

GRADUATE CERTIFICATE INTELLIGENT REASONING SYSTEMS
PRACTICE MODULE REPORT

Virtual Renting Assistant

T E A M M E M B E R S

ZUO ZONGYUAN

ZHANG JIUYUN

CAO ZIHAO

WANG SIXIANG

M A S T E R O F T E C H N O L O G Y

CONTENTS

1 EXECUTIVE SUMMARY.....	2
1.1 Problem Description.....	2
1.2 Market Research.....	2
1.3 Business Case.....	4
2 SYSTEM SOLUTION.....	4
2.1 Project Objective.....	5
2.2 System Architecture.....	5
2.3 System Workflow.....	10
3 SYSTEM IPLEMENTATION.....	11
3.1 Dialogflow.....	11
3.1.1 Conversation Workflow.....	12
3.1.2 Intent Definition.....	13
3.2 Interactive Service.....	14
3.2.1 Users describe their desired residence.....	14
3.2.2 Receive inbound message through Twilio webhook.....	14
3.2.3 Detect user indent.....	15
3.2.4 Scrape online residence data.....	16
3.2.5 Send outbound message to user.....	16
3.3 Clustering.....	16
3.3.1 Web Scraper.....	16
3.3.2 Data Preparation.....	18
3.3.3 Clustering Modelling.....	19
4 APPENDIX.....	20
4.1 Installation and User Guide.....	20
4.2 Individual Project Report.....	21

1. EXECUTIVE SUMMARY

Finding a suitable residence is one of the basic necessities that everyone needs. Currently there are various websites and applications which can list all the houses' information within a district. However, the explosion of housing information arises troubles not only for housing tenants but also sellers. To help tenants select a suitable residents without exposure to oceans of information and also reduce the heavy load of sellers, we design and develop a Virtual Renting Assistant (VRA) using Google Dialogflow, Twilio sandbox, website scraper and clustering analysis.

1.1 Problem Description

Nowadays the housing prices in Singapore has become increasingly expensive, which lead to the fact that more and more people have to rent house. In order to help tenants find a suitable residence, lots of real estate companies now will list the house information on various websites. However, the flood of the unstructured lease information can increase the time of finding useful information and waste the users' time. Besides, a majority group of these tenants are foreign students who may have trouble accessing these websites, let alone extracting information. Moreover, the demand of online renting could witness a huge increase in these two years considering the limited physical interaction among people due to COVID-19.

On the other hand, the cost of human labor for houses renting could be huge for the real estate companies. For each property listed on the websites, there would be one house manager whose responsibility is for further contact with the customers. In such a condition the number of property managers could be large thus contributing to a high cost. Meanwhile, popular renting websites such as propertyguru and stproperty do not have virtual assistant or agent, which implies a potential market for the virtual renting agent.

1.2 Market Research

The demand of housing renting is stable and huge considering the fact that every

year lots of foreign students will come to Singapore and employees may change their residence if the lease expires. As shown in Figure 1 (source: Singapore Housing & Development Board statistics), the number of houses for rent is more than thirty thousand. What's more, according to the Singapore Government HDB statistics, the amount of housing lease varies from 41351(2015), 44530(2016), 42887(2017) to 46440(2018) and 48195(2019). The market in housing renting has increased by 16% from 2015 to 2019, which presents a promising prospect in this field.

2020

Quarter	1-Room	2-Room	3-Room	4-Room	5-Room	Executive	Total
1 st	2	182	3,716	4,046	2,939	706	11,591
2 nd	6	128	3,278	3,735	2,712	680	10,539
3 rd	2	140	2,823	2,880	1,904	447	8,196
Total	10	450	9,817	10,661	7,555	1,833	30,326

However, the increasing demand of housing renting leads to a higher cost in human labor including property manager and human agents. Figure 2 provides the number of property agents from 2018 to 2020 (source: Singapore Government Council for Estate Agencies).

Number of Property Agents

Year (as at 1 Jan)	Number of Property Agents
2020	30,073
2019	29,146
2018	28,571

Number of Property Agencies

Year (as at 1 Jan)	Number of Property Agencies
2020	1,181
2019	1,229
2018	1,269

As shown in the two charts, the number of agents is rising while the amount for agencies is decreasing. This decrease implies that the increasing cost in human agents do bring troubles to the operation of agencies. Additionally, taking the average salary of property manager as 4600 \$ per month (source indeed salary) then cost for salary is more than one billion! Therefore, if there are robot agents that can replace some

human agents, then the cost for the real estate agencies could be much less. Unfortunately, currently few websites have their own virtual intelligent agents. Common renting websites such as PropertyGuru, STProperty provides a contact number for every rental house but the search for suitable residence on the websites still requires manual efforts.

1.3 Business Case

Joey is a Chinese student who have admitted into NUS ISS department for further study in 2021 spring semester. Nowadays he is busy with finding an apartment near the ISS department. However, he has no idea which website he should visit and also the access to Singapore websites is limited due to policies.

Then with the help of some VPN software, he finally logs into one house renting website. However, the web page confuses him with too many choices and unstructured data. He spends lots of time seeking the apartment one by one and compares them based on the confusing and plethoric information. Unluckily, he does not find one that satisfies his requirement. To make things even worse, he heard that there are some policies limiting the physical interaction among people, which means he cannot arrive at Singapore and find the residence by himself. Now Joey has no choice but to repeat the search process once again.

Fortunately, our Virtual Renting Assistant (VRA) can help Joey find a satisfying residence through simple conversation. The VRA is able to filter the houses by location, prices and users' preferences in the property type and built year. After filling these slots, the assistant will return a PDF file with detailed information including the google map image, the advantage of the house and some indoor images. So Joey simply types the location as around NUS and the price with around 1000, then the VRA will return corresponding recommended houses quickly with structured information.

2. SYSTEM SOLUTION

As mentioned above, searching a proper renting house on the websites could be troublesome and time-consuming for the users. To address this problem, we develop the Virtual Renting Assistant (VRA) to automatically select a dream residence for users only through chatting.

