
Software Requirements Specification

for

E-DBA

Version 4.0 approved

Prepared by

| | |
|-------------|------|
| ZHOU Le | full |
| HE Zhe | full |
| CHEN Yijing | full |
| TAO Yiying | full |
| LI Lei | full |
| ZHOU Leitai | full |

Group b03

2025/3/19

Table of Contents

| | |
|---|----------|
| 1. Introduction | 1 |
| 1.1 Purpose | 1 |
| 1.2 Document Conventions | 1 |
| 1.3 Intended Audience and Reading Suggestions | 1 |
| 1.4 Project Scope | 1 |
| 1.5 References | 1 |
| 2. Overall Description | 2 |
| 2.1 Product Perspective | 2 |
| 2.2 Product Features | 2 |
| 2.3 User Classes and Characteristics | 6 |
| 2.4 Operating Environment | 7 |
| 2.5 Design and Implementation Constraints | 7 |
| 2.6 User Documentation | 7 |
| 2.7 Assumptions and Dependencies | 7 |
| 3. System Features | 8 |
| 3.1 Login | 8 |
| 3.2 Register Organization | 9 |
| 3.3 Access Thesis | 10 |
| 3.4 Pay Service | 11 |
| 3.5 Provide Course Information | 12 |
| 3.6 Manage Own Workspace | 13 |
| 3.7 Manage Users | 14 |
| 3.8 Check Organization Log | 16 |
| 3.9 Check System Log | 17 |
| 3.10 Explain Data Sharing Policy | 18 |
| 3.11 Approve/Reject Application | 19 |
| 3.12 Answer Help Request | 20 |
| 3.13 Manage E-Admin Accounts | 21 |
| 3.14 Manage Vault Application* | 22 |
| 3.15 Browse Policy | 22 |
| 3.16 Seek Help | 22 |
| 3.17 Access Course Information | 23 |

| | |
|---|-----------|
| 3.18 Access Student Information | 24 |
| 3.19 Provide Database Interface information | 26 |
| 4. External Interface Requirements | 27 |
| 4.1 User Interfaces | 27 |
| 4.2 Hardware Interfaces | 27 |
| 4.3 Software Interfaces | 28 |
| 4.4 Communications Interfaces | 29 |
| 5. Other Nonfunctional Requirements | 30 |
| 5.1 Performance Requirements | 30 |
| 5.2 Safety Requirements..... | 31 |
| 5.3 Security Requirements..... | 31 |
| 5.4 Software Quality Attributes | 31 |
| 6. Other Requirements | 31 |

Revision History

| Name | Date | Reason For Changes | Version |
|-------------|------|---------------------------------------|---------|
| ZHOU Le | 2/25 | Complete 2.1-2.4 | 1.0 |
| HE Zhe | 2/25 | Complete 5.1-5.4 | 1.1 |
| CHEN Yijing | 2/25 | Complete 2.5-2.7 & 6 &Appendix | 1.2 |
| TAO Yiying | 2/25 | Complete section 1 | 1.3 |
| LI Lei | 2/25 | Complete section 4 | 1.4 |
| ZHOU Leitai | 2/25 | Add context model & diagram &scenario | 1.5 |
| CHEN Yijing | 3/4 | Complete 3.5 | 2.0 |
| TAO Yiying | 3/4 | Complete 3.2 & 3.5 &3.3 state diagram | 2.0 |
| ZHOU Leitai | 3/4 | Complete 3.1 UI design | 2.0 |
| ZHOU Le | 3/4 | Complete 3.1 state diagram & scenario | 2.0 |
| CHEN Yijing | 3/5 | Complete 3.3 UI | 2.0 |
| HE Zhe | 3/5 | Complete 3.3 scenario | 2.0 |
| Tao Yiying | 3/10 | Update the 3.2-3.5 state diagram & UI | 3.0 |
| LI Lei | 3/11 | Complete 3.15-3.17 features | 3.0 |
| ZHOU Le | 3/11 | Complete 3.9-3.11 features | 3.0 |
| CHEN Yijing | 3/11 | Complete 3.12-3.13 features | 3.0 |
| ZHOU Leitai | 3/11 | Complete 3.6-3.8 features | 3.0 |
| Tao Yiying | 3/12 | Complete 3.19 features | 3.0 |
| HE Zhe | 3/12 | Complete 3.18 feature | 3.0 |

| | | | |
|---------------|------|---------------------------------|-----|
| LI Lei | 3/18 | Complete sequence diagram | 4.0 |
| Left 5 member | 3/18 | Complete class diagram together | 4.0 |

1. Introduction

1.1 Purpose

The purpose of this software requirements specification is present a detailed description for the Version 1.3 of the Lightweight Education Data Bay Area (E-DBA) system. It will explain the purpose and user roles in the system, the interface of the system, what the system will do and the constraints under which it must operate. This document is intended for users of the software system and also potential developers.

1.2 Document Conventions

This SRS uses Times New Roman in Regular form for the main text. To show priority, requirements are numbered P1 - P5, with P1 being the highest. For emphasis, italic text is used for important terms and features, like new services.

1.3 Intended Audience and Reading Suggestions

For users, we recommend reading the following sections:

- 2.2 Product Features: Learn about the system's main functionalities.
- 2.6 User Documentation: Find instructions on how to use the system.
- 4.1 User Interfaces: Familiarize yourself with the system's user interface design.

For developers, we recommend reading the following sections:

- 3. System Features: Review detailed functional requirements.
- 4. External Interface Requirements: Learn about the system's interface design.
- 5. Other Nonfunctional Requirements: Check performance, security, and other non-functional requirements

1.4 Project Scope

The E-DBA system aims to streamline data sharing and collaboration among higher education institutions, offering secure and easy access to student records, thesis sharing, and course information, while supporting online payments and optional data vault services. Refer to the project scope document for further information.

1.5 References

SE_SDW_Project Long Description V1.3.docx

UI.pptx

2. Overall Description

2.1 Product Perspective

E-DBA (Education Data Bay Area) is a newly developed, stand-alone system designed to provide higher education institutions with a secure and transparent platform for data sharing and management. The system is inspired by the International Data Space (IDS) model and optimized to meet the management needs of higher education institutions. The E-DBA allows different organizations to share education-related data in a secure environment, including student identity authentication, thesis access, online payments. As a data sharing platform, E-DBA interacts with multiple external systems.

The following is the context model diagram of E-DBA:

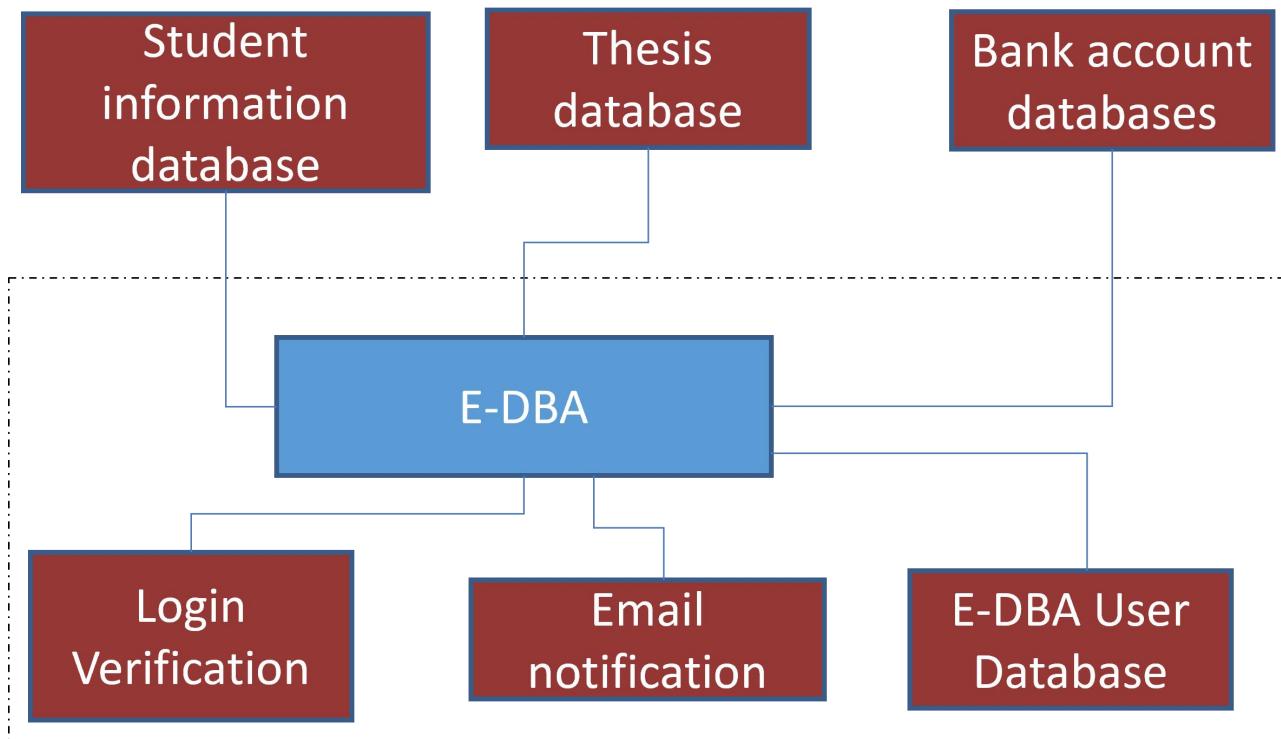


Figure 1. context model

2.2 Product Features

1) Institution Registration and Management

- Institutional Registration:

Each university organisation needs to be registered and approved by the administrator.

- O-Convener (organisation representative) management:

O-Convener is responsible for the management of users within the organisation, including member lists, permission configurations, and so on.

- Workspace:

Each university organisation has a separate space where members and data access rights can be managed.

2) Data Access and Sharing

- Public Data (Free Access):

- i. Course information is publicly accessible
- ii. Thesis sharing access is determined by the providing institution.

- Private Data (Restricted Access):

- i. Student Identity Authentication Service
- ii. GPA Record Access

3) Online Payments

Support bank transfer, PayPal, WeChat payment, Alipay, credit card, etc. Institutions can set thesis access fee, identity authentication query fee and so on.

4) Access Logs and Security Controls

All user actions are logged, O-Convener can query logs, and the system provides role-based access rights management.

5) Data Storage

E-DBA provides data safe (Vault) service, organizations can securely store and manage shared data.

Here is the scenario for the use case O-Convener Register:

Basic Scenario for "O-Convener Register" use case

- The O-Convener navigates to the "Register" portal (no login required).

- The system displays the organization registration form (fields: organization name, proof documents, convener's email address).
- The O-Convener fills in the required information and uploads proof documents.
- The O-Convener submits the registration application.
- The system confirms receipt and notifies the O-Convener: "Application submitted. Awaiting approval from E-DBA admins."

Alternative Scenario – Invalid/Missing Documents

- The O-Convener navigates to the "Register" portal.
- The system displays the organization registration form.
- The O-Convener fills in the information but uploads invalid/unreadable documents (e.g., wrong format, missing pages).
- The system detects the error and displays: "Invalid document format. Supported formats: PDF, JPG, DOCX."
- The O-Convener corrects the documents and resubmits the application.
- The system accepts the revised submission and confirms: "Application submitted. Awaiting approval."

