School of Computer Science

University of Nottingham Ningbo China (UNNC)

COMP2043. GRP Interim Group Report

Smart Class Attendance Taking System

Group 17

Supervisor: Zheng LU

Name	Username
Boya Wang	slybw2
Mingchen Li	scyml3
Yiming LI	scyyl3
Guohao Yu	scygy1
Hongyi Zhu	scyhz1

2019.12.12

1. Updated and Extended Description of Project

This project is to develop a face-recognition based student attendance recording system. There are two main problems in this project to solve: to implement face recognition and to develop a fully functional system implementation for management.

1.1 Face-recognition implementation:

The face recognition function will be realized by calling APIs from the third-party of Python libraries as described in the requirement. This is for that we may not have enough ability to develop a well-performed face recognition model which can meet expectancy, and the key of this project is not to train the model but to implement functions for practical use. For many of the open-source recognition models may not have an expected performance when recognizing Asian face, the enhancement in this project is to find an existing model which also works well for Asian face.

1.2 Face-recognition function:

For face recognition, the core is to identify and recognize the faces of students enrolled in a specific module, and then record the arrival time of these students for each teaching session noted in the school schedule. When the face of a student is identified, real-time data should be visualized and displayed on UI for observation. For example, the number of attended

students for the current teaching session will be shown in the format of a progress bar, and the name of the recognized student should be displayed. The process of face recognition can be stopped by teachers at any time so that it's more flexible to use.

After stopping face recognition, the percentage of attendance and a list of missing students should be shown as the result of attending the recording process. and the software should then send this information to the remote database.

There are some additional subproblems in face recognition problem as well. The process of face recognition should be paused and continue manually by using buttons on face recognition UI. The reason is that sometimes the teacher may want to pause the recording and he doesn't want to restart the recording.

Considering about practical usage, it's a waste of time to recognize student one by one. To improve the efficiency of face recognition, this project allows identifying faces of several students concurrently when they appear in the recognition area at the same time. Since there is corresponding support from third-party APIs for realizing this function, this problem can be solved initially.

Another point to be noted is that there is a large number of students with glasses. For the student who wears the glasses, the face recognition function should be able to accurately identify who he is rather than incorrect recognition or fail to identify.

1.3 Fully Functional System implementation:

The whole system should have a well-organized GUI in the front end to simplify the human-computer interaction, and a remote database to store all the relative information, including the information of students, teachers and attendance.

The system has two kinds of access, one is for the teacher, the other is for admin. Both of them need to login in the system with their unique username and password. The reason for two kinds of access is that the teacher is the user to use the software practically and admin is the administrator to manage the system.

1) For admin, he has permission to manage the whole system. The core work for admin is to manage the remote database.

   He can easily access each level in management system from a drop-down box in GUI, which contains several levels such as faculties, majors, modules, teaching sessions, teachers and students. Some of

them may be at the same level. In each level, admin can update or delete the existing information, and add or import the extract information. In student level and module level, if some specific students missed classes more than 3 times, there will be a warning, so that the admin can quickly alert the relative staffs.

2) For teachers, they have permission to use the system. They can start the face recognition when it's about time. There are several modules shown like the dashboard in Moodle.

The teacher can access their teaching modules and their teaching sessions shown as a list. If the teaching session has not to be finished, the teacher can start the face recognition for this specific teaching session. After the teaching session has done, the teacher can review the attendance result of that teaching session.

The teacher can search for the specific student and specific teaching session as well to quickly check the attendance of students and teaching sessions. Since it's the crucial demand for this project.

Both teachers and admin can export what they what, including attendance list for specific students, modules or teaching sessions. The reason for that

is to simplify the summarizing attendance for submitting to the superior.

2. Survey of any existing systems that address similar problems

2.1 Comparison with existing software

There are a significant number of facial recognition attendance taking systems used in companies and universities nowadays. For example, *DingTalk*, *Baidu Brain*, *Jibble*, *Integrated…*

The following includes two examples of the existing systems. *DingTalk* represents the facial recognition attendance taking system used domestically. *Jibble* is a representation in overseas.

1) DingTalk:

DingTalk, a professional office application for small and medium-sized enterprises (SMES), is an integration of multiple functionalities. Facial recognition attendance system is applied on *DingTalk* whose version is over 4.0 by using Ding Talk M2 Smart Receptionist and *DingTalk* C1 Smart Communication Center, which can identify whether picture or an alive person and recognize it even under extreme lighting condition. The whole system is able to be set up in 4 steps, downloading *DingTalk*, connecting facility by blue tooth, connecting WIFI and bonding with specific group, which is extremely user friendly. The system can identify multiple people

quickly when they expose to the camera. Once the person passes the facial recognition, the attendance sheet will be updated by the system. However, facial recognition attendance system is only a small part of the whole *DingTalk*.

By 2018, *DingTalk* had have more than 100 million users and became one of the world's largest professional communication and management application in China.(DingTalk, 2019)

2) *Jibble*:

Jibble is an application mainly for record the attendance with sign in time and sign out time if passed the facial recognition. Then the system offers timesheet views and report to the administrator. By using this application, the administrator can have direct access to employee statistics and improve the productivity. *Jibble* system is based on mobile devices like iPad, so it can only recognize one person at one time using the front camera.

However, *Jibble* doesn't have multiple functionalities like *DingTalk*. Since the system works on individual device, cannot identify multiple person does not affect a lot. (Jibble, 2019)

2.2 Market research and interview

We did our market research by interviewing a few teachers in our university. The question we asked include the problem of current attendance sheet, the expected features of our system and the time requirement of taking

attendance. According to the result, our teacher in university are willing to use a software which can take place of attendance sheet. The problems of current attendance sheet, the expected features of this software and the time requirement are shown below.

1) Problems of current attendance sheet:
    a. Inconvenient for teacher
    b. Need to print out an attendance sheet and store the attendance sheet into computer manually
    c. Student can sign or each other, since there is no verification

2) Expected features:
    a. GUI should be user-friendly.
    b. Easy and convenient to do the attendance recording.
    c. Offer the list of students who has be absent for more than three times
    d. Searching by lecture name and show details
    e. Comparation of attendance rate of two lectures in same module
    f. Comparation of overall attendance rate of two modules
    g. Make sure the safety of data
    h. Students can only sign in for themselves

3) Time requirement:

   a. If record the attendance after class, should make sure it will finish in ten minutes
   b. The system should be stable and accurate to record as soon as possible

In conclusion, the system should be easy to use for teachers and reliable enough to reduce the time of recording attendance. In general, by using this system, student cannot sign for each other anymore and the attendance sheet can be stored automatically.

2.3 Technical research

1) Platforms:

   a. Anaconda is used to finish our project since we were asked to finish it with Python and it is the easiest one to use. In addition, it is free. The version of anaconda is over 3 and it contains 3 bask components, which are Anaconda Navigator, Jupyter Notebook and Spyder.
   b. OpenCV packages are imported for picture processing and it is free to use. That is why we choose it.

c. There are 3 suitable packages for UI design. However, the UI made by Tkinter is too simple. PyQt is faster and easier than Kivy. As a result, PyQt is used to design UI.

2) Tools

Camera is the solely tools that the project requires.

3) Technologies

Face recognition system and school management system is needed.

4) Algorithms

Machine learning algorithms are used. The program should learn from the existing student pictures stores in database and recognize whether the student is in the relative department.

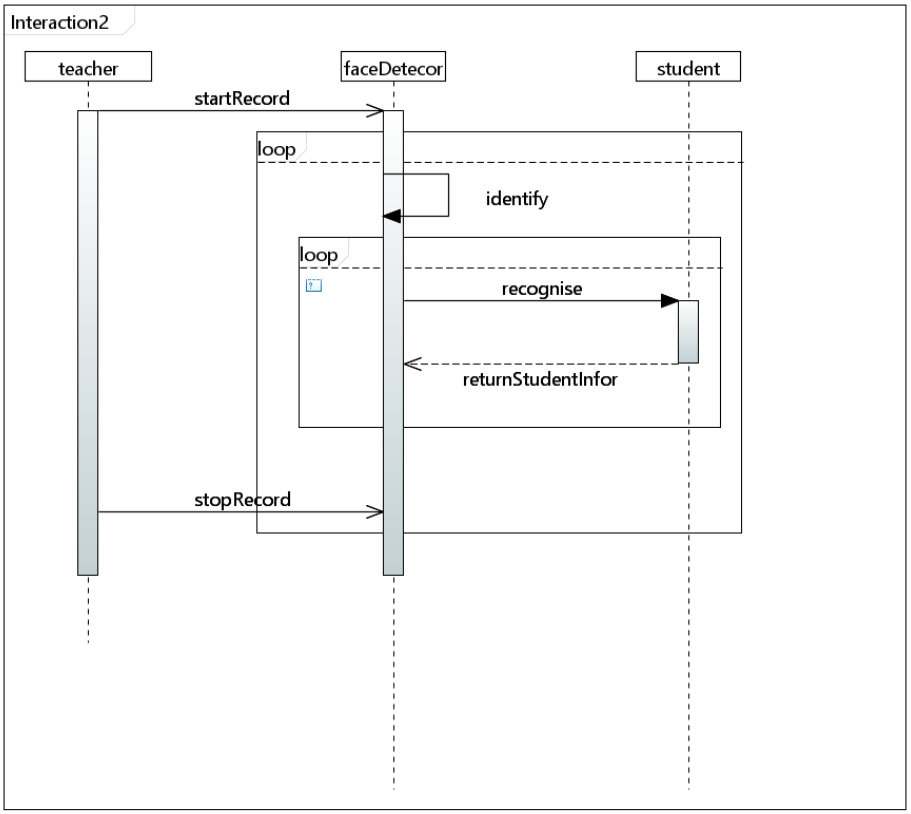
3. SRS

3.1 Diagrams

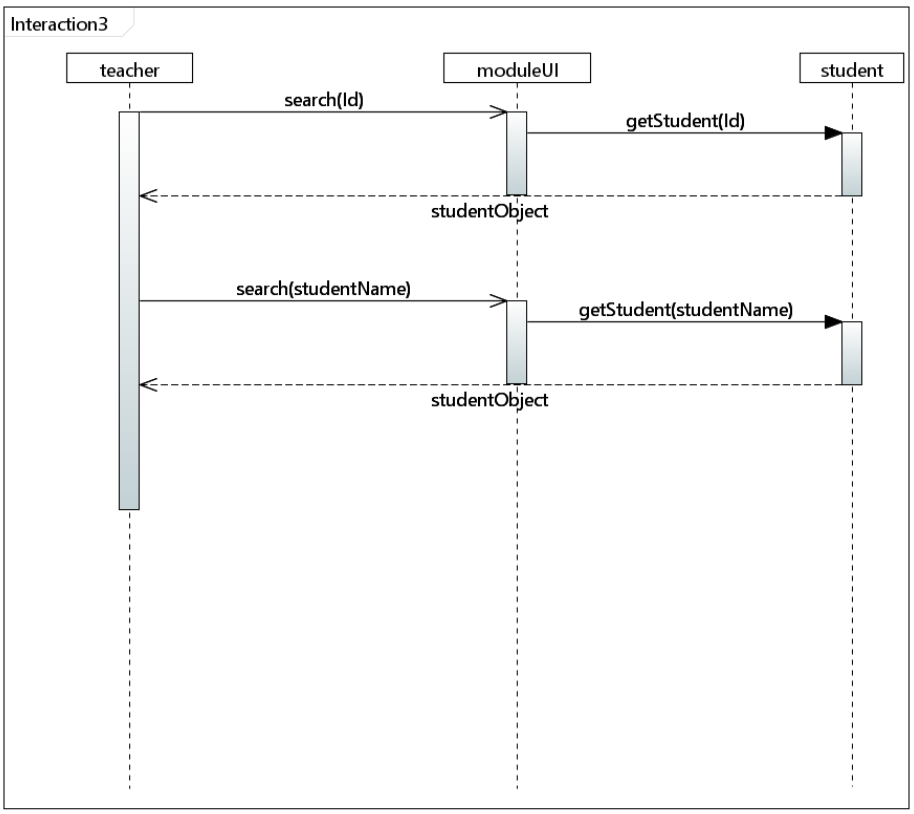
1) Use Case Diagram

The identified users include teacher, administer and face detector. The function of face detector is to recognize face. The common functions of teacher and administer include system log-in, system log-out, searching and viewing attendance information with different categories. For their singular responsibilities, the teacher can edit the information of teaching session, start and stop attendance recording of a teaching session, and edit attendance information. The administer can establish and edit the database.

2) Sequence Diagram

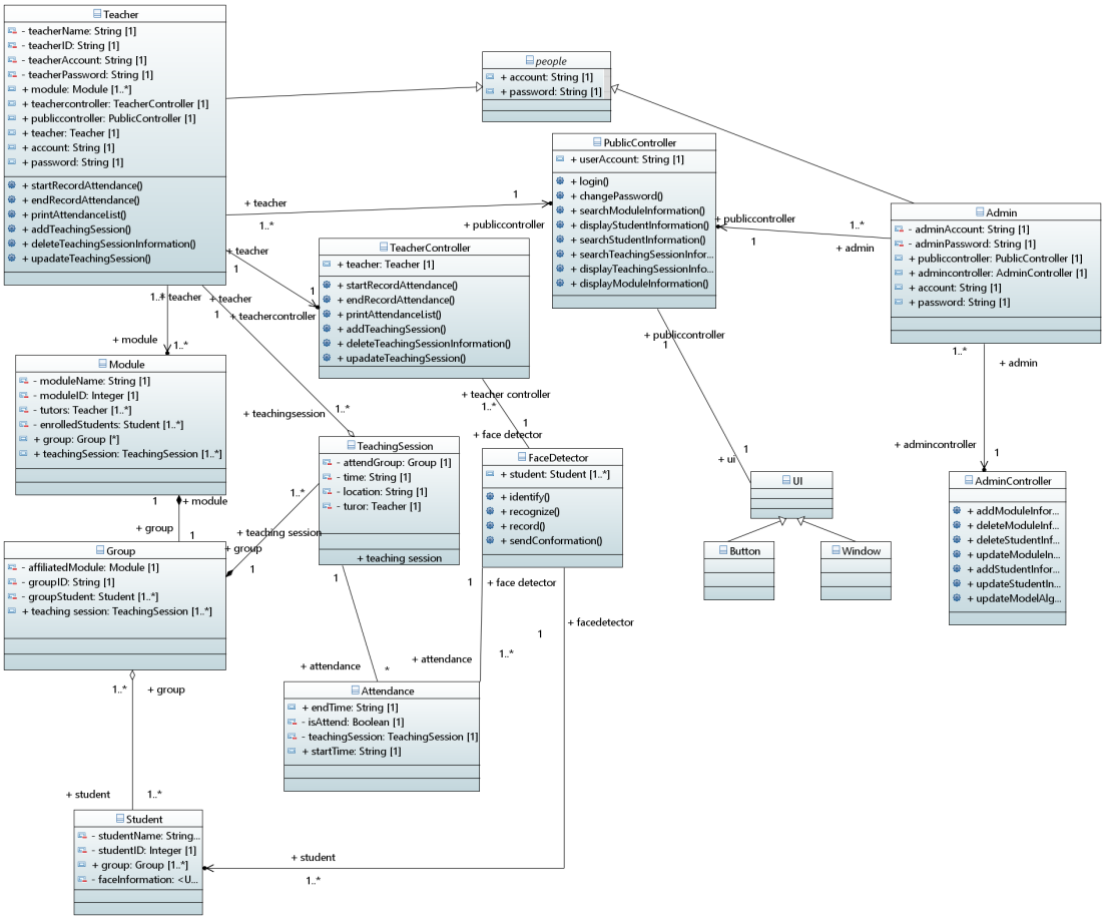


This is the sequence diagram of face recognition function of this system. The teacher will start recording firstly. Then the face detector will receive the signal and keep identifying a face until teacher stop recording. After a face is identified, the face detector will go to recognize the face to by conducting a loop of comparing the recognition information and face information of each student. The information of a successfully recognized student or an unfound signal will be returned to the face detector.