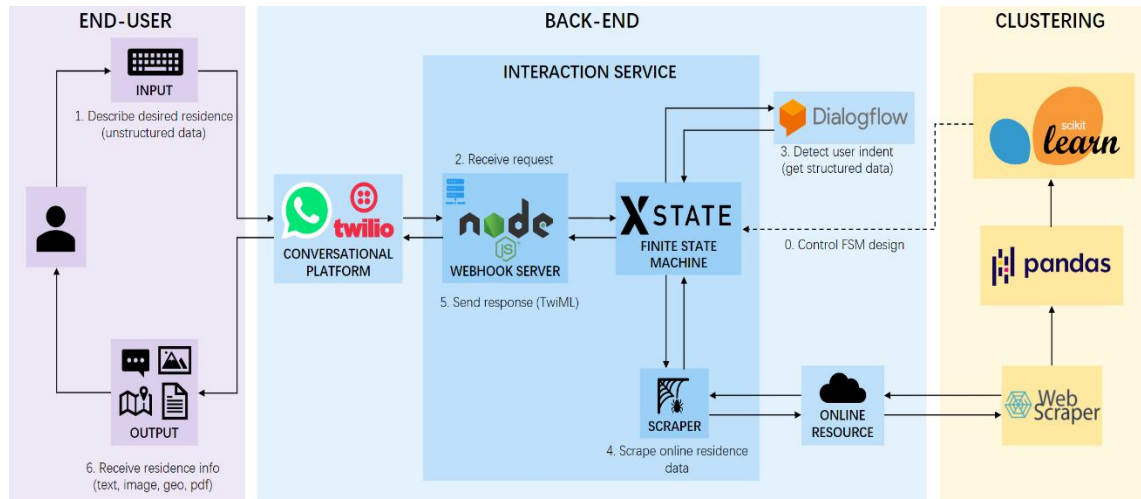
To achieve this goal, our system requires a frontend user interface allowing utterance with the users, a natural language processing (NLP) model for understanding users, a backend service for scraping useful house information. After checking some documents and previous work, we utilize Whatsapp and Twilio Sandbox as our frontend, Python Flask as our server frame, Google Dialogflow as our NLP model and web scraping for selecting and extracting information. Particularly, we regard STProperty as our testing websites.

2.1 Project Objective

Our project hopes to introduce the robotic process automation (RPA) and intelligent process automation (IPA) techniques into the field of housing renting. Specifically, we develop a virtual assistant which is capable of chatting with human, understanding their demands and preferences and extracting critical housing information for them. Besides, our assistant can have a deep understanding of the house features and is able to classify the important ones. In such a way the assistant can match the users' preferences to different types of houses and fit the users' requirements better.

2.2 System Architecture

As shown in the system architecture, our system is mainly composed of three parts: application layer, service layer and clustering model.



• End-User

Once the users greet our assistant in the chatbox, the VRA will guide them through conversations to complete the key information for renting house (price, location). These unstructured data will be handled by the virtual assistant and return them with recommended house information including geographic images (Google map), indoor images and PDF file. The returned images and PDF information are shown below.

Location

Blk 607 Clementi West Street 1 (Clementi), HDB 3 Rooms - For Rent

Room Rental, Common Room For Rent

- Fully furnished with Queen size bed and aircon
- Landlord prefer female tenants only
- Light cooking
- \$900 for 2 persons

For viewing appointment, please call/WhatsApp Debbie at 94247688.

Information	
Price	\$900
Size	67 sqm, 721.18 sqft
Room Type	N.A
Lease Term	1 year
Furnish	Fully Furnished
Take Over	No
Date Listed	1605968932
Facilities	
Address	607 Clementi West Street 1 (120607)
Property Name	Clementi West Street 1
Property Type	HDB 3 Rooms
Model	New Generation
Tenure	LEASEHOLD/99 years
Built Year	1979
HDB Town	Clementi

• Conversation Platform

The VRA is indeed a chatting robot which requires a friendly interface for communication with human. Currently there are multiple popular messaging software and application including Telegram, Wechat, Facebook Messenger and Whatsapp. Upon these platforms, Telegram is usually regarded as a notification application for messaging with official organizations such as government or college. On the other hand, Wechat has limited access to chatting bots since the web version has been shut

down, which would increase the risk of disconnection while chatting with users. Facebook messenger has a few nifty features which may be able to display the images better using carousel, however, after doing a market survey with classmates and friends around, more people use Whatsapp compared to Facebook Messenger as their chatting software. Therefore, we choose Whatsapp as our frontend platform. The comparison of these platforms is shown in the following table.

Platform	Popularity	Limited Access	Easy to Develop
Whatsapp	Used as communication tool in usual life	Have legal policies upon chatting bot	Have official business API for robot agent development
Telegram	More as a communication software with official organization	Free access	Intergrated in Google Dialogflow
Wechat	Only popular among Chinese students	Strict about the chatting bot, hard to live	Many previous applications on it
Messenger	Less used in Singapore, but popular in Europe and US	Free access	Integrated in Google Dialogflow

However, the business application API of Whatsapp is not free and requires several months to review the developers' profiles. Additionally, the Google Dialogflow does not support the direct integration with Whatsapp anymore from this year, considering these facts, we utilize a free the third-party tool called Twilio Sandbox which is a pre-configured environment available through the Twilio website in which users can prototype sending outbound messages, replying to incoming messages, and configuring things like message delivery callbacks. This tool gives us opportunities of skipping the step of waiting for WhatsApp Business Profile to be reviewed. It's a quick way to test out the functionality of the WhatsApp Business API with Twilio.

In our system we use Twilio Sandbox as a mid-ware for receiving and replying messages between Whatsapp and backend server. This is because the Sandbox's configuration supports a customized message handling method by providing our server address to its configuration. In such a way, every time a new message comes in ,

the message will transfer to our server and we can handle the message by ourselves.

- **Dialogflow**

Once our server receives the message, we firstly need to understand the texts and extract key information in it. To do this we utilize Google Dialogflow as our NLP model to classify users' intents and filling corresponding slots. By utilizing the well-configured Google Dialogflow, we don't have to train the complicated NLP model ourselves and deploy it manually into our server. Besides, we can exploit Google's huge databases of existing entities for items such as Countries/Cities and DateTime. This relieves us of crafting our own entities manually, making it easier to define the rules to trigger the intent. Dialogflow is also user-friendly and easy to use over a web interface. One of the drawbacks of using Dialogflow is that Whatsapp is not supported as a third-party integration anymore from this year. As such, we add another tool of Twilio Sandbox to solve this problem.