The following is the use case diagrams of E-DBA:

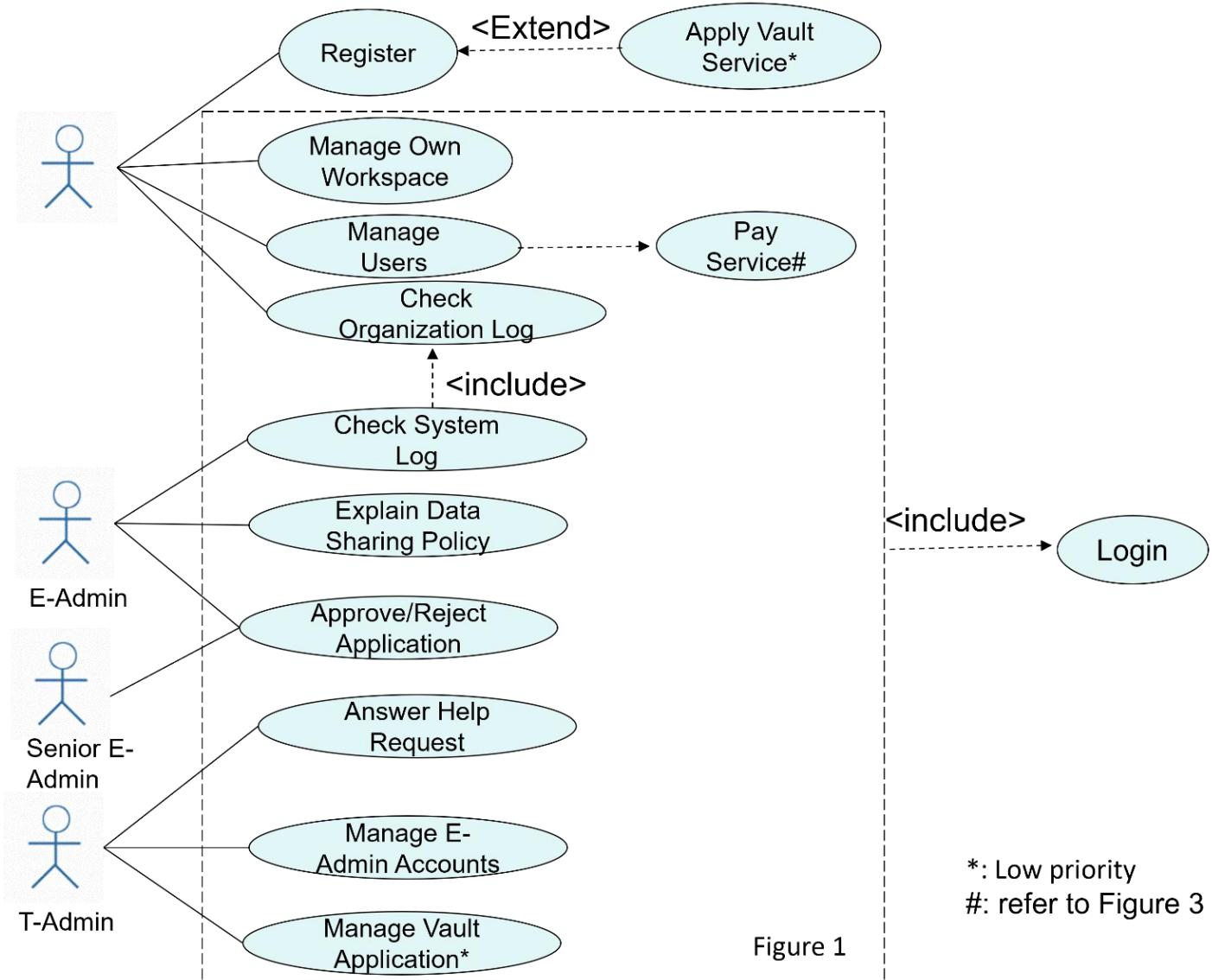
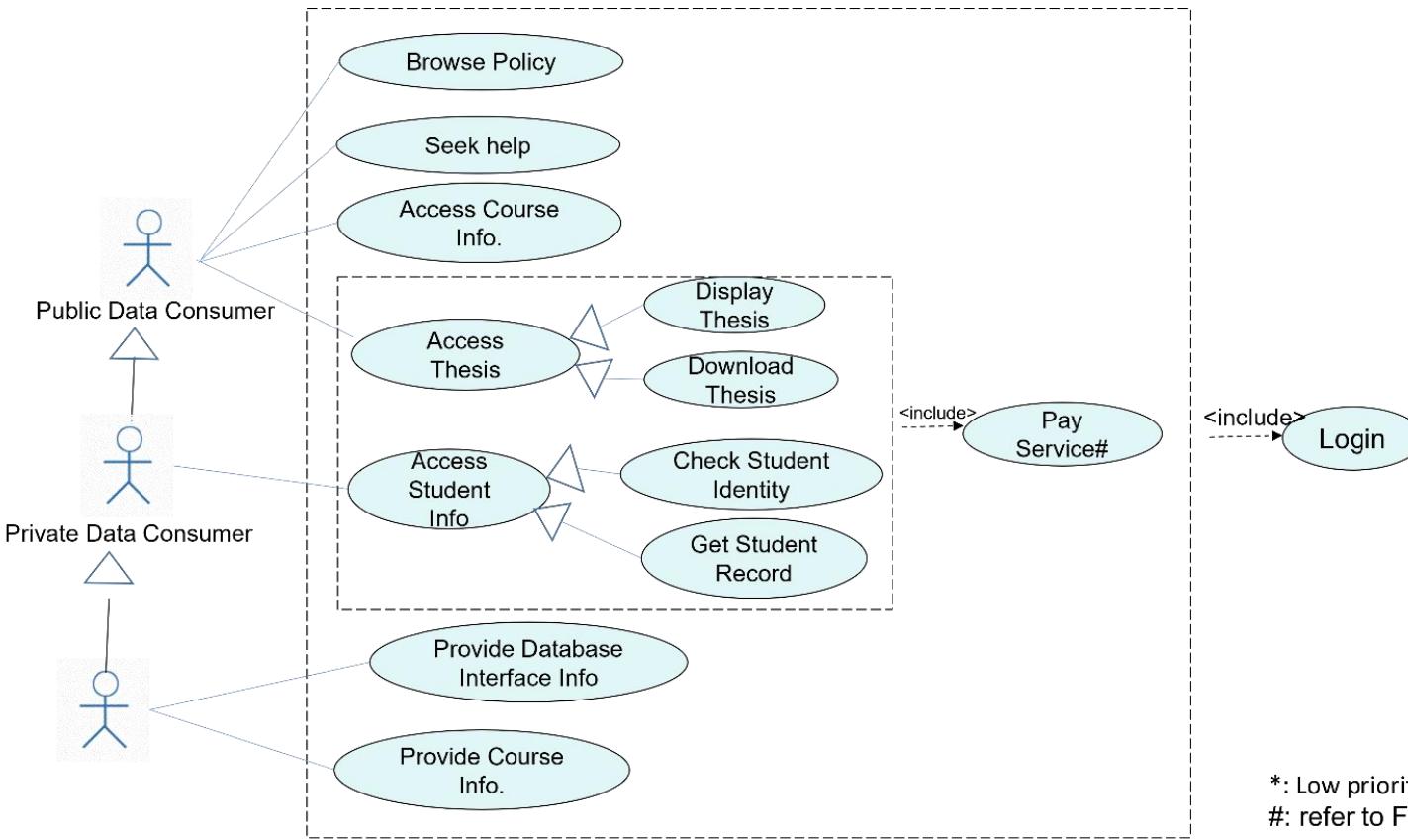


Figure 1



*: Low priority
#: refer to Figure 3

Figure 2

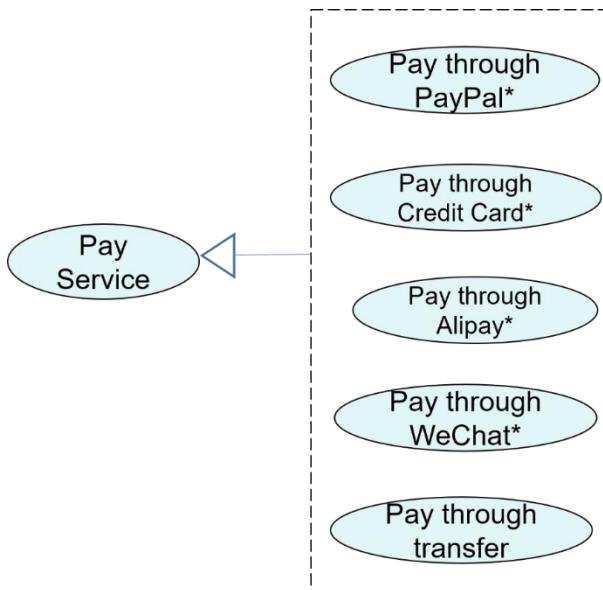


Figure 3

*: Low priority

2.3 User Classes and Characteristics

1) Administrators:

- Technical admin (will be called T-Admin):
 - Privileges: Responsible for system maintenance, user support and managing administrator accounts.
 - Characteristics: Advanced technical level, familiarity with system architecture and O&M tools required.
- Management admin:
 - E-Admin:
 - ◆ Privileges: Approve/reject the registration applications and explain data sharing policies
 - ◆ Characteristics: Medium skill level with an understanding of data management and privilege control is required.
 - Senior E-Admin:
 - ◆ Privileges: Final Approval Authority Registration.
 - ◆ Characteristics: Medium technical level and familiarity with the approval process and compliance standards.
 - O-Convener:
 - ◆ Privileges:
 - Submit registration applications and maintain a list of organization members.
 - Set data access rights and manage payment information.
 - Monitor logs and manage service options for the organization.
 - ◆ Characteristics: Basic technical level, mainly performing management operations.

2) Data provider and data consumer

- Public Data Consumer:

Characteristics: Regular users, with lower privileges, can browse policies, ask for help, access course information, limited privileges to access papers, and some services for a fee.

- Private Data Consumer:

Characteristics: Have higher privileges to access student information, including authentication and record access, usually a teacher or administrator.

- Private Data Provider:

Characteristics: Responsible for providing database and course information with the highest level of authority, usually a system administrator or data manager.

2.4 Operating Environment

The software will run in a cloud environment and supports multi-platform compatibility. The back-end system is deployed on Windows Server 2019+ to ensure system stability and high performance. Clients can be accessed through modern web browsers on Windows 10+ and major Linux distributions. The database system uses MySQL 8+ and is managed through a Docker containerized deployment. In addition, E-DBA integrates with external payment platforms such as PayPal, Stripe, WeChat Pay, and Alipay to provide flexible and secure online payment capabilities. The hardware specifications required for deployment are detailed in Section 3 of this document.

2.5 Design and Implementation Constraints

This system will be compatible with excel files because access member lists submitted by O-convenor and batch input student data in student identity authentication service.

Some databases interfaces (student information database, thesis database, bank account database) cannot be used in the code directly. Those interfaces information is inputted in the configuration service by provider.

This system will follow copyright policies to ensure data sharing and consumption comply with regulations.

2.6 User Documentation

TBD

2.7 Assumptions and Dependencies

Assumed the third-party payment platform (PayPal, Alipay, etc.) runs stably and the interface is normally available.

This system relies on external databases (student information database, thesis database, bank account database) to provide data support. Rely on third-party payment platform to realize online payment function. Rely on network infrastructure for data communication. And this system applies to Windows and Linux users.

3. System Features

3.1 Login

3.1.1 Description and Priority

Different actors can login with email code. Each type of logged in actor have different privileges. The system will automatically recognize the identity of the login role according to the email and enter different page.