This is the sequence diagram of searching student function in teacher class. A student can be found by his id or name. The teacher can firstly try to input the student id in module page, and in this page the student with input specified id will be found. The found student object will be returned to module page, and then shown for teacher. Then the teacher may search another student by name, the process will be the same.

3) Class Diagram



Teacher class and Admin class contain the constructor of all the teacher or admin information. Both of them implements corresponding Controller class and PublicController class to do log in operation, recording operation, modify lecture information operation, modify students' information operation, searching operation or print operation, because different teachers have different results when doing the same operation.

Teacher class and Admin class implement one same class which is PublicController class. It contains all the methods that teacher and admin can invoke. Meanwhile, TeacherController and AdminController respectively contains methods for only teacher or admin.

Also, Teacher class and Amin class both extend the People class, because both of them have accounts and passwords. The Student class does not have the similarity between these classes, so it does not implement the People class.

a) The Module class contains a constructor including all the module information, a teacher object and a group object.
b) The Group class contains a constructor including all the group information, a

module object and a list of student objects.

c) The Student class contains. a constructor including all the student information and a list of group objects.

d) The TeachingSession class contains a constructor including all the lecture or lab information, a group object and a teacher object.

UI classes are not completely included in the class diagram because of the complexity of it will affect the concision and readability. It can be separated into multiple subclasses as different specific buttons and windows which links to FaceDetector class and all the controller classes.

3.2 User requirement:

1) User characteristics:

The main users include database administer, teachers, students and a face detector.

2) Functional requirement:

- The users can interact with the system through a graphic user interface.
- Common functions:
    a. User login
    b. User logout
- For face detector
    a. Recognize student face image
    b. Identified student face image
    c. Signal identification success or failure
- For administer
    a. Add/ update/ delete teacher accounts in user database
    b. Import/ update/ delete student information in database
    c. Import/ update/ delete information of teaching session in database
    d. View database
- For teacher
    a. View his own profile
    b. Set/update name and time of teaching session
    c. Start/continue/end class attendance recording
    d. Search information based on teaching session/student name
    e. View searched information
    f. View attendance sheet of a module/teaching session/student
    g. View immediate attendance condition for a class
    h. Edit attendance information and add remark of the edit

3) Un-functional requirement:

a. Availability: the administer shall check and maintain the database or server, set software maintenance time and make sure the system can run

during the work period.

b. Reliability: the system shall recognize and identify students correctly and prompt notice if the student cannot be identified.
c. Safety: the system shall prompt the name of identified student in case of mis-recognizing the student.
d. Security and Privacy: only users holding the role "admin" can access database and server.
e. User friendliness: the system shall provide user-friendly GUI, user-friendly operation button, immediate relative notice to the users if their operation is wrong.
f. Maintainability: make use of the structural software design method, OOP and quality management.

4) User interface requirement:

a. Develop Graphical User Interface
b. The UI should be able to support all operations noted in functional requirement
c. The UI should be clear to understand, and easy to use
d. The UI should be responsive in short time
e. The UI should be consistent on all interfacing screens
f. The UI should have a proper appearance which suits its users

3.3 System Requirement:

1) Functional requirement:

- **Common functions**

a. User login

Pre-condition: user logged out

Operation: enters user school ID and password, click confirm button.

Post-condition:

- If information is correct, change user state to log in. Show the main page.
- If not: stay user state, stay in the current page. Prompt a warning window asking the user to enter input again.

b. User forget password

Pre-condition: user logged out

Operation: click *forget password* link

Post-condition: Show help page with instruction text on it

c. User logout

Pre-condition: user logged in

Operation: click *teacher Info* link, and click logout option

Post-condition: change user state to logout, and show log-in page

- **For face detector**

a. Recognize student face image

Pre-condition: face detector is on and the user clicked start recording button

Operation: locate a face, read in and store a shot-cut of the face

Post-condition: a frame of face recognition stream is loaded in and stored in local file

b. Identify student face image

Pre-condition: a face has been recognized

Operation: compare face data and identify the corresponding student

Post-condition:

- If a student is identified, add his attendance information
- If the face matches no student, return false value

c. Signal identification success or failure

Pre-condition: a face has been processed with identification operation

Operation:

- If recorded successfully, show name and ID of identified student on page
- If recording failed, show error message on page

Post-condition: the user receive feedback message of operation

- **For teacher**

a. View his own profile

Precondition: the teacher account must be logged in

Operation: click on his own profile link

Postcondition: show the view of current teacher account's information.

b. Set/update lecture name and time

Precondition: the teacher account must be logged in

Operation:

- choose a specific lecture to edit
- Input lecture name or lecture time
- click confirm button to save changes

Postcondition: lecture name and time is modified.

c. Start and end class attendance recording

Precondition: the teacher account must be logged in

Operation:

- choose a specific lecture to record attendance
- click the start button to start recording
- can be stop by clicking the stop button
- can continue recording by clicking the continue button after stop
- click end recording button to finish recording

Postcondition: show the view of attendance information of this lecture and the attendance information will be stored in the database.

d. Search information based on lecture/student name

Precondition: the teacher account must be logged in

Operation:

- type in the keyword as lecture or student name

- click search button to start searching

Postcondition: show the view of lecture or student list including the keyword that has been entered.

e. View searched information

Precondition: the teacher account must be logged in and has searched for a keyword of lecture/student name

Operation: click on a specific lecture

Postcondition: show the view of attendance information of this lecture.

f. View attendance sheet of a module/teaching session/student

Precondition: the teacher account must be logged in

Operation: click on the specific module/teaching session/student

Postcondition: show the view of attendance information of this a module/teaching session/student

g. View immediate attendance condition for his class

Precondition: the teacher account must be logged in and has started recording

Operation: /

Postcondition: show the percentage of attendance of current lecture on the right bottom of recording windows.

h. Edit attendance information and add remark of students

Precondition: the teacher account must be logged in

Operation:

- choose a specific student
- choose a specific lecture to edit this student's attendance status
- change the status of current student's attendance
- add notes to this change
- click confirm button to save changes

Postcondition: The student's attendance status of this lecture has been changed.

2) Un-functional requirements:

a. Availability: the administer shall check and maintain the database or server. When the database or the server loses connection, the system must have a warning alert and try to reconnect.

b. Reliability: The expected number of users that the system can support is around 30,000. To achieve this goal, the database should be large enough to accommodate at least 30,000 students, teachers and lectures records. Furthermore, the system must ensure that it does not crash when using. Also, the system shall prompt notice if the student cannot be identified and it shall allow teachers to edit attendance information manually. When this is done, the reliability of the system can be guaranteed.

c. Safety: the system shall prompt the name of identified student in case of mis-recognizing the student. The system shall recognize and identify

students correctly. A confirmation message should be alerted to let students to confirm their information.

d. Security and Privacy: All users' information should be stored in a separated database on the cloud. Documents stored in a separate database can improve the security of the data, only users holding the role "admin" can access database and server and the privacy of students' and teachers' information can also be guaranteed.
e. User friendliness: the system shall provide user-friendly GUI, user-friendly searching operation, immediate alert windows will be notified to the users if their operation is wrong.
f. Maintainability: make use of the structural software design method, OOP and quality management. The administer should be able to add, modify and delete student or teacher or module or lecture information separately or all at once by reading from files. The system must allow administer to update the face identification and recognition algorithm.

3.4 User Interface requirement:

- **Basic design principle**
  a. The sequence of user operations and composing of icons of UI should be consistent in view of the logic of user action.
  b. The function composing of UI should be organized. For example, it should be delivered with rules of hierarchy and composition.
  c. Implement UI with Qt and PyQT5 for better coupling of system functions and portability.

3.5 Method for the system

Method name	Input	Type	Output	Type
login	Username (University ID)	String	checkMessage	Bool
	password	String		
addModuleInformation	ModuleName	String	checkMessage	Bool
	TeacherID	String		
	StudentID	Integer		
	ModuleID	String		
	GroupID	Integer		
deleteModuleInformation	OriginalModuleName	String	checkMessage	Bool
changePassword	oldPassword	String	newPassword	String
deleteStudentInformation		void	checkMessage	Bool
updateModuleInformation	OriginalModuleName	String	ModuleMessage	String

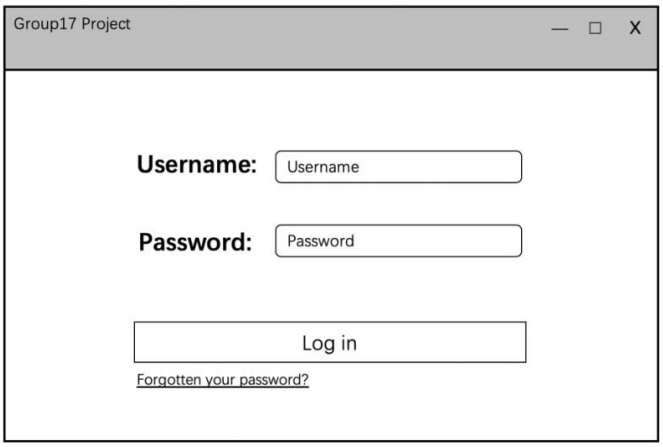
	OriginalTeacherInformation	String		
	NewTeacherName	String		
	NewTeacherID	Integer		
addStudentInformation	studentName	String	checkMessage	Bool
	studentID	String		
	studentPhoto	String		
	attendModule	String		
	attendLectureTime	Integer		
updateStudentInformation	OriginalStudentInformation	String	checkMessage	Bool
	StudentID	Integer		
	StudentName	String		
	ModuleInformation	String		
SearchModuleInformation	ModuleName	String	specificModule	String
	ModuleID	String		
displayStudentInformation		void	SpecificStudentInformation	String
SearchStudentInformation	StudentName	String	specificStudent	String
	StudentID	String		
SearchTeachingSessionInformation	6-digit date	Integer	specificLecture	String
	LectureName	String		
deleteTeachingSessionInformation	OriginalLecture	String	allLecture	String
displayTeachingSessionInformation		void	specificLectureInfor.	String
StartRecordAttendance				Null
EndRecordAttendance			checkMessage	String
PrintAttendenceList	AttendenceOfLecture	String	AttendenceList	String
displayModuleInformation		Null		Null
updateTeachingSession	OriginalLecture	String	LectureMessage	String
	Location	String		
	time	Integer		
	6-digit date	Integer		
addTeachingSession	LectureName	String	LectureMessage	String
	Location	String		
	time	Integer		
	6-digit date	Integer		
UpdateModelAlgorithm	OriginalAlgorithm	String	Confirmation_message	String
	NewAlgorithm	String		

4. Initial Design and Interface

The UML for initial design has been shown above. The initial design of whole UI system of our project is divided into user part and admin part.

