UNSW
AUSTRALIA

Assignment Cover Sheet

CVE003b | Group Assignment

Version CVE 003b.1

Never Stand Still | Faculty of Engineering | School of Civil and Environmental Engineering

SUBMISSION DETAILS

For submission to (staff name): XiaoWei Zhou

Course code: COMP 9900	**Course name:** Information Technology Project

Assignment details:

ACADEMIC REQUIREMENTS

Before submitting this assignment, students are strongly advised to:

- Review the assessment requirements contained in the briefing document for the assignment.
- Review the various matters related to assessment in the relevant Course Outline.
- Review the Plagiarism and Academic Integrity website at https://student.unsw.edu.au/plagiarism to ensure they are familiar with the requirements to provide appropriate acknowledgement of source materials.
- Retain a copy of this assessment for their records and in case it is misplaced and has to be re-submitted.

If after reviewing this material there is any doubt about assessment requirements then in the first instance the student should consult with the Course Coordinator and then if necessary with the Director of Learning and Teaching Committee.

While students are generally encouraged to work with other students to enhance learning, all assignments submitted for assessment by a student must be their entire own work and they may be required to explain any or all parts of the assignment to the Course Coordinator or other authorised persons. Collusion is where another person (s) assist in the preparation of an assignment without the consent of knowledge of the Course Coordinator.

Plagiarism and Collusion are considered as Academic Misconduct and will be dealt with according to University Policy.

STUDENT DECLARATION OF ACADEMIC INTEGRITY

Each of us declares that this assessment item is our own work, except where acknowledged, and has not been submitted for academic credit elsewhere, and acknowledge that the assessor of this item may, for the purpose of assessing this item:

- Reproduce this assessment item and provide a copy to another member of the University; and/or
- communicate a copy of this assessment item to a plagiarism checking service (which may then retain a copy of the assessment item on its database for the purpose of future plagiarism checking).
- We certify that we have read and understood the University Rules in respect of Student Academic Misconduct.

GROUP DETAILS

Group name: Runtime-Terror

Family Name: Deng	**First Name:** Lichen	**Signature:**
Family Name: Lin	**First Name:** Hanyi	**Signature:** Hanyi Lin.
Family Name: Zhang	**First Name:** ZeYu	**Signature:** ZeyuZhang
Family Name: GalagamaMudalige	**First Name:** Dinushi	**Signature:** Galagama
Family Name:	**First Name:**	**Signature:**
Family Name:	**First Name:**	**Signature:**

Intelligent Assistant for Electronic Logbook in Radiology Training

Group : RunTime Terror

SCRUM Master : Lichen Deng , z5126404@ad.unsw.edu.au, z5126404

Developers:

Hanyi Lin, z5142915@ad.unsw.edu.au, z5142915
Zeyu Zhang, z5099488@ad.unsw.edu.au, z5099488
Dinushi Galagama Mudalige, dinushi.gmud@gmail.com, z5118509

Content

Introduction

Aims

The existing system to maintain logbooks for medical candidates undertaking certification/ recertification in CCTA to date, is a tedious and lengthy process. The aim of this project will be to design and develop an easy to use web application that reduces the burden on the stakeholders of this process; medical candidates, supervisors and CCRT-CTCA administration(Conjoint Committee for the Recognition of Training in CT Coronary Angiography).

The aim of this report is to provide further background information that will be explored later on in the Needs Assessment section of this report. Followed by Epics and then user stories to explore the epics in more detail. UML diagrams and UI wireframes to show technical execution strategies. Concluding with our development plan that highlight the time and resource allocation to be expected for successful execution of this project.

Team Introduction

RunTime Terror is a team comprising of three postgraduate and one undergraduate member of UNSW CSE. The team's SCRUM master is Lichen Deng, accompanied by developers Hanyi Lin, Zeyu Zhang and Dinushi Mudalige. All members are proficient in several languages, however, for this project we will be primarily using python and javascript.

Project Choice

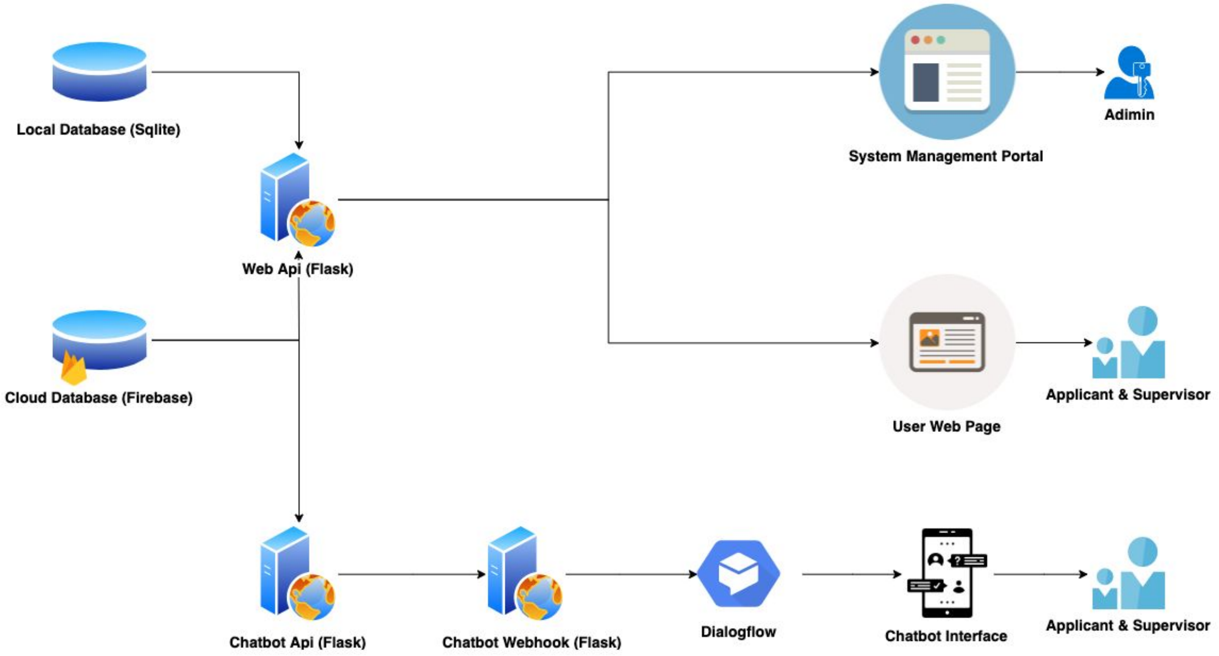
We chose the project topic based on the specialty and the interest of our team members. Since Dinushi is experienced in backend development and Zeyu focuses more on frontend technology, we decided to develop a web application which corresponds to their technology stack. At the same time, Julian and Hanyi, who is doing data science stream, would like to apply Neural Language Processing and Text Mining Techniques into this project. Thus we choose Topic 4: Intelligent Assistant for Electronic Logbook in Radiology training. In this project, we will use the Deep Learning approach to extract key information from Radiology Report, also we will build a Machine Learning based Chatbot to help users performing information recording and confirmation tasks in an interactive way. The application of this project is cloud-based, the data users, report files, logbooks will be stored in cloud database.

Project Scope

1. Only focusing on CTCA Certification Logbook.
2. The logbook is used for CTCA Registration(Level A & Level B) / CTCA Recertification (Level A & Level B) / CTCA conversion from Level A to Level B.
3. Reports similar with given Report Template is accepted.
4. Only Reports with pdf or word format is accepted.
5. The logbook is output as Excel sheet.

