ISS Project –

Intelligent Covid Detection and Chatbot Platform

Hong Xiaohui (S9476943D)

Anita Koo Shi Qi (S9444480B)

Sanjeven Ramakrishnan (S9139938E)

National University of Singapore | Institute of Systems Science

https://github.com/xiaohuihong/IRS-PM-2021-01-16-IS03PT-GRP-3Musketeers-CoviDetector.git

Table of Content

Abstract: With the recent low number of community cases of covid-19, mass gathering is allowed. However, participants will have to take an antigen rapid test for Covid-19 and obtain a negative result before admission. Hence, the event planner must ensure the participants safety/health condition before allowing participants to enter the event venue. This project proposes a solution to this problem by using a machine learning model to determine the likelihood of participants having Covid-19. In addition, it includes a chatbot to allow the participants to know current information about Covid-19 and the event itself.

1. Business Case

Recent news reported that the Singapore government allowed mass gathering but with a requirement. Before a participant can enter the venue, the event organizer has to ensure that he/she is not contacted with Covid-19. To implement this, the event organizer will have to check with each participant individually on their health status before letting them in. As social distancing enforcement is a must, a long queue will be expected. Meaning, more manpower is needed to maintain the queue. If a shorter time can be implemented to handle these, the queue will be shorted thus lesser manpower is needed.

We propose that this can be handled with a web application (mobile-friendly) that has a survey for the participants to answer to get their health status. Participants with a mobile device would be able to answer the necessary questions before reaching the front of the queue. The result of the survey will let the queue maintainer either guide the participant to the swab test area or let the participant into the venue. Moreover, it also includes a chatbot that allows the user to know (real-time) information about Covid-19 and the event itself. This may keep the participants occupied while waiting for their turn to enter the venue. The solution may be tailored to suit different types of events.

2. System Model

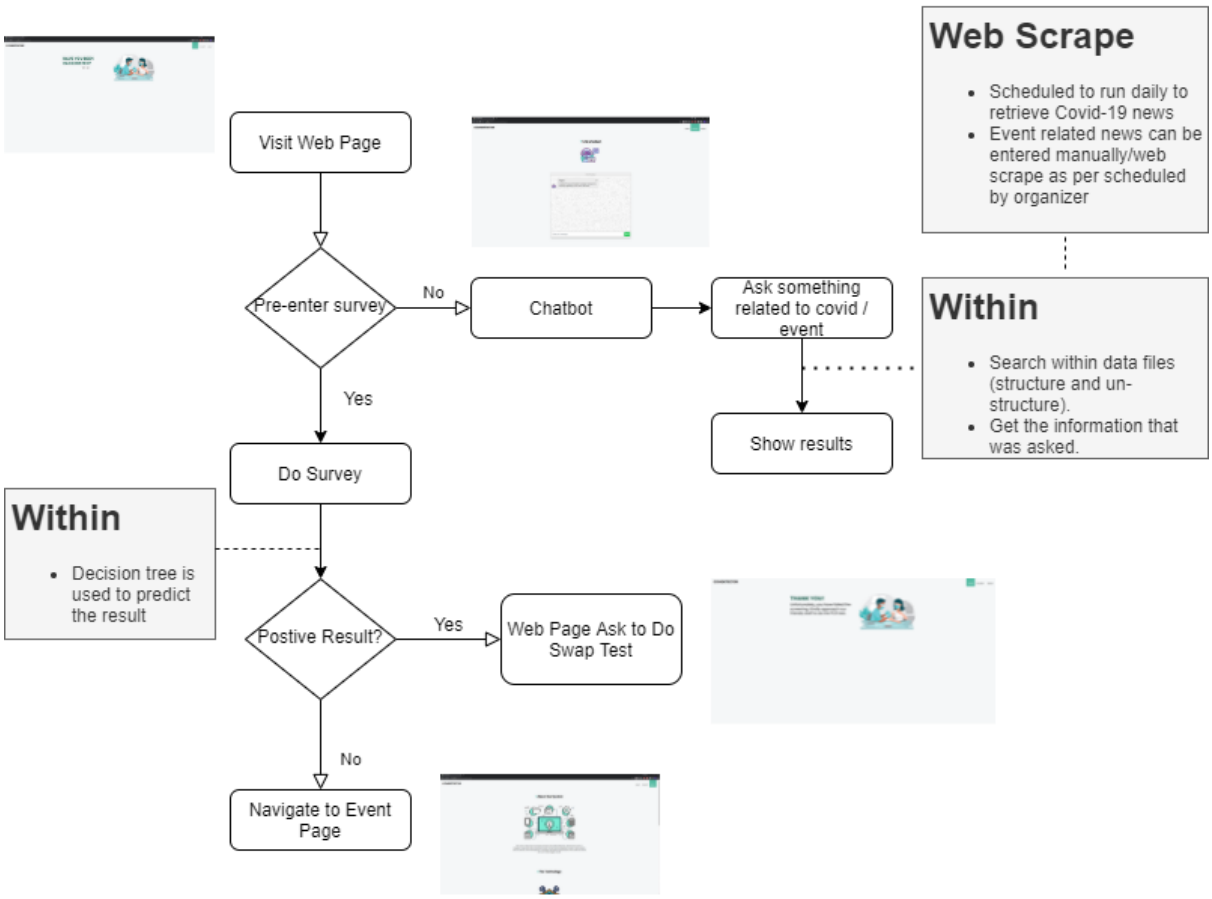
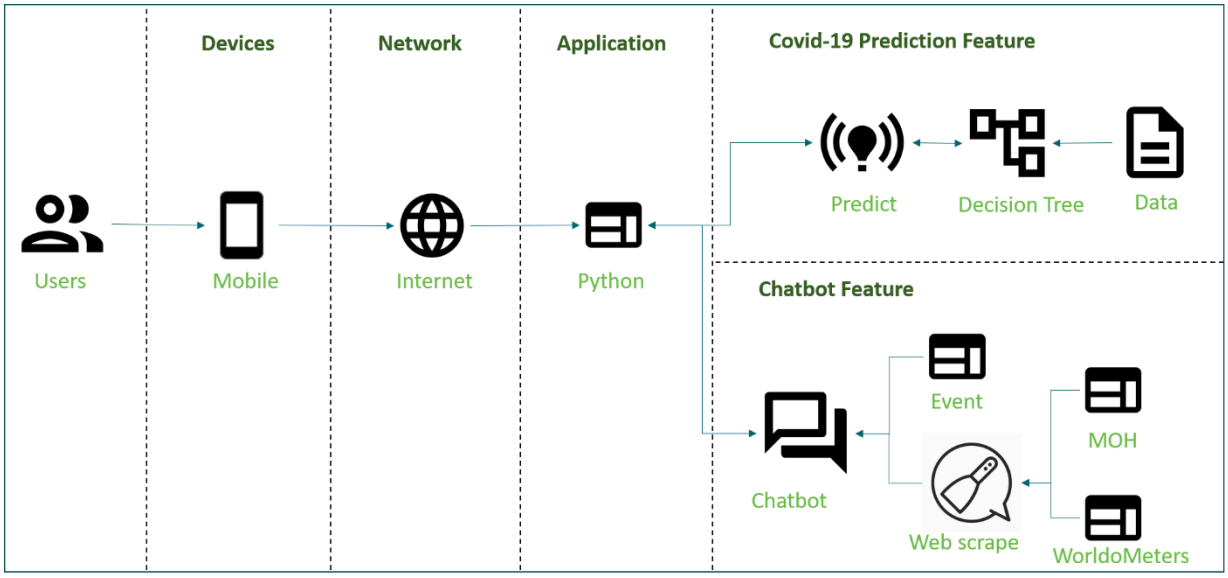