- **Interaction Service**

Our server side is composed of three parts, one for handing the whole process and interacting with the Twilio Sandbox and another one for scraping the websites.

- **Web Scraping**

We use Nodejs as our webhook server to interact with the Twilio messages and scrape critical information from house renting websites. There are several popular server frames currently including Python Flask and Django. However, considering the speed and flexibility of scraping websites, we decide to use Nodejs which can build scalable network applications with asynchronous APIs easily and conveniently.

To manage the whole conversation process, we apply the XState library to control the states. XState is a library for creating, interpreting, and executing finite state machines and statecharts, as well as managing invocations of those machines as actors.

The states of the conversation process are shown as follows:

No.	State
0	Initial
1	Location Filled
2	Price Filled
3	Preference Filled

4	House Recommended
5	User Selected
6	Detail Files Sent
7	End

For scraping online websites information, we check the structure of STProperty and implement the scraper using Javascript. One reason we decide to use Javascript considers the fact almost every modern webpage contains it. JavaScript is a scripting or programming language that allows us to implement complex features on web pages, it is the third layer of the layer cake of standard web technologies, including the other two popular techniques (HTML and CSS). Another reason is that since we apply Nodejs as our server frame, Javascript could fit the frame in a natural way.

- **Clustering**

In order to help the VRA understand the importance of house features (eg, property type, built year) and the groups of the houses, we conduct a clustering analysis to the websites data.

Firstly we utilize a Google extension called Web Scraper to extract data (545 records with 8 features) that we want from STProperty website. Then we process the data using Pandas and build our model with Kmeans. After referring to the Silhouette Coefficient and error elbow plot, the number of clusters is 5. **The detailed calculation and validation can be found in chapter 3.** Finally we derive the clustering profile using the results (centroids of the clusters) and classify the houses into different clusters using several features.

However, due to the limitation of time, we do not deploy the clustering model to websites. In contrast, we directly use the results of clustering shown as following table.

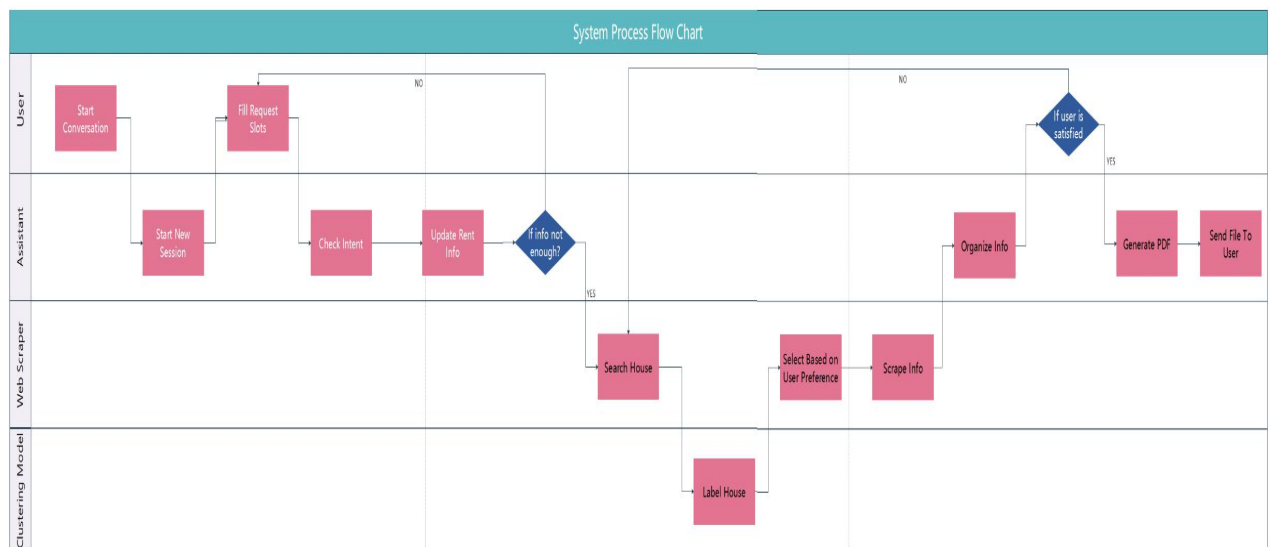
Cluster Profile	Property Type	Price	Built Year (Old/New)
Cluster #1	Condo	High	New
Cluster #2	HDB	Low	Old
Cluster #3	HDB	Low	New
Cluster #4	Apartment	High	New
Cluster #5	Terrace/Detached	High	Old

From the table we can conclude that with only three features we are able to classify the clusters, which means that despite so many features of the houses, the

actual factors that influencing the classes of houses are property type, price and built year. Therefore, we set three more preference questions in the chat process to check the users' preference. Then the virtual assistant can filter the searching results based on the preference and provide more suitable choices for users.

2.3 System Workflow

As shown in the system workflow, there are four roles in our chatting process.



➤ Start Conversation (Initial State)

Users start the conversation by greeting with our chat bots

➤ Start New Session (Initial State)

Once the user greet with the virtual assistant, the chatbot will start a new session in the server side and store the users id and other relevant information for further communication. Then the greeting message will come to Dialogflow and detect the intent (welcome intent) and several sentences including welcome message will be sent to users.

➤ Fill Request Slots-Check Intent-Update Info-Check If info is enough

After the session is created, the robot agent will ask users several questions about the location, price and preference (property type, built-year) of the house. Every time the users send a response, the assistant will detect the intent of the response and update corresponding slots in the server about the user and also update the state of

current conversation (**from location filled state to preference filled state**).

After the users' responses are sent, the state machine in the backend will check if all the slots are filled and also the state. If not completed, then the assistant will ask the users questions once again. If completed, then the assistant will search the STProperty website based on the slots.

➤ **Search House-Label House-Select based on Preference-Scrape Info-Organize Info**

The completion of preference filled state will fire the scraper. The scraper will firstly search the houses which match the requirement of users and collect a list of potential houses. Then based on the clustering model, the scraper will filter the houses further based on users' preference. Once the candidate is selected, the scraper will collect the detailed information and the backend will organize them into structured data in the format of images and descriptions per house candidate. Finally the assistant will reply recommended house list to the user and **update the state to House Recommended**.

