



성균관대학교
SUNG KYUN KWAN UNIVERSITY

AI-Tutoring CTF platform

Software Requirement Specification

2023.04.30

소프트웨어공학개론

41분반 Team 2

팀장 홍형근

팀원 김태영

김혜인

송재현

신영섭

임석현

목차

1. Introduction.....	3
1.1. Purpose.....	3
1.2 Scope.....	3
1.3. Definitions, Acronyms, and Abbreviation.....	4
1.4. References.....	6
1.5. Overview.....	6
2. Overall Description.....	6
2.1 Product Perspective.....	6
2.2. Product Functions.....	6
2.2.1. User Authentication.....	6
2.2.2. Admin Page Access.....	7
2.2.3. Problem Solving.....	7
2.2.4. User Info Editing.....	7
2.2.5. AI-Tutor.....	7
2.2.6. Problem Ranking.....	8
2.3. User Characteristics.....	8
2.3.1. Normal User.....	8
2.3.2. System Administrator.....	8
2.4. Constraints.....	8
2.5. Assumptions and Dependencies.....	9
3. Specific Requirements.....	9
3.1. Interface Requirements.....	9
3.1.1. User Interfaces.....	9
3.1.2. Hardware Interfaces.....	9
3.1.3. Software Interfaces.....	9
3.1.4. Communication Interfaces.....	10
3.1.5. Data Management Interfaces.....	10
3.2. Functional Requirements.....	10
3.2.1. Use Case.....	10
3.2.2. Use Case Diagram.....	24
3.2.3. Data Flow Diagram.....	25
3.3. Product Requirements.....	25
3.3.1. Usability Requirements.....	25
3.3.2. Performance Requirements.....	25
3.3.3. Security Requirements.....	26
3.4. Organizational Requirements.....	26
3.4.1. Environmental Requirements.....	26
3.4.2. Operational Requirements.....	26
3.5. External Requirements.....	27
3.5.1. Regulatory Requirements.....	27
3.5.2. Safety/Security Requirements.....	27

3.5.3. Ethical Requirements.....	27
3.6. Organizing System Flow.....	27
3.6.1. Context Model.....	28
3.6.2. Process Model.....	29
3.6.3. Interaction Model.....	29
3.6.4. Behavior Model.....	30
3.7. System Architecture.....	31
3.8. System Evolution.....	31
3.8.1. Assumption and Limitation.....	31
3.8.2. Evolution of Hardware and Software.....	31
3.8.3. Diverse Platforms.....	32
4. Supporting Information.....	32
4.1. Software Requirement Specification.....	32
4.2. Document History.....	32

List of Tables

Table 1. List of Terms and Definitions.....	6
Table 2. List of Acronyms and Abbreviations.....	6
Table 3. User Interface of Landing Page for the unregistered.....	11
Table 4. User Interface of Sign Up Page.....	13
Table 5. User Interface of Main Page.....	14
Table 6. User Interface of Problem Page.....	16
Table 7. User Interface of My Page.....	18
Table 8. User Interface of Edit Personal Information Page.....	20
Table 9. User Interface of Admin Page.....	21
Table 10. User Interface of Problem Manage Page.....	23
Table 11. User Interface of User Manage Page.....	24
Table 12. List of Use Case.....	26
Table 13. (F1) Show Landing Page.....	27
Table 14. (F2) Log In.....	28
Table 15. (F3) Sign In.....	29
Table 16. (F4) Log Out.....	30
Table 17. (F5) Show Main Page.....	30
Table 18. (F6) Show Problem List.....	31
Table 19. (F7) Show Top Solver List.....	32
Table 20. (F8) Show Admin Page.....	32

Table 21. (F9) Show Add & Edit Page.....	33
Table 22. (F10) Add New Problem.....	34
Table 23. (F11) Edit Existing Problem.....	35
Table 24. (F12) Show User Manage Page.....	36
Table 25. (F13) Give Admin Right.....	37
Table 26. (F14) Show Problem Page.....	38
Table 27. (F15) Pre-Catch Recommended Question.....	39
Table 28. (F16) Submit Answer.....	40
Table 29. (F17) Ask to AI-Tutor.....	41
Table 30. (F18) Show My Page.....	42
Table 31. (F19) Edit Personal Info.....	42
Table 32. Document History.....	52

List of Figures

Figure 1. Design of Landing Page for the unregistered.....	11
Figure 2. Design of Sign Up Page.....	12
Figure 3. Design of Main Page.....	14
Figure 4. Design of Problem Page.....	16
Figure 5. Design of My Page.....	18
Figure 6. Design of Edit Personal Information Page.....	19
Figure 7. Design of Admin Page.....	21
Figure 8. Design of Problem Manage Page.....	22
Figure 9. Design of User Manage Page.....	24
Figure 10. Use Case Diagram.....	44
Figure 11. Data Flow Diagram.....	45
Figure 12. Context Model.....	48
Figure 13. Process Model.....	49
Figure 14. Sequence Diagram.....	50
Figure 15. System Architecture.....	51

1. Introduction

1.1. Purpose

본 문서는 2023년 1학기 성균관대학교 소프트웨어공학개론 41분반 2팀에 의해 개발되는 “AI-Tutoring CTF Platform” 서비스의 소프트웨어 요구사항 명세서이다.

본 문서는 개발팀을 비롯한 성균관대학교 소프트웨어공학개론 수강생, 조교, 교수 모두가 열람할 수 있다. 누구나 본 문서를 재배포하거나 수정할 수 있지만, 상업적 용도로 활용 시 개발팀의 허가가 필요하다.

본 문서는 웹 어플리케이션 기반의 "AI-Tutoring CTF Platform" 서비스에 대한 개요와 요구 명세를 제시하려는 목적으로 작성되었다. 본 서비스는 사이버 보안을 공부하려는 사용자에게 도움이 되는 문제와 해결방안 등의 서비스를 제공한다. 이를 위해 웹 서버, 웹 인터페이스, 사용자 관리 시스템 등의 시스템이 개발되며 세부사항은 이하 문서에 명시한다.

1.2. Scope

본 서비스는 사이버 보안 분야의 입문자가 기본적인 지식과 이해를 얻을 수 있도록 고안되었다. 이를 위해 개발팀은 다양한 보안 문제와 해당하는 정답을 제공한다. 또한 본 서비스는 사용자의 학습 효율을 높이기 위해 AI-Tutor를 제공한다. AI-Tutor는 사용자와 상호작용하여 보다 깊은 학습을 수행하도록 유도한다. 또한 웹 환경으로 배포하여 실행 환경에서 비롯하는 이용 가능성을 증진시킨다. 이를 제공하기 위해 React 기반의 Front-end와 Nest.js 기반의 Back-end 서버를 구축한다.

본 서비스를 통해 사용자는 제공되는 문제를 해결하면서 보안에 대한 이해를 향상시킬 수 있다. 본 서비스는 확장성을 고려한 설계를 바탕으로 새로운 문제를 자유롭게 추가할 수 있다.

본 서비스는 사이버 보안 분야에 대한 지식과 이해를 높이기 위한 목적으로 제공된다. 본 서비스는 실제 시스템에 대한 해킹 등의 악의적인 행위를 유도하거나 장려하지 않는다.

1.3. Definitions, Acronyms, and Abbreviation

Table 1. List of Terms and Definitions

Term	Definition
Node.js	웹 애플리케이션을 만들기 위한 Javascript runtime 환경이다.
Nest.js	Node.js 런타임 위에서 동작하는 타입스크립트용 오픈소스 Back-end 웹 프레임워크이다.
React	User Interface를 만들기 위한 Javascript library이다.
Front-end	서비스가 사용자와 상호작용하는 모든 요소들의 집합이다.
Back-end	사용자가 볼 수 없는 영역(데이터베이스, 서버관리)에 해당하는 모든 요소들의 집합이다.
Web browser	웹에 접근하기 위한 소프트웨어이다. 예시로는 Chrome, Firefox, Safari 등이 있다.
CTF	Capture the Flag 방식으로 문제를 풀어 진행되는 해킹 대회에 사용되는 문제를 의미한다.
OpenAI	본 서비스에 사용되는 AI-Tutor에 사용되는 API를 제공하는 기업이다.
Ubuntu	Debian GNU/Linux기반 컴퓨터 운영체제이다.

Table 2. List of Acronyms and Abbreviations

Acronym & Abbreviation	Description
Admin	Administrator
AI	Artificial Intelligence

AWS EC2	Amazon Web Service Elastic Compute Cloud
API	Application Programming Interface
CSS	Cascading Style Sheets
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
JS	Javascript
VCS	Version Control System
DB	Database
PC	Personal Computer
RAM	Random-access Memory
CPU	Central Processing Unit
UML	Unified Modeling Language

1.4. References

- IEEE, “Recommended Practice for Software Requirements Specifications”.
<https://ieeexplore.ieee.org/document/720574>
- Facebook, “React”. <https://reactjs.org/>
- OpenJS Foundation, “Node.js”. <https://nodejs.org/>
- Nest, “Nest.js” <https://nestjs.com>
- 국가법령정보센터, “개인정보 보호법”.
<https://www.law.go.kr/%EB%B2%95%EB%A0%B9/%EA%B0%9C%EC%9D%B8%EC%A0%95%EB%B3%B4%EB%B3%B4%ED%98%B8%EB%B2%95>

1.5. Overview

본 소프트웨어 요구사항 명세서는 **Chapter 2~4**로 구성되어 있다. **Chapter 2**에서는 제품의 관점과 기능, 예상되는 사용자 특징, 제약 조건 및 의존성에 대해 서술한다. **Chapter 3**에서는 인터페이스와 세부기능에 대한 요구사항, 서비스 외적으로 고려되어야 할 요구사항, 시스템의 아키텍처 및 발전 가능성을 서술한다. **Chapter 4**에서는 변경 기록 등 본 문서에 대한 부가적인 정보를 서술한다.

2. Overall Description

2.1 Product Perspective

본 서비스는 사이버 보안 분야에 대한 입문자를 위해 개발되었다. 일반적으로 CTF 문제를 해결하기 위해서는 다양한 분석 툴이나 소프트웨어를 사용해야 하지만, 본 서비스의 사용자는 로컬 환경에 관계없이 웹 서비스와의 상호작용으로 문제를 해결할 수 있다. 이를 위해 사용자가 제출하는 답안은 객관식 또는 단답식으로 제한된다. 또한 기존 CTF 문제는 일반적으로 해당 문제에 대한 배경지식 없이는 접근이 어렵다. 이를 보완하기 위해 본 서비스는 **AI-Tutor**를 제공한다. **AI-Tutor**는 사용자의 질문에 대한 답변을 제공하여 사용자가 문제를 해결하는 데에 드는 부담을 경감한다.

2.2. Product Functions

2.2.1. User Authentication

사용자가 웹 브라우저를 통해 서비스에 접근하면, 서비스는 사용자에게 로그인 또는 계정 등록을 할 수 있는 인터페이스를 제공한다. 사용자는 기존의 계정으로 로그인을 하거나, 신규 계정을 생성할 수 있다. 계정의 비밀번호는 해시 암호화를 통해 서버의 데이터베이스에 저장된다.

2.2.2. Admin Page Access

로그인에 성공한 계정은 “**Normal User**” (이하 일반 사용자) 계정과 “**System Administrator**” (이하 관리자) 계정으로 구분된다. 관리자 계정은 일반적인 계정 등록을 통해 생성될 수 없으며, 다른 관리자에 의해 권한을 부여받을 수 있다. 공급자는 본 서비스의 원활한 운영을 위한 최초 관리자 계정을 제공받는다. 해당 계정을 통해 다른 관리자를 선정할 수 있다.

관리자 계정은 로그인 후 관리자 페이지로 접근할 수 있는 인터페이스를 제공받는다. 관리자는 해당 페이지에서 관리자 권한 부여와 서비스 내용 수정 작업을 수행할 수 있다.

2.2.3. Problem Solving

사용자가 로그인에 성공한 후, 문제를 선택할 수 있는 인터페이스를 제공받는다. 이후 사용자는 자신이 해결한 문제, 시도했지만 해결하지 못한 문제, 아직 시도하지 않은 문제를 확인할 수 있는 문제 목록을 볼 수 있고, 각 문제를 선택하여 풀 수 있다. 또한 사용자가 문제 리스트에서 원하는 문제를 고른 후 문제 페이지로 이동한 후, 문제에 대한 설명과 사용자의 문제 해결에 도움을 줄 수 있는 **AI-Tutor**가 제공된다. 문제에 대한 설명은 **figure**, 사진 등 다양한 리소스를 통해 제공될 수 있다. 사용자의 편의성을 위해, 문제의 정답은 객관식 또는 단답식 문제로 제공된다.

2.2.4. User Info Editing

일반 사용자가 로그인에 성공한 후, 해당 사용자의 개인 페이지에 접근할 수 있는 인터페이스를 제공받는다. 사용자는 개인 페이지에서 비밀번호 변경과 별명 변경 작업을 수행할 수 있다.

