

SHOE ME THE PRICE

AN IPA AGENT FOR GETTING THE BEST ONLINE SHOE PRICES



Team : **The Kickers**

Karamjot Singh A0198470U

Kartik Chopra A0198483L

Tarun Rajkumar A0198522X

National University of Singapore
Institute of Systems Science

INDEX

INTRODUCTION	2
SYSTEM DESIGN AND FUNCTIONALITY OVERVIEW	4
RPA/IPA OVERVIEW	6
TECHNOLOGICAL SPECIFICATIONS	7
RESULTS	8
CONCLUSION	9
FUTURE WORK	9
REFERENCES	10
APPENDIX 1 : USER GUIDE	10
APPENDIX 2 : INDIVIDUAL PROJECT REPORT	14

INTRODUCTION

Over the last few decades shopping has transitioned from the traditional **brick and mortar stores** around the block to internationally shipped **online stores** that don't have a single physical store. In a recent survey:



In a recent poll, it was found that



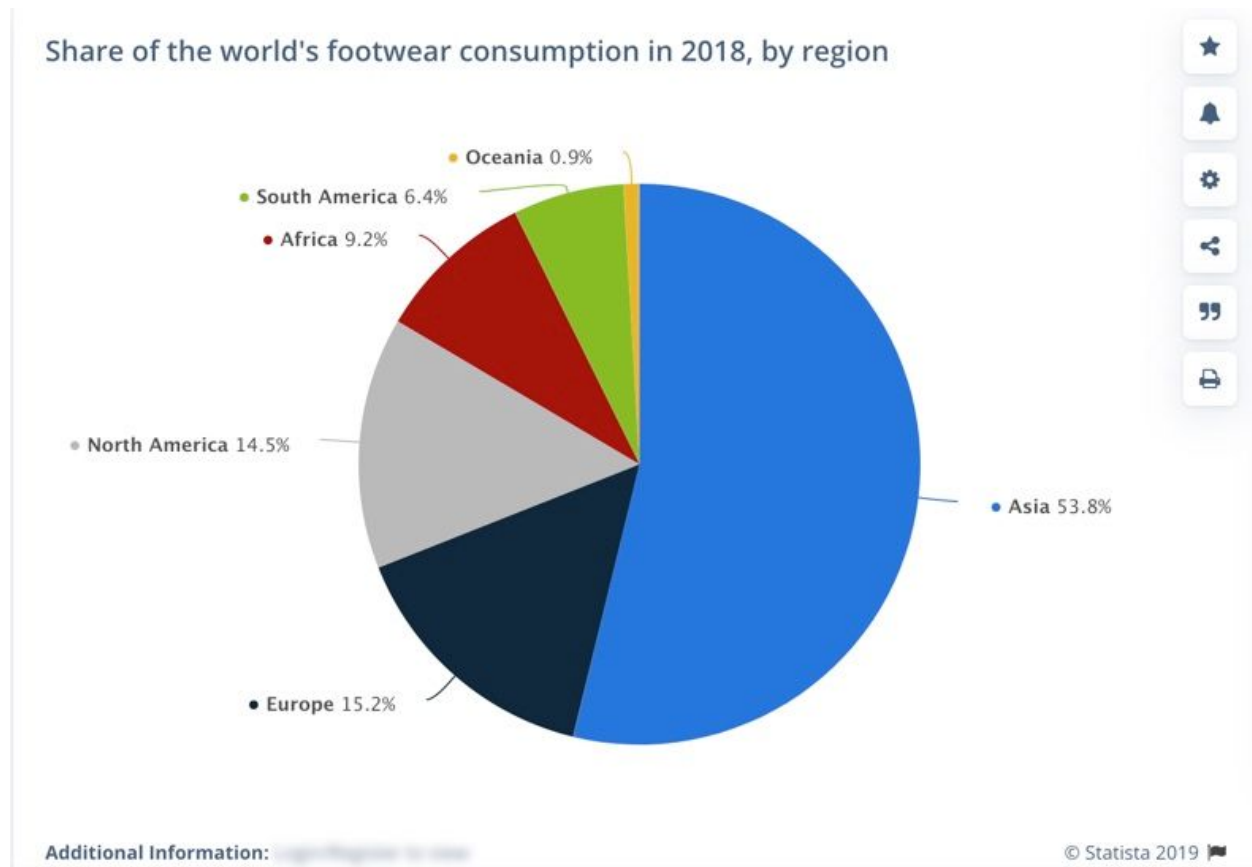
[Source](https://returnnonnow.com/2019/03/how-shopping-has-changed-in-recent-years-infographic/) : <https://returnnonnow.com/2019/03/how-shopping-has-changed-in-recent-years-infographic/>