Figure 1 CovidDetector System - Flowchart



4

Figure 2 CovidDetector System - System General Design

3. System Development & Implementation

3.1. System Architecture

3.1.1. Overview

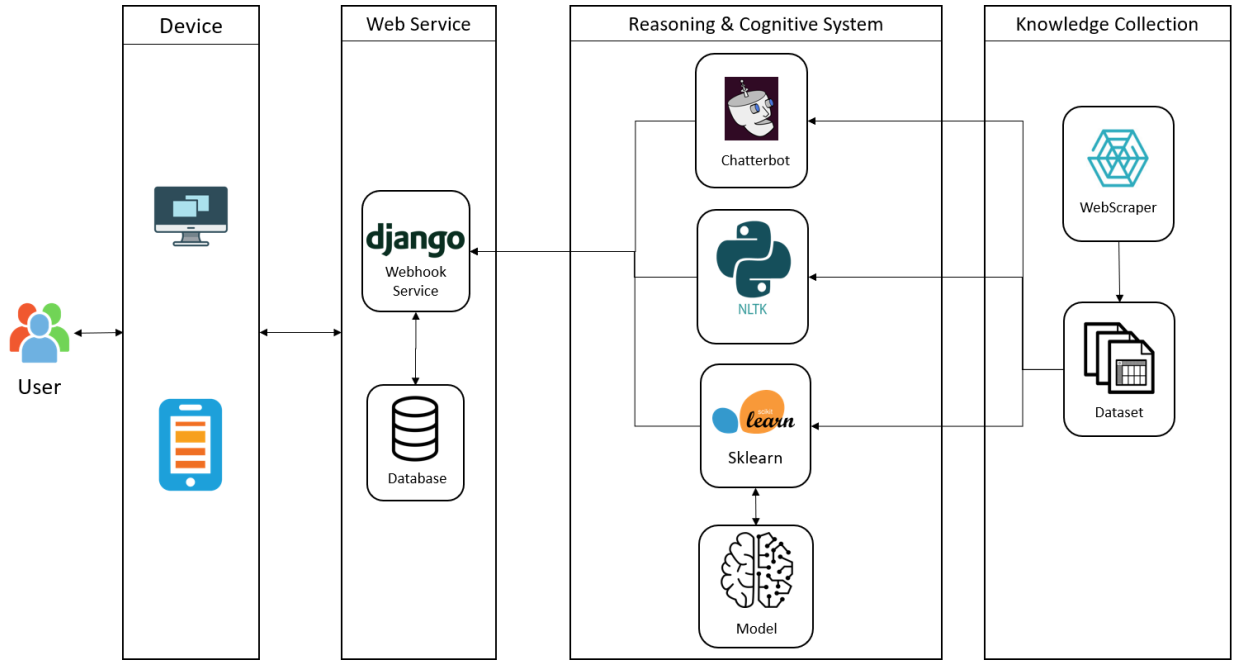


Figure 3 CovidDetector System - System Architecture

In order to build up a system which contains front-end and back-end and can integrate with machine learning libraries, as a result, Django, a python-based web framework, is decided to use this project. The other reason that we choose Django is it is built with SQLite database by default which is easier to help us deploy models.

Our eco-system is constituted by two components, predictive model for covid cases and chatbot for covid frequent questions. We choose to leverage scikit-learn which is a python library for predictive data analysis to build a model to predict if a user gets Covid-19 based on his symptoms and his profile. The chatbot consists of two types, one uses chatterbot which is a Python library that makes it easy to generate automated responses to a user's input, and the other one is developed by scikit-learn and nature language tool kit to find out the most similar pattern from dataset based on user inputs. All of the knowledge and data are coming from web scraper which would run daily to refresh the knowledge base.

3.2. Data Sources

3.2.1. Covid Symptoms and User Profile Data Set

Our dataset comes from the Israeli Ministry of Health that publicly released data of all individuals who were tested for SARS-CoV-2 via RT-PCR assay of a nasopharyngeal swab. During the first months of the COVID-19 pandemic in Israel, all diagnostic laboratory tests for COVID-19 were performed according to criteria determined by the Israeli Ministry of Health. While subject to change, the criteria implemented during the study period included the presence and severity of clinical symptoms, possible exposure to individuals confirmed to have COVID-19, certain geographical areas, and the risk of complications if infected. Except for a small minority who were tested under surveys among healthcare workers, all the individuals tested had indications for testing. Thus, there was no apparent referral bias regarding the vast majority of the subjects in the dataset used in this study. In addition, all negative and positive COVID-19 cases in this dataset were confirmed via RT-PCR assay. Therefore, we can consider this dataset is genuine and reliable.

The following list describes each features of the dataset:

A. Basic information:

1). Sex (male/female)

2). Age ≥60 years (true/false)

B. Symptoms:

3). Cough (true/false)

4). Fever (true/false)

5). Sore throat (true/false)

6). Shortness of breath (true/false)

7). Headache (true/false)

C. Other information:

8). Known contact with an individual confirmed to have COVID-19 (true/false)

3.2.2. Chatbot FAQ and Related Information

To synchronize the knowledge base used by the chatbot with the frequently updated online webpage, the data extraction of the system is required to be executed daily. This can be done with the help of Cronjob external time scheduler, keeping the information updated automatically. This allows users to get the most accurate answers at any point of time while using the chatbot. The following table shows the websites we used to scrape the latest Covid-19 information.

Websites	Data to Crawl
https://www.who.int/news-room/q-a-detail/q-a-coronaviruses	The WHO provides answers to frequently asked question regarding the coronavirus.
https://www.worldometers.info/coronavirus/	Infection Status across all countries
https://www.moh.gov.sg/covid-19	Latest updates & advisories from Ministry of Health, Singapore

3.3. Knowledge Elicitation and Extraction

3.3.1. Manual Extraction

The covid symptoms and user profile data are recorded in Israeli originally but it has been translated into English Israeli analyst. We downloaded and extracted it from *https://github.com/nshomron/covidpred*

3.3.2. Web Scraper

a. Beautiful Soup Python library:
   i. Visit the websites to retrieve Covid-19 related information.
   ii. Study the html website and identify the information to extract.
   iii. Html parser is utilised to extract the identified information.
   Extract relevant information from the html and write to the data file.
   Extract relevant urls and visit those websites to get relevant information.

3.4. Reasoning System

3.4.1. Overview

After discovering the knowledge base, we found out that we need to make a classification or prediction based on the values of known attributes or features as the final output of our model should be the result of Covid-19 that is positive and negative or we also can consider it as 0 and 1. Thus, we chose decision trees and decision rule learning as our predictive model.

3.4.2. Decision trees and Decision rule learning

Before applying the model, we converted categories to numbers by one-hot and made do with N-1 binary variables, then the data would be as following.

	cough	fever	sore_throat	shortness_of_breath	head_ache	corona_result_positive	age_60_and_above_Yes	female
0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
2	0	0	0	0	0	0	1	1
3	0	0	0	0	0	0	0	0
4	0	1	0	0	0	0	0	0
5	1	0	0	0	0	0	0	0
6	1	1	0	0	0	0	0	0
7	0	0	0	0	0	0	0	1
8	0	0	0	0	0	0	0	0
9	1	1	0	0	0	0	0	0

Figure 4 CovidDetector System - Sample Data after one-hotting

Usually, raw trees are often too big to be understandable, so we did pruning on the tree by min_samples_leaf=100 and max_depth=5 which means the minimum number of samples required to be at a leaf node is 100 and the maximum depth of the tree is 5. This pruning can improve accuracy by reducing overfitting and make the tree easier to understand. As a result, the tree is plotted as below.