System Architecture

Interaction Diagram



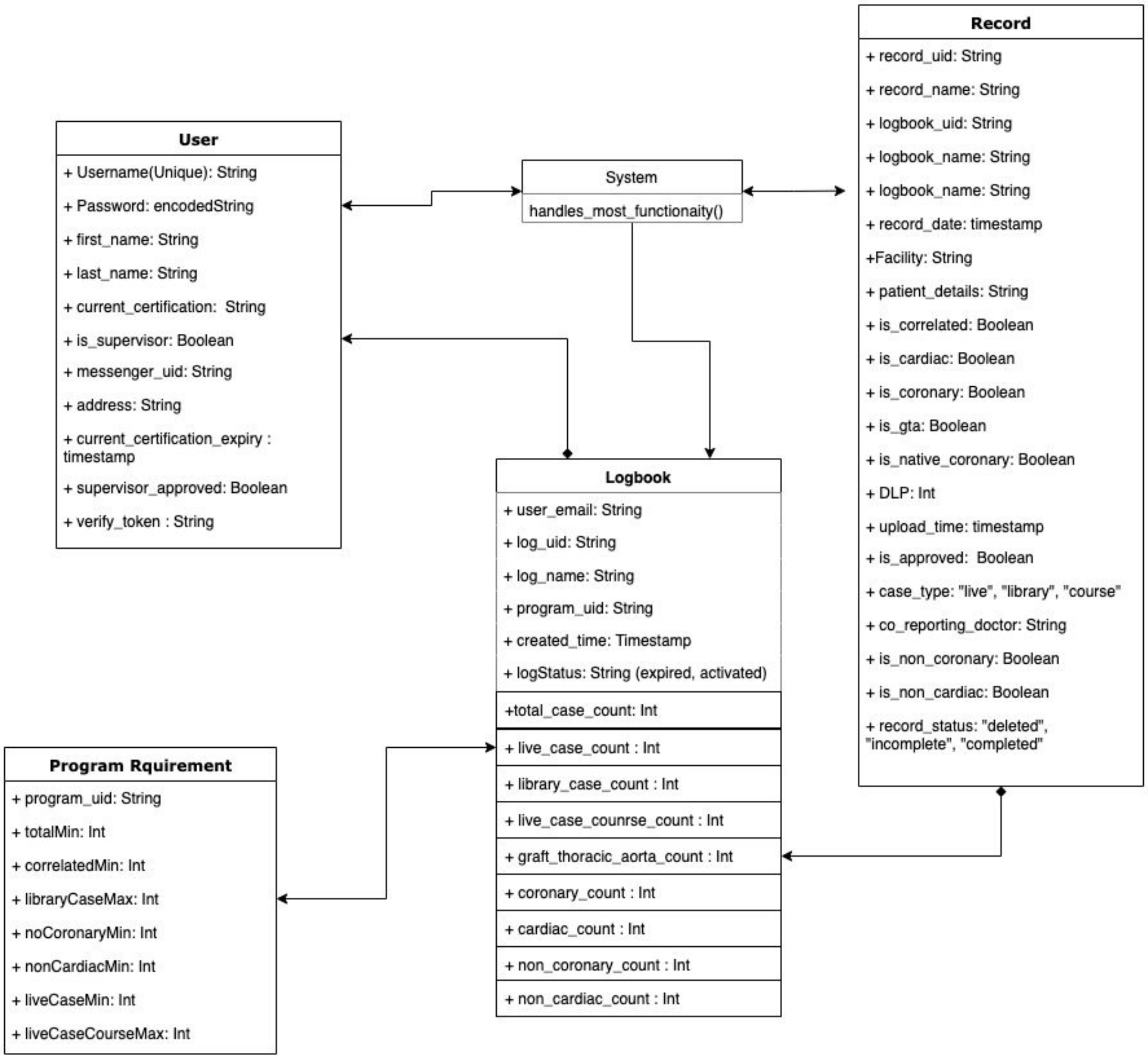
The data layer of this product is a mix of local database and cloud database. Sqlite is used for local database, which stores user register data (e.g username and encoded password token). Firebase is used for cloud database, which stores other data (e.g. logbook data, record data, user profile data…)

Flask is used for developing logic layer of this product. The logic layer include Web Data API, Chatbot Data API and Chatbot Logic. Furthermore, we use swagger editor to write API documentation.

In the Presentation layer, we use HTML to create the main structure of the web frontend. In addition, we use JavaScript to display the data from the backend dynamically. Furthermore, we use CSS and Bootstrap to contribute more styles for the website and make it more fancy.

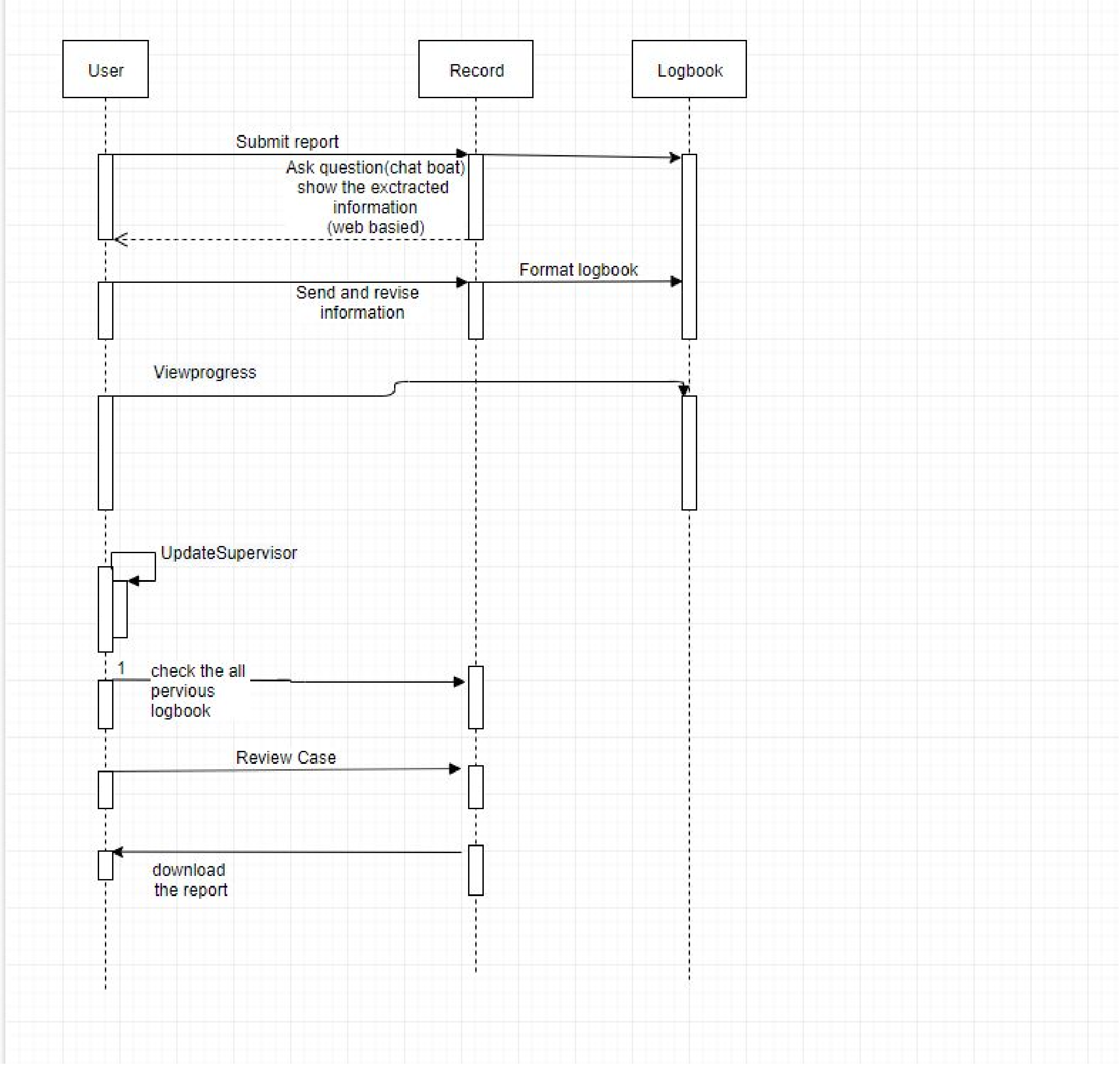
The communication between each layer is through HTTP requests.