3.1.2 Stimulus/Response Sequences

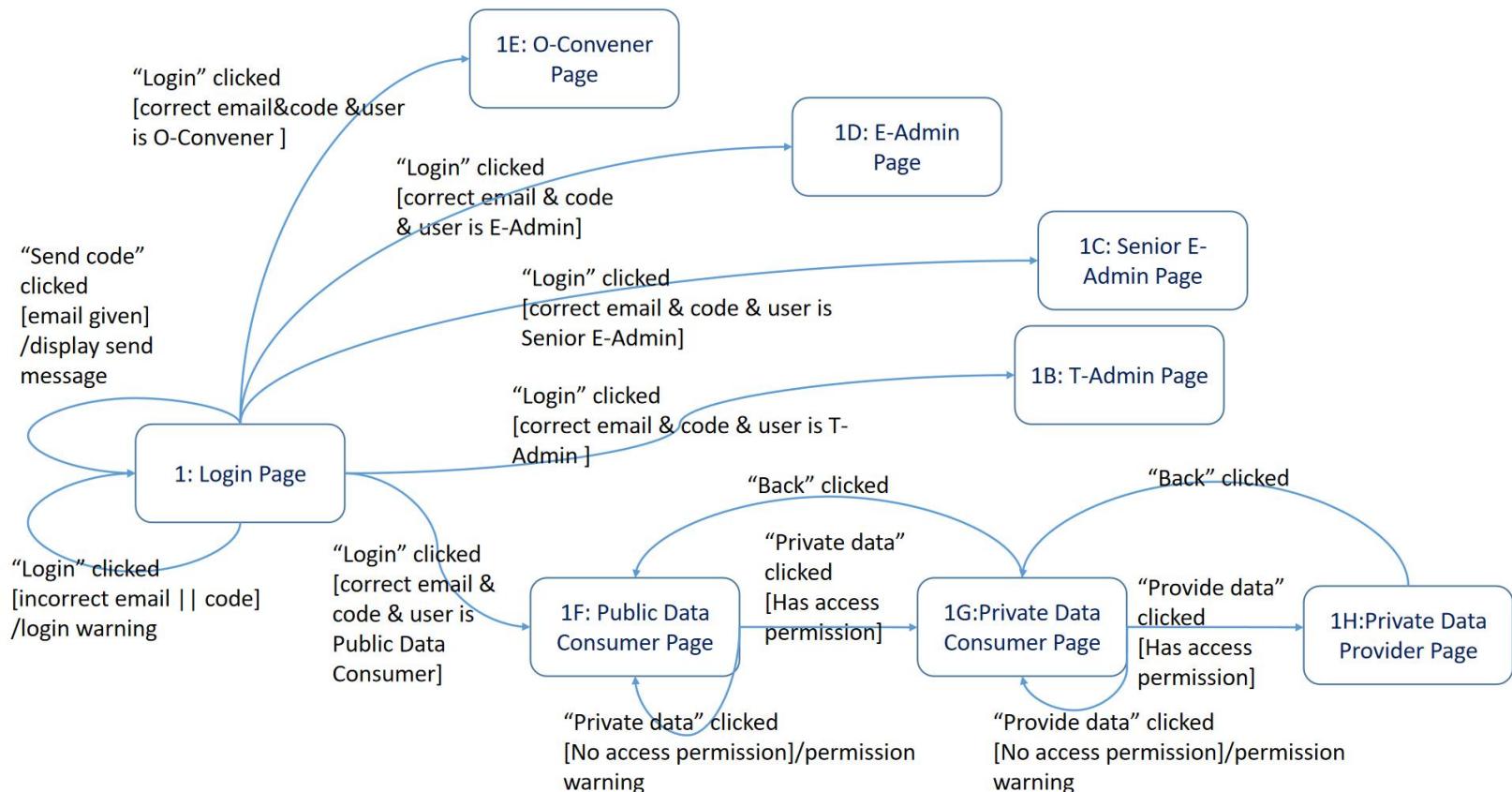


Figure 2. state transition diagram for "Login"

A basic scenario for "Login"

- User Goes to the Login Page.
- The user enters email
- The user clicks "Send code" to get message
- The user enters the correct code and clicks Login
- The system validates the email and code

- The system identifies the user's privileges.
- The system leads user to O-Convener Page if the user is O-Convener
- The system leads user to E-Admin Page if the user is E-Admin
- The system leads user to Senior E-Admin Page if the user is Senior E-Admin
- The system leads user to T-Admin Page if the user is T-Admin
- The system leads user to Public Data consumer Page if the user is Public Data consumer
- User Clicks Private data in the Public Data Consumer Page.
- The system checks the user's access rights and allows access to the Private Data Consumer Page.
- User Clicks Provide data in the Private Data Consumer Page.
- The system checks the user's access rights and allows access to the Private Data Provider Page.

3.1.3 Functional Requirements

- a. login warning will show : incorrect email or code!
- b. permission warning will show: you have no permission!

3.2 Register Organization

3.2.1 Description and Priority

The convener registers his or her organization. This involves providing a document to prove the convener represents the organization and the convener's email address.

3.2.2 Stimulus/Response Sequences

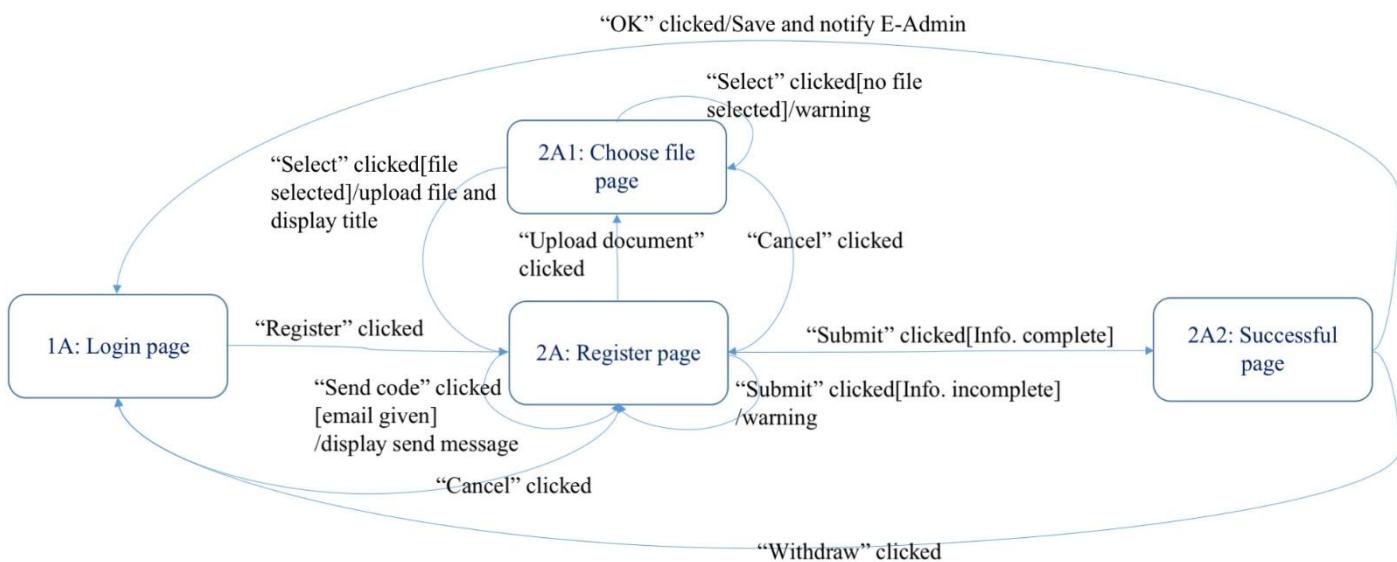


Figure 3. state transition diagram for “Register Organization”

A basic scenario

The O-Convener navigates to the login page
 The system displays the login and register options
 The O-Convener selects the register option
 The system presents the register page
 The O-Convener fills in their details and clicks “Send Code”
 The system sends a verification code to the user’s email
 The O-Convener click the file upload
 The system displays the file list selection option
 The O-Convener selects a file and clicks “select”
 The system displays the file title
 The O-Convener enters the verification code and submits the registration
 The system displays the successful page
 The O-Convener select the “OK”
 The system saves the file and the info, notify the E-Admin or Sr E-Admin then show the login page

3.2.3 Functional Requirements

None

3.3 Access Thesis

3.3.1 Description and Priority

The system allows public data consumer to download free shared papers, but some papers need to be paid for.

3.3.2 Stimulus/Response Sequences

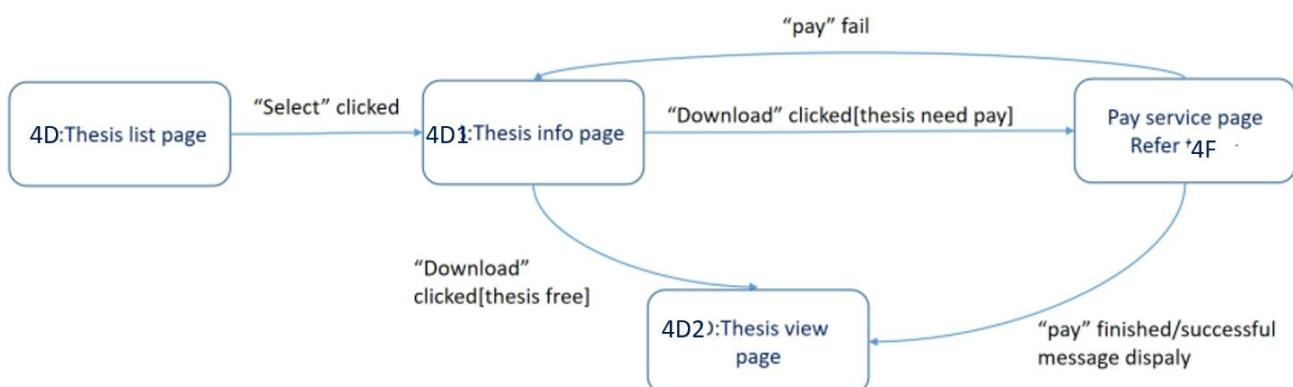


Figure 4. state transition diagram for “Access Thesis”

A basic scenario for “access thesis”

- Users browse a list of papers
- User selects a paper
- The system displays brief information about the paper, e.g. abstract, authors, field, etc.
- User clicks ‘download’ to download the paper.
- If payment is required to download the paper, the system jumps to 3.4 pay service function.
- If the payment is successful, the user successfully downloads the body of the paper and enters the body page of the paper.
- If the paper does not require payment, the user can directly download the main text of the paper and go to the main text page of the paper.

3.3.3 Functional Requirements

None

3.4 Pay Service

3.4.1 Description and Priority

The Pay Service use case allows Public Data Consumers and O-Conveners to complete payments for accessing services. In the initial version, only money transfer is supported, making this functionality essential for revenue generation.