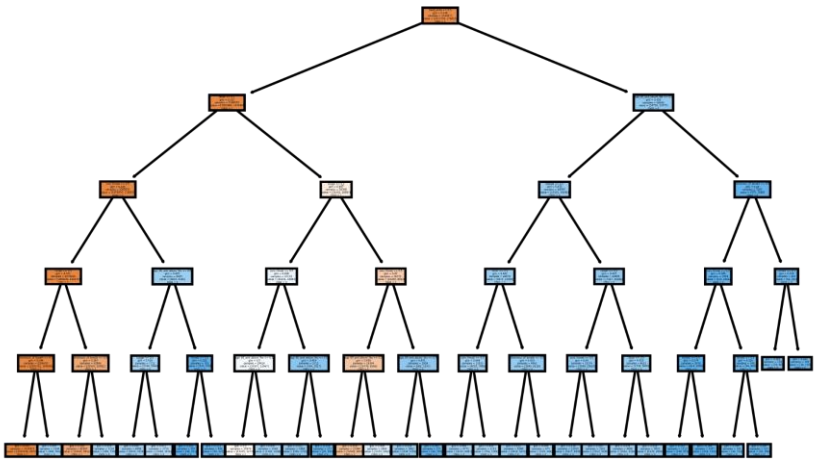


Figure 5 CovidDetector System - Decision Tree and Rules

The full size of tree and text presentation can be found in "SystemCode\CoviDetector\model\decision_tree_model.ipynb".

The classification report is shown in the table below. As we can conclude, the general accuracy is 93% but the positive case (1) is only 68% although the negative case (0) is 94%. The reason that causes this is the number of positive cases in the sample is quite small. In the future, we can get more samples from MOH or the production data after this system is deployed to real use cases to improve the accuracy.

	precision	recall	f1-score	support
0	0.94	0.99	0.96	504365
1	0.68	0.27	0.39	44154
accuracy			0.93	548519
macro avg	0.81	0.63	0.67	548519
weighted avg	0.92	0.93	0.92	548519

3.5. Cognitive System

3.5.1. Overview

While participants enter an event, they could have a requirement to ask questions to understand the current situation about Covid-19 or the specific information about the event. We planned to build a FAQ chatbot to answer participants' questions, but we found that it was not enough to respond to them precisely and it was hard for users to prepare a long worksheet about FAQ. So we decided to integrate FAQ chatbot and NLP chatbot to enhance its functionalities. Since we don't have events' details in this project, our data mainly focus on Covid-19 information.

3.5.2. Chatbot by FAQ

We use Chatterbot Library, a Python library that makes it easy to generate automated responses to a user's input, to develop the FAQ chatbot. We applied the Jaccard Similarity algorithm that we learnt from course, Reasoning System, in the statement comparison function which means we calculate the similarity of user input and question list in FAQ worksheet based on the Jaccard index. In

general, the Jaccard Similarity measures similarities between sets. It is defined as the size of the intersection divided by the size of the union of two sets that is concluded as the following formula.

$$Sim_{jaccard}(A,B) = |A \cap B| / |A \cup B|$$

The response selection method we used is get_first_response that the chatbot only returns the statement has the highest similarity. In addition, we set a threshold for the response's confidence. The confidence value represents a rating of how accurate the logic adapter expects the selected response to be. Confidence scores are used to select the best response from multiple logic adapters. We used this threshold value to tell the system if it should switch the FAQ chatbot to NLP chatbot that is explained in section 3.5.3.

3.5.3. Chatbot by NLP

Other than the FAQ list scraped from websites, we also scrawled some news and articles about Covid-19. When the FAQ cannot answer users' questions, we can dig the responses from these news and articles.

After applying the common normalization methods like tokenization, case lowering, punctuation removal and lemmatization, we used tf-idf indexing as the vectorization method and leveraged cosine similarity algorithm as similarity measurement. The formula can be found as following:

$$Sim(D_i, D_j) = \frac{D_i \bullet D_j}{|D_i| * |D_j|} = \frac{\sum_k w_{ki} w_{kj}}{\sqrt{\sum_k w_{ki}^2 \sum_k w_{kj}^2}}$$

If the tf-idf value is zero, it means none of the words in user inputs appears in our dataset. The chatbot has to reply with an apology to the user, and our system would record down these statements to ask system admin and web scraper to prepare the necessary FAQ or extract the related information from websites.

3.6. Web Application

Web application uses Django Web Framework along with CSS and JavaScript for these functions:

i. User interface for users to answer the survey questions before ushered into the venue and for people to check event related information via our chatbot
ii. The user interface is form based and requires user inputs for the decision tree to predict the result
iii. Users may navigate to the "About" tab to get event information (e.g. queue waiting time etc). Since we do not have event information as of now, we put our system introduction instead to promote it.
iv. Users may navigate to the chatbot tab to seek information with regards to the event and Covid-19 that is not stated on the website.

4. Challenge and Conclusion

4.1. challenges

4.1.1. Balancing work and project

The main challenge for our team is finding time to have meet-ups for project discussion because we have different work schedules (e.g. on different client sites, doing overtime and etc) .

4.1.2. First time using Django framework

This is the first time we are using the Django framework thus it took us some time to figure out how to use it and create a base framework for the application.

4.2. Future Improvements

If we have a longer timeframe to work on this project, we would have work on these areas:

4.2.1. Include database to store data

A database could be added to store the survey answer, result and personal information (e.g name and last 4 char of NRIC) of every participant. If one of the participants happened to be a covid-19 confirmed case, necessary information could be provided to the relevant authorities for contact tracing.

4.2.2. Real-time API for web scrape

Currently, the web scrape could only be triggered by schedule. With a real-time API and adding more relevant and trustable websites for web scrape, the user could be more informed of the latest news every time they ask a relevant question via the chatbot.

4.2.3. Chatbot icon to be shown at the bottom right of the webpage

Currently, the chatbot is within a tab. However, the ideal placement for the chatbot should be at the bottom right of every web page. The reasons being, most of the people are right-handed and they will not be comfortable with the bottom left placement because they are used to things being catered for right-handed people. In addition, having a chatbot not within the webpage, users might forget the question to ask regarding the content when they clicked on a new tab.

4.3. Conclusion

Our team has learned quite a few things while working on this project.

Technical wise, we picked-up Django and web scraping. Also, combining a few technical skills we learnt in the course to form a new one.

In addition, while completing this project, it shows us the importance of knowledge elicitation. Without it, the main features of this application will not be feasible. We used the knowledge to build the model for the Covid-19 prediction and utilised similarity-based reasoning on the knowledge base for the chatbot response.

Overall, it was a truly enjoyable process. We get to learn from one another and apply what we learned onto a solution which resolves an everyday problem in the current Covid-19 situation.

APPENDIX OF REPORT A

-

Project Proposal

GRADUATE CERTIFICATE: Intelligent Reasoning Systems (IRS)

PRACTICE MODULE: Project Proposal

Date of proposal:

1 May 2021

Project Title:

ISS Project – Intelligent Covid Detection and Chatbot Platform

Sponsor/Client: *(Name, Address, Telephone No. and Contact Name)*

Institute of Systems Science (ISS) at 25 Heng Mui Keng Terrace, Singapore
NATIONAL UNIVERSITY OF SINGAPORE (NUS)
Contact: Mr. GU ZHAN / Lecturer & Consultant
Telephone No.: 65-6516 8021
Email: zhan.gu@nus.edu.sg

Background/Aims/Objectives:

The proposed intelligent eco-system is aimed to integrate various advanced machine reasoning techniques to help screen vaccinated members of the public before they attend various events in Singapore, and to gather dynamic and heterogeneous information to build up an intelligent chatbot system to respond users' questions about COVID-19.

