Attendance Web Page SRS

Table Of Contents

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 노티 - 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

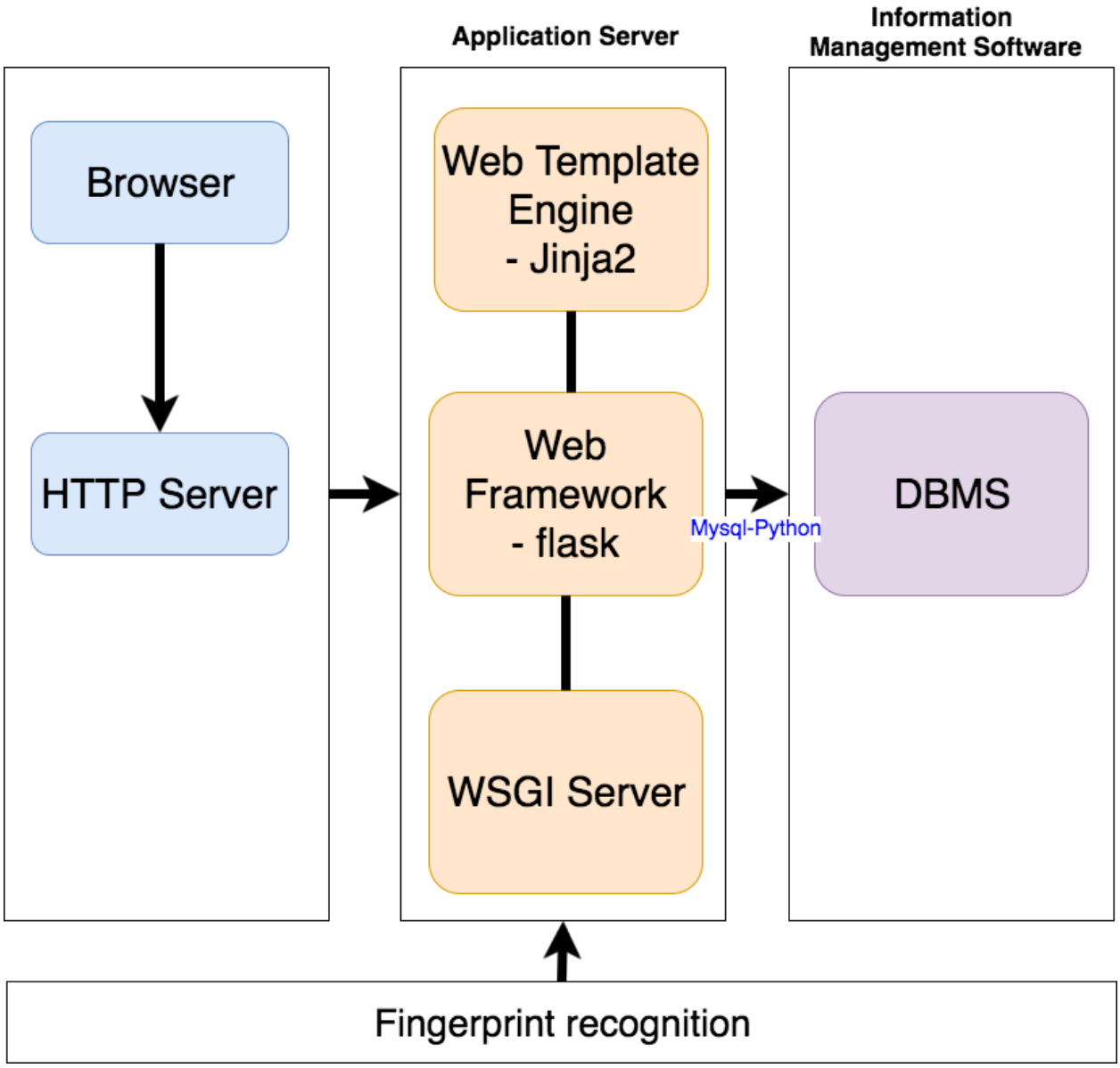
- Students, professors, and assistants are registered. (학생과, 교수와, 조교를 등록한다.)
- Admin helps to register the professor. (Admin은 교수를 등록하는데 도와준다.)
- Notification or daily report is sent. (노티피케이션이나 데일리 리포트를 보내준다.)

