Software Requirements Specification

for

Lightweight Education Data Bay Area

Version 1.4 approved

Prepared by Sun Nengke, Lyu Luyao, Lin Zhengjun, Li Ruilin,

Liang Jiayu, Wang Jiahui (All full)

AO3

2025/3/19

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
ALL	2025/2/24		1.0
ALL	2025/3/3	Fix font and appendix	1.1
ALL	2025/3/5	Add section3 and 4.1	1.2
ALL	2025/3/10	Revise and Complete UI and state transition diagram (for all user case)	1.3
ALL	2025/3/19	Add Appendix B	1.4

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to define the requirements for the **Design and Implementation of a Lightweight Education Data Bay Area (E-DBA)**. This system aims to provide a secure and interoperable platform for data sharing among registered higher education organizations. The system will facilitate student identity authentication, thesis sharing, course information sharing, and online payment services. This document outlines the functional and non-functional requirements, user roles, and system features necessary for the successful implementation of the E-DBA.

1.2 Document Conventions

This document follows the IEEE 830 standard for Software Requirements Specifications. Key terms and phrases are highlighted in **bold** for emphasis. Requirements are uniquely identified with a sequence number (e.g., REQ-1, REQ-2). Priorities for requirements are indicated as High, Medium, or Low. This article will focus on bold and bright.

1.3 Intended Audience and Reading Suggestions

This document is intended for the following stakeholders involved in the development and deployment of the Education Data Bay Area (E-DBA) system:

1. **Developers:**
   Developers will use this document to understand the system's functional and non-functional requirements, design the software architecture, and implement the system features. They should focus on **Section 3: System Features** and **Section 4: External Interface Requirements** to gain detailed insights into the system's behavior and interfaces.
2. **Lightweight Education Data Bay Area Managers:**
   Lightweight Education Data Bay Area managers will use this document to plan and monitor the Lightweight Education Data Bay Area's progress, allocate resources, and ensure that the system meets the specified requirements. They should review **Section 1:**

Introduction and **Section 2: Overall Description** to understand the Lightweight Education Data Bay Area's scope and objectives, and then refer to **Section 3: System Features** for detailed feature descriptions.

3. **Testers:**
   Testers will use this document to create test cases and validate that the system meets the specified requirements. They should focus on **Section 3: System Features** and **Section 5: Other Nonfunctional Requirements** to identify the functional and performance requirements that need to be tested.
4. **System Administrators:**
   System administrators will use this document to understand the system's operational requirements, including user roles, access controls, and security measures. They should review **Section 2.3: User Classes and Characteristics** and **Section 5.3: Security Requirements** to ensure proper system configuration and maintenance.
5. **Stakeholders (E-Admins, O-Conveners, and End Users):**
   Stakeholders, including management administrators (E-Admins), organization conveners (O-Conveners), and end users (data providers and consumers), will use this document to understand the system's capabilities and how it supports their roles. They should focus on **Section 2.3: User Classes and Characteristics** and **Section 3: System Features** to understand how the system will meet their needs.

6. **Reading Suggestions:**

- **For a high-level overview:** Start with **Section 1: Introduction** and **Section 2: Overall Description.**
- **For detailed requirements:** Proceed to **Section 3: System Features** and **Section 4: External Interface Requirements.**
- **For operational and security considerations:** Review **Section 5: Other Nonfunctional Requirements.**

1.4 Lightweight Education Data Bay Area Scope

The E-DBA system is designed to enable secure and transparent data sharing among higher education organizations while adhering to corporate policies and data sovereignty regulations. The system will provide the following key functionalities:

1. **Data Provision and Consumption**: A secure platform for sharing and accessing data among registered organizations.
2. **Student Identity Authentication**: Verification of student identities for enrolled students and graduates.
3. **Thesis Sharing**: Controlled access to student theses based on access rights set by the providing organization.
4. **Course Information Sharing**: Public access to course information provided by participating organizations.
5. **Online Payment**: Support for multiple payment methods for service fees and membership charges.
6. **Data Vault Service (Optional)**: A secure storage service for organizations that cannot maintain their own databases.

The system will be implemented as a lightweight, scalable solution that can be easily integrated with existing systems in higher education organizations.

1.5 References

The document: SE_SDW_Project Long Description V1.3.docx. SE_SDW_Project Long Description V1.6.docx.

2. Overall Description

2.1 Product Perspective

E-DBA is a self-contained system designed to facilitate data sharing among higher education institutions. It integrates with external databases, including student information, thesis repositories**,** and bank accounts, but operates independently from existing institutional systems. The system provides secure student identity authentication, thesis access, and online payment services.

Additionally, an optional data vault service is available for institutions that require data storage and controlled sharing capabilities.

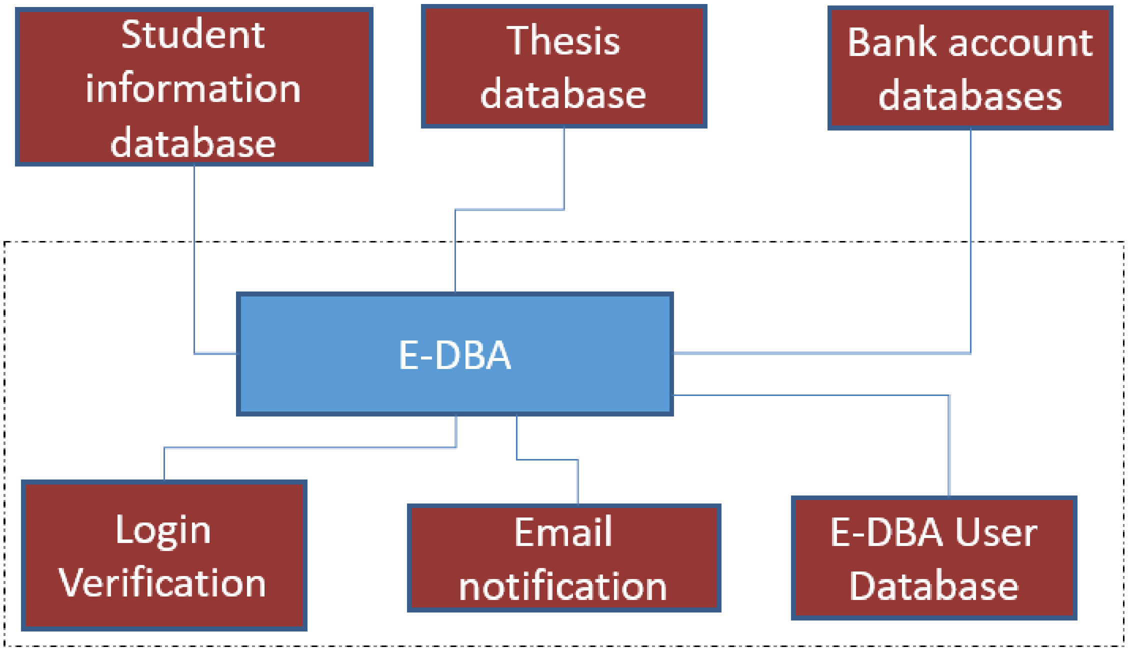


Figure 1. Context model

2.2 Product Features

The E-DBA system is designed to offer a secure, efficient, and user-friendly platform for **data sharing** among organizations. It enables **role-based** access control, ensuring that only authorized institutions and users can register, manage, and consume data. The system provides dedicated **workspaces for each organization**, allowing them to **manage members**, **configure services**, and **monitor activity logs**. E-DBA supports both **public** and **private data** access. Additionally, the platform integrates a **payment system** that supports bank transfers. An optional data **vault service** is available for organizations that require secure storage.