Requirements Overview:

- Research, data collection and cleaning ability
- Programming ability - python
- System integration ability – Integration of Django, JQuery, Javascript, HTML, CSS and machine learning libraries

Resource Requirements (please list Hardware, Software and any other resources)

Hardware proposed for consideration:

- CPU

Software proposed for consideration:

- Pertained machine learning models, e.g. NLP, decision tree
- machine learning tools, e.g. Python sklearn
- Chat-bots, e.g. ChatterBot
- Cognitive systems, e.g. nltk
- Web crawler, e.g. beautifulsoup4

- Web framework, e.g. Django

Number of Learner Interns required: (Please specify their tasks if possible)

a team of three project members. Their completed tasks are specified in their individual reports.

Methods and Standards:

Procedures	Objective	Key Activities
Requirement Gathering and Analysis	The team should meet with ISS to scope the details of project and ensure the achievement of business objectives.	1. Gather & Analyze Requirements 2. Define internal and External Design 3. Prioritize & Consolidate Requirements 4. Establish Functional Baseline
Technical Construction	To develop the source code in accordance to the design.	1. Setup Development Environment 2. Understand the System Context, Design 3. Perform Coding 4. Conduct Unit Testing
Integration Testing and acceptance testing	To ensure interface compatibility and confirm that the integrated system hardware and system software meets requirements and is ready for acceptance testing.	1. Prepare System Test Specifications 2. Prepare for Test Execution 3. Conduct System Integration Testing 4. Evaluate Testing 5. Establish Product Baseline
Acceptance Testing	To obtain ISS user acceptance that the system meets the requirements.	1. Plan for Acceptance Testing 2. Conduct Training for Acceptance Testing 3. Prepare for Acceptance Test Execution 4. ISS Evaluate Testing 5. Obtain Customer Acceptance Sign-off
Delivery	To deploy the system into production (ISS standalone server) environment.	1. Software must be packed by following ISS's standard 2. Deployment guideline must be provided in ISS production (ISS standalone server) format

		3. Production (ISS standalone server) support and troubleshooting process must be defined.

Team Formation & Registration

Team Name: 3Musketeers
Project Title (repeated): ISS Project – Intelligent Covid Detection and Chatbot Platform
System Name (if decided): CoviDetector
Team Member 1 Name: Hong Xiaohui
Team Member 1 NRIC Number: S9476943D
Team Member 1 Contact (Mobile/Email): +65-97805666 / xiaohui.hong@ncs.com.sg
Team Member 2 Name: Anita Koo Shi Qi
Team Member 2 NRIC Number: S9444480B
Team Member 2 Contact (Mobile/Email): +65-96216596 / anita.koo@ncs.com.sg
Team Member 3 Name: Sanjeven Ramakrishnan
Team Member 3 NRIC Number: S9139938E
Team Member 3 Contact (Mobile/Email): +65-98153769 / sanjeven.ramakrishnan@ncs.com.sg

For ISS Use Only		
Programme Name:	**Project No:**	**Learner Batch:**
Accepted/Rejected/KIV:		
Learners Assigned:		
Advisor Assigned: Contact: Mr. GU ZHAN / Lecturer & Consultant Telephone No.: 65-6516 8021 Email: zhan.gu@nus.edu.sg		

APPENDIX OF REPORT B -

Mapped System Functionalities against knowledge, techniques and skills of modular courses

Modular Courses	System Functionalities / Techniqe Applied
Machine Reasoning (MR)	● **Knowledge Elicitation and Extraction**: Web crawling from websites Manual extraction from Israeli MOH dataset ● **Business rule and Business process** Restrict rules to allow the vaccinated user to process the business rules ● **Constraint Satisfaction** Setup threshold for the similarity of knowledge chatbot response and auto-switch to the chatbot of content based filtering ● **Rule Based System**: Decision tree to derive rules from COVID-19 symptoms and user profile ● **Knowledge Discovery & Reasoning**: Knowledge Chatbot developed by Chatterbot based on FAQ worksheet ● **Knowledge Representation**: Decision tree diagram for COVID-19 symptoms Classification report and confusion matrix table for the training results Data representation in django SQLite database
Reasoning System (RS)	● **Knowledge Discovery & Transparency**: Decision trees and Decision rule learning, including One-Hot Encoding, on predicting covid based on symptoms Similarity-based Reasoning, including Similarity & Distance Measures by Cosine Similarity, on chatbot response from covid knowledge base
Cognitive System (CGS)	● **Cognitive System**: Chatbot developed by Nature Language Processing, tf-idf indexing and Similarity Measurements

APPENDIX OF REPORT C

-

Installation and User Guide

(Refer to separate document for Application & Deployment User Manual)

Requirements

Prerequisite

- Computer with internet access
- Python 3.6 installed

Recommended browsers

Our screening system supports the following browsers:

- Microsoft Edge version 90 and above
- Google Chrome version 90 and above
- Opera 75 and above

System Overview

Our Covid-19 event screener application is targeted at event organisers who would be managing crowd at large events. In our current setting in Singapore, event goers who have completed their vaccination do not have to go through the PCR test or any form of tests before entering the venue. However, there are cases where vaccinated personnel have caught the virus after vaccination. As such, our screener is targeted at participants who have completed their vaccination to serve as an extra layer of protection.

Besides providing screening, Our system also has a chatbot which serves as an FAQ bot answering questions related to Covid-19 or the event that is taking place.

Deployment

Our system is deployed to a windows server. In order to run our system locally, you would need python 3.6 installed along with the following packages. These packages can also be installed via the requirements files provided.

To get started, we can download the source code at https://github.com/xiaohuihong/IRS-PM-2021-01-16-IS03PT-GRP-3Musketeers-CoviDetector.git or via the git clone command below:

```
gh repo clone xiaohuihong/IRS-PM-2021-01-16-IS03PT-GRP-3Musketeers-CoviDetector
```

Navigate to \SystemCode\CovidDectector\webscrape folder. Install the required packages and start webscrapping. The script can be scheduled to run daily using window scheduler for window os or crontab on linux os.

```
pip install requirements.txt

python webscrape.py
```

Navigate to the folder \SystemCode\CoviDetector and we will then proceed to install the required packages and start up the server locally with the following commands:

```
pip install -r requirements.txt

python manage.py makemigrations

python manage.py migrate

python manage.py runserver 8000
```

User interface

Once the server is running, use your preferred browser to visit the link localhost:8000. We would recommend using Google chrome. We have three main sections to our application. Home, FAQ Bot and About.

Landing Page

This is the page users will first see when they access the application. From here, they are able to either start the questionnaire, access the FAQ bot or to read about the application.

