CarSpy - Find your Dream Car in Singapore

Wang Changqin (A0236008M)
Zhang Haolin (A0236053M)
Zhang Lei (A0235962X)
Liu Haoyang (A0235954W)

CS5224 Cloud Computing

AY2021/22 Semester 2

Department of Computer Science

National University of Singapore

April 9, 2022

1 Executive summary

Singapore is well-known for people attempting to purchase brand new car have to afford expensive extra cost (besides car price) such as additional registration fee, certificate of entitlement fee and excise duty, etc. The main motivation for government to put these restrictions on car market is to ensure that the public traffic is smooth all the time. Additionally, the amount of vehicle exhaust will increase and damage the climate if more brand new cars are purchased and used, and it is also an important concern which motivates the government to control the number of cars running in the traffic. However, there still exists a significant amount of population (such as family and businessman) who have huge interest in owning their own cars. Our team saw the incredible potential energy in the used-car market since used car usually has much lower cost and will not be counted into the increased number of cars. So, it is possible that user-car market in Singapore will be valued more and more. To fit the used-car market demand, our team decided to build an intelligent recommendation

platform for potential buyers who want to find and purchase used cars here in Singapore. The objective of our service is to recommend and suggest used-car buyers with suitable sale postings that are fairly priced.

Our CarSpy is a Software-as-a-Service application deployed on AWS cloud that provides users with customized recommendations on used-car sale postings. The recommendation system is built by using machine learning algorithm that finds matched sale posts based on users' input, their view histories and star-favourite history. The innovative part is that every recommendation made does not only consider the filter conditions that user set up in the search page, but also analyze what kinds of car posts the user viewed frequently and even starred to the favorite list. This makes our service become a more user-friendly intelligent recommendation system, instead of a simple search engine. The recommendation algorithm logic is developed and deployed by using the Function-as-a-Service Lambda on AWS, which makes our service development serverless and significantly reduces our business cost.

2 Business Case Identification

2.1 Problem Statement

It is never a wise decision to buy new vehicles in Singapore since its car prices are at the worldwide highest level. However, as an alternative, more cost-effective second-hand cars in Singapore have been gaining traction for decades. Especially after the outbreak of the COVID pandemic, people used to take public transportation are now the potential customers of used cars for affordable prices. This booming market triggered a growth in online car search and buy services [Raw22]. That said, unlike brand new cars having a relatively stable market price, used cars' prices are formulated by individuals and dealers. There are many factors leading to the price fluctuation: mileage, mechanical condition, accident history, year of manufacturing and more. As such, customers might pay for over-priced cars unwittingly in that they have no clue on what a normal price range of a specific car should be.

2.2 Purpose

In conventional cases, car dealers usually take in vehicles from various private sellers and sell them to others at higher prices than prices quoted by the seller. This amount can range from hundreds to thousands Singapore dollars or even higher for different brands and models [Ins19]. Typically, dealers will charge extra money as their profit and also for services they provided, but this charging is not regulated by the government and could be much higher than its actual value. Therefore, fair offers and fair asking prices are always crucial to make sure the mutual success of trading parties. Moreover, if dealing with private sellers, the price information could be even more significant since both customers and sellers are less likely to access the accurate market quotations. In light of this, we planned to develop an online searching tool or rather platform in order to provide listings of used cars with credible prices and information. In addition, a built-in recommendation system is applied

to the process of generating a list in order to pinpoint the most appropriate cars according to users' search conditions.

2.3 Target Users

Our web application targets at young population, taxi drivers and low-income families. People who just start a job or only worked for a few years are always short in savings. Some of them usually spend much time on commuting and most means of public transportation are crowded on weekdays. A compact car will not only save the time spent on the way but ease the tiredness as well. Nonetheless, the high price of new vehicles is often the largest concern deterring their willingness to buy. our website provides the young with an affordable option which is second-hand cars. They will no longer be frightened by exorbitant prices or worried about not knowing the used car market. Another potential group of users are taxi and Grab drivers. Comparing to a brand new car that will cost much more, a used car fits these drivers better since it is merely a tool to make a living without any performance or luxury requirement. Low-income families can also turn to our application when looking for a proper family car. Families with kids, the old or disabled members are likely in need of a vehicle for the convenient outgoing. Similar to other group of users, used cars fit these families better at a rather low price. The web tool we developed would be a perfect choice to tackle their budget problem by providing fair prices for their reference, and prevent users from falling for a price trap.