Use case diagram

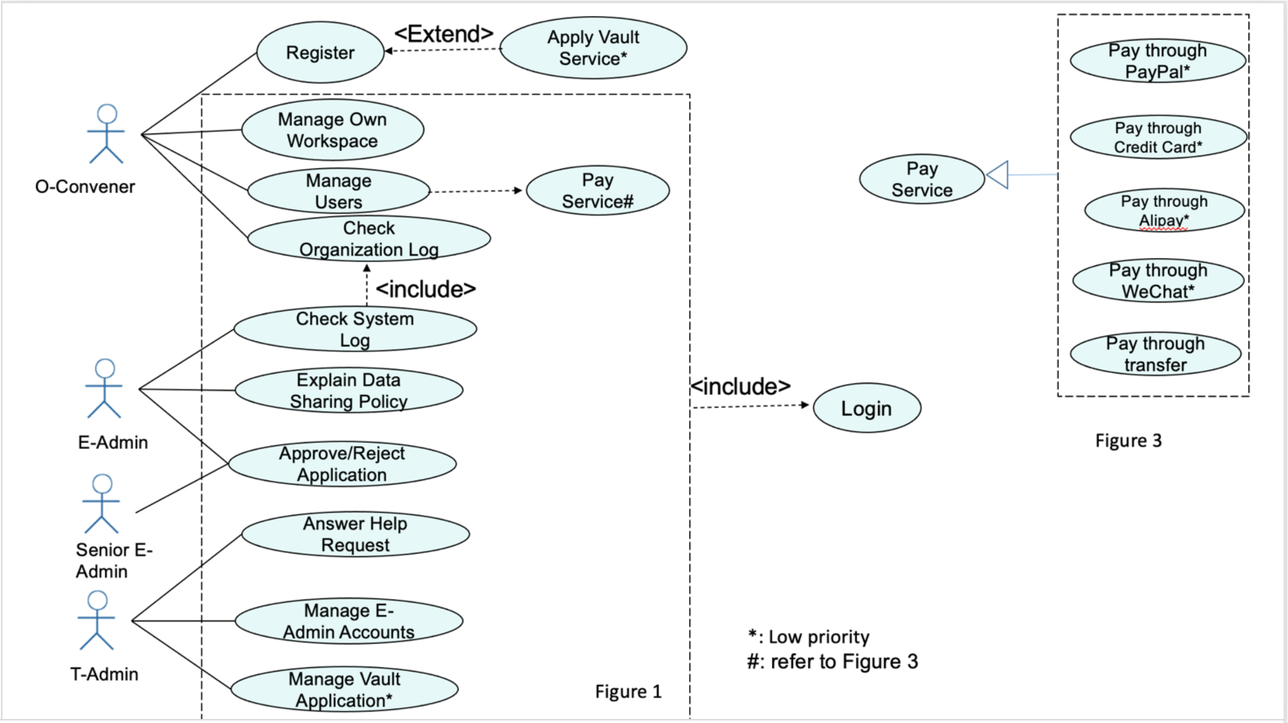


Figure 1

Figure 3

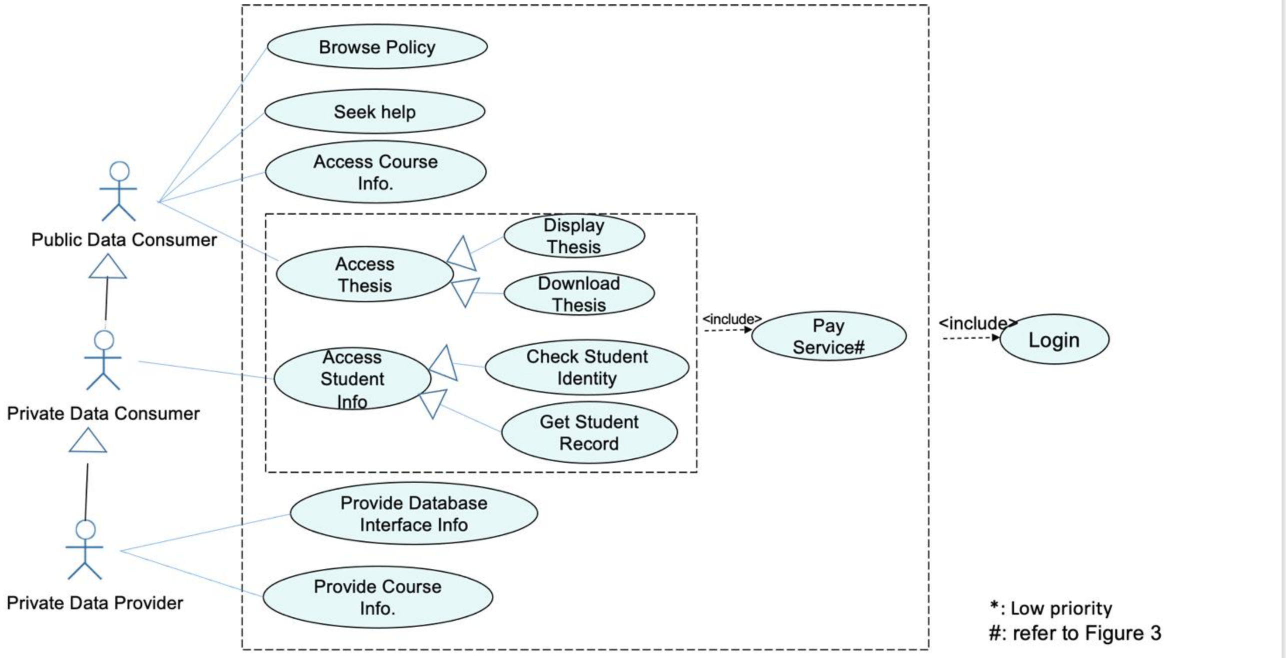


Figure 2

Basic Scenario

1. Access the Registration Page
   - The O-Convener opens the E-DBA system's public "Register" page.
2. Submit Application
   - The O-Convener fills in all required registration details: organization name, proof documents, convener's name, and a valid email address.
   - The O-Convener then clicks "Submit" to send the application.

3. System Validates and Saves
    - The system checks if all required fields are filled and formats are correct.
    - If the information is valid, the system saves the application in the "pending approval" queue.
    - If invalid, the system prompts the O-Convener to correct the data before resubmitting.
4. Waiting for Approval
    - The system displays a message indicating the application is under review.
5. E-Admin & Senior E-Admin Approval
    - The system forwards the application to the E-Admin for an initial decision (approve or reject).
    - If E-Admin approves, the request proceeds to Senior E-Admin for final approval. If E-Admin rejects, the process ends, and the O-Convener is notified of the rejection.
    - If Senior E-Admin gives final approval, the registration completes successfully. If Senior E-Admin rejects, the process ends, and the O-Convener is notified of the rejection.
6. Registration Confirmation
    - Once approved by both E-Admin and Senior E-Admin, the system sends a confirmation email to the O-Convener's registered email address.
    - The email includes login instructions or a temporary code, allowing the O-Convener to log in and manage the organization's workspace.

Alternative Scenarios

1. Incomplete or Invalid Information
    - At Step 3, if the system detects missing or malformed data, it rejects the submission and prompts the O-Convener to correct the form.
    - The O-Convener may revise and resubmit, or abandon the process.
2. Approval Rejected by E-Admin or Senior E-Admin
    - At Step 5, if E-Admin or Senior E-Admin rejects the application, the system notifies the O-Convener that registration has failed. The process terminates.
3. System or Network Error
    - At any step, if a system or network error occurs, the O-Convener may be unable to submit or proceed. The system displays an error message, and the user can retry once service is restored.

2.3 User Classes and Characteristics

Primary Users (High Priority):

1. **Technical Administrator (T-Admin)**

Responsibilities:

- Responsible for system maintenance and technical support.
- Assists other users with system-related issues.
- Configures and manages Management Administrators (E-Admins).

Usage Habits and Characteristics:

- **Usage Frequency:** High, as they must be ready to respond to system failures and maintenance requests at any time.
- **Technical Background:** Usually well-versed in IT, with experience in system deployment and security strategies.

- **Education Level:** Typically hold specialized or equivalent qualifications; proficient with system administration tools and command-line usage.

2. **Management Administrators (E-Admin, Senior E-Admin)**
   Responsibilities:

- **E-Admin:** Reviews, approves, or rejects institutional registration applications.
- **Senior E-Admin:** Provides final approval for registrations after E-Admin's review.
- Oversees data-sharing policies and can access all organizational logs.

Usage Habits and Characteristics:

- **Usage Frequency:** Medium to high, as they periodically audit and review system logs and handle approval records.
- **Technical Background:** Familiar with system administration and regulatory policies; capable of conducting compliance checks.
- **Education Level:** Often hold higher education degrees in management or technology; knowledgeable about business processes and regulatory requirements.

3. **Organization Convener (O-Convener)**
   Responsibilities:

- Acts as the liaison between their institution and E-DBA.
- Submits registration applications on behalf of their institution.
- Manages the list of authorized users within the institution.
- Controls the configuration and availability of institutional services.

Usage Habits and Characteristics:

- **Usage Frequency:** Moderate, mainly when adding or modifying user permissions and service configurations.
- **Technical Background:** May have some technical or managerial knowledge, but not necessarily deeply familiar with system internals; primarily concerned with business-level operational convenience.
- **Education Level:** Typically hold at least a bachelor's degree; well-versed in institutional processes and possess some management experience.

Secondary Users (Lower Priority):

4. **Regular Users (Students/Faculty)**
   Responsibilities:

- Access rights are assigned by the O-Convener.
- May assume one or more of the following roles:
  - Public Data Access (e.g., viewing general course information).

- Private Data Provider (e.g., uploading thesis documents, verifying student identities).
- Private Data Consumer (e.g., accessing GPA records, retrieving thesis files).

Usage Habits and Characteristics:

- Usage Frequency: Generally high, since students/faculty often need to access course materials or upload/download academic resources.
- Technical Background: Students/faculty tend to be comfortable with digital platforms. They expect a user-friendly, intuitive interface and straightforward operations.
- Education Level: Typically, college-level or above, placing high demands on the system's usability and interface clarity.

2.4 Operating Environment

Hardware Requirements

- Accessible on modern computers and mobile devices.

Operating System Compatibility

- Cross-platform support, including Windows, macOS, and Linux.

Software Requirements

- Runs in a web browser, with no additional software installation required.

2.5 Design and Implementation Constraints

Security:

- The system must enforce strict data access control, ensuring that only authorized users can access specific data.
- User activity logs must be automatically recorded and cannot be modified.
- Administrators have access to organization-level system logs for monitoring purposes.

Payment Integration:

- The system must support bank transfers as the default payment method.
- Additional payment methods, including PayPal, WeChat Pay, Alipay, and credit cards, are optional.
- All payment transactions rely on external bank account verification and transfer processing interfaces.

Scalability:

- E-DBA must be designed to handle a large number of users and databases efficiently.
- The system should support high-volume data access and transactions without performance degradation.

2.6 User Documentation

User Manual:

- Detailed instructions on registration, login, user role management, and data access permissions.
- Guidelines on using system features, managing organizational workspaces, and configuring services.

Online Help:

- In-system contextual assistance for navigation and troubleshooting.
- Frequently Asked Questions (FAQs) covering registration, data access, and payment processes.

Tutorials:

- Step-by-step guides on key functions such as registering as an organization, managing user lists, and configuring access rights.
- Tutorials tailored for O-Conveners, administrators, and regular users, explaining their specific tasks and responsibilities.

2.7 Assumptions and Dependencies

Assumptions:

- External Databases Availability: The system assumes that student information databases and thesis databases are accessible via API connections. Institutions providing student authentication and thesis-sharing services must ensure their databases are available and properly configured.
- Internet-Enabled Devices: Users must have access to computers or mobile devices with an internet connection to utilize E-DBA's services.

Dependencies:

- Payment Gateway Integration: The system relies on external payment providers such as PayPal, WeChat Pay, Alipay, and banking services for transactions. Payment processing depends on bank account database interfaces for account verification and fund transfers.
- Institutional User Management: Organizations are responsible for maintaining user lists, managing access rights, and configuring services according to their data-sharing policies.