‘**Shopping** is an activity in which a customer browses the available goods or services presented by one or more retailers with the potential intent to purchase a suitable selection of them. ‘ - wikipedia

With the above as the basis for our product we have concentrated on the vital process of

comparing the product from the above definition 'one or more retailers'. Specifically we've decided to place our product for a niche market of the population that thinks twice before spending money on a product that can be considered an essential and a luxury.

Asia is the largest consumer of footwear in the world:



Source : <https://www.statista.com/statistics/227281/share-of-the-worlds-footwear-consumption-by-continent/>

And given that asian market is considered to be one of the most conservative markets, in terms of spending, we'll plan to cater this market for now.

In respect to spending, we've built upon this idea and came up with a solution to compare shoe prices online and receive price updates of user selected shoes over time based on per user subscription. In addition to comparing price we also predict if the price of a user selected shoe will either slump in the times to come or will remain the same.

All this by just letting us know the name of the shoe the user is interested in and the users email address.

SYSTEM DESIGN AND FUNCTIONALITY OVERVIEW

1. The user can send a request via email or from the website.
 - a. Request from the website channel is a simple form that user can fill up in seconds. It requires the user to provide -
 - i. Email address
 - ii. Shoe-names
 - iii. Gender
 - iv. Shoe-size
 - v. Option to opt-in for a daily subscription
 - b. Request from the email channel, is simply sending a mail with subject as shoe-names prefixed with gender, to - shoemetheprice@gmail.com.
 - i. E.g. 1 **[M] Nike Air Max 1**
 - ii. E.g. 2 **[F] Nike Joyride, Nike Vapourmax**
2. The request for the particular shoes gets processed at the backend. Each request is inserted to our subscriber database.
 - a. Requests from email channel are periodically checked and only the new requests in the unread mailbox are inserted to the database.
 - b. Requests from website are inserted upon arrival.
3. The bots scrape the details of the shoes from various websites in real-time to ensure that the latest prices are always pushed to the user.
4. Along with the prices, a recommendation about the trend of the price is also calculated.
5. Once all the bots are finished with their processes, a dynamic response is developed for each user with the shoes they requested, the price of the shoes from various websites along with a recommendation stating whether the prices are likely to go up or down or remain the same.
6. In case the bots are unable to fetch a particular shoe from all configured shopping websites, a predefined mail, detailing the same is sent to inform the same to the user.
7. The developed response is shared with the user via the email id they provided and user can click on each shoe-name to visit the respective website for further details.