2.4 Comparison with available services

The major existing websites dealing with second-hand cars in Singapore is called "SGCarMart". First off, it is easy to tell the difference from website appearance between our web "CarSpy" and "SGCarMart". "SGCarMart" seems just stacking all information onto their page and lack of the vibrant user interface which we put much effort in. After investigating the mechanism of how to generate listings of cars on "SGCarMart", we found that the logic is rather simple. It only goes through the database after users send out the search request and return outcomes after filtering based on search conditions. Seeing this method would help little recommend suitable cars to users, we came up with a more advanced customized recommendation system with machine learning algorithm embedded. The system will create recommendations based on users' view histories and saved car posting. The users would have a feeling of our web application is reading their mind and precisely provide them with what they and most people are looking for.

3 Business Model

3.1 Revenue Model

Free Tier Our application is mainly aimed at local users in Singapore who need to buy second-hand cars. As mentioned in the previous section, the main customers of the second-

hand car market are young people or low-income families, so the initial setting of our website is completely free. Users can browse our website freely and make inquiries based on their preferred vehicle brands and models and price ranges, and we will make personalized recommendations based on users' browsing records and preferences.

Subscription As a start-up company, all development and hiring of personnel requires financial support. We have set up a subscription system for this, and set different price gradients according to the duration of a one-time subscription. Users who join the subscription system will become our VIP customers. When VIP customers browse vehicle information, we will recommend them more accurately based on their browsing and preference records. At the same time, customers can choose a low price reminder. When their favorite vehicle reaches the set target price, we will push the detailed information to customers that they can buy ideal used cars earlier.

Advertisement We collect vehicle information from various used car trading platforms. As a comprehensive used car recommendation platform, if merchants or individual sellers want to sell their cars faster, they can choose to advertise on our platform. Under the condition of ensuring the reasonable preferences of users, we will give priority to recommending these vehicles. Meanwhile, the fees charged by the platform to the merchants will also vary according to the strength of the recommendation they want.

3.2 Necessity of cloud service deployment

As a relatively rare used car recommendation platform in Singapore, it seeks benefits for consumers. With the increasing number of users, the locally deployed services will definitely not be able to meet the response needs. At the same time, the venue and hardware resources required for local deployment will be a huge cost, and using cloud services for deployment can save costs to a certain extent. Meanwhile, the functions of our application services are also constantly improving and perfecting, and local deployment of modules may lead to the rearrangement of the entire deployment, which is not we want to confront with, and cloud services can easily manage different services and the connections between services. In addition, according to the development of applications, we may go to the international market, then we hope that users in various places can access to our services with lowest latency, and the cloud services can help us solve such problems.

4 SaaS Architecture & Implementation

4.1 Architecture Overview

The overall architecture of our SaaS service is shown in Fig. 1. We use a serverless architecture, which allows us to build and run services without having to manage the underlying infrastructure, and only contains on-demand costs, making the cloud service affordable for us.

The main workflow looks simple: after the user get our CarSpy web application by HTTPS, the filtering conditions for recommendation is sent by POST request via API Gateway to Lambda function. The Lambda function select the data with filtering conditions, getting the filtered results back. The recommendation given by the recommender algorithms are sent back to Amplify, and then the searching results are shown in the front-end UI by Amplify.

