Namal University, Mianwali

# Design Document

# University Timetable Management System

18-January-2026


## Authors:

Hammad Shabbir
Ahmer Sultan
Hasnain Ali


## Requirement Provider (Client):

Mr. Abdul Rafay

*Confidential Document - For Internal Use Only*

# Table of Contents

**1 Introduction** **3**
1.1 Purpose . . . 3
1.2 Project Overview . . . 3
1.3 Design Approach . . . 3

**2 Design Assumptions and Constraints** **3**
2.1 Design Assumptions . . . 3
2.2 Design Constraints . . . 3

**3 Key Design Decisions** **4**
3.1 Architectural Decisions . . . 4
3.2 Database Design Decisions . . . 4
3.3 Security Design Decisions . . . 4
3.4 User Interface Design Decisions . . . 5

**4 System Design Diagrams** **6**
4.1 Use Case Diagram . . . 6
4.2 Data Flow Diagrams . . . 7
4.2.1 Data Flow Diagram - Level 0 . . . 7
4.2.2 Data Flow Diagram - Level 1 . . . 8
4.2.3 Data Flow Diagram - Level 2 - Dashboard . . . 9
4.2.4 Data Flow Diagram - Level 2 - User Management . . . 10
4.2.5 Data Flow Diagram - Level 2 - Notification & Communication . . . 11
4.2.6 Data Flow Diagram - Level 2 - System Configuration . . . 12
4.2.7 Data Flow Diagram - Level 2 - Conflict Management . . . 13
4.2.8 Data Flow Diagram - Level 2 - Timetable Management . . . 14
4.3 Sequence Diagrams . . . 16
4.3.1 User Login . . . 16
4.3.2 Bulk User Import . . . 18
4.3.3 Password Reset . . . 20
4.3.4 Publish Timetable . . . 21
4.3.5 Timetable Creation With Conflict Detection . . . 22
4.3.6 Conflict Resolution . . . 23
4.3.7 Notification & Communication Process . . . 24
4.3.8 Automatic Timetable Generation . . . 26
4.4 Activity Diagrams . . . 28
4.4.1 User Login . . . 28
4.4.2 User Registration . . . 29
4.4.3 Report Generation . . . 31
4.4.4 Bulk User Import . . . 32
4.4.5 Password Reset . . . 33
4.4.6 Conflict Detection and Resolution . . . 34
4.4.7 Timetable Creation and Publishing . . . 35

4.4.8 Notification Delivery . . . . . 36
4.4.9 Auto-Timetable Generation . . . . . 37
4.5 Class Diagram . . . . . 38
4.6 Component Diagram . . . . . 39

**5 Requirements-Design Traceability** **40**
5.1 Traceability Matrix . . . . . 40
5.2 Coverage Summary . . . . . 47

**6 Figma Link** **47**

**7 GitHub Repository Link** **47**

**8 Conclusion** **47**

**A Glossary** **48**

**B Design Tools Used** **48**

# 1 Introduction

## 1.1 Purpose

The purpose of this document is to describe the architecture, components, interfaces, and detailed design of the **University Timetable Management System**. This document translates the Software Requirements Specification into technical design decisions that guide the system implementation and its verification.

## 1.2 Project Overview

The University Timetable Management System is web based application that aims to automate scheduling and management of timetables. Instead of manual and busy-schedules it substitutes it with centralized timetable creation, conflict detection and timetable access services to students, faculty, administrators.

## 1.3 Design Approach

The deisgn of our system is based on these principles:

- Independent modules and interfaces with clarity.
- Every design element traces back to SRS requirements.
- Interface designs prioritize ease of use.

# 2 Design Assumptions and Constraints

## 2.1 Design Assumptions

- Users have access to modern web browsers.
- Academic, course, and user data is provided by the university.
- Email services are available for notifications.
- User roles and permissions are predefined.
- The system operates for a single university.

## 2.2 Design Constraints

- Compliance with university academic and data privacy policies.
- Deployment limited to university IT infrastructure.
- Must support concurrent users during peak periods.
- Secure handling of sensitive user information.

# 3 Key Design Decisions

## 3.1 Architectural Decisions

The University Timetable Management System is developed based on a layered, modular. architecture which is consistent with the system context and interaction of components stipulated in the SRS. The architecture is based on a three-tier architecture:

- **Presentation Layer**: A web-based user interface that was developed towards various. user roles (Admin, Faculty, Student). This layer handles user interaction, form validation, timetable visualization and dashboard rendering.
- **Application Logic Layer**: This layer fulfills fundamental system capability. such as schedule creation, schedule editing, schedule conflict detection, schedule conflict resolution proposals, schedule publishing, and schedule notification. Business rules are enforced at this layer.
- **Data Access Layer**: In responsibility of communication with the relational database. This layer deals with permanent storing, retrieval and updating of timetables, users, Audit logs, courses, locations and system configurations.

This level of architectural separation encourages scalability, maintainability and testability, and is. shown in the sequence diagram.

## 3.2 Database Design Decisions

According to the database specifications as per the SRS, a relational database management. There is a choice of system (PostgreSQL or MySQL). The database schema is created through normalized. relations (up to Third Normal Form) to decrease redundancy and guarantee the consistency of data.

Key design decisions include:

- Separate tables for users, roles, courses, department, venue, timetables.
- Enforcement of relationships between entities using foreign key constraints, such as course, lecturers and venue.
- Soft-delete of timetables and user accounts to agree with business, audit requirements and rules.

## 3.3 Security Design Decisions

Security design decisions are derived by the non-functional security requirements defined in the SRS. The system implements Role-Based Access Control (RBAC) to ensure that users can only perform actions allowed according to their roles.

Key security design elements include:

- Authentication using secure login credentials with password hashing algorithm.

- Audit logging of critical actions such as login attempts, timetable modifications, and publishing events.
- Secure session management with timeout mechanisms access.

## 3.4 User Interface Design Decisions

The user interface design follows a role-based approach aligned with the user classes identified in the SRS. Each user role is provided with a his/her dashboard according to its responsibilities.

Key UI design decisions include:

- **Admin Dashboard**: Provides access to timetable creation, conflict reports, user management, and system configuration.
- **Faculty Dashboard**: Displays assigned courses, timetable view, and umakeup class functionality.
- **Student Dashboard**: Displays a read only personalized timetable view.

Timetables are presented using calendar-style for ease of use. These design decisions are demonstrated.