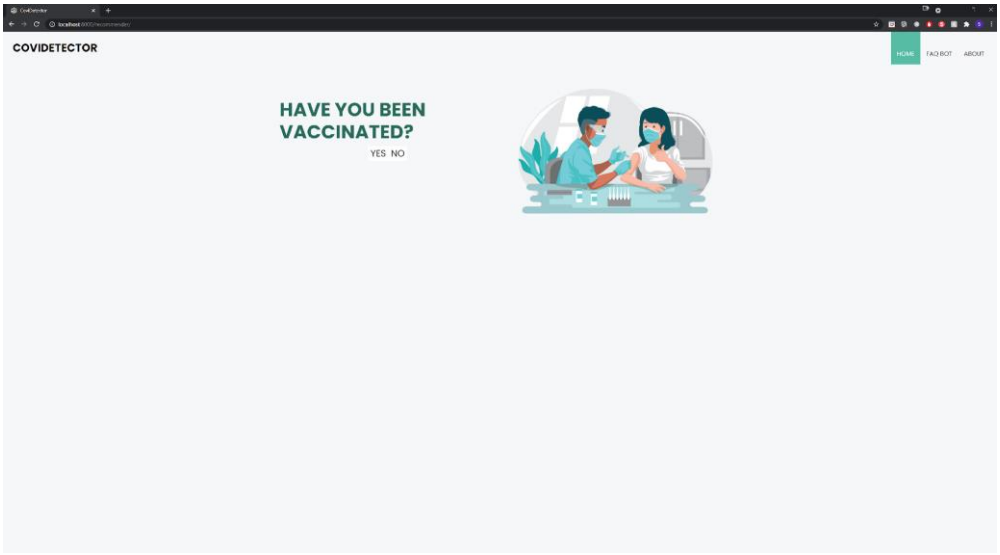


Image 1: Landing page

FAQ Bot

Here, users will have access to our FAQ bot and would be able to ask questions regarding Covid-19 or the event taking place.

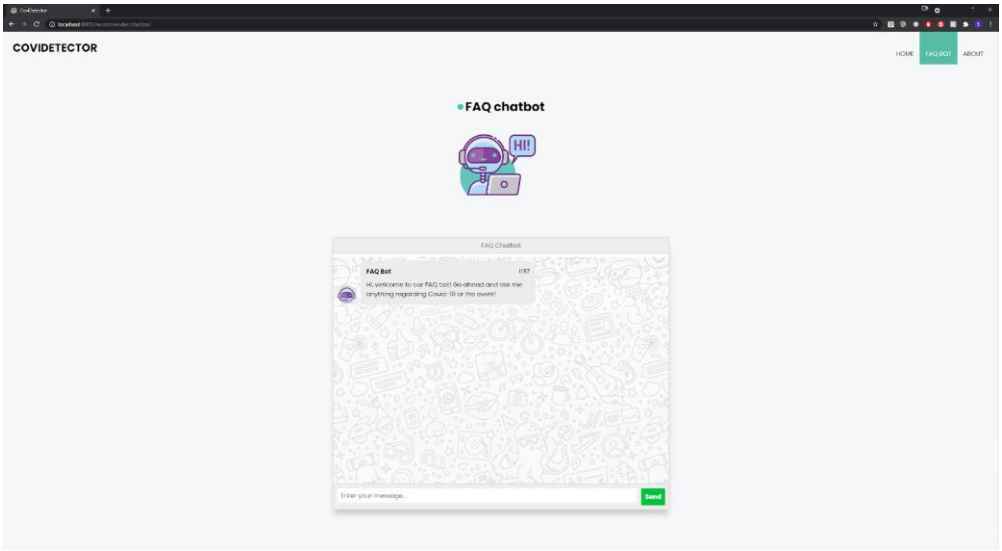


Image 2: FAQ Bot

About Page

Users will be able to read more about the project and the developers involved here.

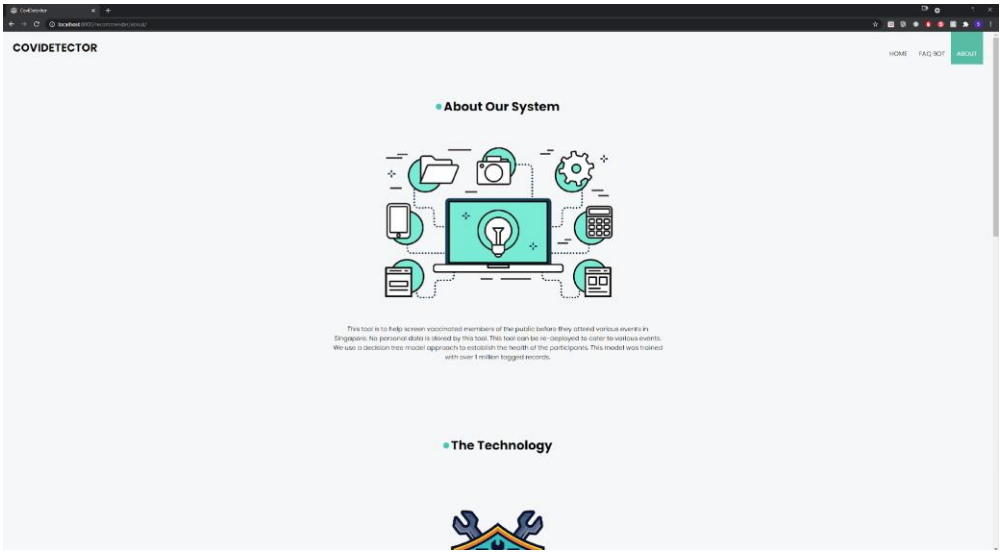
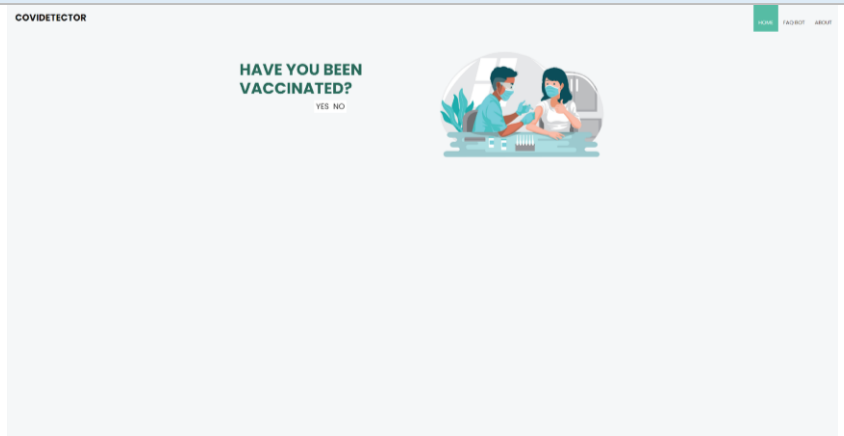
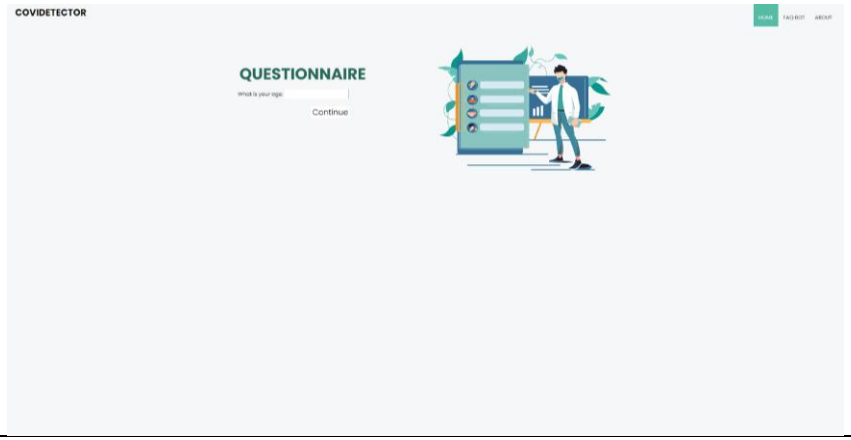