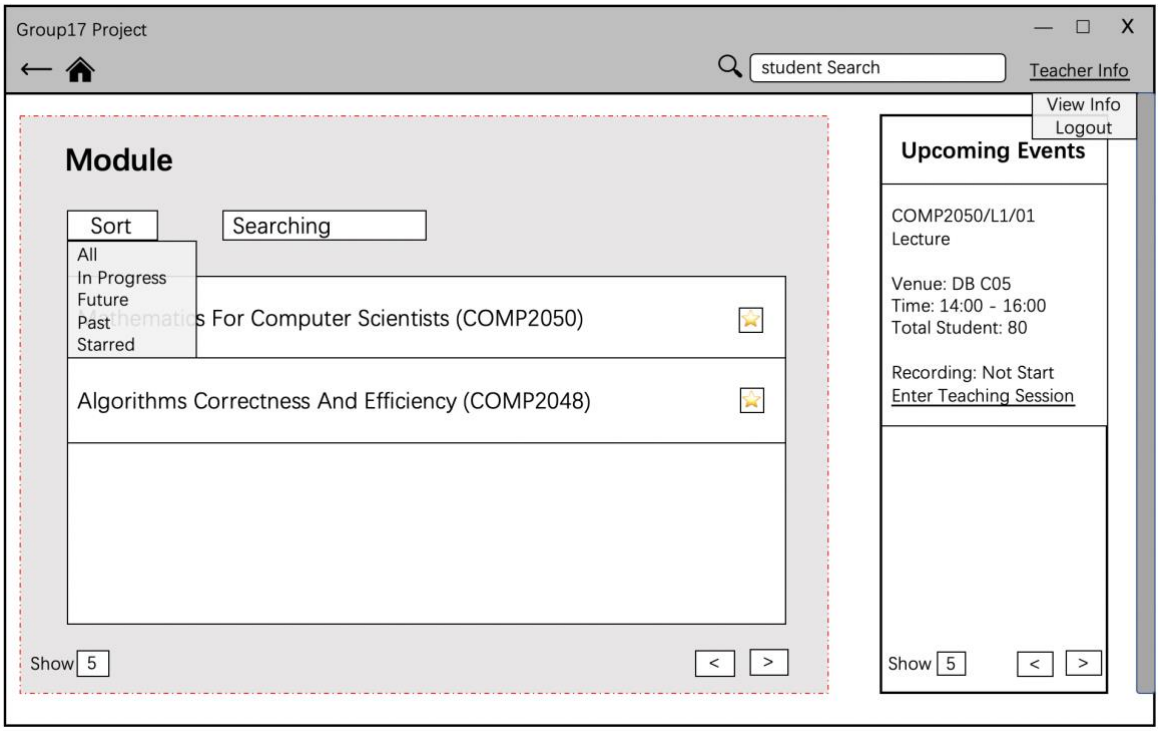
- **User:**

1) Login page



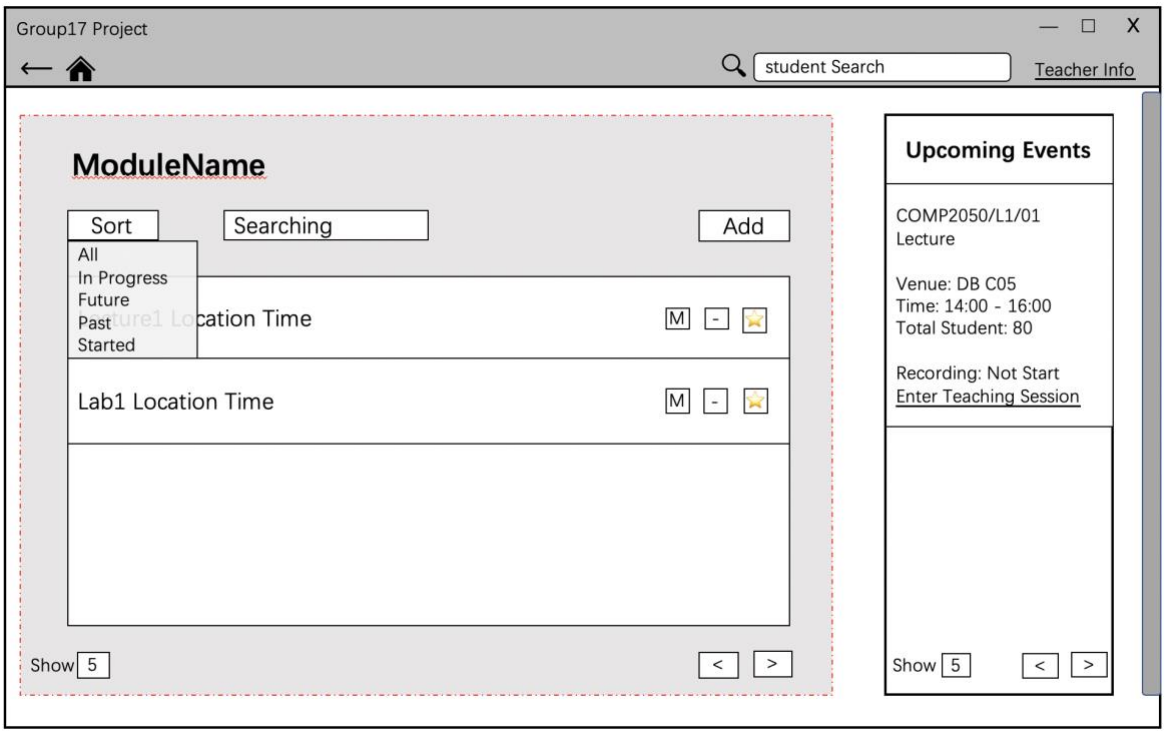
2) Main page

Teachers can select specific module of which he/she wants to record attendance. In addition, they can use “Upcoming Events” to record the closest teaching session.

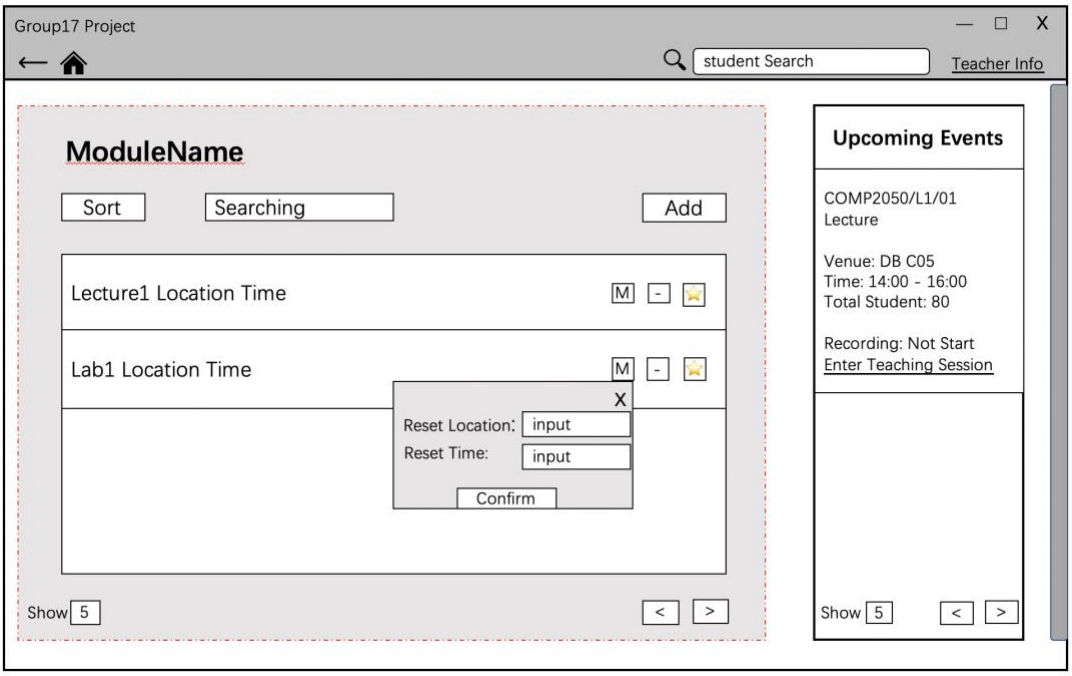


3) Session Page

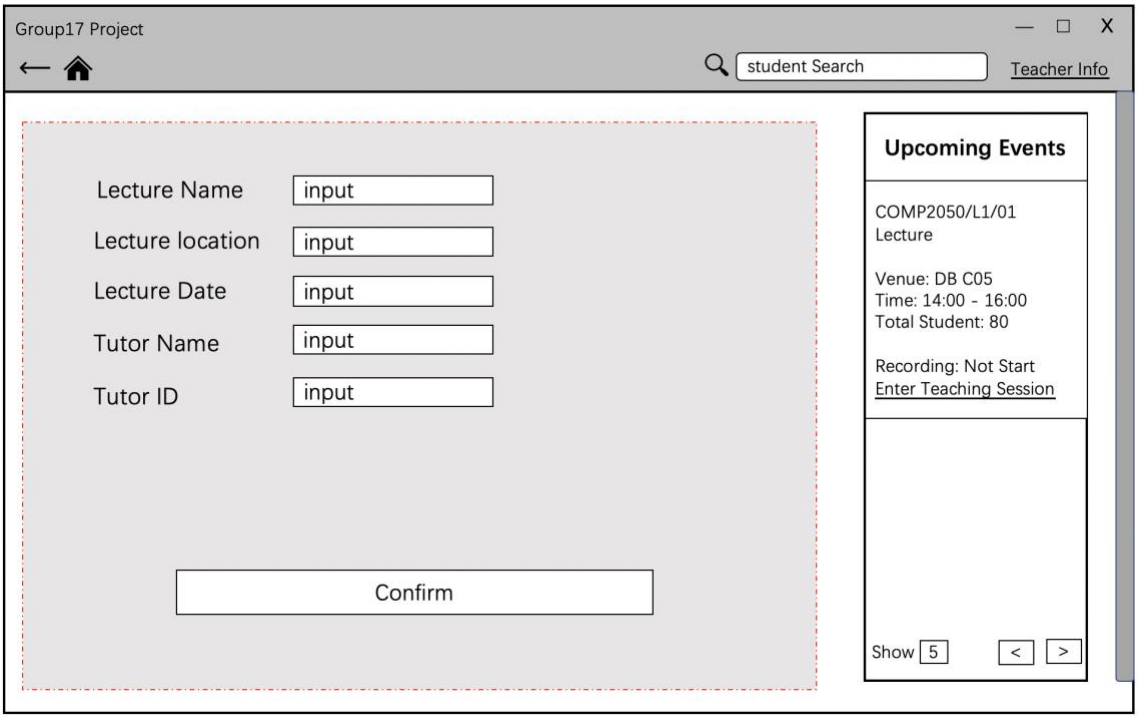
This page will be shown after click one module. In this page, teacher can modify information of each teaching session and start recording of attendance.

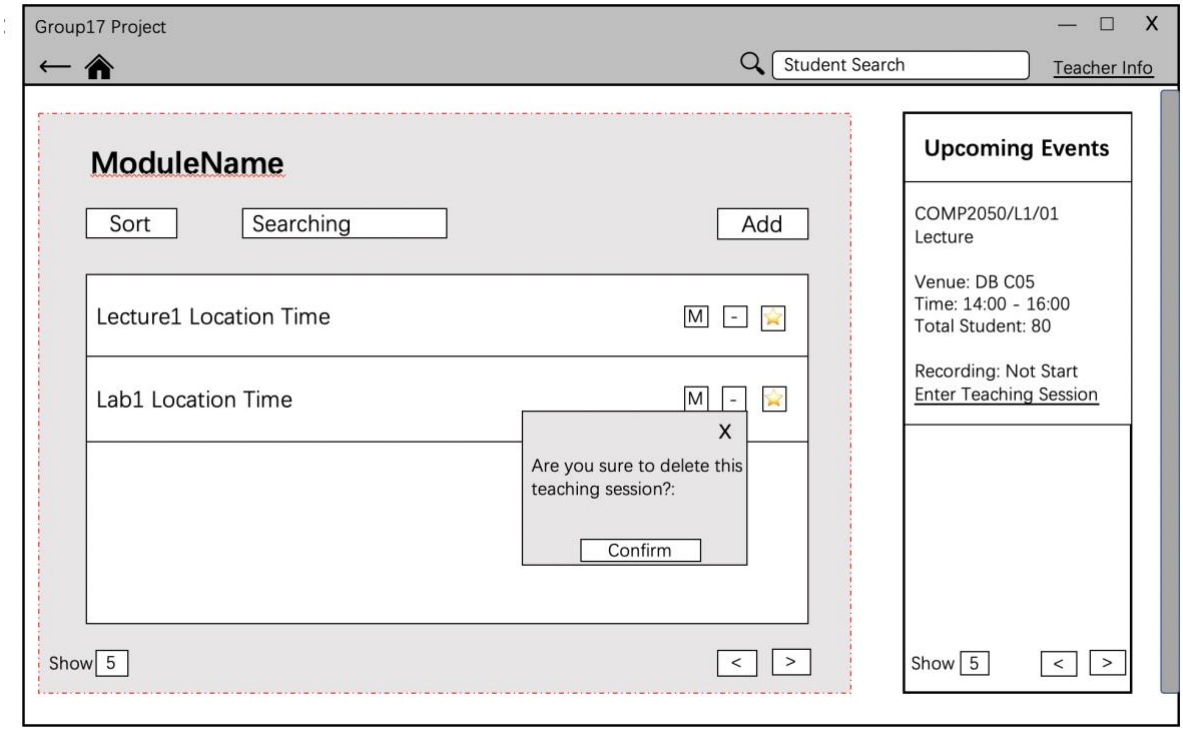


Teacher can reset time of location of one specific teaching session.



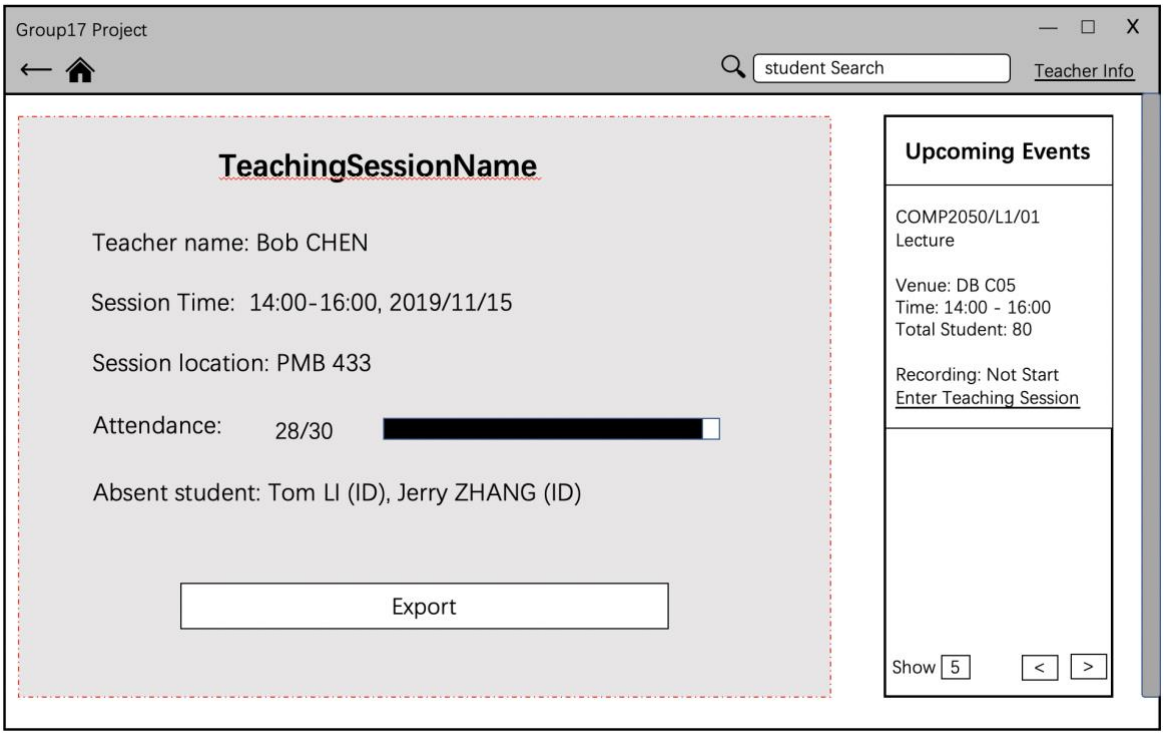
Teacher can add one teaching session by click Add button.



