

Attendance Web Page SRS

Table Of Contents

Table Of Contents	1
0. Who make this document	2
1. Introduction	2
1.1 Purpose	2
1.2 Scope	2
1.3 Definitions, Acronyms and Abbreviations	3
1.4 Reference	4
1.5 Overview	4
2. Overall description	5
2.1 Product perspective	5
2.2 Product functions	6
2.3 User characteristics	6
2.4 Constraints	6
2.5 Assumptions and dependencies	7
2.6 Apportioning of requirements	7
2.7 Class Diagram	7
3. Specific requirements	8
3.1 External interface requirements	8
3.1.1 User interfaces	8
3.1.2 Hardware interfaces	9
3.1.3 Software interfaces	9
3.1.4 Communications interfaces	10
3.2 Functional requirements	10
3.2.1 Use Case Diagram	10
3.2.2. Fully Dressed Use Cases	10
3.2.2.1 Fetch view SAGI	10
3.2.2.2 Print excel	12
3.2.2.3 Create lecture	13
3.2.2.4 Send mail	14
3.2.2.5 View my info	16
3.2.2.6 Sign up	17
3.2.2.7 Log in	18
3.2.2.8 Select lecture	20
3.2.2.9 Fetch my SAGI	21

3.2.2.10 Accept lecture	23
3.2.2.11 Register fingerprint	24
3.2.2.12 Attend	25
3.2.2.13 View information	26
3.2.2.14 Create information	27
3.3 Performance requirements	28
3.4 Design constraints	28
3.4.1 Software Constraints	28
3.4.2 Hardware Constraints	28
3.5 Software system attributes	28
3.5.1 Security	28
3.5.2 Reliability	28
3.5.3 Availability	28

0. Who make this document

1. All Parts: 남현실, 서민규, 신수용, 이재승, 한진수

1. Introduction

1.1 Purpose

출석 웹페이지는 학생들의 출석관리를 자동으로 교수님에게 리포트 해주는 출결 정보 웹 어플리케이션이다.

교수님은 수업시간에 출석을 부르지 않고 바로 수업을 할 수 있다. 이 웹 어플리케이션은 교수님이 신경쓰지 않아도 효율적으로 출결을 관리할 수 있다.

(The attendance web page is an attendance information web application that automatically reports attendance management of students to professor.

The professor can do the lesson without calling attendance in class. This web application can manage attendance efficiently without worrying about the professor.)

1.2 Scope

Basic User - Professor, Teaching Assistant, Student, Admin

All users using AWP must join. (AWP를 이용하는 유저는 모두 가입을 해야한다.)

Users must register fingerprints. (가입한 유저들은 지문 등록을 해야한다.)

The attendance information of the user is reported by mail. (유저의 출결 정보는 메일로 리포트 된다.)

Professors can register for classes.(Professor은 강의 등록을 할 수 있다.)

The student must be approved by the professor to participate in the lecture. (Student는 강의 참여를 Professor에게 승인을 받아야한다.)

Chrome notification - for students whose attendance is at risk. (크롬 noti - 출결 상황이 위험한 학생들)

Member management(attendance/absence/perception) visualization (멤버 관리 - 출석/결석/지각률 시각화)

1.3 Definitions, Acronyms and Abbreviations

AWP

- Attendance Web Page.
- Web application that automatically manages attendance of professor (교수님의 출석 관리를 자동으로 해주는 웹 어플리케이션)

User

- Students
 - Teach the people who participate in the lectures registered by Professor. (Professor가 등록한 강의에 참여하는 사람들을 가르친다.)
- Professor
 - Register the lecture to teach and approve the student's participation in the lecture. (가르칠 강의를 등록하고 Student의 강의 참여를 승인해준다.)
- TA
 - Teaching Assistant. A role to help you. You can approve student participation in the lecture.

Admin

- Administrator can grant permission to professor.

HTTP

- Hypertext Transfer Protocol. It's a service protocol.

HTTPS

- HTTPS is a protocol for secure communication over a computer network which is widely used on the Internet.

DBMS

- A database is an organized collection of data. It is the collection of schemas, tables, queries, reports, views, and other objects.

Daily report

- It contains attendance information of students who listened to the lectures today.

SAGI

- Student Attendance General Information.
- Each lecture contains student attendance, absence, and perception information, which can be visualized, and information can be output.

Flask

- A lightweight Python web framework

1.4 Reference

- PEP8: <https://www.python.org/dev/peps/pep-0008/>
- Flask: <http://flask.pocoo.org/>
- Wikipedia: www.wikipedia.com
- IEEE-Recommended Practice for SRS
- PEP249: <https://www.python.org/dev/peps/pep-0249/>
- Draw.io
- SRS for Virtual Medical Home

1.5 Overview

Drawbacks

- The new professor can not gain authority
- It have to have time for regist fingerprint.
- It cannot be used in case can't connect internet.
- It will have some case can't use fingerprint.