Class Diagram based on database classes



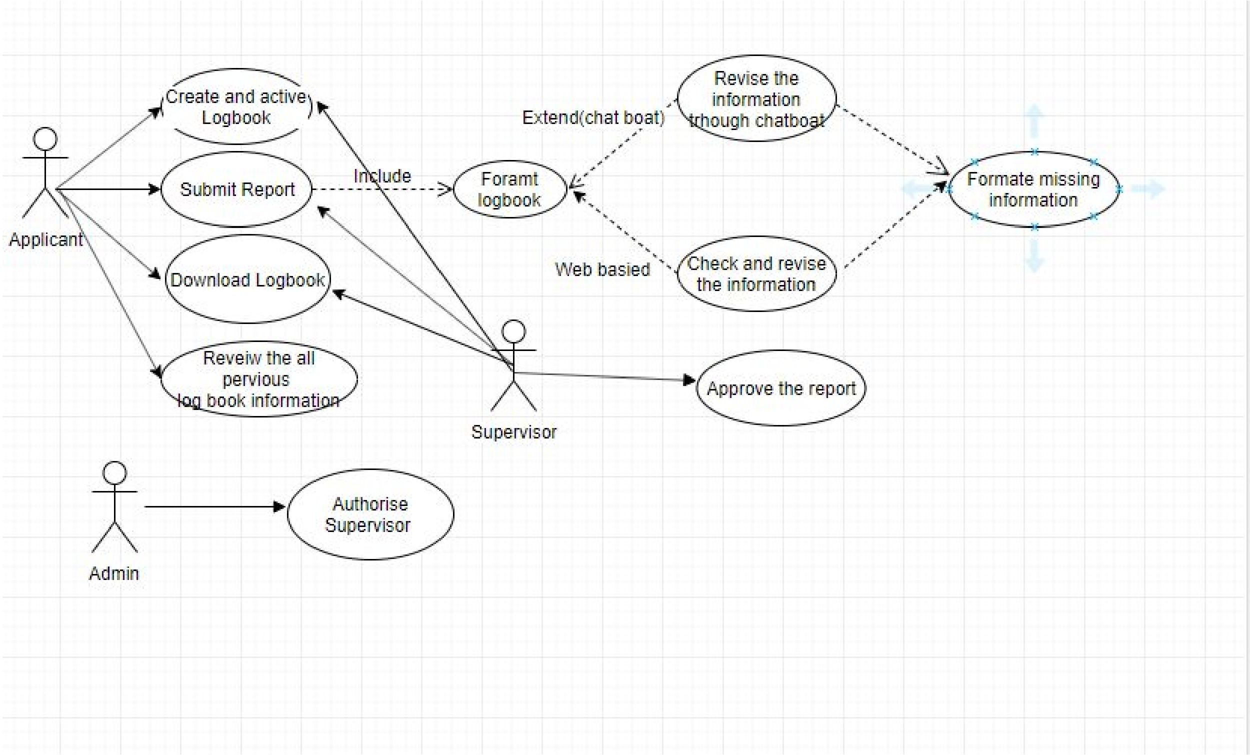
The diagram presents the interaction of the various Database classes and their associated variables. As is evident from the diagram, there are quite a few ‘composition’ relationships to maintain a rigid structure linking users to logbooks and logbooks to records. Without a User, a logbook cannot be made relevant to them and without a logbook, a record cannot be made. The database uses automatically created UIDs to keep track of the logbooks and records. This also adds a certain amount of security to the data since it is difficult to track the data with just a logbook name or record name bypassing the UIDs. I.e. It is possible that two separate users have the same logbook name but have unique IDs. The System is designed to be simple with the User classes for example being fully utilized to maintain and track three potential roles of ‘Admin’, ‘User’ and ‘Supervisor’ without having a multitude of different User Classes. The roles are solely maintained using the variables ‘is_supervisor’ and supervisor_approved’

Sequence Diagram



For the web page, user could update their personal details on the user profile page. This web page implement an input form to get the details and use a post request to transport the data to the backend and update the database.User submit the report through the system, the PDF file will convert to text information and the key word searching system will extract the key information to format the logbook. After that, the system will display the details of record and give feedback to let user to check the information. User need to correct the incorrect information and enter the miss information. Then the final information will be transported to the activated logbook. For the record detail part, it uses the same function to update the record information. In addition, for upload record page, we send a post request directly to the database and update it in the backend.

User Diagram


Create and active Logbook
Submit Report
Download Logbook
Reveiw the all pervious log book information
Applicant
Include
Foramt logbook
Extend(chat boat)
Revise the information trhough chatboat
Web basied
Check and revise the information
Formate missing information
Supervisor
Approve the report
Admin
Authorise Supervisor

Functionalities

Finished

Privileged-User system and Authentication Part

Description:

This function to be provided to each user is different. For the basic scope of this epic we identify two main privileged-users ; Candidate and Admin (ANZ-CTCA administrative-personnel). Any User that is currently certified as “Level B” is eligible to be an Admin approved Supervisor.

Functionality:

A User for the scope of this project can be:

- Undertaking certification for Level A CTCA specialisation
- Undertaking recertification for Level A CTCA specialisation
- Undertaking certification for Level B CTCA specialisation
- Undertaking recertification for Level B CTCA specialisation
- Undertaking a conversion from Level A to Level B CTCA specialisation

A Supervisor for the scope of this project is :

- Level B specialist approved by an ANZ-CTCA administrative-personnel

An Admin for the scope of this project is:

- An in-built account managed by the ANZ-CTCA administrative personnel

Functionalities (Basic and Optional)

Candidate (Basic):

- Access to creation of a new logbook
- Access to a chatbot for guidance on logbook entries
- Review ‘to-date’ logbook entries
- Access to export the report as an excel file

Supervisor(Basic):

- Access to a list of supervised candidates’ logbooks
- Dashboard upon login of relevant new logbook entries that are yet to be completed

Parsing documents for logbook creation and management Part

Description

For this part, There are two situations in which can be used keyword matching techniques to parse documents for information extraction. First in the process of creating a logbook and second in the creation of logbook entries.

Functionality:

- Creation of a logbook via the chatbot or via a button in web app.
- Creation of a logbook entry via the chatbot or via a button in web app.
- Upload a new document so that the system can extract the key information and format the logbook entry
- Generate questions according to the missing information

Chatbot Functionality

Description:

To provide a flexible and human-friendly service. the chatbot can be deployed on messenger to make sure users can easily access our service anywhere with their cell phone. Two kinds of user can use this chatbot: CTCA Applicants and CTCA Supervisors.

As a CTCA Applicant, user can:

- Bind the user account of CTCA applicant with chatbot through verify token.
- Unbind the account of CTCA applicant with chatbot.
- Create a new logbook of a specific training program.
- Check their logbook statics.
- Delete the currently activated logbook.
- View the previously uploaded record.
- Get access to the entry of maintaining/updating their personal logbook
- Access to chatbot service using Messenger via a cell phone in addition to web application.