3. System Features

3.1 Login and all admin page

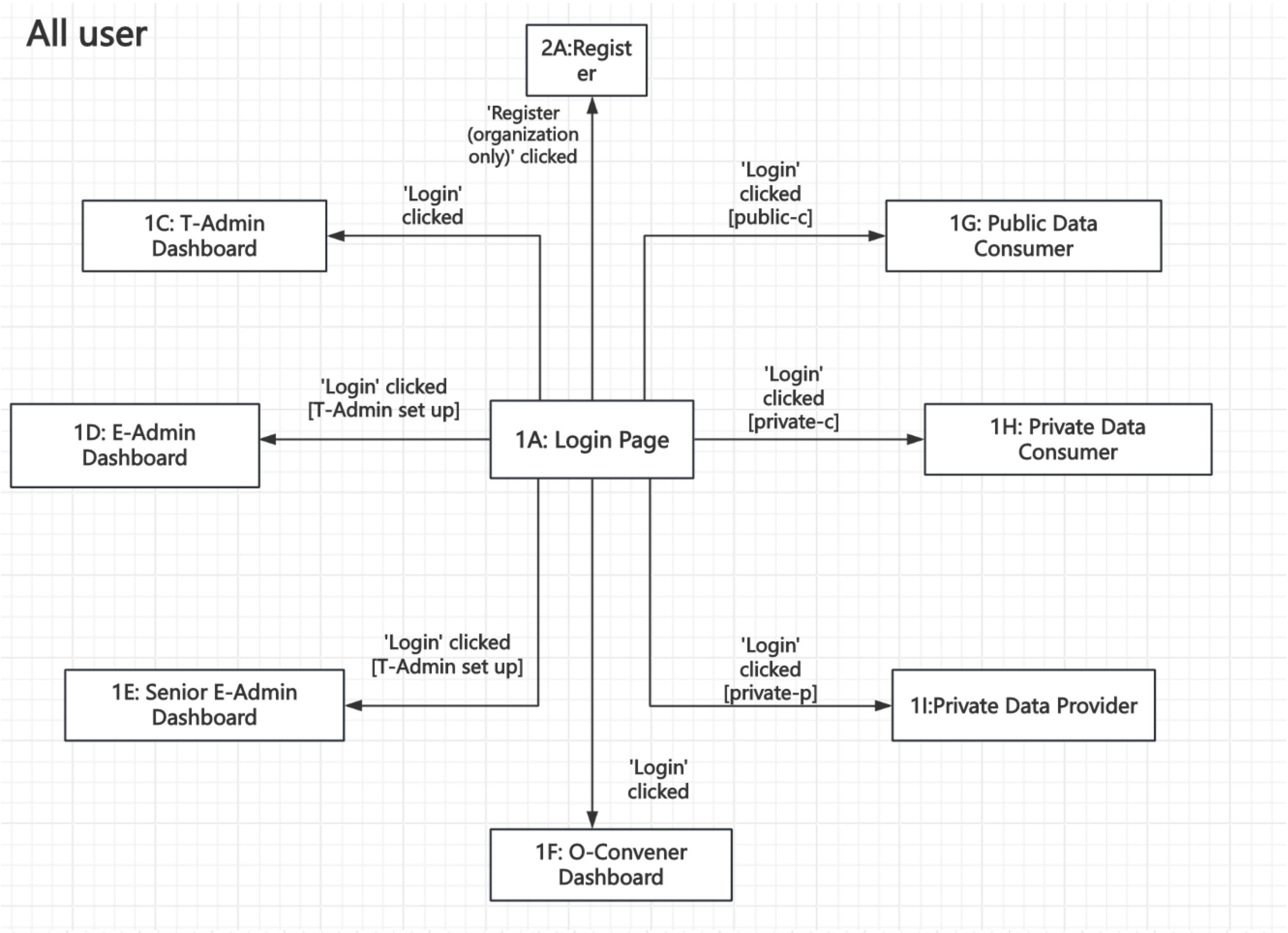
3.1.1 Description and Priority

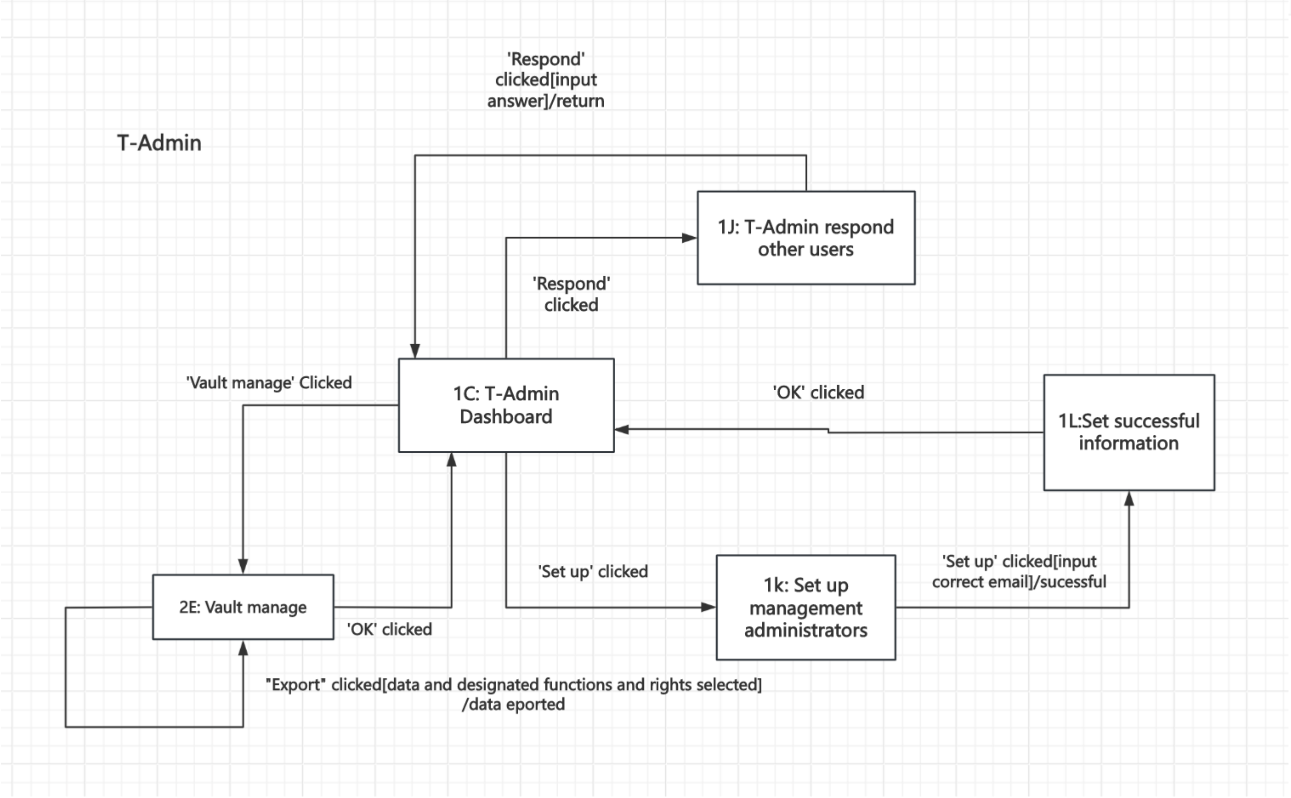
Description:

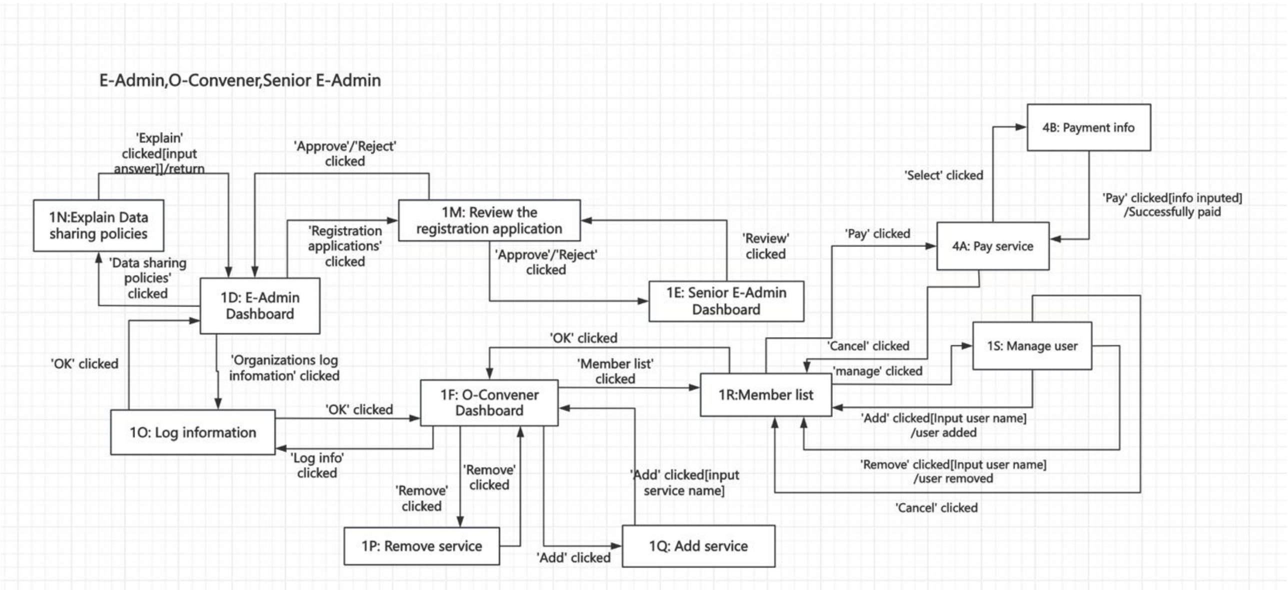
Different actors can login with email code. Each type of logged in actor can have different privileges.

Priority: High

3.1.2 Stimulus/Response Sequences


All user
2A:Register
'Register (organization only)' clicked
'Login' clicked
1C: T-Admin Dashboard
'Login' clicked [public-c]
1G: Public Data Consumer
'Login' clicked [T-Admin set up]
1D: E-Admin Dashboard
1A: Login Page
'Login' clicked [private-c]
1H: Private Data Consumer
'Login' clicked [T-Admin set up]
1E: Senior E-Admin Dashboard
'Login' clicked [private-p]
1I:Private Data Provider
'Login' clicked
1F: O-Convener Dashboard


'Respond' clicked[input answer]/return
T-Admin
1J: T-Admin respond other users
'Respond' clicked
'Vault manage' Clicked
1C: T-Admin Dashboard
'OK' clicked
1L:Set successful information
'Set up' clicked
'Set up' clicked[input correct email]/sucessful
2E: Vault manage
'OK' clicked
1k: Set up management administrators
"Export" clicked[data and designated functions and rights selected] /data eported


E-Admin,O-Convener,Senior E-Admin
'Explain' clicked[input answer]]/return
'Approve'/'Reject' clicked
1N:Explain Data sharing policies
'Data sharing policies' clicked
'Registration applications' clicked
1M: Review the registration application
'Approve'/'Reject' clicked
1D: E-Admin Dashboard
1E: Senior E-Admin Dashboard
'Review' clicked
'OK' clicked
'Organizations log infomation' clicked
1O: Log information
'OK' clicked
'Log info' clicked
1F: O-Convener Dashboard
'OK' clicked
'Member list' clicked
1R:Member list
'Remove' clicked
'Remove' clicked
1P: Remove service
'Add' clicked
1Q: Add service
'Add' clicked[input service name]
4B: Payment info
'Select' clicked
'Pay' clicked[info inputed] /Successfully paid
'Pay' clicked
4A: Pay service
'Cancel' clicked
'manage' clicked
1S: Manage user
'Add' clicked[Input user name] /user added
'Remove' clicked[Input user name] /user removed
'Cancel' clicked

Basic scenario:

All users：

1. All user input email address

2. System from the list of users search the user exists, exists to send code.

3. The user input code

4. User login

T-Admin:

1. Successfully log in and enter the page for responding to help requests.

2. T-Admin inputs the reply information and submits it.

3. Successfully log in and enter the page for managing administrators.

4. T-Admin inputs the user email and sets it as an administrator.

5. The system searches the user list to confirm the user identity.

6. If successful, return to the T-Admin homepage.

7. T-admin can manage and export the vault data with certain functions and right.

E-Admin:

1. Successfully log in and access the registration application page.

2. Review the list of registration applications and make decisions for approval or rejection.

3. Successfully log in and access the interface for data sharing policy.

4. Input explanations regarding the contents of the data sharing policy.

5. Successfully log in and access the interface for viewing organizational log information.

Senior E-Admin:

1. Senior E-Admin successfully logs in and accesses the registration application page.

2. Senior E-Admin reviews the list of registration applications that have been approved by E-Admin, and makes decisions for approval or rejection.

O-Convener:

1. O-Convener successfully logs in and enters the deletion service interface.

2. O-Convener selects the service for deletion.

3. O-Convener successfully logs in and enters the service addition interface.

4. O-Convener inputs the service content for addition.

5. O-Convener successfully logs in and enters the member list to view members.

6. O-Convener successfully logs in and enters the organization log interface to view log information.

7. O-Convener manages users by adding and removing users with their names.

8. O-Convener can input user information to pay successfully.

3.1.3 Functional Requirements

REQ-1: Only registered members can log in using email
REQ-2: No one can modify log information
REQ-3: If the number of administrators reaches the upper limit of two, adding an administrator fails.
REQ-4: Senior E-Admin will only review registration requests that have already been approved by E-Admin. The registration is considered successful only when the Senior E-Admin passes the approval.
REQ-5: T-admin can manage the vault applied by O-convener by exporting.