2.2.5. AI-Tutor

일반 사용자가 선택한 문제를 풀어 나갈 시에, 사전에 제공되는 문제와 관련된 **Recommended Question**을 포함한 질문 사항들을 **AI-Tutor**에게 질문하여 그에 대한 **AI-Tutor**의 답변을 제공받을 수 있다. 사용자가 **AI-Tutor**에게 해당 문제에 대해 입력한 질문 및 답변 내용들은 해당 문제 세션에 저장되어 사용자가 언제든지 질의 내용을 확인할 수 있다.

2.2.6. Leaderboard

모든 사용자는 **Main Page**상에서 당일 및 전체 기간 동안 문제를 가장 많이 해결한 사용자(**Top Solver**)의 목록을 확인 할 수 있다.

2.3. User Characteristics

2.3.1. Normal User

일반 사용자는 본 서비스를 이용하는 사용자 중, 관리자가 아닌 모든 사용자를 의미한다. 이들은 서비스에서 제공하는 문제를 통해 보안에 대한 지식과 역량을 기를 수 있다. 본 시스템에서 제공하는 **AI-Tutor**, 자체 제작 문제를 통해 문제를 해결하는데 도움을 받을 수 있다.

2.3.2. System Administrator

관리자는 **Normal User**가 할 수 있는 모든 기능에 대해 접근가능하며, 추가적으로 **Admin Page**에 접근하여 새로운 문제 추가, 기존 문제 수정, 유저 관리, 타 **User**에게 **admin** 권한 부여를 수행할 수 있다. 또한, 시스템 관리자는 서버에는 문제가 없는지, 보안 상의 약점은 없는지 끊임없이 모니터링 하고 해결책을 강구해 나가야만 한다.

2.4. Constraints

본 시스템은 기재된 요구 사항을 바탕으로 고안되고 구현된다. 본 문서에 명시되지 않은 세부적인 조건들은 개발자의 재량에 의해 구현되고 설계될 수 있다.

- 외부 **API**를 사용하는 경우 보안과 성능이 보장된 **API**를 사용할 것
- 사용자 친화적인 인터페이스를 구축할 것
- 라이선스를 확인 후 가능한 오픈소스를 주로 사용할 것
- 시스템의 리소스를 낭비하지 않고 개발할 것
- 추후 시스템의 확장을 고려하여 유지보수가 가능하도록 설계할 것

2.5. Assumptions and Dependencies

본 서비스는 문서에 기술된 하드웨어와 소프트웨어의 최소사양을 만족하는 환경에서 이용되는 것을 가정한다. 본 문서에 기술된 구동 환경을 만족하지 못한 환경에서는 정상적인 동작을 보장할 수 없다.

3. Specific Requirements

3.1. Interface Requirements

3.1.1. User Interfaces

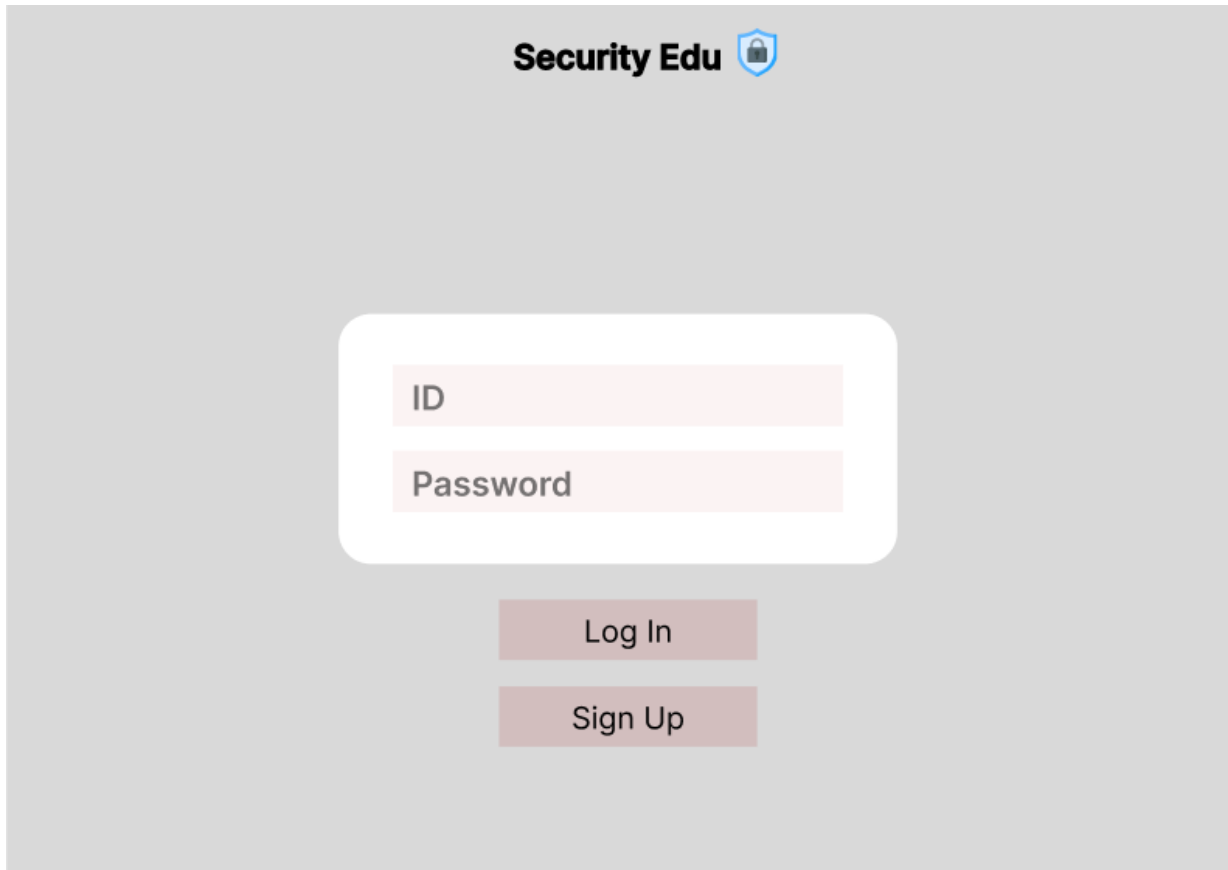


Figure 1. Design of Landing Page for the unregistered

Table 3. User Interface of Landing Page for unregistered

Name of Page	Landing Page for unregistered
목적 및 설명	사용자가 IP 주소 혹은 도메인 이름을 통해 서비스에 접근할 때 제공되는 화면이다. 인증되지 않은 사용자는 로그인 또는 회원가입을 통해 메인 페이지로 진입할 수 있다.
입력 소스, 출력대상	클라이언트 / 호스트 서버 호스트 서버 / 클라이언트
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리

구성 단위	화면(1920 * 1080)
시간 및 속도	페이지 로딩 처리: 네트워크 상태에 의존 페이지 렌더링 처리: 사용자 환경 성능에 의존 HTTP 요청 처리: 서버 상태에 의존
기타 입출력 관계	N/A
화면 형식과 구성	화면 상단에는 서비스의 이름이 나타난다. 화면 중앙에는 ID와 Password 입력란, Login 버튼, Sign up 버튼이 존재한다. 로그인을 통한 인증에 성공한 사용자는 메인 페이지로 이동한다. 로그인을 통한 인증에 실패했다면, 에러 팝업을 출력한다. Sign up 버튼을 클릭했을 경우 회원가입 페이지로 이동한다.
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동

The image shows a web page design for a sign-up form. At the top, there is a header with the text 'Security Edu' and a blue shield icon. Below the header, the page is titled 'Sign Up Page'. The form consists of four input fields stacked vertically, each with a label: 'ID', 'Password', 'Username', and 'Email'. At the bottom of the form is a blue button with the text 'Sign Up'.

Figure 2. Design of Sign Up Page

Table 4. User Interface of Sign Up Page

Name of Page	Sign Up Page
목적 및 설명	<p>상기 Landing Page for unregistered 화면에서 사용자가 'Sign up' 버튼을 클릭한 경우 Sign Up Page로 이동한다.</p> <p>사용자는 주어진 각 입력란에 정보를 입력한 후, Sign up 버튼을 눌러 회원가입을 할 수 있다.</p> <p>회원가입에 성공한 사용자는 별도의 로그인 절차 없이 메인 페이지로 리다이렉션된다.</p>
입력 소스, 출력대상	<p>클라이언트 / 호스트 서버</p> <p>호스트 서버 / 클라이언트</p>
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	<p>페이지 로딩 처리: 네트워크 상태에 의존</p> <p>페이지 렌더링 처리: 사용자 환경 성능에 의존</p> <p>HTTP 요청 처리: 서버 상태에 의존</p>
기타 입출력 관계	N/A
화면 형식과 구성	<p>사용자 ID, Password, username, email에 대한 입력란이 존재한다.</p> <p>입력받은 정보를 토대로 신규 계정을 생성하고 메인 페이지로 리다이렉션하는 Sign up 버튼이 있다.</p> <p>입력받은 정보의 중복성 문제로 인해 신규 계정 생성이 불가할 경우, 에러 팝업을 출력한다.</p> <p>신규 계정 생성이 성공하였을 경우, 회원가입 성공 팝업을 출력한다. 팝업을 닫으면 즉시 메인 페이지로 이동한다.</p>
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동

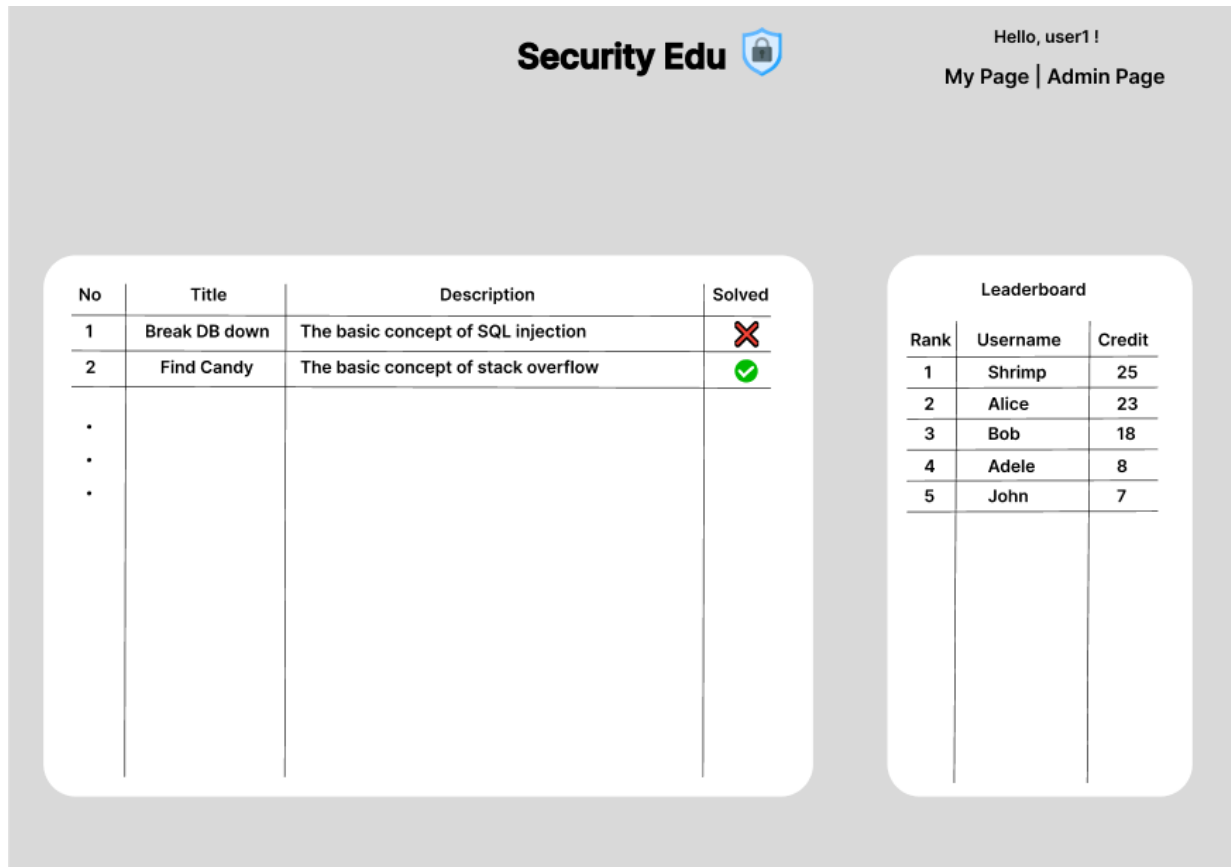


Figure 3. Design of Main Page

Table 5. User Interface of Sign Up Page

Name of Page	Main Page
목적 및 설명	<p>사용자는 우측 상단의 menu bar에 존재하는 각 페이지로 이동할 수 있다.</p> <p>사용자는 좌측의 문제 목록에서 문제를 선택할 수 있다.</p> <p>사용자는 우측의 리더보드에서 유저 랭킹을 확인할 수 있다.</p>
입력 소스, 출력대상	<p>클라이언트 / 호스트 서버</p> <p>호스트 서버 / 클라이언트</p>
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	<p>페이지 로딩 처리: 네트워크 상태 및 데이터베이스 서버 상태에 의존</p> <p>페이지 렌더링 처리: 사용자 환경 성능에 의존</p>