# 4 System Design Diagrams

## 4.1 Use Case Diagram

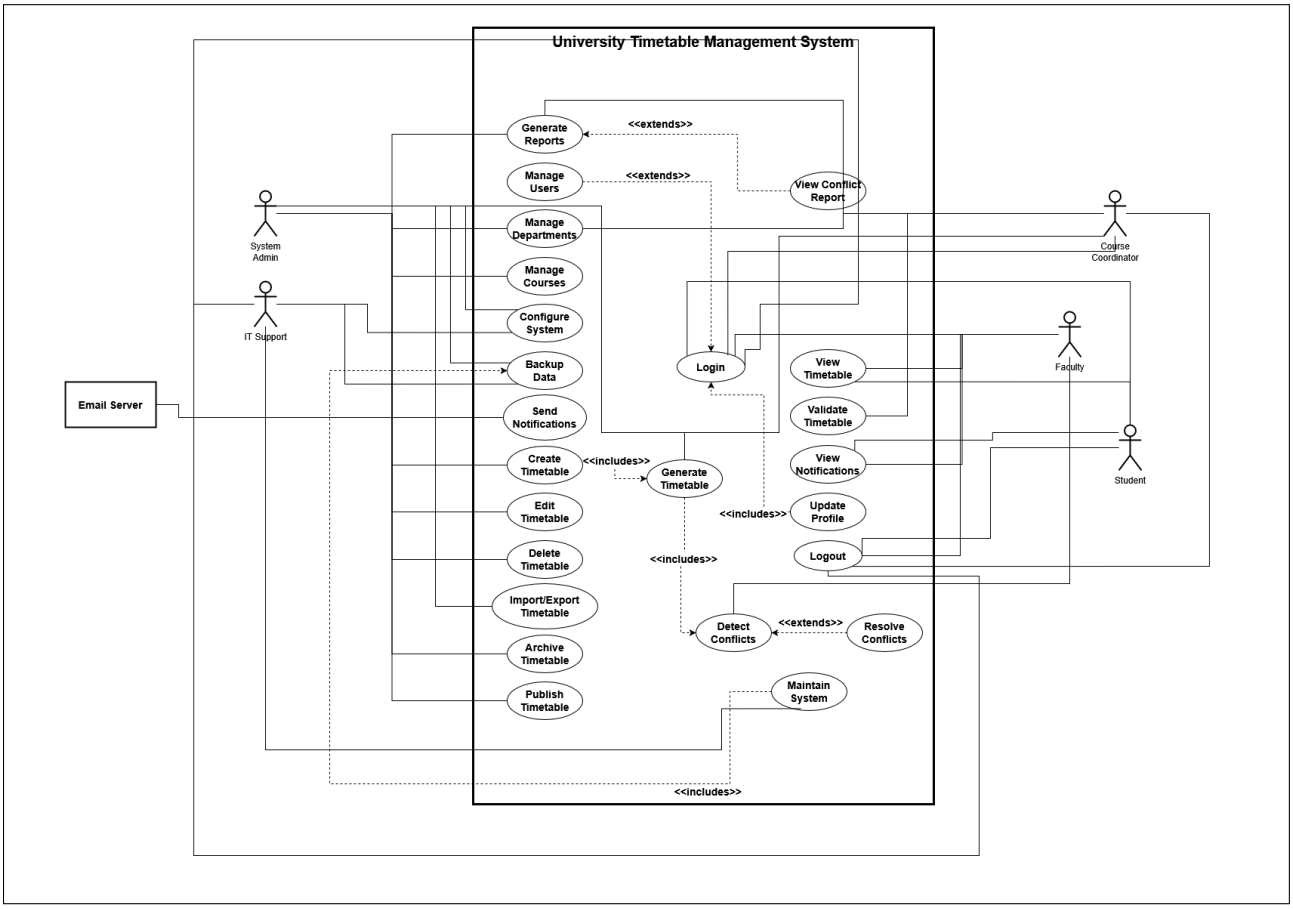



Figure 1: Use Case Diagram

## 4.2 Data Flow Diagrams

### 4.2.1 Data Flow Diagram - Level 0

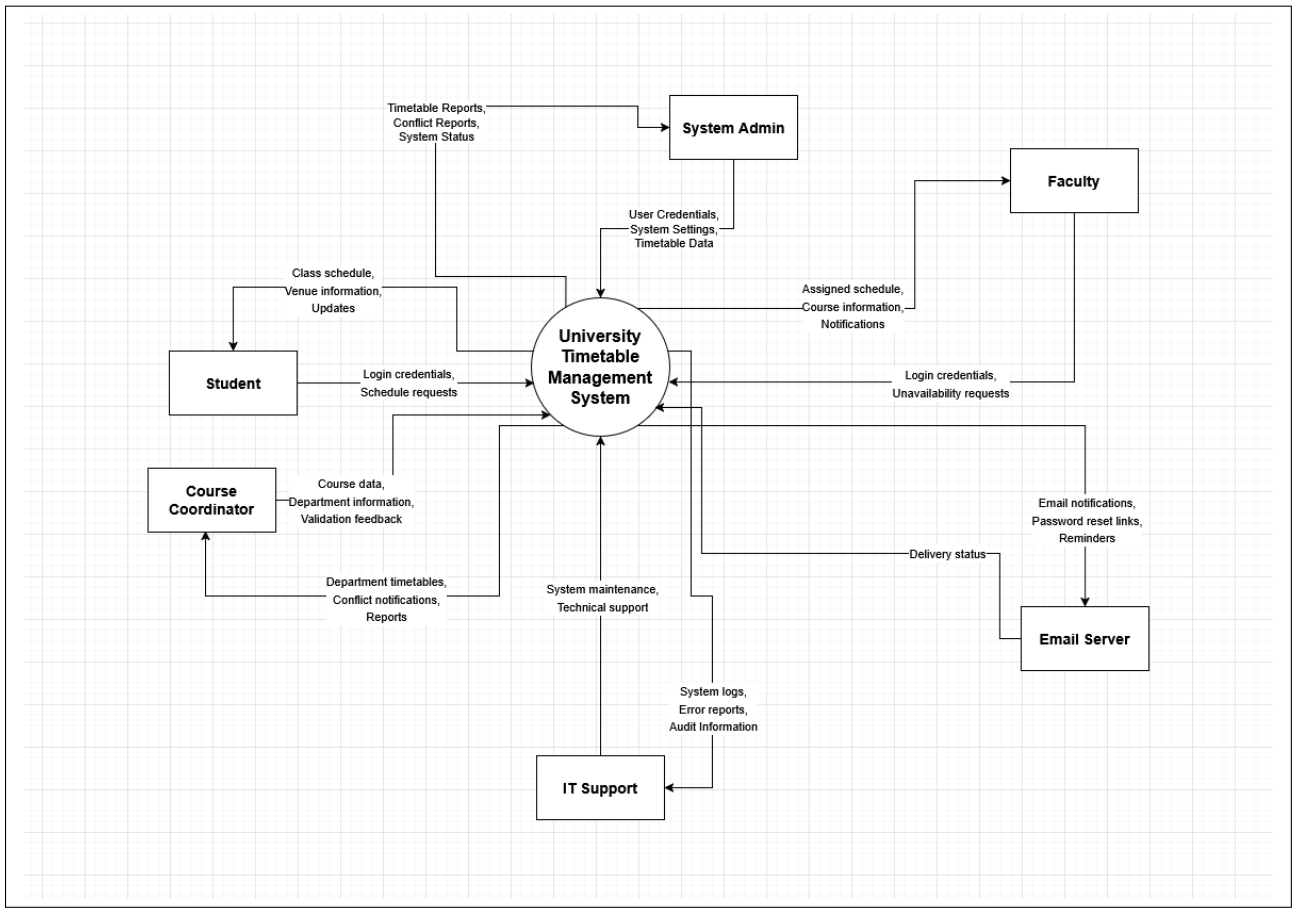



Figure 2: Data Flow Diagram - Level 0

### 4.2.2 Data Flow Diagram - Level 1

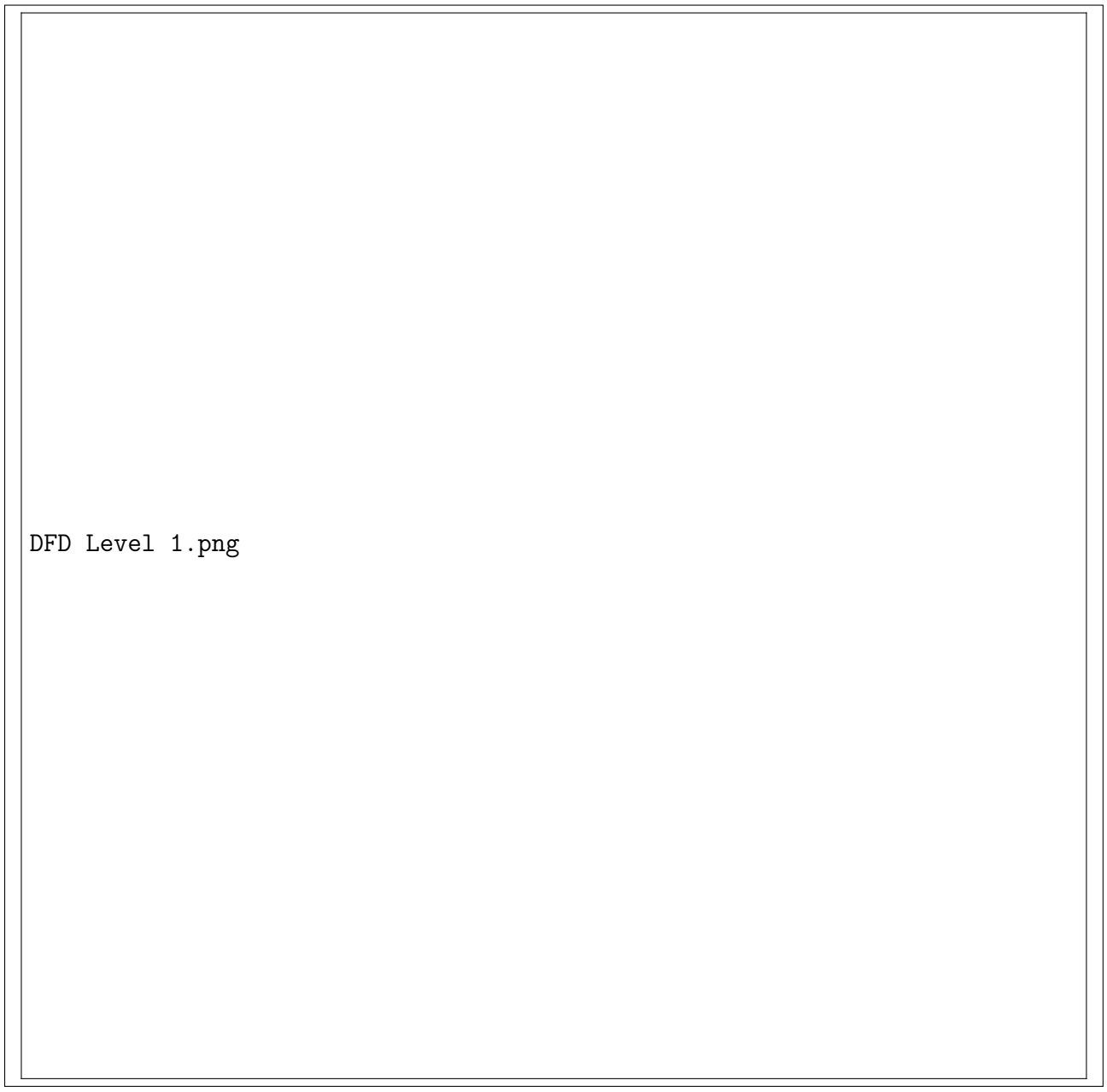



Figure 3: Data Flow Diagram - Level 1

### 4.2.3 Data Flow Diagram - Level 2 - Dashboard

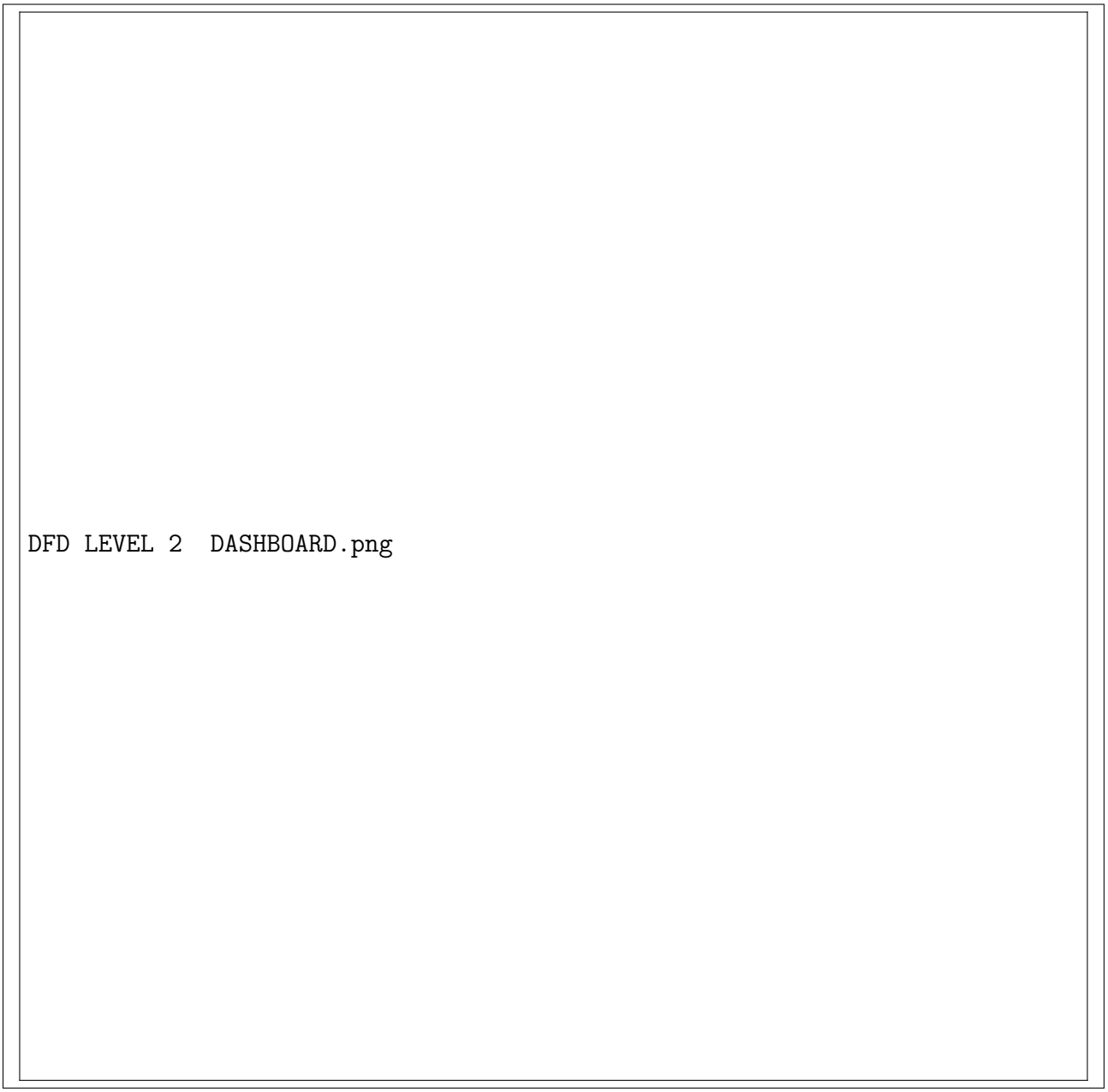



