



Online Coding Class about Web Crawler

2022.04.09

소프트웨어공학개론 41

TEAM 2 (CrawlLearn)

Team Leader	이민영
Team Member	김진환
Team Member	배지현
Team Member	이정우
Team Member	정미서

CONTENTS

1. INTRODUCTION	6
1.1. Purpose	6
1.2. Scope.....	6
1.3. Definition, Acronyms, and Abbreviation	6
1.4. References.....	7
1.5. Overview	7
2. OVERALL DESCRIPTION	8
2.1. Product Perspective.....	8
2.2. Product Functions.....	8
2.2.1. Landing Page.....	9
2.2.2. Login/Start	9
2.2.3. Landing Page for User	9
2.2.4. Lecture	10
2.2.5. Coding Practice	10
2.2.6. My page	10
2.2.7. Q&A	10
2.2.8. Asking question	11
2.2.9. Q&A detail	11
2.3. User Classes and Characteristics	11
2.3.1. System Administrator	11
2.3.2. User.....	11
2.3.3. Lecturer.....	11
2.4. Operating Environment	12
2.5. Design and Implementation Constraints	12
2.6. User Documentation.....	12
2.7. Assumptions and Dependencies	12
3. EXTERNAL INTERFACE REQUIREMENTS	13
3.1. User interface	13
3.2. Hardware Interfaces	25
3.3. Software Interfaces	25
3.4. Communication interfaces	27

4. SYSTEM REQUIREMENTS	28
4.1. Functional requirements.....	28
4.1.1. Use Case	28
4.1.2. Use case diagram	33
4.1.3. Data flow diagram	34
4.2. Non functional requirements.....	34
4.2.1. Product requirement	34
4.2.2. Organizational requirement	35
4.2.3. External requirement.....	35
4.3. Organizing system flow.....	36
4.3.1. Context model	36
4.3.2. Process model.....	37
4.3.3. interaction model	37
4.3.4. behavior model.....	37
4.4. System Architecture	39
5. SUPPORTING INFROMATION	40
5.1. software requirement specification	40
5.2. Document history	40

LIST OF FIGURES

Figure 1 Design of Landing page for unregistered	14
Figure 2 Design of Landing page for Registered	14
Figure 3 Login page	16
Figure 4 Lecture page	17
Figure 5 Practice page.....	19
Figure 6 My page	20
Figure 7 Question board	21
Figure 8 Writing Question page.....	23
Figure 9 Question page.....	24
Figure 10 Use Case Diagram	33
Figure 11 Data Flow Diagram.....	34
Figure 12 Context Model	36
Figure 13 Process Model.....	37
Figure 14 Sequence Diagram	38
Figure 15 System Architecture.....	39

LIST OF TABLES

Table 1 Table of Acronyms and Abbreviation	6
Table 2 Table of terms and definitions	7
Table 3 Table of Product Functions	8
Table 4 Use Interface of Landing page - Unregistered.....	13
Table 5 Use Interface of Landing page - Registered	15
Table 6 User Interface of Login page	16
Table 7 User Interfaces of Lecture page	17
Table 8 User Interfaces of Practice page	18
Table 9 User Interfaces of My page	19
Table 10 User Interfaces of Question board.....	20
Table 11 User Interfaces of Writing question page	22
Table 12 User Interfaces of Question page.....	23
Table 13 Hardware Interface of Client.....	25
Table 14 Hardware Interface of Server.....	25
Table 15 Software Interface of DB.....	25
Table 16 Software Interface of Account API.....	26
Table 17 Software Interface of Linux OS.....	26
Table 18 Communication Interface of Web browser.....	27
Table 19 Functional requirements abstraction.....	28
Table 20 Use Case of Lecture Information.....	29
Table 21 Use Case of Lecture Like.....	29
Table 22 Use Case of Login	29
Table 23 Use Case of Sign up	30
Table 24 Use Case of My page	30
Table 25 Use Case of Taking a lecture	30
Table 26 Use Case of Practicing code	31
Table 27 Use Case of Lecture Completion	31
Table 28 Use Case of Writting a question.....	31
Table 29 Use Case of Question board.....	32
Table 30 Use Case of Comment.....	32
Table 31 Document history.....	40

1. INTRODUCTION

1.1. PURPOSE

이 문서는 CrawlLearn(비전공자를 위한 파이썬 교육 프로그램) 서비스를 제공하기 위한 요구 명세서이다. 이 서비스는 성균관대학교 소프트웨어공학개론의 Team2 에 의해서 디자인되고 운용된다. 이 문서에는 서비스를 위한 요구사항이 분석, 정리 되어있으며 이 문서에 나와있는 콘텐츠에 의거하여 서비스가 구성되고 제공된다.

Team2 가 서비스의 주요 독자이며 이 명세서에 의거하여 Team2 가 비전공자를 위한 교육 프로그램을 제공한다. 또한 소프트웨어공학개론의 교수님과 조교님들이 이 명세서의 주요 독자가 될 수 있다.

이 문서의 목적은 비전공자 파이썬 교육 프로그램 CrawlLearn 을 제공하기 위하여 웹을 디자인하고 시행하는 과정에서 요구사항을 정리하는 것이다. 비전공자들의 프로그래밍 역량과 컴퓨팅 사고력이 요구되는 현 시점에서 여러 대학들이 소프트웨어 교육을 실시하고 있다. 이러한 현 실태에 맞춰 파이썬 기본 문법을 배운 비전공자들이 한 단계 더 심화된 학습과 실습을 통해 보다 발전된 역량을 기르는 것을 목적으로 시행되는 CrawlLearn 에 필요한 웹 디자인, 프론트엔드와 백엔드, 서버와 DB 의 사용을 위한 요구사항들을 정리하였다.

1.2. SCOPE

우리는 성균관대 학생을 위한 파이썬 실습 웹사이트를 만들 것이다. 이를 위해 인터넷이 되는 환경이라면 어떤 성균관대학교 학생이든 실습을 진행할 수 있는 것을 목표로 하고 있다. 이를 위해 먼저 UI 는 사용자의 교육이 따로 필요 없어도 바로 사용할 수 있도록 디자인 될 것이다. 서버는 시간당 5000 명의 Chrome 을 사용하는 사용자를 수용할 수 있어야하며 오류율은 10%를 초과하지 않을 것이다. 또한 학생들의 개인정보를 위해 데이터 베이스는 HIPPA 요구사항을 만족해야한다.

1.3. DEFINITION, ACRONYMS, AND ABBREVIATION

Table 1 Table of Acronyms and Abbreviation

Acronyms/Abbreviation	Description
CPU	Central Processing Unit

OS	Operating System
API	Application Prgramming Interface
HTTP	Hypertext Transfer Protocol

Table 2 Table of terms and definitions

Term	Definition
Registered	로그인이 되어있는 사용자를 의미
UnRegsitered	로그인이 되어있지 않은 사용자를 의미
Host	우리의 시스템을 의미
Client	우리 시스템을 사용하는 성균관대학교 학생을 의미
Google API	OAuth 인증을 위한 Google 의 API 를 의미
Back-End	서버와 데이터베이스를 포함한 부분을 의미
Front-End	웹 페이지를 구성하고 서버와 통신을 하는 부분을 의미

1.4. REFERENCES

- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements
- Specifications, In IEEEExplore Digital Library
- <http://ieeexplore.ieee.org/Xplore/guesthome.jsp>
- Ian Sommerville, Software Engineering-10edition

1.5. OVERVIEW

본 소프트웨어 요구 사항 명세서의 다섯개의 장으로 구성되어 있다. 두 번째 장에서는 제품에 대한 전반적인 설명을 제공한다. 제품 개관과 기능, 운영 환경 및 가정 등을 포함한다. 세 번째 장에서는 여러 인터페이스, 시스템 기능 및 다른 시스템과의 시스템 상호 작용을 포함한다. 사용자 인터페이스와 하드웨어 인터페이스, 프로그램 서비스 요구사항 등을 포함하고, 클라우드 통신망에 관한 사항을 포함한다. 네번째 장에서는 시스템 요구사항을 명시한다. 기능적인 요구사항과 비기능적 요구사항, 조직적 시스템 플로우를 포함한다. 마지막으로, 다섯 번째 장에서 프로젝트 진행에 있어 history 를 기술한다.

2. OVERALL DESCRIPTION

2.1. PRODUCT PERSPECTIVE

프로그래머에 대한 수요가 증가하고 있는 지금 대학에서는 비전공자 학생들을 상대로 코딩 교육을 진행하고 있다. 성균관대학교에서도 2016 년도부터 전교생을 대상으로 파이썬 교육을 시작했다. 파이썬에 대한 기초 문법을 배우고 알고리즘을 이용한 학생들의 작품 전시회를 개최하여 공개 투표를 통해 우수 작품에는 수상을 하는 행사를 시행하고 있다.

하지만 비전공자 학생들은 이 행사가 끝나면 더 이상 교육을 받을 플랫폼, 강의의 부재로 인해 배운 알고리즘과 향상시킨 컴퓨팅 사고력을 활용하지 못하고 있다. 지속적인 학습 경험과 실무 경험을 쌓아야 소프트웨어 교육의 취지에 맞는 결과물이 나올 수 있다. 이러한 교육 단절 사태를 작게나마 보완하고자 '파이썬을 이용한 웹크롤링 교육 시스템(웹 사이트)'을 만들 예정이다. 비전공자 학생들이 이 교육 웹 사이트에서 파이썬의 기초 문법, HTML/CSS, 웹크롤링 강의를 듣고 실습을 통해 강의 내용을 활용해봄으로써 실무에서도 배웠던 프로그래밍 기법들을 사용할 수 있도록 도와주는 것이 이 프로젝트의 목표이다.

2.2. PRODUCT FUNCTIONS

Table 3 Table of Product Functions

Function Requirement No.	Name of Function	Function Description
F1	강의 정보 불러오기	사용자가 수강 가능한 강의 정보를 불러온다.
F2	강의 검색 및 정렬	사용자가 검색한 기준에 따라 강의를 검색할 수 있어야한다.
F3	강의 좋아요 누르기	사용자가 강의에 좋아요를 누를 수 있어야한다.
F4	로그인 하기	사용자가 구글계정으로 서비스에 로그인 할 수 있어야한다.
F5	회원가입 하기	사용자가 서비스에 회원가입을 할 수 있어야한다.
F6	개인 정보 조회 및 수정	사용자가 개인의 정보를 조회하고 수정할 수 있어야한다.
F7	강의 수강	사용자가 강의를 수강할 수 있어야한다.

