COLUMBIA | ENGINEERING

The Fu Foundation School of Engineering and Applied Science

COMS 6998 Cloud Computing and Big Data Systems

CU LION NET

TONG WU tw2771

HUIYAN XING hx2302

XIAOTIAN GENG xg2358

JIAXIN MA jm5130

Abstract

In the project, a *CU Lion Net* system is built using vue as front end, python and lambda function as back end and DynamoDB as database. At the same time, API Gateway is used as a middle ware. Three main functions are implemented: Course evaluation, Discussion on specific topics and Time scheduler. Also, we add another part called ”User Profile” where user can upload their head portrait and their basic information.

Contents

1 Introduction

1.1 Motivation

Intuitively, what we want to build is a system that work as a forum for Columbia students. And what we need the most is information of different courses, like their contents and grading. That is what ”Course Evaluation” works for. Since master students are always struggling with looking for working positions. We decide to add a ”Discussion” part in which users can post information that they like to share. Since we want this system to facilitate students daily life, we decide to add a ”Time scheduler” in the form of calendar so that students can easily know what they should do during a specific time.

1.2 Problem Statement

According to 1.1, we implemented mainly three parts: Course Evaluation, Discussion, Time scheduler. For ”Course Evaluation”, we need to enable users to post information about classes which they had studied. So this means the system should first know what classes one particular user had taken. Then users could add information such as gpa, study methods, workload and comments.

We decided to add the ”User Profile” part to store this information, as well as user name, email address, linkedin link, github link and the classes they had taken.

For ”Discussion”, there are four different topics which we think would be most popular among students: NYC life, job hunting, house leasing, crime alert.

For ”Time scheduler”, it works as a calendar and users can add meetings of different time period.

2 Architecture

There are several clicks for the prototype and the architecture diagram of system is shown as figure 7.

2.1 Functionality Overview

Firstly, once you open this link, there will be a login interface, click on the blue button of login, then it will be directed to the main interface which is the interface of “group module”. Another bottom is register, click it to direct to the registration interface. Then click the button of sign up, it can go to the main interface.

For course evaluation, students can freely create a new course with the general information. For such a course, students can post comments in it, including the study methods, grading for last semester, workload and so on. With this information, students can know more about the course, receive helpful information for course selection plans.

As for topic discussion, four topics are provided, NYC life focuses on the trivial things in life related to delicious food and shopping. Job hunting focuses on chances for jobs, and house leasing is a platform for students to post sublet information, look for roommates and sell furniture.Another one is about the safety issue, the crimp map of NYC. Students can check the realtime safety status in New York.

Time scheduler is a simple tool to help generate an event schedule and remind students about the time arrangement and tasks waiting to be done.

2.2 Architecture design

For topic discussions, four functions are designed: create a new thread, reply to others' thread with comments, see the updated thread list and the comments respectively. Four lambda functions are written, and four APIs are used for

transmitting messages from the frontend.
Secondly, for course evaluations, the functions are: create a new course, write comments to the course, get the updated webpage for the evaluations. Similarly, three APIs are used.
Thirdly, we focus on writing event information and getting the schedule arrangement . Two APIs are designed here.
Finally, a complete profile is provided for each user, including the thread he/she posted and the courses he/she participated. Therefore, functions for getting a profile, getting courses and getting user photos are designed. Three APIs are for them.

3 APIs Design

There are 12 APIs in total:

1) User clicks the website link to access the frontend hosted on S3. The main page is displayed.

2) User chooses one topic and then click to open it: API 1 makes a GET call, which is validated by LF1. LF1 checks information in DB and return the page, redirect to this new page

3) User wants to read the details of thread: API 4 makes a GET call, which is validated by LF4. LF4 checks information in DB and return the page of details in thread, redirect to this new page

4) User wants to write some comments to the open thread: API 3 makes a POST call, which is validated by LF3. LF3 writes new information in DB and return the page of details in thread, redirect to this new page.

5) User wants to create a new topic: API 2 makes a POST call, which is validated by LF2. LF2 writes new information in DB and return the page, redirect to this new page.

6) User chooses to search a course code to access the course evaluation: API 6 makes a GET call, which is validated by LF6. LF6 searches information from DB, return the content of new page, redirect to this page.

7) User wants to create a new course: API 5 makes a POST call, which is validated by LF5. LF5 writes new information in DB and return the page, redirect to this new page.