As a CTCA Supervisor, user can:

- Bind the user account of CTCA applicant with chatbot through verify token.
- Unbind the account of CTCA applicant with chatbot.
- Create a new logbook of a specific training program.
- Check their logbook statics.
- Delete the currently activated logbook.
- View the previously uploaded record.
- Get access to the entry of maintaining/updating their personal logbook
- Access to chatbot service using Messenger via a cell phone in addition to web application.
- Get access to a list of entries of all the logbook record needs to be confirmed.
- Do the confirmation easily by sending message to Chatbot

Facilitation of easy to use UI Part

Description:

To make the user interface look more tidy and easier to use, we could develop a website which may contain

Functionalities :

- Dashboard Page: User can check the incomplete report and supervisor can approve the report.
- Homepage: User could select to register a new account or login as an existing user.
- Register Page: User could register as a new student.
- Login Page: User could login as an existing user.
- User Dashboard Page: User could check their incomplete records and open, edit or close the record.
- User Profile Page: User could see their profile or edit their profile on this page.
- Upload-File: User could upload the record on this page, and it will extract the key information from the record to the current logbook.
- New Logbook Page: User could create a new logbook and activate this logbook. The user could also see the list of the logbooks.
- Record Page: This page will display the records in the logbook and the complete progress of the logbook. Also, user could download a csv file from this to show the details of records in this logbook.
- Supervisor Page: For the user who is a supervisor to approve the records from their students.
- For the User Page, the user could upload his own file or request here, such as the updated logbook.
- Admin Dashboard Page: Admin could use this page to approve or disapprove the student to become a supervisor

Unimplemented Functionality

Chatbot Part

1.Pushing notification to supervisor once a new record need to be confirmed.
Reason: The nature of chatbots makes it not able to push message on its own initiative. Chatbots can only respond to users' requests.

2.Select appropriate training program with the suggestion provided by the Chatbot.

Reason: Users already can choose the training program when creating the logbook.

Feedback Description

1.A supervisor is able to easily provide advice amidst a busy schedule via the webpage .
Reason: Having inquired into the status quo we have concluded that there is no need for handling feedback. All feedback between the supervisor and candidate is face to face.

2.Give a feedback about Chatbot user experience.
Reason:This is not the relevant functionality of this project.

Web Part

1. We will display the information of our developer team (About Us page).
Reason: we think it is not necessary to display the information of the developer team, we need to focus more on the website development.

2. Get access to the entry of maintaining their personal logbook.
Reason:The attributes of logbooks are either unchangeable after creating(e.g. Logbook Name, Logbook Certification) or automatically updated by system (e.g. Record Total Count, Correlated Record Count…)

3. The candidate could type his detail to the chatbot and chatbot will extract the useful information to update his logbook.
Reason: We have implemented the chatbot as a small window at the right bottom of every web page.

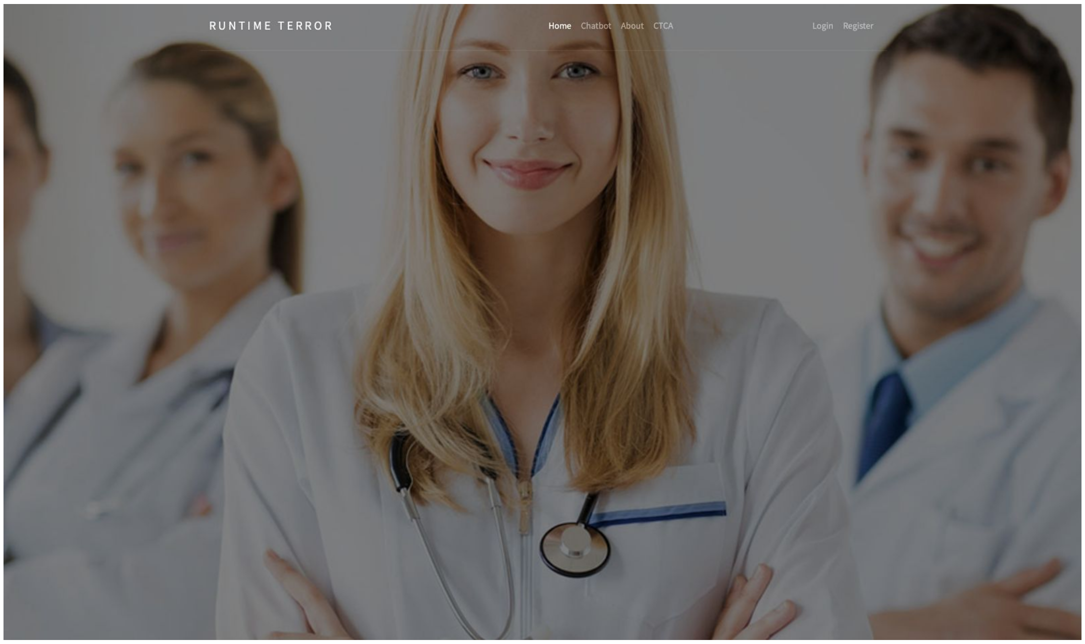
User Manual

This product could help the radiologist to save and format and record their medical report to logbook. It could automatically record the key information of the report and how many report they finished. The supervisor could approve the report on the system. Users also could check and submit the report through the mobile device.

Web Application

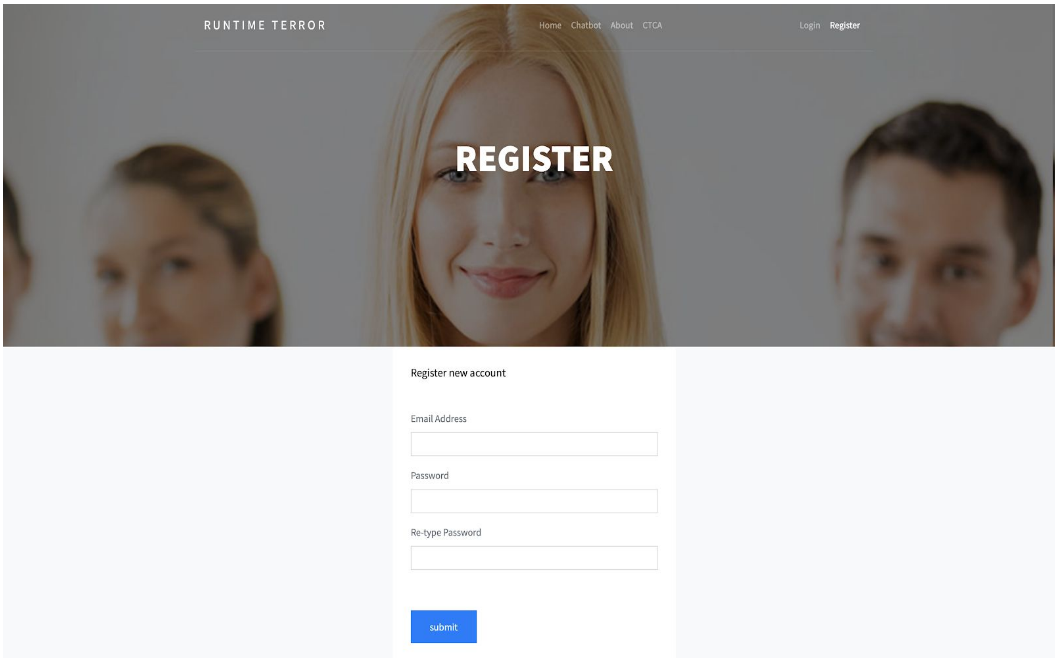
Homepage

Click the button to the login, register or get more information related to the CTCA.