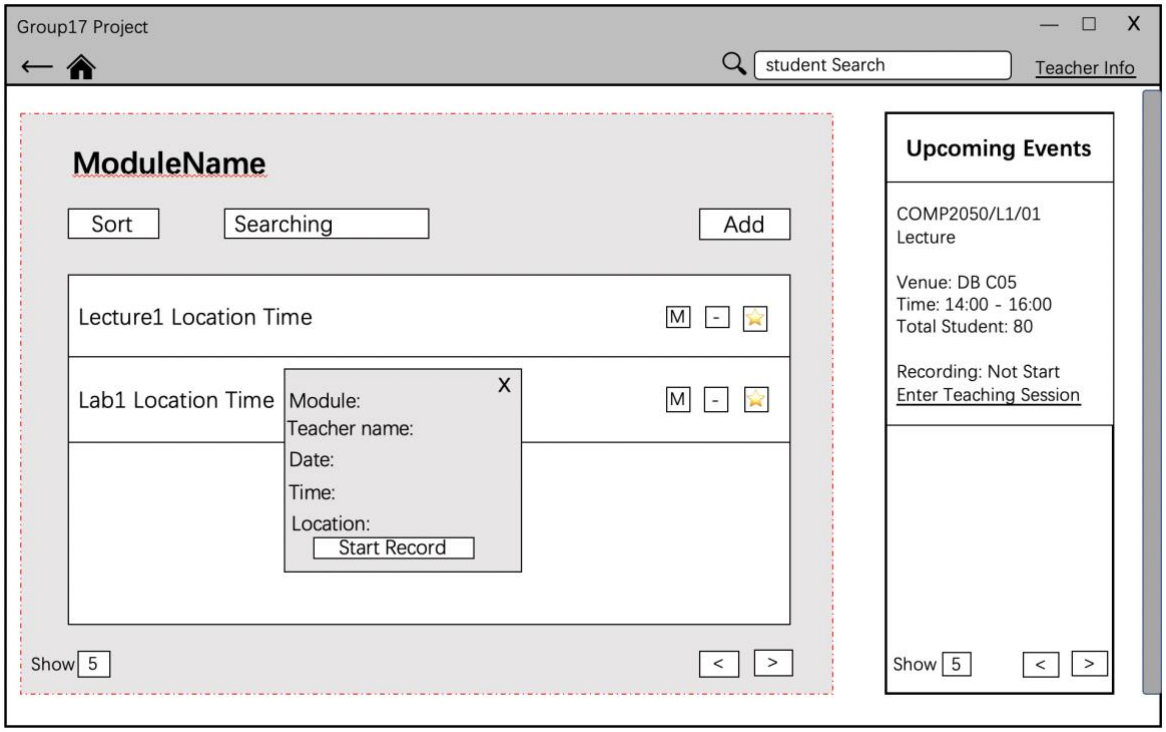


If the teacher clicks the name of past session, the application will show the

attendance information of it and allow teacher to export the attendance sheet.

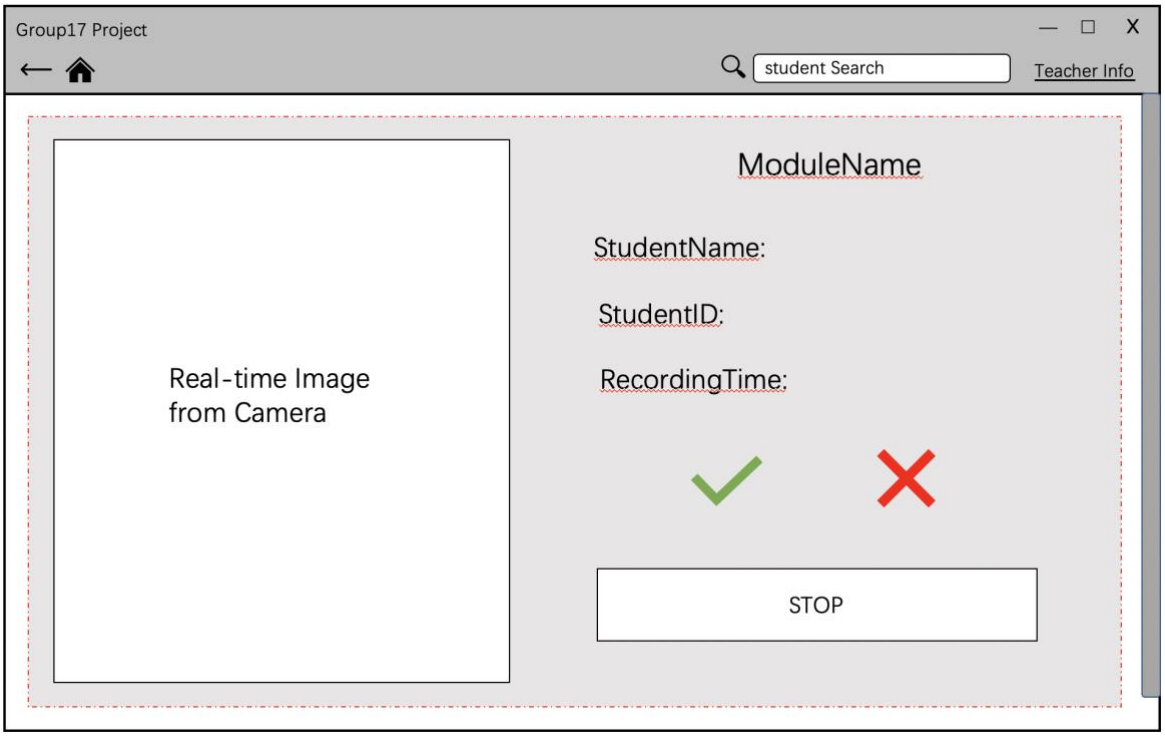


If the teacher clicks the name of future session, the application will show the box below to allow him/her to start record the attendance. If the chosen teaching session is not available for start, the button will be shadow and unavailable.



4) Recording page

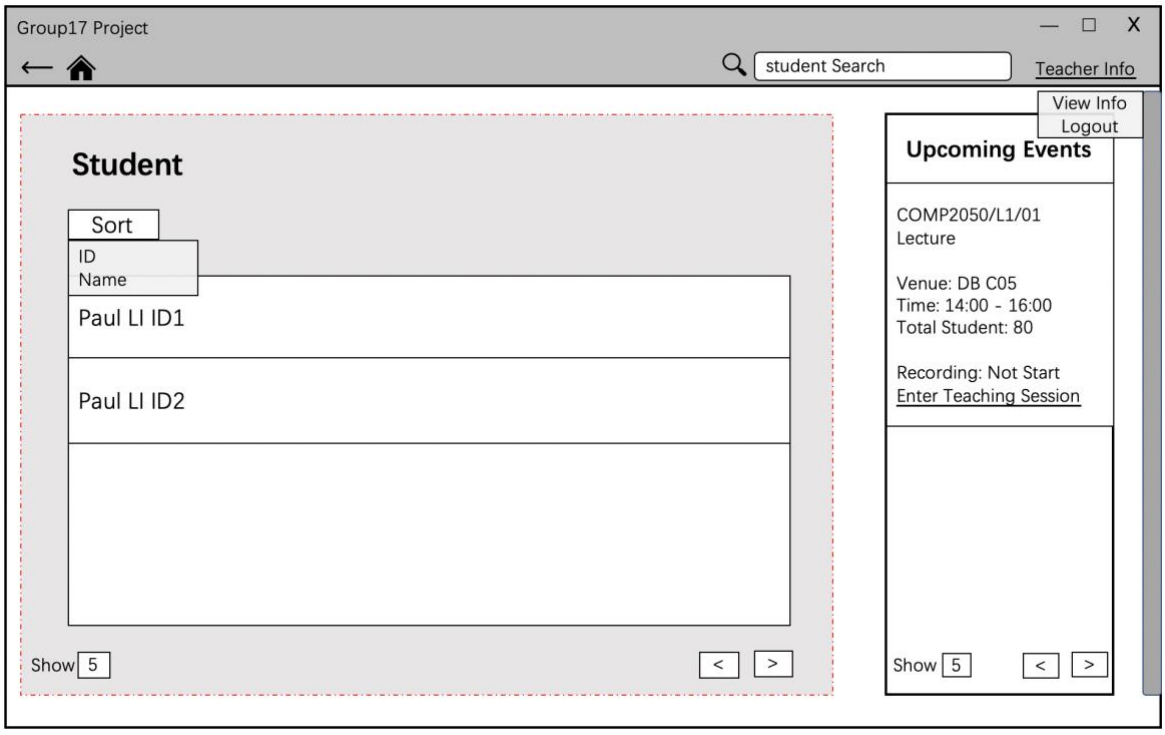
This page will be shown after the confirmation of start recording. The student Information will be updated and shown when successfully record one student.



if the record is stopped, it can be restarted as well

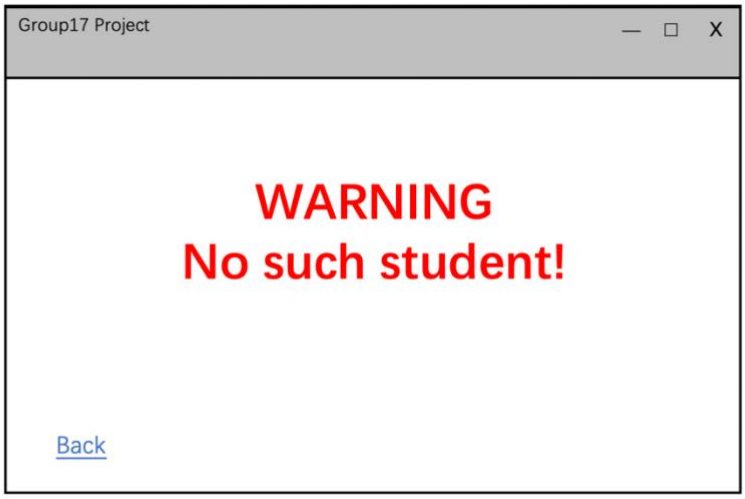
5) Search function

By searching student name or ID, teacher can get a summary of attendance information

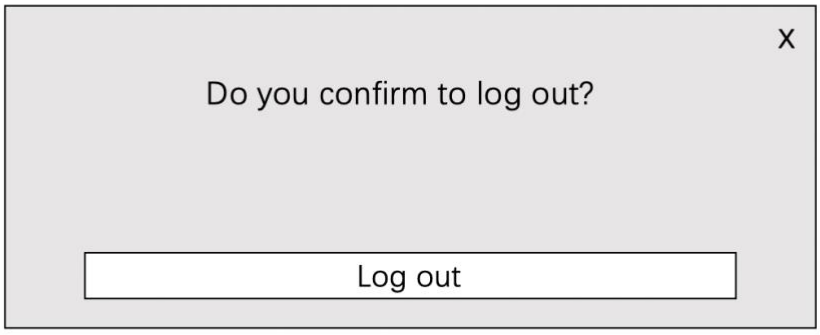


of one specific student. In addition, teacher can modify student Attendance state manually with comment. Teacher need to select student after searching, in case there are student with same name.

If there is no result, there will be a warning.