8) User wants to write comments to this course evaluation: API 7 makes a POST call, which is validated by LF7. LF7 stores information to DB, return the content of new page, redirect to this page.

9) User wants to create and get to-do list: API 8 makes a GET/POST call, which is validated by LF8/LF9. LF9 stores information to DB, return the content of new page, redirect to this page. LF8 searches information from DB, return and redirect to the new page

10) User wants to get user profile: API 9 makes a GET call, which is validated by LF10. LF10 searches information from DB, return the content of new page, redirect to this page.

11) User wants to update user profile: API 10 makes a POST call, which is validated by LF11. LF11 writes new information in DB and return the page, redirect to this new page.

12) User wants to get course information the user had: API 11 makes a GET call, which is validated by LF12. LF12 searches information from DB, return the content of new page, redirect to this page.

13) User wants to update user head portrait: API 12 makes a PUT call to store photos in S3.

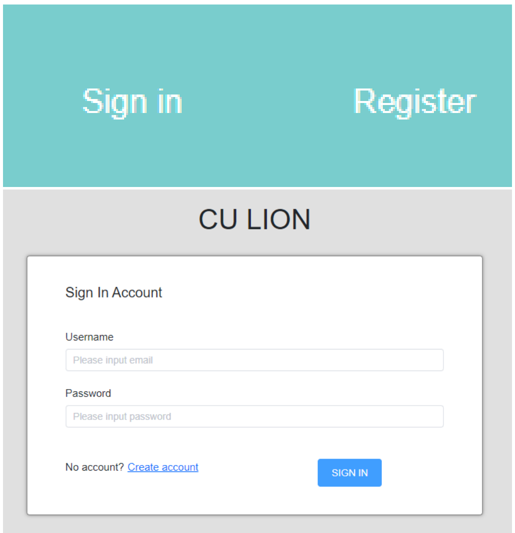
4 Data Model

There are five databases in total: topic, user profile, course, calendar and meeting.
Topic is used for storing thread and the comments for it. And it has three different tables to store different topics: NYC life, job hunting, house leasing. Course stores all the evaluations for each course. Calendar and meeting are used for time scheduler. User profile stores all the information about the users.
The structure of databases are shown in 7, 7, 7, 7.

5 Result

After multiple testing, we can make sure our system can work.
Cognito Register & Login

So for the user who is lignining in to our system for the first time, Cognito will help the user to register by taking the user's email address and sending a verification code. Then the user can sign in the system.



CU LION

Sign In Account

Username

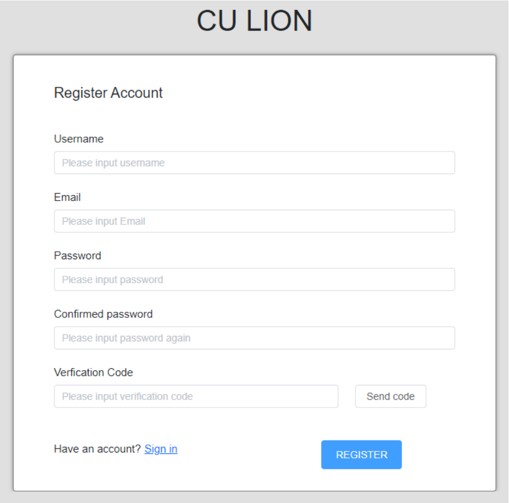
Please input email

Password

Please input password

No account? Create account

SIGN IN



CU LION

Register Account

Username

Please input username

Email

Please input Email

Password

Please input password

Confirmed password

Please input password again

Verfication Code

Please input verification code

Send code

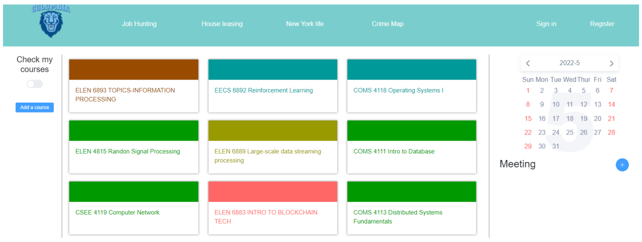
Have an account? Sign in

REGISTER

Fig. 5.1: Login register page

Main Page

In the main page, there are mainly two different parts.
The part at the top displays four different topics. The part in the middle shows all courses in our system. At the same time, user can choose to show all courses in the system or the courses in the user profile. The right part shows a calendar, if the user has signed in, the calendar would also display the agenda of this user. There is a lion in the upper left corner of every page, and by clicking this lion, we can go back to main page.