F8	강의 코드 작성 및 실행	사용자가 실습 코드를 작성하고 실행할 수 있어야한다.
F9	강의 수강 완료 처리	사용자가 강의를 수강 완료한것을 처리할 수 있어야한다.
F10	질문 작성	사용자가 강의에 대해 질문을 작성할 수 있어야한다.
F11	질문 보여주기	사용자가 질문의 상세 내용을 볼 수 있어야한다.
F12	댓글 작성	사용자가 댓글을 작성할 수 있어야한다.

2.2.1. LANDING PAGE

처음 웹 사이트에 들어가면 랜딩 페이지에 도착한다. 오른쪽 상단에 있는 'About Us'를 누르면 웹 사이트 제작자에 대한 정보가 나온다. 오른쪽 하단에 있는 '도움말'을 통해서 웹 사이트의 개괄적인 이용 방법을 알 수 있다. 좌측 하단에는 성균관대학교 자연과학 캠퍼스의 실제 주소와 웹 사이트 관리자의 이메일 주소가 적혀있다. 로그인 하지 않은 상태로 첫 랜딩 페이지에서는 강의 목차, 강의 제목 등의 개괄적인 정보만 얻을 수 있다. 이 외의 다른 곳을 누르게 되면 로그인/회원가입 url로 넘어 가게 된다.

2.2.2. LOGIN/START

로그인, 회원가입 사이트로 넘어가면 로그인하기 항목이 나온다. Google 계정 혹은 성균관대학교 아이캠퍼스 계정이 있으면 이와 연동하여 로그인이 가능하다. 계정이 없는 경우에는 하단에 파란색으로 되어있는 가입하기 버튼을 누르면 가입절차가 진행된다.

2.2.3. LANDING PAGE FOR USER

로그인을 완료하면 다시 랜딩 페이지로 돌아오게 된다. 첫 랜딩 페이지에는 없었던 MY와 로그아웃 버튼이 오른쪽 상단에 생긴다. MY에서는 나의 정보, 수강 목록 등의 정보가 보여진다. 로그인 후에는 강의 목차, Q&A 등의 항목을 이용할 수 있다. 페이지 중앙에 있는 강의 제목과 강의에 대한 칸에는 해당 강의의 난이도를 나타내주는 별 모양의 표시가 있다. 사용자는 이를 통해 본인이 수강할 혹은 수강하고 있는 강의에 대한 정보를 얻을 수 있다.

2.2.4. LECTURE

로그인 후 이용할 수 있는 랜딩 페이지의 강의 목록 칸을 누르면 본격적인 강의가 시작된다. 상단에는 강의 제목, 강의 소제목과 이에 대한 짧은 설명이 있다. 그 아래에는 마지막 업데이트 시간을 알려주고 강의에 대한 '좋아요'를 누를 수 있는 기능이 있다. 가운데 회색 부분에 강의 제목에 해당하는 내용이 적혀있고 학생들은 이를 통해 해당하는 강의를 학습할 수 있다. 왼쪽 부분에는 강의 듣기와 Q&A 작성 버튼이 있다. 강의 듣기를 누르면 해당 강의 내용이 제공될 것이다. Q&A 작성 버튼을 누르면 Q&A를 통해 질의응답을 할 수 있는 게시판으로 이동하게 된다.

2.2.5. CODING PRACTICE

강의를 수강한 후에는 수강한 강의에 대한 연습문제를 풀어봄으로써 복습과 응용을 할 수 있는 기회가 주어진다. 이론적인 학습뿐만 아니라 실습을 통해서 학생들은 강의에 대한 이해를 높이고 프로그래밍을 활용할 수 있는 능력을 길러 학습 목적을 달성할 수 있다. 왼쪽 상단에는 강의 이름이 적혀 있으며 같은 줄 오른쪽 상단에는 dark 모드와 light 모드를 설정할 수 있는 버튼이 존재한다. 그 옆에는 실행 버튼이 존재하는데 이를 이용하면 유저가 작성한 코드를 실행할 수 있다. '파일.py' 항목 아래 칸에 유저는 자신의 코드를 작성할 수 있으며 그 결과는 '결과' 항목 아래에 나타나게 된다. 오른쪽 상단에는 컴파일 옵션이 있으며 왼쪽 하단에는 Q&A 버튼을 이용하여 언제든지 Q&A 게시판으로 넘어 갈 수 있다.

2.2.6. MY PAGE

랜딩 페이지에서 로그인한 유저가 오른쪽 상단에 있는 'My' 버튼을 클릭하면 '마이페이지'로 이동할 수 있다. 기본 정보에서는 본인의 이름과 이메일을 오른쪽 상단에 있는 수정하기 버튼을 통해 수정할 수 있다. 활동 내역에서는 수강한 강의 목록을 확인하고 Q&A 게시판에 본인이 작성한 글을 확인 할 수 있다.

2.2.7. Q&A

Q&A 게시판에서는 유저가 학습을 하다 의문점이 생기면 질문을 작성하고 응답을 얻을 수 있다. 오른쪽 상단의 검색 기능을 통해 질문을 검색할 수 있으며 원하는 정보 순으로 정렬도 가능하다. 중앙에는 질문의 제목과 질문자 그리고 질문 시각을 확인할 수 있으며 그에 대한 댓글도 확인이 가능하다.

2.2.8. ASKING QUESTION

Q&A 를 작성하는 페이지에는 강의명, 제목, 내용을 작성하게 되어있다. 강의명 목록에서 본인이 수강하였던 강의 중 질문에 해당하는 강의를 선택할 수 있다. 질문의 제목을 만들고 내용 칸에 질문의 내용을 작성하고 등록 버튼을 클릭하면 질문이 Q&A 게시판에 등록된다.

2.2.9. Q&A DETAIL

Q&A 게시판에 작성되어 있는 질문 목록 중 하나를 클릭하면 대항 질문에 관한 자세한 질의응답이 이루어진 페이지로 이동한다. 질문의 작성자, 작성 날짜, 질문 경로 등의 정보와 함께 질문의 내용이 제시되며 그 아래에는 댓글의 작성자와 작성 날짜가 기록된다.

2.3. USER CLASSES AND CHARACTERISTICS

2.3.1. SYSTEM ADMINISTRATOR

시스템 관리자는 웹 페이지에 대한 이해를 가지고 있어야 한다. 어떠한 기능들을 제공하고 있으며 유저들이 어떠한 과정을 통해 학습을 하는지 알고 있어야 한다. 이 프로그램에서 제공하는 커리큘럼에 대한 전반적인 지식인 파이썬의 기본 문법과 이를 통한 웹 크롤링 방법을 알고 있어야 하며 프론트엔드와 백엔드에 대한 지식도 갖추고 있어야 한다. 소프트웨어 전공자가 적합할 것이며 유저들의 불만 사항을 처리할 수 있어야 한다.

2.3.2. USER

이 프로그램의 유저는 기본적으로 성균관대학교 전교생이 될 것이다. 신입생들은 2 학기에 걸쳐 의무적으로 프로그래밍 교육을 받게 되는데 이 두 과목을 이수한 학생들이 유저가 될 것이다. 기본적인 파이썬 문법에 대하여 학습과 실습이 이루어진 상태여야 이 프로그램을 통해 조금 더 심화된 웹크롤링에 대해서 학습하고 문제를 해결할 수 있을 것이다.

2.3.3. LECTURER

유저들이 이 교육 프로그램을 이용할 때 질문 게시판을 이용하여 의문점을 해결하는데 도움을 주기 위해 질문에 답변을 할 능력이 있는 사람이 될 것이다. 이 파이썬 웹 크롤링 교육 프로그램의 커리큘럼을 알고 있으며 그에 대한 지식도 갖춘 소프트웨어 전공자가 적합할 것이다.

2.4. OPERATING ENVIRONMENT

- Window 2000
- Window XP
- Mac OS
- Linux
- Window 7
- Window 8
- Window 10

2.5. DESIGN AND IMPLEMENTATION CONSTRAINTS

- 가능한 공개된 오픈소스를 주로 사용
- 별도의 라이선스 혹은 비용을 지불해야 하는 소프트웨어의 사용을 자제
- 전반적인 시스템 성능을 향상시킬 수 있는 방향으로 개발
- 유저가 친숙하고 편리하게 사용할 수 있는 인터페이스를 구축
- 시스템의 유지 보수 비용을 고려하여 개발
- 시스템 리소스를 낭비하지 않도록 소스코드를 최적화

2.6. USER DOCUMENTATION

유저는 랜딩 페이지와 코딩 페이지에 있는 도움말을 통해 프로그램 이용에 도움을 받을 수 있다. 도움말 페이지에 들어가면 이와 같이 각 기능들에 대한 가벼운 설명이 표시되어 있어 유저들은 더 쉽게 인터페이스에 접근할 수 있다.

2.7. ASSUMPTIONS AND DEPENDENCIES

이 명세서에 명시된 시스템은 웹을 통해 운용되고 제공되는 것을 전제한다. 따라서 이 시스템의 유저는 laptop 혹은 노트북을 사용하여 인터넷에 접속할 수 있어야 한다.