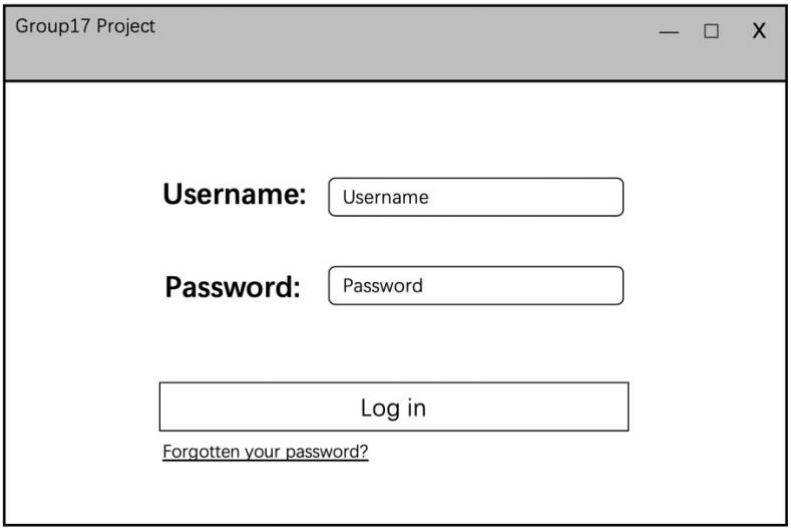


6) Log out



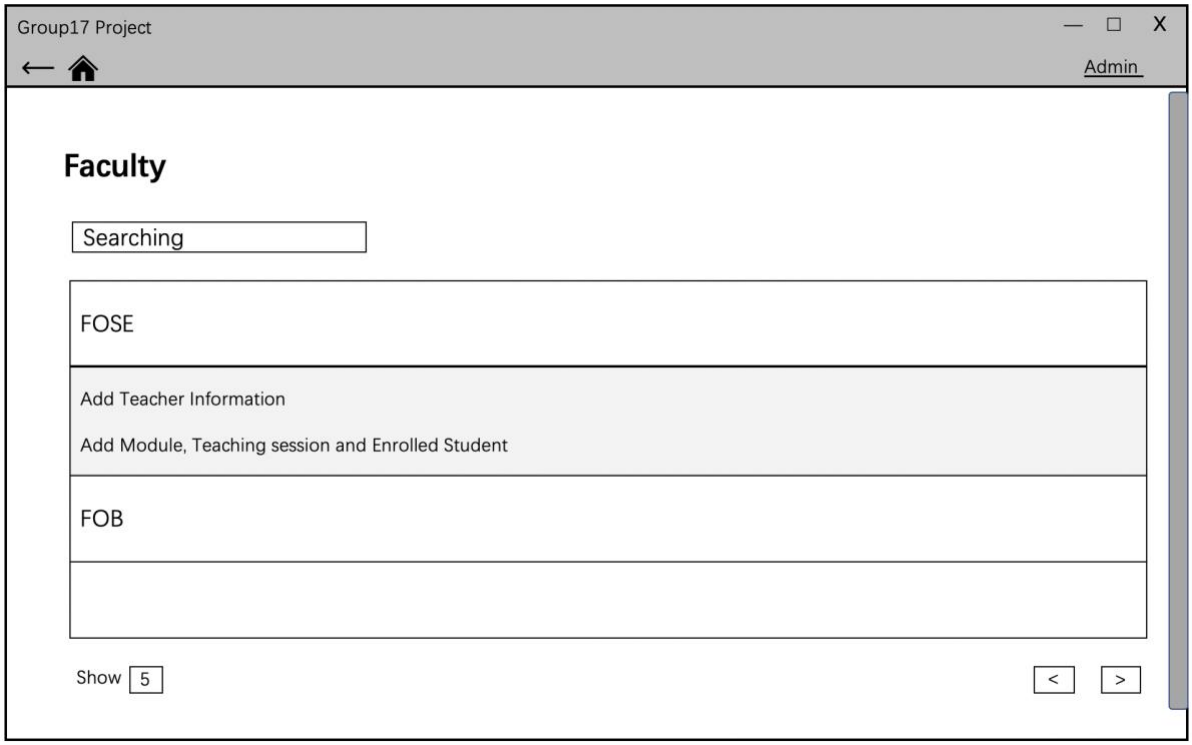
- **Admin:**

1) Login page



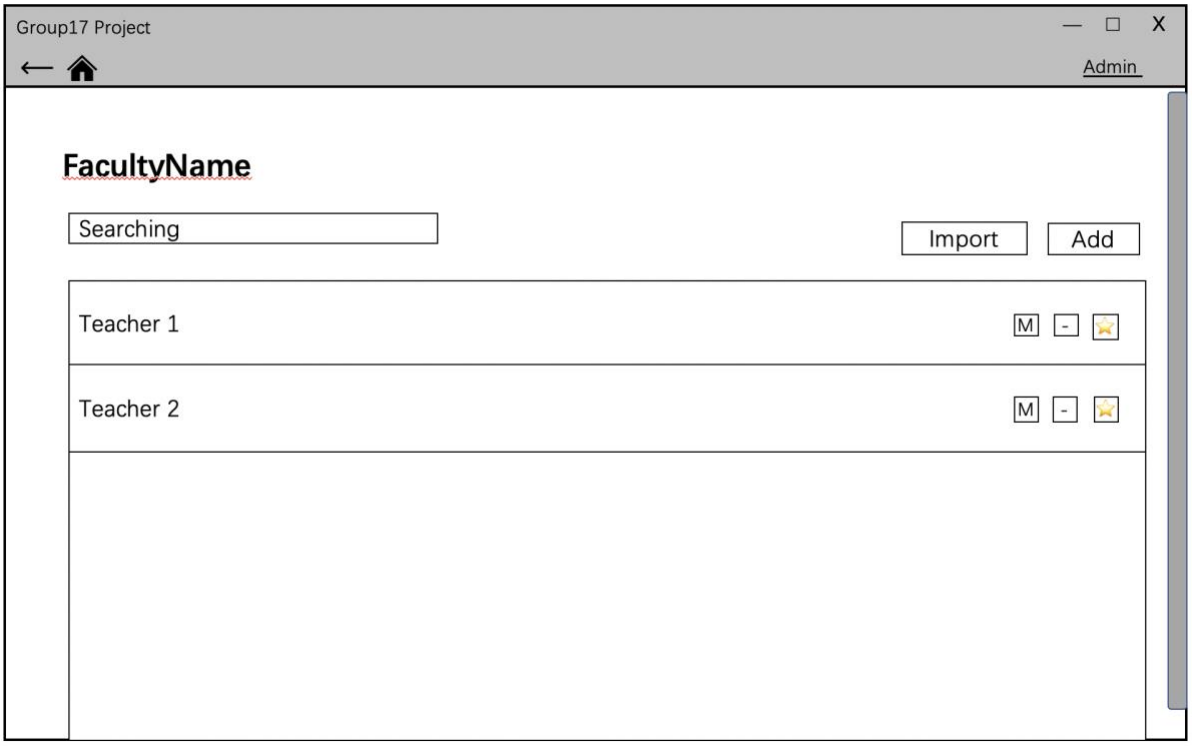
2) Faculty Page

This is the first page of admin. Each faculty has two functions, adding teacher and adding module, session and student.



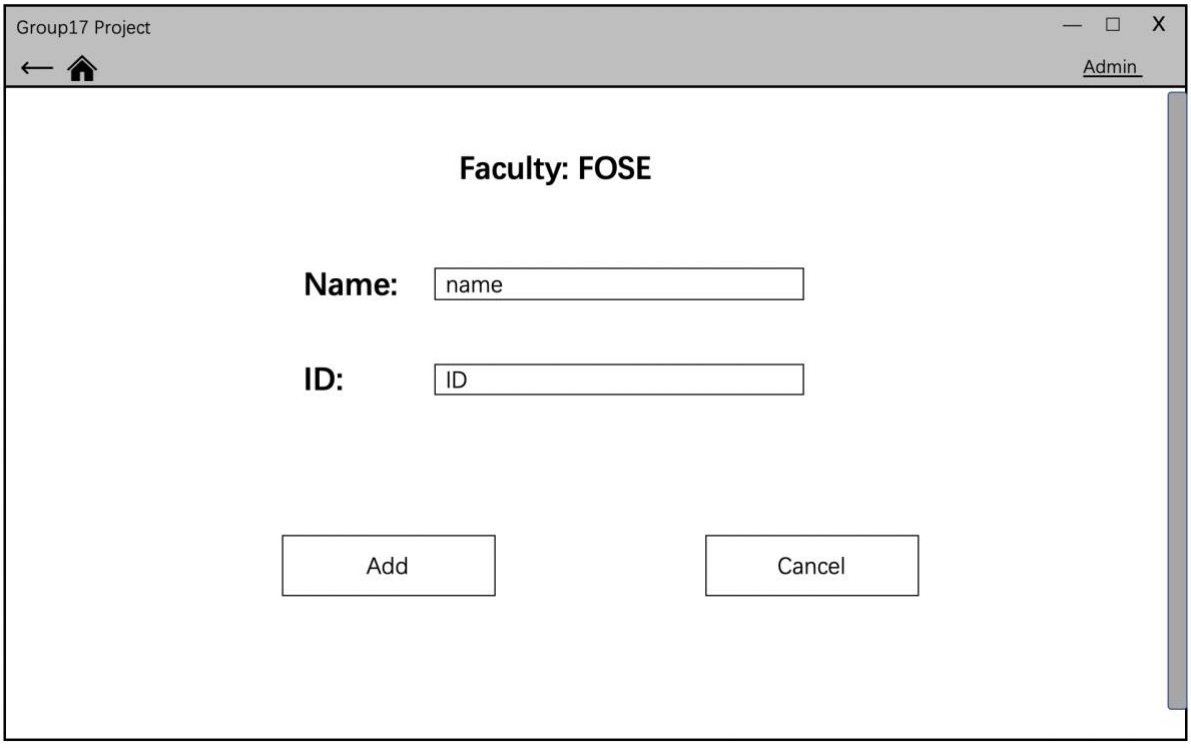
3) Add teacher page

This page will be shown after clicking “Add Teacher Information”. Admin can add one specific teacher or import all teacher together. Admin can also modify teacher information or delete teacher by clicking button “M” or “-”

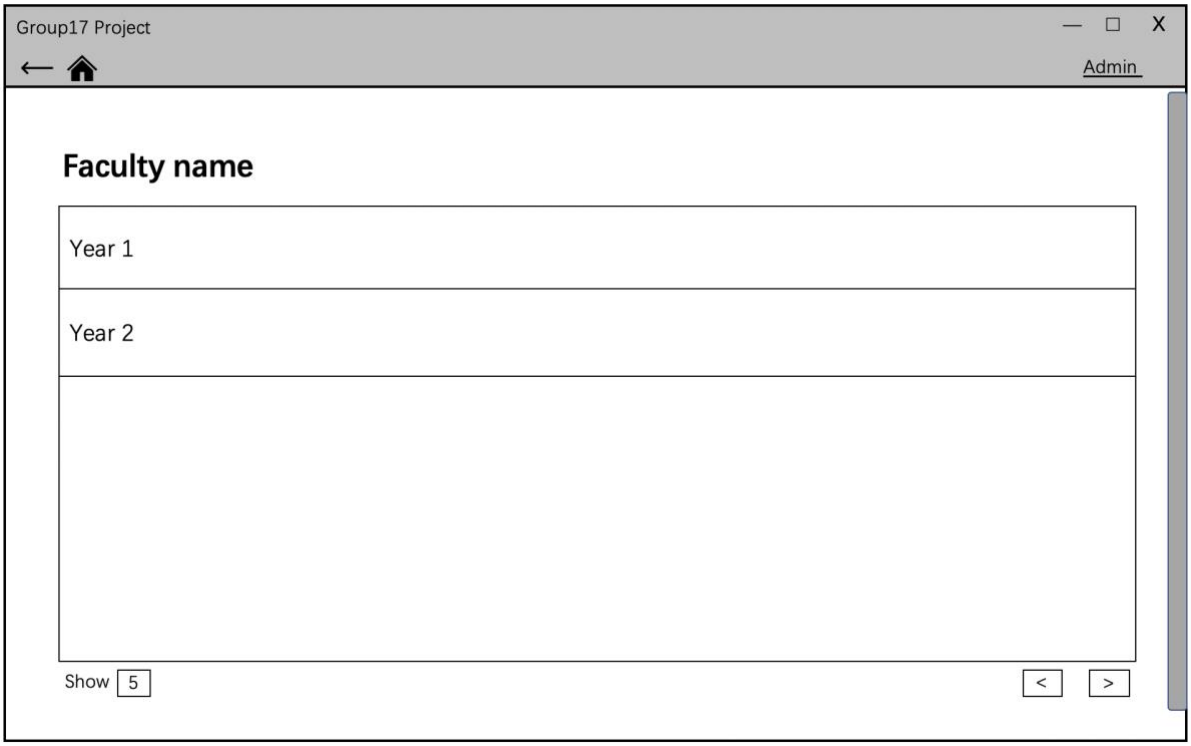


4) Add module

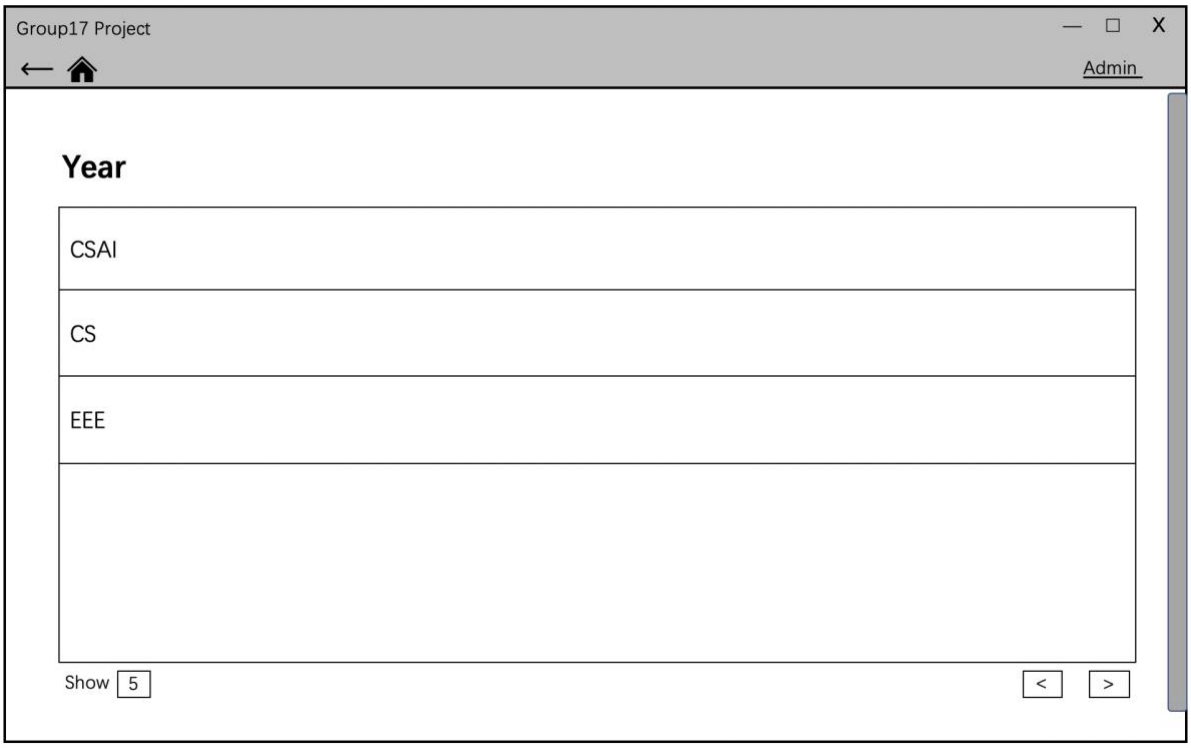
This page will be the first page shown after clicking “Add Module, Teaching session and Enrolled Student”.



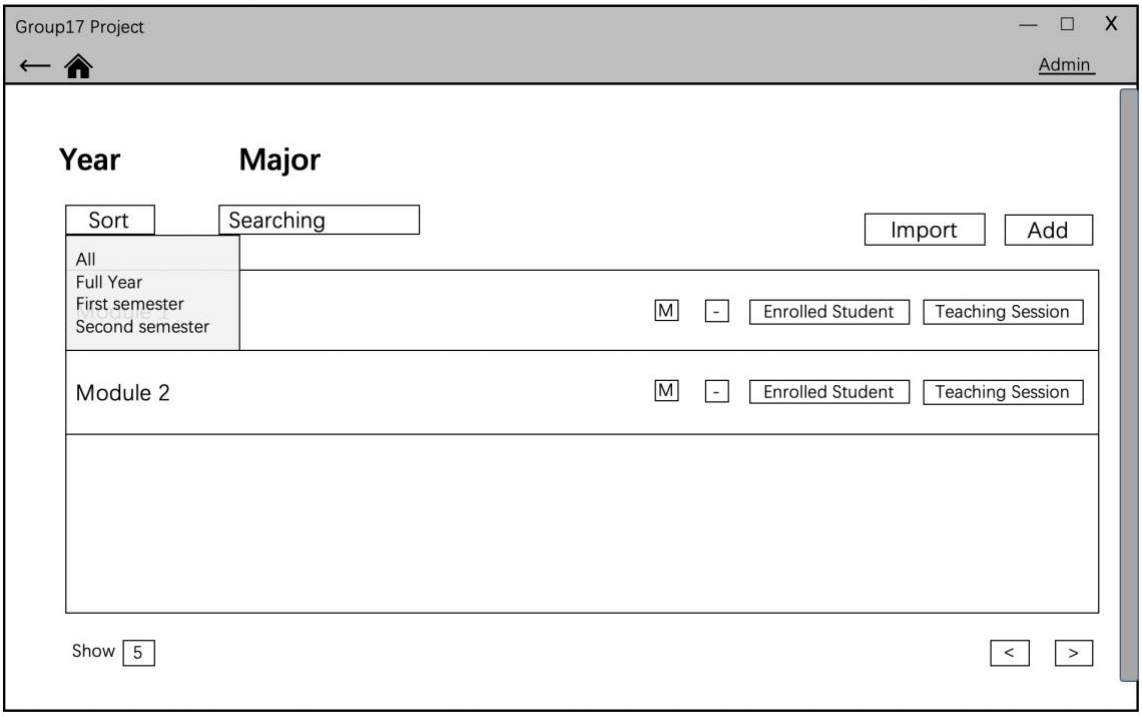
This page will allow admin to select which year to add module.