3.2 Register Organization

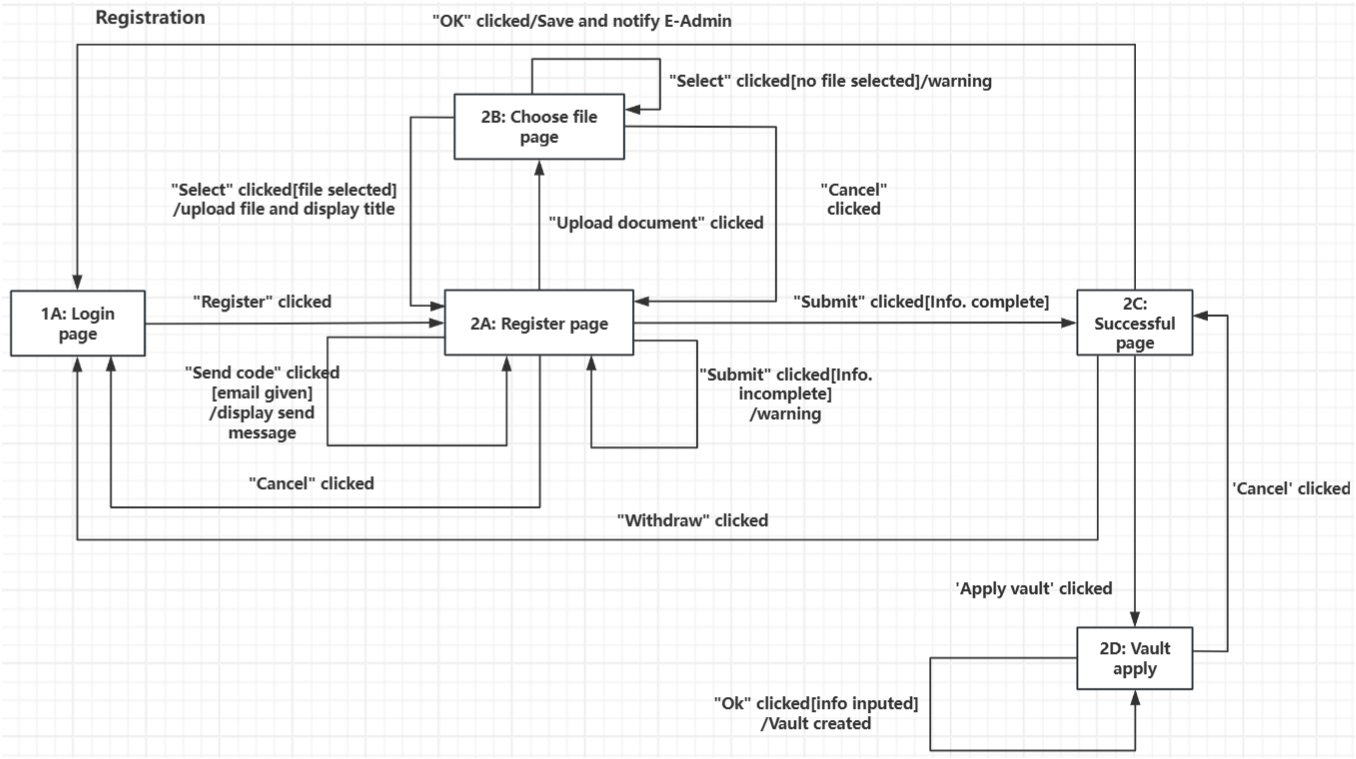
3.2.1 Description and Priority

Description:

The Register Organization feature allows an O-Convener to register their organization in the system. The process involves submitting a document proving representation, entering an email address, verifying the email with a code, and requesting registration. The request is reviewed and approved or rejected by an E-Admin or Sr. E-Admin.

Priority: High

3.2.2 Stimulus/Response Sequences


Registration
"OK" clicked/Save and notify E-Admin
"Select" clicked[no file selected]/warning
2B: Choose file page
"Select" clicked[file selected] /upload file and display title
"Upload document" clicked
"Cancel" clicked
1A: Login page
"Register" clicked
2A: Register page
"Submit" clicked[Info. complete]
2C: Successful page
"Send code" clicked [email given] /display send message
"Submit" clicked[Info. incomplete] /warning
"Cancel" clicked
"Withdraw" clicked
'Cancel' clicked
'Apply vault' clicked
2D: Vault apply
"Ok" clicked[info inputed] /Vault created

Use Case: Register Organization

Assumptions:

- The O-Convener accesses the system and intends to register their organization.
- The O-Convener does not have an account yet, so they can enter the registration page without logging in.
- The system supports verification code sending and document uploads, and the E-Admin is responsible for approving registration requests.
- After a successful registration, the O-Convener has the option to apply for the Vault Service.

Basic Scenario:

1. The O-Convener enters the system and clicks "Register" to access the registration page.
2. The O-Convener fills in the organization details, provides an email address, and requests a verification code.
3. The O-Convener enters the verification code and uploads the required proof document.

4. The O-Convener submits the registration request, and the system validates the completeness of the information and stores the request.

5. The system notifies the E-Admin for approval, and the O-Convener waits for the review result on the success page.

6. The O-Convener can either confirm (OK) or withdraw (Withdraw) the application. If confirmed, the request proceeds for review; if withdrawn, the registration is canceled.

7. If registration is approved, the O-Convener can submit a Vault request with necessary details to apply for the Vault Service.

8. The system processes the request and sets up the Vault.

9. If canceled, the O-Convener returns to the system without applying.

3.2.3 Functional Requirements

REQ-1: The system should provide a Login Page with an email input field, verification code input field, a Register button, and a Send Code button.

REQ-2: Upon clicking the Register button, the system should redirect the O-Convener to the Register Page, displaying input fields for Organization Name, Email, Verification Code, and an Upload Document button.

REQ-3: The system should provide a "Send Code" button, and when the O-Convener enters an email and clicks it, the system should send a verification code to the provided email and display a confirmation message.

REQ-4: The system should provide an "Upload Document" button on the Register Page, and clicking it should navigate to the Choose File Page.

REQ-5: The system should provide a "Select File" button on the Choose File Page, allowing the user to upload proof documents and display the file list.

REQ-6: If the user clicks the "Select File" button without selecting a file, the system should display a warning message prompting the user to choose a file.

REQ-7: When the user clicks the "Submit" button, the system should check for completeness:

If all required information is complete, the system should save the data, redirect to the Successful Page, and notify the E-Admin for approval.

If the information is incomplete, the system should display a warning message requesting the user to complete the missing fields.

REQ-8: The system should provide a "Cancel" button, and clicking it should return the user to the Login Page.

REQ-9: On the Successful Page, the system should display a message stating "The application will be handled within three weekdays. The result will be notified by email."

REQ-10: The system should provide an "OK" button on the Successful Page, which, when clicked, should save the registration information and notify the E-Admin for approval.

REQ-11: The system should provide a "Withdraw" button on the Successful Page, which, when clicked, should cancel the registration request and return the user to the Login Page.

REQ-12: The system shall provide an option for the O-Convener to apply for the Vault Service after registration approval.

REQ-13: The system shall require the O-Convener to enter necessary details, such as organization name and vault usage specifications.

REQ-14: The system shall allow the O-Convener to submit the Vault request for processing.

REQ-15: If the O-Convener cancels the Vault application, the system shall return them to the main system interface without applying.

3.3 Download thesis process

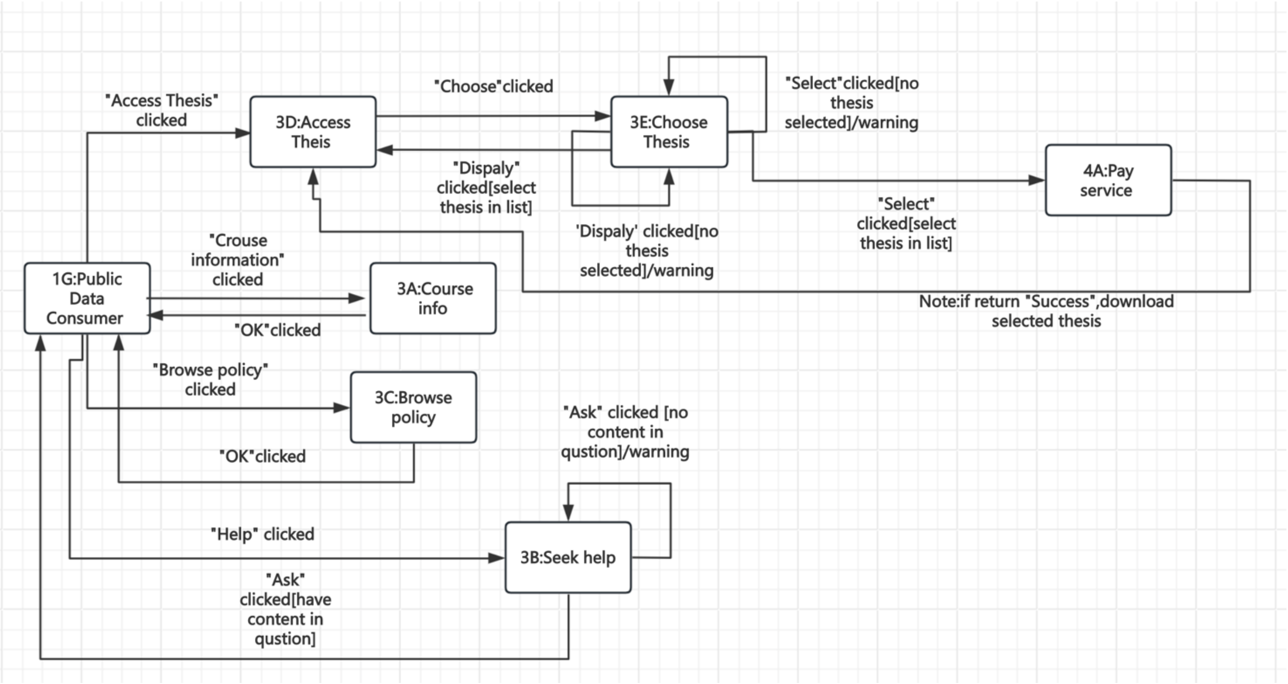
3.3.1 Description and Priority

Description:

The public data consumer selects a thesis listed and makes payment. If payment is successful, the thesis is then downloaded to the actor.