3. EXTERNAL INTERFACE REQUIREMENTS

3.1. USER INTERFACE

Table 4 Use Interface of Landing page - Unregistered

이름	랜딩 및 메인 페이지-로그인 전
목적/설명	로그인 전 및 미등록 사용자는 강의를 클릭하여 강의 상세 설명 페이지로 진입할 수 있으며 검색 창과 정렬 버튼을 사용하여 이 사이트의 강의를 찾을 수 있다. 로그인 전 및 미등록 사용자는 '로그인' 버튼을 클릭하여 로그인 및 가입 페이지로 진입할 수 있다. 상단의 'About Us', 하단의 '도움말' 버튼으로 해당 사이트 정보를 얻을 수 있다.
입력 소스/ 출력 대상	클라이언트/호스트서버 호스트서버/클라이언트
범위/ 정확도/ 오차 범위	한 페이지에 최대 9 개의 강의/ 정렬 알고리즘의 정확도/ 정렬 알고리즘의 오차 범위를 따름
구성 단위	화면
시간/속도	사용자가 해당 항목을 클릭했을 때/클라이언트와 서버간 통신 속도
기타 입출력관계	N/A
화면 형식과 구성	<ol style="list-style-type: none"> 상단 'About Us', 하단 '도움말' 버튼은 사이트 정보를 페이지로 링크된다. 헤더에는 사이트 이름과 사이트 설명이 있다. 바디의 상단 네비게이션 바에는 '강의', 'Q&A', '로그인' 버튼이 있으며 강의를 제외한 버튼을 누르면 로그인 화면으로 전환된다. 바디의 왼쪽 네비게이션 바에는 '강의' 컴포넌트에 해당하는 강의 목차 리스트 버튼이 있다. 해당 커리큘럼의 대분류에 따라 구성된다. 클릭 시 하위 커리큘럼이 표시되고, 하위 커리큘럼 클릭 시 바디에 표시된 강의 종류가 해당 분류에 따라 바뀐다. 3 번 밑에 있는 검색 창에서는 강의 제목으로 검색할 수 있다. 5 번 옆 정렬 버튼으로 현재 표시되는 강의를 해당 값으로 정렬할 수 있다. 바디의 강의 레이아웃에는 강의 테이블과 순서가 있다. <ol style="list-style-type: none"> 강의 테이블은 강의 컴포넌트를 3x3 테이블로 표시하며 강의 컴포넌트는 제목, 강의 설명, 난이도, 수강자, 좋아요 수의 정보를 제공한다. 순서 선택 레이아웃에는 '이전', '다음' 버튼과 1-10 까지 수의 버튼이 있어 해당 레이아웃을 표시할 수 있다.
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 메시지	N/A



Figure 1 Design of Landing page for unregistered

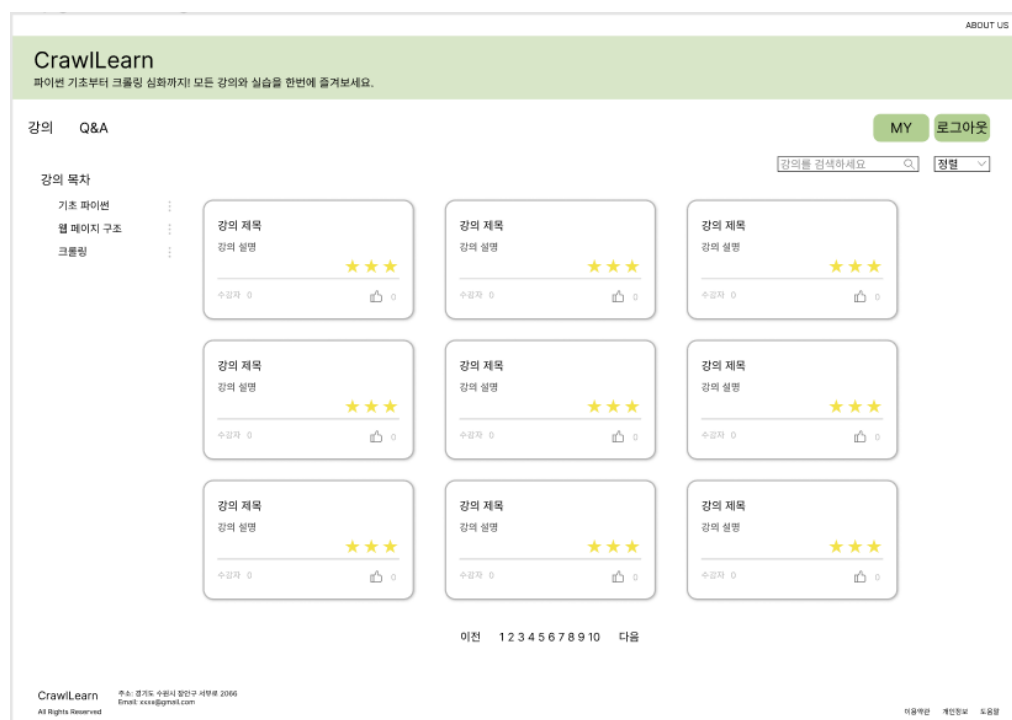


Figure 2 Design of Landing page for Registered

Table 5 Use Interface of Landing page - Registered

이름	랜딩 및 메인 페이지-로그인 후
목적/설명	<p>사용자는 강의를 클릭하여 강의 상세 설명 페이지로 진입할 수 있으며 검색 창과 정렬 버튼을 사용하여 이 사이트의 강의를 찾을 수 있다.</p> <p>사용자는 '로그아웃' 버튼을 클릭하여 해당 사이트에서 로그아웃 할 수 있으며 'MY' 버튼을 클릭하여 개인 정보 페이지에 진입할 수 있다.</p> <p>'Q&A' 버튼을 클릭하여 질문 게시판을 이용할 수 있다.</p> <p>상단의 'About Us', 하단의 '도움말' 버튼으로 해당 사이트 정보를 얻을 수 있다.</p>
입력 소스/ 출력 대상	<p>클라이언트/호스트서버</p> <p>호스트서버/클라이언트</p>
범위/ 정확도/ 오차 범위	<p>한 페이지에 최대 9 개의 강의/ 정렬 알고리즘의 정확도/ 정렬 알고리즘의 오차 범위를 따름</p>
구성 단위	화면
시간/속도	사용자가 해당 항목을 클릭했을 때/클라이언트와 서버간 통신 속도
기타 입출력과 관계	N/A
화면 형식과 구성	<ol style="list-style-type: none"> 상단 'About Us', 하단 '도움말' 버튼은 사이트 정보를 페이지로 링크된다. 헤더에는 사이트 이름과 사이트 설명이 있다. 바디의 상단 네비게이션 바에는 '강의', 'Q&A', 'MY', '로그아웃' 버튼이 있으며 클릭 시 해당 페이지 및 레이어로 이동한다. 바디의 왼쪽 네비게이션 바에는 '강의' 컴포넌트에 해당하는 강의 목차 리스트 버튼이 있다. 해당 커리큘럼의 대분류에 따라 구성된다. 클릭 시 하위 커리큘럼이 표시되고, 하위 커리큘럼 클릭 시 바디에 표시된 강의 종류가 해당 분류에 따라 바뀐다. 3 번 밑에 있는 검색 창에서는 강의 제목으로 검색할 수 있다. 5 번 옆 정렬 버튼으로 현재 표시되는 강의를 해당 값으로 정렬할 수 있다. 바디의 강의 레이어에는 강의 테이블과 순서가 있다. <ol style="list-style-type: none"> 강의 테이블은 강의 컴포넌트를 3x3 테이블로 표시하며 강의 컴포넌트는 제목, 강의 설명, 난이도, 수강자, 좋아요 수의 정보를 제공한다. 순서 선택 레이어에는 '이전', '다음' 버튼과 1-10 까지 수의 버튼이 있어 해당 레이어를 표시할 수 있다.
데이터 유형	이미지, 텍스트, 위젯
명령 유형	각 위젯마다 대응되는 실행 코드 값
종료 메시지	N/A

Table 6 User Interface of Login page

이름	로그인 페이지
목적/설명	사용자는 구글 계정을 통해 로그인 또는 회원가입을 할 수 있다.
입력 소스/ 출력 대상	클라이언트/호스트서버(본 사이트) 호스트서버/구글서버, 구글서버/호스트서버
범위/ 정확도/ 오차 범위	N/A
구성 단위	화면
시간/속도	N/A
기타 입출력과 관계	사용자 계정 정보의 유효함을 구글 API와 연동하여 그 값을 받는다.
화면 형식과 구성	1. Google 계정으로 로그인' 클릭 시 연동된 API 로그인 화면으로 이동한다. 2. 파란색의 '가입하기' 클릭 시 연동된 API 가입 화면으로 이동한다.
데이터 유형	암호화된 API 키, 쿼리
명령 유형	'가입하기'와 로그인 버튼에 해당하는 명령
종료 메시지	가입 절차 시 '가입 완료!'

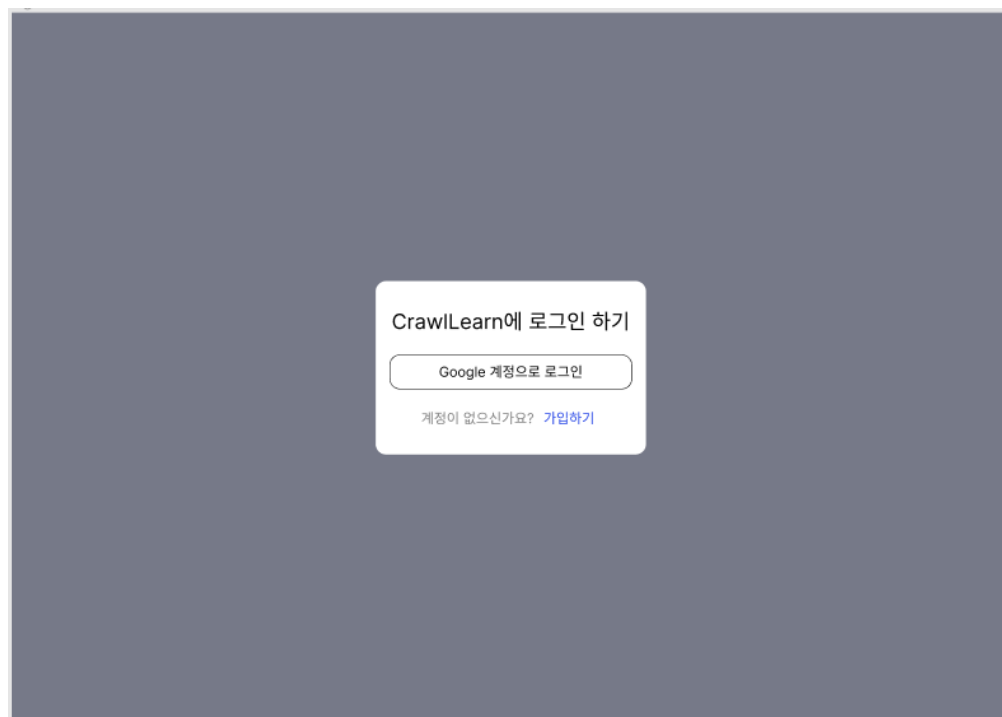


Figure 3 Login page

Table 7 User Interfaces of Lecture page

이름	강의 상세 페이지
목적/설명	사용자는 제시된 강의 설명을 읽고 강의 듣기 및 Q&A 를 작성할 수 있다.
입력 소스/ 출력 대상	클라이언트/호스트서버 호스트서버/클라이언트
범위/정확도/ 오차 범위	N/A
구성 단위	화면
시간/속도	N/A
기타 입출력과 관계	사용자 로그인 정보를 바탕으로 해당 사용자 데이터베이스에 업데이트
화면 형식과 구성	<ol style="list-style-type: none"> 1. 페이지 헤더에는 홈페이지 이름과 강의 분류에 따른 경로를 표시한다. 2. 위 네비게이션 바에는 강의 제목, 소제목 및 짧은 설명, 좋아요 수, 마지막 업데이트 날짜 정보를 표시한다. '좋아요'는 로그인 사용자가 클릭 시 해당 정보를 사용자 개인 정보 데이터베이스에 업데이트 한다. 3. 왼쪽 네비게이션 바에는 '강의듣기'와 'Q&A 작성' 버튼이 있다. <ol style="list-style-type: none"> 3-1) '강의 듣기'버튼 클릭 시 Practice 페이지로 이동하며 사용자 개인 정보에 해당 데이터를 업데이트한다. 3-2) 'Q&A 작성' 버튼 클릭 시 질문 작성 페이지로 이동한다. 4. 강의 설명 페이지에는 이미지와 텍스트로 해당 강의 내용이 설명되어 있다.
데이터 유형	텍스트, 이미지
명령 유형	'강의 듣기', 'Q&A 작성' 버튼에 따른 명령
종료 메시지	N/A