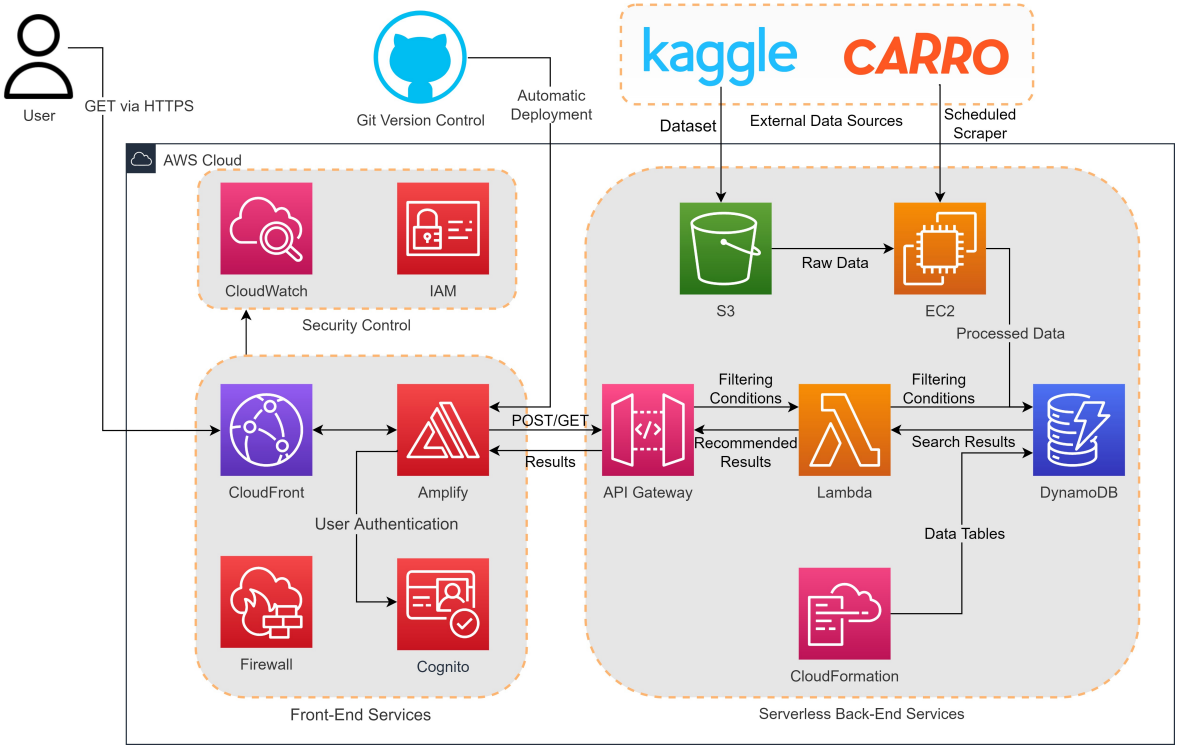


Figure 1: Overall Architecture

4.1.1 Load Balanced Front-End Service

AWS CloudFront Amazon CloudFront is a content delivery network (CDN) service built for high performance, security, and developer convenience. We deploy our Amplify app on Amazon CloudFront and S3, which can help reduce the latency and improve the security of our web service.

AWS Web Application Firewall AWS Web Application Firewall (WAF) inspects and manages web requests for CloudFront distributions based on the conditions identified in the web ACL. Meanwhile, CloudFront provides some features that enhance the AWS WAF functionality. We configure CloudFront with AWS WAF to protect our web application against external attacks that may affect the website's security.

AWS Amplify AWS Amplify is an extensible tool hosting full-stack web application. We set up our Amplify by linking it to our GitHub repository and deployed our front-end code

files to it. Amplify automatically identifies the *index.html* as the entry point of our web application and provides us an url that can redirect us to our launched main page. Every time our updated code files get synchronized to Amplify, it will automatically proceed four stages: provision, build, deploy and verify, to make sure our web application is hosted correctly and provisioned.

AWS Cognito Cognito let developers add user sign-up, sign-in, and access control to the web applications quickly and easily. Its User Pools provide a secure identity store that scales to millions of users, and can be more easily set up without provisioning any infrastructure. Our CarSpy use this service to build sign up and login page with new user email confirmation, meanwhile recording users' information in the user pool which are organized to be applied when needed. And in the sign-in process, we can get user id as the identification of current user and pass it to Amplify service, then the view and star history can be written to the database which can be used for following recommendation procession.

4.1.2 Serverless Back-End Service

AWS API Gateway We use API Gateway as the bi-directional interface connecting the front-end logic to the back-end logic. Both GET and POST types of request methods can be defined in API Gateway and linked to corresponding Lambda functions in the back-end. Our front-end code can initialize requests and passed user's inputs to a specific method in the API Gateway, then the corresponding Lambda function will be triggered and start to handle the request with implemented business logic. Lastly, the API Gateway will pass back the result generated by Lambda back to the front-end and the result will be rendered on the web page.