3.4.2 Stimulus/Response Sequences

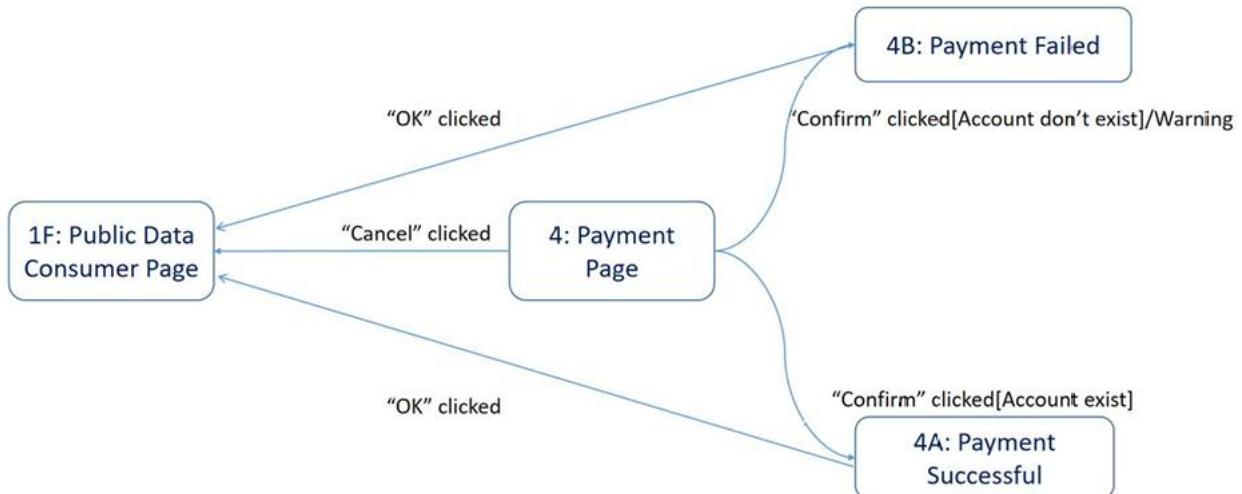


Figure 5. state transition diagram for “Pay Service”

A basic scenario

The Public Data Consumer selects the desired Thesis to purchase.

The system provides the payment options (only money transfer in the initial version).

The system verifies the payment details and processes the transaction.

If the payment is successfully completed, the system download the Thesis and notifies the Public Data Consumer.

If the payment fails, the system notifies the Public Data Consumer and prompts for contact with O-Convenor

3.4.3 Functional Requirements

REQ-1: The system shall provide the payment options, limited to money transfer in the initial version.

REQ-2: Pay Service returns "success" or "fail" to the use case that includes Pay Service.

3.5 Provide Course Information

3.5.1 Description and Priority

Each organization can provide new courses information or edit the information of an existing course in its lists.

3.5.2 Stimulus/Response Sequences

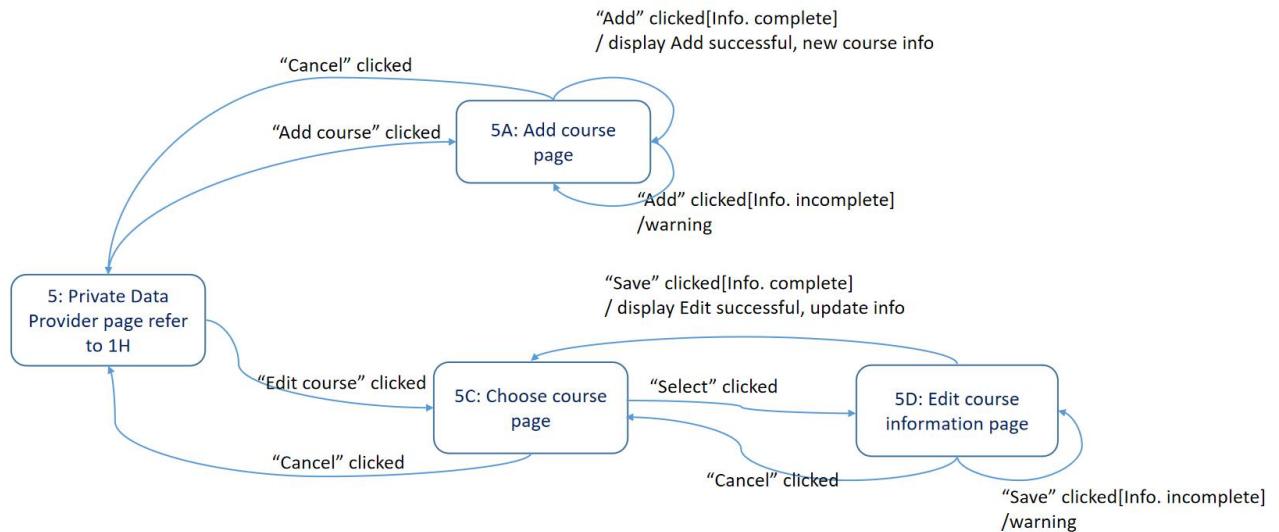


Figure 6. state transition diagram for "Provide Course Information"

A basic scenario

The private data provider chooses add course or edit course.

If choose add course, the system displays the boxes for new course info.

The private data provider enters new course info.

The system displays the new course in course list.

If choose edit course, the private data provider selects a course from existing list to edit.

The system displays the boxes for the selected course info.

The private data provider changes the course info.

The system displays the edited course in course list.

3.5.3 Functional Requirements

If there is a blank box appears when add or edit course information, a warning will appear.

3.6 Manage Own Workspace

3.6.1 Description and Priority

Each organization O-Convener can manage their own workspace, including adding and removing services. Currently, only two kinds of services can be provided: student identity authentication, thesis sharing.

3.6.2 Stimulus/Response Sequences

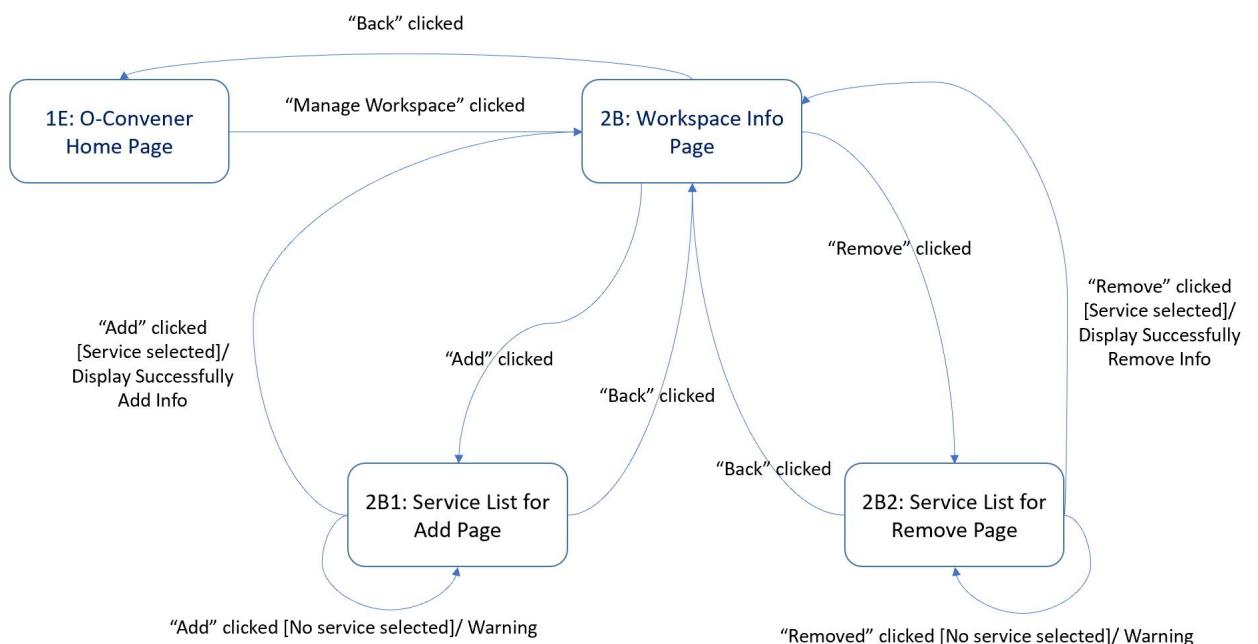


Figure 7. state transition diagram for “Manage Own Workspace”

Scenarios for "Manage Own Workspace" Use Case

Basic Scenario – Add a Service Successfully

- The O-Convener logs in and enters to O-Convener Home Page.
- The O-Convener clicks "Manage Workspace" from the Home Page.
- The system displays the Workspace Info Page.
- The O-Convener clicks "Add Service" and enter the Service List for Add Page.
- The system displays the services that allow to be added.
- The O-Convener selects some services and then clicks "Add".
- The system confirms with the Successfully Add Page and updates the workspace.

Basic Scenario – Remove a Service Successfully

- The O-Convener logs in and enters to O-Convener Home Page.
- The O-Convener clicks "Manage Workspace" from the Home Page.
- The system displays the Workspace Info Page.
- The O-Convener clicks "Remove Service" and enter the Service List for Add Page.
- The system displays the services that allow to be removed.
- The O-Convener selects a service and clicks "Remove".
- The system confirms with the Successfully Remove Page and updates the workspace.

3.6.3 Functional Requirements

REQ-1: After adding service, display "The services selected have been add to workspace."

"

REQ-2: After remove service, display "The services selected have been removed from workspace."

3.7 Manage Users

3.7.1 Description and Priority

Each O-Convener in his workspace can then submit the list of members in the organization who can access E-DBA. O-Convener can also change user permissions and help users pay fees.

3.7.2 Stimulus/Response Sequences

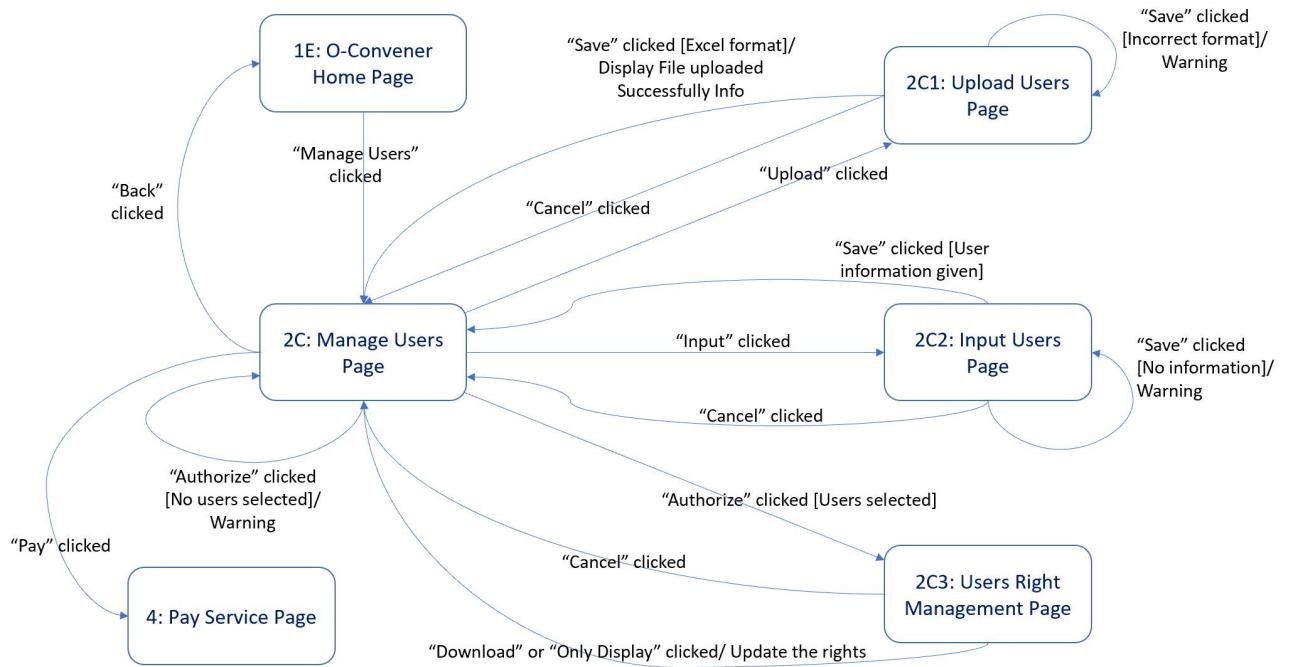


Figure 8. state transition diagram for “Manage Users”

Scenarios for "Manage Users" Use Case

Basic Scenario – Authorize Selected Users

- The O-Convener logs in and enters to O-Convener Home Page.
- The O-Convener clicks "Manage Users" from the O-Convener Home Page.
- The system displays the Manage Users Page.
- The O-Convener selects users and clicks "Authorize".
- The system displays the right of "Only Display" and "Download".
- The O-Convener clicked "Only Display" or "Download"
- The System updates the rights.