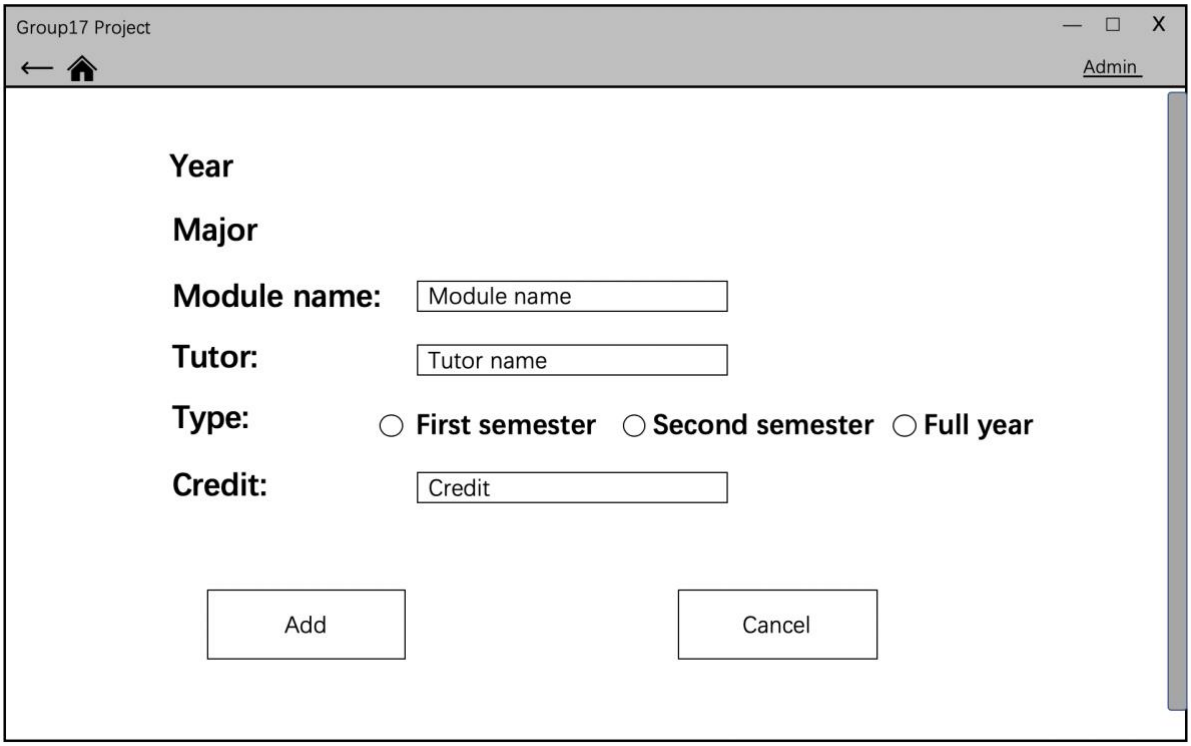
Then admin need to choose which major the module belong to.



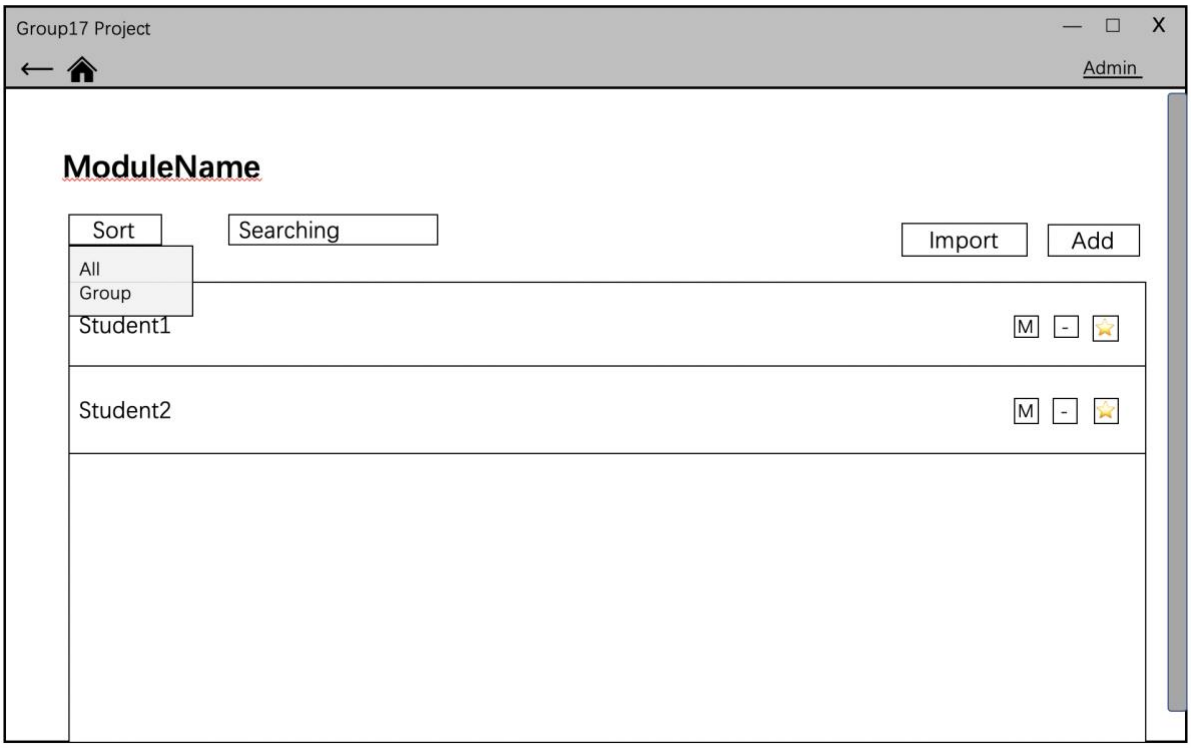
Then admin can add module, enrolled student and teaching session. In addition, module can also be deleted or modified through button "-" or "M".

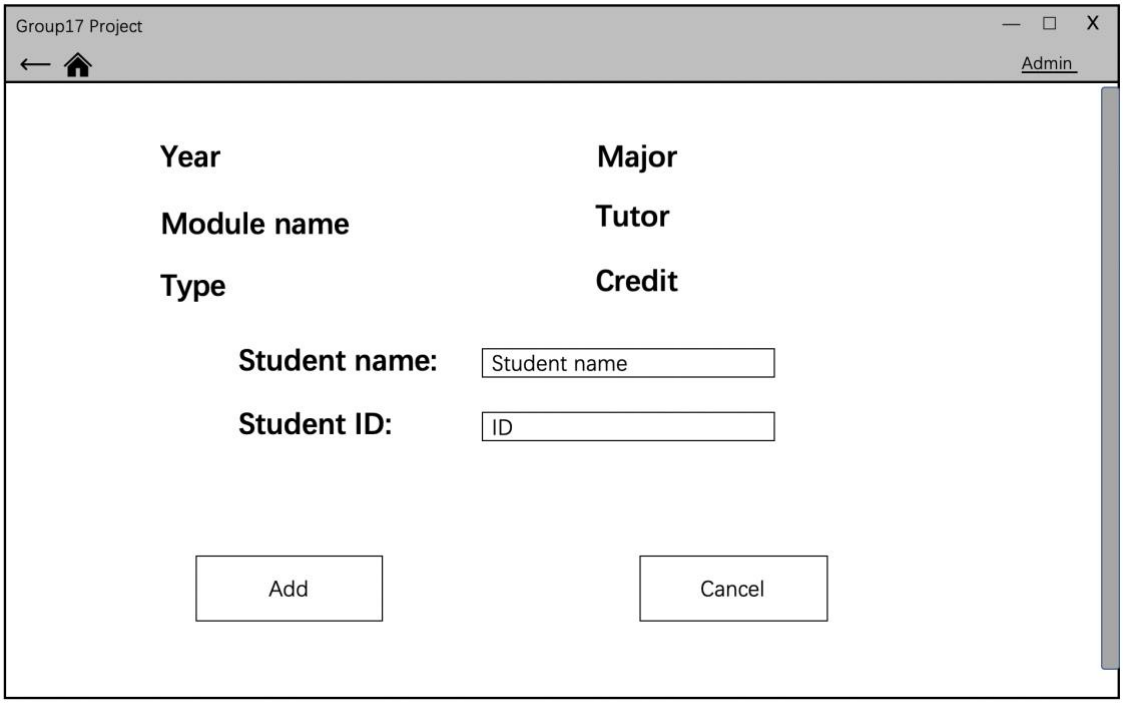


By clicking “Add” button, admin is able to add module under specific year and major.



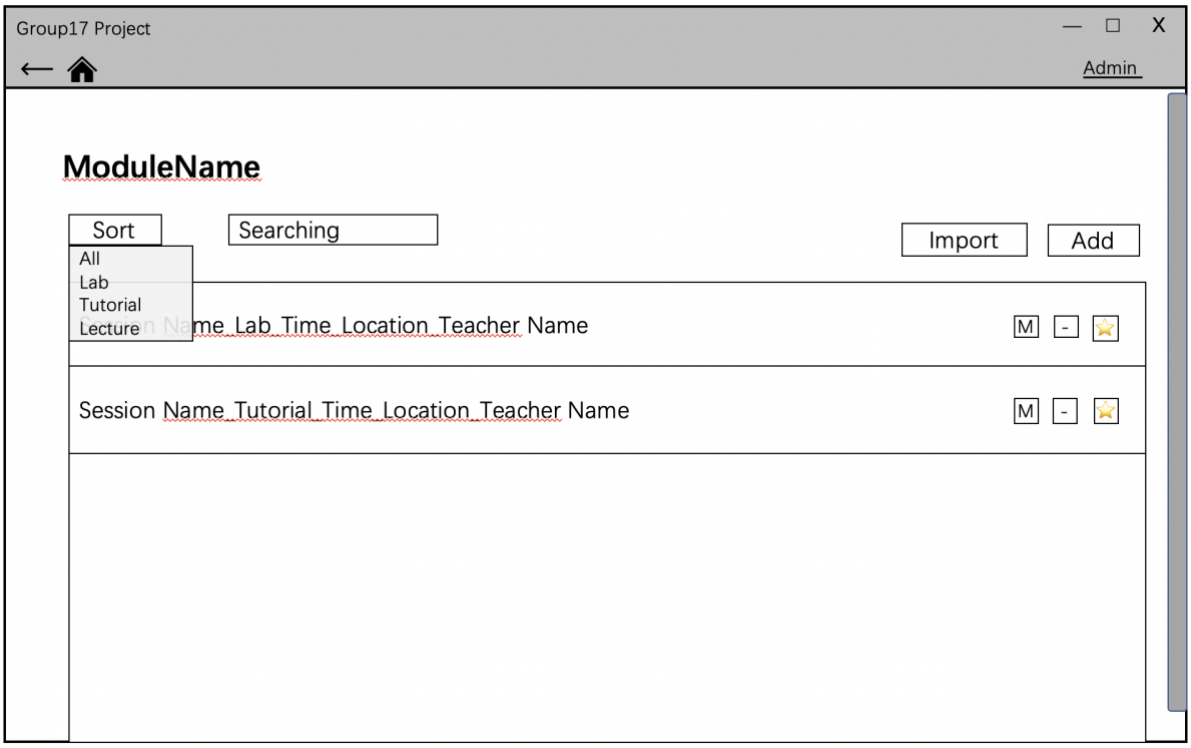
By clicking “Enrolled Student” button, the page will show the student list of the module. Admin is able to add student under specific year, major and module by clicking button “Add”. Admin can also modify and delete them in the same way mentioned before.

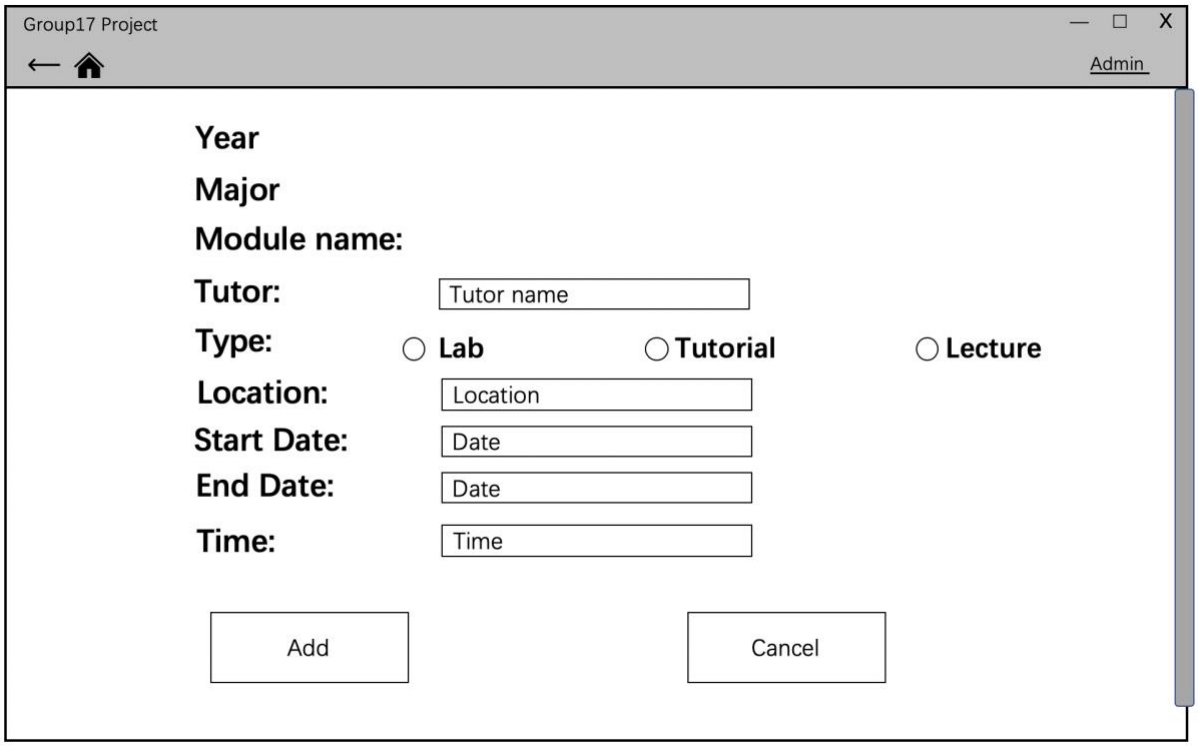




By clicking “Teaching Session” button, the page will show the teaching session list of the module.

Admin is able to add teaching session under specific year, major and module by clicking button “Add”. Admin can also modify and delete them in the same way mentioned before.





5. Key implementation decision

5.1 Programming language: Python

Python has the largest free third-party library for artificial intelligence implementation, especially for face recognition. Compared to other languages, it can support the development of the key function in this project – student recognition better. Meanwhile, Python is compatible with our chosen programming paradigm: Object-Oriented Programming. Hence the main developing programming language is Python.

5.2 Operating system: cross-platform

For Python is a cross-platform programming language and this project will provide an executable file for software installing, it supports the implementation on different platforms.

5.3 Hardware:

1) **Network:** Campus network. Also, the servers and user desktops running this software should be able to access the campus network to set up service.
2) **User computer:** A computer equipped with a camera, with a minimum of 100MB available space on disk for software implementation.
3) **Remote server:** One server in Artificial Intelligence Lab, which can set up MySQL database. It should provide a minimum of 2GB available space on disk for software implementation, larger space is recommended.

5.4 Software:

All the developing tools support cross-platform developing. Most of tools are open source and can be used for this non-profitable project according to their licenses. If this project will be implemented for business use in the future, more details about license will be checked.

1) Project management: Microsoft Office Word, Microsoft Office Excel

Word is used for normal text documenting such as meeting minutes. Excel provides easy implementation of project scheduling diagram such as Gant chart.