AWS Lambda AWS Lambda is a serverless compute service that allows us to create Python functions. It provides the basic packages like *NumPy*, with which we can implement our recommendation algorithms after selecting data from AWS DynamoDB using the Boto3 package. We use Lambda to handle the POST and GET requests sent from Amplify via API Gateway, including getting the view history of the current user and used car recommendation, and then recommended cars will be sent back to front-end services. Meanwhile, Lambda is affordable since it only charges us for the number of events we process, not the idle time.

AWS Elastic Compute Cloud Amazon Elastic Compute Cloud (EC2) provides scalable computing capacity in the AWS Cloud. We choose the t4g.large instance for our computing demands, which contains 2 vCPUs, 8 GiB Memory, and 30 GB SSD storage. Our recommendation models are trained on the EC2 on a regular basis, and the processed data are stored in DynamoDB for the recommendation system.

4.1.3 Database and File Storage

AWS DynamoDB AWS DynamoDB is a fast, flexible NoSQL database service for single-digit millisecond performance at any scale and in CarSpy development, we use it to store our users' information and both with our cars' detailed information. For the usage of recommendation, we also put the view and star history in the database and we have achieved a perfect implementation within its high scalability.

AWS S3 We use S3 to store the car information scheduled scrapped from Kaggle and CARRO website which can be fetched by EC2 service to be processed and automatically store in the final DynamoDB.

4.1.4 Security Services

AWS IAM IAM provides fine-grained access control across all of AWS, we create several roles for developers to access multiple kinds of services we have used, and each role has different access with these services which help us to solve the danger of overreaching.

AWS CloudWatch We use CloudWatch to monitor and observe operation situation of CarSpy and it provides us with data and actionable insights to monitor our application, respond to system-wide performance changes, and optimize resource utilization. Also during development, we have got many logs and events to find out what issues we had trouble with.

4.2 Data Acquisition and Cleaning

Our original Singapore used car data is obtained from the Kaggle dataset: 100,000 Singapore Used Car Dataset, which contains the scraped data of 100,000 used cars listings, which have been separated into files corresponding to each car manufacturer. This dataset has 10 variables, including model, price, transmission, and mileage, that can be used for our recommendation algorithms.

In addition, we use Python *Selenium* to schedule a daily scraper for the used car data on CARRO, one of the largest used car trading platforms in Singapore, to obtain daily data regarding Singapore's used cars. The daily data, combined with the original dataset, are stored in the standard storage S3. The raw data files are processed by a CloudFormation function to store in DynamoDB.

4.3 Recommendation Algorithms

Linear Low-rank Autoencoder Our recommender system uses an User-based collaborative filtering algorithm to recommend the cars for users. We implemented the Linear Low-rank Auto-Encoder (LLAE) [YST+21] to extract the hidden features of each user, of which the objective function is as follows:

$$\min_{\mathbf{W}} \left\|\mathbf{X} - \mathbf{W}^\top \mathbf{W}\mathbf{X}\right\|_F^2 + \beta \,\mathrm{rank}(\mathbf{W}),\ \text{s.t. } \mathbf{W}\mathbf{X} = \mathbf{S} \tag{1}$$

Given an input data matrix X, suppose that we can learn a mapping W which projects matrix X into a latent space S, and another mapping M which can reconstructs X from S. As an optimization problem, our aim is to minimize the reconstruction error as in Eq. 1.

One of the challenge problems in real-world recommender system is that we need to handle very high-dimensional and sparse matrix, because there are millions of items and users but a specific user only have few interactions with few items. Hence, we introduce the low-rank constraint term, to avoid the undesired correlations between different users.

Cold Start The right target audience for our recommender system is best calculated by looking at the former visitors for the database. According to the basic assumption of the collaborative filtering concept, if an ad was already popular with a certain group of people, then others that fit the group's profile are likely to respond well to the recommendations. However, each time a new user is registered on our website, it goes through the cold start phase due to the lack of valuable user interactions.