➤ **If user is satisfied-Generate PDF-Send File to User**

Once the candidate houses are sent, users can choose the one which they are interested in. By choosing one house, the assistant will generate a PDF file containing all the detailed information including the advantage of the house, the google map image, indoor image and so on. **The state will come from User Selected to File Sent and end state** in this branch.

However, if the users is not satisfied with the response, **the state will go back to House Recommended state** and repeat the search process again until find a suitable residence for user.

After the PDF file is sent, the agent will send a goodbye message and stop the session.

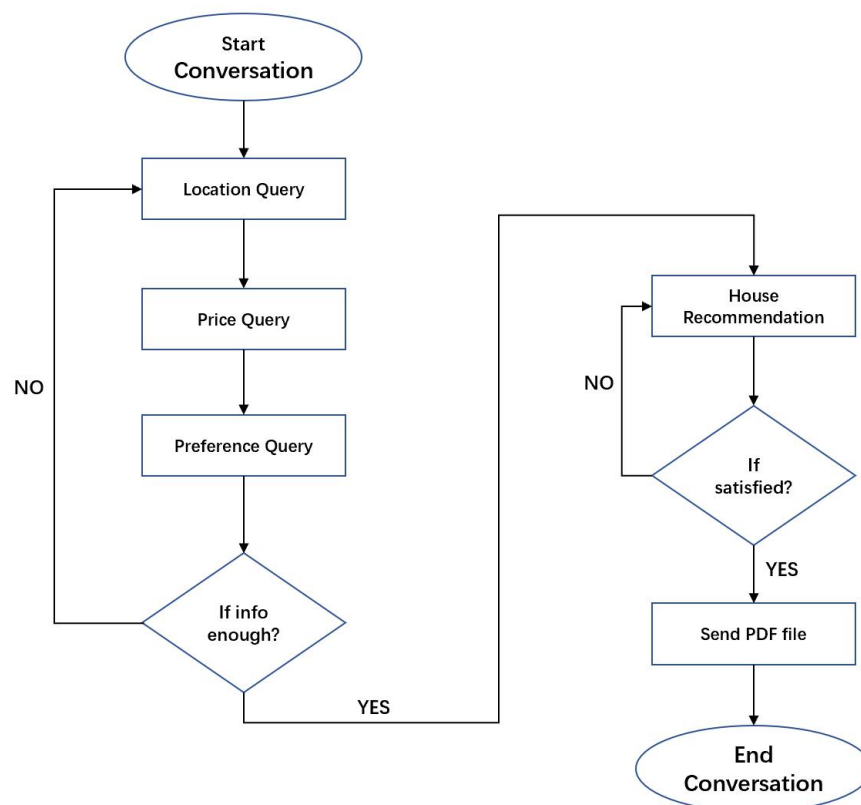
3. SYSTEM IPLEMENTATION

3.1 Dialogflow

Our virtual assistant utilizes the Google Dialogflow as our NLP model to detect users' intents and extract slots. Therefore, we need to design a stable and meaningful conversation flow to ensure a good user experience. Based on the conversation, we define several intents and corresponding entities for identifying slots better.

3.1.1 Conversation Workflow

The figure below shows our Whatsapp conversation workflow. First of all, the user can invoke the virtual assistant by saying welcome message such as “Hello”, then the assistant will guide the users to finish necessary queries through conversation. The assistant needs to know the location, price of the house and the property type and built year (old or new). After all the mandatory information is completed, the chatbot will search and scrape detail information of relevant houses and return a list of candidates for users' selection. If the users are not satisfied with the recommendations, then the agent will repeat the search process again until a good residence is found. Finally, the agent will organize the detail information in a generated PDF file and send back to user for their further use and end the conversation.



3.1.2 Intent Definition

The table below shows the mandatory information needed for searching a residence, for example location, price, property type, built year (old, new). When asking for the request, user needs to mention relevant slots in their request text. Our system will try to recognize all the information we need.

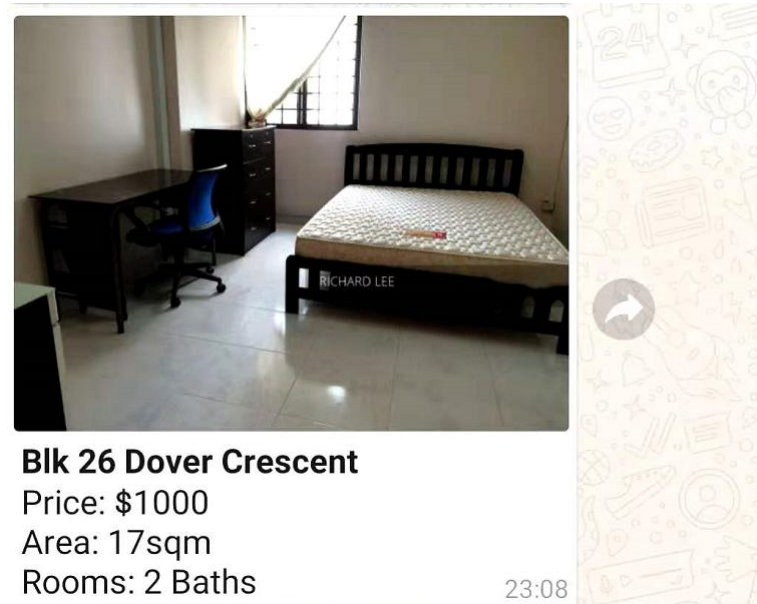
Intent	Slot	Description	Example
Welcome	/	Welcome message for users	Hello, Hi
Location	Location Entity	Identify the house location and surroundings	Around NUS Jurong East
Price	Integer Entity	Identify the range of rent that the user can afford	From 700 to 1000 Below 1000 Above 700
Property Type	Customized Property Entity	The type of residence: condo, apartment, HDB, detached or terrace	Condo, semi-detached
Built year	Date Entity	The construction time of the residence, representing an old or new residence	After 2000 Around 2010
Fallback	/	When users' input is not recognized, the agent will ask the query again	Could you say it again?
End	/	End the conversation	Hope it helps, bye~

Here are some sample texts users can use for searching houses:

- User: Hello.
- Agent: Hello, this is Joey, your renting virtual assistant, I am here to help you find a dream residence~ I am still a very young robot, so please type one question at a time. Now let's start! Could you please describe your criteria? (e.g. location, price)
- User: Around NUS
- Agent: How much are you willing to pay for the tenancy?
- User: From 700 to 1000.
(Agent returns a list of residence candidates)
- Agent: Which one would you like to learn more? Please type its name or number (e.g. /1). If none of these residences match your need, you can type /next to other

candidates, or type /search to adjust your criteria.