	HTTP 요청 처리: 서버 상태에 의존
기타 입출력 관계	문제 목록 및 사용자의 정답여부를 불러오기 위해 데이터베이스로부터 필요한 값들을 받는다. Leaderboard에 표기할 항목들을 불러오기 위해 데이터베이스로부터 필요한 값들을 받는다.
화면 형식과 구성	상단 사이트 제목을 클릭할 시 Main Page 로 이동한다. 본 페이지가 Main Page 인 관계로 새로고침 기능과 동일하다. 우측 상단의 menu bar 에서 각 텍스트를 클릭할 시 해당 페이지로 이동한다. 단, 일반 사용자가 Admin Page 를 클릭할 경우 권한 없음 팝업이 출력된다. 좌측의 문제 목록에서 No, Title, Description 의 텍스트를 클릭 할 시 해당 문제 페이지로 이동한다. 우측의 Leaderboard 는 전체 기간의 Top Solver 를 내림차순으로 보여준다.
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 위젯 클릭을 통한 페이지 이동

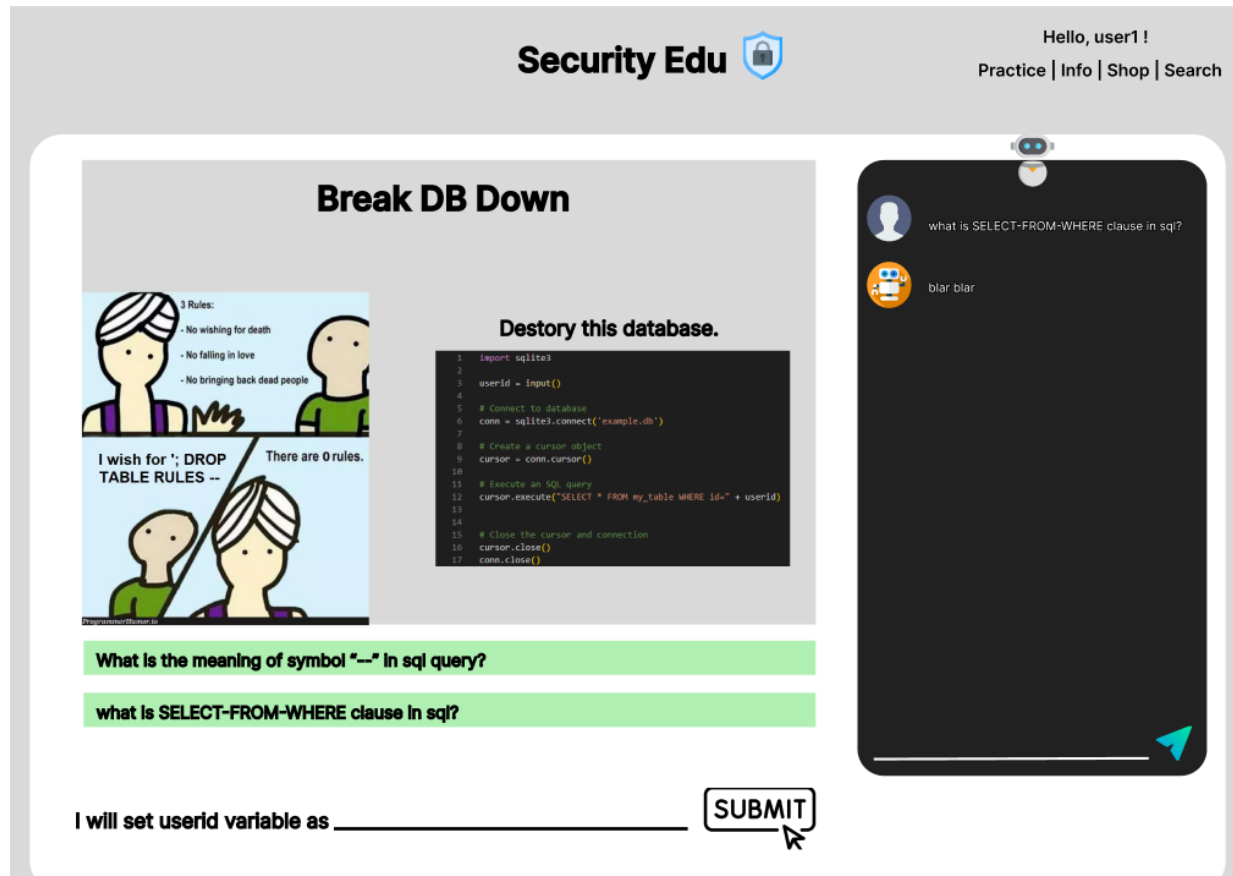



Figure 4. Design of Problem Page

Table 6. User Interface of Problem Page

Name of Page	Problem Page
목적 및 설명	<p>사용자가 선택한 문제에 대한 설명과 AI-Tutor를 제공해주는 page이다. 사용자는 문제에 대한 답을 하단 Submit 버튼을 이용하여 제출할 수 있다.</p> <p>사용자는 우측 AI-tutor Interface를 통해 AI-tutor에게 문제와 관련된 내용들을 질문할 수 있다.</p>
입력 소스, 출력대상	<p>클라이언트 / 호스트 서버</p> <p>호스트 서버 / 클라이언트</p>
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	<p>페이지 로딩 처리: 네트워크 상태 및 데이터베이스 서버 상태에 의존</p> <p>페이지 렌더링 처리: 사용자 환경 성능에 의존</p> <p>HTTP 요청 처리: 서버 상태 및 openAI API 상태에 의존</p>

기타 입출력 관계	<p>CTF problem에 대한 설명과 추천 질문을 데이터베이스로부터 불러온다.</p> <p>사용자가 AI-Tutor와 상호작용한 질의 기록을 데이터베이스로부터 불러온다.</p> <p>사용자가 submit 버튼을 통해 제출한 답이 문제의 답과 일치한다면 이를 데이터베이스로 전달하여 해당 문제를 정답을 맞춘 문제로 변경한다.</p>
화면 형식과 구성	<p>상단 Security Edu 아이콘은 Main Page로 링크된다.</p> <p>좌측 화면에는 문제에 대한 내용 설명이 표시되며 이에 대한 정답을 하단 Submit 버튼 옆 입력 box에 사용자가 입력한 뒤 submit 버튼을 click하여 정답인지 아닌지를 확인하게 된다.</p> <p>우측 AI-Tutor interface에서는 사용자가 해당 문제에 대해 질문한 내용들과 지난 번 시도했던 질문들에 대한 내용이 표시된다.</p> <p>AI-Tutor interface 입력란을 통해 질문을 작성하고, 키보드의 엔터 버튼이나 전송 아이콘을 클릭하여 응답을 받을 수 있다.</p> <p>연녹색 배경의 추천 질문 버튼을 클릭하면, 해당 질문의 텍스트가 AI-Tutor interface 입력란에 채워진다.</p>
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동

Security Edu 

My Page

Correct

1,3,5,8

Incorrect

2,4

Rate

66.67%

My Ranking

15th

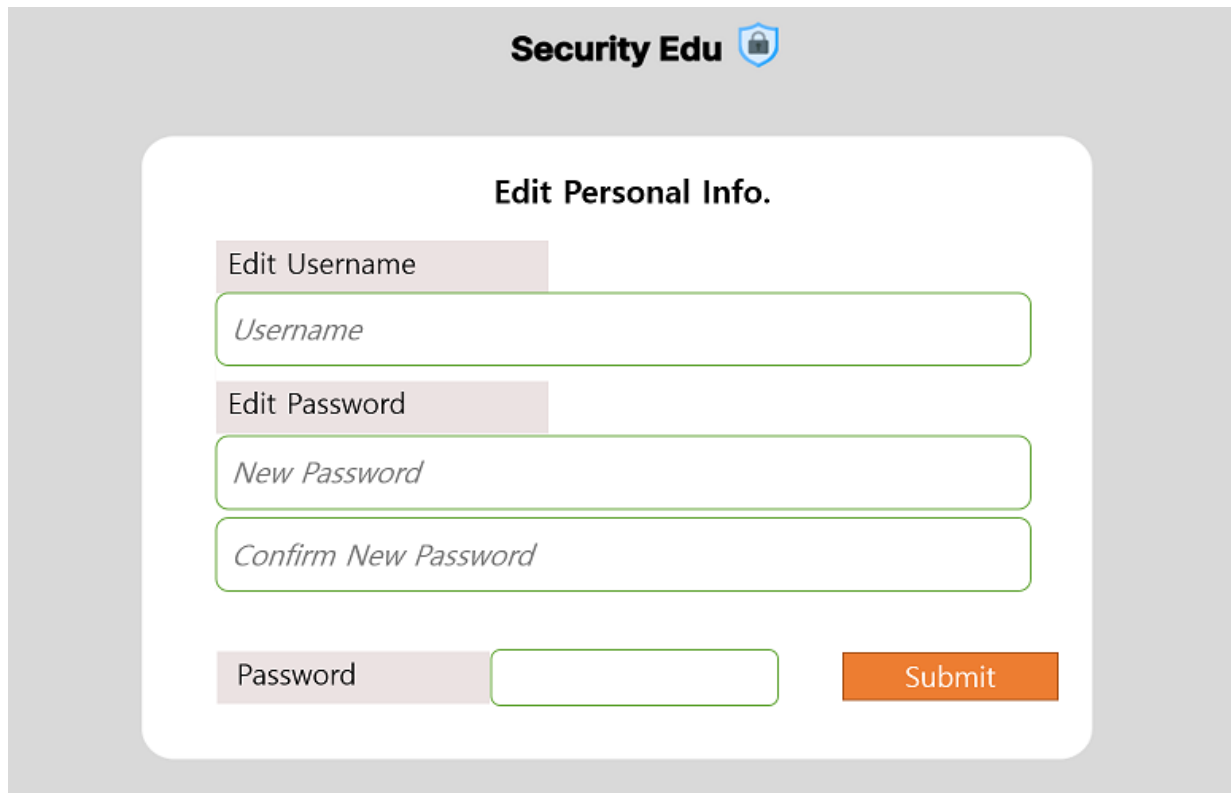
Edit My Info.

Figure 5. Design of My Page

Table 7. User Interface of My Page

Name of Page	My Page
목적 및 설명	<p>사용자는 Main Page에서 텍스트 클릭을 통해 My Page로 이동할 수 있다.</p> <p>사용자는 해당 페이지에서 자신이 풀었던 문제에 대한 정답 여부와 정답률, 그리고 자신의 랭킹을 조회할 수 있다.</p> <p>사용자는 Edit My Info. 버튼을 통해 수정 페이지로 이동할 수 있다.</p>
입력 소스, 출력대상	<p>클라이언트 / 호스트 서버</p> <p>호스트 서버 / 클라이언트</p>
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	<p>페이지 로딩 처리: 네트워크 상태 및 데이터베이스 서버 상태에 의존</p> <p>페이지 렌더링 처리: 사용자 환경 성능에 의존</p> <p>HTTP 요청 처리: 서버 상태에 의존</p>

기타 입출력 관계	사용자와 상호작용한 문제들에 대한 필요한 정보를 데이터베이스로부터 받는다.
화면 형식과 구성	상단 Security Edu 아이콘은 Main Page 로 링크된다. 화면에는 사용자가 시도했던 문제 번호들이 표시된다. 표시되는 문제 번호들은 맞은 문제, 시도했으나 틀린 문제들이다. 시도하지 않은 문제들은 표시되지 않는다. 문제 번호 하단에는 정답률 및 랭킹 정보가 표시된다. 하단 Edit My Info. 버튼은 Edit Personal Information Page 로 링크된다.
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동



The image shows a web interface for "Security Edu" with a shield icon. The main heading is "Edit Personal Info.". Below this, there are three input fields with labels: "Edit Username" (with placeholder "Username"), "Edit Password" (with placeholder "New Password"), and "Confirm New Password". At the bottom, there is a "Password" label next to an input field, and an orange "Submit" button.