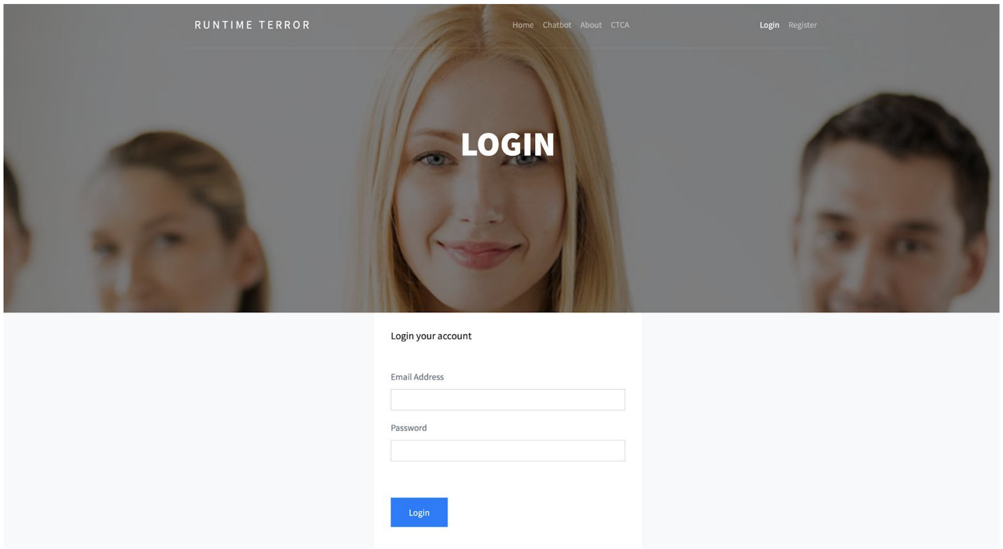
Build the User information

Enter the personal information and register for a new user. After registering successfully, it will redirect to the login page.



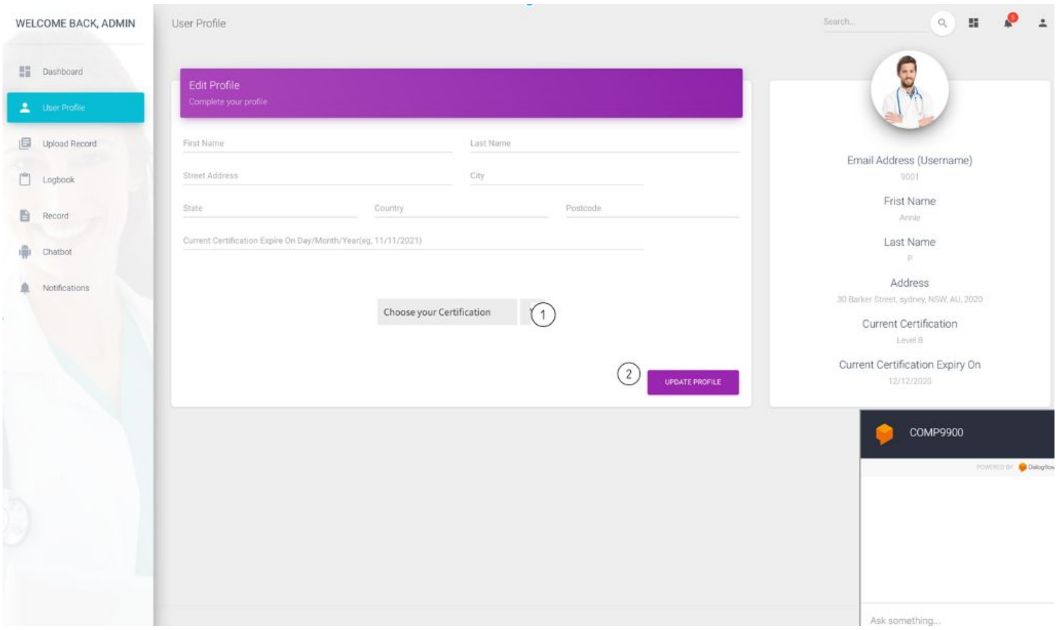
Login page

The user could input their username and password in this webpage. If it check the user exist and the identification is student, it automatically redirect to student dashboard. If it check the user exist and the identification is admin, it automatically redirect to admin dashboard.



Update personal information

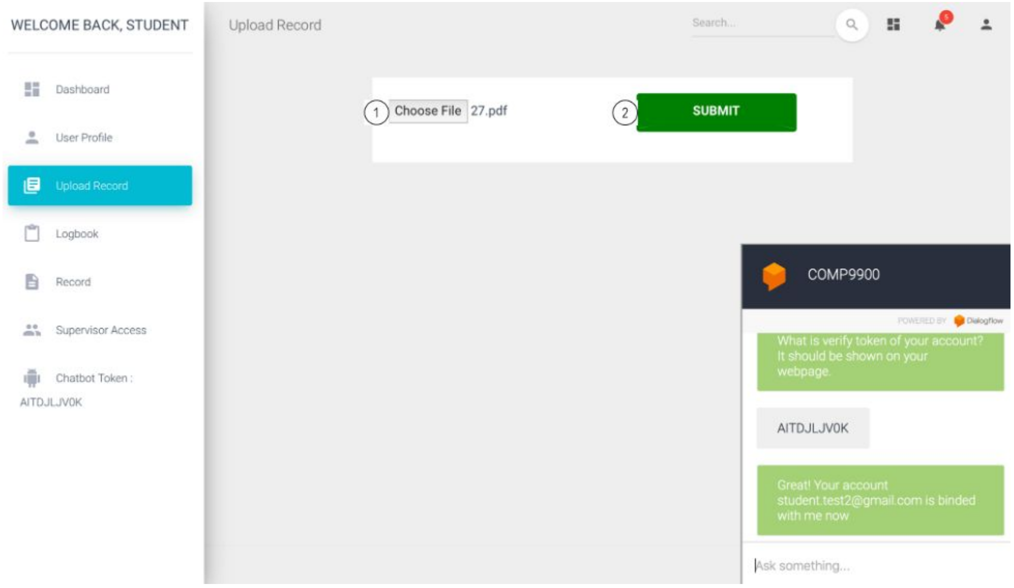
Enter and change the information of the user. Click the button ① to choose the level of the radiologist. If choice is level B, it seems the user apply to be a supervisor, this request needs to be approved by an administrator. Click ② to submit personal information.



Email Address (Username)
9001
Frist Name
Annie
Last Name
P
Address
30 Barker Street, sydney, NSW, AU, 2020
Current Certification
Level B
Current Certification Expiry On
12/12/2020

Report submit

Click ① to choose the record. Click ② to submit it.



Chatbot Token : AITDJLJV0K

27.pdf

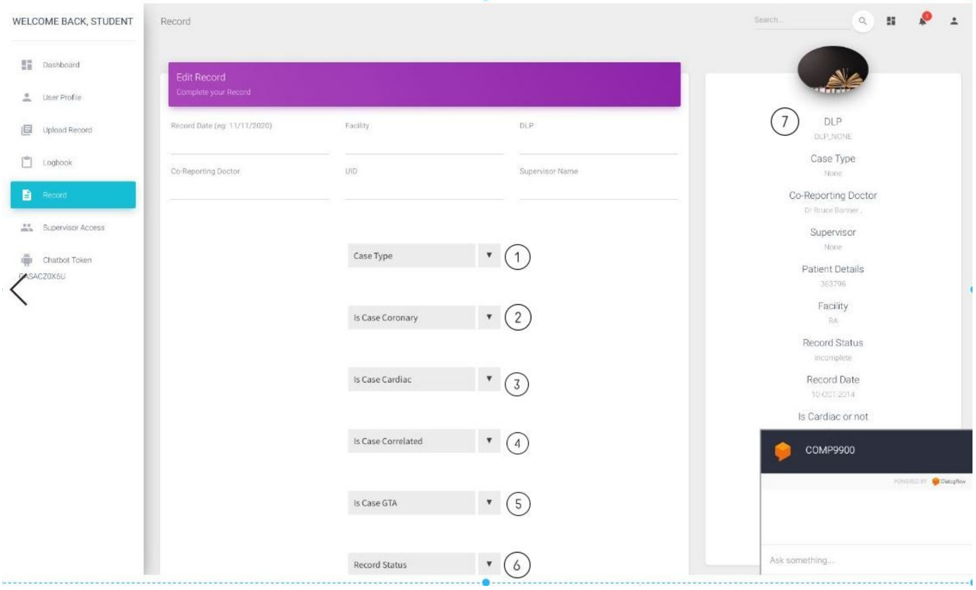
What is verify token of your account? It should be shown on your webpage.