Proposed System

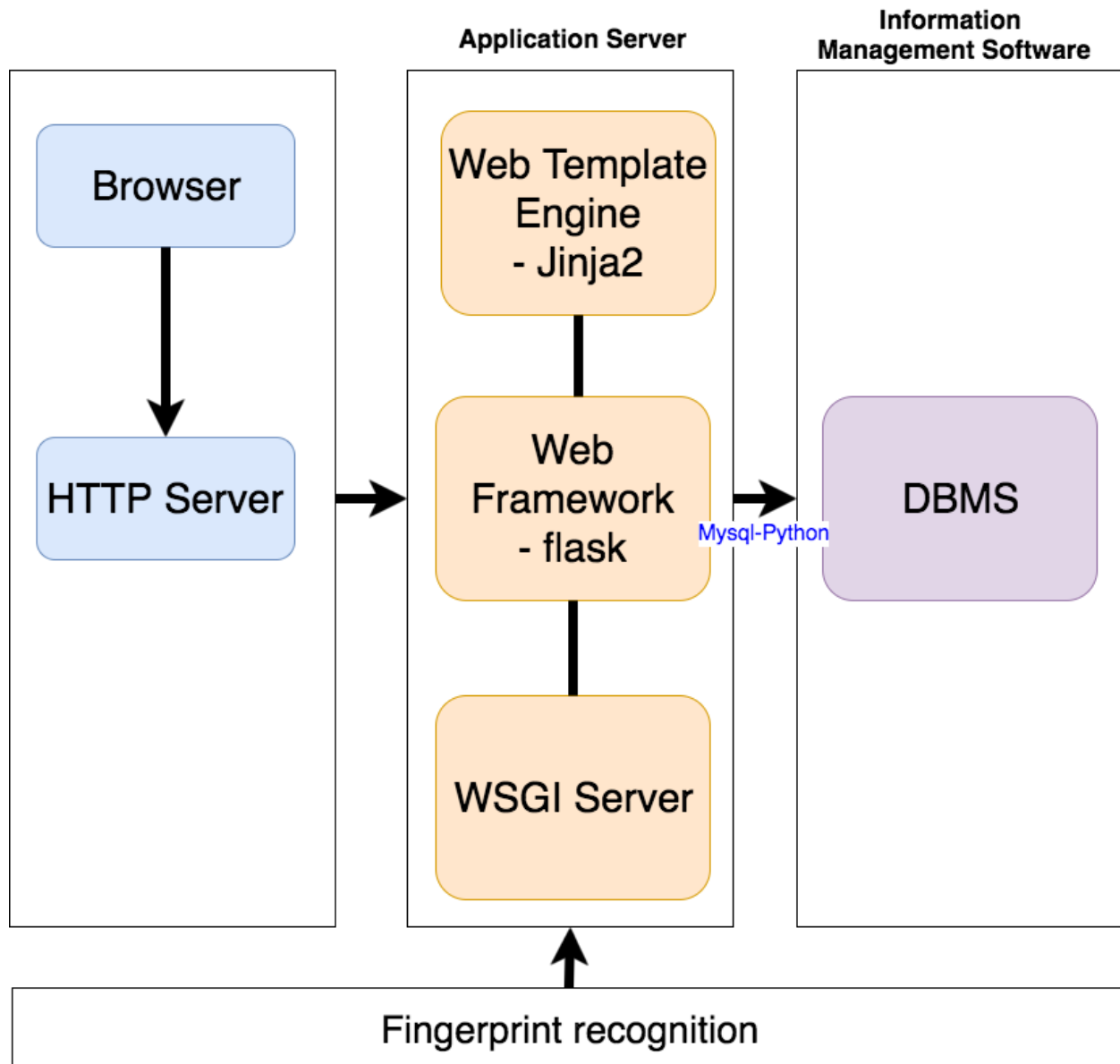
- Students, professors, and assistants are registered. (학생과, 교수와, 조교를 등록한다.)
- Admin helps to register the professor. (Admin은 교수를 등록하는데 도와준다.)
- Notification or daily report is sent. (노티피케이션이나 데일리 리포트를 보내준다.)

Our Plan

- User joins up (User가 회원가입을 한다.)
- The user registers the fingerprint.(User가 지문등록을 한다.)
- The professor creates the lecture.(교수가 강의를 생성한다.)
- The student applies for the lecture. (학생이 강의를 신청한다.)
- The professor approves the student's participation in the lecture. (교수가 학생의 강의 참여를 승인한다.)
- The student is present through the fingerprint. (학생이 지문을 통해 출석한다.)
- Email the daily report to the professor. (데일리 리포트를 교수님께 메일로 보내준다.)
- We send notifications or reports to students who are in bad condition. (출결 상황이 안좋은 학생에게 노티해준다.)
- The professor can visualize SAGI.(교수님이 SAGI를 시각화 하여 볼 수 있다.)
- SAGI can be printed to excel.

2. Overall description

2.1 Product perspective



2.2 Product functions

- Attend with fingerprints.(지문으로 출석을 한다.)
- Email the daily report to the professor.(데일리 리포트를 교수님께 메일로 보내준다.)
- If you are in bad condition, please let me know by mail.(출결 상황이 안좋은 학생에게 메일로 noti해준다.)
- The professor can visualize SAGI.(교수님이 SAGI를 시각화 하여 볼 수 있다.)
- You can output SAGI to Excel.(SAGI를 엑셀로 출력할 수 있다.)

2.3 User characteristics

Student

- You can attend lectures and attend. They can also see their attendance information. Students who are in bad condition will be notified by email.(강의를 참여하고 출석 할 수 있습니다. 그들은 또한 그들의 출결 정보를 볼 수 있습니다. 출결 현황이 안좋은 학생은 자신의 이메일로 알림을 받습니다.)

Professor

- You can register for classes and approve your students' participation in the classes. You can output comprehensive information about the attendance of students listening to the lecture, and visualize it at a glance. TA can be registered.(강의를 등록 할 수 있고, 학생들의 강의 참여를 승인해 줄 수 있습니다. 강의를 듣는 학생들의 출결 종합 정보를 엑셀로 출력 할 수 있고, 한눈에 시각화하여 볼 수 있습니다. TA를 등록할 수 있습니다.)

TA

- You can approve student participation in the lecture.(학생들의 강의 참여를 승인해 줄 수 있습니다.)

Admin

- You can grant authority to the professor.(교수님에게 권한을 부여할 수 있습니다.)

2.4 Constraints

- The GUI is only available in Korean.(GUI는 한글로만 제공 됩니다.)
- Login and password is used for the identification of users.
- Limited to HTTP/HTTPS.
- Only registered students, professor and TA will be authorized to use the services.
- Programming Language is Python
- This system is working for single server.

- Constraints on fingerprint recognition
 - TBD. Describe this when developing hardware part
- I can not use it when the internet is not available.(인터넷이 안될 때는 사용할 수 없다.)