Figure 6. Design of Edit Personal Information Page

Table 8. User Interface of Edit Personal Information Page

Name of Page	Edit Personal Information Page
목적 및 설명	사용자가 기존 username 과 비밀번호를 수정하기 위한 페이지이다.
입력 소스, 출력대상	클라이언트 / 호스트 서버 호스트 서버 / 클라이언트
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	페이지 로딩 처리: 네트워크 상태 및 데이터베이스 서버 상태에 의존 페이지 렌더링 처리: 사용자 환경 성능에 의존 HTTP 요청 처리: 서버 상태에 의존
기타 입출력 관계	사용자의 수정 요청을 할 때, 기존 비밀번호가 정확히 입력되었는지를 사용자 데이터베이스에서 확인한다. 기존 비밀번호가 적절히 입력되었을 때, 사용자가 요청한 수정사항을 데이터베이스에 반영한다.
화면 형식과 구성	상단 Security Edu 아이콘은 Main Page 로 링크된다. 화면에는 username 수정란, 비밀번호 수정란, 기존 비밀번호 입력란이 나타난다. 사용자가 username 변경을 원하는 경우, username 수정란에 자신이 원하는 username 을 입력한다. 사용자가 비밀번호 변경을 원하는 경우, 비밀번호 수정란에 원하는 비밀번호를 입력한다. 오탈자를 방지하기 위하여 비밀번호 확인란도 작성하도록 한다. 사용자가 변경한 사항들을 반영하고자 하는 경우, 기존 비밀번호 입력란에 기존 비밀번호를 입력한 뒤, 수정 완료 버튼을 클릭하여 정보를 갱신한다.
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동

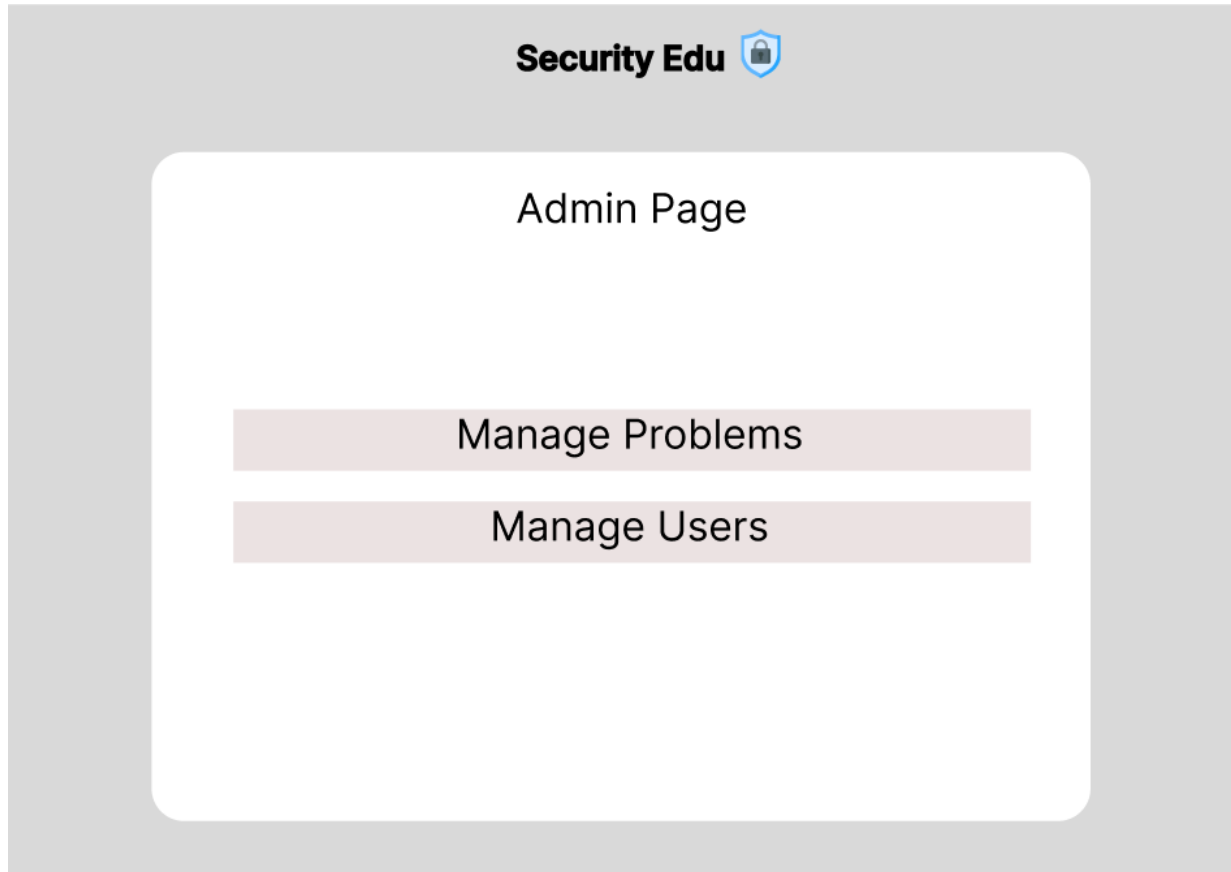


Figure 7. Design of Admin Page

Table 9. User Interface of Admin Page

Name of Page	Admin Page
목적 및 설명	관리자는 문제와 사용자의 관리를 위해 Admin Page에 접근할 수 있다.
입력 소스, 출력대상	클라이언트 / 호스트 서버 호스트 서버 / 클라이언트
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	페이지 로딩 처리: 네트워크 상태 및 데이터베이스 서버 상태에 의존

	페이지 렌더링 처리: 사용자 환경 성능에 의존 HTTP 요청 처리: 서버 상태에 의존
기타 입출력 관계	N/A
화면 형식과 구성	화면 상단에 Page 제목인 Admin Page 가 나타난다. Manage Problems 버튼을 클릭할 시 Problem Manage Page 로 이동한다. Manage Users 버튼을 클릭할 시 User Manage Page 로 이동한다.
데이터 유형	텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동

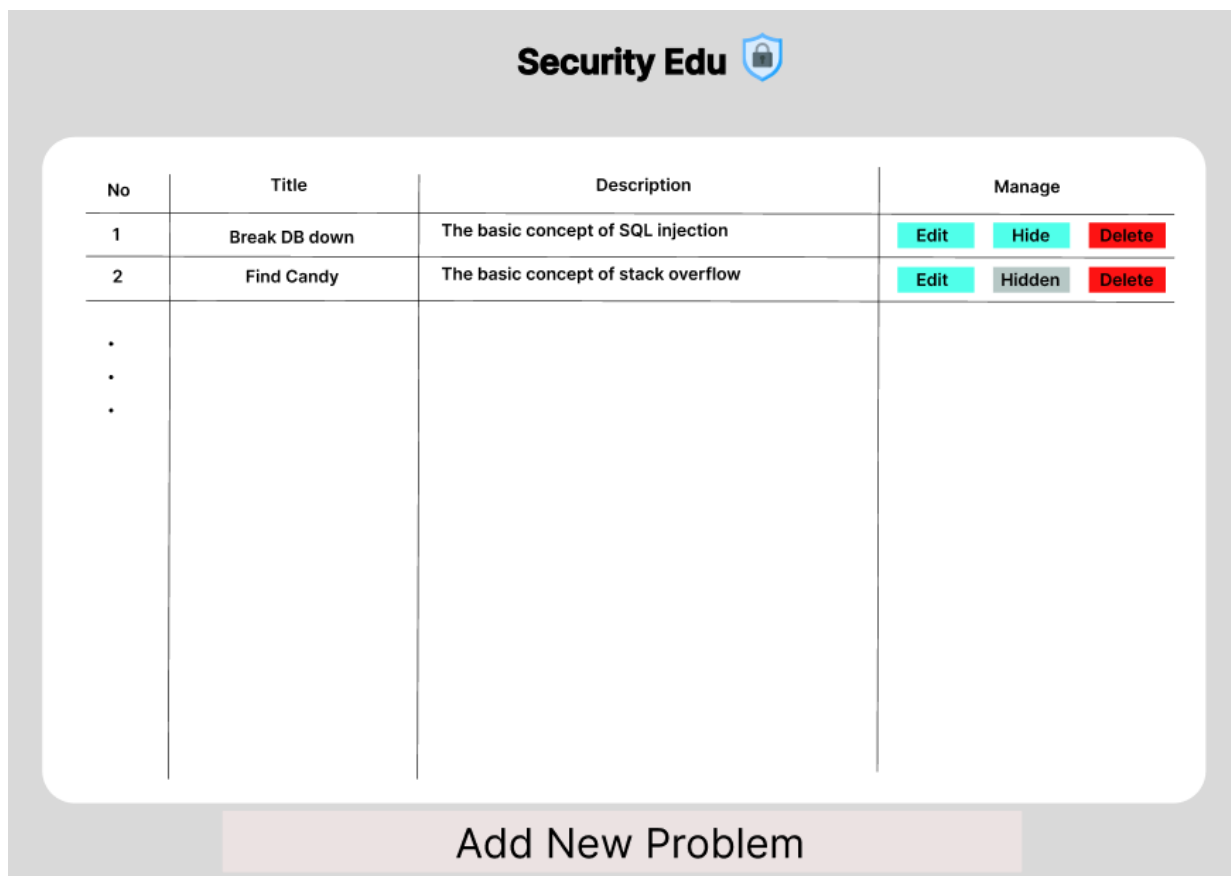


Figure 8. Design of Problem Manage Page

Table 10. User Interface of Problem Manage Page

Name of Page	Problem Manage Page
목적 및 설명	관리자가 새로운 문제를 추가하거나, 기존 문제를 변경, 숨김 처리 및 삭제할 수 있다.
입력 소스, 출력대상	클라이언트 / 호스트 서버 호스트 서버 / 클라이언트
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	페이지 로딩 처리: 네트워크 및 데이터베이스 서버 상태에 의존 페이지 렌더링 처리: 사용자 환경 성능에 의존 HTTP 요청 처리: 서버 상태에 의존
기타 입출력 관계	현재 저장되어 있는 문제 목록을 불러오기 위해 데이터베이스로부터 필요한 값들을 받는다.
화면 형식과 구성	현재 존재하는 모든 문제들의 리스트를 테이블 형식으로 보여준다. 각 문제는 Edit 버튼을 통해 수정할 수 있다. 각 문제는 Delete 버튼을 통해 삭제할 수 있다. 각 문제는 Hide/Hidden 토글 버튼으로 숨김 여부를 설정할 수 있다. Add New Problem 버튼을 클릭하여 새로운 문제를 추가할 수 있다. 문제 추가 및 수정 버튼 클릭 시 보여지는 별도의 페이지 디자인은 Figure 4. Design of Problem Page 와 동일하다. 해당 페이지에서 문제 설명, 추천 질문의 내용을 수정할 수 있다.
데이터 유형	텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동

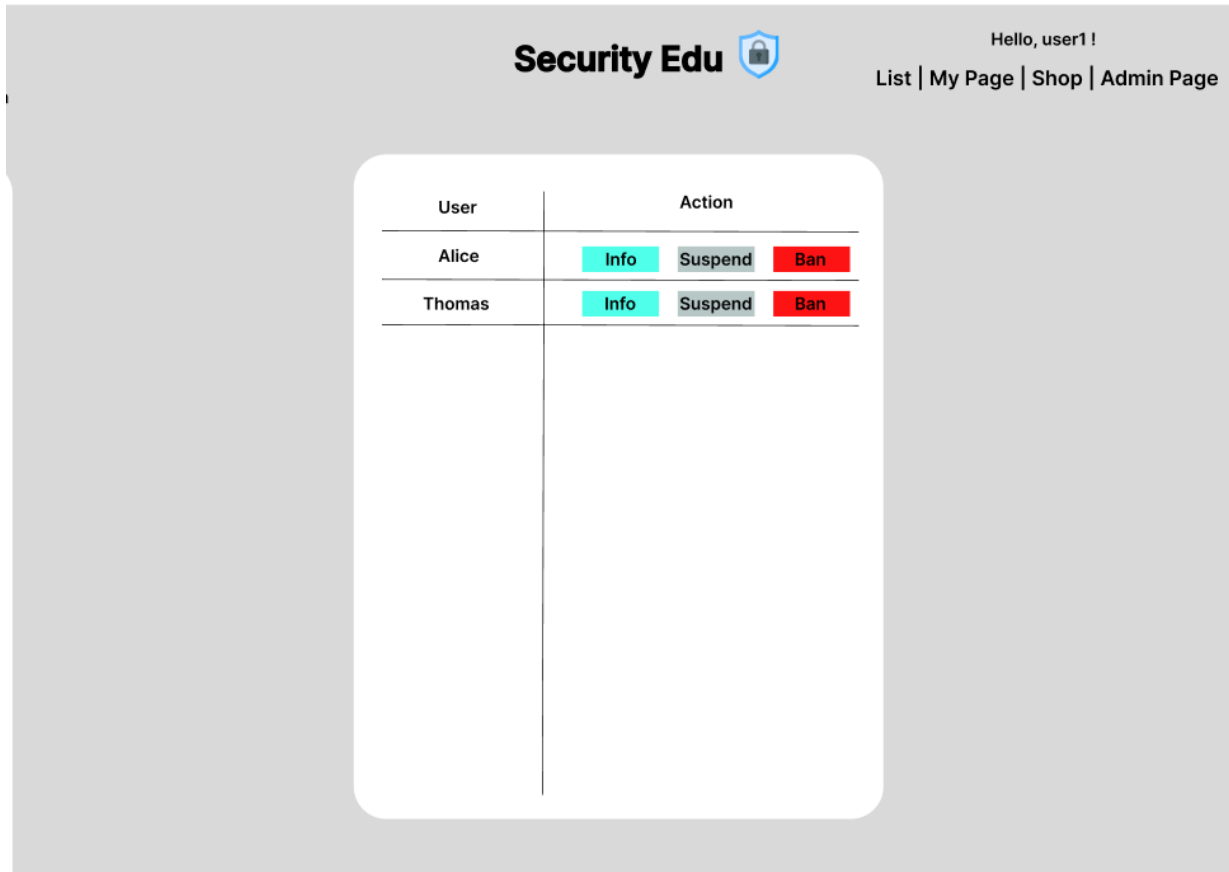


Figure 9. Design of User Manage Page

Table 11. User Interface of User Manage Page

Name of Page	User Manage Page
목적 및 설명	관리자는 Admin 권한 부여, 사용자 정보 확인, 사용자 정지로 구성된 사용자 관리를 할 수 있다.
입력 소스, 출력대상	클라이언트 / 호스트 서버 호스트 서버 / 클라이언트
범위, 정확도, 오차범위	표준에 의거한 HTML, CSS, JS 처리
구성 단위	화면(1920 * 1080)
시간 및 속도	페이지 로딩 처리: 네트워크 및 데이터베이스 서버 상태에 의존 페이지 렌더링 처리: 사용자 환경 성능에 의존