Figure 4: Data Flow Diagram - Level 2 - Dashboard

### 4.2.4 Data Flow Diagram - Level 2 - User Management

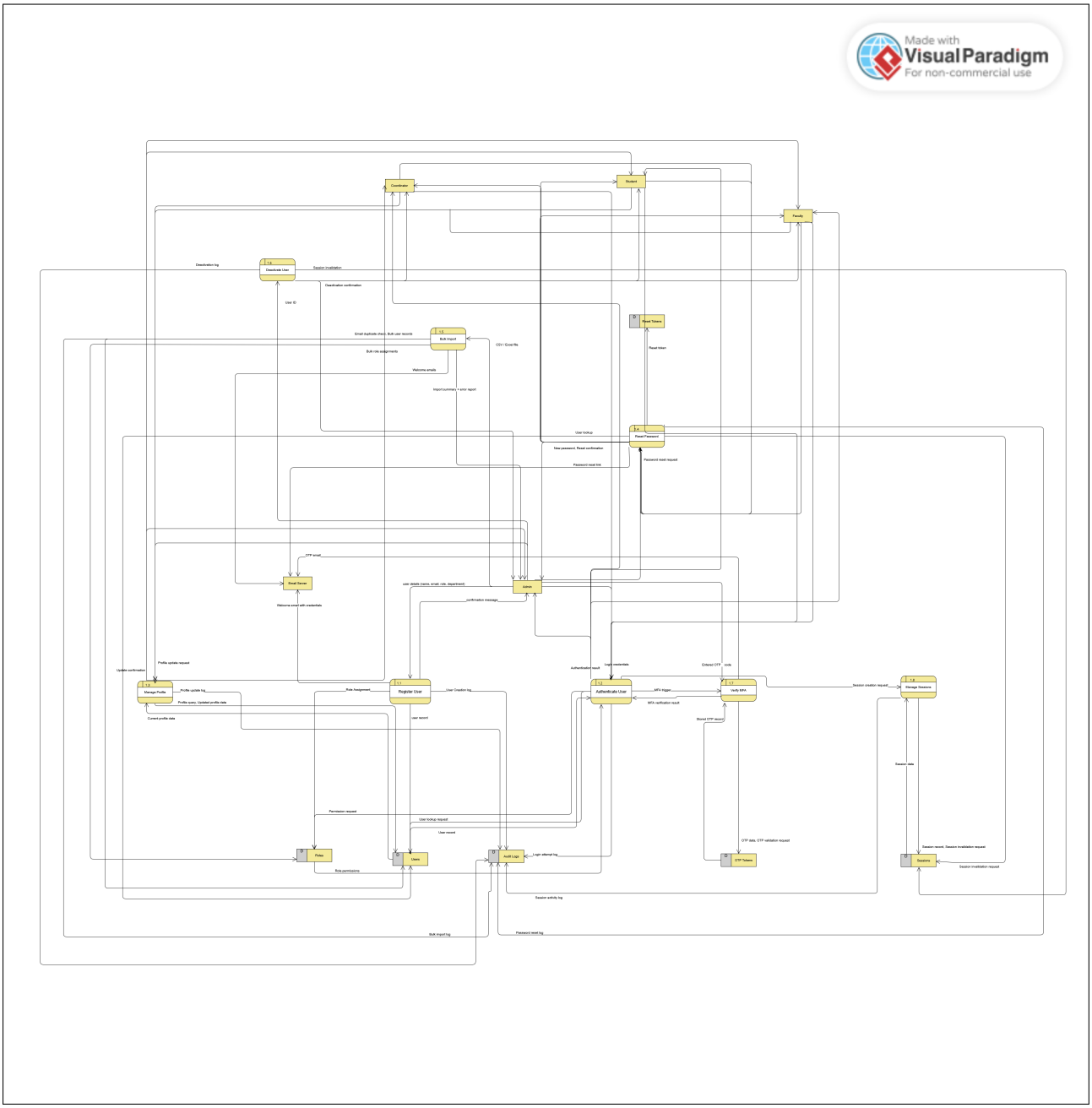



Figure 5: Data Flow Diagram - Level 2 - User Management

### 4.2.5 Data Flow Diagram - Level 2 - Notification & Communication

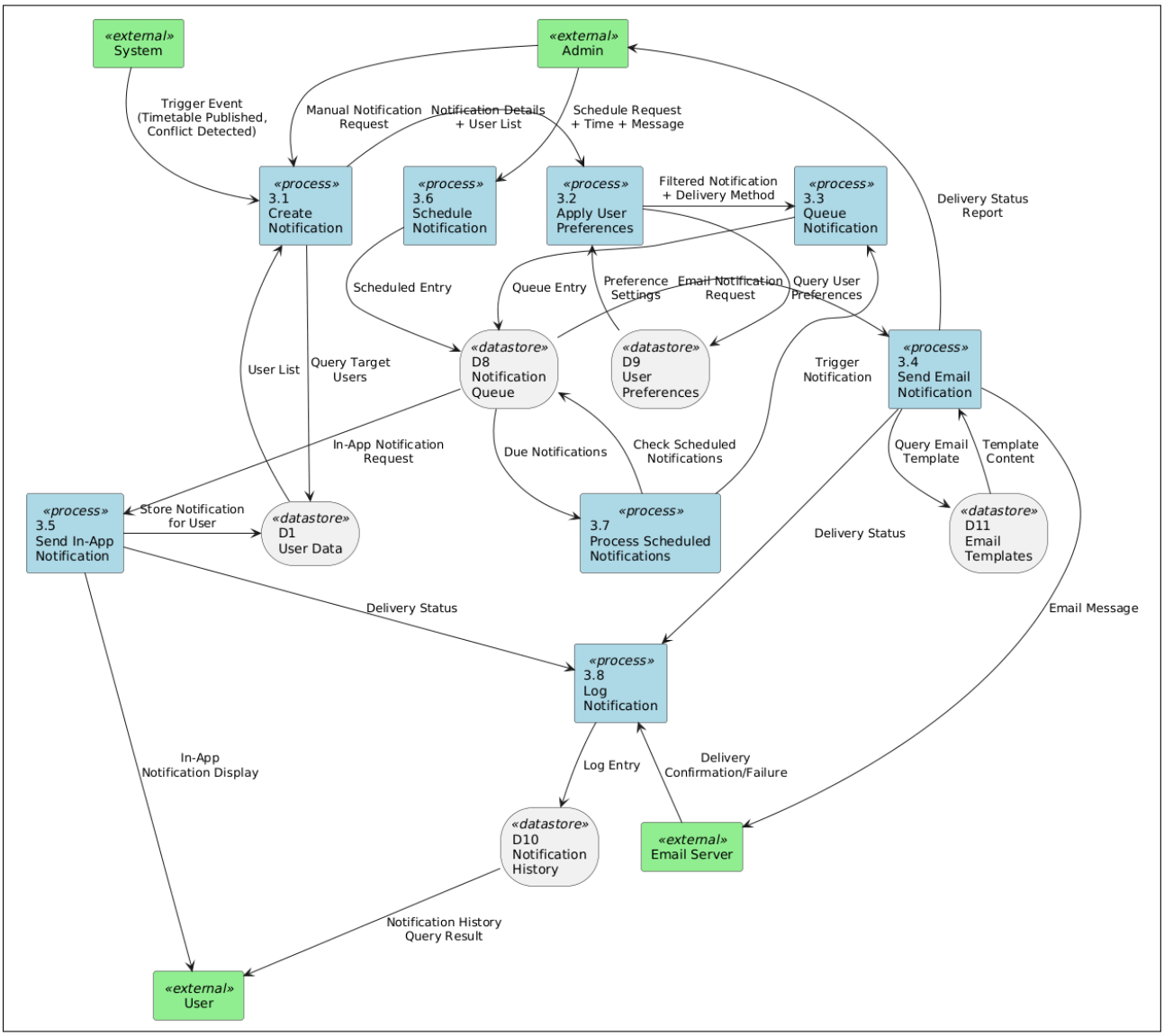



Figure 6: Data Flow Diagram - Level 2 - Notification & Communication

### 4.2.6 Data Flow Diagram - Level 2 - System Configuration

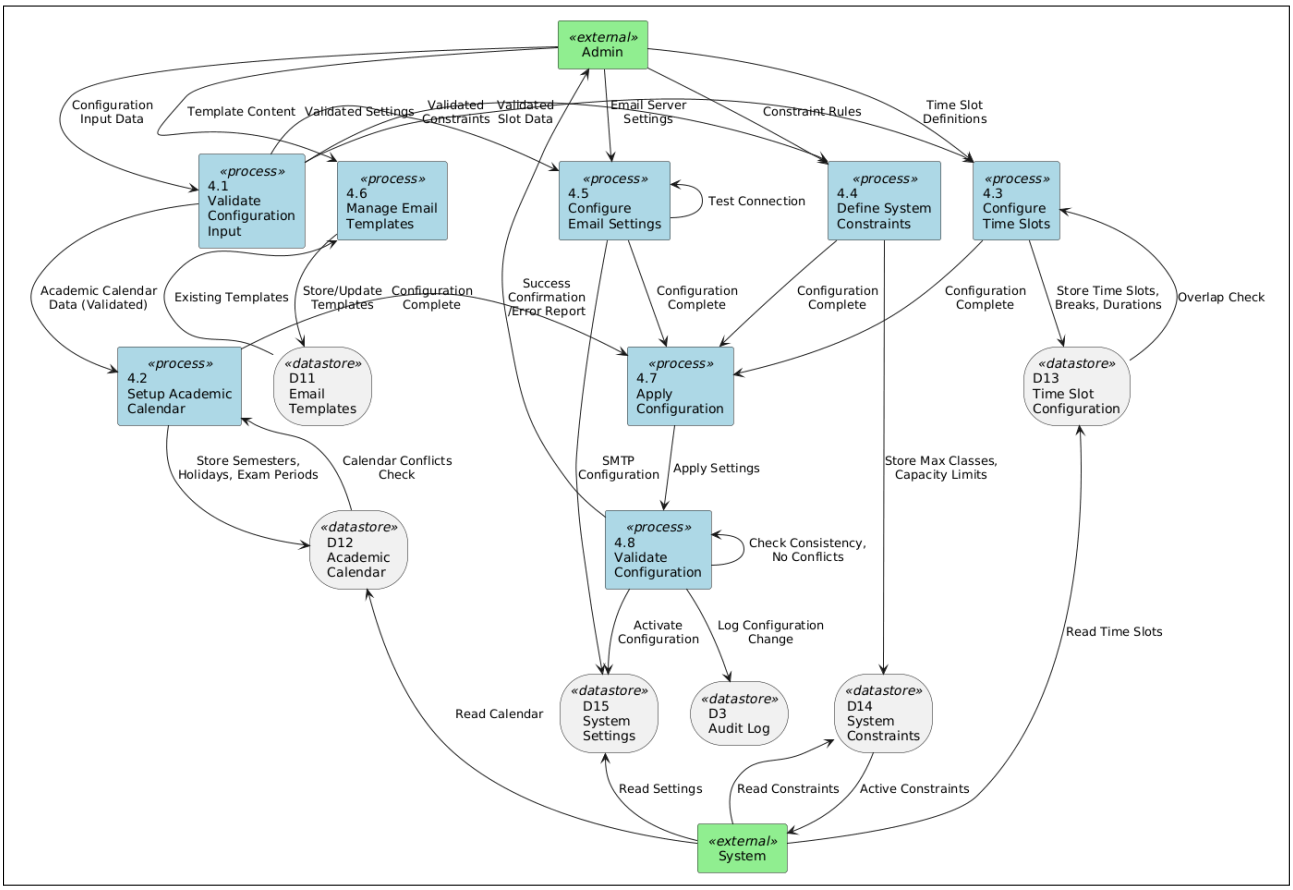