2) Resource control and document management: Git and Github

Git supports distributed version control. It enables the group members to share resources and work on a project simultaneously. It takes good control of the work merging and version recording.

Project Team Github url: https://github.com/Yiming-Li666/GRP-P19

3) Reference management software: EndNote

EndNote provides interfaces with the sites of academic resources and document editor. It enables the developing group to manage references and generate documentation with high efficiency.

4) UML drawing software: Papyrus

Papyrus provides full functions for UML diagram design, which can be used for project analysis.

5) Editor: Notepad++ (Windows Only)

Easy to convert text from Linux (MacOS) to Windows and vice-versa.

6) Development environment: Anaconda

Anaconda provides the management of multiple languages developing environments. Here is used for building the environment of python.

7) Integrated Development Environment (IDE): Pycharm

Pycharm provides strong support for development in Python with functions such as refactoring, integrated test and the using of external tools.

8) Database: MySQL

MySQL is a classic database management system, and our developers have rich experience with it.

9) GUI design: PowerPoint, Photoshop

PowerPoint provides enough support for the wireframe design of UI. Photoshop enables the designers to achieve the ideal visual effects.

10) GUI implementation: PyQt5, Qt

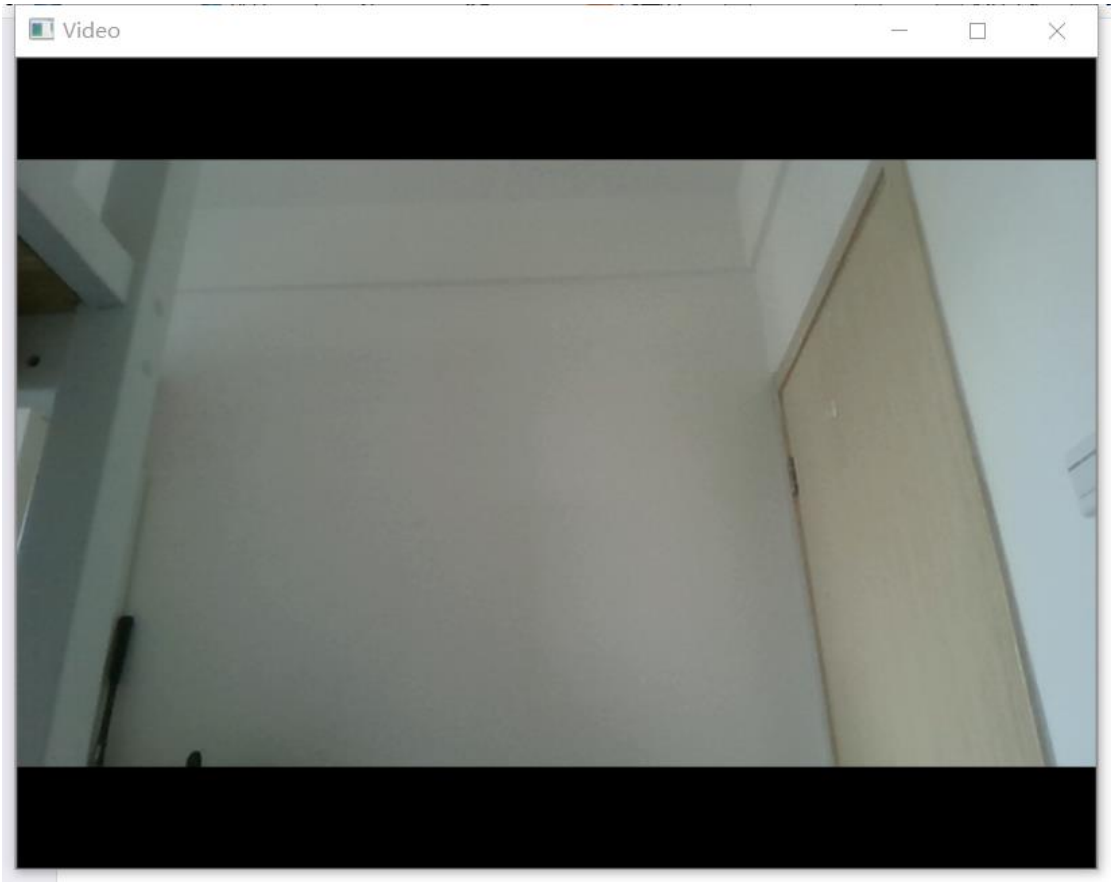
Qt is one of the main-stream GUI libraries, it supports cross-platform development, is of rich resources, and is easy to implement. It provides graphic interface design tool – Qt Designer.

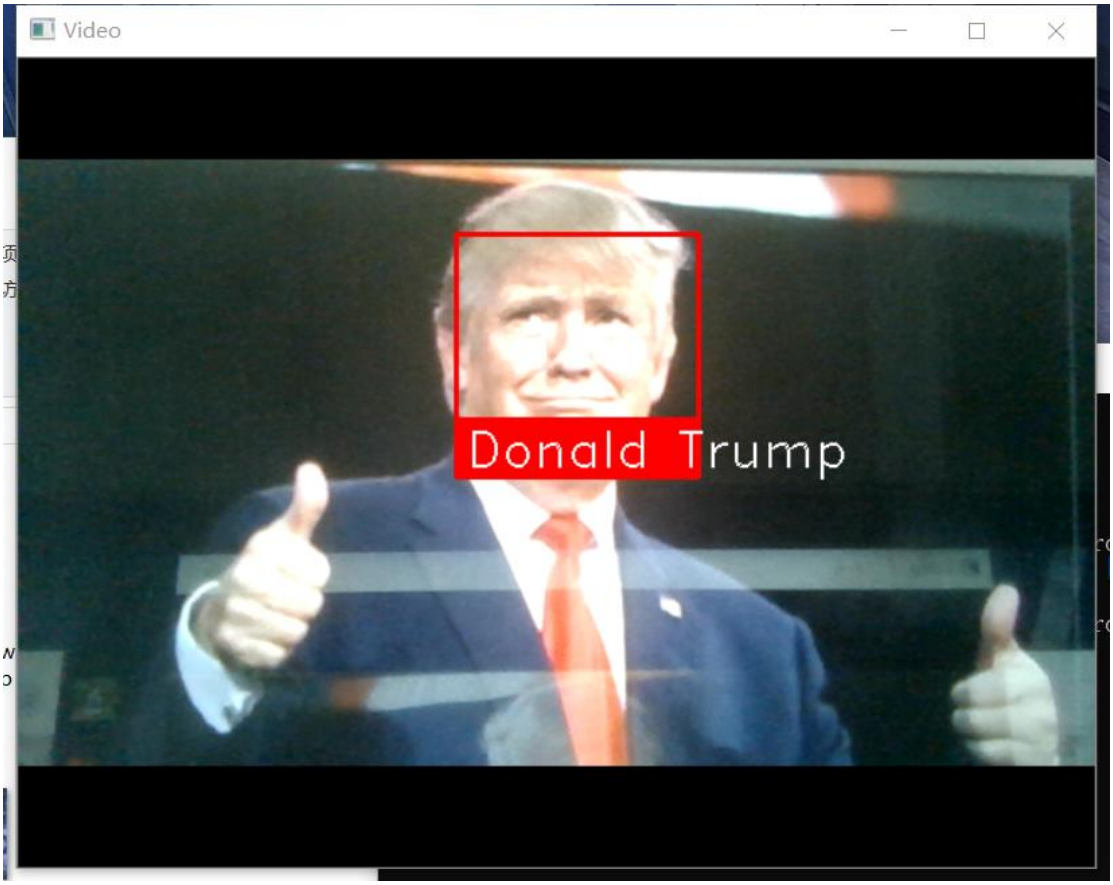
PyQt5 enables the developers to use APIs in Qt with Python. It provides collections of modules which can fully support our GUI development. For example, it enables multimedia control, Bluetooth device connect, unit testing and better visual effects.

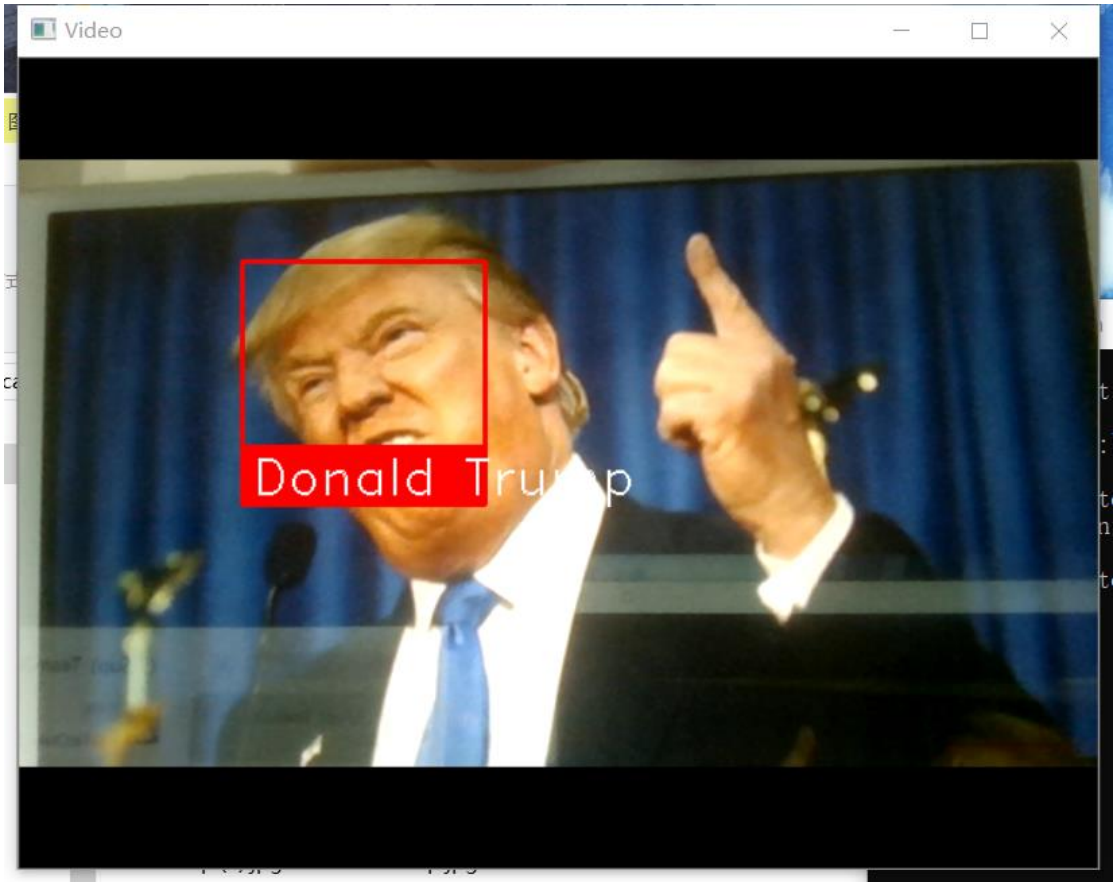
6. Result of any initial implementation

The open source third party face recognition algorithm is implemented and tested on Windows. If photos and the ID or name of the photo are given, the program will take three to four seconds to study the photo and store all the face information as a 128 * 128 matrix.

When starting recording, the camera image will be shown on the screen. Once the face is recognized by the system, there will be a red frame on your face and your name will be shown on the top of the frame.


Video


Video
Donald Trump



7. Discussion of any problems encountered

7.1 System analysis

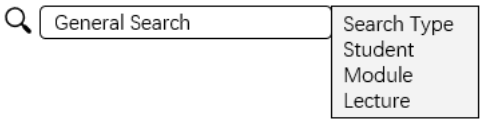
When we design our app, OO Design Principles is not been used.

There are no teacher, student and module classes. The app will always link to database. However, the cost is too much and it is hard for later maintenance. Hence, teacher, student and module classes are added and database is linked inside these classes.

7.2 Function design

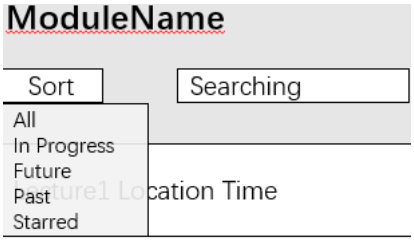
1) Search function

There are 2 major part of our UI, one for teacher, the other for Admin. In teacher part, there is a search method that students can be searched by their name and id. It confused us where the function should be put. As a result, a general search function is designed, and users can choose what kind of data they want to search.



However, this general search is not needed because teachers can search what they

need in the sub-pages. For example, when users attached to some specific module, there is a search function that they can search lectures.



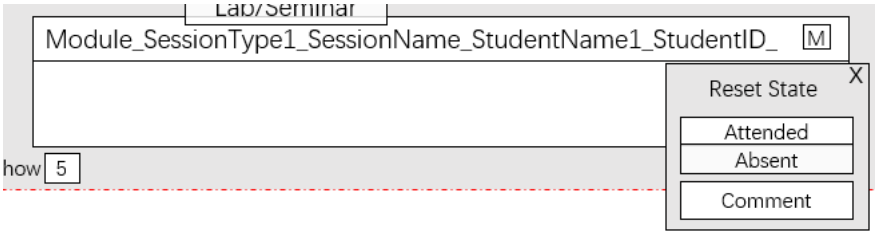
As a result, the general search is canceled, and student search function is been added at the top of UI. Teachers can search students whenever they want.

a. Students can only be found by their ID when first time we design this function. However, this is unreasonable because students should be able to be found by their names as well. This error is already modified.
b. There should be a warning when teacher there are no searching result.

2) Attendance recording function
The teacher should be able to stop and restart the record function. A restart button is added. The related function is changed as well.
3) Reused functions
Although there are different users, the search function and view function they used is actually the same. In the end a public controller class is designed, and it stores all the functions that can be both used by teachers and admins.

7.3 UI design

1) There are 2 plans for Admin UI. The second one is too hard for us to achieve although it is more convenience to use. As a result, the first one is used.
2) UI location of function
When one specific student is gotten by teacher, teacher can modify some information of this student such as attendance information. A new page is added to show all student information and teacher can change them and add comments there. However, this is inconvenience because there is one additional function in the new page. So, this function is added at original page.



7.4 Management issue

1) Delay
Time plan is set to distribute the whole program into two semesters with a proper working load every week. However, there exists delay of the work, which means the

work has not been finished within the decided time.

In this case, deadline is abided and there will be a punishment if any group member exceeded his/her time to finish the task.

2) Argument

Arguments always exists while making decisions. There has been one case happened when deciding which third party to use.

In this case, communication is needed within the group members and team leader should do this work. A rule is set that if anyone has any disagreements, he/she needs to list the advantage of his idea or the disadvantage of the current one.

3) Assignment

UI and code implementation are two main parts in this project. As the development tools of them have difference, we decided to separate our team into two groups, one for UI and one for functionality implementation. In this way, we can do both UI and functionality implementation simultaneously, which is more efficiency.