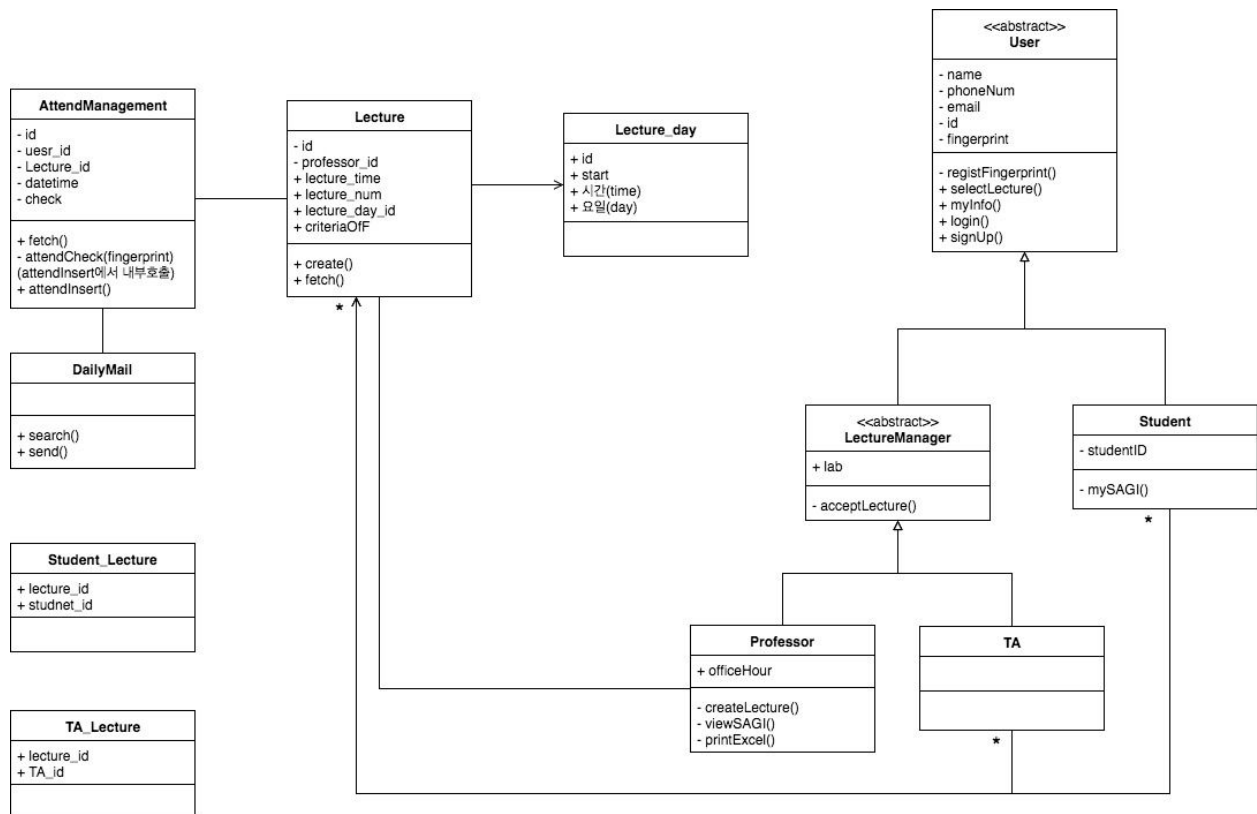
2.5 Assumptions and dependencies

- All students listening to the lecture must register their fingerprints.(강의를 듣는 모든 학생은 지문을 등록해야한다.)
- Every student listening to a lecture should have a fingerprint.(강의를 듣는 모든 학생은 지문이 있어야 한다.)
- All students who take classes should join.(강의를 듣는 모든 학생은 가입을 해야한다.)
- The Internet is always good.(인터넷은 항상 잘된다.)
- The user has all of the email.(유저는 이메일을 다 가지고 있다.)
- The user should be able to understand Korean.(유저는 한국어를 이해 할 수 있어야 한다.)

2.6 Apportioning of requirements

- Chrome Notification

2.7 Class Diagram



3. Specific requirements

3.1 External interface requirements

3.1.1 User interfaces

- Login

Sign In

User Name:

johndoe

Password:

SIGN IN

[Forgot Password?](#)

New User

SIGN UP

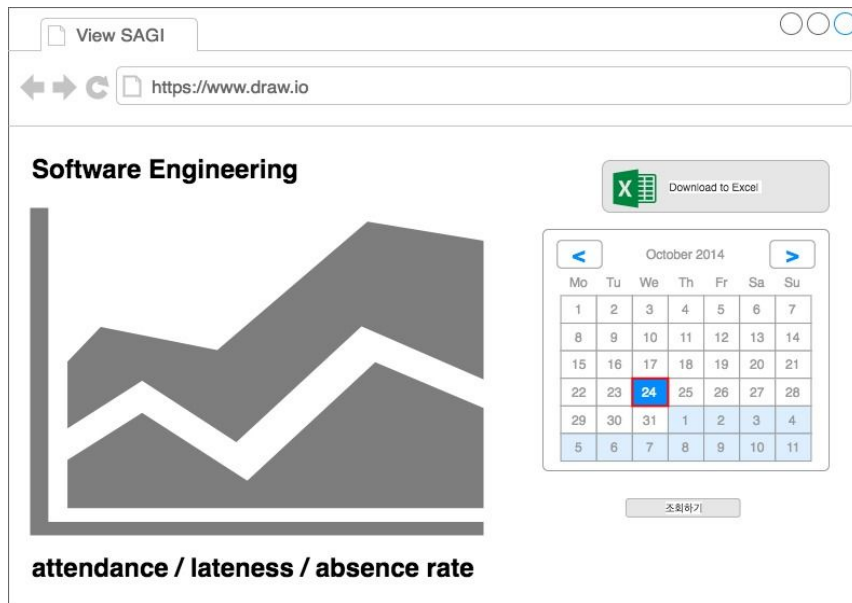
- View My Info

안녕하세요 000님!