Figure 4 Lecture page

Table 8 User Interfaces of Practice page

이름	Practice 페이지
목적/설명	<p>사용자는 좌측 강의 레이어에서 강의 내용을 읽으며 실습을 진행한다.</p> <p>사용자는 우측 코드 레이어에서 실습 코드를 작성하고 상단에 있는 실행 버튼을 눌러 자신의 코드를 실행할 수 있다.</p> <p>사용자는 하단의 결과 레이어에서 작성한 코드의 실행 결과 및 오류 메시지를 확인하며 실습을 진행할 수 있다.</p>
입력 소스/ 출력 대상	<p>클라이언트/호스트서버</p> <p>호스트서버/클라이언트</p>
범위/ 정확도/ 오차 범위	<p>해당 실습 페이지의 사용자의 실습 코드/ 파이썬 인터프리터와 사용자의 크롤링 실습 코드에 따른 정확도/ 파이썬 인터프리터와 사용자의 크롤링 실습 코드에 따른 오차</p>
구성 단위	화면
시간/속도	<p>사용자가 실행 버튼을 눌렀을 때/ 실행 코드의 복잡도 및 클라이언트와 서버간 통신 속도</p>
기타 입출력과 관계	<ol style="list-style-type: none"> 1. 해당 코드를 실행시키는 파이썬 인터프리터의 결과를 사용자에게 표시 2. 크롤링 코드의 대상 사이트에서 가져온 자료를 사용자에게 표시 3. 사용자 로그인 정보를 바탕으로 해당 사용자 데이터베이스에 업데이트
화면 형식과 구성	<ol style="list-style-type: none"> 1. 페이지 헤더에는 홈페이지 이름과 강의 분류에 따른 경로를 표시한다. 2. 바디의 네비게이션 바에는 강의 이름이 명시되어 있으며 화면 배경 색깔을 선택하는 'dark', 'light' 버튼과 '실행' 버튼이 있다. 2-2) 'dark' 버튼은 페이지 바디 영역을 어두운 회색으로 바꾸고 'light' 버튼은 배경 색을 희색으로 바꾼다. 2-3) '실행' 버튼은 실행 파일 영역에 작성한 코드를 서버에 전달하고 실행 결과에 서버로부터 받은 실행 결과를 출력하는 버튼이다. 3. 네비게이션 바 밑 좌측에는 강의 내용이 표시된다. 4. 강의 내용 우측에는 'Code Editor' 영역이 있다. 이 영역의 상단에는 파이썬 파일 이름이 표시되어 있고 그 밑에 사용자가 코드를 작성할 수 있는 영역이 있다. 이 'Code Editor'는 줄 맞춤과 파이썬 키워드 별 다른 글자 색깔 서비스를 제공해 준다. 5. 'Code Editor'영역 하단에 결과 영역에는 사용자의 코드를 실행한 결과를 서버로부터 받아 이를 출력한다. 6. 페이지 좌측 하단 Q&A 버튼은 클릭 시 질문 작성 페이지로 넘어가며 질문 경로는 자동으로 해당 실습 강의 내용으로 정해진다. 7. 페이지 우측 하단 '도움말' 버튼 클릭 시 도움말 페이지 문서를 제공하여 화면 구성에 관한 정보를 제공한다.
데이터 유형	텍스트, 이미지
명령 유형	'실행' 및 'dark', 'light' 버튼에 해당하는 명령
종료 메시지	'해당 실습 파일이 저장되었습니다.'

The screenshot shows a web interface for a coding practice session. At the top, there's a green navigation bar with the text 'CrawlLearn' and a breadcrumb '강의 대분류 > 강의 소분류'. On the right of this bar is a '컴파일 옵션' (Compile Options) button. Below the navigation bar, the main content area is a form. It starts with a '강의 이름' (Lecture Name) input field. To its right are three buttons: 'dark', 'light', and '실행' (Run). The form is then divided into two vertical sections. The left section is labeled '파일.py' (File.py) and is intended for code input. The right section is labeled '결과' (Result) and is for the output. At the bottom of the form, there are two buttons: 'Q&A' on the left and '도움말' (Help) on the right.

Figure 5 Practice page

Table 9 User Interfaces of My page

이름	My 페이지
목적/설명	사용자는 자신이 가입한 이메일 주소를 확인할 수 있으며 이 사이트에서 사용하는 이름을 변경할 수 있다. 또한 활동 내역으로 수강한 강의, 작성한 Q&A, 좋아요 표시한 강의를 확인할 수 있다.
입력 소스/ 출력 대상	클라이언트/호스트서버
범위/ 정확도/ 오차 범위	N/A
구성 단위	화면
시간/속도	N/A
기타 입출력과 관계	구글 API 연동
화면 형식과 구성	<ol style="list-style-type: none"> 상단에는 기본 정보 레이어 하단에는 활동 내역 레이어로 구성된다. 기본 정보 레이어에는 이름과 가입시 사용한 이메일 주소, '수정하기' 버튼이 있다. <ol style="list-style-type: none"> 이름은 사용자가 이 사이트에서 이용하는 별칭으로 사용자 별 중복 가능하며 '수정하기' 버튼을 클릭하여 수정 또한 가능하다. 가입시 사용한 이메일은 변경이 불가능하다. 해당 사이트 탈퇴를 위해서는 가입한 이메일 주소의 관리 사이트에서 관리 가능하다.

	<p>3. 활동내역 레이어에는 ‘수강 강의’, ‘Q&A’, ‘좋아요 표시한 강의’ 위젯이 존재한다.</p> <p>3-1) ‘수강 강의’ 클릭 시 사용자가 그동안 수강한 강의 리스트를 확인할 수 있다.</p> <p>3-2) ‘Q&A’ 클릭 시 사용자가 작성한 질문 리스트와 질문 리스에 작성한 댓글리스트를 확인할 수 있다.</p> <p>3-3) ‘좋아요 표시한 강의’ 클릭 시 사용자가 강의에 좋아요 표시한 강의 리스트를 확인할 수 있다</p>
데이터 유형	텍스트
명령 유형	‘수정하기’와 ‘수강 강의’, ‘Q&A’, ‘좋아요 표시한 강의’ 버튼에 해당하는 명령
종료 메시지	N/A

Figure 6 My page

Table 10 User Interfaces of Question board

이름	질문 게시판 페이지
목적/설명	<p>사용자는 질문을 클릭하여 작성된 질문 내용 페이지로 진입할 수 있으며 검색 창과 정렬 버튼을 사용하여 이 사이트의 질문을 검색 및 정렬할 수 있다.</p> <p>사용자는 ‘강의’ 버튼을 클릭하여 강의 레이어로 이동할 수 있다.</p> <p>상단의 ‘About Us’, 하단의 ‘도움말’ 버튼으로 해당 사이트 정보를 얻을 수 있다.</p>
입력 소스/ 출력 대상	<p>클라이언트/호스트서버</p> <p>호스트서버/클라이언트</p>
범위/ 정확도/	<p>한 페이지에 최대 5 개의 질문/</p> <p>정렬 알고리즘의 정확도/</p>

오차 범위	정렬 알고리즘의 오차 범위를 따름
구성 단위	화면
시간/속도	사용자가 정렬 항목을 클릭했을 때/클라이언트와 서버간 통신 속도
기타 입출력과 관계	N/A
화면 형식과 구성	<ol style="list-style-type: none"> 상단 'About Us', 하단 '도움말' 버튼은 사이트 정보를 페이지로 링크된다. 헤더에는 사이트 이름과 사이트 설명이 있다. 바디의 상단 네비게이션 바에는 '강의', 'Q&A', '작성하기' 버튼이 있으며 클릭 시 해당 페이지로 이동한다. 3 번 밑에 있는 검색 창에서는 질문 제목으로 검색할 수 있다. 4 번 옆 정렬 버튼으로 현재 표시되는 질문을 해당 값으로 정렬할 수 있다. 질문 바디에는 질문 테이블 레이어와 그 하단에 순서 레이어가 있다. <ol style="list-style-type: none"> 질문 테이블에는 질문 컴포넌트가 5 개씩 표시된다. 질문 컴포넌트는 질문의 제목, 질문 작성자, 업로드 시각, 댓글 수 가 표시된다. 질문 컴포넌트를 클릭하면 작성된 해당 질문 페이지로 이동한다. 순서 선택 레이어에는 '이전', '다음' 버튼과 1-10 까지 수의 버튼이 있어 해당 레이어를 표시할 수 있다.
데이터 유형	텍스트
명령 유형	'강의', 'Q&A', '작성하기', '정렬' 버튼과 질문, 검색 컴포넌트에 해당하는 명령
종료 메시지	N/A

CrawlLearn
파이썬 기초부터 크롤링 심화까지! 모든 강의와 실습을 한번에 즐겨보세요.