Figure 7: Data Flow Diagram - Level 2 - System Configuration

### 4.2.7 Data Flow Diagram - Level 2 - Conflict Management

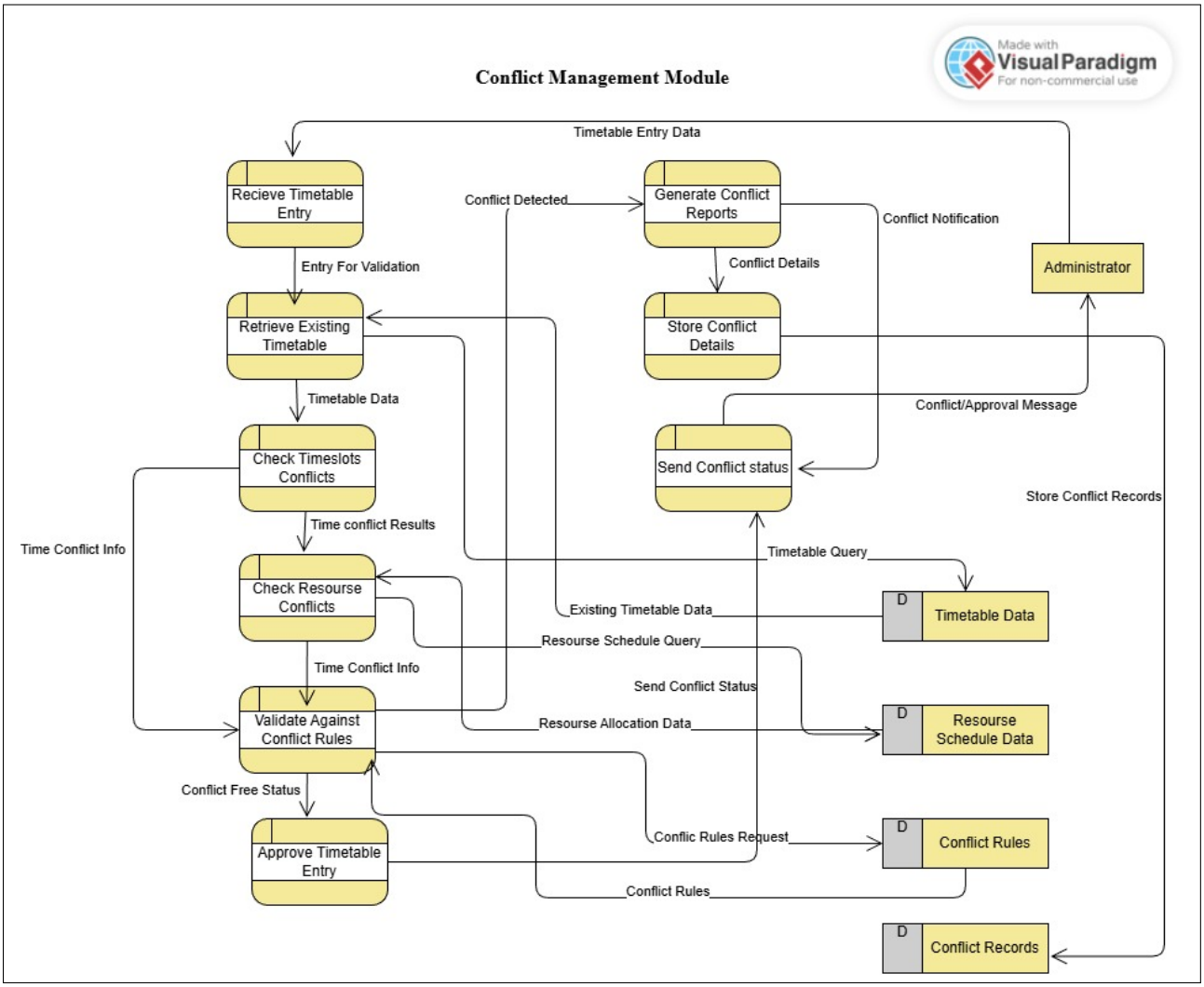



Figure 8: Data Flow Diagram - Level 2 - Conflict Management

### 4.2.8 Data Flow Diagram - Level 2 - Timetable Management

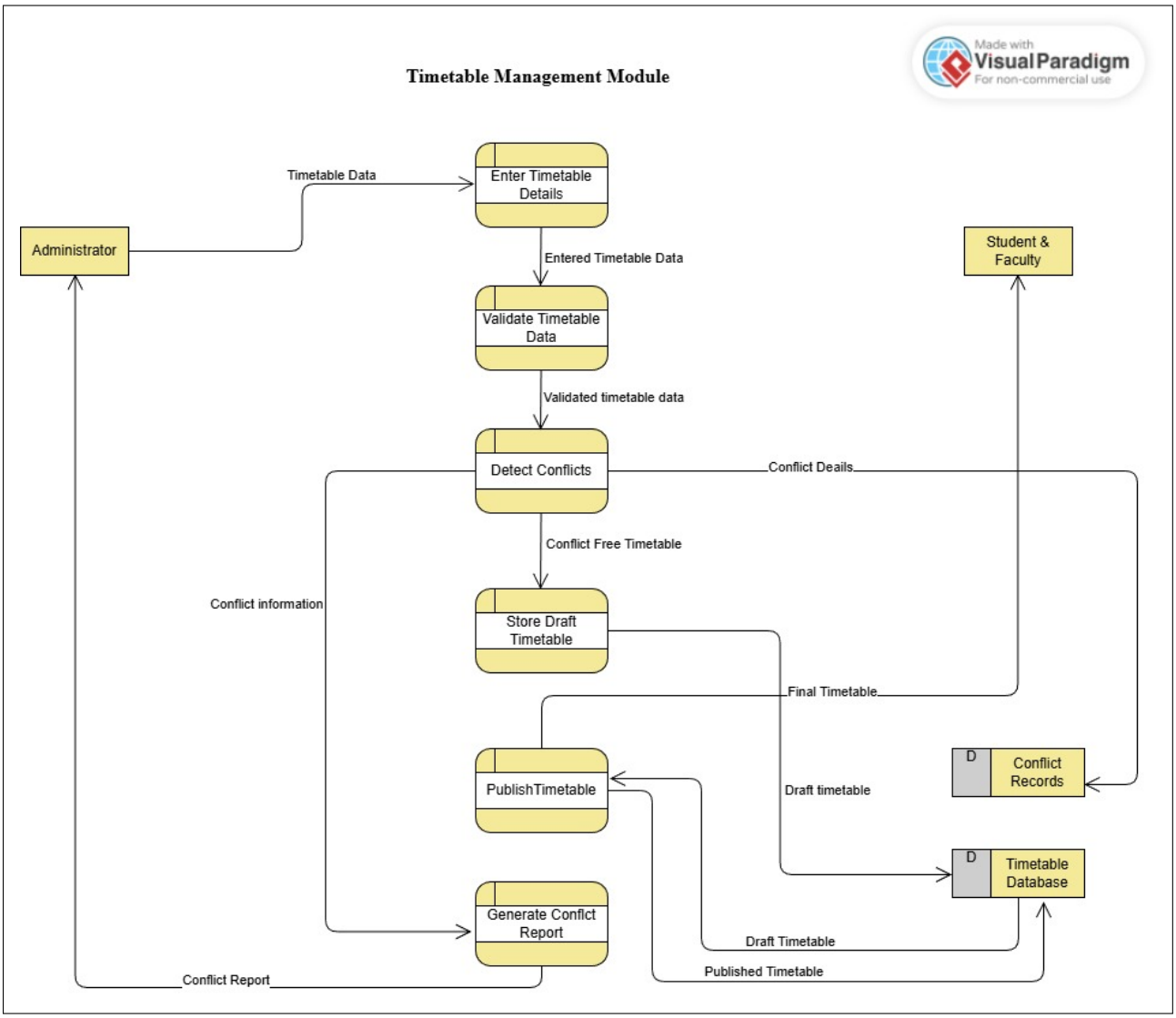



Figure 9: Data Flow Diagram - Level 2 - Timetable Management

## 4.3 Sequence Diagrams

### 4.3.1 User Login

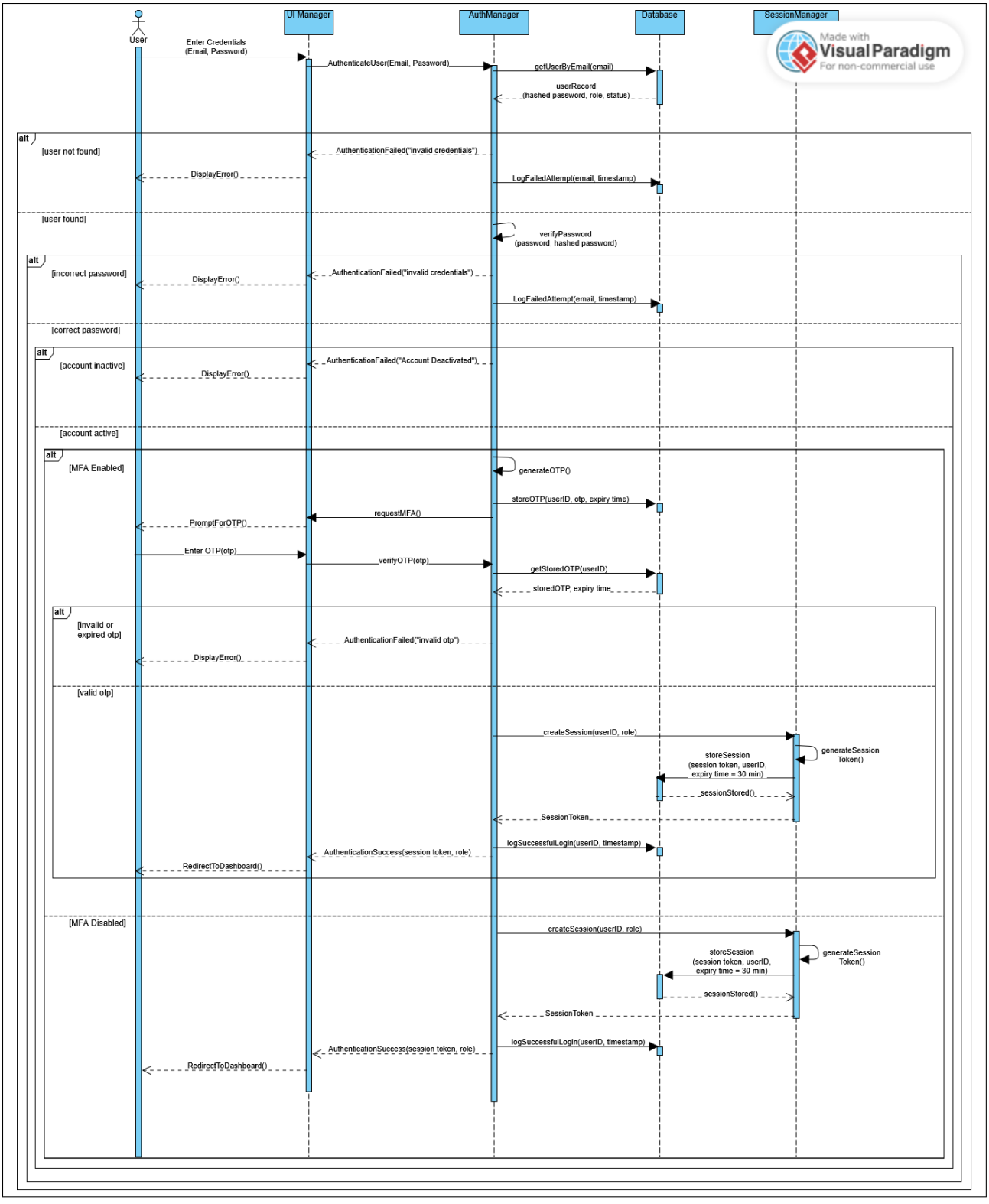
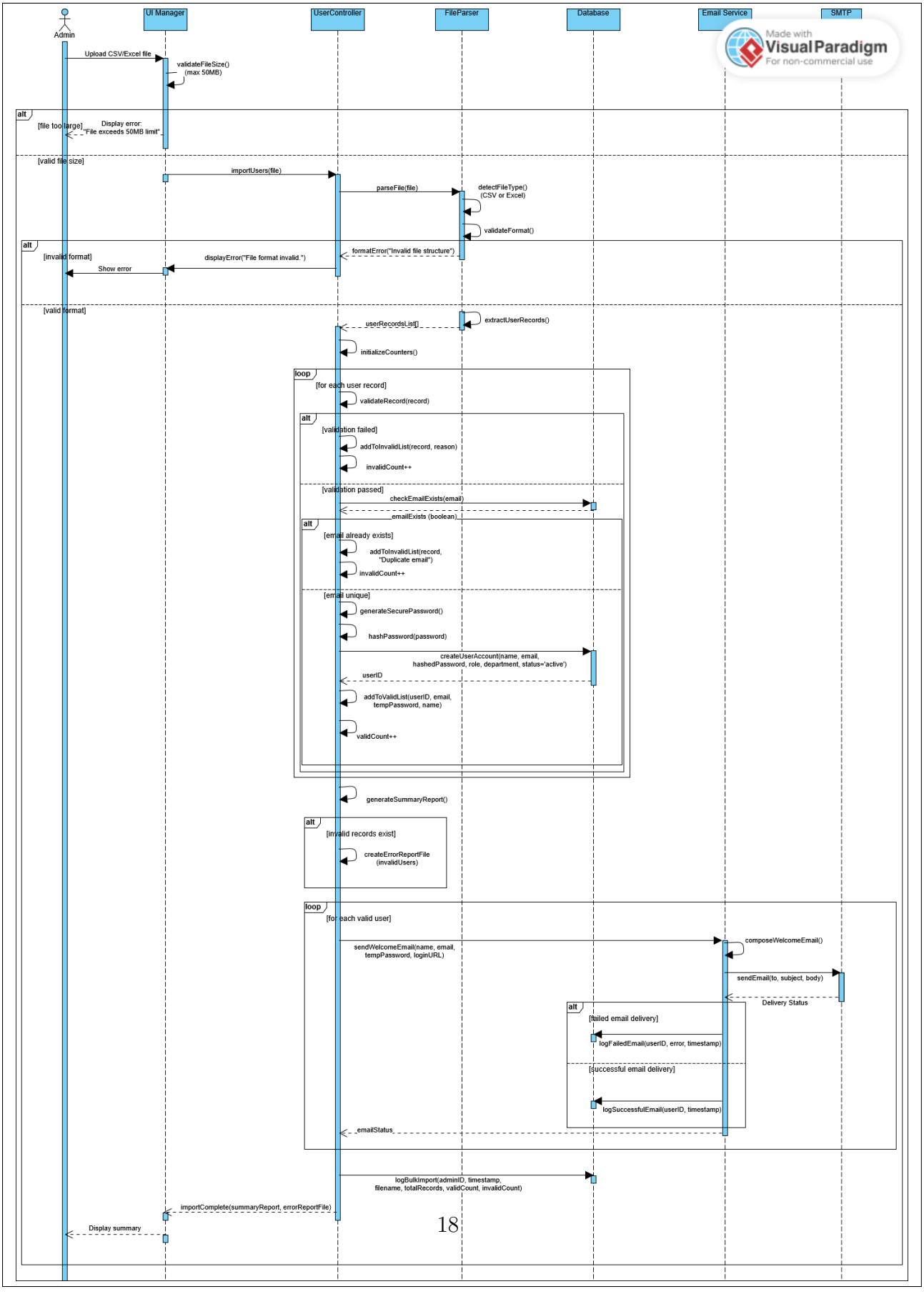