Priority: *High*

3.3.2 Stimulus/Response Sequences


"Access Thesis" clicked
3D:Access Theis
"Choose"clicked
3E:Choose Thesis
"Select"clicked[no thesis selected]/warning
"Dispaly" clicked[select thesis in list]
4A:Pay service
"Select" clicked[select thesis in list]
'Dispaly' clicked[no thesis selected]/warning
"Crouse information" clicked
1G:Public Data Consumer
3A:Course info
"OK"clicked
Note:if return "Success",download selected thesis
"Browse policy" clicked
3C:Browse policy
"Ask" clicked [no content in qustion]/warning
"OK"clicked
"Help" clicked
3B:Seek help
"Ask" clicked[have content in qustion]

Basic Scenario:

Public Data Consumer: Access Thesis

1. Public Data Consumer clicks **"Access Thesis"** to access the **Thesis Page**.

2. On the Thesis Page:

Public Data Consumer clicks **"Choose"** to select a thesis.

Public Data Consumer clicks **"Display"** to view details of the selected thesis.

If no thesis is selected, the system displays a warning: **"Please select a thesis."**

3. Public Data Consumer clicks **"Select"** to choose the desired thesis.

   If no thesis is selected, the system displays a warning: **"Please select a thesis."**

4. Public Data Consumer is redirected to the **Pay Service Page**.

5. On the Pay Service Page:

   Public Data Consumer clicks **"Course Information"** to view course details.

   Public Data Consumer is redirected to the **Course Info Page**.

   Public Data Consumer clicks **"OK"** to return to the Pay Service Page.

6. Public Data Consumer clicks **"Browse Policy"** to view the policy details.

   Public Data Consumer is redirected to the **Browse Policy Page**.

   Public Data Consumer clicks **"OK"** to return to the Pay Service Page.

7. Public Data Consumer clicks **"Ask"** to seek help or ask a question.

   If no content is entered in the question, the system displays a warning: **"Please enter your question."**

   If content is entered, Public Data Consumer is redirected to the **Seek Help Page**.

8. Public Data Consumer clicks **"Help"** to access the **Seek Help Page**.

9. If payment is successful, the system returns **"Success"** and allows the user to download the selected thesis.

3.3.3 Functional Requirements

REQ-1: Access Thesis Page

The system shall allow the user to click "Access Thesis" to access the Thesis Page.

REQ-2: Choose Thesis

The system shall allow the user to click "Choose" to select a thesis on the Thesis Page.

REQ-3: Display Thesis Details

The system shall allow the user to click "Display" to view details of a selected thesis on the Thesis Page.

If no thesis is selected when "Display" is clicked, the system shall display a warning: "Please select a thesis."

REQ-4: Select Thesis

The system shall allow the user to click "Select" to choose a thesis on the Thesis Page.

If no thesis is selected when "Select" is clicked, the system shall display a warning: "Please select a thesis."

REQ-5: Navigate to Pay Service Page

The system shall allow the user to navigate to the Pay Service Page after selecting a thesis.

REQ-6: View Course Information

The system shall allow the user to click "Course Information" on the Pay Service Page to view course details.

The system shall redirect the user to the Course Info Page.

The system shall allow the user to click "OK" to return to the Pay Service Page.

REQ-7: Browse Policy

The system shall allow the user to click "Browse Policy" on the Pay Service Page to view policy details.

The system shall redirect the user to the Browse Policy Page.

The system shall allow the user to click "OK" to return to the Pay Service Page.

REQ-8: Seek Help

The system shall allow the user to click "Ask" on the Pay Service Page to seek help or ask a question.

If no content is entered in the question, the system shall display a warning: "Please enter your question."

If content is entered, the system shall redirect the user to the Seek Help Page.

The system shall allow the user to click "Help" to access the Seek Help Page.

REQ-9: Complete Payment and Download Thesis

If payment is successful, the system shall return "Success" and allow the user to download the selected thesis.

3.4 Pay Service

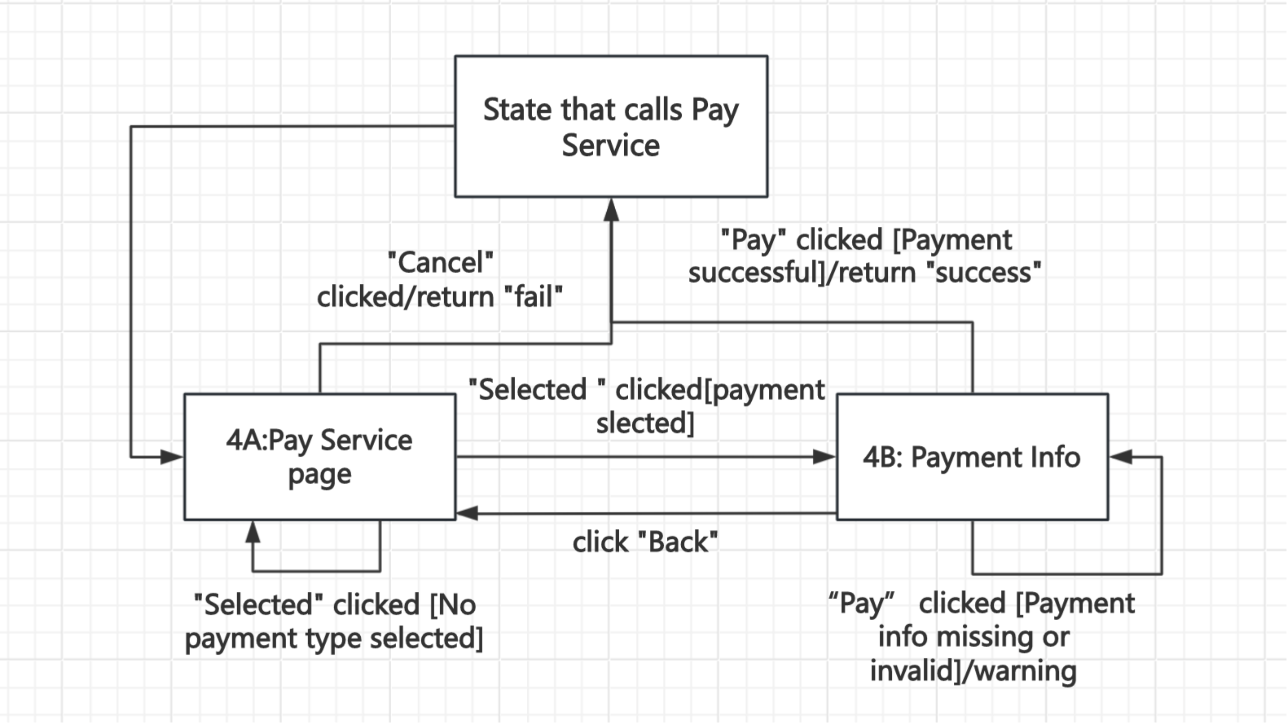
3.4.1 Description and Priority

Description:

The Pay Service allows the O-Convener to pay organization-related expenses (e.g., annual fees) via bank transfer in the initial version, with other methods (PayPal, etc.) disabled. The public data consumer is also the same. Payments start from the Data Access Page and grant content access upon success.

Priority: Except bank transfer, others payment methods have low priority.

3.4.2 Stimulus/Response Sequences


State that calls Pay Service
"Cancel" clicked/return "fail"
"Pay" clicked [Payment successful]/return "success"
"Selected " clicked[payment slected]
4A:Pay Service page
4B: Payment Info
click "Back"
"Selected" clicked [No payment type selected]
"Pay" clicked [Payment info missing or invalid]/warning

Basic scenario:

1. O- convener

The O-Convener has successfully logged into the E-DBA system, registered their organization, and submitted a member list requiring payment (e.g., 5 members, ¥500 due). The Workspace Page reflects this pending payment status.

1. The O-Convener clicks "**Pay Annual Fee**" to initiate payment.
2. The O-Convener clicks "**Confirm Payment**" to proceed.
3. The system shows the Pay Service Page, where only "**Bank Transfer**" is selectable.
4. The user click "**cancel**" return to the State that calls Pay Service page and return fail
5. If no payment type selected and click select, still in the Payment Service Page, warning show up
6. The O-Convener selects "**Bank Transfer**" and clicks "**Select**."
7. The system shows the Payment Info Page where the user should enter **Transferring bank name, Transferring Account Name, Transfer out account, password, Transfer bank name, Transfer account, Amount**. After input the above info, click "**pay**". If pay successful, it will turn to State that calls Pay Service and show "**success**"
8. If the input info are invalid, warning show up
9. User can also click "back" to return the Pay Service page

2. Public data consumer

The Public Data Consumer has logged into the E-DBA system with valid credentials and has public data access rights. They are browsing the Data Access Page, where a thesis is listed as chargeable with a fee of ¥20.

1. The user clicks "**Thesis**" in Thesis list page to access it.
2. The system shows the Payment Service Page, where only "**Bank Transfer**" is selectable.
3. The user click "cancel" return to the State that calls Pay Service page and return fail
4. If no payment type selected and click select, still in the Payment Service Page, warning show up
5. The user selects "Bank Transfer" and clicks "**Select**."
6. The system presents the Bank Transfer Details Page with the pre-filled amount and fields for bank details.
7. The user enters valid bank details (e.g., account number, password) and clicks "**Pay**" If pay successful, it will turn to State that calls Pay Service
8. If the input info are invalid, warning show up
9. The user can also click "**Back**" to Pay service page and show "success"

3. Private Data consumer
   1. Private Data consumer click "**Student identity authentication service**" or "**Student GPA record access**" and turn to the Pay service page
   2. The system shows the Payment Service Page, where only "**Bank Transfer**" is selectable.
   3. The user click "**cancel**" return to the State that calls Pay Service page and return fail
   4. If no payment type selected and click select, still in the Payment Service Page, warning show up
   5. The user selects "**Bank Transfer**" and clicks "**Select**."
   6. The system presents the Bank Transfer Details Page with the pre-filled amount and fields for bank details.
   7. The user enters valid bank details (e.g., account number, password) and clicks "**Pay**", If pay successful, it will turn to State that calls Pay Service
   8. If the input info are invalid, warning show up
   9. The user can also click "**Back**" to Pay service page and show "success"

3.4.3 Functional Requirements

REQ-1:
Different users have different triggers to enter the payment service. O-convener is "pay annual fee", Public data consumer is "download", and private data consumer is "Student identity authentication service" or "Student GPA record access".

REQ-2:
The system shall log all payment attempts (successful or failed) in the organization's immutable activity log, accessible to the O-Convener and E-Admin, including transaction ID, timestamp, amount, and result.

REQ-3:
The system shall encrypt all sensitive payment data (e.g., account numbers, passwords) using TLS/SSL during transmission to the external bank database.

REQ-4:
The system shall notify the Public Data Consumer of the payment result via email after transaction.

REQ-5:
The system shall provide a Payment Method Selection Page where bank transfer is the only selectable option in the initial version, with PayPal, Credit Card, Alipay, and WeChat displayed as disabled options.