ABOUT US

강의 Q&A

작성하기

질문을 검색하세요. 정렬

제목

질문자 시각 댓글

제목

질문자 시각 댓글

제목

질문자 시각 댓글

제목

질문자 시각 댓글

제목

질문자 시각 댓글

이전 1 2 3 4 5 6 7 8 9 10 다음

CrawlLearn 주소: 경기도 수원시 팔달구 서우로 2066 Email: xxxxx@gmail.com All Rights Reserved

이동

Figure 7 Question board

Table 11 User Interfaces of Writing question page

이름	질문 작성 페이지
목적/설명	사용자는 강의 내용 혹은 이 사이트를 이용하면서 궁금한 점을 질문으로 작성하여 등록할 수 있다.
입력 소스/ 출력 대상	클라이언트/호스트서버
범위/ 정확도/ 오차 범위	N/A
구성 단위	화면
시간/속도	사용자가 등록을 클릭했을 때/클라이언트와 서버간 통신 속도
기타 입출력과 관계	사용자 로그인 정보를 바탕으로 해당 사용자 데이터베이스에 업데이트
화면 형식과 구성	<ol style="list-style-type: none"> 1. 페이지 헤더에는 'Q&A' 텍스트가 표시되어 있다. 2. 페이지 바디 상단에는 '목록으로' 버튼이 있으며 클릭 시 질문 게시판 페이지로 돌아간다. 작성 중이 질문은 저장되지 않고 사라진다. 3. 질문 작성 레이아웃에는 질문 경로, 제목, 내용, 취소 및 등록 버튼이 있다. <ol style="list-style-type: none"> 3-1) 질문 경로 레이아웃에는 질문을 하게 된 경로를 선택할 수 있으며 특별한 경로가 없을 경우 '특별한 경로 없음' 버튼을 클릭하고, 경로가 존재한다면 '강의명' 버튼을 클릭하면 나타나는 강의명 리스트에서 해당 강의를 선택한다. 만일 강의 상세페이지나 실습 페이지에 존재하는 'Q&A' 버튼을 클릭하면 이 페이지로 이동하고 질문 경로는 해당 강의 경로로 자동 지정된다. 3-2) 제목 레이아웃에는 제목을 입력할 수 있는 텍스트 박스가 있으며 사용자는 해당 영역에 질문 제목을 입력한다. 3-3) 내용 레이아웃에는 질문 내용을 입력할 수 있는 텍스트 박스가 있으면 사용자는 해당 영역에 질문 내용을 입력한다. 3-4) 취소 및 등록 레이아웃에는 '취소'와 '등록' 버튼이 있다. '취소' 버튼을 클릭하면 작성 중 질문은 사라지며 종료 메시지를 표시하며 질문 게시판 페이지로 돌아간다. '등록' 페이지를 클릭 시 작성한 질문은 작성 시간과 함께 호스트서버로 전달되며 사용자에게 종료 메시지를 표시하며 질문 게시판 페이지로 돌아간다. 서버는 전달받은 질문 내용을 질문 게시판 페이지에 표시하며 사용자 개인 정보에 업데이트한다.
데이터 유형	텍스트
명령 유형	질문 경로의 두 버튼과 '취소', '등록', '목록으로' 버튼에 해당하는 값
종료 메시지	<p>'취소', '목록으로': '질문 작성을 취소합니다.'</p> <p>'등록': '질문이 등록되었습니다.'</p>

Q&A

< 목록으로

질문 경로

특별한 경로 없음 강의명

제목

내용

취소 등록

Figure 8 Writing Question page

Table 12 User Interfaces of Question page

이름	질문 페이지
목적/설명	사용자는 질문 내용과 답변 내용을 읽으면서 질문 내용을 습득할 수 있다. 사용자는 질문에 댓글을 작성하면서 이 사이트 공동체에 활발히 참여할 수 있다.
입력 소스/ 출력 대상	클라이언트/호스트서버 호스트서버/클라이언트
범위/ 정확도/ 오차 범위	N/A
구성 단위	화면
시간/속도	N/A
기타 입출력과 관계	사용자 로그인 정보를 바탕으로 해당 사용자 데이터베이스에 업데이트
화면 형식과 구성	<ol style="list-style-type: none"> 1. 페이지 헤더에는 'Q&A' 텍스트가 표시되어 있다. 2. 페이지 바디 상단에는 '목록으로' 버튼이 있으며 클릭 시 질문 게시판 페이지로 돌아간다. 작성 중인 댓글은 저장되지 않고 사라진다. 3. 내부 질문 내용 레이어에는 질문 레이어와 댓글 테이블 레이어로 구성된다. <ol style="list-style-type: none"> 3-1) 질문 레이어는 질문 헤드 영역과 질문 내용 영역으로 나뉜다. 질문 헤드 영역에는 질문 제목, 작성자, 작성 날짜, 질문 경로가 표시된다. 질문 내용은 작성자가 질문 작성 페이지에서 작성한 질문 내용에 해당하는 데이터가 표시된다.

	<p>3-2) 댓글 테이블 레이어에는 해당 질문에 작성된 댓글 컴포넌트와 댓글을 작성할 수 있는 댓글 작성 컴포넌트로 구성된다.</p> <p>3-2-1) 댓글 컴포넌트는 작성자와 작성 날짜 정보가 있는 헤드와 댓글 내용이 있는 바디로 구성된다.</p> <p>3-2-2) 댓글 작성 컴포넌트는 구성은 댓글 컴포넌트 구조와 동일하지만 추가로 '취소', '등록' 버튼이 하단에 있어 클릭 시 각각 작성한 댓글을 취소, 등록한다.</p>
데이터 유형	텍스트
명령 유형	'취소', '등록' 버튼에 해당하는 명령
종료 메시지	N/A

Q&A

< 목록으로

제목

작성자

작성 날짜

질문 경로

질문 내용

작성자

작성 날짜

댓글 내용

취소

등록

Figure 9 Question page

3.2. HARDWARE INTERFACES

Table 13 Hardware Interface of Client

이름	클라이언트 - 사용자가 이 시스템에 접근할 수 있는 기기
목적/설명	네트워크 통신이 가능하고, 기본 입출력 장치가 가능한 기기로 사용자는 웹 브라우저를 통해 이 시스템을 이용할 수 있다.

Table 14 Hardware Interface of Server

이름	서버 - AWS 이용
목적/설명	클라우드 컴퓨팅 서비스 이용하여 서버를 구축한다.
구성	프론트 엔드 서버 2 대와 백 엔드 서버 2 대는 각각 로드 밸런싱 되어있고 이들의 경로는 라우팅 되어있다. 백 엔드 서버에는 데이터 베이스가 연결되어 있다.
구성 단위	Amazon EC2
스펙	vCPU:2, 메모리:1GiB, 기준 성능/vCPU:10%, 획득한 CPU 크레딧/시간:6, 네트워크 버스트 대역폭:최대 5Gbps, EBS 버스트 대역폭:최대 2085Mbps

3.3. SOFTWARE INTERFACES

Table 15 Software Interface of DB

이름	AWS 기반 DB
목적/설명	사용자 정보를 입출력 하기 위한 쿼리 데이터
입력 소스/ 출력 대상	클라이언트/호스트서버 호스트서버/클라이언트
범위/ 정확도/ 오차 범위	Mysql DB 의 정확도
구성 단위	쿼리
시간/속도	즉시 반응
기타 입출력과 관계	서버의 모든 입출력 데이터와 연관된다.
화면 형식과 구성	N/A
데이터 유형	mysql 쿼리
명령 유형	mysql 명령어
종료 메시지	N/A

Table 16 Software Interface of Account API

이름	구글 계정 인증 API
목적/설명	구글에서 제공하는 API 를 사용하여 사용자 인증을 진행한다.
입력 소스/ 출력 대상	호스트서버/구글서버 구글서버/호스트서버
범위/ 정확도/ 오차 범위	N/A
구성 단위	패킷
시간/속도	네트워크 이동 속도
기타 입출력과 관계	사용자가 입력한 구글 계정과 비밀번호
화면 형식과 구성	N/A
데이터 유형	사용자 인증 쿼리
명령 유형	사용자 인증 확인
종료 메시지	N/A

Table 17 Software Interface of Linux OS

이름	Linux OS
목적/설명	이 시스템은 리눅스 운영 시스템 위에서 동작한다.
입력 소스/ 출력 대상	호스트서버/호스트서버
범위/ 정확도/ 오차 범위	운영 시스템의 일관성
구성 단위	N/A
시간/속도	사용자 요청 시 반응 3 초 이내
기타 입출력과 관계	서버의 모든 입출력과 연관된다.
화면 형식과 구성	N/A
데이터 유형	모든 타입의 데이터
명령 유형	리눅스 시스템 명령어
종료 메시지	N/A

3.4. COMMUNICATION INTERFACES

Table 18 Communication Interface of Web browser

이름	웹 브라우저
목적/설명	이 시스템 사용자는 웹 브라우저를 통해 본 서버와 통신하며 사이트에서 제공하는 기능을 사용할 수 있다.
입력 소스/ 출력 대상	클라이언트/호스트서버 호스트서버/클라이언트
범위/ 정확도/ 오차 범위	해당 프로토콜의 정확도
구성 단위	패킷
시간/속도	네트워크와 해당 프로토콜 알고리즘의 속도
기타 입출력과 관계	사용자와 서버 간 모든 데이터 입출력
화면 형식과 구성	N/A
데이터 유형	TCP, TLS, UDP, QUICK 등 웹 브라우저가 사용하는 프로토콜 데이터
명령 유형	해당 프로토콜의 명령
종료 메시지	N/A

4. SYSTEM REQUIREMENTS

4.1. FUNCTIONAL REQUIREMENTS

4.1.1. USE CASE

Table 19 Functional requirements abstraction

Function Requirement No.	Name of Function	Function Description
F1	강의 정보 불러오기	사용자가 수강 가능한 강의 정보를 불러온다.
F2	강의 검색 및 정렬	사용자가 검색한 기준에 따라 강의를 검색할 수 있어야한다.
F3	강의 좋아요 누르기	사용자가 강의에 좋아요를 누를 수 있어야한다.
F4	로그인 하기	사용자가 구글계정으로 서비스에 로그인 할 수 있어야한다.
F5	회원가입 하기	사용자가 서비스에 회원가입을 할 수 있어야한다.
F6	개인 정보 조회 및 수정	사용자가 개인의 정보를 조회하고 수정할 수 있어야한다.
F7	강의 수강	사용자가 강의를 수강할 수 있어야한다.
F8	강의 코드 작성 및 실행	사용자가 실습 코드를 작성하고 실행할 수 있어야한다.
F9	강의 수강 완료 처리	사용자가 강의를 수강 완료한것을 처리할 수 있어야한다.
F10	질문 작성	사용자가 강의에 대해 질문을 작성할 수 있어야한다.
F11	질문 보여주기	사용자가 질문의 상세 내용을 볼 수 있어야한다.
F12	댓글 작성	사용자가 댓글을 작성할 수 있어야한다.