Figure 10: Sequence Diagram - User Login

### 4.3.2 Bulk User Import



### 4.3.3 Password Reset

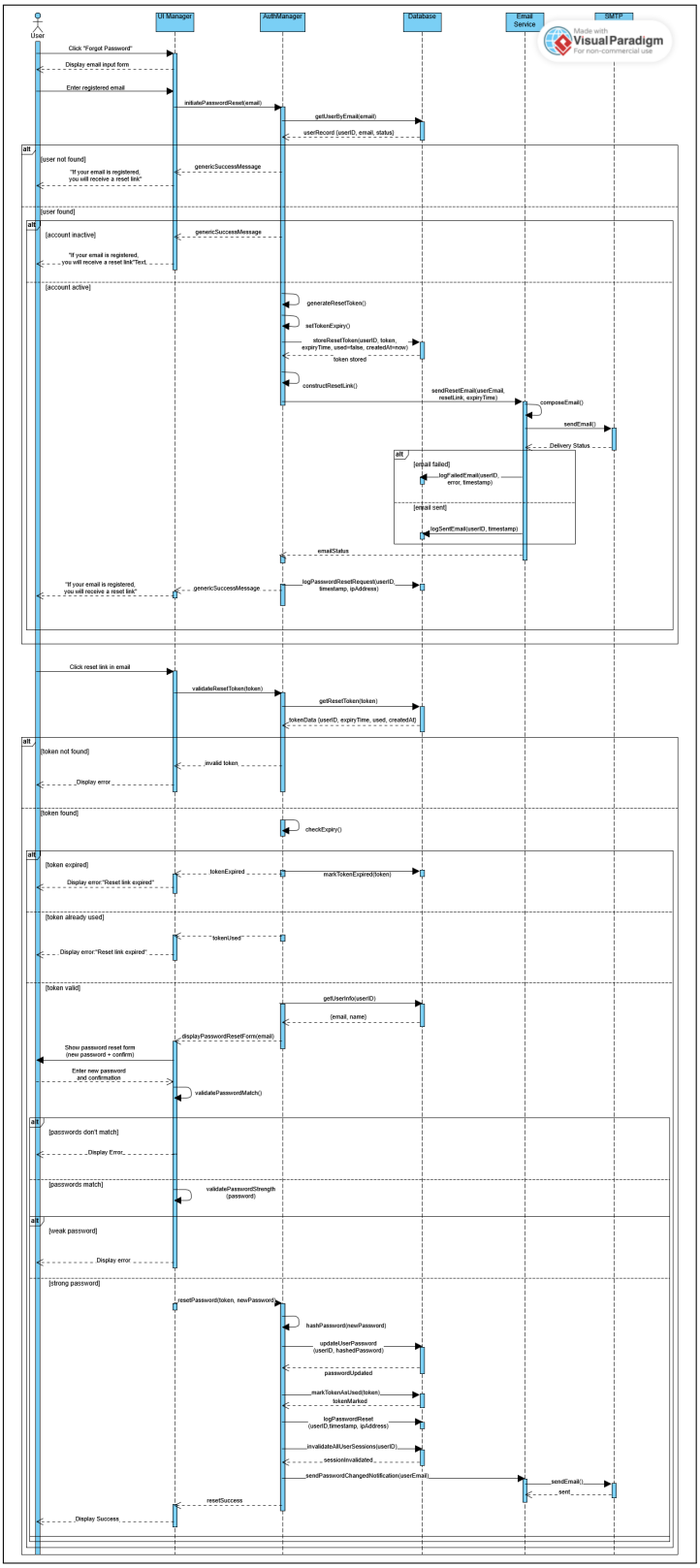



Figure 12: Sequence Diagram - Password Reset

### 4.3.4 Publish Timetable

Publish Timetable SEQ.jpeg

Figure 13: Sequence Diagram - Publish Timetable

### 4.3.5 Timetable Creation With Conflict Detection

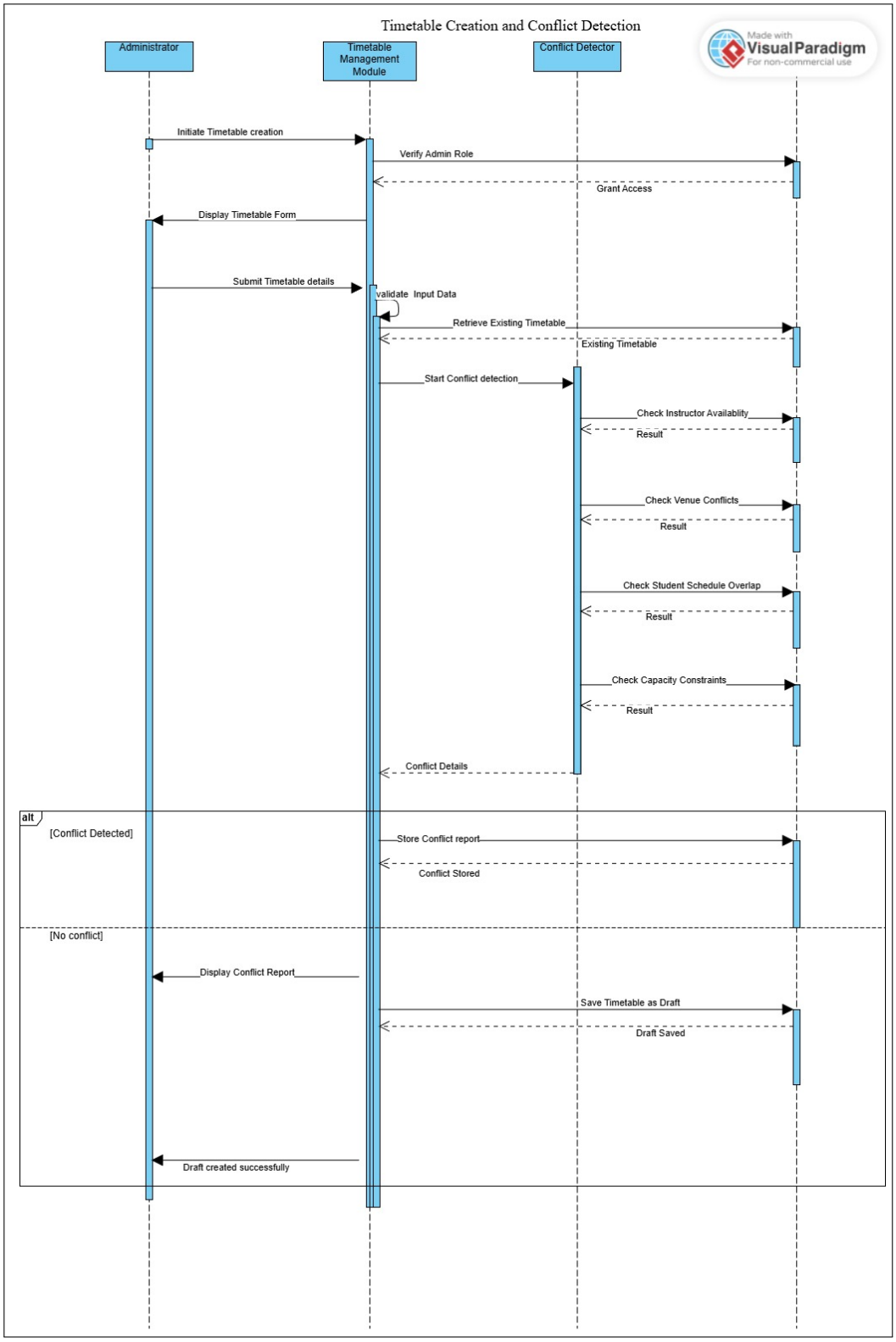



Figure 14: Sequence Diagram - Timetable Creation With Conflict Detection

### 4.3.6 Conflict Resolution

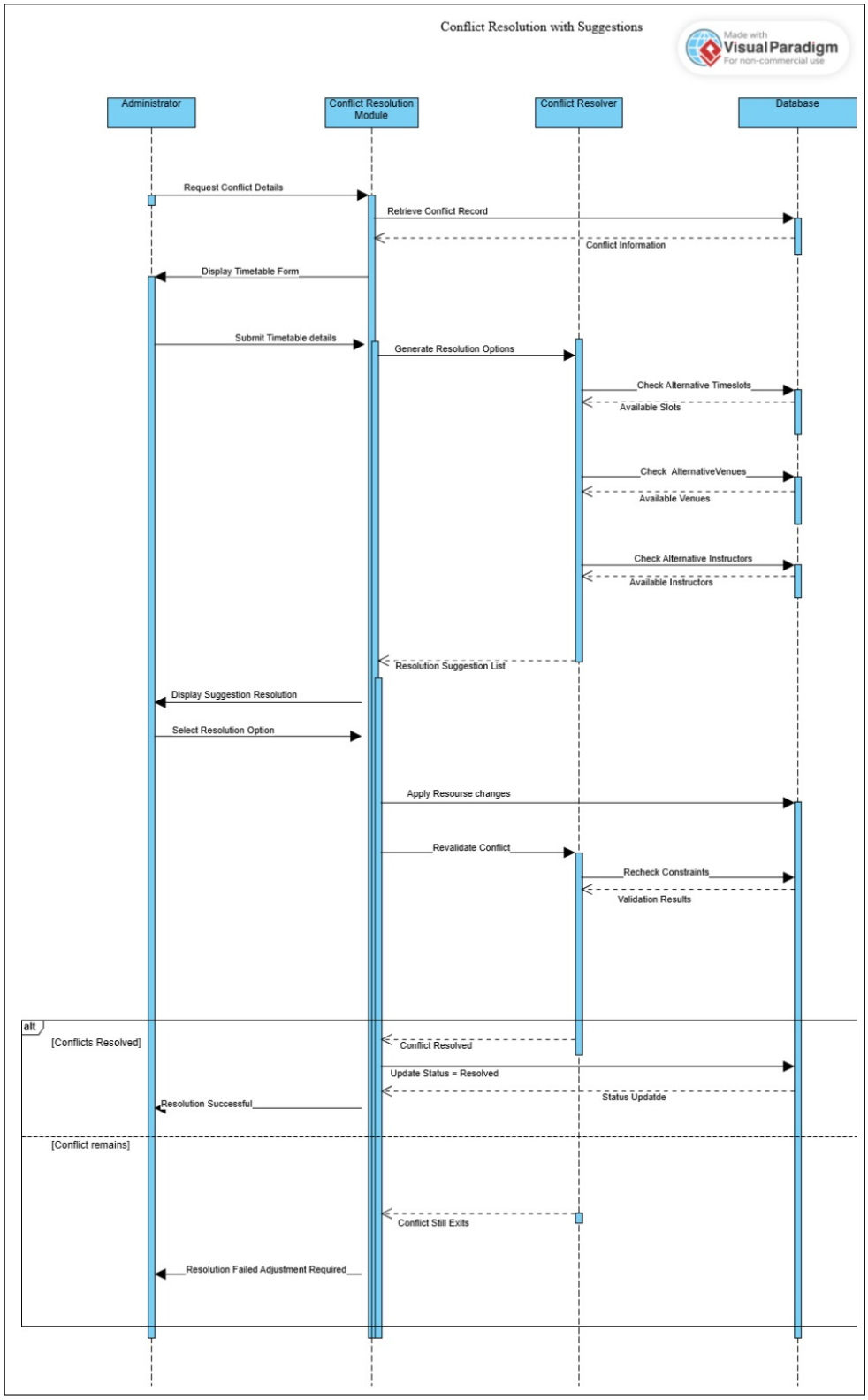



Figure 15: Sequence Diagram - Conflict Resolution

### 4.3.7 Notification & Communication Process

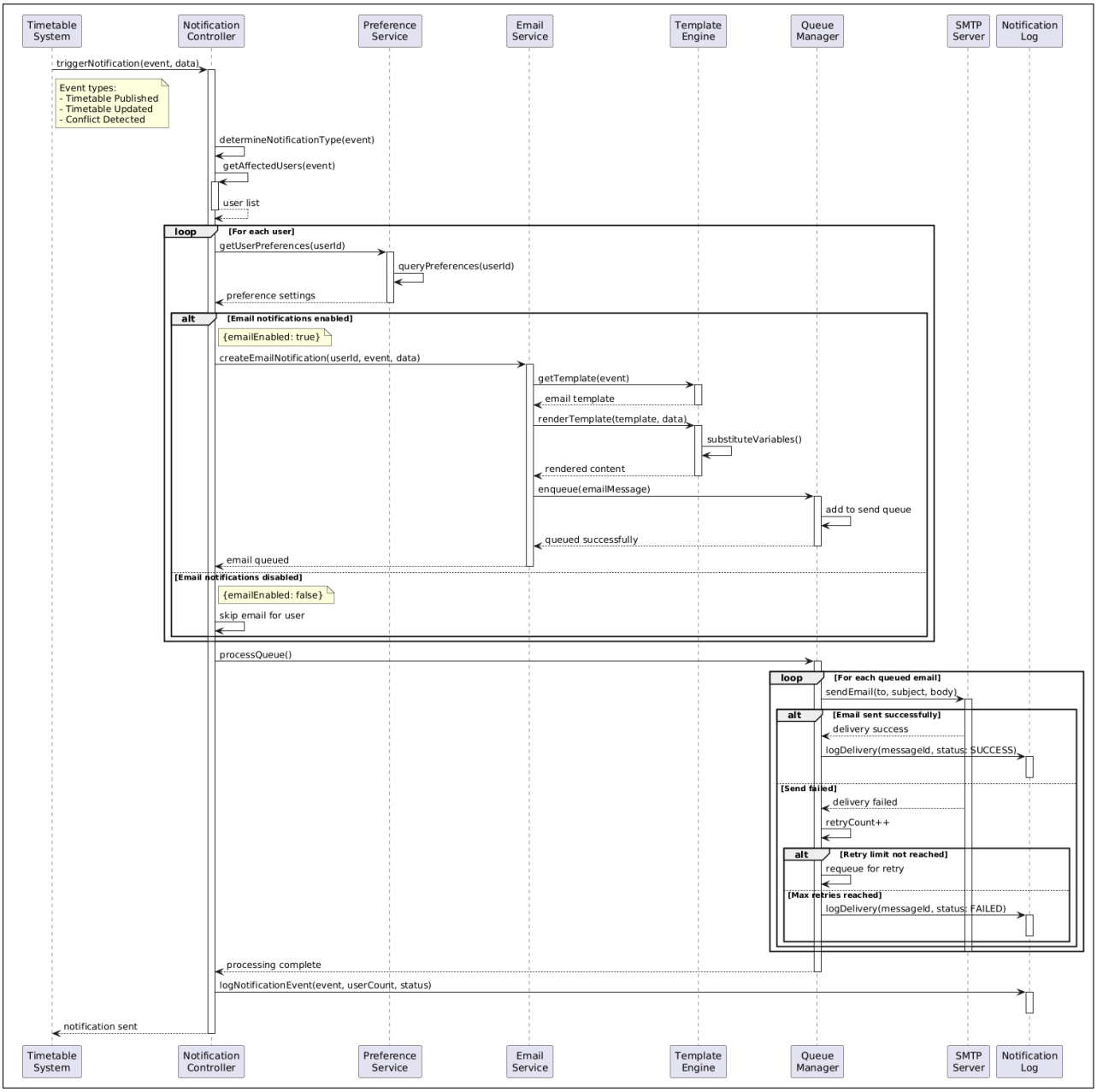
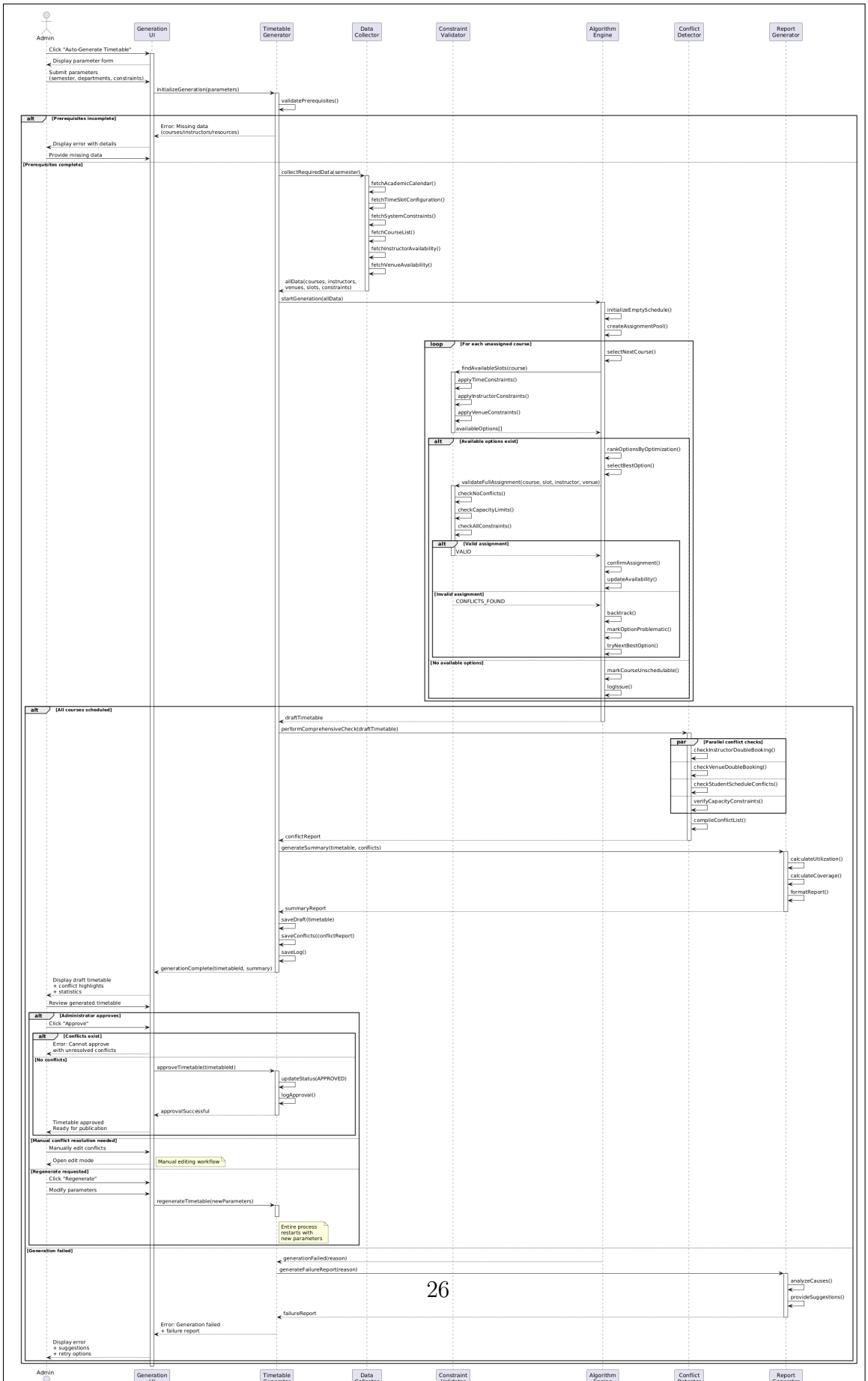