	HTTP 요청 처리: 서버 상태에 의존
기타 입출력 관계	현재 저장되어 있는 사용자의 목록을 데이터베이스로부터 불러온다.
화면 형식과 구성	<p>현재 존재하는 모든 사용자들의 리스트를 테이블 형식으로 보여준다.</p> <p>Info 버튼을 클릭하여 생성되는 팝업을 통해 각 사용자의 정보를 조회할 수 있다.</p> <p>팝업에 존재하는 별도의 버튼을 통해 일반 사용자에게 Admin 권한을 부여하여 관리자로 승격시킬 수 있다.</p> <p>Suspend 버튼을 클릭하고, 팝업창에서 기간을 설정하여 각 사용자를 서비스 이용 중지시킬 수 있다.</p> <p>Ban 버튼을 클릭하여 각 사용자를 영구적으로 서비스 이용 중지시킬 수 있다.</p>
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 동작	브라우저 종료 혹은 페이지 이동

3.1.2. Hardware Interfaces

본 서비스는 **Chrome** 웹 브라우저, **Windows** 운영 체제를 통해서 이용되는 것을 상정하여 개발된다.

본 문서에서 명시된 시스템은 사용자의 구동 환경이 다음과 같은 최소사항을 충족하였다고 가정한다.

1. 1GHz 이상 32비트(x86) 또는 64비트(x64) 프로세서 (Core I5 이상)
2. 1GB RAM(32비트) 또는 2GB RAM(64비트)
3. 16GB 사용 가능한 하드 디스크 공간(32비트) 또는 20GB(64비트)
4. WDDM 1.0 이상 드라이버와 DirectX 9 그래픽 디바이스

본 문서에서 서술한 구동 환경을 만족하지 못한 환경에서는 정상 동작을 보장할 수 없다.

또한, 본 시스템을 운영하는 서버는 다음과 같은 최소사항을 충족하였다고 가정한다.

1. Ubuntu 20.04 운영체제 및 NIC를 제공하는 메인보드
2. 4GB 이상의 RAM 환경

3. 25GB 이상의 하드 드라이브

3.1.3. Software Interfaces

본 서비스는 **Windows 7** 이상의 운영체제에서 최신 안정화 **Chrome** 웹 브라우저 버전을 통해 이용되는 것을 상정하여 개발된다. 보다 높은 안정성과 성능을 위해, **Windows 10** 이상 운영체제에서 **Chrome** 웹 브라우저 버전 **100** 이상을 사용하는 것이 권장된다.

3.1.4. Communication Interfaces

본 서비스의 웹 페이지는 **HTTP** 프로토콜을 기반으로 서버와의 통신을 수행한다. 사용자는 웹 페이지를 통해 본 서버와 통신하여 시스템이 제공하는 기능을 사용할 수 있다. **HTTP** 프로토콜이 전송되기 위해 안정된 **Internet** 환경에서 시스템을 사용하는 것이 권장된다.

3.1.5. Data Management Interfaces

서버의 메모리는 **4GB RAM**을 최소사양으로 한다. 보다 높은 안정성을 위해, **16GB** 이상의 메모리 공간이 권장된다.

사용자는 최소 **1GB RAM** 이상의 메모리를 사용한다. 보다 높은 안정성을 위해, **4GB** 이상의 메모리 공간이 권장된다.

3.2. Functional Requirements

3.2.1. Use Case

Table 12. List of Use Case

No.	Name of Function	Function Description
F1	Show Landing Page	사용자는 회원가입 및 로그인 인터페이스를 확인할 수 있다.
F2	Log In	사용자가 이미 등록된 계정을 통해 로그인을 할 수 있다.
F3	Sign In	사용자가 신규 계정을 생성할 수 있다.
F4	Log Out	사용자가 계정 이용 중, 세션을 끊고 나갈 수 있다.
F5	Show Main Page	사용자가 로그인 후, Main Page 에 접근 시에 인터페이스와 기능을 확인할 수 있다.

F6	Show Problem List	Main Page 내에서 현재의 Problem List를 조회할 수 있다.
F7	Show Top Solver List	Main Page 내에서 현재의 Top Solver List를 조회할 수 있다.
F8	Show Admin Page	관리자가 Admin page에 접근 시에 인터페이스와 기능을 확인할 수 있다.
F9	Show Add & Edit Page	Admin Page에서 Add or Edit Button을 눌렀을 때 Add & Edit Page에 접근하여 인터페이스와 기능을 확인할 수 있다.
F10	Add New Problem	관리자는 새로운 problem을 생성할 수 있다.
F11	Edit Existing Problem	관리자는 기존의 Problem을 수정할 수 있다.
F12	Show User Manage Page	Admin Page에서 Manage Users 버튼을 눌렀을 때 User Manage Page의 인터페이스와 기능을 확인할 수 있다.
F13	Give Admin Right	일반 사용자에게 Admin 권한을 부여할 수 있다.
F14	Show Problem Page	사용자가 문제를 풀기 위한 Problem Page의 인터페이스와 기능을 확인할 수 있다.
F15	Pre-Catch Recommend Question	사용자가 문제의 추천 질문에 대한 답변을 확인할 수 있다.
F16	Submit Answer	사용자가 답변을 제출하고 결과를 확인할 수 있다.
F17	Ask AI-Tutor	사용자가 AI-Tutor에게 질문할 수 있다.
F18	Show My Page	사용자 문제풀이 기록 정보를 확인 할 수 있다.
F19	Edit Personal Info	사용자가 개인 정보를 수정할 수 있다.

Table 13. (F1) Show Landing Page

F1	Show Landing Page
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	웹 페이지에 들어왔을 때 처음 보이는 페이지를 생성하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 웹 페이지의 제목, 목적 등이 간략히 서술되어 있다. 회원 가입, 로그인 버튼이 있어 각각을 누르게 되면 회원 가입 페이지,

	로그인 페이지로 이동한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 없음
사후 조건 (Post-condition)	<ul style="list-style-type: none"> 지정된 페이지로의 이동
가정 (Assumptions)	<ul style="list-style-type: none"> 서버 정상

Table 14. (F2) Log In

F2	Log In
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	아이디와 비밀번호를 받아 로그인을 하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 사용자로부터 아이디와 비밀번호를 입력받는다. 만약 아이디가 데이터베이스에 없다면 '없는 아이디입니다.'를 출력한다. 만약 아이디가 데이터베이스에 있지만, 비밀번호가 일치하지 않는다면 '비밀번호가 일치하지 않습니다.'를 출력한다. 아이디와 비밀번호의 조합이 데이터베이스에 있다면, 사용자의 정보 (문제풀이 정보, 질문 정보, 랭킹) 등이 적용된 Main Page로 이동한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 현재 로그인되지 않은 상태이다. 회원가입 정보가 데이터베이스에 저장되어 있다. 사용자의 정보가 데이터베이스에 저장되어 있다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> 사용자의 정보가 적용되고 로그인 된 상태의 Main Page로 이동한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 프로그램과 데이터베이스와의 정상 연결

	<ul style="list-style-type: none"> • 서버 정상
--	---

Table 15. (F3) Sign In

F3	Sign In
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	회원 가입을 위한 기본 정보를 사용자로부터 받아 데이터베이스에 저장하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> • 사용자의 username, 아이디, 비밀번호를 받는다. • 아이디가 중복되지 않는지 확인한다. 중복된다면 '중복된 아이디입니다', 그렇지 않다면 '사용 가능한 아이디입니다'를 출력한다. • 사용자가 제시한 비밀번호가 비밀번호 조건에 맞는지 확인한다. • 사용자가 모든 정보를 제공한 후 회원가입 버튼을 누르면 사용자의 회원 가입 정보(username, 아이디, 비밀번호)가 데이터베이스에 저장되고 log-in 페이지로 이동한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> • 현재 로그인되지 않은 상태이다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> • 사용자의 정보가 데이터베이스에 저장된다. • 로그인 페이지로 이동한다.
가정 (Assumptions)	<ul style="list-style-type: none"> • 프로그램과 데이터베이스와의 정상 연결 • 서버 정상

Table 16. (F4) Log Out

F4	Log Out
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	User가 자신의 계정을 이용하다 연결을 끊고 시스템에서 빠져나오는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 로그인 되어 있는 사용자가 Log-Out 버튼을 클릭한다. 현재의 Session Data를 삭제한다. 다시 로그인 페이지로의 Redirection을 수행한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> Log-in이 되어 있어 사용자의 Session정보가 유지되고 있는 상태여야 한다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> Session이 삭제된 상태여야 한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 서버 정상 Delete 명령이 수행 가능한 Session Database

Table 17. (F5) Show Main Page

F5	Show Main Page
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	본 프로젝트의 Main Page에 접속하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 로그인을 성공하면 Main Page의 IP 주소로 redirection 한다. 해당 IP 주소로 HTTP Protocol을 사용해 HTTP Request로 만들어 해당 IP 주소의 웹 서버로 전송한다. 전송된 Request Message는 웹 페이지 URL 로 변환되고, 해당 Data를 검색하여 다시 HTTP Protocol을 사용해 HTTP Response Message로, Web Page 로 변경하여 Web Browser에 의해 출력 된다.

전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 로그인을 성공적으로 수행한 사용자 이어야 한다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> 접속 로그가 데이터베이스에 저장된다. Main Page로 이동한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 원활한 인터넷 환경 서버 정상

Table 18. (F6) Show Problem List

F6	Show Problem List
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	Pre-defined Problem과 그 solved status의 List를 보여주는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 사용자가 로그인 후 Main Page에 접속한다. REST API를 사용하여 DB에 저장되어 있는 Problem Data, Problem Status에 대해 Query를 수행한다. 가져온 Data로 HTML Rendering을 수행한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 문제들은 미리 만들어져 있는 상태여야 한다. 문제가 존재하지 않는 경우 빈 목록을 출력해야 한다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> 현재 사용자의 각 문제에 대한 해결 여부가 문제와 함께 표시되어야 한다. Problem 클릭 시 Problem Page로 Redirect 가능한 상태여야 한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 서버 정상 Select 명령이 수행 가능한 Problem Database

Table 19. (F7) Show Top Solver List

F7	Show Top Solver List
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	Daily, Entire로 정렬된 Top Solver들의 List를 각각 보여주는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> • 사용자가 Main Page에 접속한다. • REST API를 사용하여 DB에 저장되어 있는 사용자 Data에서 각 기준에 따라 Order by 된 Data Query를 수행한다. • 가져온 Data로 HTML Rendering을 수행한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> • 사용자와 푼 문제의 개수가 DB에 함께 저장되어 있는 상태여야 함. • 사용자가 없는 경우 빈 List를 출력한다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> • 정렬된 Top Solver의 List가 정확하게 보여야 한다.
가정 (Assumptions)	<ul style="list-style-type: none"> • 서버 정상 • Select 명령이 수행 가능한 User Database

Table 20. (F8) Show Admin Page

F8	Show Admin Page
행위자 (Actor)	관리자
설명 (Description)	Admin Page를 생성하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> • 로그인을 성공한 뒤, Main Page Access 함수를 거쳐 Main Page로 접근한다. • 관리자 권한이 있을 경우, Main page에 생성된 Admin page 버튼을 눌렀을 때 Admin page를 생성한다.

전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 로그인을 성공적으로 수행한 사용자의이어야 한다. 로그인을 성공한 사용자는 Admin 권한을 갖고 있어야 한다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> Admin page의 정상 요소들이 정확하게 보여야 한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 원활한 인터넷 환경 서버 정상

Table 21. (F9) Show Add & Edit Page

F9	Show Add & Edit Page
행위자 (Actor)	관리자
설명 (Description)	admin page에서 add or edit button을 눌렀을 때 add & edit page을 생성하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 로그인을 성공한 뒤, Main Page Access 함수를 거쳐 Main Page로 접근한다. 사용자가 관리자 권한이 있을 경우, Main page에 생성 된 admin page 버튼을 눌렀을 때 admin page를 생성한다. 생성된 admin page에서 add or edit button을 누른 후 add & edit page 생성한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 로그인을 성공적으로 수행한 사용자의이어야 한다. 로그인을 성공한 사용자는 admin 권한을 갖고 있어야 한다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> add & edit page의 정상 요소들이 정확하게 보여야 한다. add button을 눌렀을 시에 빈 입력창이 나와야하며, edit button을 눌렀을 시에는 문제의 정보가 각각 입력창에 들어가 있어야 한다.