(Below is the returned list with a fixed format: Residence Image + Residence Name + Price + Room Area +Room Facilities.)



3.2 Interactive Service

3.2.1 Users describe their desired residence

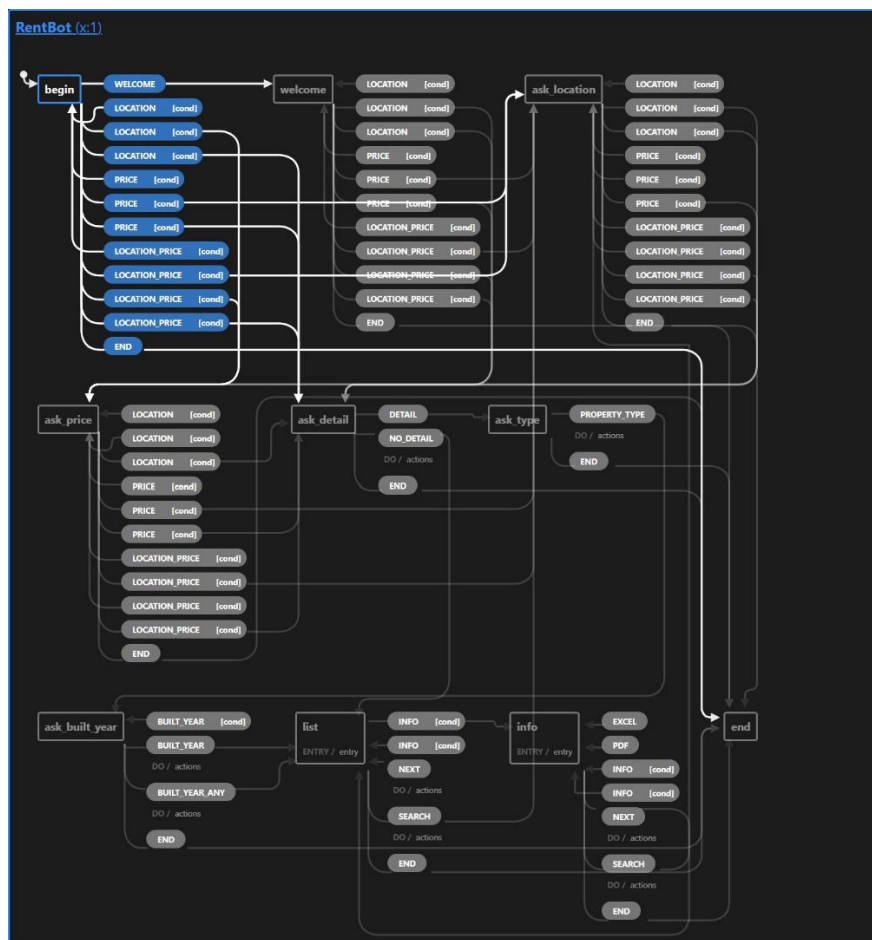
We use WhatsApp as the conversational platform of our chatbot. WhatsApp is world's most popular messaging app. With the help of Twilio Programmable Messaging API and Twilio Markup Language (Twiml), we can receive and respond to user messages on WhatsApp. User can simply send a greeting, and our bot will guide user to provide their requirements through Conversational UI.

3.2.2 Receive inbound message through Twilio webhook

Twilio use webhook to notify the application when events happen, in this case receiving an inbound WhatsApp message. We wrote a web server with Node.js to receive and parse the POST request from Twilio. In order to identify different users, we create a session for each user when they start the conversation, and remove the session when the conversation ends.

3.2.3 Detect user indent

The message we received from user is generally unstructured, so we utilize Google Dialogflow to handle the user message with NLP. When received a message from user, our system first check whether the message is one of the predefined commands (starts with a slash). If not, our system will send the text to Dialogflow, detect the user indent (what type the message is), and extract key information (structured data) from the text. In order to better control the conversation, we only use the basic feature of Dialogflow (detect indent), and map our control execution flow, or business process, to a Finite State Machine, and let the FSM do the rest. The structure of the FSM is shown in Figure 0. The FSM will then use indent type as event and extracted data as parameters to transition from current state to next state, and trigger corresponding actions (e.g., ask further details of user requirement, return matched residences). After user claim all their requirements, the FSM will then enter “*list*” state and pass all user demands to the web scraper.



3.2.4 Scrape online residence data

Since JavaScript natively supports manipulating the Document Object Model (DOM), there is no overhead of learning the syntax of new libraries. We only installed one additional module to bring the WHATWG DOM and HTML Standards from browser to Node.js. Depends on user instruction, sometimes we need to provide them a document containing detail information of the residence. We use pdfmake to compose the scraped data into PDF file.

3.2.5 Send outbound message to user

As stated in 3.2.2, our web server will receive POST request from Twilio when user send us a message. Now that we have already composed our response from either predefined in the FSM (e.g., further instructions) or the web scraper, we need to send a message back to user. So, our web server will serialize the composed message into TwiML format, and use it as the response body to the previous POST request.

3.3 Clustering

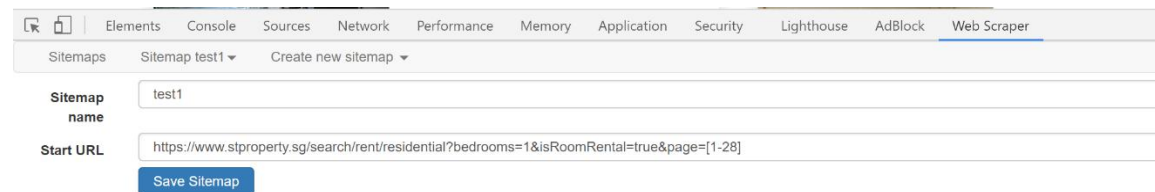
To do the clustering analysis on the residence data, we firstly need to scrape data from STProperty website using Google extension web scraper, then we utilize Panda library to prepare the data for further modelling. Finally we derive the clusters using Scikit-learn Kmeans model.