3.5 Course Information Management

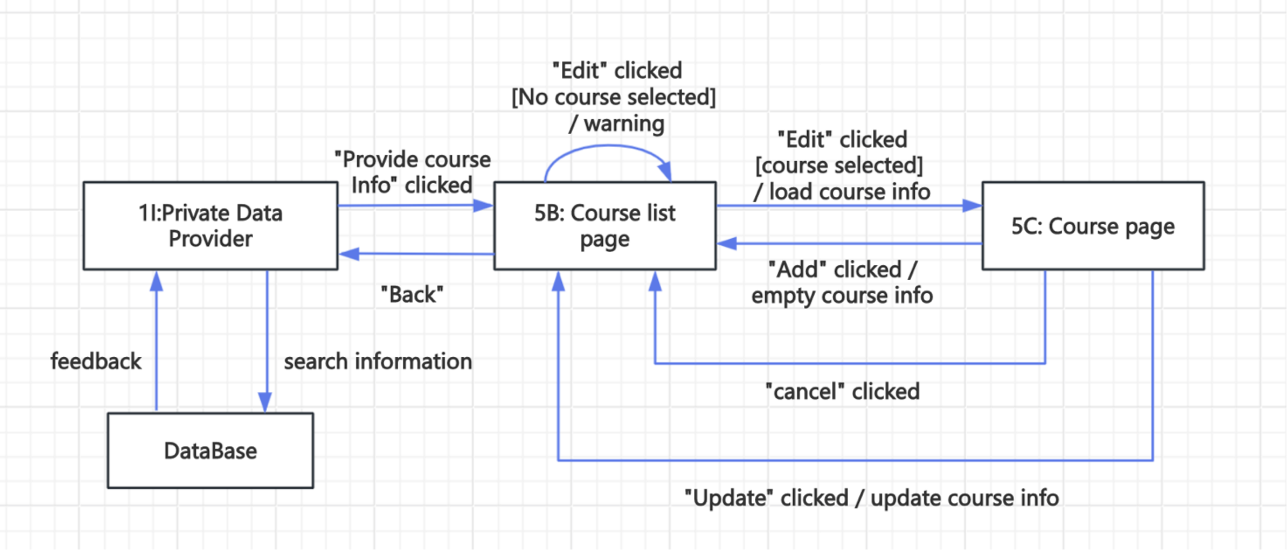
3.5.1 Description and Priority

Description:

This feature allows authorized users to view course information, edit course details, and confirm or cancel edits within the system. It enables efficient management of course data, ensuring it remains accurate and up to date. Private Data Provider enter search information and click the search button. The system then sends a request to the database, which processes the query and returns the relevant information to be displayed to the Private Data Provider.

Priority: High

3.5.2 Stimulus/Response Sequences



Basic scenario:
Actors: Private Data Provider
Preconditions: The user has successfully logged into the system.
The user has the necessary permissions to edit course information.
Course information already exists in the system.
Main Flow:
1.The user navigates to the Course Info Page.
2.The system displays the current course information.
3.The user clicks the "Edit" button.
4.The system navigates to the Course Info Edit Page.
5.The user modifies the necessary course details (e.g., course name, description, schedule).
6.The user clicks the "Save" button.
7.The system navigates to the Confirm Page and displays the updated information for review.

8.The user clicks "Confirm" to finalize the changes.
9.The system updates the course information in the database.
10.The system redirects the user back to the Course Info Page, displaying the updated information.

Alternative Flow:

1.(A1) User Cancels Edit:
At step 6, if the user clicks "Cancel", the system discards changes and returns to the Course Info Page.
2.(A2) User Cancels Confirmation:
At step 8, if the user clicks "Cancel", the system returns to the Course Info Edit Page with the previously entered data intact.

Postconditions:
If confirmed, the course information is successfully updated in the system.
If canceled, no changes are applied, and the original course information remains intact

3.5.3 Functional Requirements

REQ-1: View Course Info
The system shall display existing course information on the Course Info Page upon successful login.

REQ-2: Edit Course Info
The system shall allow the user to navigate to the Course Info Edit Page by clicking "edit" on the Course Info Page.
The system shall allow the user to modify the existing course information.

REQ-3: Save Course Info
The system shall allow the user to click "save" after editing, leading to the Confirm Page for review.

REQ-4: Confirm Changes
The system shall apply the course information updates only after the user clicks "confirm" on the Confirm Page.
After confirmation, the system shall return the user to the Course Info Page with the new information displayed.

REQ-5: Cancel Actions
If the user clicks "cancel" on the Course Info Edit Page, the system shall discard changes and return to the Course Info Page.
If the user clicks "cancel" on the Confirm Page, the system shall return to the Course Info Edit Page with previously entered data intact.

3.6 Private date consumer

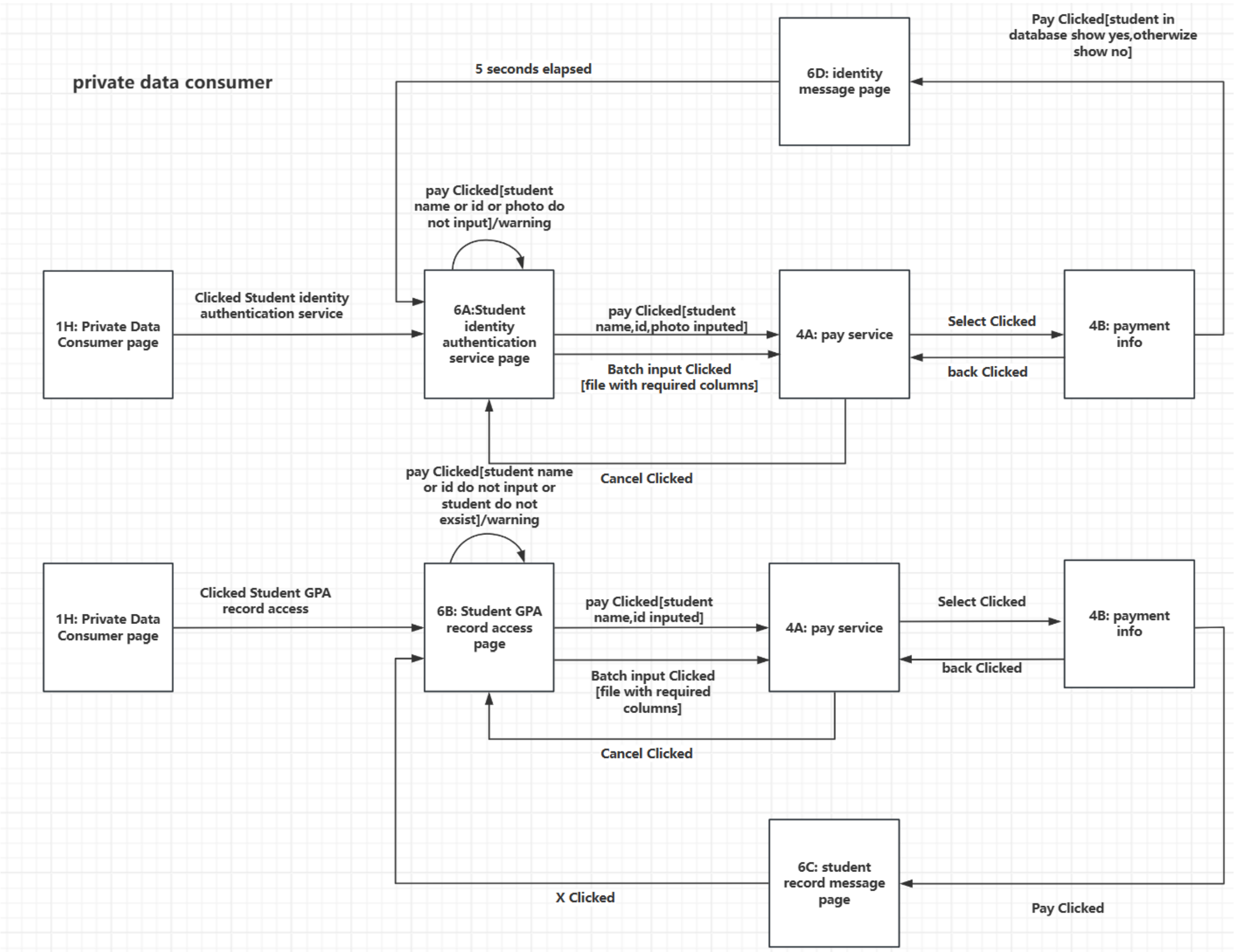
3.6.1 Description and Priority

Description:

Private data consumers access student information services, including authentication and record queries. Identity verification by entering the name, student number, etc. to confirm the identity of the student; Record query returns enrolment, graduation year and GPA.

Priority: High

3.6.2 Stimulus/Response Sequences


private data consumer
5 seconds elapsed
6D: identity message page
Pay Clicked[student in database show yes,otherwize show no]
pay Clicked[student name or id or photo do not input]/warning
1H: Private Data Consumer page
Clicked Student identity authentication service
6A:Student identity authentication service page
pay Clicked[student name,id,photo inputed]
Batch input Clicked [file with required columns]
4A: pay service
Select Clicked
back Clicked
4B: payment info
Cancel Clicked
pay Clicked[student name or id do not input or student do not exsist]/warning
1H: Private Data Consumer page
Clicked Student GPA record access
6B: Student GPA record access page
pay Clicked[student name,id inputed]
Batch input Clicked [file with required columns]
4A: pay service
Select Clicked
back Clicked
4B: payment info
Cancel Clicked
6C: student record message page
X Clicked
Pay Clicked

Basic scenario:
Actors: Private Data consumer
Preconditions: The user has successfully logged into the system.
The user has the necessary permissions to upload student information.
Student data can store in the service database.
Main Flow:

1. The user navigates to the Private Data Consumer Page.
2. The system displays available student info services: “Check Student Identity” and “Get Student Record.”
3. The user selects one of the services (choose one batten).
4.The system navigates to the 1R or 1S Page.
5. The user enters required details (or uploads an Excel file) and clicks “Pay” to get in pay service.
6.The user clicks the "Select" button to choose their payment (only transfer now).
7. The system processes the payment.
8. Upon successful payment, the system displays the requested student information.
9. The user reviews the result and returns to the Private Data Consumer Page.

Alternative Flow:

1.if user do not fill all the required student information. The warning will appear.
2. if the user clicks “Cancel,” the system discards changes and returns to the 1R or 1S Page.

3. if the user clicks “back,” the system discards changes and returns to the Pay service Page.

3.6.3 Functional Requirements

REQ-1: Select Student Info Service
The system shall allow the user to select either "Check Student Identity" or "Get Student Record" from the displayed list.
REQ-2: Submit Query
The system shall allow the user to navigate to the Student Info Input Page by selecting a service, and provide options for manual entry or Excel file upload containing the required student details.
REQ-3: Validate Input
The system shall validate the entered data or the uploaded file for correct format and required columns before proceeding.
REQ-4: Prompt Payment
The system shall prompt the user for payment if the selected service requires a fee, prior to processing the query.
REQ-5: Process Payment and Display Results
The system shall process the payment and, upon successful payment, display the requested student information:
For Check Student Identity, a verification result (e.g., Yes/No).
For Get Student Record, detailed student data including enrollment year, graduation year, and GPA.If the user clicks "cancel" on the Confirm Page, the system shall return to the Course Info Edit Page with previously entered data intact.External Interface Requirements

4. External Interface Requirements

4.1 User Interfaces

All user interfaces are described in another UI.xlsx file.