ELEN 6893 TOPICS-INFORMATION PROCESSING

EECS 6892 Reinforcement Learning

COMS 4118 Operating Systems I

ELEN 4815 Randon Signal Processing

ELEN 6889 Large-scale data streaming processing

COMS 4111 Intro to Database

CSEE 4119 Computer Network

ELEN 6883 INTRO TO BLOCKCHAIN TECH

COMS 4113 Distributed Systems Fundamentals

< 2022-5 >

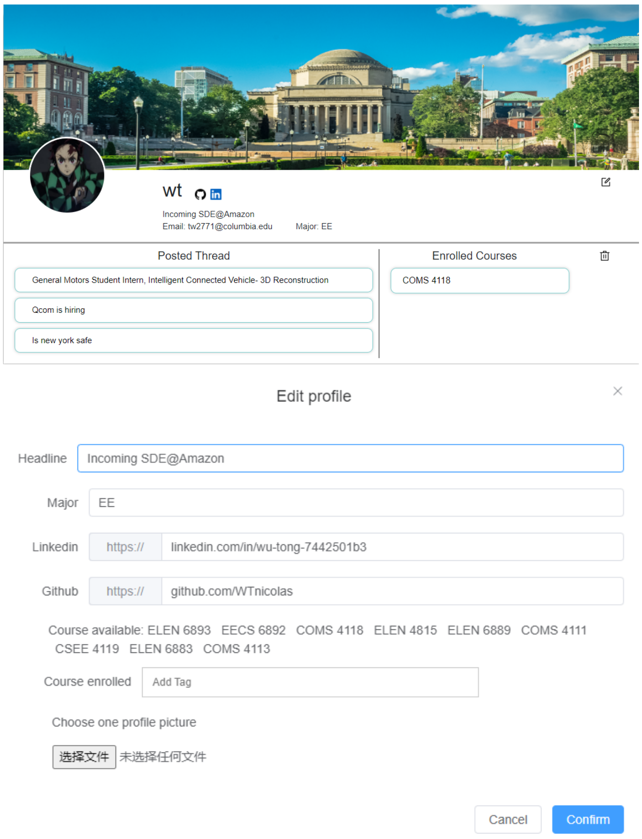
Sun	Mon	Tue	Wed	Thur	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Fig. 5.2: Login register page

User Profile

After signed in, a "User Profile" button would be displayed at the upper right corner. In User Profile page, the user can edit / update the profile by providing information such as, Head Line, Major, Linkedin, Girhub and courses enrolled (only those in the database). At the same time, user can also upload a photo as head portrait.

What's more, users can update the profile whenever they want.



wt

Incoming SDE@Amazon

Email: tw2771@columbia.edu Major: EE

Posted Thread

- General Motors Student Intern, Intelligent Connected Vehicle- 3D Reconstruction
- Qcom is hiring
- Is new york safe

Enrolled Courses

- COMS 4118

Edit profile

Headline: Incoming SDE@Amazon

Major: EE

Linkedin: https:// linkedin.com/in/wu-tong-7442501b3

Github: https:// github.com/WTnicolas

Course available: ELEN 6893 EECS 6892 COMS 4118 ELEN 4815 ELEN 6889 COMS 4111 CSEE 4119 ELEN 6883 COMS 4113