3.3.1 Web Scraper

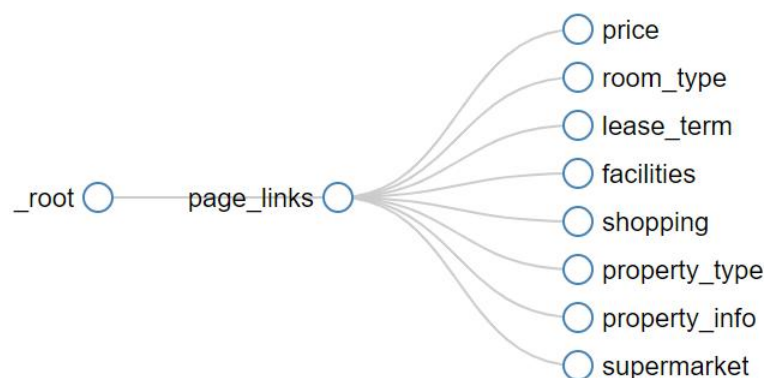
Web scraper is a tool to make web data extraction as simple as possible. Configure scraper by simply pointing and clicking on elements. No coding is required for extracting the data. Besides, Web Scraper can extract data from sites with multiple levels of navigation. It can navigate a website on all levels including categories and subcategories, pagination or product pages. At the same time, Web Scraper allows us to build Site Maps from different types of selectors including HTML element, text, links and so on. This system makes it possible to tailor data extraction to different site

structures. We can also export the extracted data into CSV file for further use.

To extract the data from STProperty, we configure the Web Scraper into three level: web page - each page - each house. There are 28 pages in the websites for single renting room, on each page there will be multiple house links and each link will provide a detail information for the house. Therefore, we design the URL of our websites to fit all the page as follows:



Then for each page, we use the pagination feature of Web Scraper to handle the two-level structure. The scraping structure shows as follows. The Scraper will start from root and iterate over all the page links within each web page. For each web link, the Scraper will extract certain information based on the selectors.



To add a new selector, the Scraper only needs to click the element on the web page and choose corresponding selector type such as text. Our selectors show as follows:

ID	Selector	type	Multiple	Parent selectors	Actions
price	.listing-price div	SelectorText	no	page_links	Element preview Data preview Edit Delete
room_type	div.col-12:nth-of-type(1) .col-12 div:nth-of-type(1) div.col-lg-8	SelectorText	no	page_links	Element preview Data preview Edit Delete
lease_term	div#about-this-property:nth-of-type(1)	SelectorText	no	page_links	Element preview Data preview Edit Delete
facilities	#facilities-div div.col-12	SelectorText	no	page_links	Element preview Data preview Edit Delete
shopping	div.Shopping-Malls	SelectorText	no	page_links	Element preview Data preview Edit Delete
property_type	div#property-type	SelectorText	no	page_links	Element preview Data preview Edit Delete
property_info	div.col-12.col-xl-10:nth-of-type(2)	SelectorText	no	page_links	Element preview Data preview Edit Delete
supermarket	div.Markets	SelectorText	no	page_links	Element preview Data preview Edit Delete

We run the Scraper and derive the data in the figure below.

Sitemaps	Sitemap test1	Create new sitemap
----------	---------------	--------------------

Refresh Data	Experienced any problems with web scraping? Help us understand! Sure!
--------------	---

web-scraper-order	web-scraper-start-url	page_links	page_links-href	price	room_type	lease_term	facilities	shopping
1606014711-1099	https://www.stproperty.sg/search/rent/residential?bedrooms=1&isRoomRental=true&page=6	Tan Tong Meng Tower	https://www.stproperty.sg/listings/68490911/for-rent-tan-long-meng-tower-common-bedroom-for-rent-thomson-tan-long-meng-to	\$850	Common	Key Information Room Type Common Available from Now Lease Term 1 year Furnish Fully Furnished Take Over No Date Listed 1605867970	Facilities Car Park Securities Swimming Pool Wading Pool Renovated Air Conditioning Cooker Hob/hood Water Heater Flooring For Living/dining Kitchen Flooring For Bedroom Wardrobe Balcony	Shopping Malls Balestier Hill Shopping Centre 0.25 km Zhongshan Mall 0.56 km Square 2 0.84 km

Finally, we export the data to csv file and derive the data we need.

3.3.2 Data Preparation

The data in the csv file is highly unstructured as the element on the websites have tab characters (\t) and newline characters (\n). Besides, some houses have certain fields of information while others do not have, which lead the problem that many records have missing values.

To address these two problems, we firstly replace all tab characters and newline characters with space. Then we pick up useful features from the original table and form a new one. Lastly we replace the missing value with the average of the feature if it is numerical. Otherwise we use the majority type to replace the missing values. The newly formed table is shown as follows.

	price	p_type	built_year	...	super_count	shop_avg	shop_count
0	999	HDB	1974	...	3	0.926667	3
1	1350	conda	2019	...	3	0.433333	3
2	850	HDB	2001	...	3	0.920000	3
3	700	HDB	2012	...	3	0.916667	3
4	2500	detached	0	...	3	0.393333	3
..
540	1200	HDB	2015	...	3	1.093333	3
541	2100	apartment	2016	...	3	0.646667	3
542	800	HDB	1996	...	3	0.573333	3
543	1000	apartment	2000	...	3	0.473333	3
544	950	detached	0	...	3	0.320000	3

[545 rows x 8 columns]

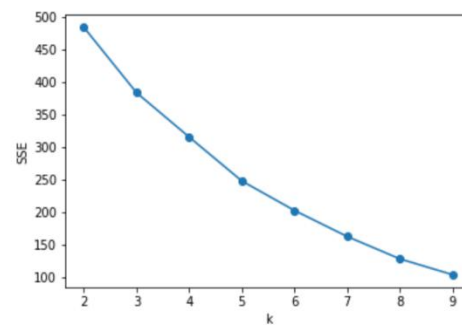
3.3.3 Clustering Modelling

Once the data is prepared, we could establish the model using Scikit-learn Kmeans method. However, several more steps are still required to obtain a better result. Firstly, we conduct one-hot encoding to the categorical features such as the property type. Then we discretize the built-year feature to categorical one using equal-frequency binning. Next we normalize the data with min-max scaling since the model will apply Euclidean Distance to the features.