Table 20 Use Case of Lecture Information

이름	Lecture Information
사용자	REGISTERED/UNREGISTERED
설명	현재 수강 가능한 강좌에 대한 강의 정보를 불러올 수 있어야한다. 사용자는 메인 페이지에서 해당 내용을 확인 할 수 있으며 강의의 제목, 상세설명, 난이도, 수강 학생 수, 좋아요 갯수를 확인 할 수 있어야한다.
정상 프로세스	사용자가 강의 목록을 볼 수 있다.
사전 조건	서버와 데이터베이스가 정상적으로 연결되어있다.
사후 조건	X
가정	사용자의 인터넷 환경이 원활하다

Table 21 Use Case of Lecture Like

이름	Set Lecture Like
사용자	REGISTERED/UNREGISTERED
설명	사용자는 강의 좋아요 버튼을 누를 수 있다. 이때 시스템은 사용자의 좋아요 버튼을 데이터 베이스 상에 반영할 수 있어야한다.
정상 프로세스	사용자가 좋아요를 클릭하면 좋아요 갯수가 늘어난다.
사전 조건	서버와 데이터베이스가 정상적으로 연결되어있다.
사후 조건	데이터 베이스의 LECTURE TABLE 에 해당 강좌의 LIKE_COUNT Attribute 에 해당하는 갯수가 증가한다.
가정	사용자의 인터넷 환경이 원활하다

Table 22 Use Case of Login

이름	Login
사용자	REGISTERED/UNREGISTERED
설명	사용자는 구글 로그인 버튼을 사용해서 로그인을 할 수 있어야한다. 이때 서버는 OAuth2.0 을 바탕으로 발급되는 토큰을 사용자에게 전달하고 해당 토큰의 인증을 처리해줘야한다. 프론트엔드 서버의 경우 고객이 sessionStorage 등에 해당 토큰을 저장할 수 있어야 한다. 이후 발생하는 HTTP 통신에 Authorization 부분에는 Bearer Token 으로 서버에 해당 토큰을 보낼 수 있어야한다.
정상 프로세스	사용자가 로그인을 할 수 있다.
사전 조건	웹 서비스가 구글에 정상적으로 등록되어있다. 구글의 OAUTH2.0 인증이 정상적으로 동작한다. 서버가 정상적으로 동작한다.
사후 조건	사용자가 인증받은 토큰을 서버에게 넘겨주고 서버는 해당 토큰을 인증하고 id_token 을 추출하여 사용자에게 넘겨줄 수 있다.
가정	사용자의 인터넷 환경이 원활하다 구글 서버가 정상적으로 동작한다.

Table 23 Use Case of Sign up

이름	Sign up
사용자	UNREGISTERED
설명	처음 이용하는 사용자는 자신의 이름을 회원가입을 통해 설정할 수 있어야한다. 이때도 Google 을 통한 OAuth2.0 방식을 사용해서 토큰을 발행하게 한다. 이는 성균관대 소속만이 회원가입을 하게 하기 위함이다. 이를 위해 서버는 console.google.com 에 등록이 되어있어야하고 이때 Resource Server 에 요청하는 정보의 scope 는 이름과 이메일이다.
정상 프로세스	사용자가 회원가입을 할 수 있다.
사전 조건	웹 서비스가 구글에 정상적으로 등록되어있다. 구글의 OAUTH2.0 인증이 정상적으로 동작한다. 서버가 정상적으로 동작한다.
사후 조건	사용자가 인증받은 토큰을 서버에게 넘겨주고 서버는 해당 토큰을 인증하고 id_token 을 추출하여 사용자에게 넘겨줄 수 있고 사용자가 정한 이름을 데이터베이스에 반영할 수 있어야한다.
가정	사용자의 인터넷 환경이 원활하다 구글 서버가 정상적으로 동작한다.

Table 24 Use Case of My page

이름	View/edit user information
사용자	REGISTERED
설명	사용자는 자신의 개인 정보를 조회할 수 있어야한다. 이때 사용자에게 시스템이 제공해야하는 개인정보는 이름, 수강 강의, 질문 목록 등이 존재한다. 또한 사용자의 이름을 수정할 수 있어야한다.
정상 프로세스	사용자가 개인정보를 확인 할 수 있다.
사전 조건	서버가 데이터베이스와 원활하게 연결되어있다.
사후 조건	수정한 개인정보가 데이터베이스에 반영되어야한다.
가정	사용자의 인터넷 환경이 원활하다

Table 25 Use Case of Taking a lecture

이름	Take a lecture
사용자	REGISTERED
설명	사용자는 우리 서비스가 제공하는 강의를 들을 수 있어야한다. 강의 내용은 글과 그림으로 되어있다. 시스템은 사용자가 선택한 강의에 대한 정보를 서버로 부터 불러와서 사용자에게 보여줄 수 있어야한다.
정상 프로세스	사용자가 강의를 들을 수 있다.
사전 조건	서버가 데이터베이스와 원활하게 연결되어있다.
사후 조건	X
가정	사용자의 인터넷 환경이 원활하다

Table 26 Use Case of Practicing code

이름	Take a practice coding and Run
사용자	REGISTERED
설명	사용자는 강의에 있는 실습 코드를 작성할 수 있다. 이를 위해 서비스는 사용자가 코드를 입력하기 위한 Editor 를 제공해야하고 해당 코드를 서버에 전송하고 실행시킨뒤 사용자에게 실행 결과를 보여줄 수 있어야한다. 이때 사용자는 import 를 통한 라이브러리 불러오기 등을 할 수 있다. 또한 사용자가 잘못된 코드를 작성했을 시에 해당 코드의 오류 내용을 사용자에게 전달할 수 있어야한다. 서버는 사용자의 코드 작성 로그를 저장하여 추후에 다시 강의에 들어왔을 때 자신이 작성한 코드를 확인할 수 있게 한다.
정상 프로세스	사용자가 코드를 치고 실행 시킬 수 있다.
사전 조건	사용자는 로그인하고 있어야한다. 서버가 정상 동작한다.
사후 조건	X
가정	사용자의 인터넷 환경이 원활하다

Table 27 Use Case of Lecture Completion

이름	Lecture completion processing
사용자	REGISTERED
설명	시스템은 사용자가 올바른 코드를 작성하여 실습을 완료한 경우 사용자의 해당 강의를 수강했다는 것을 인증할 수 있어야한다. 또한 강좌의 모든 강의를 들은 경우 수료 상태에 대한 처리도 가능해야한다.
정상 프로세스	사용자가 강의를 모두 들은 경우 완료 표시가 자동적으로 처리된다.
사전 조건	사용자는 로그인하고 있어야한다. 서버가 정상 동작한다.
사후 조건	사용자의 강의 내역에 완료 표시를 해준다.
가정	사용자가 강의를 마무리 하였다. 사용자의 인터넷 환경이 원활하다.

Table 28 Use Case of Writting a question

이름	Write a question
사용자	REGISTERED
설명	사용자는 질문을 작성할 수 있어야하고 해당 질문 내용은 서버에 저장되게 된다. 이때 사용자는 질문경로를 설정할 수 있는데 해당 질문 경로는 어떤 강의로 부터 왔는지 혹은 경로를 표시하지 않을지 설정할 수 있다. 서버는 이렇게 작성한 질문을 데이터 베이스에 저장 할 수 있어야한다.
정상 프로세스	사용자가 질문을 작성할 수 있다.
사전 조건	사용자는 로그인하고 있어야한다. 서버가 정상 동작한다.
사후 조건	사용자가 작성한 댓글이 데이터 베이스에 정상적으로 반영되어있다.

가정	사용자의 인터넷 환경이 원활하다.
----	--------------------

Table 29 Use Case of Question board

이름	View question
사용자	REGISTERED
설명	사용자는 특정 질문에 대해서 볼 수 있어야한다. 이때 제목, 작성날짜, 질문경로 등이 사용자에게 보여줘야하고 해당 질문에 작성된 댓글들 또한 볼 수 있어야한다. 댓글의 경우 작성자와 작성 내용을 볼 수 있다. 이를 위해 시스템은 특정 질문에 대한 내용을 불러올 수 있어야하며 해당 질문에 연관된 댓글들 또한 작성자와 내용등을 모두 데이터베이스로 부터 불러올 수 있어야한다.
정상 프로세스	질문 상세에 들어갔을 때 사용자가 질문에 상세한 내용을 볼 수 있다.
사전 조건	서버가 정상 동작한다.
사후 조건	X
가정	사용자의 인터넷 환경이 원활하다.

Table 30 Use Case of Comment

이름	Write comment
사용자	REGISTERED
설명	사용자는 질문 글에 댓글을 작성할 수 있어야한다. 해당 질문을 쓰다가 취소하는 경우 댓글 내용이 지워지게 되고 등록을 하게 되면 데이터베이스 상에 사용자의 댓글이 등록되게 된다. 시스템은 사용자가 댓글을 작성하거나 지울 수 있게 해야하고 작성된 댓글의 경우 데이터베이스 상에 등록할 수 있어야한다.
정상 프로세스	질문 상세에 들어갔을 때 댓글을 남길 수 있다.
사전 조건	서버가 정상 동작한다.
사후 조건	데이터 베이스 COMMENT Table 에 정상적으로 댓글 정보가 저장된다.
가정	로그인이 되어있다. 사용자의 인터넷 환경이 원활하다.