가정 (Assumptions)	<ul style="list-style-type: none"> ● 원활한 인터넷 환경 ● 서버 정상
------------------	---

Table 22. (F10) Add New Problem

F10	Add New Problem
행위자 (Actor)	관리자
설명 (Description)	새로운 problem을 입력하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> ● 로그인을 성공한 뒤, Main Page Access 함수를 거쳐 Main Page로 접근한다. ● 관리자 권한이 있을 경우, Main page에 생성 된 admin page 버튼을 누른 뒤 show admin page 함수 실행 및 admin page로 이동한다. ● admin page에서 add버튼을 눌러 show add & edit page 함수를 이용하여 add & edit page로 이동한다.. ● add & edit page의 입력 공간에 problem의 제목, 내용, 추가 질문을 입력 후 submit 버튼을 눌렀을 때, 서버의 DB에 problem저장 및 problem list에 표기한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> ● 로그인을 성공적으로 수행한 사용자여야 한다. ● 로그인을 성공한 사용자는 admin 권한을 갖고 있어야 한다. ● 추가하는 추천 질문 각각은 AI-tutor에게 질문을 하였을 때 검열 되지 않는 질문이어야 한다.

사후 조건 (Post-condition)	<ul style="list-style-type: none"> DB에 입력한 Problem의 정보(제목, 내용, 추천 질문)가 저장되어야 한다. 입력한 Problem이 문제 풀이 화면에 표시되며, 표시된 문제를 통해 문제를 풀 수 있어야 한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 원활한 인터넷 환경 서버 정상

Table 23. (F11) Edit Existing Problem

F11	Edit Existing Problem
행위자 (Actor)	관리자
설명 (Description)	기존의 problem 을 수정하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 로그인을 성공한 뒤, Main Page Access 함수를 거쳐 Main Page로 접근한다. 관리자 권한이 있을 경우, Main page에 생성 된 admin page 버튼을 누른 뒤 show admin page 함수 실행 및 admin page로 이동한다. Admin page에서 Add버튼을 눌러 Show add & Edit page 함수를 이용하여 Add & Edit page로 이동한다.. Add & Edit page의 입력 공간에 Problem의 제목, 내용, 추가 질문의 수정사항을 입력 후 Submit 버튼을 눌렀을 때, 서버의 DB에 Problem 저장 및 Problem List에 정상 표기한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 로그인을 성공적으로 수행한 사용자의이어야 한다. 로그인을 성공한 사용자는 admin 권한을 갖고 있어야 한다. 추가하는 추천 질문 각각은 AI-tutor에게 질문을 하였을 때 검열 되지 않는 질문이어야 한다.

사후 조건 (Post-condition)	<ul style="list-style-type: none"> DB에 입력한 기존 Problem의 수정된 정보(제목, 내용, 추천 질문)가 저장되어야 한다. 입력한 Problem이 문제 풀이 화면에 표시되며, 표시된 문제를 통해 문제를 풀 수 있어야 한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 원활한 인터넷 환경 서버 정상

Table 24. (F12) Show User Manage Page

F12	Show User Manage Page
행위자 (Actor)	관리자
설명 (Description)	User manage page를 생성하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 로그인을 성공한 뒤, Main Page Access 함수를 거쳐 Main Page로 접근한다. 관리자 권한이 있을 경우, Main page에 생성된 Admin Page 버튼을 눌렀을 때 Admin Page를 생성한다. 생성된 Admin Page에서 Manage User 버튼을 눌렀을 때 User Manage Page를 생성한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 로그인을 성공적으로 수행한 사용자이어야 한다. 로그인을 성공한 사용자는 Admin 권한을 갖고 있어야 한다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> User Manage Page의 구성 요소들이 정확하게 보여야 한다.

가정 (Assumptions)	<ul style="list-style-type: none"> ● 원활한 인터넷 환경 ● 서버 정상
------------------	---

Table 25. (F13) Give Admin Right

F13	Give Admin Right
행위자 (Actor)	관리자
설명 (Description)	사용자에게 admin 권한을 부여하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> ● 로그인을 성공한 뒤, Main Page Access 함수를 거쳐 Main Page로 접근한다. ● 관리자 권한이 있을 경우, Main page에 생성된 Admin Page 버튼을 누른 뒤 Show Admin Page 함수 실행 및 Admin Page로 이동한다. ● Admin Page에서 manage users 버튼을 눌러 Show Manage Users page 함수를 이용하여 Manage User Page로 이동한다. ● Manage User Page의 목록에서 사용자를 선택하여 권한을 부여한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> ● 로그인을 성공적으로 수행한 사용자이어야 한다 ● 로그인을 성공한 사용자는 Admin 권한을 갖고 있어야 한다. ● 권한을 부여받는 사용자는 Admin 권한을 갖고 있지 않아야 한다.

사후 조건 (Post-condition)	<ul style="list-style-type: none"> 권한을 받은 사용자는 Admin 권한을 갖고 있어야 한다.
가정 (Assumptions)	<ul style="list-style-type: none"> 원활한 인터넷 환경 서버 정상

Table 26. (F14) Show Problem Page

F14	Show Problem Page
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	선택된 Problem Page가 보여질 때 필요한 화면 구성을 보여주는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 사용자가 선택한 문제 설명과 추천 질문이 보여진다. 추천질문을 OpenAI API를 통해 미리 답변을 받아둔다. 사용자가 로그인 후 지금까지 이 문제를 풀면서 AI-Tutor와 대화한 내역이 보여진다. 문제에 대한 해결 여부를 표시해 사용자가 시도해서 틀렸던 문제인지, 해결한 문제인지, 시도하지 않았던 문제를 구분할 수 있다. 상단에 Main Page로 돌아갈 수 있는 버튼이 존재하여, 클릭 시 Show Main Page와 같은 기능을 한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 사용자가 로그인이 되어있는 상태

	<ul style="list-style-type: none"> ● 사용자의 해당 문제의 최종 정답여부, 로그인 후 이 순간까지 AI-Tutor에게 한 질문과 그 답변기록이 데이터베이스에 저장되어 있다. ● 해당 문제에 대한 추천질문들이 데이터베이스에 저장되어 있다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> ● 화면에 선택한 문제의 질문과 추천질문이 보인다. ● 이전대화 내역이 AI-Tutor 화면에 보여지고 있다.
가정 (Assumptions)	<ul style="list-style-type: none"> ● 프로그램과 데이터베이스와의 정상 연결 ● 서버 정상

Table 27. (F15) Pre-Catch Recommended Question

F15	Pre-Catch Recommended Question
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	화면에 보여지는 문제에 관련된 추천 질문들 중 하나를 선택해 그에 대한 AI-Tutor의 답변을 보여주는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> ● 사용자가 추천 질문들 중 질문하고 싶은 문항을 클릭한다. ● 클릭한 질문이 AI-Tutor화면에 보여진다. ● 미리 전달받은 답변이 대화창에 보여진다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> ● 사용자가 로그인이 되어있는 상태이다. ● 해당 문제에 대한 추천질문들이 데이터베이스에 저장되어 있다. ● OpenAI를 통해 질문에 대한 답변을 받을 수 있다. ● 이전에 대화한 내역이 AI-Tutor 화면에 보여지고 있다. ● 현 Problem Page가 로드되었을 때 미리 OpenAI를 통해 질문에 대한 답변을 받은 상태이다.

사후 조건 (Post-condition)	<ul style="list-style-type: none"> 질문과 그 답변이 데이터베이스에 저장된다. 이전에 대화한 내역 아래에 새로 한 질문과 답변이 표시된다.
가정 (Assumptions)	<ul style="list-style-type: none"> 프로그램과 데이터베이스와의 정상 연결 서버 정상 OpenAI API가 정상적으로 동작하고 있다

Table 28. (F16) Submit Answer

F16	Submit Answer
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	입력한 답변을 제출하고, 답변 결과를 확인하는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> 사용자가 답변란에 답을 입력하고, Submit 버튼을 클릭하면서 답변을 제출한다. 문제 ID와 입력된 답변이 전송되어, 현 문제의 정답과 답변이 동일한지 확인하고 그 결과를 반환한다. 정답여부에 따라 문제 해결 여부가 출력된다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> 사용자가 로그인되어있는 상태 데이터베이스에 문제와 그에 상응하는 정답이 저장되어 있어야 한다. 사용자의 해당 문제 해결 여부가 데이터베이스에 저장되어 있다.

사후 조건 (Post-condition)	<ul style="list-style-type: none"> ● 사용자의 해당문제의 정답여부가 데이터베이스에 저장된다. ● 정답여부에 맞게 화면의 최종 정답여부가 출력된다.
가정 (Assumptions)	<ul style="list-style-type: none"> ● 프로그램과 데이터베이스와의 정상 연결 ● 서버 정상

Table 29. (F17) Ask to AI-Tutor

F17	Ask to AI-Tutor
행위자 (Actor)	사용자
설명 (Description)	AI-Tutor에게 사용자가 직접 질문하고 그 질문에 대한 답을 보여주는 함수
정상 흐름 (Action)	<ul style="list-style-type: none"> ● 사용자가 AI-Tutor 인터페이스에 질문을 입력해 전송 버튼을 클릭한다. ● 질문내용이 서버를 통해 전달되고, 그 질문에 대한 답변을 수신한다. ● 전달받은 답변이 인터페이스에 출력된다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> ● 사용자가 로그인되어있는 상태이다. ● OpenAI를 통해 질문에 대한 답변을 받을 수 있다. ● 이전 대화내역이 AI-Tutor 화면에 보여지고 있다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> ● 질문과 그 답변이 데이터베이스에 저장된다. ● 기존 대화내역에 이어서 새로운 대화내역이 출력된다.
가정 (Assumptions)	<ul style="list-style-type: none"> ● 프로그램과 데이터베이스와의 정상 연결

	<ul style="list-style-type: none"> • 서버 정상 • OpenAI가 정상적으로 동작하고 있다.
--	---

Table 30. (F18) Show My Page

F18	Show My Page
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	본 시스템에서 사용자가 풀었던 문제에 대한 기록을 볼 수 있는 함수이다. 해결한 문제, 시도했지만 해결하지 못한 문제, 자신의 랭킹을 보여준다.
정상 흐름 (Action)	<ul style="list-style-type: none"> • 사용자가 마이페이지 - 문제풀이 기록에 접근한다. • 사용자의 문제별 정답여부를 데이터베이스에 쿼리한다. • 가져온 데이터를 렌더링한다.
전제 조건 (Pre-condition)	<ul style="list-style-type: none"> • 사용자가 정상적으로 로그인되어 있다. • 사용자-문제별 정답여부 정보가 데이터베이스에 저장되어 있다.
사후 조건 (Post-condition)	<ul style="list-style-type: none"> • 화면에 사용자의 문제풀이 기록 정보가 보인다.

Table 31. (F19) Edit Personal Info

F19	Edit Personal Info
행위자 (Actor)	일반 사용자, 관리자
설명 (Description)	본 시스템에 등록할 때 남긴 사용자 정보(username, 비밀번호)를 수정 하는 함수이다.
정상 흐름 (Action)	<ul style="list-style-type: none"> • 본 시스템에 등록할 때 남긴 사용자 정보를 비밀번호를 제외하고 보여준다. • 변경을 희망하는 정보를 클릭 시 수정이 가능하게 한다.