We use user cold start method [Mil22] for our recommender system when there is few interactions for the new registered user. Our cold start is implemented based on Content-based Filtering [VMVS00]. The content of each car item is extracted by Principal Component Analysis (PCA), and then the Euclidean distance between the new vehicles and the viewed vehicle is calculated and leveraged for recommendation. The distance of vehicle items is calculated by Eq. 2. The top-5 similar items would be recommended on the web page at that time.

$$Dist = \left(\sum_{i=1}^{n} (p_i - q_i)^2\right)^{1/2} \tag{2}$$

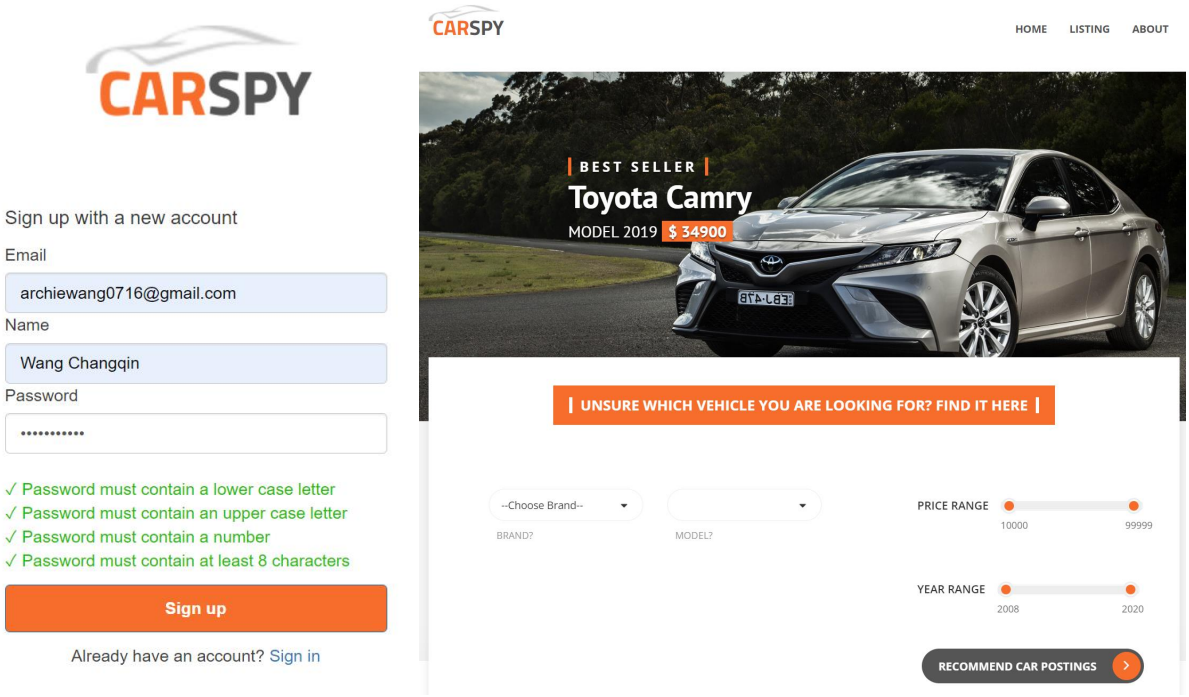
4.4 Implementation Results

4.4.1 Log In and Sign Up

A new user is required to sign up before logging in to enter our website page. A verification email will be sent to the user's mailbox.