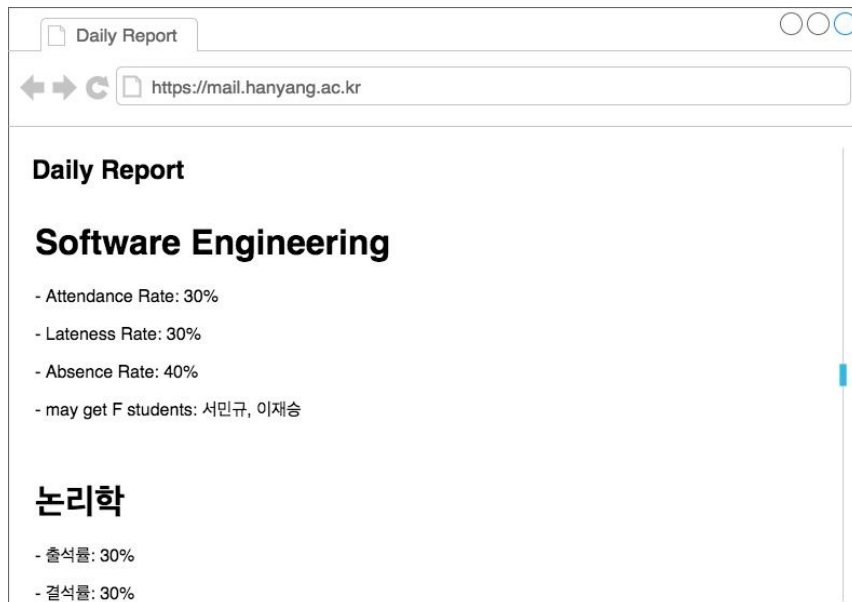
- ID: professor_lee
- E-Mail: abc@hanyang.ac.kr
- PhoneNum: 010-0000-0000

내 강의 목록
소프트웨어 공학
데이터베이스
AI

- View Lecture



- Daily Report



3.1.2 Hardware interfaces

TBD: It is now AWP development plan, and additional technology will be developed when developing fingerprint recognition.(지금은 AWP개발 계획이라서 지문 인식 개발할 때 추가 기술 예정.)

3.1.3 Software interfaces

- User on Internet

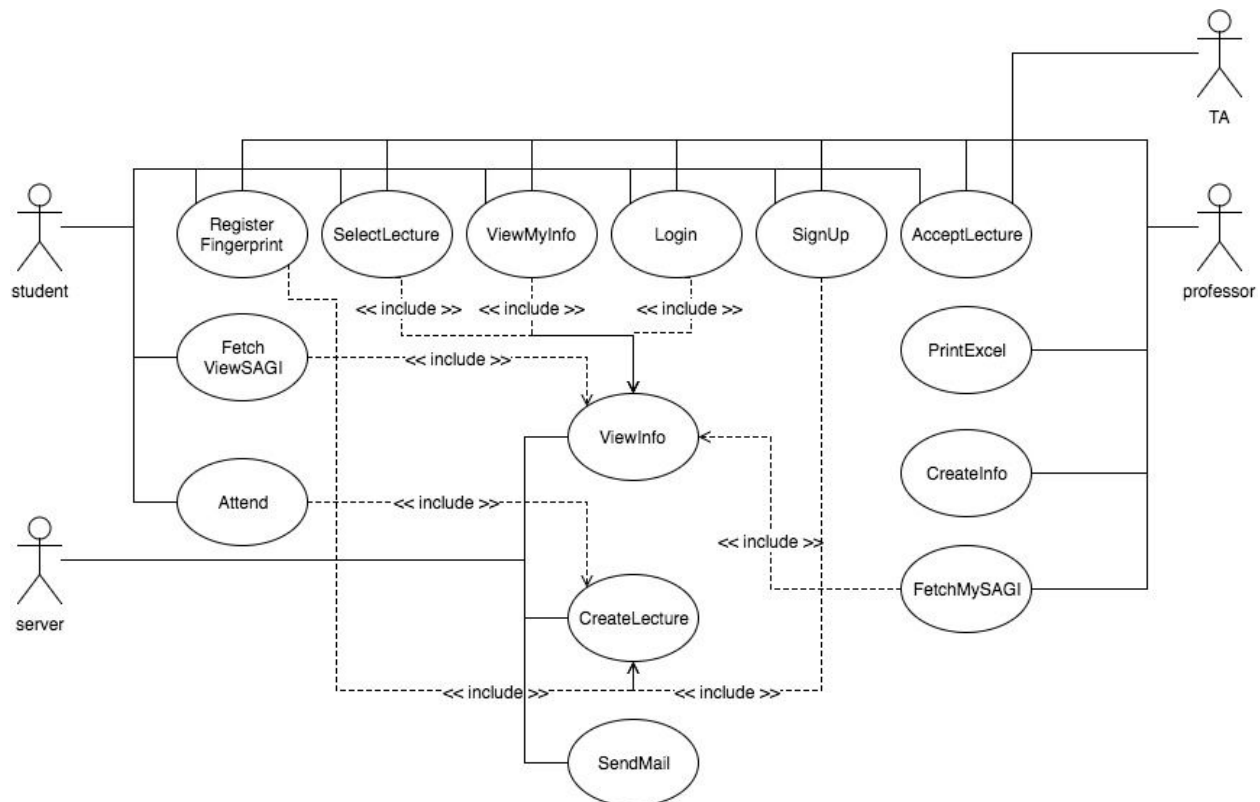
- Web Browser(without IE less than version 9), Operating System(any)
- Web Server
 - Flask, Operating System(linux)
- Database Server
 - Mysql or MariaDB, Operating System(linux)
- Development End
 - HTML, AJAX, Jinja2, Operating System(any)

3.1.4 Communications interfaces

- User on Internet will be using HTTP/HTTPS protocol.

3.2 Functional requirements

3.2.1 Use Case Diagram



3.2.2. Fully Dressed Use Cases

3.2.2.1 Fetch view SAGI

Use Case Name	Fetch view SAGI
---------------	-----------------

	(학생 출결 정보 조회)
Goal in Context	Professor can see the attendance information of the student. (학생의 출결 정보를 볼 수 있다.)
Scope	Professor
Level	!
Primary Actor	Professor
Secondary Actor	Server
Stakeholders and Interests	Member Management - Attendance / Absence / Perception Rate Visualization (멤버 관리 - 출석/결석/지각률 시각화)
Preconditions	Professor must log in. (교수님은 로그인 해야한다.)
Post-condition	Show visualized data showing attendance, absences and perceptions. (출석과 결석과 지각률이 표현된 시각화된 자료를 보여준다.)
Trigger	Professor clicks on student attendance information. (교수님이 학생 출결 정보를 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Professor: Press the Check Status button. (출결 상황 확인을 누른다.) 2. Server: Bring the attendance of the subject. (해당 과목의 출결을 가져온다.) 3. Web browser : Visualize the attendance situation. (출결 상황을 시각화한다.)
Extensions(Alternative flows)	3a. When there is nothing to attend: Give no results. (출결이 아무 것도 없을 때: 결과 없음을 띄워준다.)
Special requirements	<ol style="list-style-type: none"> 1. Express in Korean. (한국어로 보여준다.)
Technology and data variations list	2a. Server requests data from DB (Server는 DB에 데이터를 요청한다.)