Course enrolled: Add Tag

Choose one profile picture

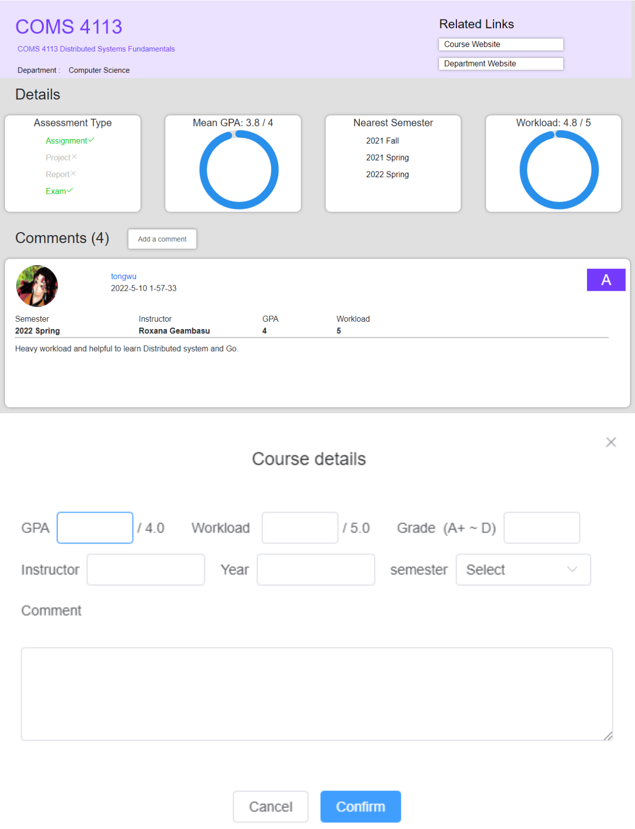
选择文件 未选择任何文件

Cancel Confirm

Fig. 5.3: Profile & profile edit

Course Page

By clicking the courses enrolled in user profile, user will be redirected to the specific evaluation page which shows information like, course code, department, course website, department website, inspection form, overall GPA, overall workload, nearest semester and comments. By clicking the website links, user can be redirected to those pages. By clicking "add comments" button, user can add comments of this course.



COMS 4113

COMS 4113 Distributed Systems Fundamentals

Department : Computer Science

Related Links

Course Website

Department Website

Details

Assessment Type: Assignment✓ Project✕ Report✕ Exam✓

Mean GPA: 3.8 / 4

Nearest Semester: 2021 Fall, 2021 Spring, 2022 Spring

Workload: 4.8 / 5

Comments (4) Add a comment

tongwu

2022-5-10 1-57-33

A

Semester	Instructor	GPA	Workload
2022 Spring	Roxana Geambasu	4	5

Heavy workload and helpful to learn Distributed system and Go.

Course details

GPA / 4.0 Workload / 5.0 Grade (A+ ~ D)

Instructor Year semester Select

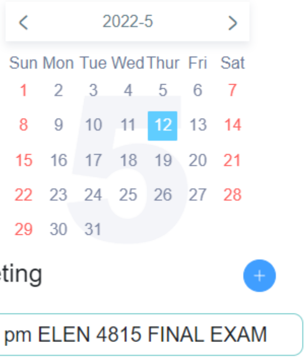
Comment

Cancel Confirm

Fig. 5.4: Course page & comment

Calendar

By clicking the particular date of the calendar and click the "+" button, user can add meetings or events on the calendar by providing information like, time, description and place. And the meetings will be shown once the particular date is selected.