4.1.2. USE CASE DIAGRAM

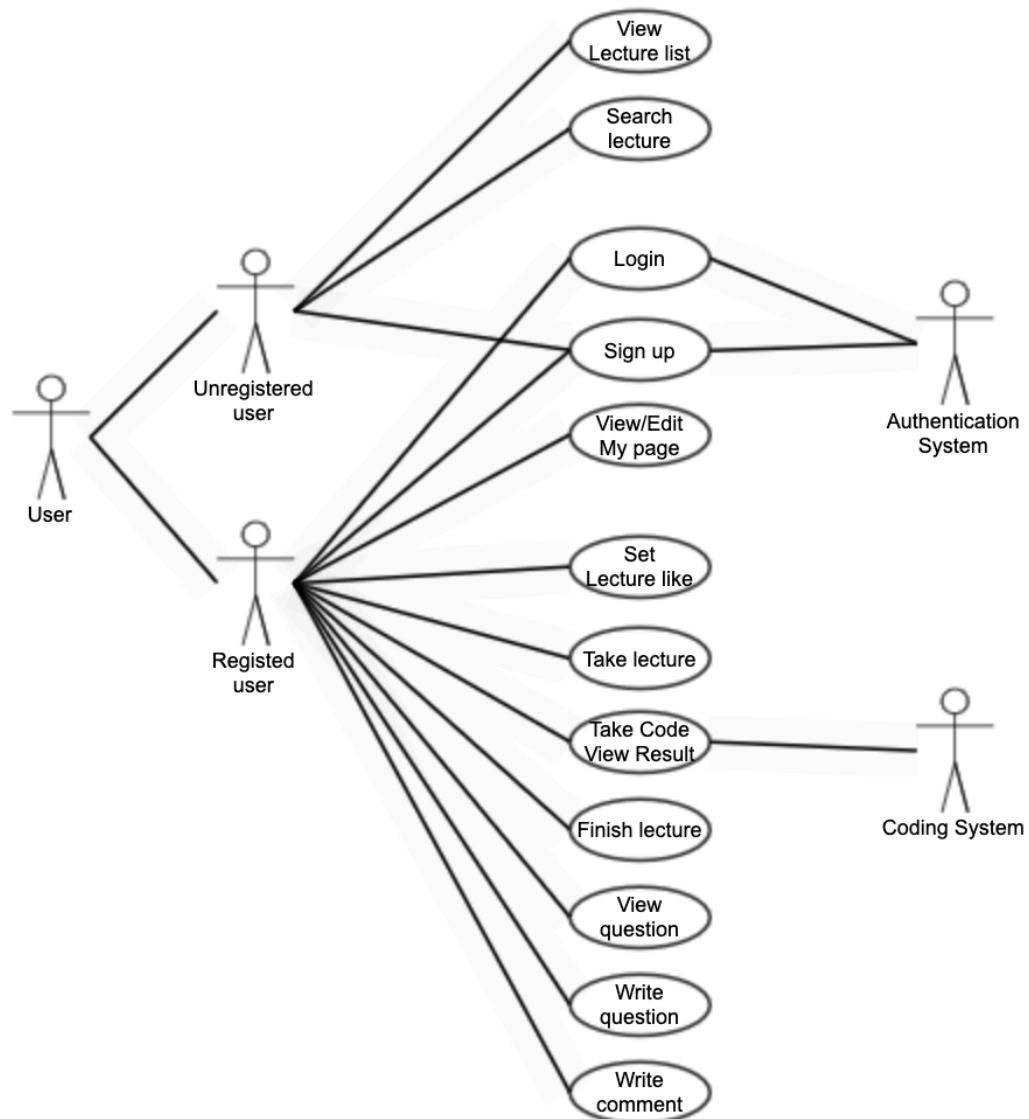


Figure 10 Use Case Diagram

4.1.3. DATA FLOW DIAGRAM

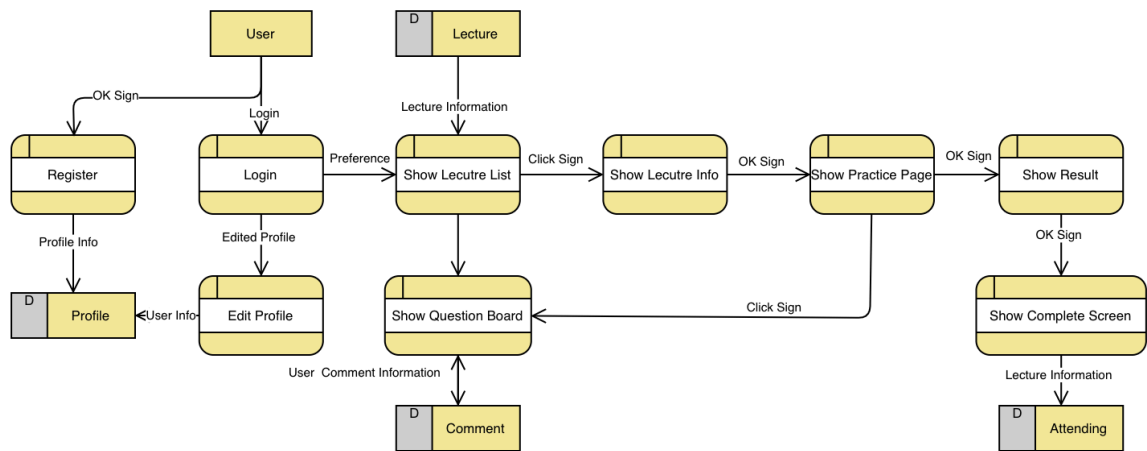


Figure 11 Data Flow Diagram

4.2. NON FUNCTIONAL REQUIREMENTS

시스템이 제공하는 서비스나 시스템 속성 및 제약 조건을 정의한다.

4.2.1. PRODUCT REQUIREMENT

특정한 방식으로 작동해야 하는 것을 명시하는 요구 사항이다.

4.2.1.1. USABILITY REQUIREMENT

시스템은 전문적이지 않은 사용자가 사용하기 쉬워야 한다. 사용자의 오류를 최소화하도록 구성되어야 한다. 전문적인 용어의 사용을 최소화해야 하고, 이해하기 쉽게 설명해야 한다. 각 사용자는 별도의 매뉴얼 없이 모든 시스템 기능을 직관적으로 사용할 수 있어야 한다. 로그인 페이지와 code-editor 페이지에서 사용자의 오류율은 10%를 초과하지 않아야 한다.

4.2.1.2. PERFORMANCE REQUIREMENT

시간당 5,000 명의 사용자를 지원하는 해당 시스템은 LTE 연결을 통해 텍스트 및 이미지 렌더링을 포함하여 Chrome 데스크톱 브라우저에서 5 초 이하의 응답 시간을 제공해야 한다.

4.2.1.3. SPACE REQUIREMENT

시스템은 지정된 시간에 원하는 서비스를 제공해야 한다. 시스템은 오류를 최소화하면서 서비스를 제공해야 한다. 결과는 사용자가 예상한 대로 신뢰할 수 있어야 한다.

4.2.1.4. SECURITY REQUIREMENT

교육 시스템을 사용하려면 사용자의 계정을 인증하고 등록해야 한다. 권한이 없는 사용자가 시스템 관리자로 액세스할 수 없고 시스템을 사용할 수 없도록 해야 한다. 또한 권한이 없는 사용자는 시스템 사용자로서 사용자의 개인 정보, ID 및 암호와 같은 기밀 정보에 액세스할 수 없다. 데이터베이스는 HIPPA 요구사항을 충족해야 한다.

4.2.2. ORGANIZATIONAL REQUIREMENT

사용하는 프로세스 표준, 구현 요구사항 등과 같이 조직의 정책 및 절차의 결과로 발생하는 요구사항이다.

4.2.2.1. ENVIRONMENTAL REQUIREMENT

시스템 사용자는 자신의 이메일을 사용하여 자신의 신원을 확인해야 한다. 사용자인 교육 대상자는 해당 시스템을 데스크탑 또는 노트북을 이용하여 사용하기를 권장한다.

4.2.2.2. OPERATIONAL REQUIREMENT

호환성이 높은 시스템은 일반적으로 다른 응용 프로그램이 장치에서 실행될 때 잘 작동한다. Windows 7, Windows 8, Windows 8.1, Windows 10 이상이고, Intel Pentium 4 프로세서 또는 SSE3 를 지원하는 그 이상 버전으로 Chrome 브라우저를 이용한다.

4.2.3. EXTERNAL REQUIREMENT

시스템 및 개발 프로세스의 외부 요인에서 파생된 모든 요구 사항을 다룹니다.

4.2.3.1. REGULATORY REQUIREMENTS

본 시스템은 사용자의 로그인 정보를 구글의 OAuth 2.0 을 이용하여 진행한다. OAuth 의 경우 일일 토큰 부여 한도는 10,000 회 이다. 이 이상 진행될 경우 구글에 상향 요청을 보내야한다.

4.2.3.2. ETHICAL REQUIREMENTS

웹 크롤링의 규약인 robot.txt 에서 허락된 내용 안에서의 크롤링을 진행한다.

4.2.3.3. SAFETY/SECURITY REQUIREMENTS

개인정보보호법에 따라 사용자의 개인정보를 외부/내부적으로 안전하게 저장하고, 보호해야한다.