Basic Scenario – Upload Users via Excel

- The O-Convener logs in and enters to O-Convener Home Page.
- The O-Convener clicks "Manage Users" from the O-Convener Home Page.
- The system displays the Manage Users Page.
- The O-Convener clicks "Upload" and enter the Upload Users Page.
- The O-Convener uploads a valid Excel file on the Upload Users Page.

- The system validates the format and displays "File Uploaded Successfully Page".

Basic Scenario – Input Users Manually

- The O-Convener logs in and enters to O-Convener Home Page.
- The O-Convener clicks "Manage Users" from the O-Convener Home Page.
- The system displays the Manage Users Page.
- The O-Convener clicks "Input" on the Manage Users Page.
- The system navigates to the Input Users Page.
- The O-Convener enters user details (name, email, rights) and clicks "Save".
- The system confirms and updates the Pages.

Basic Scenario – Pay Users Fee

- The O-Convener logs in and enters to O-Convener Home Page.
- The O-Convener clicks "Manage Users" from the O-Convener Home Page.
- The system displays the Manage Users Page.
- The O-Convener clicks "Pay" on the Manage Users Page.
- The system navigates to the Pay Service Page for payment configuration.

3.7.3 Functional Requirements

REQ-1: After uploading correct file format, display file uploaded successfully info.

3.8 Check Organization Log

3.8.1 Description and Priority

O-Convener can check the log of any activities, services related to its own organization.

3.8.2 Stimulus/Response Sequences

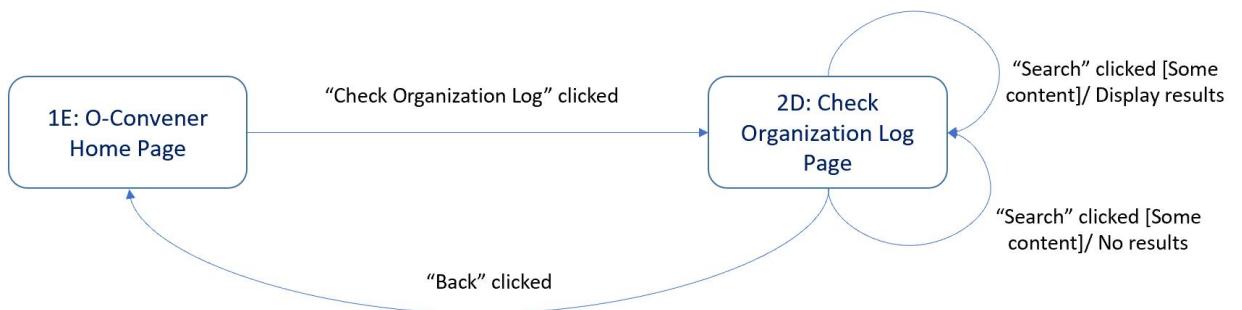


Figure 10. state transition diagram for “Check Organization Log”

Basic Scenario – Search Logs Successfully

- The O-Convener logs in and enters to O-Convener Home Page.
- The O-Convener clicks "Check Organization Log" from the O-Convener Home Page.
- The system displays the Check Organization Log Page.
- The O-Convener enters search criteria and clicks "Search".
- The system displays matching log entries.

3.8.3 Functional Requirements

None

3.9 Check System Log

3.9.1 Description and Priority

E-Admin can check system logs and organization logs for error monitoring. E-Admin can search system log in certain date or type by enter keyword.

3.9.2 Stimulus/Response Sequences

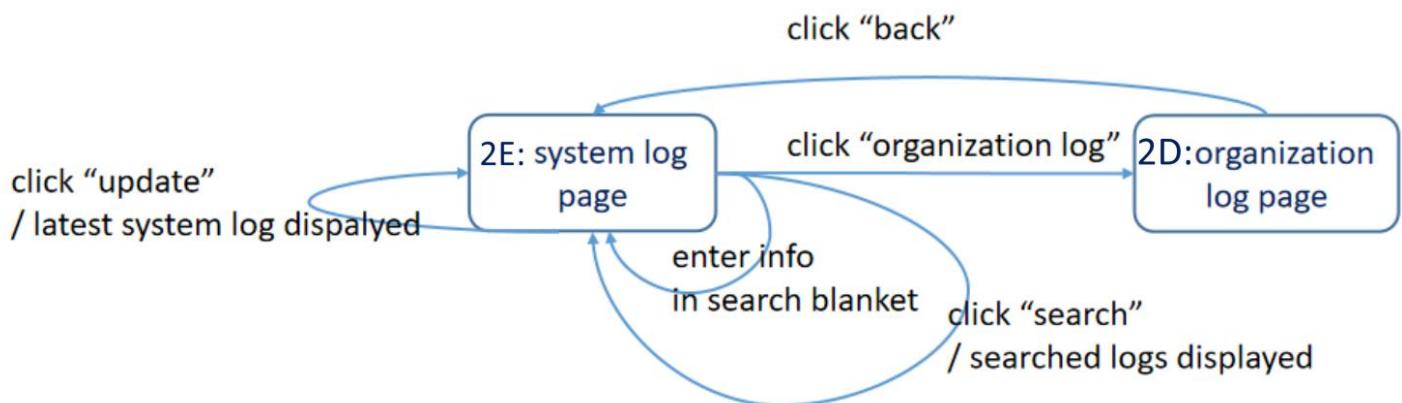


Figure 11. state transition diagram for “Check System Log”

A basic scenario for “Check System Log”

- user view current system log list
- user clicks "update"
- system display the latest system logs
- user enter key word in search blanket
- user click "search"
- system displays filtered system logs
- user click "organization log"

- system turn to organization log page

3.9.3 Functional Requirements

None

3.10 Explain Data Sharing Policy

3.10.1 Description and Priority

E-Admin can upload and update data sharing policies for other users to view.

3.10.2 Stimulus/Response Sequences

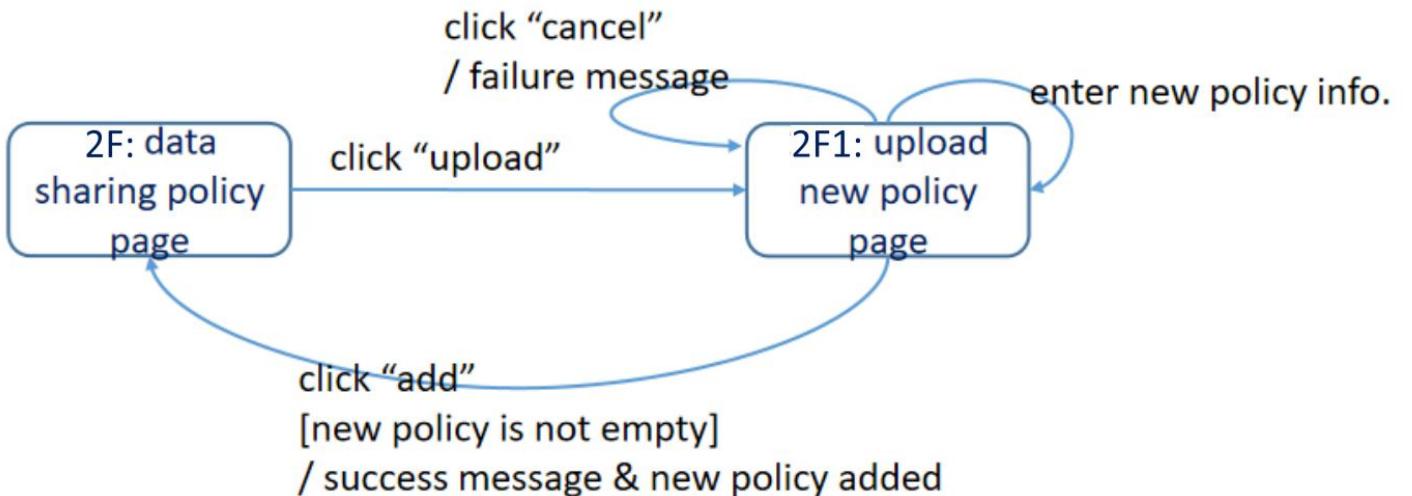


Figure 11. state transition diagram for “Explain Data Sharing Policy”

A basic scenario for “Explain Data Sharing Policy”

- user view current data sharing policies
- user click "upload"
- if the user is E-Admin, system will turn to upload new policy page
- user enter new policy information
- if user click "add"
- system display successful message and turn to update data sharing policy page
- if user click "cancel"
- system display failure message and back to upload new policy page

3.10.3 Functional Requirements

- a. Success message will show: New policy is uploaded successfully!

- b. Failure message will show: Upload is canceled! 3.11 Provide Course Information

3.11 Approve/Reject Application

3.11.1 Description and Priority

E-Admin can approve or reject the applications for registration from O-conveners. Once E-Admin approve one application, it will be sent to Senior E-Admin for final approval.

3.11.2 Stimulus/Response Sequences

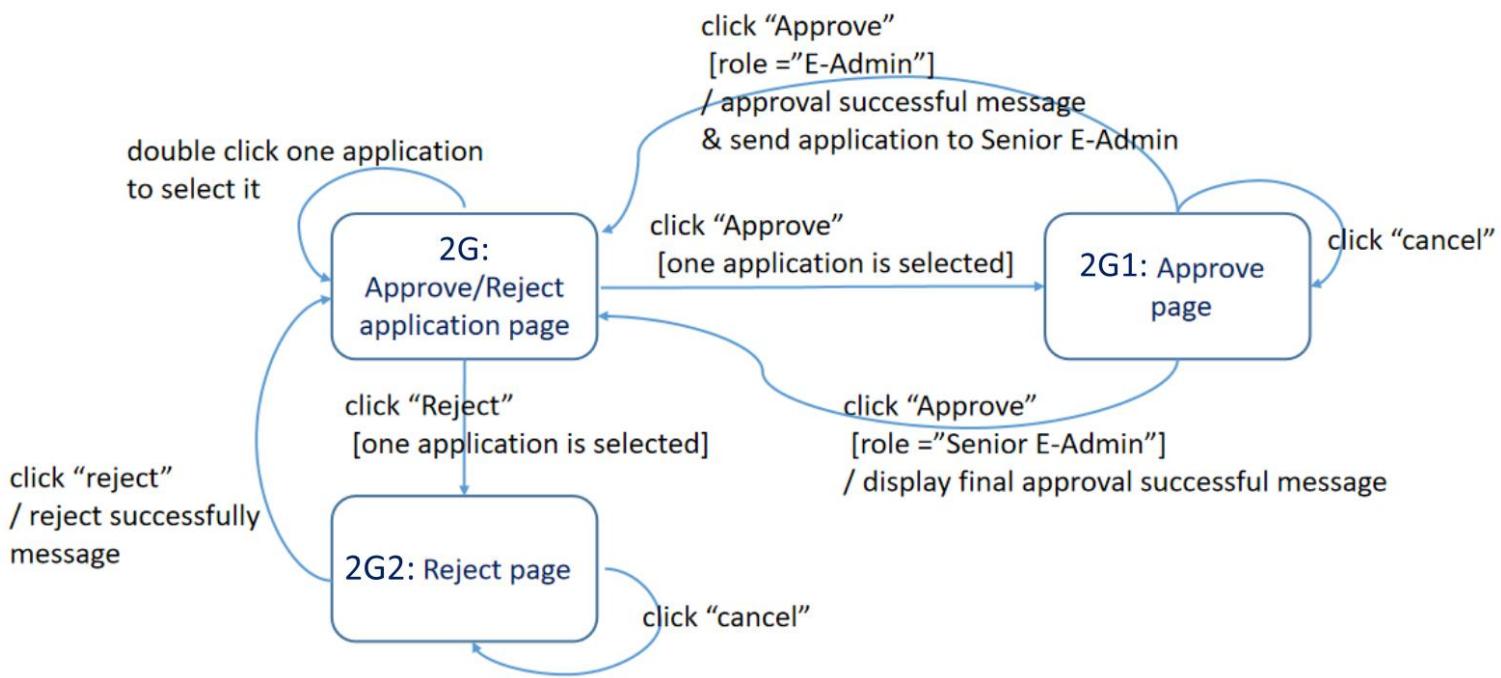


Figure 12. state transition diagram for “Approve/Reject Application”

A basic scenario for approve application

- user double click one application to select it
- user click "Approve"
- system turn to Approve page
- user click "Approve"
- if user is E-Admin, system will display successful message and send application to Senior E-Admin
- if user is Senior E-Admin, system will display successful approval message
- system back to Approve/Reject application page

A basic scenario for reject application

- user double click one application to select it
- user click "Reject"
- system turn to Reject page
- user click "Reject"
- system display successful rejection sage
- system back to Approve/Reject application page

3.11.3 Functional Requirements

- a. Reject successful message will show: This application is rejected!
- b. Approval successful message will show: This application is approved!
- c. Final approval successful message will show: Final approval of this application is done!