	3a. Attendance situations visualize attendance, absences and perceptions.(출결상황은 출석과 결석과 지각률을 시각화한다.)
Frequency of occurrence	When professor want to see(교수님이 보고싶을 때)
Miscellaneous	1. When the network environment is bad. (통신이 안될 경우)

3.2.2.2 Print excel

Use Case Name	Print Excel (종합정보 Excel 출력)
Goal in Context	The professor can extract the attendance information of the lecture to Excel file. (교수가 해당 강의의 학생 출결정보를 Excel 파일로 추출할 수 있다.)
Scope	Professor
Level	!
Primary Actor	Professor
Secondary Actor	Server
Stakeholders and Interests	Professor: I would like to save attendance information as a separate file. (교수 : 출결정보를 별도 파일로 저장하고 싶다.)
Preconditions	The professor must log in. (교수님은 로그인을 해야 한다.)
Post-condition	Extract the attendance information to Excel. (출결 정보를 Excel로 추출한다.)
Trigger	The professor presses the extract button. (교수가 추출 버튼을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Professor : Press the Extract button.(추출 버튼을 누른다.) 2. Server : Get attendance information.(출결 정보를 가져온다.) 3. Server : Create the attendance

	information as an Excel file.(출결 정보를 Excel 파일로 생성한다.)
Extensions(Alternative flows)	2.a. Poor data transfer: error popup (데이터 전송이 안됐다 : 에러 팝업)
Special requirements	1. Express in Korean. (한국어로 보여준다.)
Technology and data variations list	1.a. Web server to the DB server to request the attendance status of the subject. (Web서버에서 DB서버로 해당 과목의 출결 상황을 요청한다.) 2.a. The Web server receives the request result from the DB server and sends the response.(Web서버는 DB서버의 요청 결과를 받아서 response를 보낸다.) 3.a. The web server generates the received information as an Excel file (Web서버는 전달받은 정보를 Excel파일로 생성한다.)
Frequency of occurrence	Every time professor needer an Excel file. (교수님이 Excel 파일을 필요로 할 때마다.)
Miscellaneous	1. When the network environment is bad. (통신이 안될 경우)

3.2.2.3 Create lecture

Use Case Name	Create Lecture (강의 생성)
Goal in Context	Professors can create their lecture (교수가 자신의 강의를 등록할 수 있다.)
Scope	Professor
Level	!
Primary Actor	Professor
Secondary Actor	Server
Stakeholders and Interests	Professor: I want to be able to register my teaching lectures on the lecture list. (교수 : 내가 가르치는 강의를 강의목록에 등록할 수 있었으면 한다.)

Preconditions	Professor has to login (교수는 로그인 해야한다.) When a professor wants to enroll his lecture on a lecture list (교수가 자신의 강의를 강의목록에 등록하고 싶을때)
Post-condition	The professor presses the lecture creation completion button. (교수가 강의 생성 완료 버튼을 누른다.)
Trigger	The professor clicks the button to finish creating the lecture. (강의 생성 버튼을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Professor: Click Create Lecture (강의 생성을 누른다.) 2. Professor: Save the lecture. (강의를 저장한다.) 3. Server: Add the generated lectures to the lecture list. (생성된 강의를 강의목록에 추가한다.)
Extensions(Alternative flows)	3a. Poor data transfer: error popup (Data전송이 안됐다. : Error팝업)
Special requirements	<ol style="list-style-type: none"> 1. I register a lecture in Korean. (한국어로 강의를 등록한다.)
Technology and data variations list	<ol style="list-style-type: none"> 1. Invoke a new class registration service on the Web. (Web에서 새로운 강의등록 서비스를 호출한다.) 2. The web server sends the lecture information generated by the DB server. (Web서버에서 DB서버로 생성된 강의 정보를 보낸다.)
Frequency of occurrence	When the new semester begins (새 학기가 시작될 때)
Miscellaneous	<ol style="list-style-type: none"> 1. When the network environment is bad. (통신이 안될 경우)

3.2.2.4 Send mail

Use Case Name	SendMail
---------------	----------

	(일일 출결 메일 서비스)
Goal in Context	The professor receives the attendance of the student every day (매일 매일 학생의 출결을 교수가 받아볼 수 있다.)
Scope	Students can check the attendance status daily. (학생들의 출결상황을 매일 확인가능한 것.)
Level	!
Primary Actor	Server
Secondary Actor	Professor
Stakeholders and Interests	Professor: You can get information about students' attendance. (교수 : 학생들의 출결에 대한정보를 정보를 받아볼 수 있다.)
Precondition	You must be connected to the service. (해당 서비스에 접속해 있어야 한다.)
Post-condition	The professor can confirm the attendance status of students on the day by e-mail in a certain time. (교수가 일정 시간에 그날 학생들의 출결상황을 메일로 확인할 수 있다.)
Trigger	It should be a certain time to send mail. (메일을 보낼 특정한 시간이 되어야 한다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Server: View today's student attendance status of the subject. (해당 과목의 오늘의 학생 출결현황을 조회한다.) 2. Server: It sends information to the professor's e-mail. (해당 교수의 메일로 정보를 전송해준다.)
Extensions(Alternative flows)	1.a. Poor data transfer: error popup (Data전송이 안됐다. : Error팝업)
Special requirements	<ol style="list-style-type: none"> 1. Express in Korea. (한국어로 보여준다.)
Technology and data variations list	<ol style="list-style-type: none"> 1. Obtain today's student attendance

	<p>information from DB server (DB서버에서 오늘의 해당 과목 학생 출결 정보를 가져온다.)</p> <p>2. The information will be collected and sent to the professor of the subject in a certain time by mail. (그 정보를 모아 일정 시간이 되면 해당 과목의 교수님께 출결 정보를 메일로 발송한다.)</p>
Frequency of occurrence	When the time comes (해당 시간이 되었을 때)
Miscellaneous	<p>1. You entered the wrong email address. (메일 주소를 잘못 입력한 경우)</p>