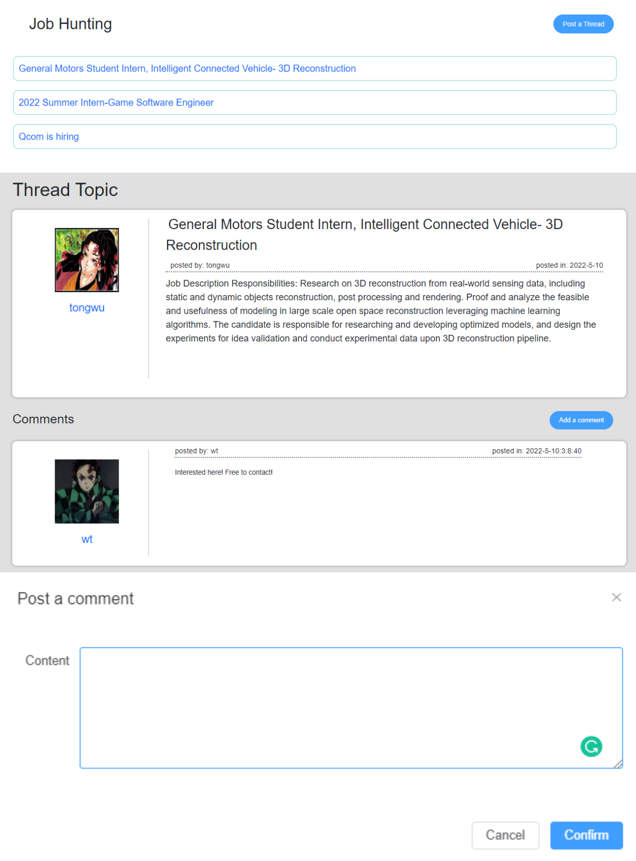


4:00 pm ELEN 4815 FINAL EXAM

Fig. 5.5: Calendar

Thread & Comment

There are four topics and three of them can be manipulated with the same method and let's take Job Hunting as an example. By clicking "Job Hunting" button, user will be redirected to the page of this topic and all threads of this topic would be displayed. By clicking "post thread" button, user can post a new thread by providing information like, subject and content. By clicking one thread, user could see the details of this thread, like, contents, comments, the exact time when a thread or comment is posted and the user who posted this thread or comment. User can add comments to this thread. And by clicking the username of the thread or comments, one could be redirected to the user profile of the owner of this thread or comment.
For NYC life and house leasing, the system works in the same way.
For crime map, by clicking this button, user will be redirected to "NYC Crime Map" website.



Job Hunting

General Motors Student Intern, Intelligent Connected Vehicle- 3D Reconstruction

2022 Summer Intern-Game Software Engineer

Qcom is hiring

Thread Topic

tongwu

General Motors Student Intern, Intelligent Connected Vehicle- 3D Reconstruction

posted by: tongwu posted in: 2022-5-10

Job Description Responsibilities: Research on 3D reconstruction from real-world sensing data, including static and dynamic objects reconstruction, post processing and rendering. Proof and analyze the feasible and usefulness of modeling in large scale open space reconstruction leveraging machine learning algorithms. The candidate is responsible for researching and developing optimized models, and design the experiments for idea validation and conduct experimental data upon 3D reconstruction pipeline.

Comments

wt

posted by: wt posted in: 2022-5-10:3:8:40

Interested here! Free to contact!

Post a comment

Content

Fig. 5.6: Thread & Comment

All operations can only be done after user signing in the system, like, post new information and add meetings.

6 Conclusion and Future Work

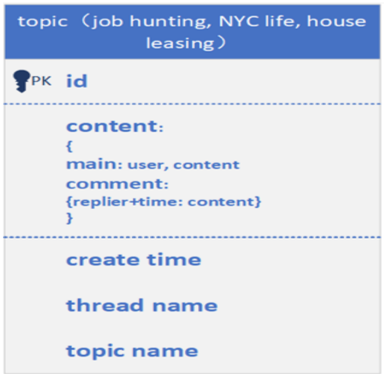
In this project we implemented an online blog platform for Columbia University students to share their academic experiences and colorful youth life. It contains the most important elements in a university student's life and will help them to select the most useful courses, rent the best apartments and enjoy the fantastic cuisines in NYC. Also it can be a helpful crime alarm for students to avoid physical harm.
In the future plan, we may add more functions to enrich our application:

1. Build an online shopping board for students to sell and buy second-hand textbooks and other stuff.

2. Build an instant message tool for students to have instant communication with each other like Facebook or Instagram.

3. Build a feedback system for students to report error issues and their feedback in the application.

7 Appendix



topic（job hunting, NYC life, house leasing）

PK id

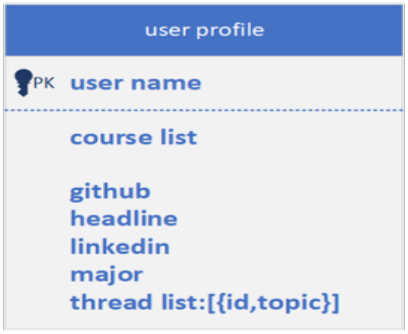
content:
{
main: user, content
comment:
{replier+time: content}
}

create time

thread name

topic name

- User Profile



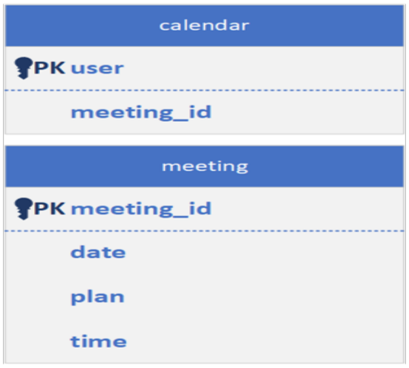
user profile

PK user name

course list

github
headline
linkedin
major
thread list:[{id,topic}]