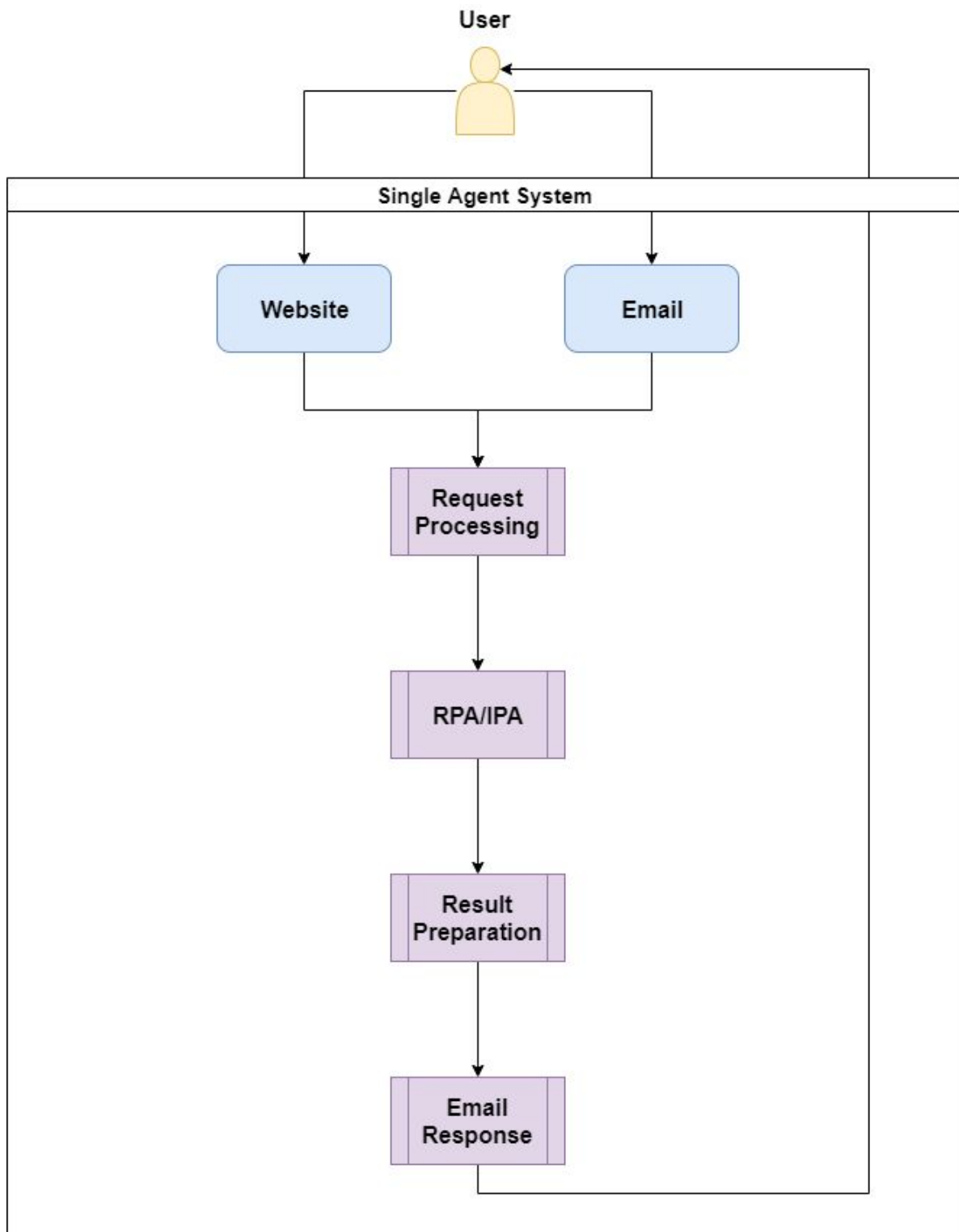


Figure A: ShoeMeThePrice : System Design

RPA/IPA OVERVIEW

1. Software robots have been developed that each scrape a single website. The bots have been created for popular online websites for shoes -
 - a. Nike
 - b. Footlocker
 - c. JD Sports
 - d. FarFetch
2. Each bot visits the respective website, finds the search box and types in the requested shoe.
3. Along with this, each bot is able to handle whether the requested shoe is for men or women.
4. The bots after finding the appropriate shoes returns top 3 results from each website. The top 3 results are considered to provide flexibility to the user as a single query may yield multiple shoes of the same name.
5. After fetching the results from the bots, a lookup is performed on the price history database table for the current shoe.
6. This is done to perform time series analysis for each shoe so that a recommendation about the price trend can be shared with the user.
7. The time series trend is estimated using data of the last 15 days of the prices. Delta (change in daily price) is calculated and averaged over 15 days. If the delta is greater than \$2 (considering minimum price change has to be by \$2), then the trend of the shoe price is likely to go up and if the trend is less than negative \$2 then the shoe price is likely to go down.
8. In case, we don't have data for a particular shoe, a generic response is sent in response & the details of that shoe is added to the database - so that future responses can be improved for the same shoe.
9. The application is robust. The bots are designed in such a manner that changes on the popular shoe websites can easily be handled. All the bots are loosely coupled with the code.