3.2.2.5 View my info

Use Case Name	View My Info(내 정보 조회)
Goal in Context	Show the information of the subscribed users. (가입한 유저의 정보를 보여준다.)
Scope	Professor, Student, TA
Level	!
Primary Actor	Professor, Student, TA
Secondary Actor	Server
Stakeholders and Interests	<p>Show my basic information, not attendance information. (출결 정보가 아닌 내 기본 정보를 보여준다.)</p> <p>Change Password (비밀번호 변경)</p>
Preconditions	The user must log in.(유저가 로그인 해야한다.)
Post-condition	<p>I need to change my ID and password If you are a student, the number of lectures you Professors & assistants show the number of lectures. (내 정보에 아이디와 비밀번호 변경과 학생이면 듣고있는 강의의 갯수가 교수&조교면 강의를 한 갯수를 보여준다.)</p>

Trigger	Click the View My Info button. (내 정보 조회 버튼을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. User: Click the View My Info button. (내 정보 조회 버튼을 누른다.) 2. Server: Request data from DB Server. (DB Server로 데이터를 요청한다.) 3. Web Browser: Display user information(유저의 정보를 보여준다.)
Extensions(Alternative flows)	3a. When there is no lecture, it shows that there is no lecture. (강의가 없을때는 강의가 없다고 보여준다.)
Special requirements	1. Express in Korea. (한국어로 보여준다.)
Technology and data variations list	<p>2a. The query is sent from the Web server to the database server. (웹서버에서 디비서버로 쿼리를 날린다.)</p> <p>3a. The user's information consists of a password change, an ID, and a list of lectures (유저의 정보는 비밀번호 변경, 아이디, 강의 목록으로 구성된다.)</p>
Frequency of occurrence	<p>Every time you press the My Info button. (내 정보 버튼을 누를때마다.)</p> <p>Every time a user wants to view. (유저가 조회를 하고 싶을 때마다.)</p>
Miscellaneous	1. When the network environment is bad.(통신이 안될 경우)

3.2.2.6 Sign up

Use Case Name	Sign up(회원가입)
Goal in Context	A user can register as a member of a web page(사용자가 웹페이지의 회원으로 등록할 수 있다.)
Scope	User
Level	!
Primary Actor	Student, Professor, TA
Secondary Actor	Server

Stakeholders and Interests	Professor : I want to manage the students who take lectures. (강의를 들을 학생들을 관리하고 싶다.)
Preconditions	Access attendance inquiry website. (출석 조회 웹사이트에 접속한다.)
Post-condition	Registered as a member (회원으로 등록된다.)
Trigger	Click the Subscribe button. (회원가입 버튼을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. User : On the Membership page, fill out your personal information.(회원가입 페이지에서 개인 정보를 기입한다.) 2. Server : Store information in DB (정보를 DB에 저장한다.)
Extensions(Alternative flows)	2.a. Poor data transfer: error popup (데이터 전송이 안됐다 : 에러 팝업)
Special requirements	1. Express in Korea. (한국어로 보여준다.)
Technology and data variations list	<ol style="list-style-type: none"> 1.a. Transfers the content written to the Web server. (Web서버로 기입된 내용을 전송한다.) 2.a. The Web server stores the content and responds to the results. (Web서버는 내용을 저장하고 결과를 response 한다.)
Frequency of occurrence	Every time a user joins a membership. (유저가 회원가입을 할 때마다.)
Miscellaneous	1. When the network environment is bad. (통신이 안될 경우)

3.2.2.7 Log in

Use Case Name	Log in(로그인)
Goal in Context	The user accesses the website through login. (사용자가 웹사이트에 로그인을 통해 접속한다.)

Scope	Professor, Student, TA
Level	!
Primary Actor	Student, Professor, TA
Secondary Actor	Server
Stakeholders and Interests	Professor : I want to use the site according to my information. (나의 정보에 맞게 사이트를 이용하고 싶다.)(My information inquiry, my classroom, my SAGI, etc.)
Preconditions	Go to the login page and enter ID / PW. (로그인 페이지로 가서 ID/PW를 입력한다.)
Post-condition	I get confirmation of login (로그인 여부를 확인받는다)
Trigger	Press the Login button. (로그인 버튼을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. User : Enter your ID / PW on the login page. (로그인 페이지에서 사용자의 ID/PW를 입력한다.) 2. Server : Get the ID / PW of the user from DB and compare them.(사용자의 ID/PW를 DB에서 가져와서 비교한다.) 3. User : Receive login success(로그인 성공 여부를 받는다)
Extensions(Alternative flows)	2.a. Poor data transfer: error popup (데이터 전송이 안됐다 : 에러 팝업)
Special requirements	<ol style="list-style-type: none"> 1. Press the login button, and the popup will show whether it succeeds or fails.(로그인 버튼을 누른 후 성공,실패 여부는 팝업으로 띄워준다) 2. Show in Korean. (한국어로 보여준다) 3. ID / PW If you do not have one of them, press the login button to pop up a message and not go to the next step. (ID/PW 둘 중에 하나 적지 않았을 시 로그인 버튼을 누르면 팝업을 띄어주고 다음 단계로 넘어가지 않게 한다.)

Technology and data variations list	<p>1.a. Request ID / PW information of user from Web server to DB server. (Web서버에서 DB서버로 유저의 ID/PW 정보를 요청한다.)</p> <p>2.b. The Web server compares the request result of the DB server, receives the response, and sends the response. (Web서버는 DB서버의 요청 결과를 비교한 후 받아서 response를 보낸다.)</p>
Frequency of occurrence	When a user wants to use a website (유저가 웹사이트를 이용하고 싶을 때)
Miscellaneous	1. When the network environment is bad. (통신이 안될 경우)

3.2.2.8 Select lecture

Use Case Name	Select Lecture (강의 선택)
Goal in Context	<p>Professor,TA : Select one of the list of lectures registered by the professor. (교수가 등록한 강의목록중 하나를 선택한다.)</p> <p>Student : Find the list of lectures you took from the lecture list. (강의 목록에서 자신이 수강한 강의 목록을 찾는다.)</p>
Scope	Professor, TA, Student
Level	!
Primary Actor	Professor, TA, Student
Secondary Actor	Server
Stakeholders and Interests	<p>Professor, TA : I want you to check the list of your lectures registered by the professor. (교수가 등록한 자신의 강의목록을 확인했으면 좋겠다.)</p> <p>Student : The list of courses the student is taking can be found in the entire course list and can be selected. (학생이 수강중인 강의목록을 전체 강의목록에서 찾을 수 있고, 선택할 수 있다.)</p>
Preconditions	Professor, TA : Press Select Lecture