Figure 16: Sequence Diagram - Notification & Communication Process

### 4.3.8 Automatic Timetable Generation



## 4.4 Activity Diagrams

### 4.4.1 User Login

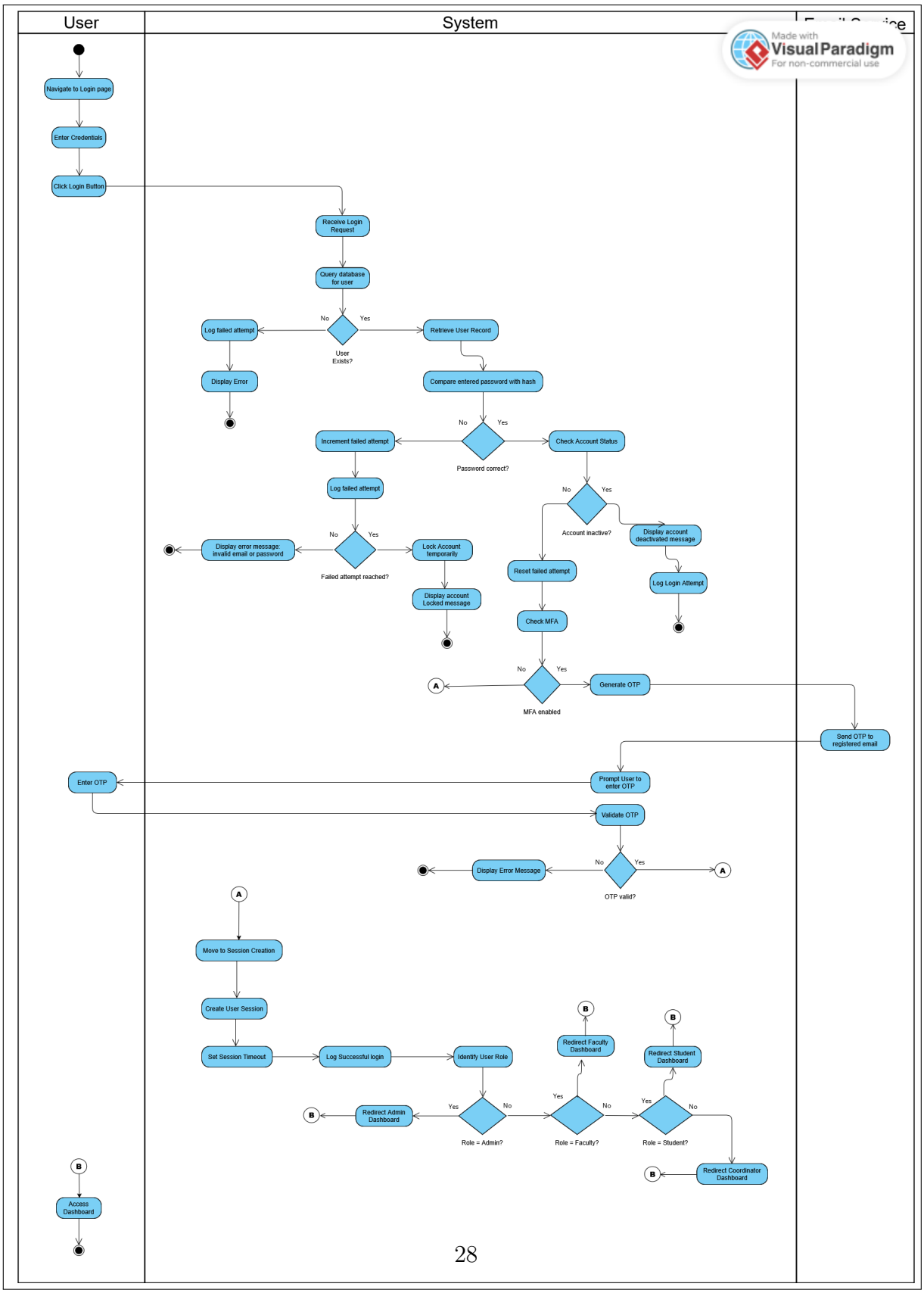





Figure 18: Activity Diagram - User Login

### 4.4.2 User Registration

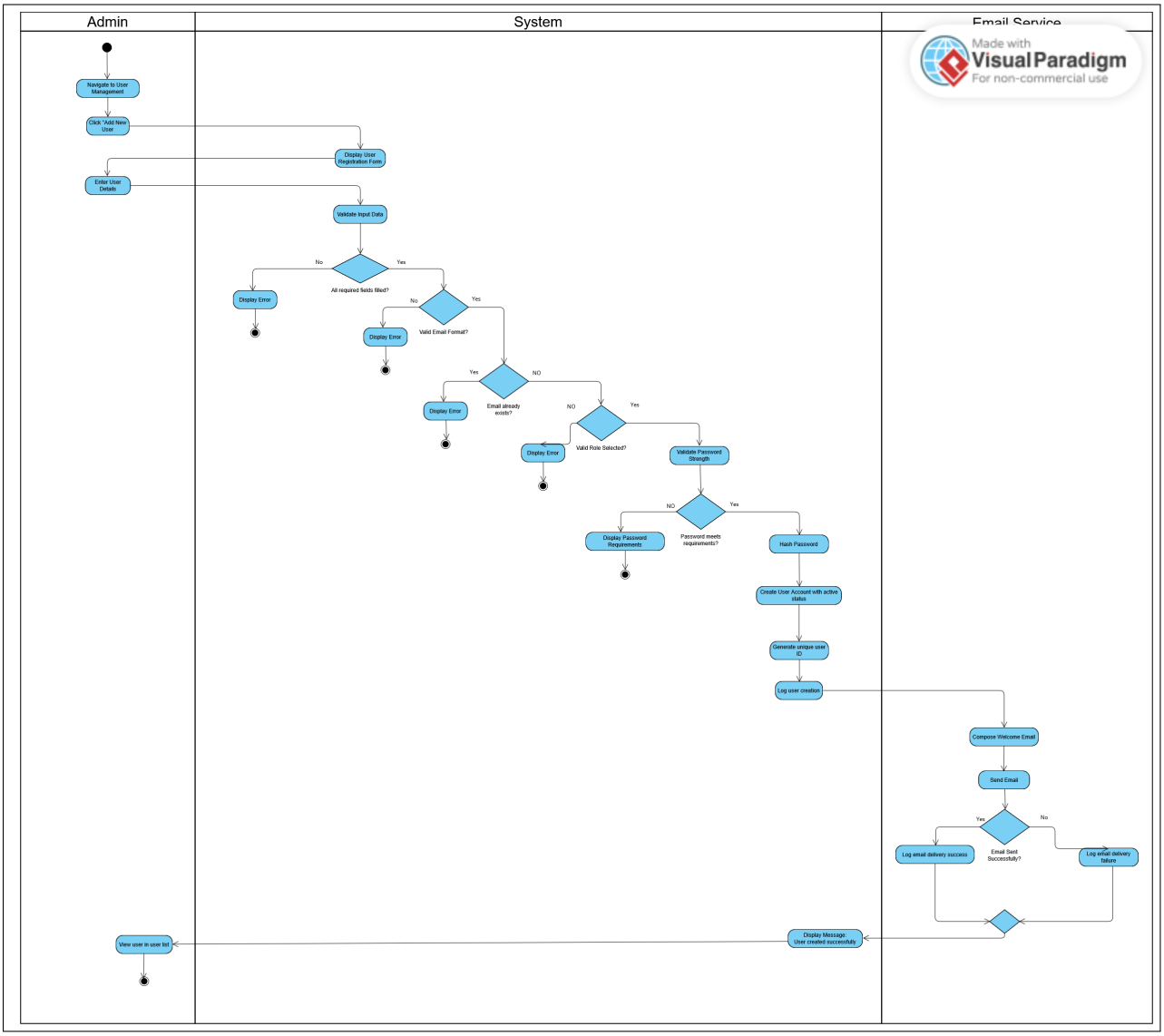



Figure 19: Activity Diagram - User Registration

### 4.4.3 Report Generation

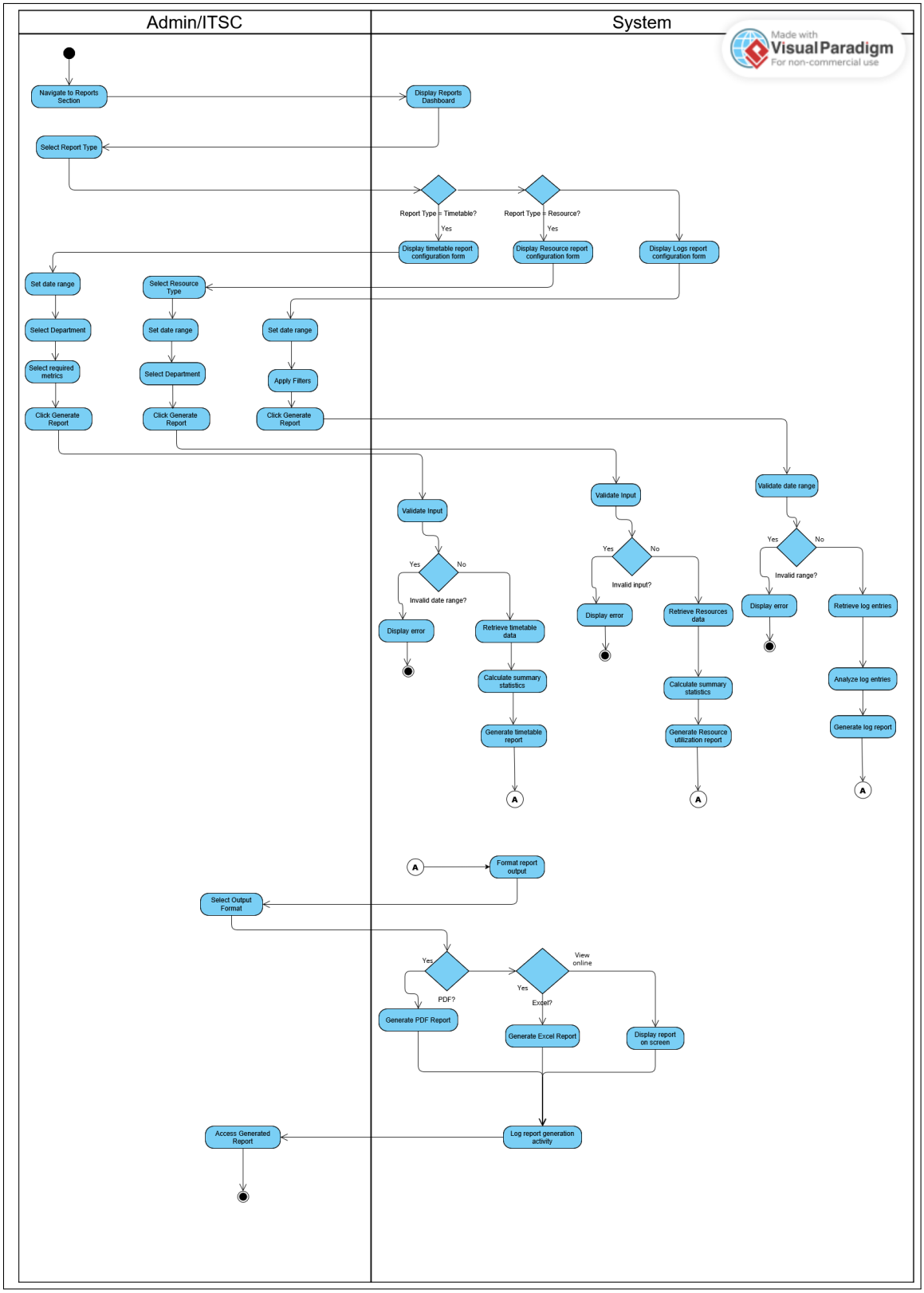



Figure 20: Activity Diagram - Report Generation

### 4.4.4 Bulk User Import

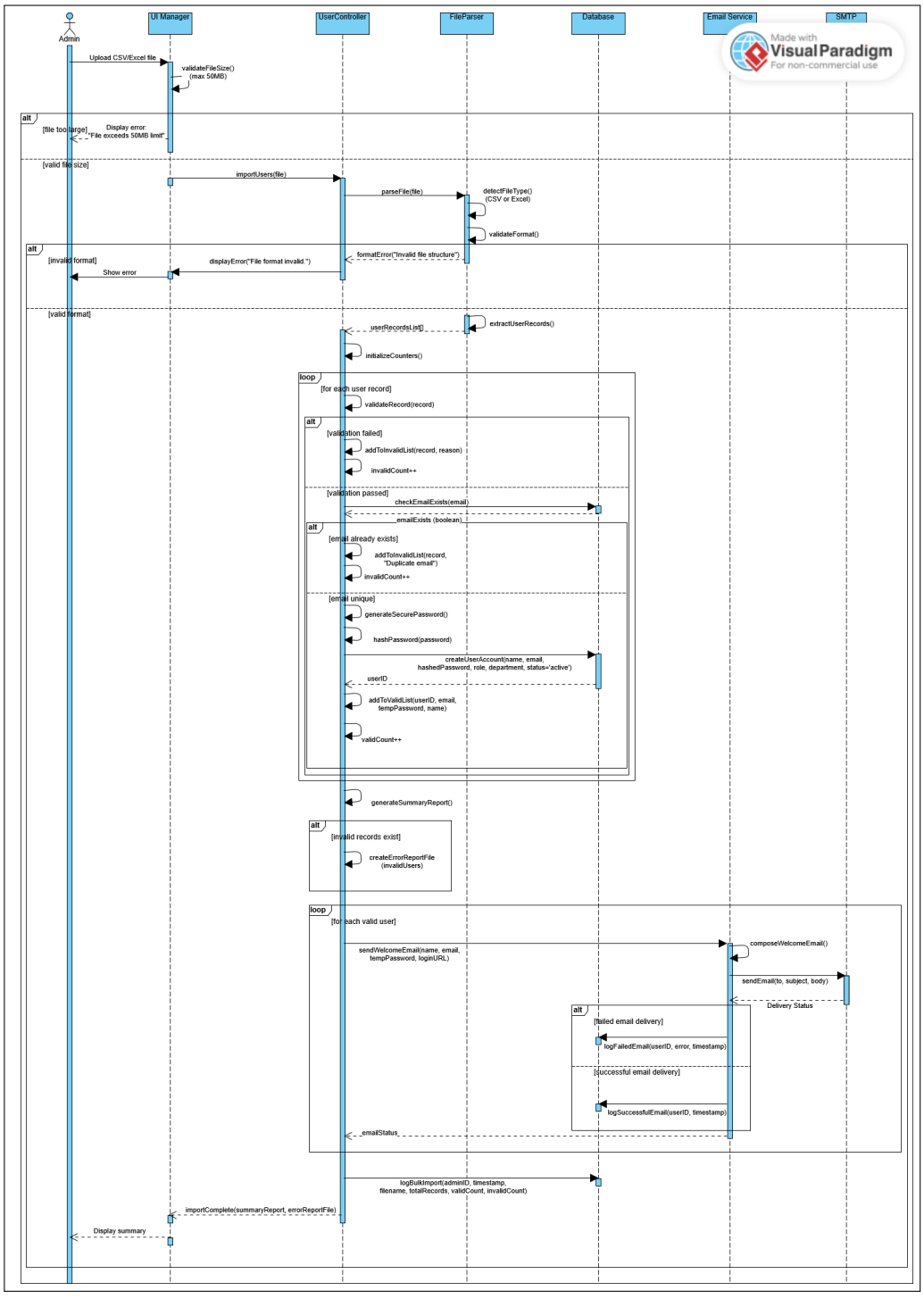



Figure 21: Activity Diagram - Bulk User Import

### 4.4.5 Password Reset

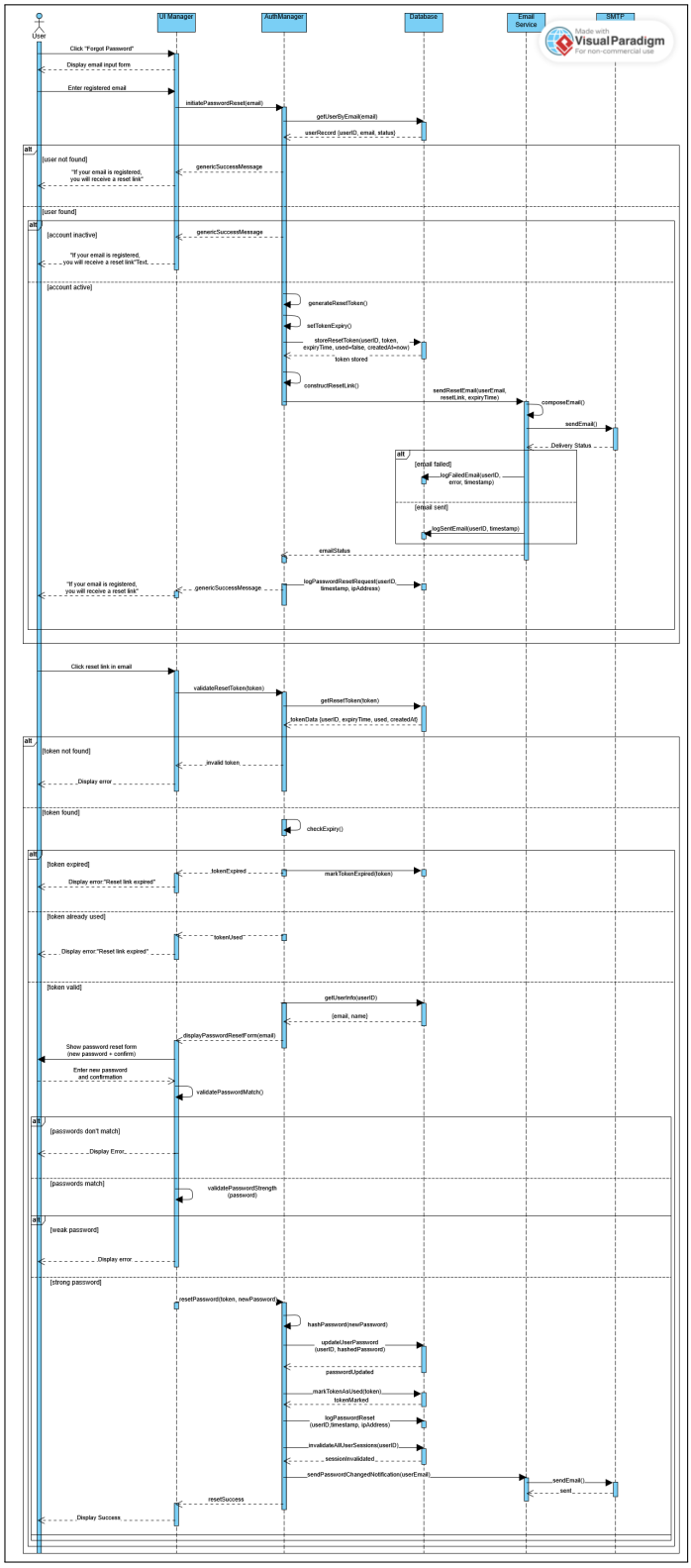



Figure 22: Activity Diagram - Password Reset

### 4.4.6 Conflict Detection and Resolution

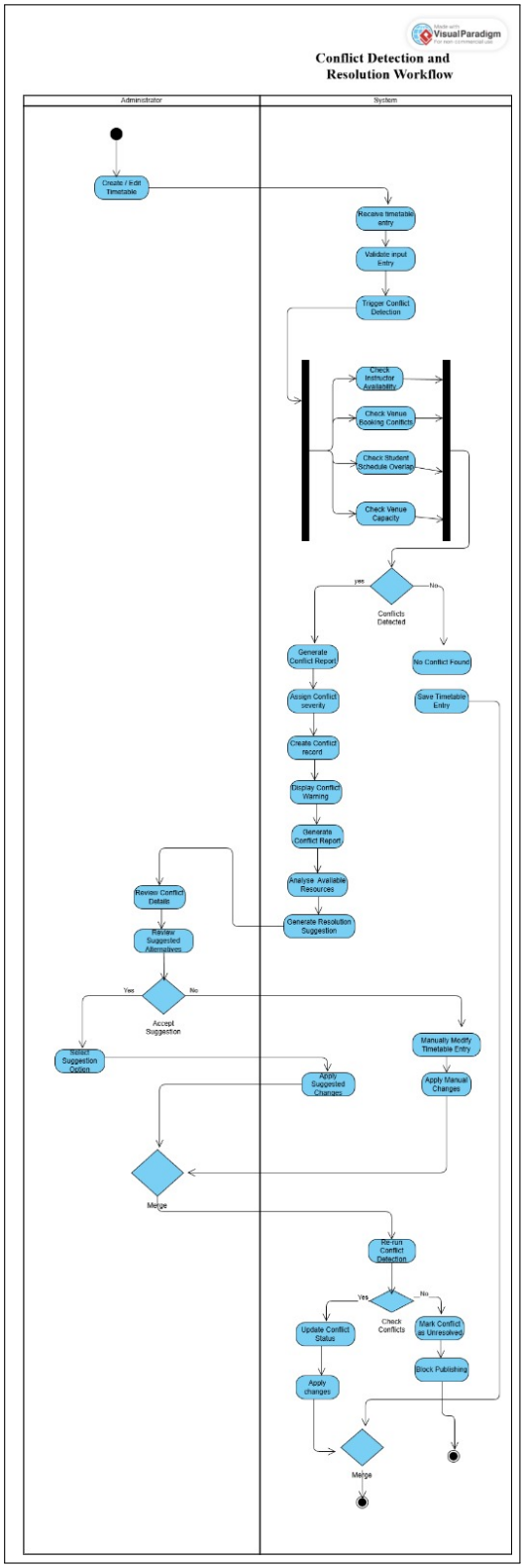



Figure 23: Activity Diagram - Conflict Detection and Resolution

### 4.4.7 Timetable Creation and Publishing

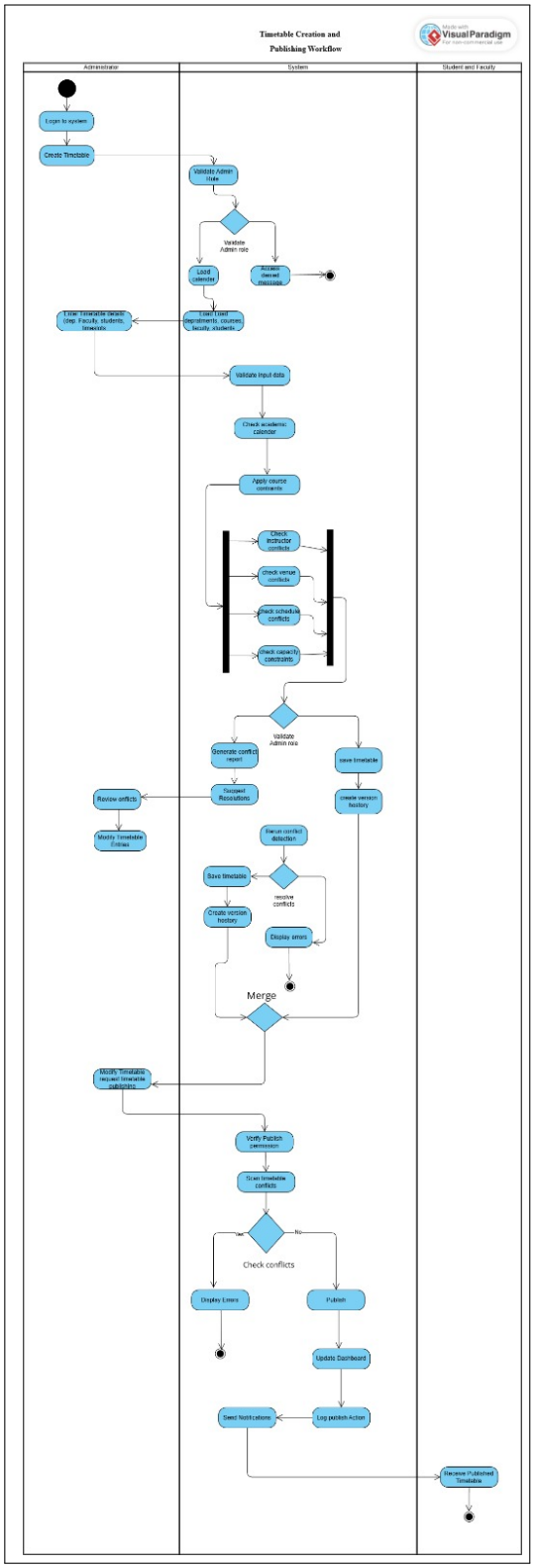



Figure 24: Activity Diagram - Timetable Creation and Publishing

### 4.4.8 Notification Delivery

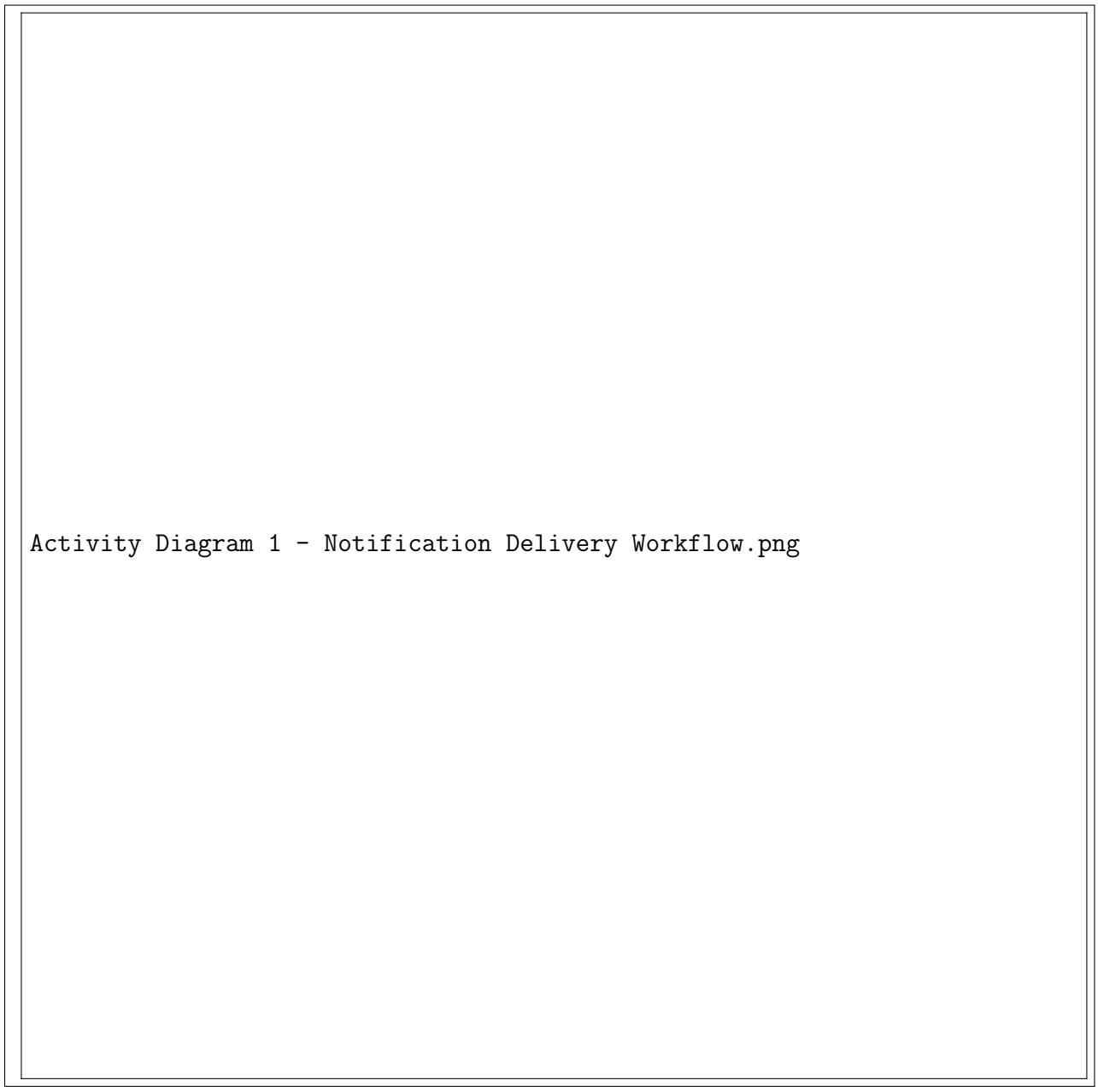



Figure 25: Activity Diagram - Notification Delivery

### 4.4.9 Auto-Timetable Generation

Figure 26: Activity Diagram - Auto-Timetable Generation

## 4.5 Class Diagram

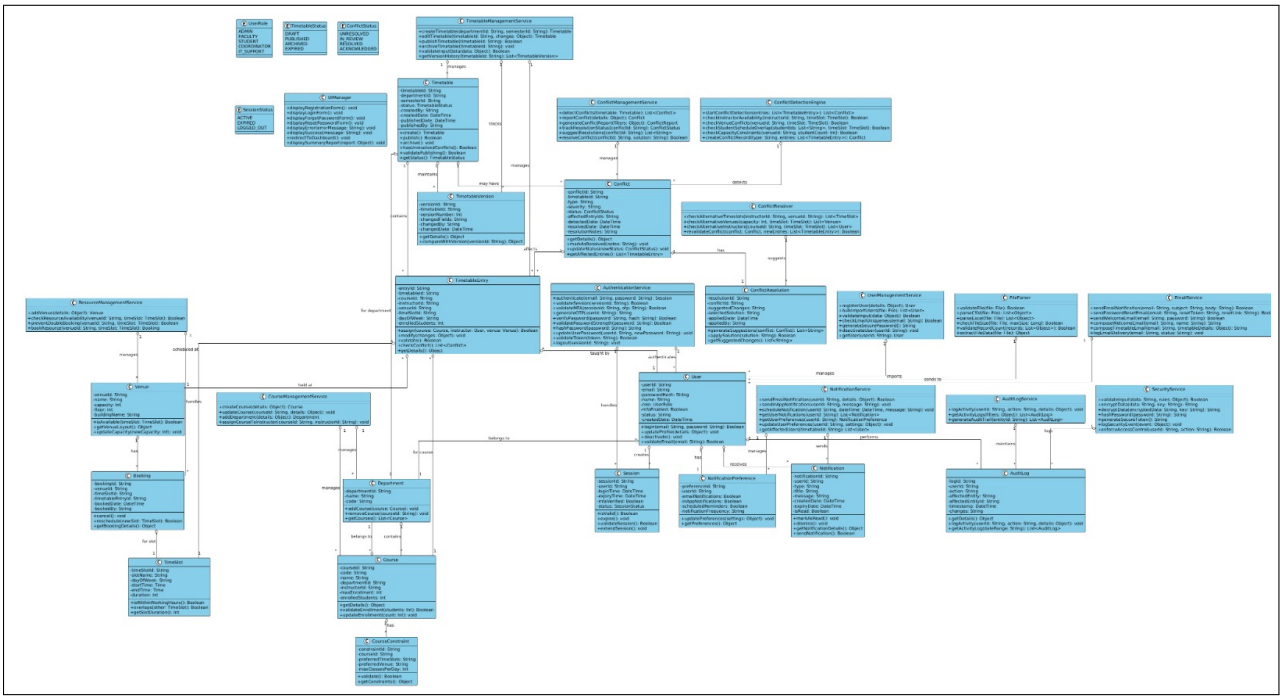


Figure 27: Class Diagram

## 4.6 Component Diagram

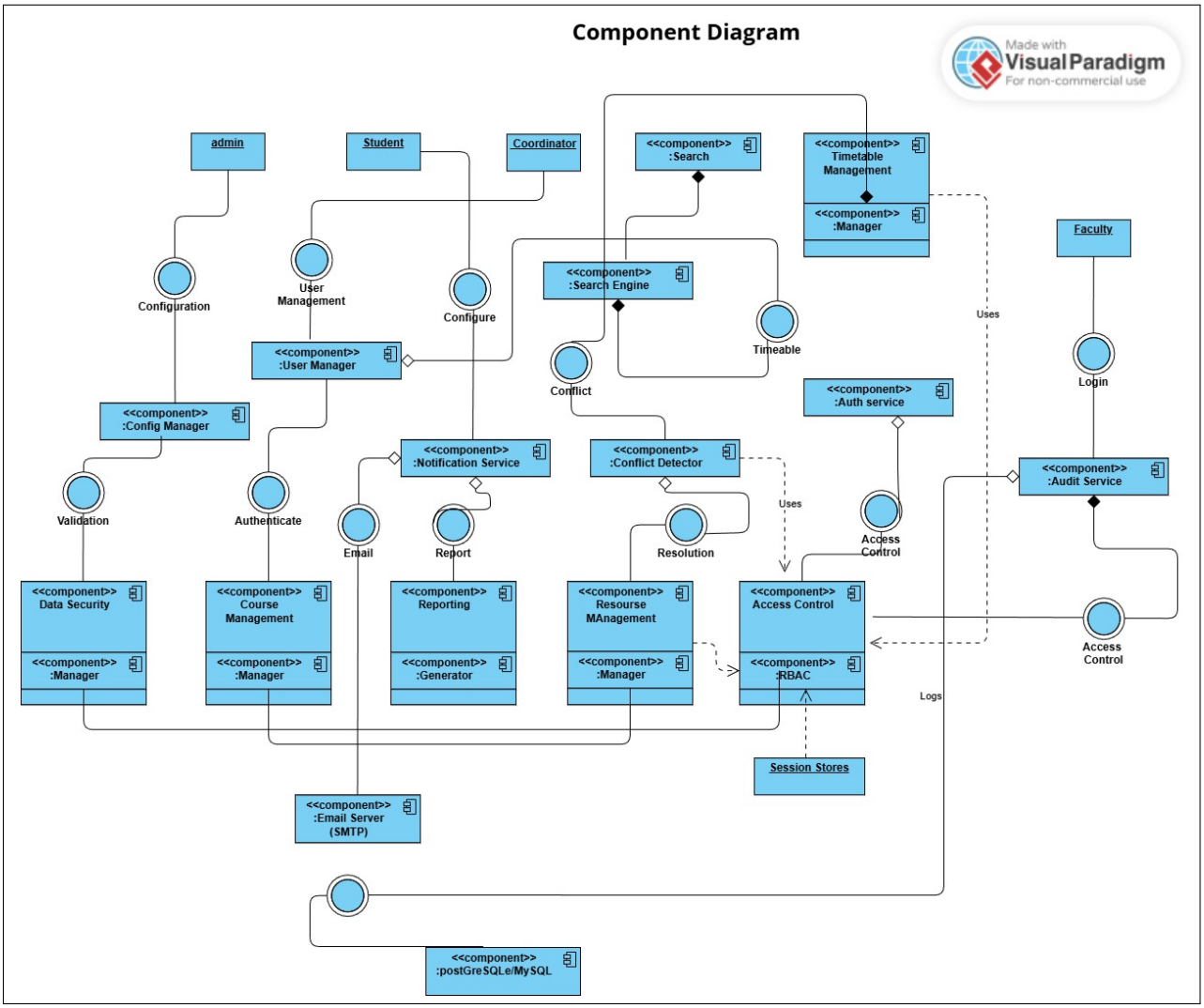



Figure 28: Component Diagram

# 5 Requirements-Design Traceability

## 5.1 Traceability Matrix

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR1 | Create Timetable | - Use Case: Create Timetable<br>- Context Diagram (Level 0)<br>- DFD Level 2: Timetable Management Module<br>- Sequence Diagram: Timetable Creation with Conflict Detection<br>- Activity Diagram: Timetable Creation and Publishing Workflow<br>- Classes: Timetable, TimetableManagementService, TimetableEntry, Course, User, Venue, TimeSlot, Department<br>- Components: Timetable Management |
| FR2 | Edit Timetable | - Use Case: Edit Timetable<br>- DFD Level 2: Timetable Management Module<br>- Activity Diagram: Timetable Creation and Publishing Workflow<br>- Classes: Timetable, TimetableManagementService, TimetableEntry, TimetableVersion<br>- Components: Timetable Management |
| FR3 | Delete Timetable | - Use Case: Delete Timetable<br>- DFD Level 2: Timetable Management Module<br>- Classes: Timetable, TimetableManagementService<br>- Components: Timetable Management |
| FR4 | Conflict Detection | - Use Case: Detect Conflicts<br>- Context Diagram (Level 0)<br>- DFD Level 2: Conflict Management Module<br>- DFD Level 2: Timetable Management Module<br>- Sequence Diagram: Timetable Creation with Conflict Detection<br>- Activity Diagram: Conflict Detection and Resolution Workflow<br>- Classes: Conflict, ConflictDetectionEngine, TimetableEntry, ConflictManagementService<br>- Components: Conflict Detector |

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR5 | Conflict Resolution Suggestions | - Use Case: Resolve Conflicts<br>- DFD Level 2: Conflict Management Module<br>- Sequence Diagram: Conflict Resolution with Suggestions<br>- Activity Diagram: Conflict Detection and Resolution Workflow<br>- Classes: ConflictResolution, ConflictResolver, ConflictManagementService<br>- Components: Resolution, Conflict Detector |
| FR6 | Publish Timetable | - Use Case: Publish Timetable<br>- DFD Level 2: Timetable Management Module<br>- Sequence Diagram: Timetable Publishing<br>- Activity Diagram: Timetable Creation and Publishing Workflow<br>- Classes: Timetable, TimetableManagementService, NotificationService<br>- Components: Timetable Management |
| FR7 | Archive Timetable | - Use Case: Archive Timetable<br>- DFD Level 2: Timetable Management Module<br>- Classes: Timetable, TimetableManagementService<br>- Components: Timetable Management |
| FR8 | Import and Export Timetable | - Use Case: Import/Export Timetable<br>- DFD Level 2: Timetable Management Module<br>- Classes: FileParser, ExportService, Report<br>- Components: Timetable Management |
| FR9 | Bulk Operations | - Use Case: Edit Timetable<br>- DFD Level 2: Timetable Management Module<br>- Activity Diagram: Timetable Creation and Publishing Workflow<br>- Classes: TimetableEntry, TimetableManagementService<br>- Components: Timetable Management |
| FR10 | Timetable Versioning | - DFD Level 2: Timetable Management Module<br>- Classes: TimetableVersion, TimetableManagementService<br>- Components: Timetable Management |
| FR11 | Conflict Report | - Use Case: View Conflict Report<br>- DFD Level 2: Conflict Management Module<br>- DFD Level 2: Timetable Management Module<br>- Activity Diagram: Report Generation Workflow<br>- Classes: ConflictReport, ReportingService, ConflictManagementService<br>- Components: Reporting |

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR12 | Track Resolution Status | - Use Case: View Conflict Report<br>- DFD Level 2: Conflict Management Module<br>- Activity Diagram: Conflict Detection and Resolution Workflow<br>- Classes: Conflict, ConflictManagementService<br>- Components: Conflict Detector |