AITDJLJV0K

Great! Your account student.test2@gmail.com is binded with me now

Information Confirmation

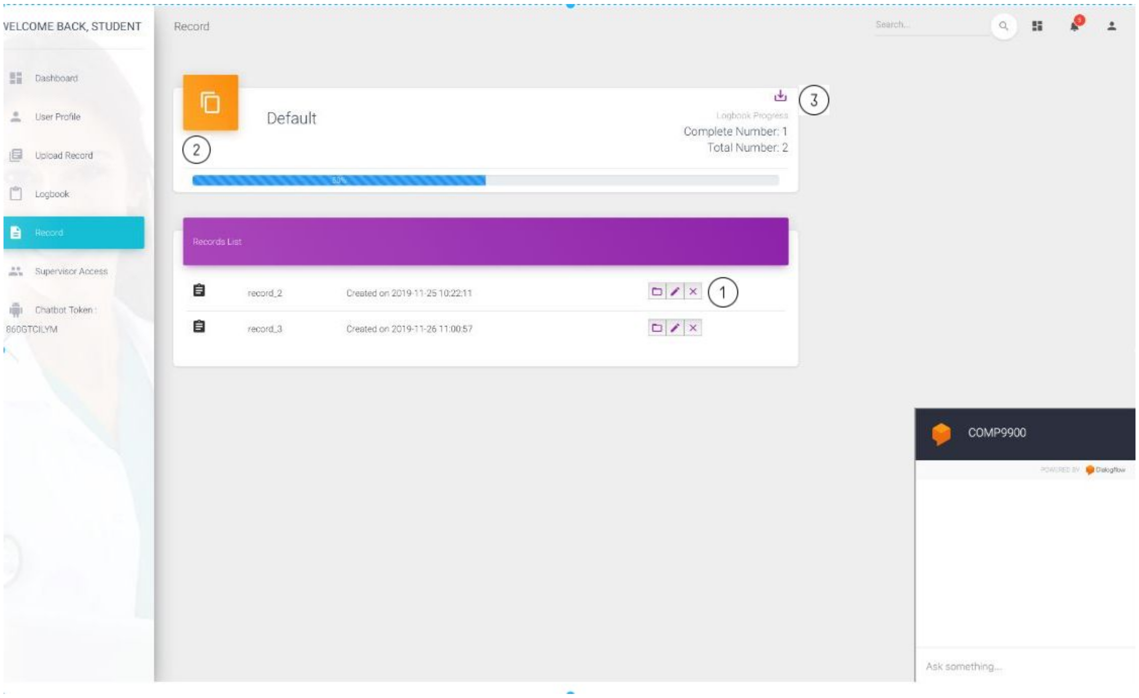
After submitting the report, the website will give a feedback. Check the key information of report through the window ⑦. The user could also click the ①②③④⑤⑥ to change the incorrect information.



DLP
DLP_NONE
Case Type
None
Co-Reporting Doctor
Dr Bruce Banner ,
Supervisor
None
Patient Details
363796
Facility
RA
Record Status
incomplete
Record Date
10-OCT-2014
Is Cardiac or not

Check Report

The ① part will give user three options -- open record (display the details of record), edit record and close record. Click ③ to download the CSV file of current logbook which contains the details of each record. The user can check the report progress from the progress bar ② which could display the percentage of complete records.



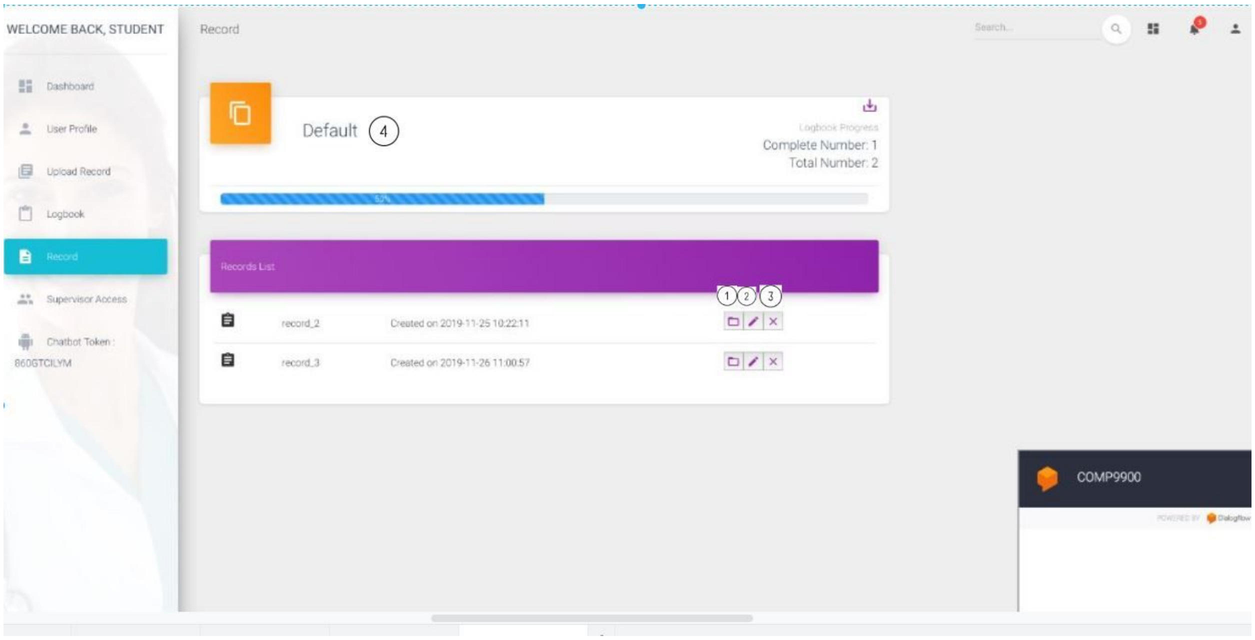
Default
Logbook Progress
Complete Number: 1
Total Number: 2

Records List

record_2 Created on 2019-11-25 10:22:11
record_3 Created on 2019-11-26 11:00:57

Dashboard

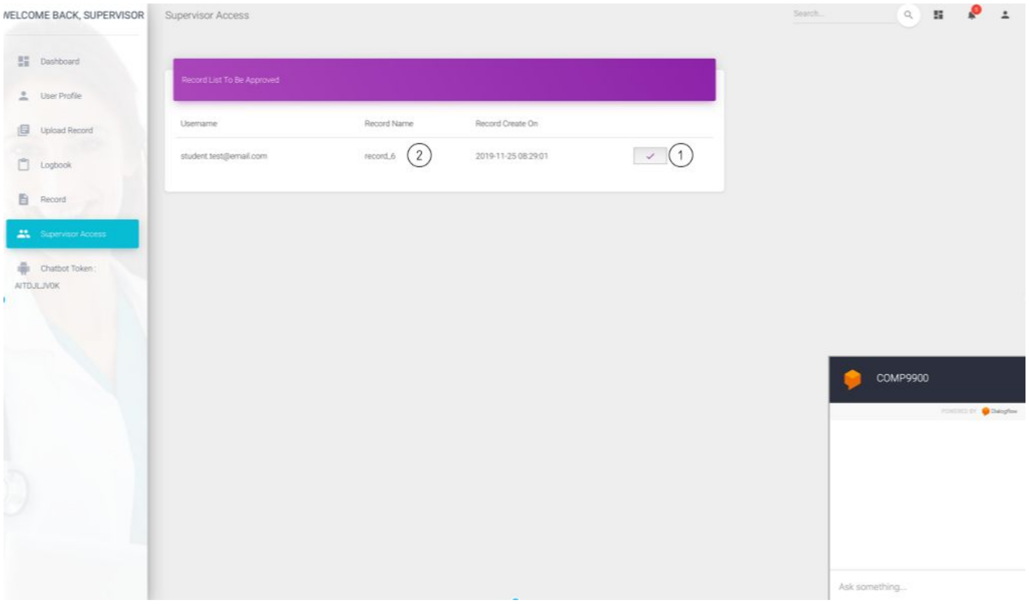
All the incomplete record can be found in dashboard. Click ①to review, click② to edit and click③ to delete the incomplete record information. The④ could show the progress of the incomplete report.