	<p>(강의 선택을 누른다.)</p> <p>Student : Press Select Lecture.</p> <p>(강의 선택을 누른다.)</p>
Post-condition	<p>Professor, TA :</p> <p>Identify and select the lectures the professor has enrolled.(교수가 등록한 강의를 확인 및 선택한다.)</p> <p>Student : Find and select your own lecture. (나만의 수강중인 강의를 찾고, 선택한다.)</p>
Trigger	<p>Press the lecture selection button.</p> <p>(강의 선택버튼을 누른다.)</p>
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Student,Professor,TA : Press the lecture selection button. (강의 선택 버튼을 누른다.) 2. Server : Get a list of all the lectures. (전체 강의목록을 가져온다.) 3. Student : Select the lecture you are taking in the entire lecture. (전체 강의중 자신의 수강중인 강의를 선택한다.) 4. Professor,TA : Identify and select the lectures the professor has enrolled.(교수가 등록한 강의를 확인하고 선택한다.)
Extensions(Alternative flows)	<p>1a. Server can not transfer data. Error popup. (Server에서 Data전송이 안됐다.Error팝업)</p>
Special requirements	<ol style="list-style-type: none"> 1. A list of lectures is shown in Korean. (한국어로 강의목록이 보여진다.)
Technology and data variations list	<p>1a.Request a list of all registered courses from Web server to DB server. (Web서버에서 DB서버로 전체 등록된 강의목록을 요청한다.)</p> <p>2a. The Web server receives the request result from the DB server and sends the response. (Web서버에서는 DB서버의 요청 결과를 받아서 response를 보낸다.)</p>
Frequency of occurrence	<p>When the new semester begins (새학기가 시작될 때)</p>

Miscellaneous	1. When the network environment is bad.(통신이 안될 경우)
---------------	--

3.2.2.9 Fetch my SAGI

Use Case Name	Fetch my SAGI (자신의 강의출결 조회)
Goal in Context	You can see your attendance status. (자신의 출결 상황을 볼 수 있다.)
Scope	Student
Level	!
Primary Actor	Student
Secondary Actor	Server
Stakeholders and Interests	Professor: Professor would like to be able to easily check the attendance status of each course taken by students.(학생들이 수강한 과목당 출결 현황을 쉽게 확인할 수 있었으면 한다.)
Precondition	Access attendance inquiry page. (출석 조회 페이지에 접속한다.) You must login. (로그인을 해야한다.)
Post-condition	The user must listen to the subject. (사용자가 해당 과목을 들어야 한다.)
Trigger	The student clicks on his / her subject to see the attendance. (학생이 출결을 보고 싶은 자신의 과목을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Student: Choose the course you want to attend.(수강하는 과목중 출결을 보고 싶은 과목을 선택한다.) 2. Server: Bring the attendance of the subject.(해당 과목의 출결을 가져온다.) 3. Student: Check the status of attendance.(출결 상황을 확인한다.)
Extensions(Alternative flows)	2.a. The data transfer is not successful: an

	error popup appears. (데이터 전송이 안됐다.: 에러 팝업을 띄워준다.)
Special requirements	1. Show in Korean.(한국어로 보여준다.)
Technology and data variations list	1.a. Web server to the DB server to request the attendance status of the subject. (Web서버에서 DB서버로 해당 과목의 출결 상황을 요청한다.) 2.b. The Web server receives the request result from the DB server and sends the response. (Web서버는 DB서버의 요청 결과를 받아서 response를 보낸다.)
Frequency of occurrence	When a student is interested in viewing (학생이 조회할 마음을 먹었을 때)
Miscellaneous	1. When the network environment is bad. (통신이 안될 경우)

3.2.2.10 Accept lecture

Use Case Name	Accept lecture (학생 강의 신청 승인)
Goal in Context	When a student requests a course, they can approve it. (학생이 강의수강을 신청하면 그것을 승인해줄 수 있다.)
Scope	Professor, TA
Level	!
Primary Actor	Professor, TA
Secondary Actor	Server
Stakeholders and Interests	Professor : I would like to add only students who attend classes to the list of attendance. (수업을 듣는 학생들만 출결 명단에 추가하고 싶다.)
Preconditions	Access students' application list page. (학생들의 수강신청 명단 페이지에 접속한다.)
Post-condition	The student is added to the lecture attendance list. (학생이 강의 출결 명단에

	추가된다.)
Trigger	Press the acceptance button for student enrollment. (학생의 수강신청 승인 버튼을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Student : Apply for enrollment(수강신청을 한다.) 2. Professor : View the list of applications and click the Accept button.(신청 내역 목록을 보고 승인 버튼을 누른다.) 3. Server : Add the student to the attendance list of the lecture.(강의 출결 명단에 해당 학생을 추가한다.)
Extensions(Alternative flows)	2.a. Poor data transfer: error popup (데이터 전송이 안됐다 : 에러 팝업)
Special requirements	1. Write in Korean. (한국어로 작성한다.)
Technology and data variations list	<ol style="list-style-type: none"> 1.a. Add students to the application queue. (신청 대기열에 학생을 추가한다.) 2.a. The web server adds the student to the lecture. (Web서버가 학생을 강의에 추가한다.)
Frequency of occurrence	When a professor wants to manage students in the subject (교수님이 해당 과목의 학생들을 관리하고 싶을 때)
Miscellaneous	<ol style="list-style-type: none"> 1. If you accidentally click Unapproved. (실수로 승인취소 누르는 경우)

3.2.2.11 Register fingerprint

Use Case Name	Register fingerprint (지문 등록)
Goal in Context	A student, a professor, or a TA registers his / her fingerprint. (학생이나 교수, TA가 본인의 지문을 등록한다.)
Scope	Student, Professor, TA
Level	!
Primary Actor	Student, Professor, TA