Now we can group the data using Kmeans and validate the effects of clusters using Silhouette Coefficients and choose the number of clusters using elbow plot. To find a suitable number of clusters, we also use grid search to find the optimum number from 2 to 10.

The Silhouette Coefficients and elbow plot are shown as follows:

```
2 clusters: 0.39052496942144727
3 clusters: 0.41267962309358613
4 clusters: 0.447736720597301
5 clusters: 0.5303975817384255
6 clusters: 0.5883501741739658
7 clusters: 0.6414887438024189
8 clusters: 0.6885301193488044
9 clusters: 0.7207835500309656
```



According to the results, we choose the number clusters as 5. Then we print out the centroids of each cluster and establish corresponding cluster profiles.

```
[ [ 1.00000000e+00  7.77156117e-16 -3.60822483e-16  8.32667268e-17
    1.80411242e-16  6.20119669e-02  7.14285714e-01  1.14770800e-01
    1.00000000e+00  1.26415555e-01  9.43452381e-01  1.00000000e+00
    0.00000000e+00 ],
  [-5.55111512e-16  5.27355937e-16  6.94214876e-01  1.57024793e-01
    1.48760331e-01  1.46183218e-01  9.00826446e-01  1.10868379e-01
    9.97245179e-01  1.13891451e-01  9.29752066e-01  0.00000000e+00
    1.00000000e+00 ],
  [-4.44089210e-16  1.00000000e+00 -3.05311332e-16 -6.24500451e-17
    9.71445147e-17  1.21029191e-01  8.47457627e-01  1.07165471e-01
    9.83050847e-01  9.29462183e-02  9.15254237e-01  0.00000000e+00
    1.00000000e+00 ],
  [-5.55111512e-16  5.28925620e-01  2.97520661e-01  2.47933884e-02
    1.48760331e-01  1.07331622e-01  8.92561983e-01  1.17192380e-01
    1.00000000e+00  1.38696135e-01  9.04958678e-01  1.00000000e+00
    0.00000000e+00 ],
  [ 1.00000000e+00 -5.55111512e-17 -3.05311332e-16 -7.63278329e-17
    1.24900090e-16  6.94926919e-02  6.71052632e-01  9.98188074e-02
    1.00000000e+00  1.08293370e-01  9.80263158e-01  0.00000000e+00
    1.00000000e+00 ] ]
```

```
1 # cluster1  conda, high price, new house
2 # cluster2  HDB, low price, old house
3 # cluster3  HDB, low price, new house
4 # cluster4  apartment, high price, new house
5 # cluster5  terrace or detached, high price, old house
```

From the cluster profiles, we can conclude that the property type, price and built-year are able to determine the types of residence. Therefore, we will set these three features as our virtual assistant's preference queries.

4. APPENDIX

4.1 Installation and User Guide

Please make sure you are using Node.js 12.0.0 or higher (see Node.js column in [compatibility table](#)).

1. Download the source file and extract, or Clone the repository from git:

git clone

<https://github.com/nus-iss-isa-pm-group-3/ISA-IPA-2020-11-15-IS01FT-GRP3-Virtual-Renting-Assistant.git>

2. cd to the System Code directory, run the following command in terminal:

- 1) For production:

```
npm install --production
```

```
npm start
```

- 2) For development:

```
npm install --production=false
```

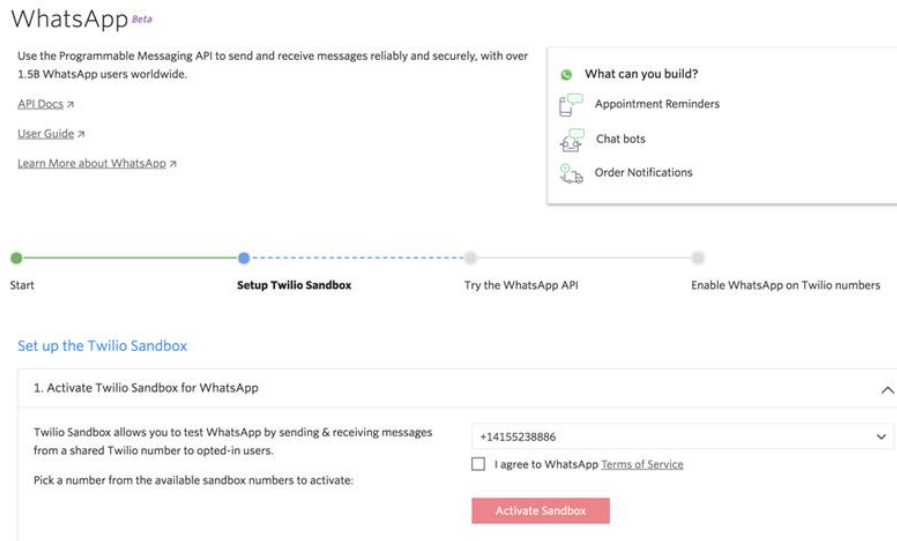
```
npm run dev
```

3. Configure Twilio

- a) [Sign up for Twilio and activate the Sandbox](#)

Before you can send a WhatsApp message from your web language, you'll need to sign up for a Twilio account or sign into your existing account and activate the Twilio Sandbox for WhatsApp. It allows you to prototype with WhatsApp immediately using a shared phone number, without waiting for a dedicated number to be approved by WhatsApp.

To get started, select a number from the available sandbox numbers to activate your sandbox.



Be sure to take note of the phone number you choose in the Sandbox. You will need this later when we're ready to send some messages.

b) [Connect the Sandbox to the server](#)

When someone replies to one of your messages, you will receive a webhook request from Twilio.

You can configure webhooks by connecting your Sandbox to an app you've already built for handling incoming messages, or build a new one for WhatsApp messages.