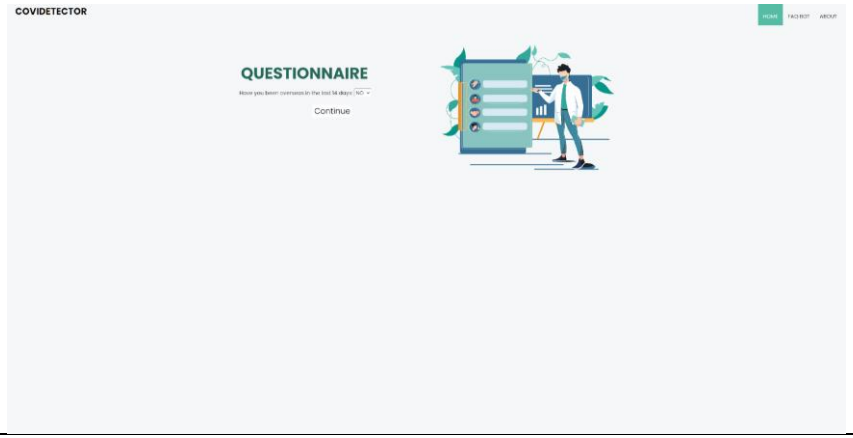
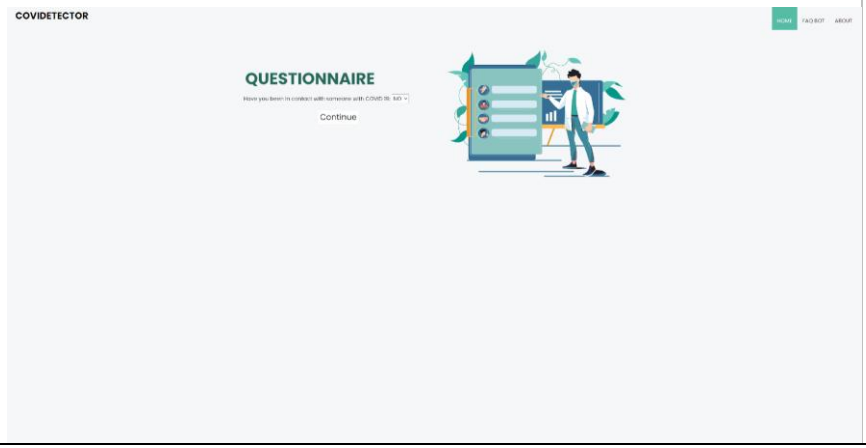
Image 3: About us

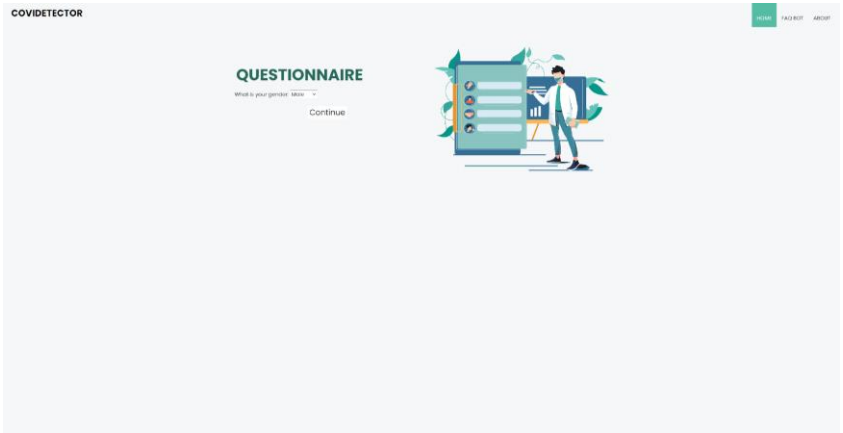
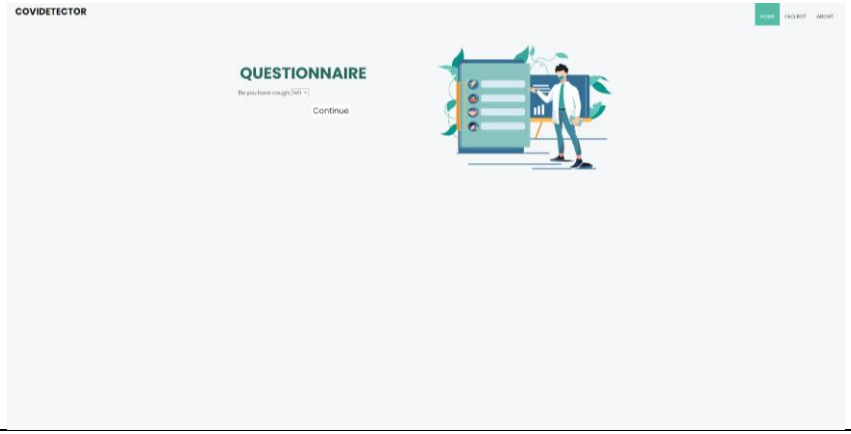
Questionnaire

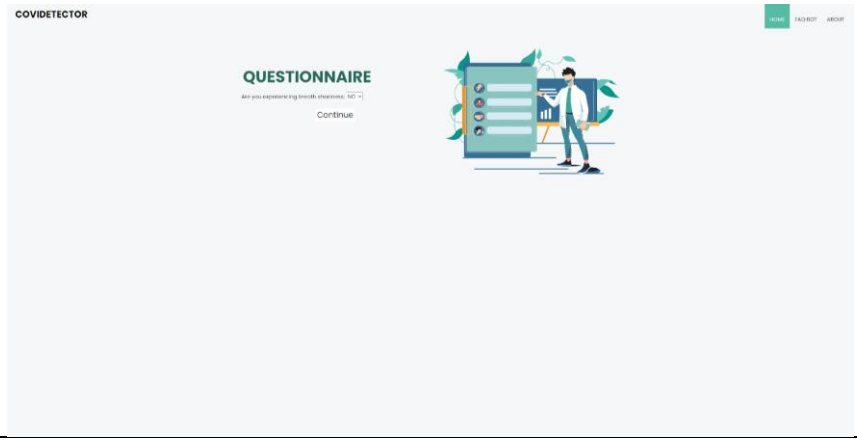
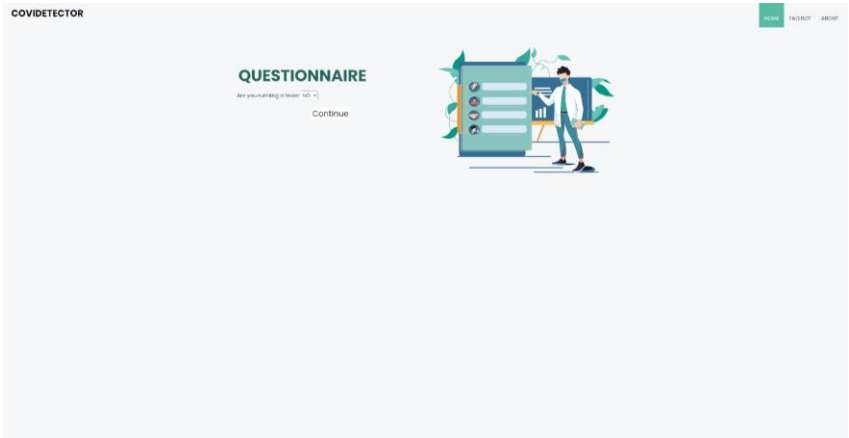
Users will go through the following questions:

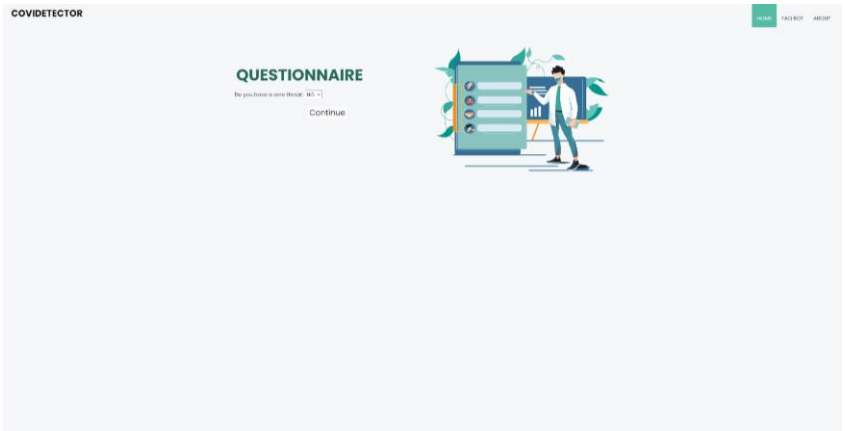
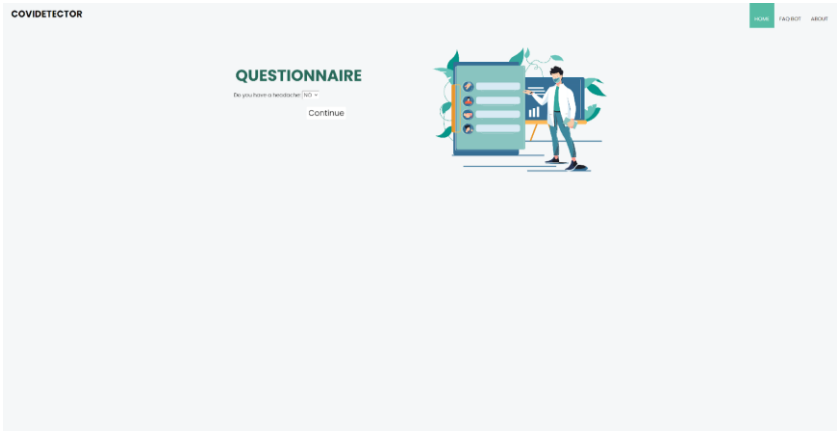
1. Have you been vaccinated?
2. What is your age?
3. Have you been overseas in the last 14 days?
4. Have you been in contact with someone with COVID-19?
5. What is your gender?
6. Do you have a cough?
7. Are you experiencing breath shortness?
8. Are you running a fever?
9. Do you have a sore throat?
10. Do you have a headache?

S/N	Question	User Interface	Possible answers
1	Have you been vaccinated?		Yes – Proceed with the questionnaire No – The questionnaire will end as we are targeting vaccinated participants. They will be directed to S/N 11
2	What is your age?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing

3	Have you been overseas in the last 14 days?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing
4	Have you been in contact with someone with COVID-19?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing

5	What is your gender?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing
6	Do you have cough?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing

7	Are you experiencing breath shortness?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing
8	Are you running a fever?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing

9	Do you have a sore throat?		
10	Do you have a headache?		Yes – Proceed with the questionnaire and response will be collected for processing No – Proceed with the questionnaire and response will be collected for processing

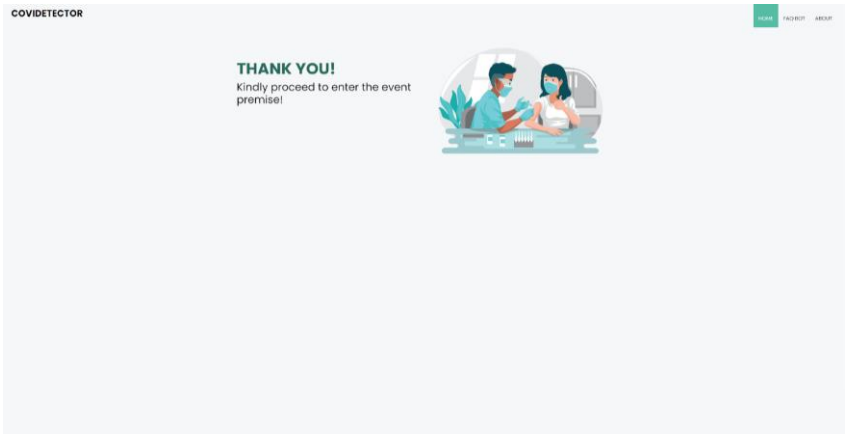
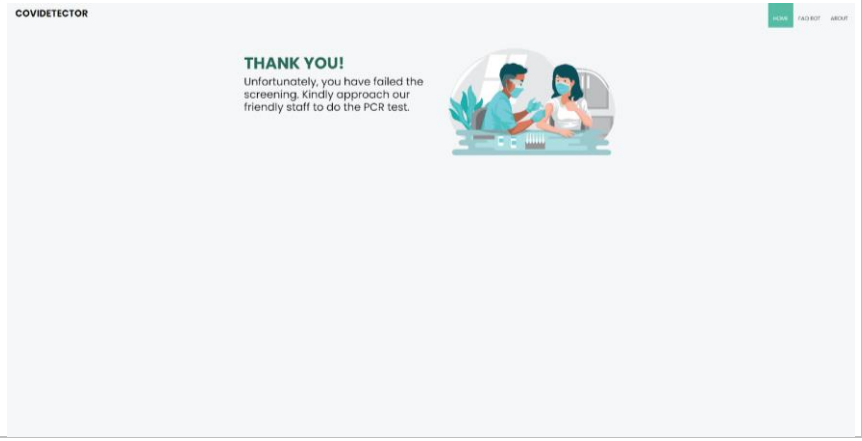
11	Results page	 	The results collected previously will be analysed based on our trained model and displays a results page.

Table 1: Depicting the questionnaire flo

FAQ Chatbot

The chat bot uses the previously scrapped data to display the latest and relevant information pertaining to Covid-19 and the event. Users will be able to use the chat bot by navigating to the FAQ bot tab in the top right and key in their questions in the space provided.

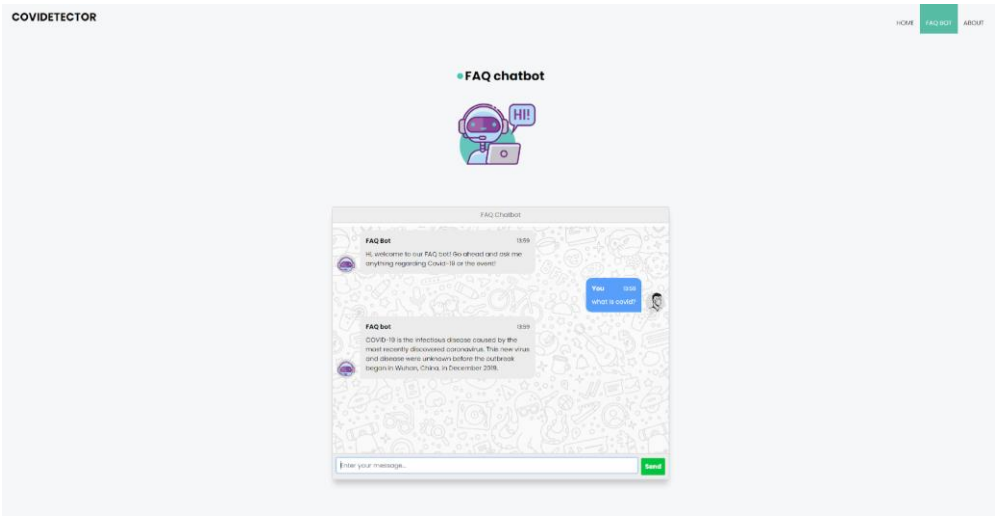


Image 4: Chatbot replying to covid related questions

As our solution is dynamic we are able to change the events content to suit each individual event. For demonstration purposes, we are using a Westlife concert as an example.