3.12 Answer Help Request

3.12.1 Description and Priority

T-Admin respond to and answer the help requests from other users on the problems in using the system.

3.12.2 Stimulus/Response Sequences

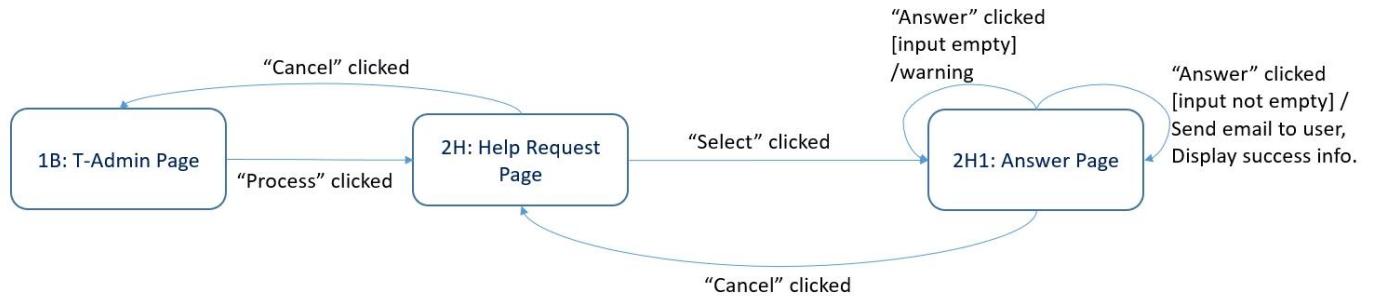


Figure 13. state transition diagram for “Answer Help Request”

A basic scenario

T-Admin choose to process the help request from users.

The system displays the pending and solved help request lists.

T-Admin select request from pending help request list to answer.

The system displays the help request information and answer box.

T-Admin answer the request.

The system sends e-mail to the user and display success info.

3.12.3 Functional Requirements

REQ-1: Display success info. is “Answer Successful!” in Answer Page.

3.13 Manage E-Admin Accounts

3.13.1 Description and Priority

T-Admin can set up two management administrators, E-Admin and Senior E-Admin. And can also delete their accounts, but should set up new accounts.

3.13.2 Stimulus/Response Sequences

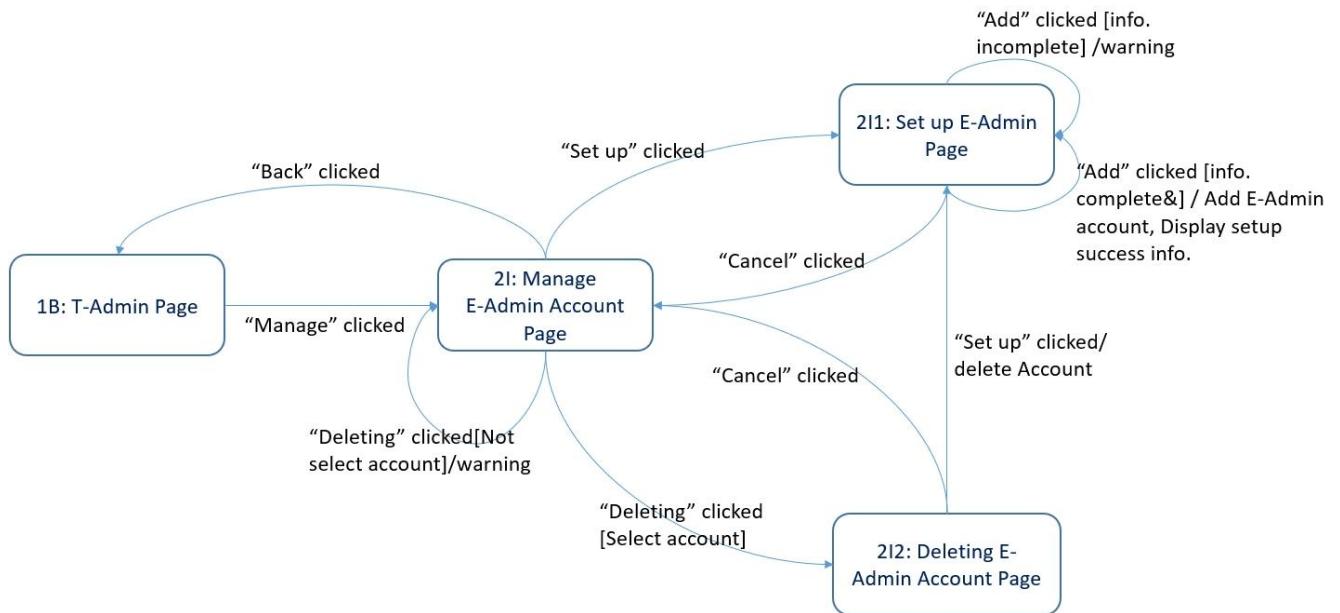


Figure 14. state transition diagram for “Manage E-Admin Accounts”

A basic scenario

T-Admin choose to manage E-Admin accounts.

The system displays E-Admin accounts info.

If T-Admin choose to set up E-Admin, the system displays the boxes for new account info.

The T-Admin enter new E-Admin account info.

The system adds E-Admin accounts and display in account lists.

If T-Admin select an account and choose to delete it, the system deletes the accounts and update in account lists.

After deleting the T-Admin should set up new E-Admin.

3.13.3 Functional Requirements

REQ-1: Display set up success info. is “Set up E-Admin successful!” in Set up E-Admin Page.

3.14 Manage Vault Application*

3.14.1 Description and Priority

TBD

3.15 Browse Policy

3.15.1 Description and Priority

Public Data Consumer can browse existing policies listed in the system.

3.15.2 Stimulus/Response Sequences



Figure 15. state transition diagram for “Browse Polciy”

A basic scenario

The user clicks "Browse Policy."

The system displays the policy list.

The user views the policy list.

The user clicks "Back."

The system returns to the previous page.

3.15.3 Functional Requirements

Functional Requirements

If there is a failure in access the policy list, an error message will appear.

3.16 Seek Help

3.16.1 Description and Priority

Public Data Consumer can submit a help request, view pending requests, and review resolved issues.

3.16.2 Stimulus/Response Sequences

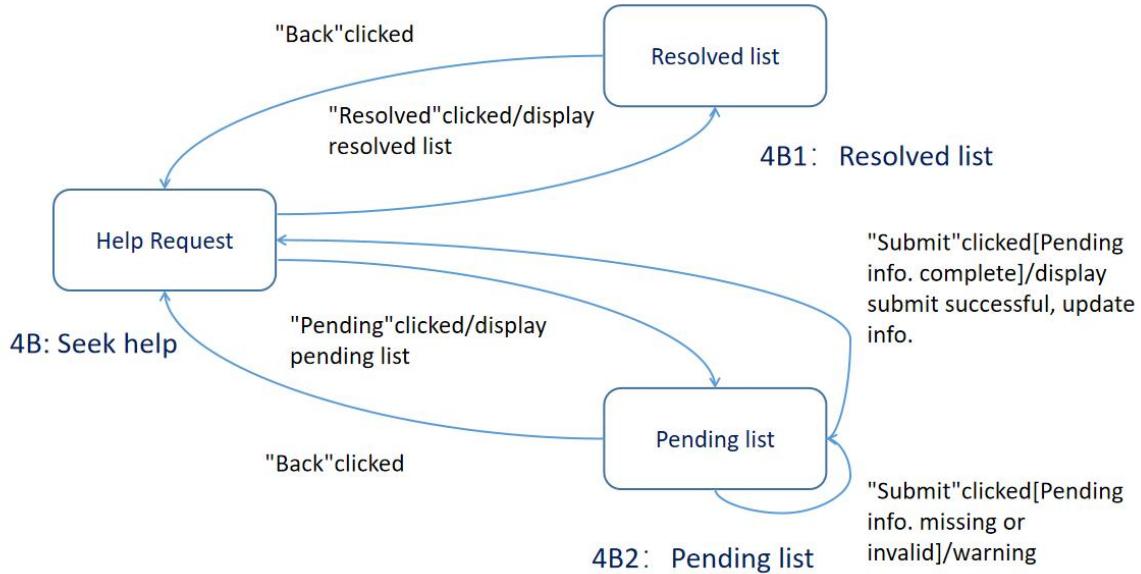


Figure 16. state transition diagram for “Provide Course Information”

A basic scenario

The user can view the pending list by clicking "Pending."

The system displays the pending help requests.

The user submits a new help request.

If the request information is complete, the system updates the status and displays "Submit Successful."

The user clicks "Back."

The system returns to the previous page.

If the request information is incomplete or invalid, the system displays a warning.

The user can view resolved requests by clicking "Resolved."

The system displays the resolved list.

The user clicks "Back."

The system returns to the previous page.

3.16.3 Functional Requirements

If input are missing in the help request, a warning will be displayed.

3.17 Access Course Information

3.17.1 Description and Priority

Public Data Consumer can access and view detailed course information from the system.

3.17.2 Stimulus/Response Sequences

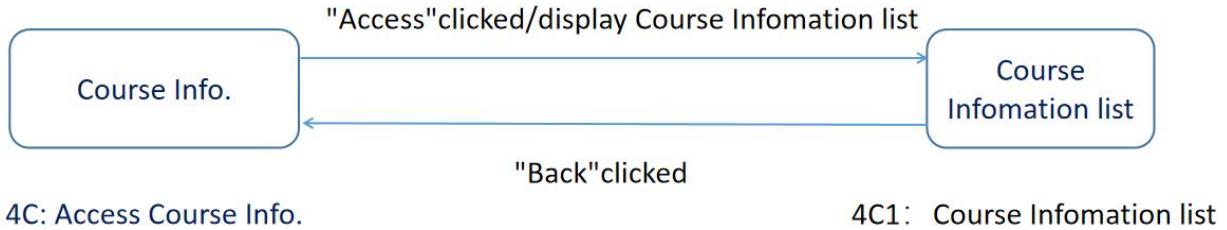


Figure 17. state transition diagram for “Access Course Information”

A basic scenario

The user clicks "Access"

The system displays the course information list.