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 노티해준다.)
- 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.(지문 인식에 대한 constraints => TBD)
- 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.1)

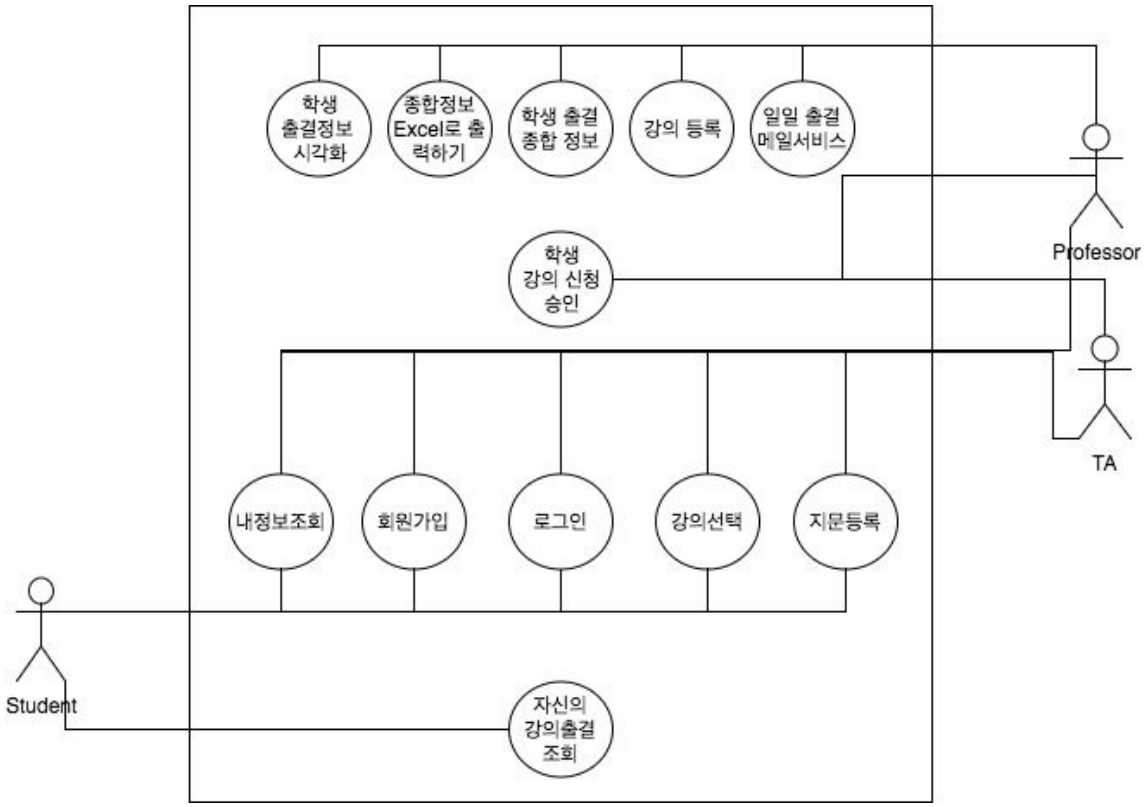
2.6 Apportioning of requirements

- Chrome Notification

3. Specific requirements

3.1 External interface requirements

3.1.1 User interfaces



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 User class (Professor)

3.2.1.1 Make class

3.2.1.2 Accept student in class

3.2.1.3 Accept TA in class

3.2.1.4 Export attendance to excel

3.2.1.5 View attendance visually

3.2.2 User class (TA)

3.2.2.1 Accept student in class

3.2.3 User class (Student)

3.2.3.1 Sign in class

3.2.3.2 View their attendance

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.)