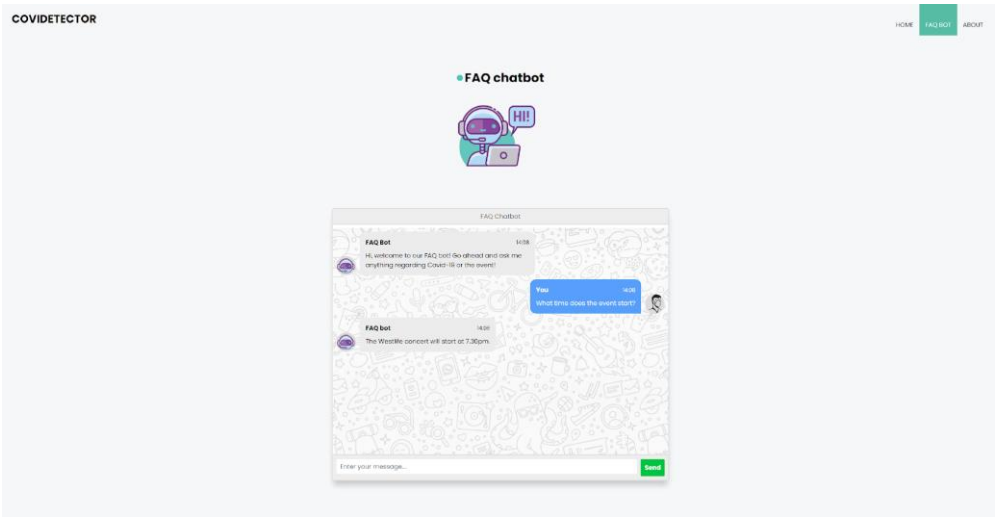


Image 5: Chatbot replying to events questions

APPENDIX OF REPORT D

-

Individual Reports

Individual Report by Hong Xiaohui (S9476943D)

1). Personal contribution to the project

As a team lead of this project, I am responsible for the following items:

a. Project topic and idea generation

b. System architecture design, development and implementation including:

i. Django web service backend and frontend setup, e.g. data schema, questionnaire workflow webpages and integration of predictive model, chatbot and backend codes.

ii. Training dataset and knowledge base selection, e.g. covid-19 symptoms dataset and FAQ worksheet preparation.

iii. Predictive model build for Covid-19 detection, e.g. decision tree and rules model

iv. FAQ chatbot build for both chatterbot and natural language processing

c. Project report generation

i. Section System Development & Implementation

ii. Project Proposal

iii. Mapped System Functionalities against knowledge, techniques and skills of modular courses

2). What learnt is the most useful for you

Throughout this project module, I have learnt quite a number of useful things:

a. Django web framework. In order to do this project, I had in-depth understanding on how to utilize different features of it such as multi-page forms, views and models.

b. Predictive model. Although at the end we chose the decision tree model, we still went through different types of predictive models we learnt from the course, Reasoning System. It helped enhance our understanding and our programming skill about how to use scikit-learn API to deploy these algorithms.

c. Natural Language Processing. While building the chatbot, I learnt more in tokenization, case lowering, punctuation removal and lemmatization.

d. Similarity Measurements. While developing the chatbot, two similarity measurements, Jaccard Similarity and Cosine Similarity, were implemented. I learnt how to use scikit-learn library and chatterbot library to apply these algorithms.

e. System Integration. The last thing I learnt from this project is how to integrate a web service framework with machine learning models to make a real product.

3). How you can apply the knowledge and skills in other situations or your workplaces

Since the Covid-19 pandemic happened from 2020, applications about it in different fields have sprung up rapidly. The experience of processing real Covid-19 data is valuable and precious for a data engineer like me in the career path or job market. In the future, if I need to process the similar type of dataset, this project is a quite good reference and I can also apply the same model to those datasets.

With the chatbot development knowledge that I have learnt, I would also apply the knowledge in helping my company to develop the similar chatbot in some projects. For example, as a data engineer, I am responsible for building data lake systems. I can develop a chatbot to answer customers' questions about how to use our system.

Individual Report by Anita Koo Shi Qi (S9444480B)

1). Personal contribution to the project

I have contributed to the followings:

- Add-on idea towards the main idea (covid detector)
- Code: Web scrape
- Project report generation
    - Abstract and Business Case
    - Section - System Model
    - Section - System Development & Implementation
    - Section - Challenge and Conclusion
- Video for System Design

2). What learnt is the most useful for you

The importance of knowledge elicitation. Without a valid knowledge base, an intelligent system is not possible to be created. As a data engineer, I had processed various typeof data. Often, I ponder "why is this data needed? It seems insignificant". However, through the process of this project, my thoughts started to change. Instead, I started thinking scenarios where such data are useful , using different algorithms. With different scenarios of the data usage, one might be able to spot missing data that will be required to fulfil the scenarios thus expanding the knowledge base.

3). How you can apply the knowledge and skills in other situations or your workplaces

From the viewpoint of a data engineer and someone who interacts with the user, I would be able to suggest the type of information that the user might be interested in for their use cases and where to get those information.

Internally in my workplace and externally on client sites, the chatbot that was created in this project can be used for different aspects.

Externally, the system operation guide document we created for a project could act as the knowledge base for the chatbot. Our user could just type their queries (e.g how to stop a service) as an input and the chabot will output the instructions/commands to run.

Internally, this chatbot can be re-use as

- a FAQ chatbot for company's welfare, events etc.
- a chatbot which provides solutions to common issues faced by the team on different projects. (the knowledge base could be past jira tickets from the previous and current projects)
- a chatbot that recommends solution/system architecture base on the similarity of the requirements from past solution and system we implemented

Individual Report by Sanjeven Ramakrishnan (S9139938E)

1). Personal contribution to the project

I contributed to the following:

- Business implementation plan and use case to address business needs in our current covid-19 situation
- Did the integration of the various components into the application and did the user interfaces for the project
- Project report generation
- Appendix D - User and installation guide
    - Section - Installation
        - Requirements
        - System Overview
        - Deployment
    - Section - User guide
        - Questionnaire
        - FAQ Bot
- Video for project proposal and use case example

2). What learnt is the most useful for you

In this project, I learned that systems integration is an important aspect in any project. To make the application operate and for the look and feel to be seamless, I had to pick up the Django framework, CSS and javascript to make the application user friendly. This allowed me to deep dive on user interaction components of the project and to tackle problems that would affect a users experience. I feel that this is an important skill to have as it brings utility to an idea.

I also learned to work with video editing tools. This is a useful skill to have as with our current situation where most meetings/ proposals are done online, promoting our product via a video is required and learning how to make attention grabbing videos is a skill I would like to learn further.

3). How you can apply the knowledge and skills in other situations or your workplaces

In my daily work environment, I need to make sure that all the softwares used by our application is compatible. I usually do this by looking at the compatibility matrix. However, this is not always the case. I am now able to reason as to why these software is compatible and what steps I need to take to overcome compatibility issues if it does not work.

The questionnaire can be repurposed to be used as an extra checkpoint for personnel who would need access to the various data centers. As data centers are dense, the probability of human contact or the spread of the virus is very likely. As such having an extra layer of protection would be good to protect the other engineers using the data center.

The chatbot designed in this project is reusable. As such, this chatbot can also be used to compliment our company's internal events in the future. It can also be repurposed to be applied to other business entities such as HR, L&D and scheduling to help improve our company's process flow.