The user selects a course to view detailed information.

The system displays the detailed course information.

The user clicks "Back."

The system returns to the previous page.

3.17.3 Functional Requirements

If there is a failure in access course information, an error message will appear.

3.18 Access Student Information

3.18.1 Description and Priority

Private Data Consumer can access student Info,check student identity and get student record.

3.18.2 Stimulus/Response Sequences

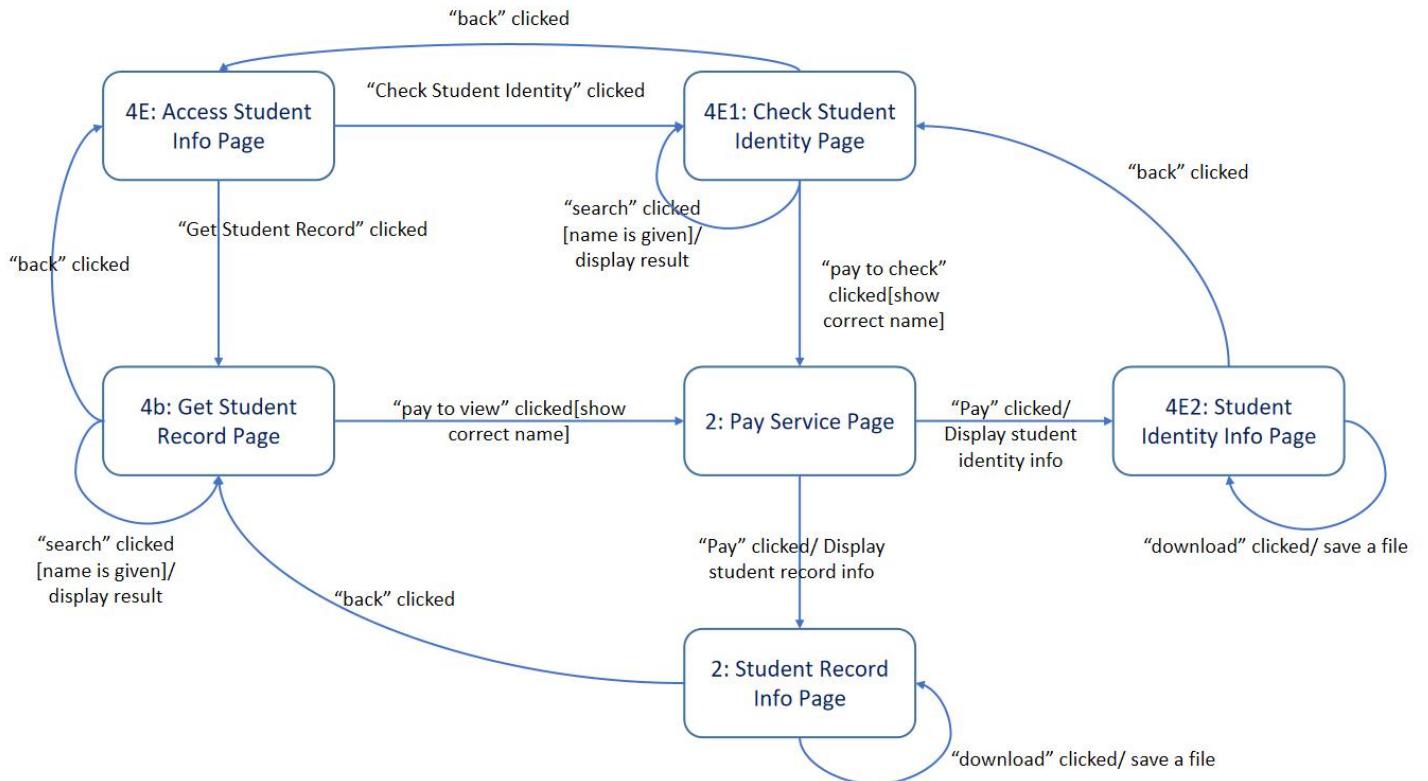


Figure 18. state transition diagram for “Access Student Information”

A basic scenario

If consumer choose to “check student identity”, the system displays the check student identity page.

Consumer search the student’s name.

The system displays pay service page.

Consumer pay to check student identity.

The systems show the student identity info.

If consumer choose to “get student record”, the system displays the check student identity page.

Consumer search the student’s name.

The system displays pay service page.

Consumer pay to get student record.

The systems show the student record info.

Consumer click download.

The systems download the student record info. and save as file.

3.18.3 Functional Requirements

None

3.19 Provide Database Interface information

3.19.1 Description and Priority

Enables data providers to configure database interface information for their organizations so that systems can interact and operate data through these interfaces.

3.19.2 Stimulus/Response Sequences

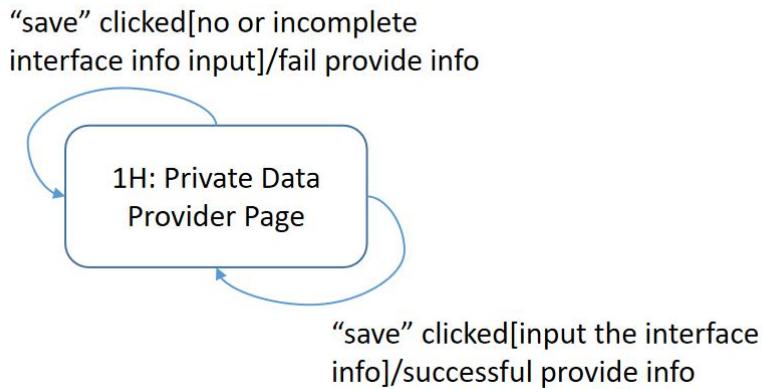


Figure 19. state transition diagram for “Provide Database Interface info.”

Basic scenario:

- The data provider enters the database interface information on the configuration form
- Click "save" to save the database interface info
- If the information is complete, the system displays successful info

3.19.3 Functional Requirements

None

4. External Interface Requirements

4.1 User Interfaces

Refer to UI.pdf

4.2 Hardware Interfaces

- Supported Device Types:
 - User Devices: The E-DBA platform can be accessed on various devices, including desktops, laptops, tablets, and smartphones, by students, administrators, and other users.
 - Server Hardware: The platform operates on dedicated servers or cloud servers, which manage the system, user data, and services like student authentication, thesis access, and payment processing.
 - Database Servers: Physical servers that store and manage the database, including student records, thesis information, and other data provided by educational organizations.
- Nature of Data and Control Interactions:
 - Data Exchange: Communication between user devices and the system takes place over HTTP/HTTPS for web applications or through APIs for mobile/desktop apps. This includes sensitive data such as user credentials, student identity, course details, and payment information.
 - Control Interactions: Control signals ensure that only authorized users have access to specific services like thesis sharing or student authentication. The control flow governs data access rights based on user roles (such as admin, data provider, or consumer).
 - Security Protocols: Protocols like HTTPS, OAuth, and TLS will be employed to ensure the secure exchange of data between devices, servers, and software systems.
- Communication Protocols:
 - HTTP/HTTPS: These protocols will be the main communication channels between the web client (browser) and the server for the E-DBA system.
 - RESTful APIs: The frontend (user interface) will communicate with the backend using REST APIs for user registration, thesis sharing, and data access, ensuring scalability and ease of integration.
 - Database Protocols: SQL protocols will be used to manage communication between the system and the database, facilitating the retrieval and storage of data such as user information, course schedules, and payment records.
- Interaction with External Devices:
 - Authentication Devices: The system integrates authentication mechanisms like biometrics or card-based systems, it will use specialized drivers or APIs to enable communication with the platform.
 - Payment Gateways: The system will interface with external payment systems (PayPal, Alipay) using APIs, following specific protocols such as HTTPS or REST for secure communication.

- File Upload/Download Devices: For large file exchanges (like theses or course materials), the platform will rely on cloud or local file storage systems. Protocols like FTP/SFTP or cloud storage APIs (AWS S3) will be used to facilitate transfers.

4.3 Software Interfaces

- Operating System Interfaces:

- Operating Systems: The platform will be hosted on servers running Windows Server 2019 or cloud environments (AWS or Azure). These operating systems will handle processes, memory management, and network communication for the platform.
- Communication Protocols: Standard protocols like HTTP/HTTPS, REST, and TCP/IP will enable internal communication between system components.

- Database Interfaces:

- Database Management Systems:
 - MySQL 8.0 : These relational database systems will manage structured data, such as user credentials, course data, thesis information, and payment records.
- Data Flow:
 - Incoming: The system will receive SQL operations for querying or updating the database, such as storing user information or logging organizational data.
 - Outgoing: The system retrieves data from the database for actions like displaying course options, verifying thesis access, or generating reports.
- Data Sharing: Data like user details and course information will be shared between multiple software components (the user interface, authentication service, and payment gateway).

- Library and Framework Interfaces:

- Backend Framework: Node.js 18.17.1 with Express.js 4.18.2 (for REST API development) will manage server-side logic and handle API calls for tasks like user registration, data retrieval, and authentication.
- Frontend Framework: React.js 18.2.0 will be used for building the user interface, interacting with the backend via REST APIs.
- Authentication and Authorization: OAuth2 will be employed for secure user authentication and role-based access control.
- Data Validation: Tools like Joi 17.7.0 (for Node.js) will validate incoming and outgoing data, ensuring accuracy and proper formatting.

- External API Interfaces:

- Payment Gateway Integrations: The system will integrate with third-party payment APIs like:
 - PayPal API
 - WeChat Pay API
 - Alipay API
 - Stripe API
- Data Flow:
 - Incoming: Payment details, including transaction status, will be received from payment gateways.

- Outgoing: Payment requests (transaction amount, payer information) will be sent to the relevant gateway, with responses returned containing payment status or transaction ID.
- File Storage APIs: The system will integrate with services AWS S3 for file uploads and retrieval.
 - Incoming: Upload requests for storing files.
 - Outgoing: Retrieval of stored files for access by authorized users.
- External Services and Data Provider Interfaces:
 - Student Authentication Service: The system may communicate with external student databases for identity verification, ensuring students' details are validated.
 - Incoming: Requests for identity verification using student names and IDs.
 - Outgoing: Responses indicating whether the student identity was successfully validated.
 - Thesis Database: The system will interface with thesis databases to retrieve documents based on metadata like titles and authors.
 - Incoming: Search queries for thesis data.
 - Outgoing: Responses containing the thesis metadata or document files.
- Data Sharing Mechanism:
 - Global Data Area: The system will utilize a global data area (shared memory or cache) to store session information and user data that needs to be accessed across different components of the system in real-time. This ensures that information like user login status, authentication tokens, or temporary session data is available to all parts of the system without the need for repetitive database queries. Shared Data Items: User data (name, email, role), course data, thesis metadata, student records, and payment details will be shared across the system.
 - Global Data Areas: Shared memory or cache (Redis 7.0.12) will store session information, ensuring consistency across different parts of the system (user login status).