| FR13 | Automated Conflict Resolution | - Use Case: Resolve Conflicts<br>- DFD Level 2: Conflict Management Module<br>- Sequence Diagram: Conflict Resolution with Suggestions<br>- Activity Diagram: Conflict Detection and Resolution Workflow<br>- Classes: ConflictResolution, ConflictResolver, ConflictManagementService<br>- Components: Resolution, Conflict Detector |
| FR14 | User Registration | - Use Case: Manage Users<br>- Context Diagram (Level 0)<br>- Activity Diagram: User Registration Workflow<br>- Classes: User, UserManagementService, AuthenticationService<br>- Components: User Manager |
| FR15 | User Login and Authentication | - Use Case: Login<br>- Context Diagram (Level 0)<br>- Sequence Diagram: User Login<br>- Activity Diagram: User Login Workflow<br>- Classes: User, AuthenticationService, Session<br>- Components: Auth Service |
| FR16 | Role-Based Access Control | - Use Case: Login<br>- Context Diagram (Level 0)<br>- Classes: User, AuthenticationService, SecurityService<br>- Components: RBAC, Access Control |
| FR17 | User Profile Management | - Use Case: Update Profile<br>- Classes: User, UserManagementService<br>- Components: User Manager |
| FR18 | Password Reset | - Use Case: Login<br>- Sequence Diagram: Password Reset<br>- Activity Diagram: Password Reset Workflow<br>- Classes: AuthenticationService, EmailService<br>- Components: Auth Service, Email Server |

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR19 | Bulk User Import | - Use Case: Manage Users<br>- Sequence Diagram: Bulk User Import<br>- Activity Diagram: Bulk User Import Workflow<br>- Classes: UserManagementService, FileParser, User, EmailService<br>- Components: User Manager |
| FR20 | User Deactivation | - Use Case: Manage Users<br>- Classes: User, UserManagementService<br>- Components: User Manager |
| FR21 | Multi-Factor Authentication | - Use Case: Login<br>- Sequence Diagram: User Login<br>- Activity Diagram: User Login Workflow<br>- Classes: AuthenticationService, Session<br>- Components: Auth Service |
| FR22 | Email Notifications | - Use Case: Send Notifications<br>- Context Diagram (Level 0)<br>- DFD Level 2: Notification and Communication Process<br>- Sequence Diagram: Notification and Communication<br>- Activity Diagram: Notification Delivery Workflow<br>- Classes: Notification, EmailService, NotificationService<br>- Components: Notification Service, Email Server |
| FR23 | In-App Notifications | - Use Case: View Notifications<br>- DFD Level 2: Notification and Communication Process<br>- Sequence Diagram: Notification and Communication<br>- Activity Diagram: Notification Delivery Workflow<br>- Classes: Notification, NotificationService<br>- Components: Notification Service |
| FR24 | Scheduled Notifications | - Use Case: Send Notifications<br>- DFD Level 2: Notification and Communication Process<br>- Activity Diagram: Notification Delivery Workflow<br>- Classes: Notification, NotificationService<br>- Components: Notification Service |
| FR25 | Notification Preferences | - Use Case: Update Profile<br>- DFD Level 2: Notification and Communication Process<br>- Classes: NotificationPreference, NotificationService<br>- Components: Notification Service |

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR27 | Timetable Summary Report | - Use Case: Generate Reports<br>- Context Diagram (Level 0)<br>- Activity Diagram: Report Generation Workflow<br>- Classes: TimetableSummaryReport, ReportingService, ReportGenerator, Report<br>- Components: Reporting |
| FR28 | Resource Utilization Report | - Use Case: Generate Reports<br>- Activity Diagram: Report Generation Workflow<br>- Classes: ResourceUtilizationReport, ReportingService, ReportGenerator, Report<br>- Components: Reporting |
| FR29 | Audit Log Report | - Use Case: Generate Reports<br>- Context Diagram (Level 0)<br>- Activity Diagram: Report Generation Workflow<br>- Classes: AuditLogReport, AuditLog, AuditLogService, ReportingService<br>- Components: Audit Service |
| FR30 | Venue and Classroom Management | - Use Case: Configure System<br>- Context Diagram (Level 0)<br>- Classes: Venue, ResourceManagementService, Booking<br>- Components: Resource Management |
| FR31 | Lab Management | - Use Case: Configure System<br>- Classes: Venue, ResourceManagementService, Booking<br>- Components: Resource Management |
| FR32 | Resource Constraints | - Use Case: Configure System<br>- Classes: ConflictDetectionEngine, CourseConstraint, TimetableEntry<br>- Components: Resource Management |
| FR33 | Resource Conflict Detection | - Use Case: Detect Conflicts<br>- DFD Level 2: Conflict Management Module<br>- Activity Diagram: Conflict Detection and Resolution Workflow<br>- Classes: ConflictDetectionEngine, ResourceManagementService, Booking<br>- Components: Conflict Detector, Resource Management |
| FR34 | Course Registry | - Use Case: Manage Courses<br>- Context Diagram (Level 0)<br>- Classes: Course, CourseManagementService, Department<br>- Components: Course Management |