4.2 Hardware Interfaces

1. Server: Responsible for data storage, processing, calculation and interaction with other components. Equipped with multi-core CPU, supports multi-task parallel processing, thus achieving sufficient computing power. It is also equipped with sufficient memory and hard disk storage to handle complex calculations and store user information, and has strong security protection measures such as firewalls and encryption mechanisms.
2. Network infrastructure: used to connect various components of the E-DBA system, external databases, and user devices. Sufficient stability must be maintained to ensure data transmission and bandwidth must be guaranteed to cope with the increase in the number of users. About network security, the system should use encryption protocols, such as TLS (Transport Layer Security), to prevent interception or tampering.
3. Control interaction: used to manage hardware resources, such as read and write control of database servers, flow control of network devices, etc.
4. Communication Protocol: The communication between the hardware will use TCP/IP based protocols such as HTTP, HTTPS and FTP.

4.3 Software Interfaces

1. **Operating system**: The system supports Windows Server, Mac, and Linux (such as Ubuntu and CentOS). These operating systems provide a running environment that supports Web servers, databases, and other necessary services.
2. **Database interface**: The system connects to multiple databases to store and retrieve data, including:

1. **Course information database**: The system will interact with the course information database for course information sharing. Course information is free of charge. Each organization can manage the courses it offers. Common areas remain within the organization space and are open to any user with public access.
   - Enter: User (any user with public access) make a query request to view course information.
   - Output: Returns the appropriate course information.
2. **Student Identification database**: The system will interact with the student identification database for verifying student identity. The interface will support batch input via Excel files and single input via UI. Data consumers will need to pay on a per-query basis.
   - Enter: student name, ID, photo.
   - Output: Yes/No.
3. **Student records database**: The system will interact with the student records database for access to student GPA records. The interface will support batch entry via Excel files and single entry via UI. This service pays by the number of records.
   - Enter: Student name, student number
   - Output: Student information (name, year of enrollment, year of graduation, GPA).
4. **Thesis database**: The system will interact with the thesis database to retrieve thesis information. Users can access papers from that organization based on the access rights set by that organization. Access to papers may be fee-based and set by the institution that provides the essay sharing service. Paper access may be chargeable.
   1) For title unknown:
      - Enter: Keywords
      - Output: List of papers with keywords in the title
   2) For known titles:
      - Enter: thesis title
      - Output: pdf file
5. **Bank account database**: The system will interact with the bank account database for payment processing.
   1) checkAccountValidity interface:
      - Enter: Bank name, account name, account number.
      - Output: Success/failure.
   2) transferMoney Interface:
      - Input: Output bank: Bank name, account name, account number, password; Input bank: bank name, account name, number of account number
      - Output: Success/failure

3 Libraries and tools

- **Payment interface tool**: The system integrates the payment platform tools, including the API interface of PayPal, wechat Pay, Alipay, Transfer and Credit card. **Purpose**: To provide users with online payment services, processing fee payment and return of payment results.
- **File processing library**: Use file processing library (such as Apache POI or openpyxl) to handle Excel file uploading and parsing, supporting batch import of student information, member list, etc. **Purpose**: Allow users to upload Excel files (such as student list or member list) for batch processing.

4. Nature of service and communication

- **Service type**: The services provided by the system include work space, user identity authentication service, student GPA record inquiry service, course information and paper sharing service, payment service, etc. Each service is accessed and processed through different apis.
- **Communication protocols**: Standard communication protocols are used between the system and external components (such as payment platforms, file upload interfaces, etc.) :
  - **HTTP/HTTPS**: Used for data interaction between the front end of the Web and the back end.
  - **RESTful API**: Service communication between internal and external components of the system. All external requests interact with the system through RESTful apis that support data exchange in JSON or XML format.
  - **OAuth2/JWT**: Used for user authentication and authorization to ensure the security of the user's identity.

5. Data sharing mechanism

1. Data sharing: Data is shared between different modules within the system. In order to ensure data consistency and security, all data sharing requires access control through apis.
2. Implementation constraints: In a multi-task operating system, a Global Data Area may be used to store shared configuration information, service status, etc., to ensure data synchronization between different modules.
3. Data access permissions: Data access permissions are controlled by system administrators and organization conveners, and only users with appropriate permissions can access the corresponding data (such as papers, student GPA, etc.).

4.4 Communications Interfaces

1. e-mail interface: The system will send email notifications for user registration, login verification codes, and other system-related communications.
Format: Emails will include a subject line, a clear message body, and any necessary links or codes.
Frequency: Emails will be sent on-demand
2. web browser interface: Users will interact with the system via a web browser. The system will use HTTP/HTTPS protocols to communicate with the browser. Data exchanged between the browser and the system will be in JSON format for API requests/responses and HTML for web pages.
3. network server communications protocols:

- HTTP/HTTPS: All communication between the E-DBA system and external systems (e.g., payment gateways, external databases) will use HTTPS to ensure secure data transmission.

Port: 443 by default

- SMTP: The system will use SMTP to send email notifications to users.
Port:22 by default
- FTP/SFTP: the system use SFTP for secure file transfers.
Port:22 by default

4. Access List Management:
The Organization Convener (O-Convener) will provide an access list to E-DBA, specifying the rights for each user. The system will process this list and assign appropriate access rights.
5. Communication Security

- Encryption: All sensitive data (e.g., user credentials, payment information) will be encrypted during transmission using TLS/SSL protocols.
Encryption Standards: AES-256 for data encryption.
- Identity Verification: The system will use API keys, OAuth, or other authentication mechanisms to ensure that only authorized systems can access external services.

- Data Integrity: The system will use checksums or digital signatures to ensure that data has not been tampered with during transmission.

6. Data Transfer Rates: The system will support data transfer rates of up to 1 Gbps for internal communication and 100 Mbps for external communication.
7. Latency: The system will aim for a maximum latency of 200 ms for API requests.
8. Synchronization Mechanisms: The system will use timestamp-based synchronization to ensure that data is consistent across different components (e.g., external databases, payment gateways).

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Payment Service Performance:
- Description: The system must support at least bank transfer, with optional methods like PayPal, WeChat, Alipay, and credit cards. It must handle high concurrent payment requests efficiently and ensure stability.
- Rationale: The system must meet diverse payment needs and ensure timely transactions, avoiding delays or crashes during peak periods.

Student Identity Authentication Performance:
- Description: The system must support high concurrency for student identity queries, especially during peak periods, ensuring fast authentication processing.
- Rationale: The system must handle multiple authentication requests simultaneously without delays or failures due to high load.

Data Access Performance:
- Description: Data access, including student records and thesis sharing, should allow quick retrieval, even with large data volumes, responding within seconds.
- Rationale: Users expect quick responses for both public and private data. The system should prevent performance bottlenecks, even under heavy usage.

System Load Handling:
- Description: The system should use load balancing to evenly distribute user load during high concurrency, particularly for data storage, identity authentication, and payment modules.
- Rationale: Efficient load management is critical to maintaining performance during peak usage times.

5.2 Safety Requirements

Data Privacy and Security:
- Requirement: Protect sensitive data (e.g., student identities, thesis, GPA) from unauthorized access or loss using encryption and secure authentication methods.
- Safeguards: Encrypt data at rest and in transit, enforce strict user roles and access control.

Payment System Security:
- Requirement: Ensure secure handling of payment transactions to prevent fraud.
- Safeguards: Use secure payment gateways.
- Regulations: Follow financial regulations for payment processing.

Student Identity Authentication:
- Requirement: Prevent unauthorized access to student identity data.
- Safeguards: Implement two-factor authentication (2FA) or other secure methods.
- Regulations: Comply with laws like FERPA regarding student data privacy.

Data Vault Service:

- Requirement: Ensure the safety and integrity of data stored in the vault.
- Safeguards: Use data backup systems and restrict access to authorized users.
- Regulations: Adhere to data protection laws for stored sensitive information.

System Resilience:
- Requirement: Ensure system resilience to prevent data loss or service interruptions.
- Safeguards: Implement fault-tolerant architecture, regular backups, and disaster recovery plans.
- Regulations: Follow industry standards for disaster recovery and data protection.

5.3 Security Requirements

User Identity Authentication:
- Requirement: The system must authenticate all users to ensure that only authorized individuals can access the system and its services. Specifically, the system must provide secure authentication mechanisms for sensitive operations such as student identity verification and thesis access.
- Safeguards: Implement strong password policies and multi-factor authentication (e.g., 2FA) to ensure the uniqueness and security of user identities. For sensitive actions like student identity verification or accessing private data, multi-factor authentication should be required.

Data Privacy Protection:
- Requirement: The system must protect all user data (e.g., student information, grades, thesis, payment information) from unauthorized access, leakage, or alteration.
- Safeguards: All sensitive data should be encrypted, both in transit and at rest. Additionally, the system should ensure that only users with appropriate permissions can access specific types of data, such as student identity information or academic records.

Payment Information Security:
- Requirement: The payment system must ensure the security of user payment information to prevent financial fraud.
- Safeguards: Payment transactions should be processed through secure payment gateways, in compliance with PCI DSS (Payment Card Industry Data Security Standard), to ensure the encryption and protection of payment data.

5.4 Software Quality Attributes

Adaptability:
- Requirement: The system should be flexible and scalable to meet the needs of different organizations. It should be able to scale resources to handle increased user load as the number of users grows.
- Specific Requirement: The system should support dynamic configuration for organization members, data storage, and services based on the needs of different organizations.

Availability:
- Requirement: The system should have high availability, especially for critical services like data sharing and identity verification, ensuring the system is accessible most of the time.
- Specific Requirement: The system should maintain an availability rate of 99.9%, and be capable of quickly recovering in the event of a failure.

Correctness:
- Requirement: The system should provide accurate and reliable services, ensuring that user requests and data are processed correctly, particularly for sensitive functions like student identity verification, thesis sharing, and payment.
- Specific Requirement: The system must undergo unit testing and integration testing to ensure that all operations and data processing are correct.

Flexibility:

- Requirement: The system should support various operational modes and user configurations, allowing different roles (e.g., administrators, students, and organization members) to customize their operations as needed.
- Specific Requirement: The user interface should support flexible configurations, allowing services to be enabled or disabled based on user roles and permissions.

Interoperability:

- Requirement: The system should be able to interoperate with other educational systems and platforms, particularly for data exchange and payment interfaces, ensuring smooth integration with external systems.
- Specific Requirement: The system should support standardized data formats and communication protocols to ensure seamless integration with external databases, payment systems, and educational platforms.

Maintainability:

- Requirement: The system should be easy to maintain and update to accommodate future feature extensions or technical updates.
- Specific Requirement: The code should be clear, modular, and well-documented, allowing easy maintenance and future upgrades.

Portability:

- Requirement: The system should be able to run on different hardware and operating system environments, allowing flexible deployment across various educational institutions.
- Specific Requirement: The system should support common operating systems (such as Windows and Linux) and browsers (such as Chrome and Firefox).