4.3. ORGANIZING SYSTEM FLOW

4.3.1. CONTEXT MODEL

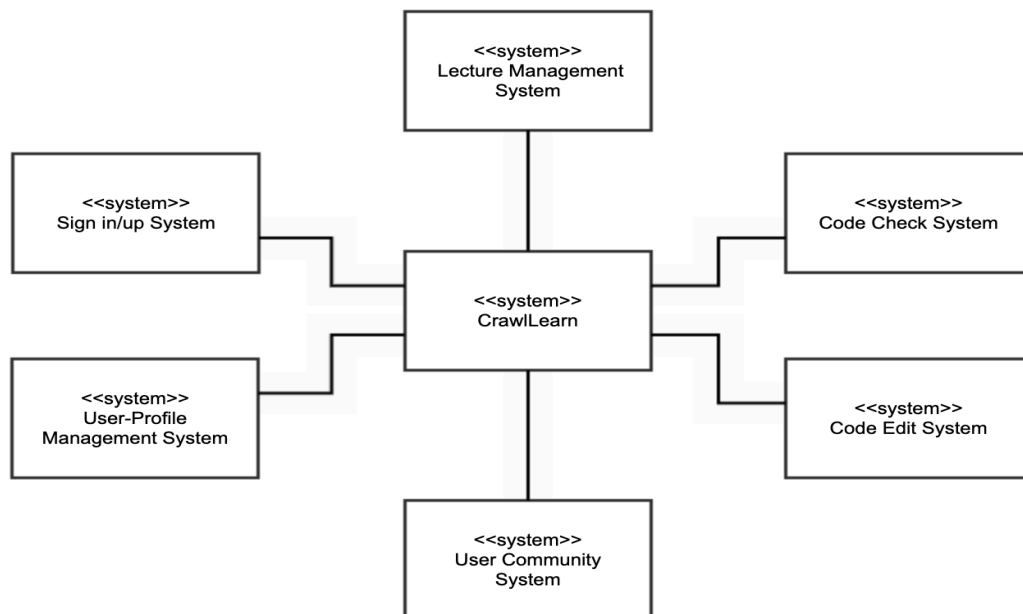


Figure 12 Context Model

4.3.2. PROCESS MODEL

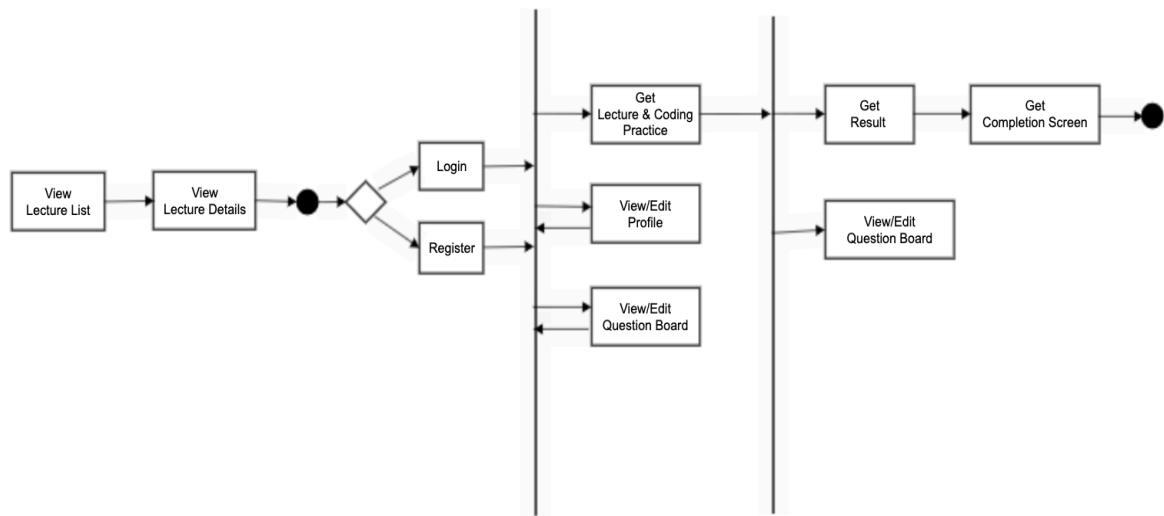


Figure 13 Process Model

4.3.3. INTERACTION MODEL

4.1.2. Use Case Diagram 참고

4.3.4. BEHAVIOR MODEL

4.3.4.1. DATA FLOW DIAGRAM

4.1.3. data flow diagram 참고

4.3.4.2. SEQUENCE DIAGRAM

메인 시스템 설계 상태

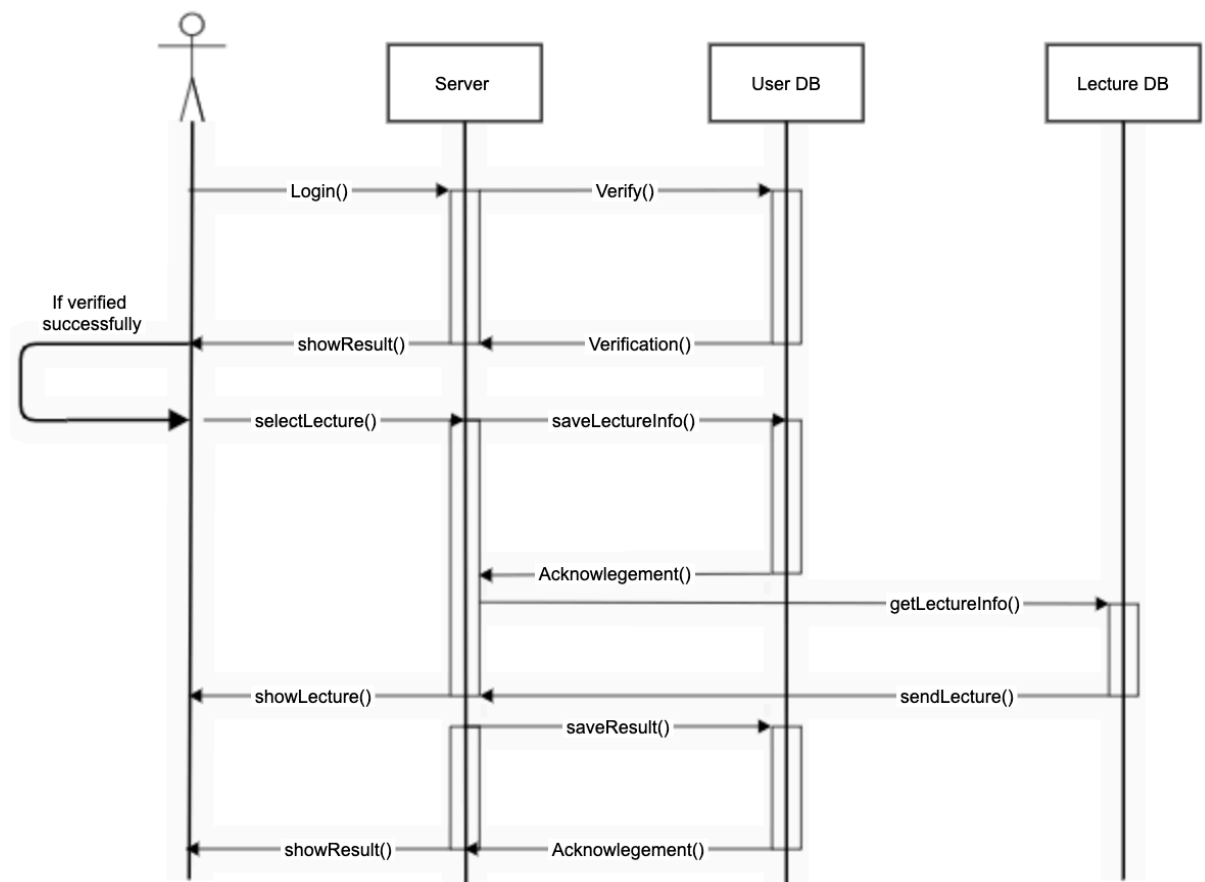


Figure 14 Sequence Diagram

4.4. SYSTEM ARCHITECTURE

이 섹션에서는 예상되는 시스템 아키텍처에 대한 높은 수준의 개요를 제시하고 시스템 모듈 전반의 기능 분포를 보여준다. MVC 패턴을 아키텍처의 그래픽 모델로 사용한다.

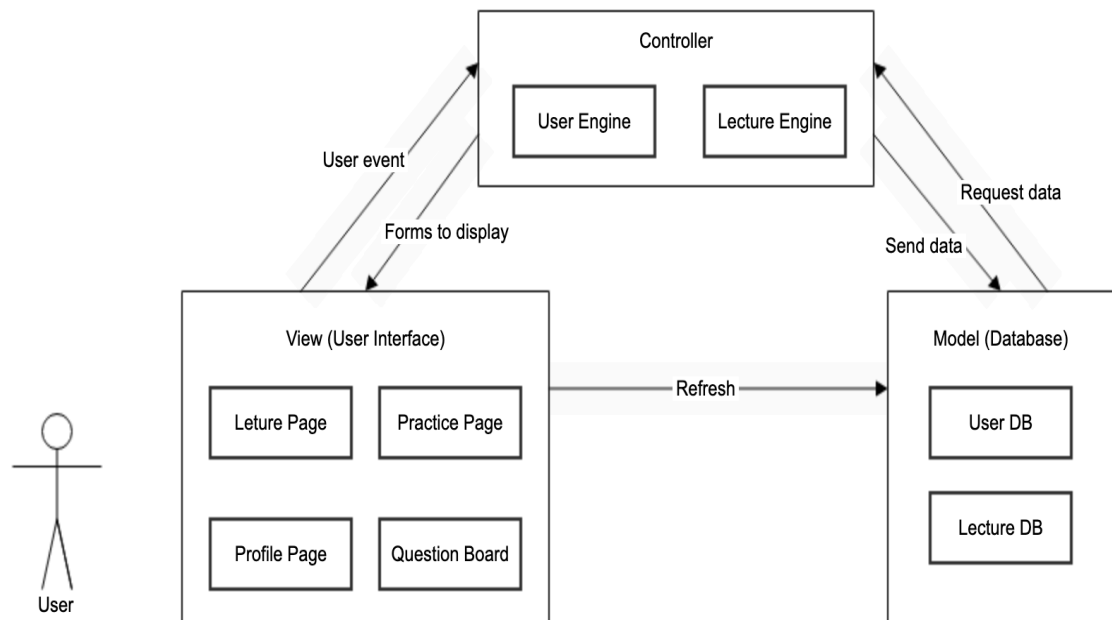


Figure 15 System Architecture

5. SUPPORTING INFROMATION

5.1. SOFTWARE REQUIREMENT SPECIFICATION

이 소프트웨어 요구사항 명세서는 IEEE Recommended Practice for Software Requirements Specifications 문서를 기준으로 작성되었다.

5.2. DOCUMENT HISTORY

Table 31 Document history

Date	version	Description	Writer
2022/04/07	0.1	2	이정우
2022/04/07	0.1	3	배지현
2022/04/07	0.1	4.1	김진환
2022/04/07	0.1	4.2.1, 4.2.2	정미서
2022/04/07	0.1	4.2.3, 4.3	이민영
2022/04/08	0.1	1.1	이정우
2022/04/08	0.1	1.2, 1.3	김진환
2022/04/08	0.1	1.4, 1.5	정미서
2022/04/08	0.1	5	배지현
2022/04/08	0.2	2	이정우
2022/04/08	0.2	3	배지현
2022/04/08	0.2	4.1	김진환
2022/04/08	0.2	4.2.1, 4.2.2	정미서
2022/04/08	0.2	4.2.3, 4.3, 4.4	이민영
2022/04/09	1.0	통합	이민영