Secondary Actor	Server
Stakeholders and Interests	Professor : Students want to authenticate with personal information that they can not steal. (교수 : 학생이 함부로 도용할 수 없는 개인 정보로 인증하고 싶다.)
Preconditions	Students, professors and TA join the membership. (학생, 교수, TA가 회원가입을 한다.)
Post-condition	Fingerprints of students, professors and TA are registered. (학생, 교수, TA의 지문이 등록된다.)
Trigger	Press the fingerprint registration button. (지문 등록 버튼을 누른다.)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Student, Professor, TA :Join the membership. (회원가입을 한다.) 2. Student, Professor, TA : Fingerprint registration is done. (지문 등록을 한다.)
Extensions(Alternative flows)	2.a. Poor data transfer: error popup(데이터 전송이 안됐다 : 에러 팝업)
Special requirements	1. Write in Korean. (한국어로 작성한다.)
Technology and data variations list	TBD(To be decided.)
Frequency of occurrence	As soon as you sign up.(회원 가입을 하자마자.)
Miscellaneous	<ol style="list-style-type: none"> 1. If the device is not turned on. (기기의 전원이 켜져있지 않은 경우) 2. If the machine fails. (기계가 고장이 난 경우)

3.2.2.12 Attend

Use Case Name	Attend (출석)
Goal in Context	Save student attendance check.(학생들의 출석 체크를 저장한다.)
Scope	Student
Level	!

Primary Actor	Student
Secondary Actor	Server
Stakeholders and Interests	Professor : The professor wants to have the student attend the fingerprint recognition without the attendance check by the professor.(교수가 직접 호명하면서 출석체크를 하지 않고 학생이 알아서 지문인식으로 출석하게 하고 싶다.)
Preconditions	Students print the fingerprints on the fingerprint register (학생들이 지문등록기에 등록된 지문을 댄다.)
Post-condition	Whether or not attendance checks are stored in the DB. (출석 체크 여부가 DB에 저장된다.)
Trigger	When students take their fingerprints on the fingerprint register (학생들이 자신의 지문을 지문등록기에 찍었을 때)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Student : Print fingerprints on the fingerprint register.(지문등록기에 지문을 찍는다.) 2. Server : Save the attendance check in the DB.(출석 체크 여부를 DB에 저장한다.)
Extensions(Alternative flows)	<p>2.a. Fingerprints do not match: Error Popup (지문이 일치하지 않는다 : 에러 팝업)</p> <p>2.b. Fingerprint not sent: error popup(지문이 전송되지 않는다: 에러 팝업)</p>
Special requirements	1. Write in Korean. (한국어로 작성한다.)
Technology and data variations list	TBD
Frequency of occurrence	Every time a lecture begins (강의가 시작할 때 마다)
Miscellaneous	<ol style="list-style-type: none"> 1. If the device is not turned on(기기의 전원이 켜져있지 않은 경우) 2. If the machine fails(기계가 고장이 난 경우)

3.2.2.13 View information

Use Case Name	View information
Goal in Context	The system mediates the information needed. (시스템이 필요로 하는 정보를 중개해준다.)
Scope	Server
Level	!
Primary Actor	Server
Preconditions	Server turns on
Post-condition	Get the information you need for the function. (기능에 필요한 정보를 얻는다.)
Trigger	When information is requested from the user (유저로부터 정보를 요청받았을때)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Server : Request information from the user.(유저로부터 정보를 요청 받는다.) 2. Server : returns information.(정보를 리턴해준다.)
Extensions(Alternative flows)	2.a. Poor data transfer : error popup. (데이터 전송이 안됐다 : 에러 팝업)
Frequency of occurrence	Every time information is requested. (정보가 요청될 때 마다)
Miscellaneous	<ol style="list-style-type: none"> 1. If the server is dead(서버가 죽은경우) 2. If you can not communicate with the server(서버와 통신이 안되는경우) 3. If you do not have the information you need(필요로 하는 정보가 없는 경우)

3.2.2.14 Create information

Use Case Name	Create information
Goal in Context	Store information on the system (시스템에 정보를 저장한다.)
Scope	Server
Level	!

Primary Actor	Server
Preconditions	Server turns on
Post-condition	Inputs the received information (입력받은 정보를 넣는다.)
Trigger	When asked to save new information to the server (새로운 정보를 서버에 저장해달라는 요청이 들어왔을때)
Main success scenario(Basic flow)	<ol style="list-style-type: none"> 1. Server : I was asked to store new information on the server.(새로운 정보를 서버에 저장해달라는 요청을 받았다.) 2. Server : The information is received and stored in the server.(정보를 받아 서버로 저장한다.)
Extensions(Alternative flows)	2.a. Poor data transfer: error popup (데이터 전송이 안됐다 : 에러 팝업)
Frequency of occurrence	Every time I receive a request to save new information. (새로운 정보를 저장해달라는 요청을 받을때마다)
Miscellaneous	<ol style="list-style-type: none"> 1. If the server is dead(서버가 죽은경우) 2. If you can not communicate with the server(서버와 통신 안되는경우) 3. If the category you want to save is wrong (저장하려는 카테고리 잘못 되었을 경우)

3.3 Performance requirements

- No requirement.(요구사항 없었음.)

3.4 Design constraints

3.4.1 Software Constraints

- Coding with Python.(파이썬으로 코딩한다.)
- Conventions follow PEP8.(컨벤션은 follow PEP8.)
- The Database API uses mysql-python.(Database API는 mysql-python을 사용한다.)
- Template Engine uses Jinja2.(Template Engine은 Jinja2를 사용한다.)

3.4.2 Hardware Constraints

- TBD: Technology in the development of hardware parts.(하드웨어 파트 개발 시에 기술)

3.5 Software system attributes

3.5.1 Security

- The password is encrypted and stored.(비밀번호를 암호화하여 저장한다.)
- Students will not be able to see other people's attendance information.(학생들은 다른 사람의 출결 정보를 볼 수 없다.)

3.5.2 Reliability

- AWP Service Downtime does not exceed 4 hours.(AWP의 Service 정지 시간이 4 hours를 넘기지 않는다.)

3.5.3 Availability

- The device on which the web browser is running must be connected to the Internet.(Web browser가 실행되는 device는 인터넷에 연결되어야만 한다.)