record_2	Created on 2019-11-25 10:22:11	
record_3	Created on 2019-11-26 11:00:57	

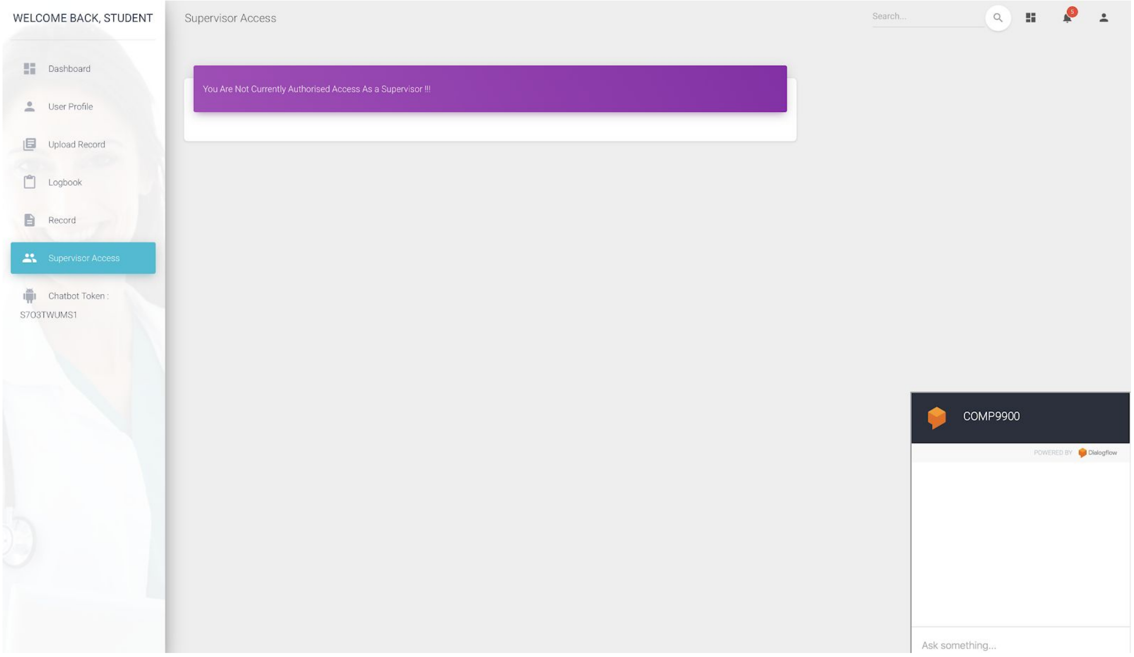
The special function in Supervisor part

For the supervisor account, click ① to approve the report from the level A student. They could see the record name and record creation time at part ②.



Username	Record Name	Record Create On	
student.test@email.com	record_6	2019-11-25 08:29:01	✓

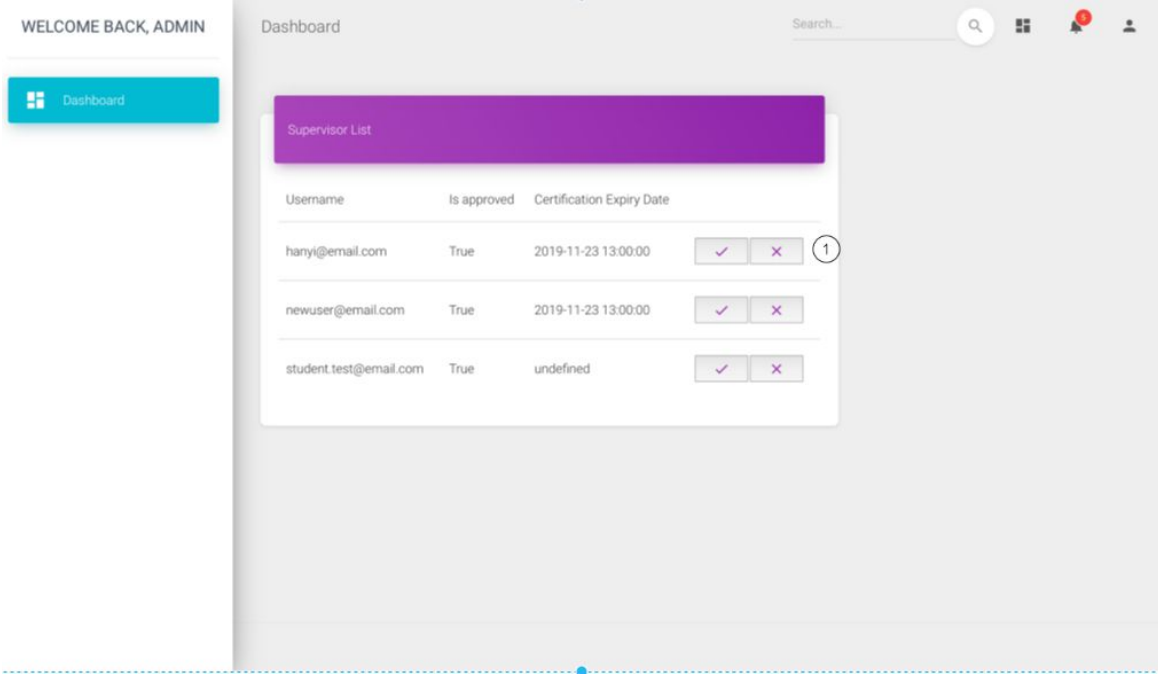
As the following graph shows, if the user has not been approved as a supervisor by the administrator, he will not have access to the supervisor page.



You Are Not Currently Authorised Access As a Supervisor !!!

The special function in Administrator part

If the user plan to become a supervisor. All the requests will be sent to the Administrator account. For the administrator, click ① to approve the application or close the request. After approved by the administrator, the user could become a supervisor.



Username	Is approved	Certification Expiry Date	
hanyi@email.com	True	2019-11-23 13:00:00	✓ ✕
newuser@email.com	True	2019-11-23 13:00:00	✓ ✕
student.test@email.com	True	undefined	✓ ✕

Chatbot Part

You can use both the website and Facebook Messenger to get access to the chatbot service.

For now the messenger chatbot is in test mode, if you want to use messenger bot, you please contact me (Email: hanyilin0315@outlook.com OR Facebook: Hanyi Lin) to add your facebook account to the messenger bot test user.

After that, please go to your facebook main page, search page **Page for 9900** and chat with this page! Once the above steps are setting correctly, the chabot should be able to chat with you.

1. **Account Binding**

   At the first time you go to the chatbot, you are not bound with any account of our website.

   In website chatbot, you can bind your account by sending 'bind webpage chatbot'.

   In the Messenger chatbot, you can bind your account by sending 'bind messenger chatbot'.

   After that, the bot will ask you for verifying token of your account. You can find your verify token in the left hand side bar of our website.

   If the token is correct, you should be able to bind with the chatbot.

2. **Cancel Binding**

   You can also unbind your account with chatbot.

   In website chatbot, just simply refresh the webpage, because the webpage chatbot doesn't store the user information persistently.

   In the Messenger chatbot, you need to unbind the account by sending 'unbind account' or 'cancel binding'.

3. **Create Logbook**

   You can create a logbook by sending 'create logbook'.

