Final Project Overview

Our CST 438 Team (Code Busters) collaborated and developed a student registration and gradebook web application as a set of connected microservices. In this project, we created functionality that enables:

- **Administrators:** Manage users, courses, and sections.
- **Instructors:** Manage assignments, enrollments, and grades.
- **Students:** Enroll in courses, view schedules, check assignments, and access academic transcripts.

The application utilizes a **React** frontend connected to separate **Spring Boot** registrar and gradebook backend services.

Technology Stack

Category	Technologies / Tools Used
Languages	Java, JavaScript, JSX
Frameworks & Runtimes	Spring Boot 3.4.5, React 19, Vite
Core Libraries & Messaging	Spring Data JPA, Spring Security, JWT, MySQL, H2, RabbitMQ, React Router
Testing & Build Tools	Maven, Java 17, Selenium

Project Architecture & File Organization

1. Frontend (`cst438-team03-frontend/`)

- `src/App.jsx` — Authentication and role-based routing
- `src/Login.jsx` — Login form and registrar authentication requests
- `src/Constants.jsx` — Backend API URLs, user roles, and semester values
- `src/components/admin/` — User, course, and section management components
- `src/components/instructor/` — Assignment, grade, enrollment, and section management
- `src/components/student/` — Course enrollment, schedule views, assignment tracking, and transcripts

2. Registrar Backend (`cst438-team-03-registrar/`)

- `controller/` — Login, user, course, section, and student APIs
- `domain/` — Registration-related data entities
- `dto/` — Data-transfer objects
- `service/` — Business logic and gradebook service communication
- `SecurityConfiguration.java` — Security and JWT resource-server configuration
- `GradebookServiceProxy.java` — Inter-service communication proxy for the gradebook service

3. Gradebook Backend (`cst438-team03-gradebook/`)

- `controller/` — Assignment, enrollment, grade, login, and health check APIs
- `domain/` — Gradebook-related data entities
- `dto/` — Data-transfer objects
- `service/` — Gradebook logic and registrar service communication
- `SecurityConfiguration.java` — Security and JWT resource-server configuration
- `RegistrarServiceProxy.java` — Inter-service communication proxy for the registrar service

---

System Integration & Workflow

1. **Authentication:** The React frontend authenticates users through the registrar service and securely stores the returned JSON Web Token (JWT) in session storage.
2. **Role-Based Routing:** Following login, users are routed to tailored dashboards depending on their assigned role (Administrator, Instructor, or Student).
3. **Service Separation:** The *Registrar Service* manages core user records, course offerings, section schedules, and student enrollments. The *Gradebook Service* manages assignment creation, grade submissions, and academic evaluations.
4. **Inter-Service Communication:** The two Spring Boot services communicate via dedicated service proxy classes, enabling a clean, decoupled microservices architecture.

Execution Instructions

1. Start the Registrar Backend

```
Bash
cd cst438-team-03-registrar

./mvnw spring-boot:run
```

2. Start the Gradebook Backend

```
Bash
cd cst438-team03-gradebook

./mvnw spring-boot:run
```

3. Start the React Frontend

```
Bash
cd cst438-team03-frontend

npm install

npm run dev
```

Default Local Endpoints

- **Registrar Service:** http://localhost:8080
- **Gradebook Service:** http://localhost:8081

Note: Backend services run on Java 17 / Maven and include support for MySQL, H2, Spring Security, JWT validation, Selenium testing, and RabbitMQ. Ensure environment variables or application configuration files are set before launching.

Repositories

- https://github.com/cst438-team03-org0/cst438-team-03-registrar
- https://github.com/cst438-team03-org0/cst438-team03-frontend
- https://github.com/cst438-team03-org0/cst438-team03-gradebook