	<ul style="list-style-type: none"> ● 비밀번호 변경을 희망할 시 기존 비밀번호, 새로운 비밀번호, 비밀번호 확인을 입력 받아 수정하게 한다. ● 수정 완료 버튼을 누르면 정보가 갱신된다.
<p>전제 조건 (Pre-condition)</p>	<ul style="list-style-type: none"> ● 사용자가 정상적으로 로그인되어 있다. ● 사용자의 회원정보가 데이터베이스에 저장되어 있다.
<p>사후 조건 (Post-condition)</p>	<ul style="list-style-type: none"> ● 수정한 정보로 회원정보 데이터베이스를 갱신한다.
<p>가정 (Assumptions)</p>	<ul style="list-style-type: none"> ● 프로그램과 데이터베이스와의 정상 연결 ● 서버 정상

3.2.2. Use Case Diagram

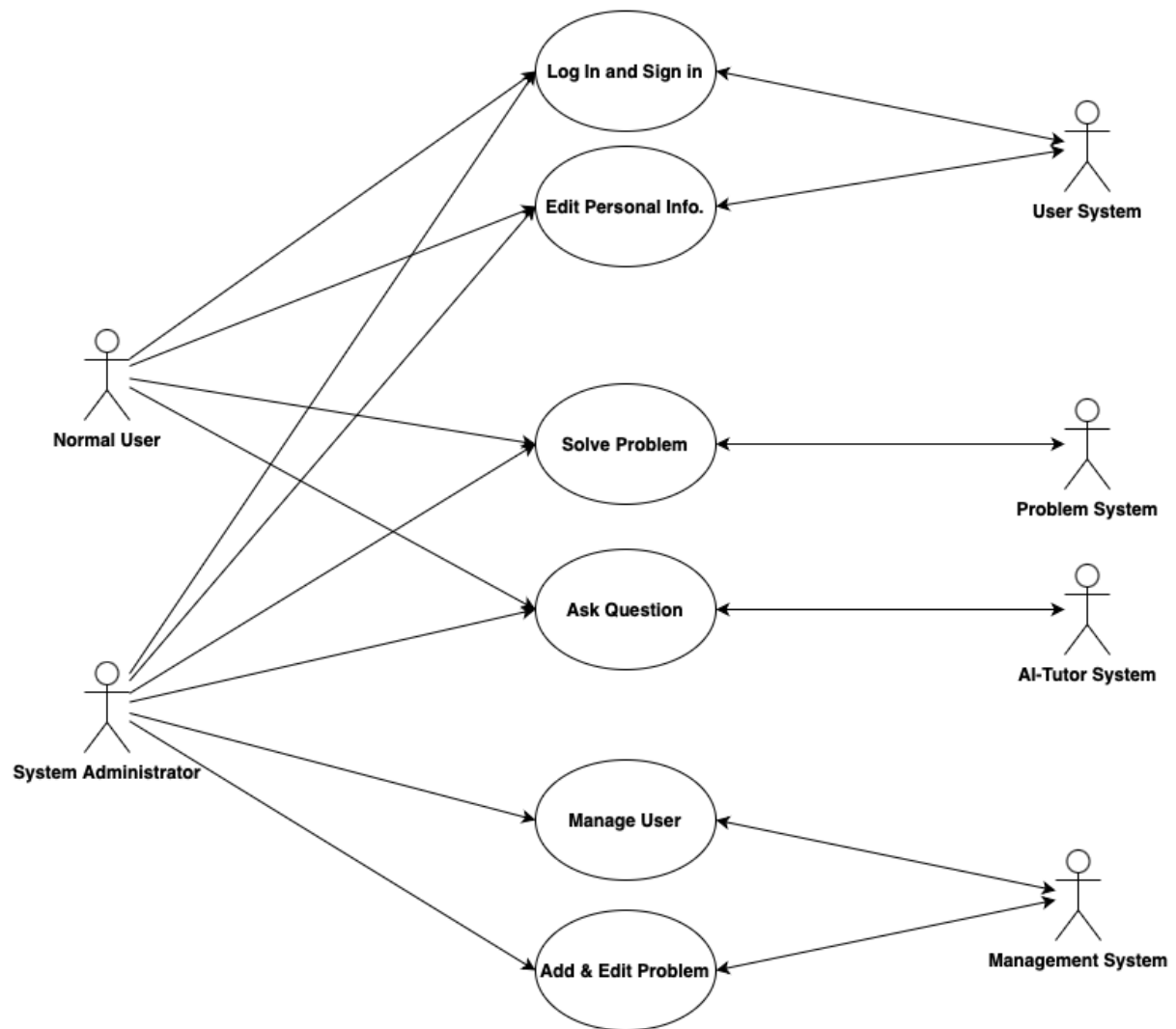


Figure 10. Use Case Diagram

3.2.3. Data Flow Diagram

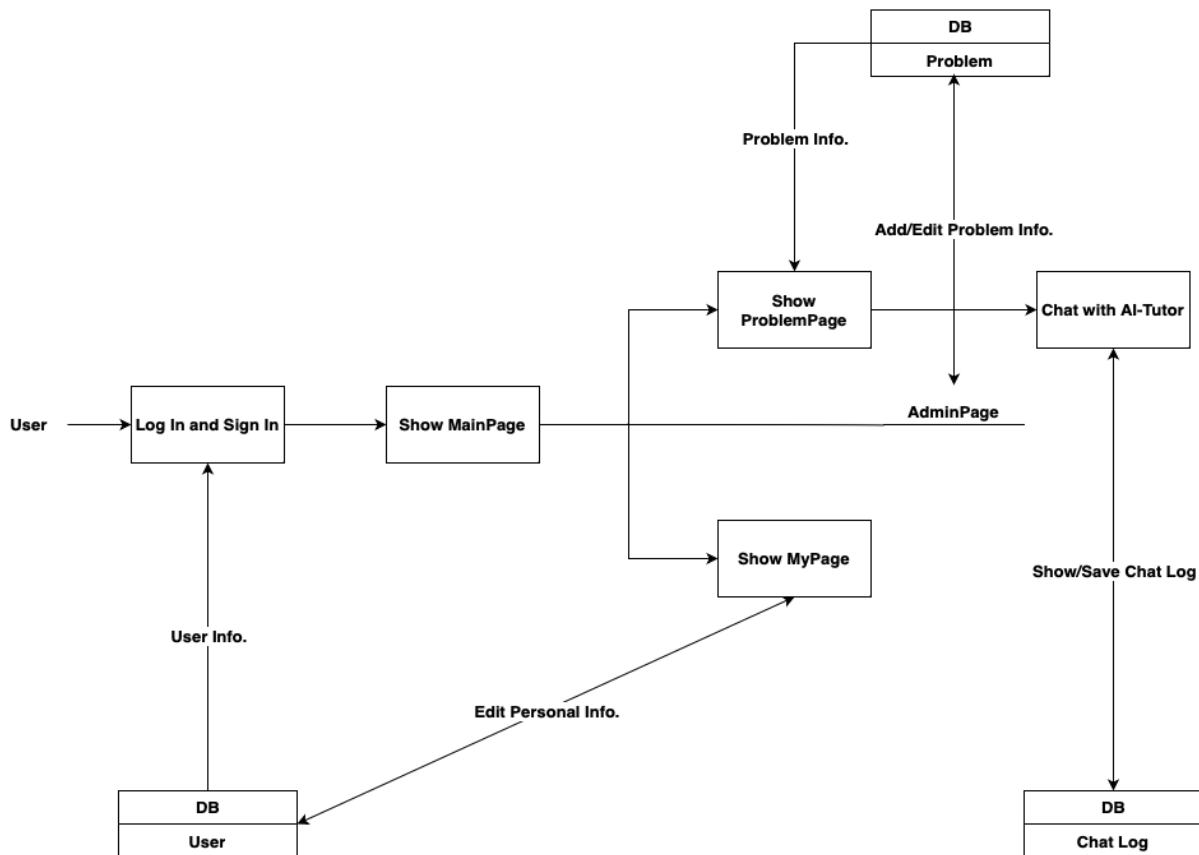


Figure 11. Data Flow Diagram

3.3. Product Requirements

3.3.1. Usability Requirements

본 서비스는 사용자가 명시된 권장 사양보다 높은 수준의 환경에서 접속하였을 때 올바른 동작을 보장한다.

본 서비스에서 제공하는 보안 문제 풀이 환경은 사용자가 별도의 시스템이나 소프트웨어를 활용하지 않고도 보안에 관련된 지식을 제공하는 것을 목적으로 한다. 따라서 본 시스템의 문제는 명확한 해결법을 수반해야 하며, 사용자가 본 서비스에서 제공하는 기능만으로 해결할 수 있도록 난이도가 책정되어야 한다.

3.3.2. Performance Requirements

본 시스템은 서버와 네트워크에 이상이 없다는 가정 하에, 페이지 로딩 시 5초 이상의 지연이 발생하지 않아야 한다. 이를 위해 메모리나 데이터베이스 상의 사소한

공간적인 낭비를 허용한다.

본 시스템은 최대 **100명**까지의 동시 접속을 상정하여 개발된다. 상정된 접속 규모를 넘어서는 트래픽이 발생하였을 경우, 관리자 또는 설계자는 서비스 이용에 장애가 발생하지 않도록 서버 리소스를 재설계해야 한다.

3.3.3. Security Requirements

본 서비스는 **Back-end** 서버에 가해질 수 있는 잠재적인 **Denial-of-Service** 공격을 방지하기 위해 **Whitelist**를 통해 **Back-end** 서버에 접근 가능한 **IP** 주소를 제한한다.

3.4. Organizational Requirements

3.4.1. Environmental Requirements

본 시스템은 웹 기반 애플리케이션이다. 사용자는 **PC**를 통해 시스템에 접속하는 것을 상정하며, **PC** 환경에서의 올바른 작동을 보장한다. 시스템은 **Windows, MacOS, Linux**와 같은 일반적인 운영 체제와 호환됨을 보장한다. 또한 **Chrome, Firefox, Safari**와 같은 널리 사용되는 웹 브라우저와 호환됨을 보장한다.

본 시스템은 **React**와 **Nest.js**를 통해 개발된다. 시스템은 **HTML, CSS, Javascript**와 같은 웹 표준을 준수하여 개발된다. 개발 팀은 필요한 경우 라이선스를 준수하여 오픈소스 소프트웨어를 시스템에 포함시킬 수 있다.

본 시스템을 개발하는 과정에 있어, 개발 팀은 버전 관리와 협업을 위해 **Git**과 같은 **VCS**를 사용한다. 또한 개발 팀은 유지 보수성과 확장성을 보장하기 위해 주석과 리팩토링과 같은 모범적인 방법론을 적극적으로 사용한다.

3.4.2. Operational Requirements

해당 시스템은 가용성을 보장하기 위해 정기적으로 모니터링되어야 한다. 시스템의 가동 중 문제가 발생하였을 경우 신속하게 처리되어야 한다.

해당 시스템은 데이터를 정기적으로 백업하여 안전한 여분의 스토리지에 저장해야 한다. 문제가 발생하였을 경우 데이터의 복원이 이루어질 수 있는지 사전에 주기적으로 검사되어야 한다.

해당 시스템은 일반 사용자와 관리자로 구분되는 접근 제어 방식을 채택하고

있으며, 일반 사용자가 상위 권한을 가질 수 없도록 인증 방식이 작동되어야 한다.

3.5. External Requirements

3.5.1. Regulatory Requirements

본 시스템에서 사용되는 외부 API에는 openAI API가 있다. 외부 API를 사용할 때에는 그 license를 충분히 검토하여야 하며, license 간의 충돌은 허용되지 않는다.

사용자에게 제공되는 보안 문제들은 오픈소스로 공개된 예제 코드들을 기반으로 제작되었다. 본 시스템에서는 문제 자료를 사용할 때 대한민국 법령이 공표한 저작권법을 따라야 하며, 이를 위배해서는 안 된다.

3.5.2. Safety/Security Requirements

본 시스템은 대한민국 법령이 공표한 개인정보보호법에 의거하여 사용자의 개인정보를 처리하고 보호한다. 각 사용자의 비밀번호는 통신 과정에서 암호화되어, 중간자 공격 등 잠재적인 보안상 취약점을 효과적으로 방지할 수 있다.

3.5.3. Ethical Requirements

본 서비스의 모든 콘텐츠는 오직 교육 목적으로만 제공되며, 실제 시스템에 대한 공격을 비롯한 모든 불법 행위를 장려하지 않는다. 서비스에서 제공하는 모든 문제는 가상의 상황을 상정하며, 실제 시스템을 대상으로 한 불법 행위로 이어질 수 있는 구체적인 맥락을 제공하지 않는다.

3.6. Organizing System Flow

본 서비스의 시스템 모델을 명확하게 나타내기 위해, UML에 기반한 시스템 흐름도를 작성하였다.