- Calendar & Meeting



calendar

PK user

meeting_id

meeting

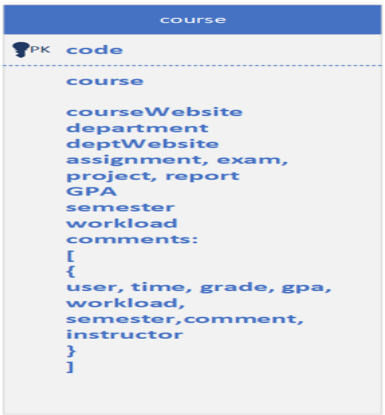
PK meeting_id

date

plan

time

- Course

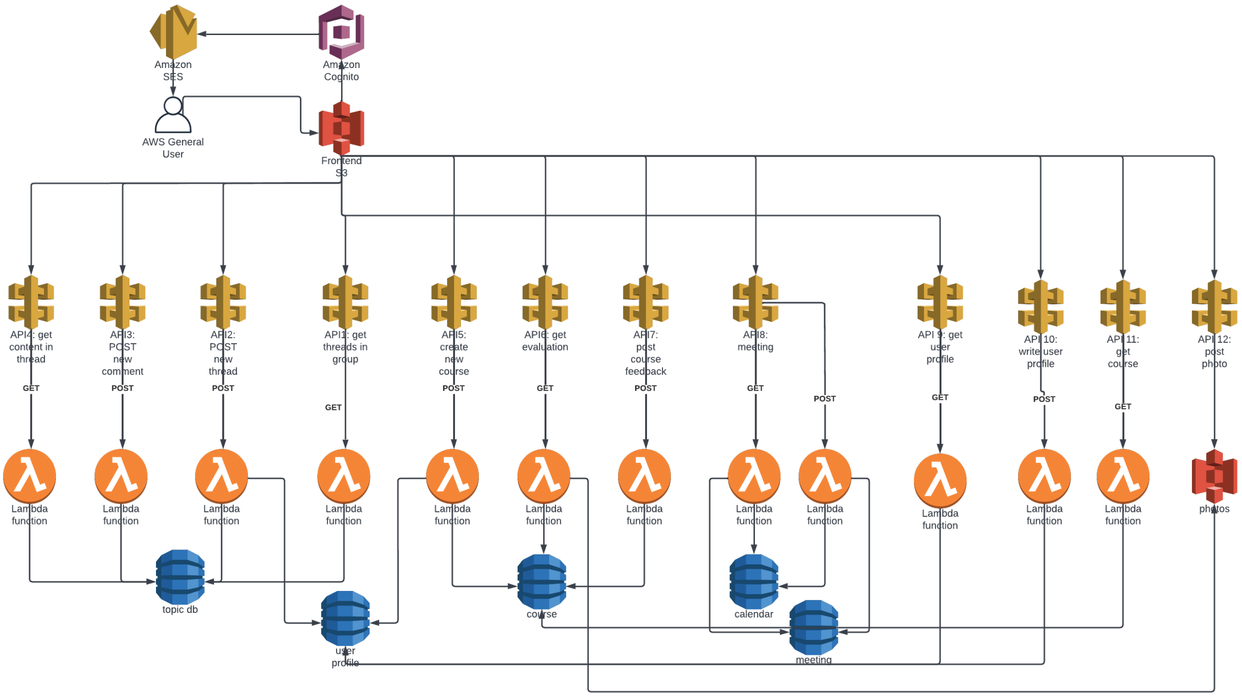


course

PK code

course

courseWebsite
department
deptWebsite
assignment, exam,
project, report
GPA
semester
workload
comments:
[
{
user, time, grade, gpa,
workload,
semester,comment,
instructor
}
]


Amazon SES
Amazon Cognito
AWS General User
Frontend S3
API4: get content in thread
API3: POST new comment
API2: POST new thread
API1: get threads in group
API5: create new course
API6: get evaluation
API7: post course feedback
API8: meeting
API 9: get user profile
API 10: write user profile
API 11: get course
API 12: post photo
GET
POST
POST
GET
POST
GET
POST
GET
POST
GET
POST
GET
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
Lambda function
photos
topic db
user profile
course
calendar
meeting