4.4.2 Main Page

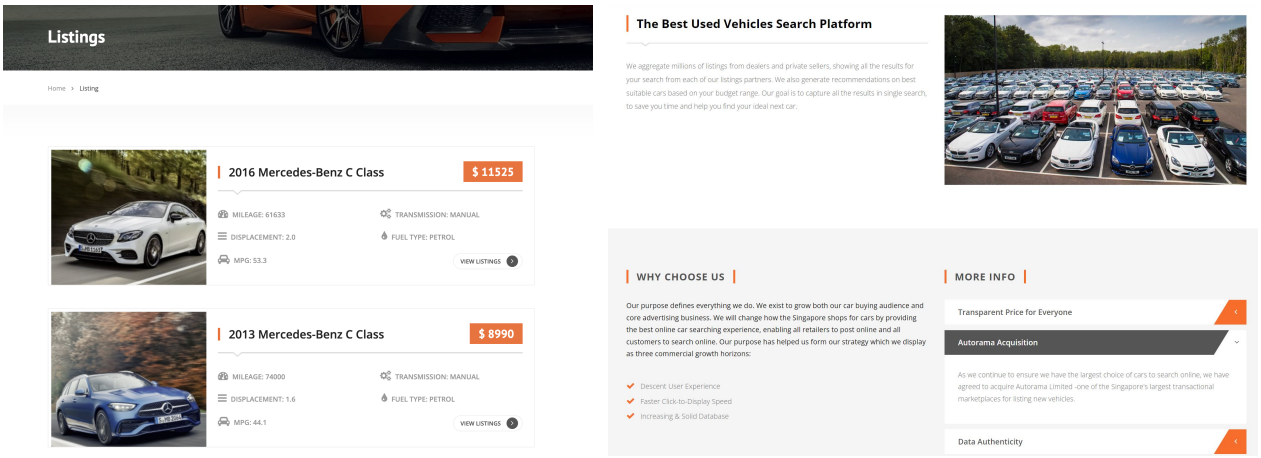
The car search function of our web application is demonstrated in Figure 2 (b). First off, select a brand from the drop-down list and the model list will be updated according to the



(a) Sign Up Page (b) Main Page

Figure 2: Baseline Model and Best Model Comparison

brand. Then moving to sliders on the right side of the page, a user can readily filter cars by the year of manufacturing or a specific price range. After clicking the ”RECOMMEND CAR POSTINGS” button, users will be directed to the listing website page.



(a) Searching Result (b) About Us Page

Figure 3: Baseline Model and Best Model Comparison

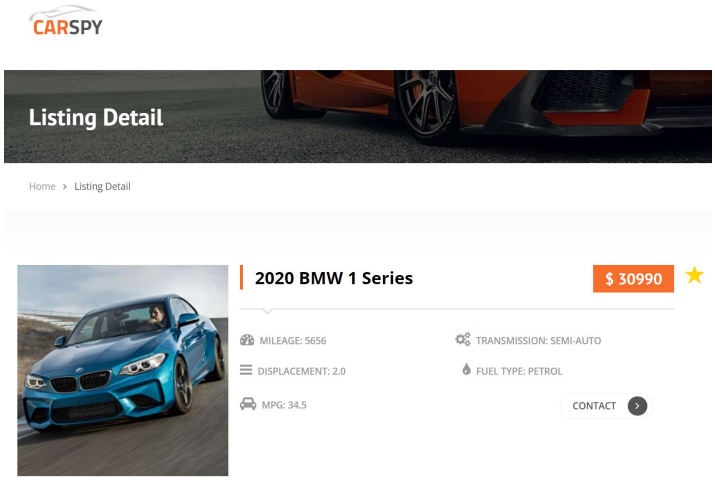


Figure 4: Car Details

4.4.3 Searching Results

Figure 3 (a) is a sample searching result of "Benz C Class". Considering the limited space in the report, only two available car postings are cropped within the screen shot. By clicking the "VIEW LISTING" button, users will be further redirected to a detailed listing page where they can contact our agents as shown in Figure 4. In addition, we have a page introducing the business we are running in detail that is given in Figure 3 (b).

5 Economic Factors

5.1 Economic benefits and key considerations

Our CarSpy application can bring a lot of economic benefits. Through personalized used car recommendations based on user preferences, it can greatly promote local used car transactions in Singapore and bring new traffic to the local used car market and individual sellers. At the same time, the increase in the number of customers buying used cars can further alleviate the country's congestion problem to a certain extent, and environmental protection problems can also be improved in some aspects, which can have positive effects on the economy. When users visit our web page, they can choose to subscribe to our premium services, whose initial price list is as Table1, and we undertake advertisement delivery from either used car merchants or individual sellers. We will charge a certain amount for this part, and there will be different price points for sellers to choose according to the strength they want to recommend.