4.4 Communications Interfaces

- Web Browser Communications:
 - Protocol: HTTP/HTTPS will be used to securely connect the browser to the E-DBA system, ensuring encrypted communication.
 - Message Formatting: Data will be exchanged in JSON format between the browser and the server, carrying user inputs, transaction statuses, or query results.
 - Security: SSL/TLS encryption will protect user data, with HSTS ensuring secure communication.
- Email Communications:
 - Protocols: SMTP will handle outgoing emails, while IMAP/POP3 will be used to retrieve emails.
 - Message Formatting: Emails will be in HTML format for a clean and professional look, with plain-text versions for compatibility.
 - Security: Emails containing sensitive data will be encrypted using S/MIME or PGP, and anti-spoofing protocols (DKIM and SPF) will protect against fraudulent emails.
- Network Server Communications:

- Protocol: Internal communication will occur over HTTPS using RESTful APIs for all operations (CRUD). WebSockets may be used for real-time updates.
- Data Flow: The backend will process incoming API requests and interact with databases using SQL queries.
- Security: API keys, OAuth2 will secure the communication, with TLS/SSL encryption applied to all data transfers.
- Electronic Forms (Web-based Forms):
 - Form Data: HTML forms will collect user data and submit it to the server via POST requests.
 - Security: Data validation will occur both on the client and server side, with CSRF protection implemented to prevent unauthorized requests.
- File Transfer and Storage:
 - Protocol: FTP or SFTP will be used for secure file uploads and transfers, and REST APIs will be used for storing/retrieving files from cloud storage.
 - Security: Files will be encrypted during transfer and stored securely using AES-256 encryption.
- Communication Security:
 - TLS/SSL: All communication between the client and server will be encrypted to maintain data integrity and privacy.
 - Firewall and Network Security: The system will employ firewalls and VPNs to safeguard against unauthorized access.
- Data Transfer Rates and Synchronization:
 - Data Transfer Rates: The system will support thousands of users with optimized database queries and efficient API calls.
 - Synchronization: Asynchronous tasks (file uploads, payments) will be managed using background jobs, and cache systems (Redis) will enhance response times. Real-time notifications will be pushed using WebSockets or similar technologies.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The system shall allow all student operations—including login, data retrieval, and online payment processing—to complete within 60 seconds under normal operating conditions.

- Under peak load conditions (up to 200 concurrent user sessions), the system shall maintain an average response time of less than 60 seconds for all critical operations.
- Performance tests shall be conducted using simulated loads (e.g., 150 concurrent users) to verify that these response time targets are met.

5.2 Safety Requirements

To ensure that no user loses any data while using the system (due to crashes, bugs, or other failures), the development team will release regular updates to address known issues. A bug tracking system will be available for users to report any problems encountered, enabling the team to resolve them in future releases. Additionally, the system will implement automated backup mechanisms (e.g., daily incremental backups and full weekly backups) to protect user data and ensure rapid recovery in case of unexpected failures. Fault-tolerant mechanisms, such as transaction rollback and redundant storage, will also be implemented to minimize data loss and maintain system stability.

5.3 Security Requirements

This system enforces robust security measures, including multi-factor authentication (MFA) for administrative and sensitive user access, and encrypt all data transmissions using industry-standard protocols (e.g., TLS 1.2 or higher). Sensitive user data, such as credentials, shall be stored using secure, salted hashing algorithms, and session tokens must include expiration times and secure management practices. The system shall implement role-based access control (RBAC) to restrict access to authorized users only, log and monitor all critical operations in real-time, and comply with applicable data protection regulations (e.g., GDPR) to ensure user privacy and data security.

5.4 Software Quality Attributes

- The system architecture shall be modular and scalable, allowing isolated testing of components such as the E-admin module and enabling straightforward maintenance and future enhancements.
- An integrated continuous testing framework shall be established, including automated unit, integration, performance, and security tests that must be passed before deployment.
- Real-time monitoring and logging mechanisms shall be implemented to capture system errors, performance metrics, and security incidents. These logs shall be reviewed periodically to detect anomalies.

6. Other Requirements

TBD

Appendix A: Glossary

| | |
|-------|-------------------------|
| E-DBA | Education Data Bay Area |
|-------|-------------------------|

Appendix B: Analysis Models

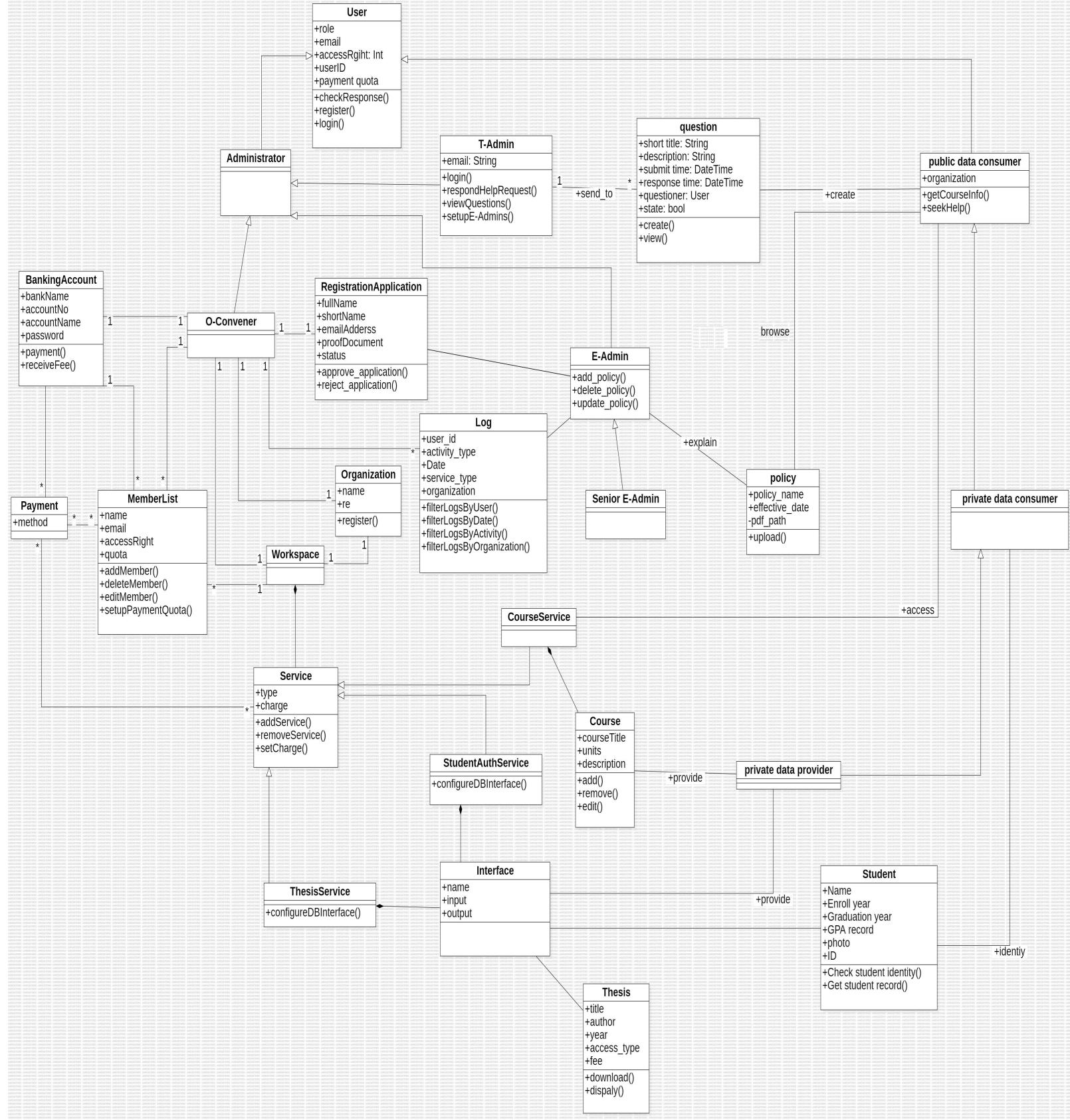


Figure 20. class diagram

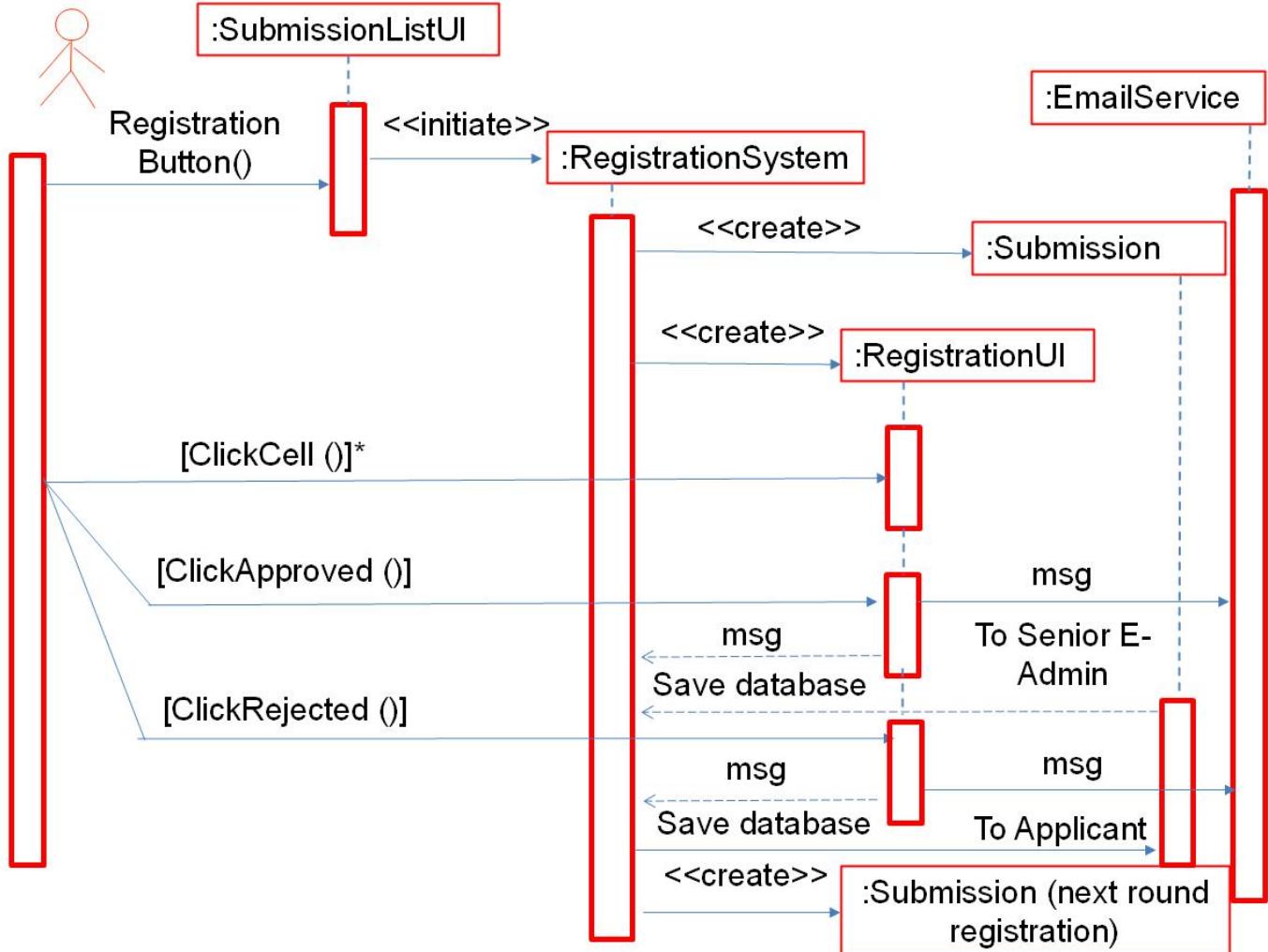


Figure 21. sequence diagram

Appendix C: Issues List

1. User Documentation is TBD. Figure 1 class diagram
2. Other Requirements part is TBD.
3. Analysis Models part is TBD.

