers, and trainees, ensuring efficient management and real-time insights.

## 1.2 System Architecture & Modules

### 1.2.1 1. User Management

* **Features:** Registration, role assignment, profile management, facial recognition, bulk operations, password reset.
* **User Roles:** Administrator, HOD, Trainer, Trainee.
* **Highlights:** Secure email verification, unique email enforcement, facial image upload for attendance, bulk user import.
* **Integration Points:** User data is linked to attendance, scheduling, notifications, and reporting modules. Role-based access controls all system features.

### 1.2.2 2. School Management

* **Features:** School, department, programme, class group, academic year/term management.
* **Highlights:** Unique school instance, department/HOD assignment, unique class group names, non-overlapping academic terms.
* **Integration Points:** School structure underpins course mapping, timetable generation, and reporting granularity.

### 1.2.3 3. Course Management

* **Features:** Course creation, topic/subtopic definition, mapping to programmes, enrollment, analytics, bulk registration.
* **Highlights:** Prerequisite enforcement, session structure, Excel import, analytics dashboard.
* **Integration Points:** Courses are mapped to programmes and class groups, feeding into timetable and attendance modules.

### 1.2.4 4. Timetable Management

* **Features:** Room management, trainer availability, session scheduling, conflict detection, draft/publish workflows.
* **Highlights:** Real-time conflict checks, geolocation for attendance, session types (lecture, lab, etc.), notifications for changes.
* **Integration Points:** Timetables are generated based on course mappings, trainer/class group availability, and room resources. Changes trigger notifications and update attendance expectations.

### 1.2.5 5. Attendance Management

* **Features:** Session lifecycle, attendance marking, validation (facial, geolocation), analytics, troubleshooting.
* **Highlights:** Automated validation, session status tracking, analytics by course/group/trainer.
* **Integration Points:** Attendance is linked to timetable sessions, user profiles (facial data), and reporting/analytics modules.
* **Example:** A trainee marks attendance using facial recognition and geolocation at the scheduled session time; the system validates and logs the record.

### 1.2.6 6. Notifications

* **Features:** In-app, email, push notifications, user preferences, broadcast, delivery tracking.
* **Highlights:** Customizable preferences, mandatory system alerts, Do Not Disturb, delivery troubleshooting.
* **Integration Points:** Notifications are triggered by events in attendance, timetable, reporting, and user management modules.
* **Example:** A trainer receives a push notification when a timetable session is rescheduled.

### 1.2.7 7. Reports & Analytics

* **Features:** Attendance, performance, operational, and custom reports; export (PDF, Excel, CSV, JSON); scheduling; notifications.
* **Highlights:** Real-time and scheduled reports, advanced filtering, export for stakeholders.
* **Integration Points:** Reports aggregate data from all modules, supporting compliance, performance tracking, and decision-making.
* **Example:** An admin schedules a weekly attendance report for all departments, delivered via email and in-app notification.

## 1.3 Cross-Module Workflows & Examples

* **Enrollment to Attendance:** Admin enrolls a class group in a course → Timetable is generated → Trainees and trainers receive notifications → Attendance is marked and validated during sessions.
* **Bulk Operations:** Admins can import users or courses in bulk, with system validation and feedback for errors.
* **Role-Based Access:** HODs can manage only their department’s structure and trainers; trainers see only their assigned courses and schedules.
* **Automated Alerts:** System notifies users of schedule changes, low attendance, or new reports based on their preferences.

## 1.4 Security & Data Integrity

* **Authentication:** Email/password, email verification, password reset.
* **Authorization:** Role-based access control for all modules, with audit trails for sensitive actions.
* **Data Validation:** Unique constraints, prerequisite checks, conflict detection, facial/geolocation validation.
* **Audit & Logs:** System logs for key actions, error tracking, and troubleshooting support. Logs are accessible to admins for compliance.
* **Privacy:** Facial images and sensitive data are stored securely and used only for attendance/security purposes.

## 1.5 Maintenance & Support (Signox)

### 1.5.1 What We Maintain

* **Core Application:** All backend and frontend code, APIs, and integrations as delivered.
* **Database Schema:** Structure, migrations, and integrity of the delivered database.
* **Bug Fixes:** Resolution of bugs in delivered modules within the agreed support period.
* **Security Updates:** Patching vulnerabilities in the delivered codebase.
* **Documentation:** Updates to technical and user manuals for delivered features.
* **Deployment Scripts:** Docker, compose files, and CI/CD scripts as provided.
* **Monitoring Integrations:** Provided monitoring scripts and dashboards (e.g., Prometheus, Grafana).
* **Initial Training:** Onboarding and documentation for system users and admins.

### 1.5.2 What We Do Not Maintain

* **Custom Code Changes:** Any modifications made by the client or third parties after delivery.
* **External Integrations:** Third-party services or APIs not included in the original scope.
* **Client Infrastructure:** Hardware, network, or cloud resources not provisioned by Signox.
* **User Data Management:** Data entry, migration, or manual corrections performed by client staff.
* **Unsupported Plugins/Extensions:** Any add-ons not delivered or approved by Signox.
* **Ongoing Training:** Training beyond the initial onboarding and documentation.
* **End-User Devices:** Maintenance of client/user hardware, browsers, or mobile devices.

### 1.5.3 Maintenance Best Practices

* **Backups:** Regularly back up the database and configuration files. Test restore procedures periodically.
* **Monitoring:** Use provided dashboards to monitor system health and performance. Set up alerting for critical issues.
* **Updates:** Apply security and bug fix updates as released by Signox. Review release notes for impact.
* **User Management:** Periodically review user roles and access rights to prevent privilege creep.
* **Incident Reporting:** Report issues via the agreed support channel for timely resolution. Provide logs and error details for faster troubleshooting.
* **Change Management:** Document and review any client-side changes to ensure maintainability and support eligibility.

## 1.6 Contact & Support

For support, maintenance requests, or documentation updates, contact the Signox support team as per your service agreement. Include relevant logs, screenshots, and a description of the issue for efficient resolution.

**Last updated:** July 2025