   The bot will ask you for the certification level and logbook name. If you currently have no active logbook, you will successfully create a new one.

4. **Delete Logbook**

   You can also delete your current active logbook by sending ‘delete logbook’.

   If you currently have an active logbook, you will successfully delete that one.

5. **View Logbook**

   You can view your currently active logbook statistics by sending ‘view logbook’.

   If you currently have an active logbook, the bot will show you the statistic information of that logbook.

6. **View Record**

   You can view your record by sending ‘view record’.

   The bot will ask you for the logbook name and record index.

   If your record index and logbook name is valid, you will get the record information as well as the url of the report of this case.

7. **Update Record**

   You can update your record by sending update record’.

   The bot will ask you for the logbook name and record index as well as the field you want to change and the value.

   If your record index,logbook name, field name and value is valid, the bot will return the current record information as well as the url of the report of this case, and ask you to confirm your change.

   If you confirm it, the bot will change this field for you.

8. **Supervisor View Record List**

   If you are an admin-approved supervisor, you can view the list of records to be confirmed by sending ‘check the records to be confirmed’.

9. **Supervisor Approve Record**

   If you are an admin-approved supervisor, you can approve the record need to be confirmed by sending ‘approve record’.

The bot will ask you for the logbook name and record index and the owner username of this record.

If the information is correct, the chatbot will return you the detailed information as well as case report url to you, and ask for confirmation.

If you confirmed, the status of this record will become approved.

Deploy Instruction

Web Application

1. **Download File:**

   Download file **capstone-project-runtime-terror-master.zip** from https://github.com/comp3300-comp9900-term-3-2019/capstone-project-runtime-terror and unzip it.

   Create a local version of python (virtualenv).

   ```
   virtualenv -p python3 env
   ```

2. **Activate virtualenv**

   ```
   source env/bin/activate
   ```

3. **Install required modules.**

   ```
   pip install -r requirements.txt
   ```

4. **Install a local database and initialize with an in-built 'Admin account' (NOTE: This step only needs to be done in the very first instance)**

   ```
   python3 db_create.py
   ```

   ```
   python3 db_populate.py
   ```

5. **Good to go! Run the server.**

   ```
   python3 run.py
   ```

Chatbot

1. **Setup Web-Tunneling Tool:**

   Open a new terminal and go to the directory **/capstone-project-runtime-terror-master/chatbot,** then input:

   ```
   npm install -g localtunnel
   ```

2. **Install Python package:**

   Open a new terminal and go to the directory **/capstone-project-runtime-terror-master/chatbot,** then input:

   ```
   pip3 install -r requirements.txt
   ```

   If successful, you should see:

   ```
   Successfully installed Flask-1.1.1 Flask-RESTful-0.3.7 Jinja2-2.10.3 MarkupSafe-1.1.1 Werkzeug-0.16.0
   aniso8601-8.0.0 cachecontrol-0.12.5 cachetools-3.1.1 chardet-3.0.4 click-7.0 dialogflow-0.7.2
   firebase-admin-3.1.0 google-api-core-1.14.3 google-api-python-client-1.7.11 google-auth-1.7.1
   google-auth-httplib2-0.0.3 google-cloud-core-1.0.3 google-cloud-firestore-1.6.0
   google-cloud-storage-1.23.0 google-resumable-media-0.5.0 googleapis-common-protos-1.6.0 grpcio-1.25.0
   httplib2-0.14.0 idna-2.8 itsdangerous-1.1.0 jsonschema-2.6.0 msgpack-0.6.2 protobuf-3.10.0 pyasn1-0.4.8
   pyasn1-modules-0.2.7 pytz-2019.3 requests-2.22.0 rsa-4.0 six-1.12.0 uritemplate-3.0.0 urllib3-1.25.7
   ```

3. **Launch the API Service at Port 7000**

   Open a new terminal and go to the directory **/capstone-project-runtime-terror-master/chatbot/app13/demo,** then input:

   ```
   python3 __init__.py
   ```

   If successful, you should see:

   ```
   * Serving Flask app "__init__" (lazy loading)
   * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
   * Debug mode: on
   * Running on http://127.0.0.1:7000/ (Press CTRL+C to quit)
   * Restarting with stat
   * Debugger is active!
   * Debugger PIN: 285-156-347
   ```

4. **Launch the Chatbot Service at Port 5000**

Open a new terminal and go to the directory **/capstone-project-runtime-terror-master/chatbot/,** then input:

```
python3 app.py
```

If successful, you should see:

```
* Serving Flask app "app" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

After that, please open a new terminal and input:

```
lt --port 5000 --subdomain 9900chatbot2
```

If successful, you should see:

```
your url is: https://9900chatbot2.localtunnel.me
```

5. **For Messenger Chatbot User:**

   Because the formatting issue cannot be fixed on webpage chatbot, we **highly recommend you to use messenger bot for better experience**.

   For now the messenger bot is in test mode, if you want to use messenger bot, you please contact me (Email: hanyilin0315@outlook.com OR Facebook: Hanyi Lin) to add your facebook account to the messenger bot test user.

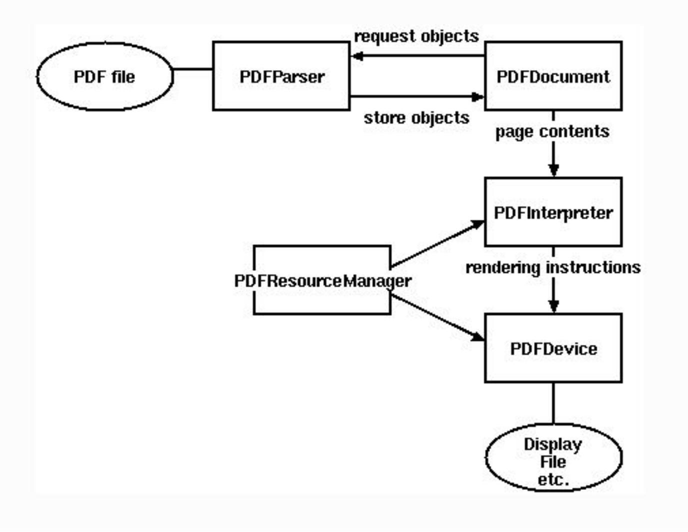
   After that, please go to your facebook main page, search page **Page for 9900** and chat with this page! Once the above steps are setting correctly, the chabot should be able to chat with you.

   After that, please open your chatbot on webpage and Messenger, and say hello!

API References

PDF Miner

PDFMiner is free API and it can extract information from PDF documents. It could extract the key information from the report. And this api is free for commercial purposes.



API Structure (Levia3 2017)

DialogFlow

Dialogflow is a powerful chatbot building framework, which provides robust Machine Learning based NLU Algorithm, as well as serval good conversation management feature (e.g Context, Follow-up Intent, Prompt). It also supports integration in several different platforms (e.g. Messenger, Slack, Twitter). We use REST API to connect DialogFlow intent detector and chatbot logic in backend.

The DialogFlow is free for non-commercial usage,so the license does not have an expiry date.

Localtunnel

Localtunnel is a web tunneling tool that allows you to easily share a web service on user's local development machine without messing with DNS and firewall settings. Users are able to customize their own subdomain name when using localtunnel, other users can get access to the service through the url with this subdomain name.

Local tunnel is a free tool for now. It doesn't need any credential.

Firebase SDK

In order to deal with the large amount of data on a real time basis we used a Cloud database by Google called Firestore. We used a python supported SDK called ‘firebase_admin.firestore module’ that contains utilities for accessing the GoogleCloud Firestore databases. This resource can be freely accessed and all documentation that supports its use is freely available. Thereby, it does not require any credentials.

References

Levia3 (2017) *pdfminer.* [Online]. Available at: *https://media.readthedocs.org/pdf/pdfminer-docs/latest/pdfminer-docs.pdf* (Accessed: 26 November 2019).