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR35 | Department Management | - Use Case: Manage Departments<br>- Context Diagram (Level 0)<br>- Classes: Department, CourseManagementService<br>- Components: Course Management |
| FR36 | Course Constraints | - Use Case: Configure System<br>- Classes: CourseConstraint, Course<br>- Components: Course Management |
| FR37 | Admin Dashboard | - Use Case: Generate Reports, Manage Users, Manage Departments, Manage Courses, Configure System<br>- Context Diagram (Level 0)<br>- DFD Level 2: Dashboard<br>- Classes: Dashboard, AdminDashboard, DashboardService<br>- Components: Search, Search Engine |
| FR38 | Faculty Dashboard | - Use Case: View Timetable, View Notifications, Update Profile<br>- Context Diagram (Level 0)<br>- DFD Level 2: Dashboard<br>- Classes: Dashboard, FacultyDashboard, DashboardService<br>- Components: Search, Search Engine |
| FR39 | Student Dashboard | - Use Case: View Timetable, View Notifications, Update Profile<br>- Context Diagram (Level 0)<br>- DFD Level 2: Dashboard<br>- Classes: Dashboard, StudentDashboard, DashboardService, ExportService<br>- Components: Search, Search Engine |
| FR40 | Filter and Sort | - Use Case: View Timetable<br>- DFD Level 2: Dashboard<br>- Classes: DashboardService<br>- Components: Search, Search Engine |
| FR41 | Academic Calendar Setup | - Use Case: Configure System<br>- Context Diagram (Level 0)<br>- DFD Level 2: System Configuration<br>- Classes: TimeSlot<br>- Components: Config Manager |
| FR42 | Time Slot Configuration | - Use Case: Configure System<br>- DFD Level 2: System Configuration<br>- Classes: TimeSlot<br>- Components: Config Manager |

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR43 | System Settings | - Use Case: Configure System<br>- DFD Level 2: System Configuration<br>- Classes: NotificationService, EmailService<br>- Components: Config Manager |
| FR44 | Constraint Configuration | - Use Case: Configure System<br>- DFD Level 2: System Configuration<br>- Classes: CourseConstraint, ConflictDetectionEngine<br>- Components: Config Manager |
| FR45 | Data Validation | - Context Diagram (Level 0)<br>- DFD Level 2: Timetable Management Module<br>- DFD Level 2: Conflict Management Module<br>- DFD Level 2: System Configuration<br>- Classes: TimetableManagementService, UserManagementService, FileParser, SecurityService, TimetableEntry<br>- Components: Validation |
| FR46 | Data Consistency | - Context Diagram (Level 0)<br>- Classes: Course, Timetable<br>- Components: PostgreSQL/MySQL |
| FR47 | Access Control | - Use Case: All system operations<br>- Context Diagram (Level 0)<br>- Sequence Diagram: User Login<br>- Classes: User, AuthenticationService, SecurityService<br>- Components: Access Control, RBAC |
| FR48 | Data Backup | - Use Case: Backup Data<br>- Context Diagram (Level 0)<br>- Classes: SecurityService<br>- Components: Data Security, Timetable Management |
| FR49 | Automatic Timetable Generation | - Use Case: Generate Timetable (Create Timetable)<br>- DFD Level 2: Timetable Management Module<br>- Sequence Diagram: Automatic Timetable Generation<br>- Activity Diagram: Auto Timetable Generation Workflow<br>- Classes: Timetable, TimetableEntry, TimetableManagementService, Course, User, Venue, TimeSlot, CourseConstraint, ConflictDetectionEngine<br>- Components: Generator |

| Req ID | Requirement | Design Artifacts |
|---|---|---|
| FR50 | Draft Review and Approval | - Use Case: Validate Timetable<br>- DFD Level 2: Timetable Management Module<br>- Sequence Diagram: Timetable Publishing<br>- Activity Diagram: Timetable Creation and Publishing Workflow<br>- Classes: Timetable, TimetableVersion, TimetableManagementService, ConflictDetectionEngine<br>- Components: Timetable Management |

### 5.2 Coverage Summary

The traceability matrix confirms that all functional and non-functional requirements defined in the SRS are addressed by corresponding design components, ensuring complete coverage.

## 6 Figma Link

Here is the link to Figma:

`https://www.figma.com/proto/MWsrDnsdrzYqJAjm4uXEGP/UTMS?node-id=32-6589&t=b2sLpg9edA9X8euu-1`

## 7 GitHub Repository Link

Here is the link to repository:

`https://github.com/hammad-013/universty-timetable-management-system`

## 8 Conclusion

This Software Design Specification has been effective in converting the SRS requirements into a detailed technical outline of the University Timetable Management System. The three level architecture, detailed diagrams and requirements traceability matrix give a clear base to be used in implementation. Having all the requirements (functional and non-functional) satisfied and confirmed by the prototype, the given design document will create an easy way to develop the system.

# A Glossary

- SRS: Software Requirements Specification
- RBAC: Role-Based Access Control
- DFD: Data Flow Diagram

# B Design Tools Used

- Figma
- Draw.io
- Visual Paradigm
- PlantUML