Reliability:

- Requirement: The system must maintain high reliability during normal use, ensuring minimal interruptions to service.
- Specific Requirement: The system should maintain an error rate below 0.1%, and be capable of recovering quickly in case of failure.

Reusability:

- Requirement: The system’s code and modules should be reusable for other similar Lightweight Education Data Bay Areas.
- Specific Requirement: Good coding practices, such as modular design and interface-based development, should be followed so that components can be reused and extended in other systems.

Robustness:

- Requirement: The system should remain stable under varying conditions, preventing crashes or errors in exceptional situations.
- Specific Requirement: The system should handle exceptional scenarios, such as incorrect user input or network delays, and continue to function stably.

Testability:

- Requirement: The system should be easy to test, particularly for validating key functions such as payments, identity verification, and data access.
- Specific Requirement: The system should provide easy access to logs and debugging information, and support automated testing tools.

Usability:

- Requirement: The system should provide a good user experience, particularly for operations such as student identity verification, thesis access, and payments, allowing users to complete tasks quickly.
- Specific Requirement: The system should have an intuitive user interface, supporting the needs of different user roles, enabling users to complete tasks easily.

6. Other Requirements

- **Scalability**: The database must handle growing data volumes (e.g., student records, theses).

- **Integrity & Backup**: Ensure data integrity and perform regular backups.
- **Security**: Encrypt sensitive data and restrict access based on roles.
- **Multi-language Support**: Support English and Chinese initially.
- **Localization**: Adapt date, time, and currency formats to user regions.
- **Data Sovereignty**: Comply with regional data laws (e.g., GDPR).
- **Copyright**: Enforce copyright policies for shared theses and materials.
- **Audit Logs**: Maintain immutable logs of user activities.
- **Modular Design**: Use modular components for easy integration.
- **APIs**: Provide APIs for key functionalities (e.g., authentication).
- **UI Accessibility**: Comply with WCAG 2.1 standards.
- **Documentation**: Provide accessible formats for user guides.
- **System Monitoring**: Include tools for performance monitoring and updates.
- **User Support**: Provide a helpdesk for issue reporting.
- **Data Vault**: Design for future implementation of secure data storage.
- **Analytics**: Plan for advanced analytics and reporting features.

Appendix A: Glossary

T-Admin Technical Administrator: Handles system setup and maintenance.

E-Admin Management Administrator: Approves registration applications.

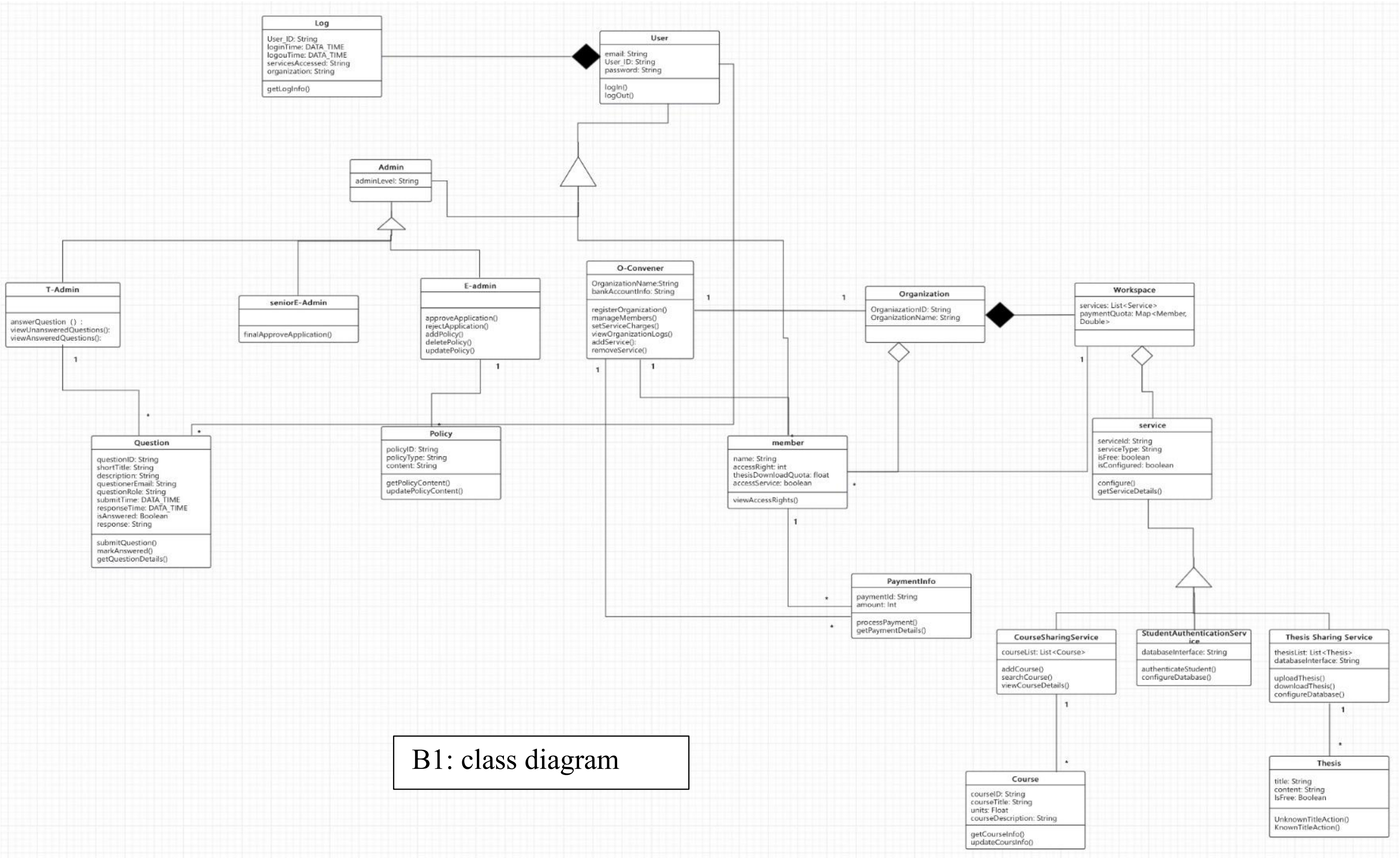
O-Convener **Organization Convener:** Manages user lists and services for their organization.

JSON： JavaScript Object Notation

SMTP： Simple Mail Transfer Protocol

FTP/SFTP： File Transfer Protocol/Secure File Transfer Protocol

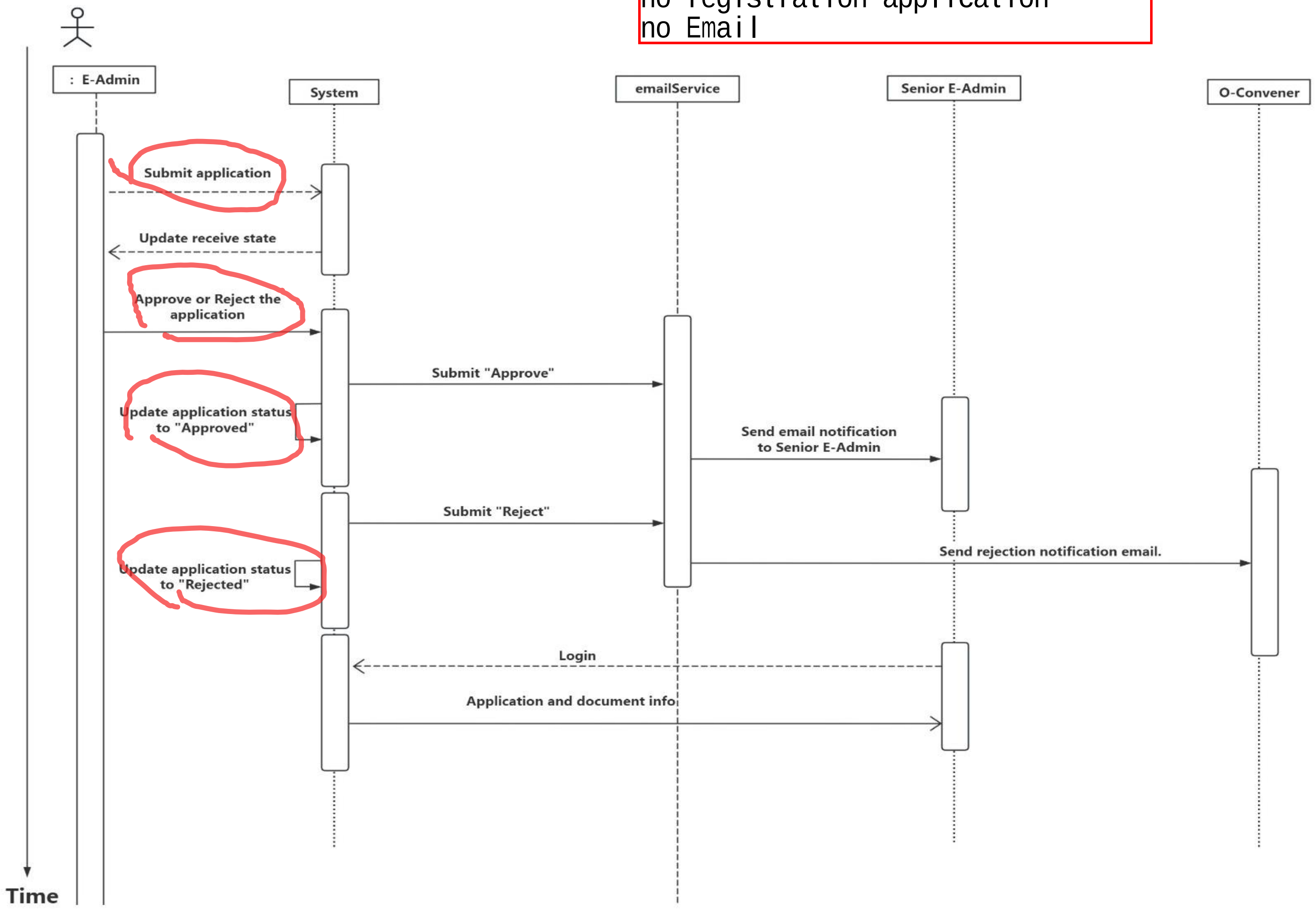
Appendix B: Analysis Models



B1: class diagram

B2: sequence diagram

no control,
no registration application
no Email


: E-Admin
System
emailService
Senior E-Admin
O-Convener
Submit application
Update receive state
Approve or Reject the application
Submit "Approve"
Update application status to "Approved"
Send email notification to Senior E-Admin
Submit "Reject"
Send rejection notification email.
Update application status to "Rejected"
Login
Application and document info
Time

Appendix C: Issues List

- **Identity Authentication**: Input: Name, ID, photo. Output: Yes/No.
- **Student Record**: Input: Name, ID. Output: Name, enrollment year, GPA.
- **Unknown Title**: Input: Keywords. Output: List of matching theses.
- **Known Title**: Input: Thesis title. Output: PDF file.
- **Check Account Validity**: Input: Bank name, account number. Output: Success/Fail.
- **Transfer Money**: Input: Bank details, amount. Output: Success/Fail.