Twilio Sandbox for WhatsApp

Sandbox Configuration

To send and receive messages from the Sandbox to your Application, configure your endpoint URLs. [Learn more](#)

WHEN A MESSAGE COMES IN	<input type="text" value="https://e3da6f3ec9b9.n"/>	HTTP Post
STATUS CALLBACK URL	<input type="text"/>	HTTP Post

Note: The webhook URL should either be your distributed service URL (production) or the URL shown in terminal in step 2 (development).

4.2 Individual Project Report

Your name	Zhang Jiuyun
Your	

personal contribution to the project	a) Market research and System design b) Clustering analysis c) Conversation platform development d) Project Report
What you have learnt from the project	<p>From this project, I am impressed by the prospect of house renting market. On one hand, the demand in the market is stably increasing mainly due to foreign students. On the other hand, there are only few intelligent robots within this field. There are lots of steps in the renting process that can be automated such as searching a suitable residence. If these steps can be completed by intelligent agent, then not only the users can have a better user experience, but also the real estate companies could save lots of cost in human labor. I will focus on this field furthermore after this project is completed and see if I can achieve more business values.</p> <p>Secondly, the Web Scraper that I used to extract data from websites is a powerful tool for deriving data. With this tool I can retrieve data from any website quickly since it does not require any coding and can support multi-level data extraction. Once the data is derived, I could apply machine learning techniques to these data and obtain more accurate analysis on these data and websites.</p> <p>Last but not the least, the data in real life is always highly unstructured and disordered. There are lots of surprising problems in the data: some features may have lots of missing values, some features may have different units such as size, some features may be completely useless since their values are all the same or completely random. What's worse, some data on the websites is fabricated to hook visitors' eyes. Therefore, before utilize these data, it is necessary to check the quality of data. We can use relevant statistical</p>

	features of the data to judge and analyze the data.
How you can apply the knowledge and skills in other situations	<p>I will focus more on the field of house renting. Particularly, I will automate one step within the process and deploy several robots on the websites. Then I would touch with agencies and show the advantages of intelligent agents in economy and efficiency to make the business value of IPA come true.</p> <p>Secondly, Web Scraper is powerful tool which can help me obtain data from websites in my future projects. I can use this tool to extract information on popular websites such as Grab. Then I am able to build machine learning models to analyze the trend of the websites.</p>

Your name	Cao Zihao
Your personal contribution to the project	a) System design b) Conversation workflow design c) Dialogflow design and implementation d) Demo video and Project report
What you have learnt from the project	<p>My main job is to design the conversation flow and define corresponding intents in Google Dialogflow. This experience enhances my capability of designing a workflow of chatting agent. Specifically, the design of conversation needs to be clear and comprehensive to handle unexpected user inputs. To achieve this, the knowledge base of the robot needs to be large and the agent should guide the users properly.</p> <p>Secondly, I am more familiar with the Google Dialogflow thanks to the projects. In this project I apply the context technique in the platform to remember certain variables. Besides, Dialogflow can integrate with several popular applications such as Messenger easily. With the help of integration, people could build their chatbots quickly on the social websites.</p> <p>Thirdly, gathering knowledge is also an important part of this process. Without a reliable knowledge base, it is impossible to build a system based on all the rules. I have a better understanding of the general methods of building a knowledge base and I mainly use research methods in our projects. Investigation can help you find the problem more completely and quickly.</p>
How you can apply the	With the design and implementation experience about the chatting bot, I could establish an intelligent manager agent in the future to help me receive emails and analyze them correspondingly. If

knowledge and skills in other situations	<p>successful, the agent could save me lots of time in understanding and managing the flood of emails that I received every day.</p> <p>Besides, basic knowledge is also an important factor for future career. Without a solid knowledge base, users' needs cannot be better understood in the development process. In the process, I learned many ways to create repositories, which helped me to complete this project better in the future.</p>
---	---

Your name	Wang Sixiang
Your personal contribution to the project	<ul style="list-style-type: none"> • Set up web server • Twilio interaction • PDF generation • Testing
What you have learnt from the project	<p>First, I learned how to setup a web server in multiple ways. Before, I only experienced with setting up a server with Apache or Nginx. This time I have to learn how to use Node.js to create a server. Unlike other approaches I previously used, Node.js server is event-based. It may not be that intuitive at first, but after using it for some time I'm finally used to it.</p> <p>Second, I get to know how to further process the data obtained by the web scraper. Previously in class, we only learned how to scrape the data, and we did little on processing the data. In this project, I get the change to practice in filter and compose the data into a PDF file. I realized that besides using the scraped data in our own system, it's also important to manipulate and construct the data and show them to the end-user.</p>
How you can apply the knowledge and skills in other situations	<p>First of all, with the knowledge of Intelligent Process Automation, I know how to design, develop and deploy an intelligent system in real life which can help me offer more values in work.</p> <p>Second, with the experience of processing the scraped data, I'm more interested in web scraping. I'll use my knowledge to create better scraper as well as better data product.</p>

Your name	Zuo Zongyuan
Your personal contribution to the project	<ul style="list-style-type: none"> • Web scraper • Finite State Machine • Dialogflow interaction • Project report
What you have learnt from the project	<p>I've learned how to apply my pre-existing knowledge, e.g. Finite State Machine, to practical projects in order to solve real-world problems. At first, I never thought about modelling the business process into FSM. However, while I was implementing the system, I found it rather hard to better describe the business logic with simple control statements. After discussing with Zihao, who's in charge of designing conversation flow design, I finally come up with the idea of using FSM to abstract the flow. It works pretty well, and also decreases the degree of coupling.</p> <p>Besides, I also gained the experience of creating a Robotic Process Automation System, and combine with Machine Learning methods like Natural Language Processing to make the system more intelligent. In the past, chatbots are usually command-based, with Conversational Interface. While they are indeed functioning, user may complain that they are boring and less flexible. Today, with the help of NLP, we can make our chatbot more powerful by understanding what user meant without the need to input in some specific format.</p> <p>The most important thing I think is to co-work with my teammates, and I really learned a lot from them.</p>
How you can apply the	I will apply my experience in this project to build more powerful chatbots in future projects. Since now I've already got the experience of integrating NLP into the system, I'll further focus on combining

knowledge and skills in other situations	Voice Recognition and NLP, which will give the bot the ability to hear. Besides, the response of this bot is fairly simple and dull. I'll investigate in Natural Language Generation, to provide better user experience.
---	--