3.6.1. Context Model

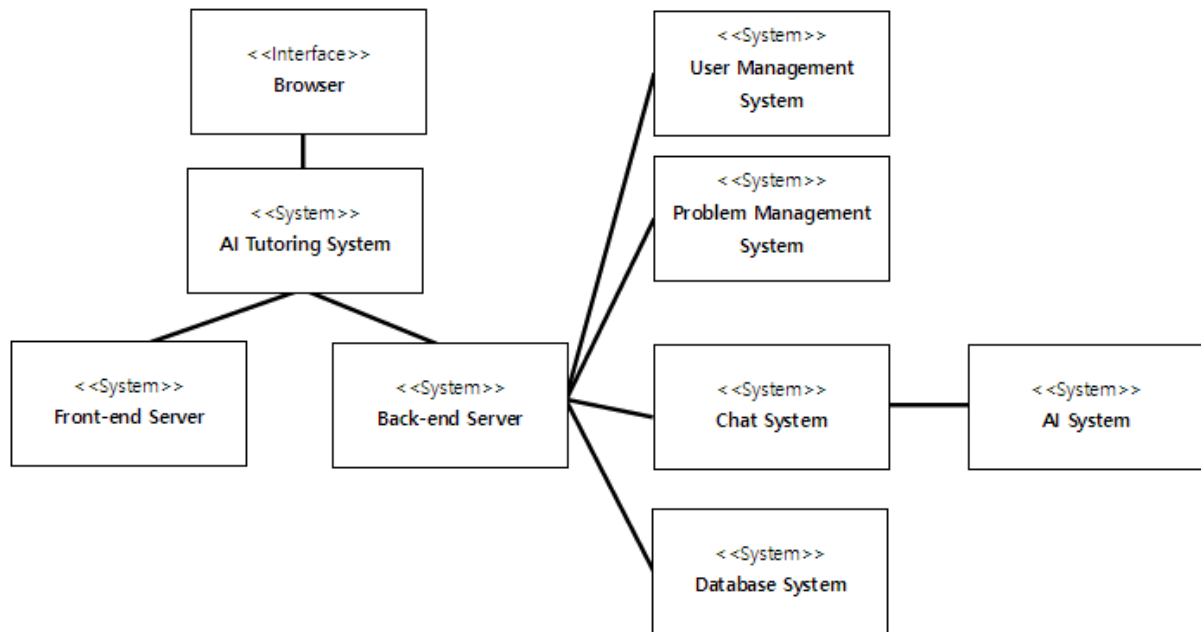


Figure 12. Context Model

3.6.2. Process Model

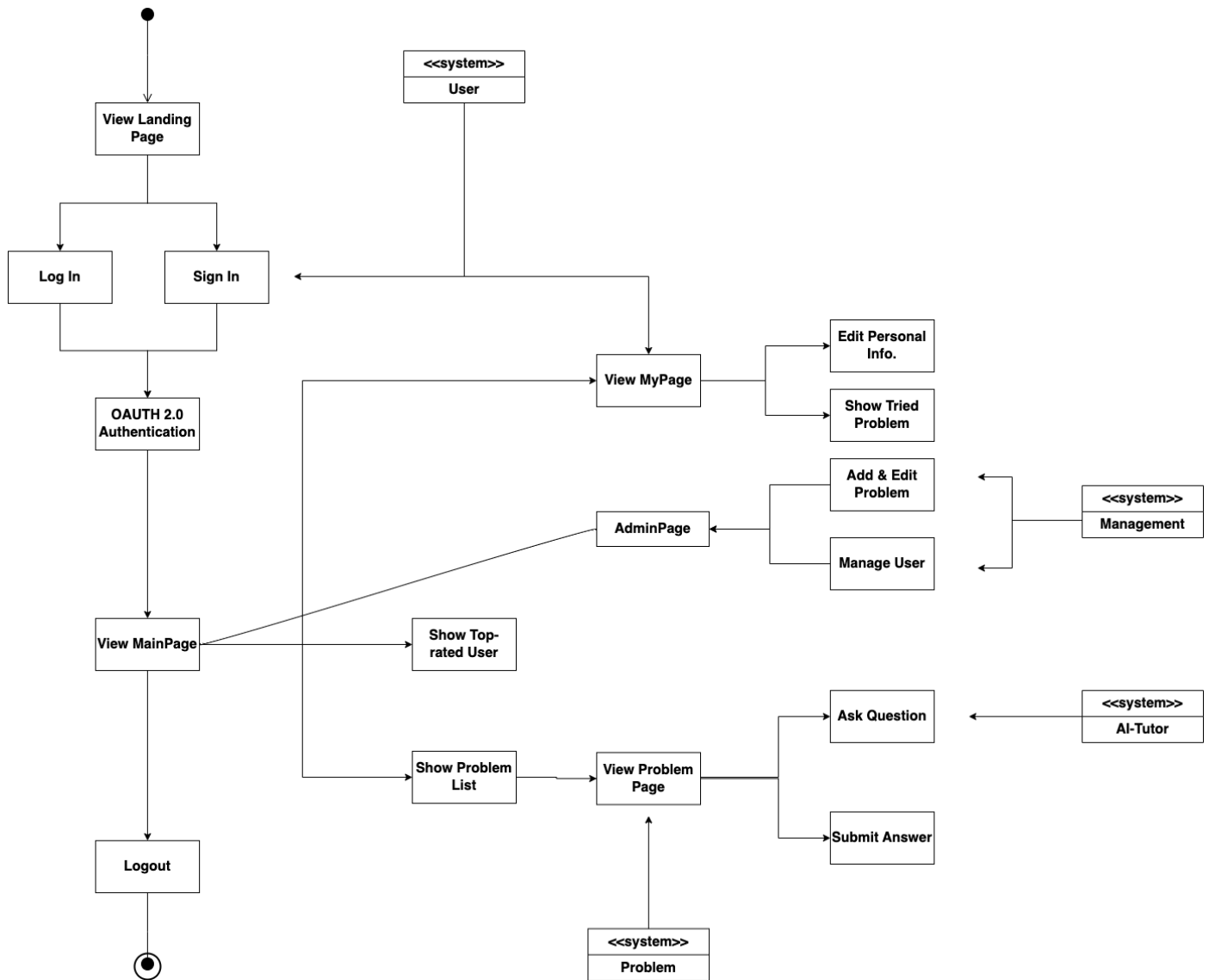


Figure 13. Process Model

3.6.3. Interaction Model

3.2.2 항목의 Use Case Diagram 형태로 표현되었다.

3.6.4. Behavior Model

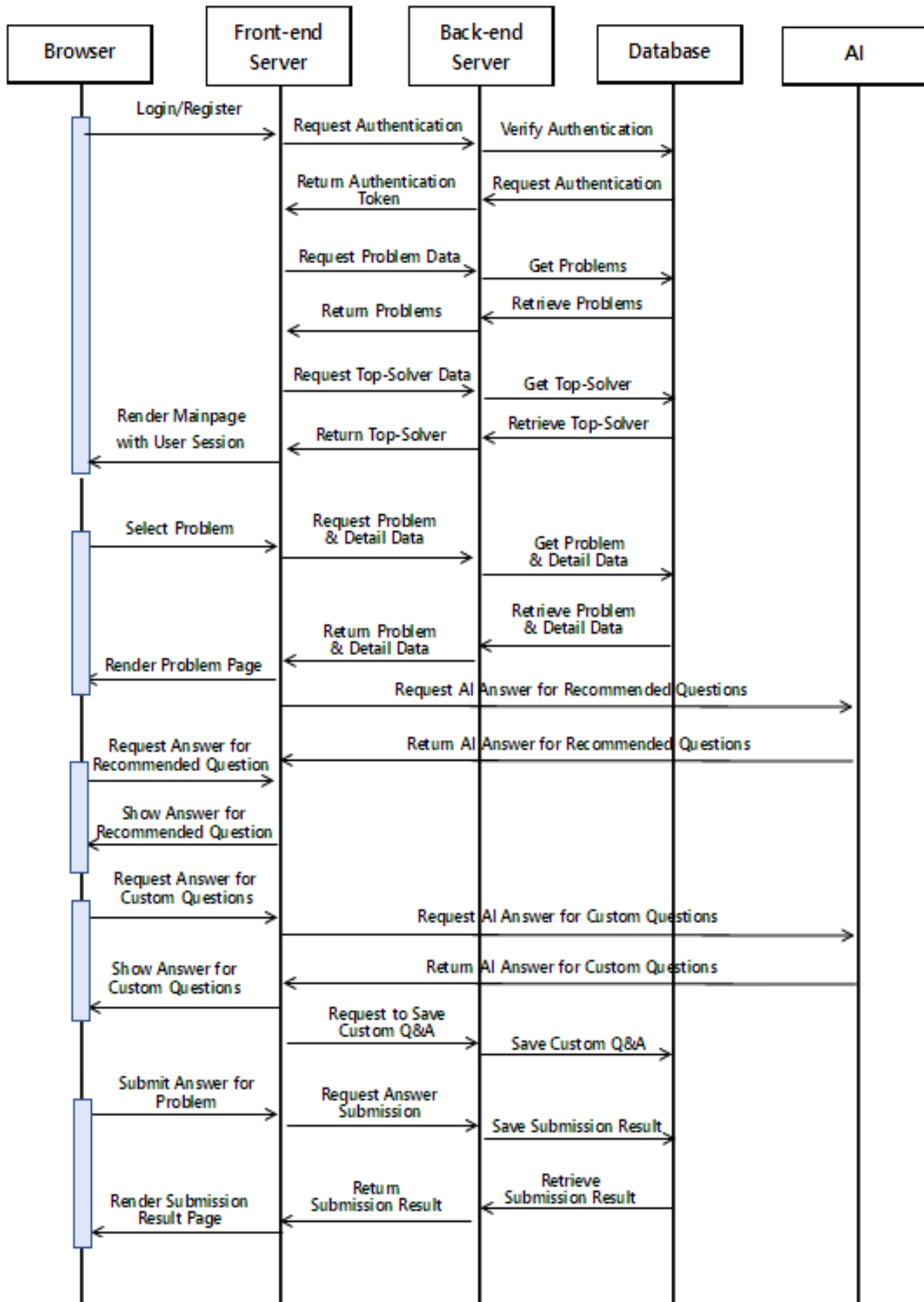


Figure 14. Sequence Diagram

3.7. System Architecture

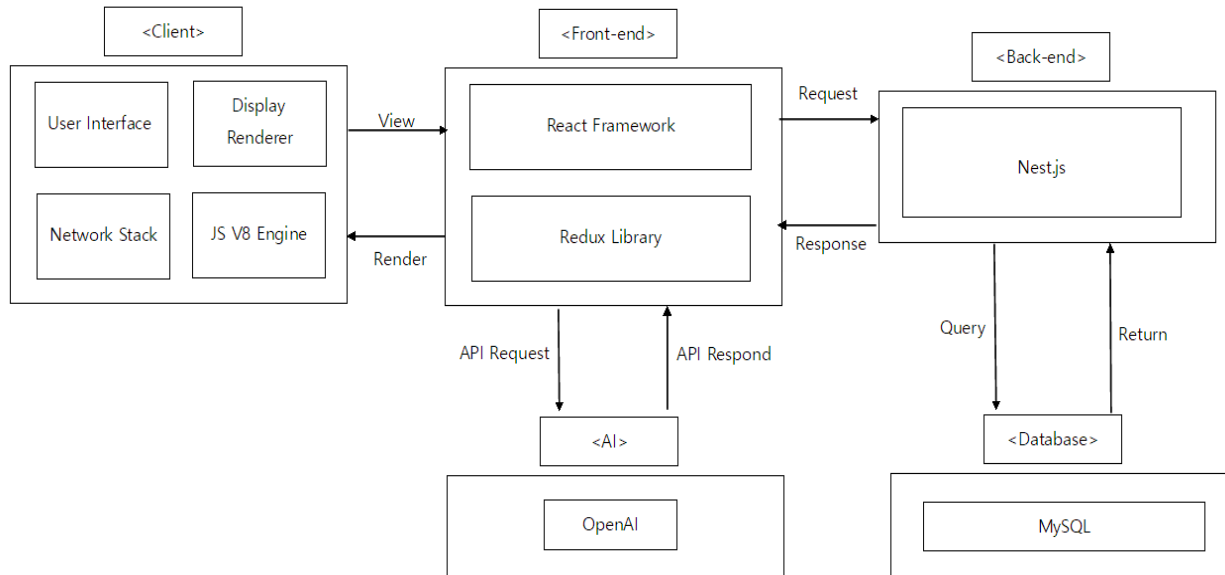


Figure 15. System Architecture

3.8. System Evolution

3.8.1. Assumption and Limitation

본 서비스는 웹 브라우저 기반으로 사용자가 사이버 보안 분야를 학습할 수 있는 환경을 제공한다. 본 서비스를 출시하는 시점에, 사용자가 문제에 답을 입력하는 방식은 객관식과 단답형으로 제한된다. 또한 질문 추천 시스템은 수동으로 관리 되어야 한다.

그러나 추후 추천 질문 생성 및 관리를 자동화하는 등의 다양한 기능이 추가될 수 있다.

3.8.2. Evolution of Hardware and Software

첫 출시 시점에서 시스템은 **AWS EC2** 클라우드 서버에서 호스팅된다. 추후 사용자 트래픽 및 데이터 양이 증가함에 따라 더 높은 수준의 리소스를 가진 서버로 전환될 가능성이 있다.

본 시스템은 **React**와 **Nest.js**를 사용하여 개발되었다. 해당 소프트웨어의 지속적인 버전 관리가 필요하다.

본 서비스는 추천 질문 시스템을 수동으로 관리한다. 향후 **OpenAI API**의 기능이

더 발전된다면 해당 시스템에 대한 완전한 자동화를 구축할 가능성이 있다.

3.8.3. Diverse Platforms

본 서비스의 초기 플랫폼은 PC 환경의 웹 브라우저 환경이다. 추후 서비스의 수요 증가가 증가함에 따라 모바일 애플리케이션이나 독립된 프로세스로 작동하는 데스크탑 애플리케이션으로 확장이 발생할 수 있다.

4. Supporting Information

4.1. Software Requirement Specification

본 소프트웨어 요구 명세서는 IEEE 권장사항에 맞추어 작성되었다 (IEEE Recommended Practice for Software Requirements Specifications, IEEE-Std-830). 본 서비스의 요구 조건을 명확히 할 수 있도록 본래의 양식에서 일부 추가되거나 제외된 바 있다.

4.2. Document History

Table 32. Document History

Date	Version	Description	Writer
2023.04.24	0.0.1	초본 작성	홍형근 외 5인
2023.04.29	1.0.0	최종본 작성	홍형근 외 5인