The car sales market, especially the second-hand car market, has a strong off-peak season. During some major festivals, most businesses will seize the opportunity to prompt the vehicle and reduce the price significantly. In the face of such complicated discount information, our application can Play a great role, but it also brings huge access peaks, which can lead to

Table 1: Prices of Premium Services in CarSpy

Type	Price	Per Month
Monthly	$19.9	$19.90
Semi-Annual	$79.9	$13.32
Annual	$119.9	$9.99

increased costs for the cloud services we use.

5.2 Pricing models

At present, we use a lot of services to build CaySpy, from front-end access to back-end response, the initial settings of the services we mainly use and their prices are shown in Table 2 below.

Table 2: Estimated Monthly Cost Breakdown

Service	Configuration	Monthly Cost
Amplify	Data served per month (15 GB)	$12.37
API Gateway	Requests (500,000 per month)	$0.5
CloudFormation	Total number of operations (2000 per month)	$0.9
Cognito	Number of monthly active users (5000)	$6.75
DynamoDB	On-Demand, Data storage size (10 GB)	$6.25
EC2	Instance type (*t4g.large*), Storage amount (30 GB)	$33.73
Lambda	Number of requests (500000 per month)	$0.11
S3	Standard storage (5 GB per month)	$0.13

As an information query website, users have extremely high demand on accessing to the website with high speed and quickly response, so we initially estimated the user pool of Cognito to 5,000, and its price is $6.75 per month, and we chose a higher configuration for Amplify, which gave us more space to store our web pages, and can improve the user's access speed. At the same time, data is also the top priority of our application. In order to allow users to accurately obtain information from our application, from data acquisition to data processing (EC2, lambda) and storage (S3, DynamoDB), all better configuration being implemented. For our application is beginning to take shape, we choose an on-demand price model. Later as the number of users increases, we will continue to improve the configuration according to the situation. At that time, we will adopt the reserved price model.

5.3 Trade-off between cost and SLA

Deploying CaySpy on cloud services saves a certain cost on infrastructure. By using a series of cloud services, we save a lot of time and money resources, and under this condition, we

can have more resources to guarantee the quality of our service. In the Cognito service phase, we use the OAuth 2.0 protocol to authenticate users, which ensures user privacy and maintains the reliability of the system. In addition, our data sources are major local used car trading platforms, and the accuracy and security of the information are guaranteed. In the continuous development and implementation of the application, it caters to the growth of users and the storage of data, on the basis of ensuring the SLA possibility to increase instance redundancy to improve availability and reliability.

6 Conclusion

In conclusion, our project is a SaaS cloud service that uses a machine-learning recommendation algorithm to give suggested sale posts of used cars, based on users' input and their past post viewing history. We deployed our front-end to AWS Amplify, connecting to API Gateway which exposes back-end services that we implemented in serverless AWS Lambda. DynamoDB hosts our car posting data and also user data like viewing/starring history. Comparing to hosting on-premise servers, deploying our services on AWS cloud significantly reduces our business cost. By using serverless architecture, there will be less time consumption on developing and maintaining server logic. Instead, we can focus more and put more effort on refining back-end business logic like machine learning algorithm for recommendation. We gain profit by collecting membership subscription fee from our users, advertisement fee and partnership fee from our cooperating used car dealers. Lastly, we hope that by using our recommendation platform, more and more people can find their satisfactory used car with a fair market price. And furthermore, we want to make some contribution to the growing development of used car market here in Singapore.

References

[Ins19] Budget Direct Insurance. Why are cars so expensive in singapore? 2019.

[Mil22] M. Milankovich. The cold start problem for recommender systems. *Medium*, 2022.

[Raw22] A. Rawat. More multi-brand players plan foray into the singapore used cars market as the segment shows resilience to the economic crisis during covid-19 due to a pandemic incited surge in demand. *Ken Research*, 2022.

[VMVS00] Robin Van Meteren and Maarten Van Someren. Using content-based filtering for recommendation. In *Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop*, volume 30, pages 47–56, 2000.

[YST+21] Vikash Yadav, Rati Shukla, Aprna Tripathi, Anamika Maurya, et al. A new approach for movie recommender system using k-means clustering and pca. *Journal of Scientific and Industrial Research (JSIR)*, 80(02):159–165, 2021.