8. Time plan

Date	10/21	10/28	11/04	11/11	11/18	11/25	12/2	12/9	12/12
Tasks\Week Index	1	2	3	4	5	6	7	8	9
1. Preparation									
1.1 Fundamental project analysis and information searching									
1.2 Set project plan, draw use case diagram and class diagram									
1.3 Establish project website									
2. Requirement specification and risk analysis									
2.1 Feasibility study and publish feasibility report									
2.2 Requirements elicitation and analysis, develop system models									
2.3 Requirement definition, publish user and system requirements									
2.4 Requirement validation, publish requirements document									
2.5 Risk analysis									
2.6 Publish customer test plan									
3. System and software design									
3.1 Architecture design									
3.2 Database design: admin/teacher/module/student information, attendance, log-in									
3.3 Interface design									
3.3.1Internal interface: UI interface and software-interface									
3.3.2 External interface: back-end interface, web									
3.4 Component design, draw component diagram									
3.5 Publish sub-system integration test plan									
3.6 Publish system integration test plan									
4. System prototype development and unit testing									
4.1 Database development									
4.2 UI development									
4.3 Database management									
4.4 Face-recognition Attendance System									
4.5 Database-software and server interface development									
4.6 Back-end interface development									
4.7 Set test plan and test									
5. Publish Interim Report									

Time	12/12	12/19	12/26	1/02	1/09	1/16	1/23	1/30	2/06
Tasks\Week Index	9	10	11	12	13	14	15	16	17
6. System improvement and unit testing									
6.1 UI improvement									
6.2 Database management improvement									
6.3 Face-recognition Attendance System improvement									
6.4 Database-software and server interface improvement									
6.5 Back-end interface improvement									
6.6 Information visualization									
6.7 Set test plan and test									
7. Improvement and validation									
7.1 Improve UI design									
7.2 Validation									

Time	2/13	2/20	2/27	3/05	3/12	3/19	3/26	4/02	4/09
Tasks\Week Index	18	19	20	21	22	23	24	25	26
8. Integration and system test									
8.1 Sub-system integration test									
8.2 System integration test									
8.3 Customer test									
8.4 Publish test documentation									
9. Operation and maintenance									
10. Document and report									
10.1 Publish final report									
10.2 Publish user handbook									
10.3 Document generation									

Reference

1. DingTalk, https://www.dingtalk.com/en , Available at:2019/12/1
2. Jibble, http://www.jibble.io/ , Available at: 2019/12/11

Appendix I – formal minutes

Meeting Minutes of Group 17

Meeting ID: 1

Time: 14:00 – 14:30, 2019/10/17

Attended: Lu Zheng, LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: AI Lab

Minutes Taker: WANG Boya

1. Meeting Time

Time: Every week, random time (online chat group)

2. Set-up Advice

1) Project requirement:
   - Focus on software development rather than image recognition
   - CV accuracy: make use of developed model (refer to source code on Github, etc.)
   - Learn from similar developed software
2) Hardware
   - Camera Purchase: Not necessary, TBC
   - Laptop: make use of AI lab
3) Software
   - Open source library (but noted in the document)

4) Programming language

   - To use Python (good for various OS, easier GUN development)

5) Database

   - Data generation: take picture of members or friends (compare 1-2 pictures with user)
   - Functions: adding, deleting, etc.

Meeting Minutes of Group17

Meeting ID: formal meeting 2

Time: 15:00 – 15:30, 2019/10/23

Attended: LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: PMB 426

Minutes Taker: ZHU Hongyi

1. Set-up Advice

1) Check our previous work

   Team website, use case diagram, time plan

2) Further development
   a) Add information about team members on website
   b) Submit project plan to the website
   c) Make use case diagram more specified (what users can do)
      i. Start/finish recording by teacher
      ii. Set Module time, place
      iii. Add/ delete/ update student information
3) Add class diagram, sequence diagram, component diagram
4) Rewrite time plan
   a) Add system design (several components, class, relation)

b) Update Gantt chart, development cycles

c) Refer to the waterfall model or something similar (each step, how much time)

5) Do work according the time plan, follow software procedure

6) Discuss interim report, prototype

7) Separate into 2 class, one for Camera function, one for UI design (screen, button…) and how to do these in program

8) New meeting time

Meeting Minutes of Group17

Meeting ID: formal meeting 3

Time: 12:10 – 13:30, 2019/10/28

Attended: Zheng Lu, LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: PMB 426

Minutes Taker: LI Mingchen

1. Review Class Diagram

(1) Admin should do the "search" operation as well

(2) Since there is "search" method, there should be "show" or "view" method to show the search result.

(3) UI class required:

a. Module display window: Search Button, Return button, etc.

b. Detector window: Start detecting Button, End Button, Immediate attendance ratio, etc.

c. Etc.

(4) Control class (admin/teacher method class) required:

a. Activity of teacher/admin should be separated into a class from teacher/admin class

b. Teacher and admin have individual control class

(5) Face recognizer class required:

   a. Face recognize function

   b. Real-time detecting function

(6) The method under specific class, which must be individual

(7) Redo!!! class diagram must be worked for one week.

2. Review Use case Diagram

(1) Can use Override.

(2) Not need so detailed, put some similar function together.

3. Review Timeplan

(1) Not need so detailed, put some small plan together.

(2) Separate the Prototype and final release clearly (UI development), one is UI development, the other is Enhance UI.

(3) Take care of Integration test.

Meeting Minutes of Group17

Meeting ID: formal meeting 4

Time: 18:00 – 20:00, 2019/11/07

Attended: LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: Zhiya

Minutes Taker: Guohao YU

1. **Problem about class diagram**
    a. Add method in faceinformation
    b. Add time in attendance
    c. Rethink the relationship between module and group
    d. Unlink Student and Module
    e. Add teachingSession in Module
    f. Do we need to put picture information in Attendance?
    g. Add relationship between FaceDetector and Student
    h. Add relationship between teacher and teacherController
    i. Add user information in PublicController
    j. Add new class to be extended by Teacher and Admin
    k. Think the relationship between Controller and UI
    l. Decide if the Student will connect to SQL or local TXT
2. **UI**

a. Design UI first

b. Search for good UI libraries

c. UI is more important than functions in prototype

3. **Implements**

a. Better to use mysql

Meeting Minutes of Group17

Meeting ID: formal meeting 5

Time: 17:30 – 18:45, 2019/11/15

Attended: LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: PMB 432

Minutes Taker: Yiming LI

1. **Problem about UI design**
    a. Delete selection of type (teacher/Admin) in log in page
    b. Remove the delete lecture button of teacher page
    c. After stop, there should be a resume to continue recording
    d. General search needs to show everything correctly
    e. searching for student (in case two students have the same name) should always first return a list of students
    f. Need to be easy for teacher to change the student's attendance status
    g. Add command for changing status
    h. Start a session when time is wrong should have a warning
    i. Add time and attendance rate on the record page
    j. If cannot be recognized, should have a way to record
    k. Have a summary of the attendance of the session/module/student for teacher to export

2. **Class diagam**
    a. Link people class to public controller class
    b. Rename public controller to general controller
    c. Rename people class to staff class

3. **Sequence diagram**
    a. Video
    b. Searching

4. **Market research**
    a. Ask people around us
    b. Compare with exist application

Meeting Minutes of Group17

Meeting ID: formal meeting 6

Time: 17:30 – 18:00, 2019/11/21

Attended: LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: PMB 432

Minutes Taker: Mingchen LI

1. **Quick View Admin UI**
    a. Admin should add teaching sessions initially.
    b. Able to add teaching session weekly, like every Wednesday morning 9:00 to 11:00.
    c. Once it's able to add something, it should provide importing function as well to increase efficiency.
    d. Admin is able to set teacher's password and reset it. Teachers are allowed to change their password as well. Or another mechanism can be used, using default password to let teacher log in at first time.
    e. If there is students who are absent more than three times, a warning will be alerted.
    f. Teachers belongs to faculty, add one more level "Faculty" between "Teacher" and "School".
    g. ONLY ONE admin!

2. **Quick Review Teacher UI**
    a. Generally, it's fine.
3. **GUI Development Tools Select**
    a. **Qtpython** is recommended.

Meeting Minutes of Group17

Meeting ID: Formal meeting 7

Time: 11:00 – 11:30, 2019/11/29

Attended: LU Zheng, LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: PMB 409

Minutes Taker: WANG Boya

1. Sequence Diagram

- Redesign sequence diagram
- Only the classes included in class design can be implemented in sequence diagram
- Keep code writing in mind when design UML

2. Interim Report

- About task updating: the updated design strategies in implementing process
- Include more solid content such as details in report

Meeting Minutes of Group17

Meeting ID: Formal meeting 8

Time: 18:00 – 18:30, 2019/12/11

Attended: LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi, LI Mingchen

Absence: None

Venue: PMB 4 floor

Minutes Taker: YU Guohao

Report:

1. Add reason of update. Why we do it
2. More market research example. Declare whether there is a lot of system.
3. The way of do the research. Who we interviewed? The question we ask.
4. The reason of why we choose the platform.
5. Double check whether bullet point is ok

Sequence diagram:

1. For sequence diagram, think which class do the job.

Appendix II – informal minutes

Meeting Minutes of Group 17

Meeting ID: informal meeting 1

Time: 13:00 – 13:30, 2019/10/18

Attended: LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: PB hall

Minutes Taker: Li Yiming

1. Plan for next week

1) Set up time plan for the project - Yiming and Boya

2) Draw use case diagram – Hongyi

```
core functionality {
    recognize and identify the faces of the students enrolled in a specific module.
    recording time of arrival of the students given the time schedule once the face of the student is
    identified.
    statistical data such as how many students have attended for the current session and who are missing.
    how many classes are missed by a specific student.
    // addition
    student information database{
        Face
        Student id
    }
    management log in, set lecture time, search data from particular lecture or student.
    adminstrator log in, set parametor to adjust accuracy.
}
```

- recognize the faces
- identify the faces
- record time of arrival
- database for statistical data
- … (other use case)

3) Build website – Guohao and Mingchen

- Project title, team members and supervisor

- Project introduction
- Meeting minutes involved by links

4) Learn programming language
   - Python (IDE, complier, file, format)

Meeting Minutes of Group1817

Meeting ID: informal meeting 2

Time: 13:00 – 13:30, 2019/10/23

Attended: LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: LI Mingchen

Venue: PMB 4 floor

Minutes Taker: ZHU Hongyi

1. Search Introduce to Software engineering coursework feedback
2. Identify actors and use cases.
3. Assign jobs for next week

    1) Update use case diagram – ZHU Hongyi

    2) User Requirement and System Requirement - LI Yiming, WANG Boya, LI Mingchen

    3) Update time plan - LI YiMing, WANG Boya

    4) Method for the system – ZHU Hongyi

    5) Class diagram - YU Guohao

    6) Update website - YU Guohao, LI Mingchen

Meeting Minutes of Group1817

Meeting ID: informal meeting 3

Time: 15:00 – 16:00, 2019/11/01

Attended: LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi, LI Mingchen

Absence: None

Venue: Building 23

Minutes Taker: LI Yiming

1. Identify all the class in the class diagram

   Add controller class and UI class

2. Unify all the method name in class diagram and method checklist

Meeting Minutes of Group17

Meeting ID: informal meeting 4

Time: 14:00 – 15:30, 2019/11/09

Attended: LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: Shizu

Minutes Taker: WANG Boya

1. **Check Requirement of Interim Report**

   Task: To confirm with supervisor whether the background information and noted researches should be done.

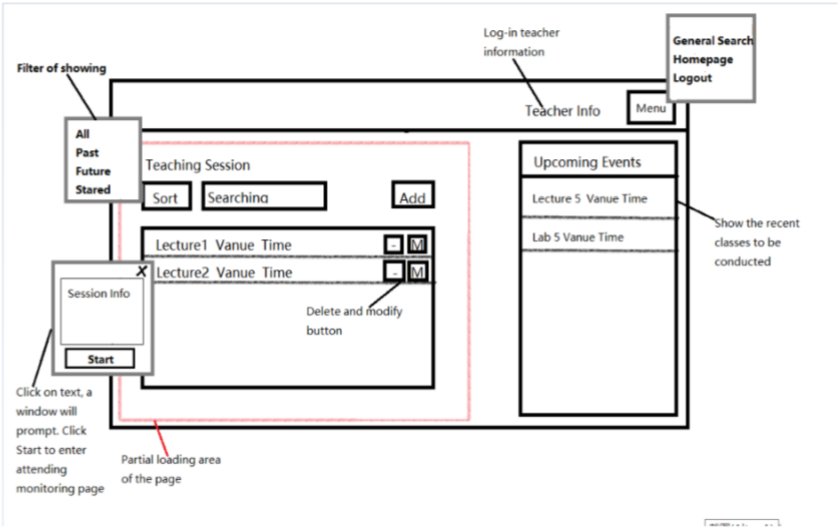
2. **Instance UI Design for Teacher**

1) Design and composing reference: Moodle

2) Design features:

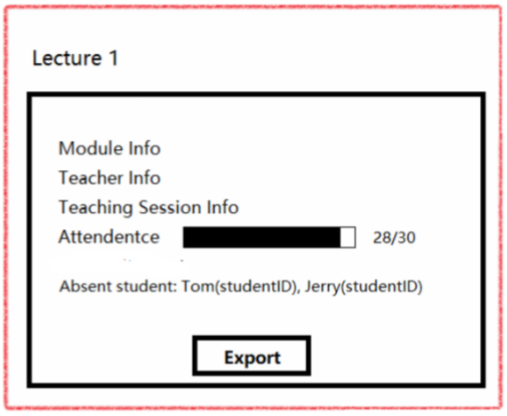
   - To use partial page refresh
   - To develop a general search that any input is available, and a partial search under located content to find target easier.

3) Design for teaching session:



4) Design for recorded session:

Click on unrecorded teaching session, it will prompt the above window.
If click on a recorded session, the patial loading area will refresh and show as below:



3. Task before Next Meeting:

1) Complete class diagram

2) Complete plane UI design for teacher class

Meeting Minutes of Group1817

Meeting ID: informal meeting 5

Time: 15:00 – 15:30, 2019/10/18

Attended: LI Mingchen, LI YiMing, WANG Boya, YU Guohao, ZHU Hongyi

Absence: None

Venue: PB 1 floor

Minutes Taker: ZHU Hongyi

1. UI design and assignment

1) Change Teachers UI
   Add select box in general search.
   Add warning information page, same name student page
   Add export function
   Add summary of students', lecture and module absence
   Add export function on module
   Add export page
   Add more information on student page
2) Change usecase diagram
   Delete login time
3) Design UI of Admin
   Import information function
4) Discuss interview question
5) Add sequence diagram at search and recognize function