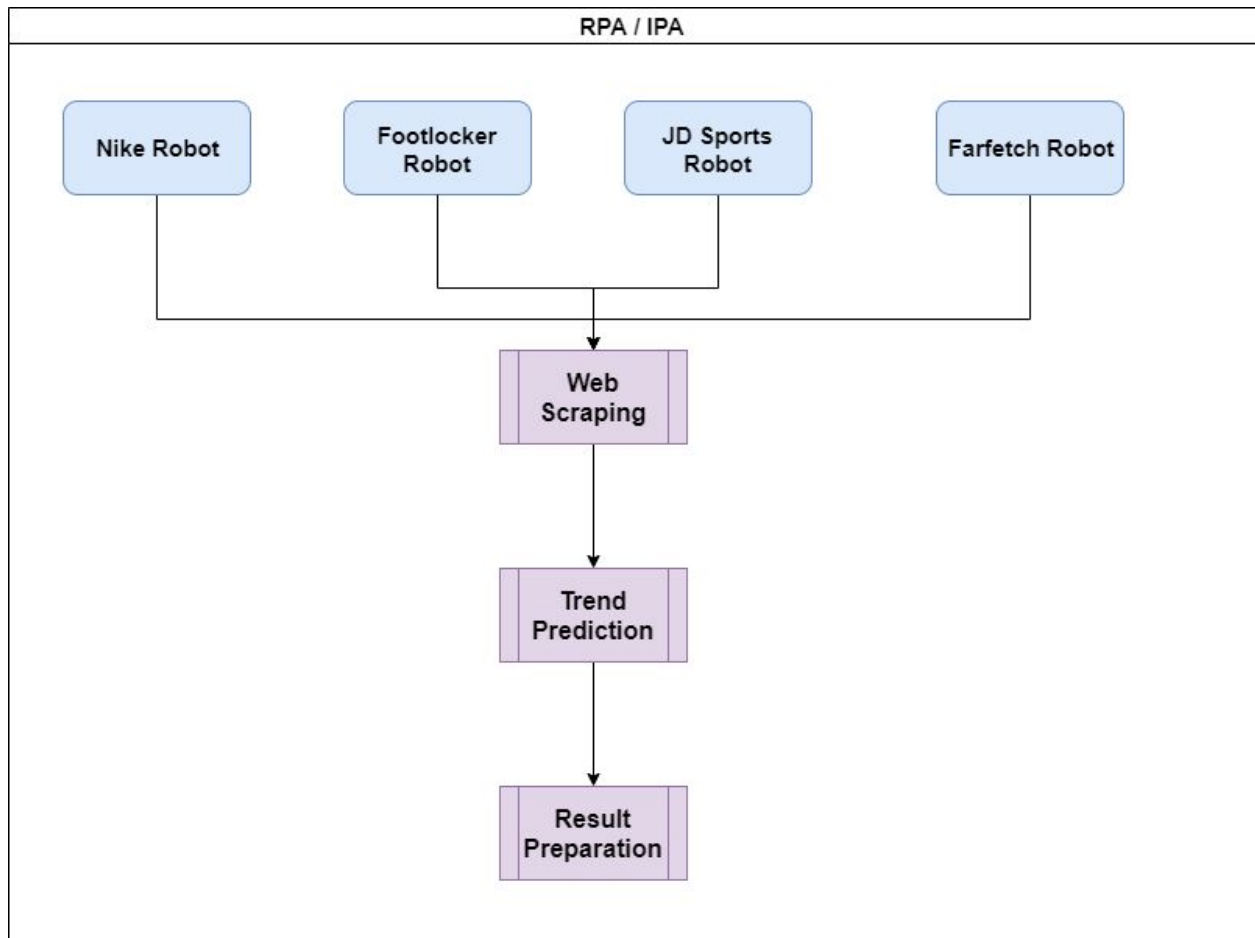
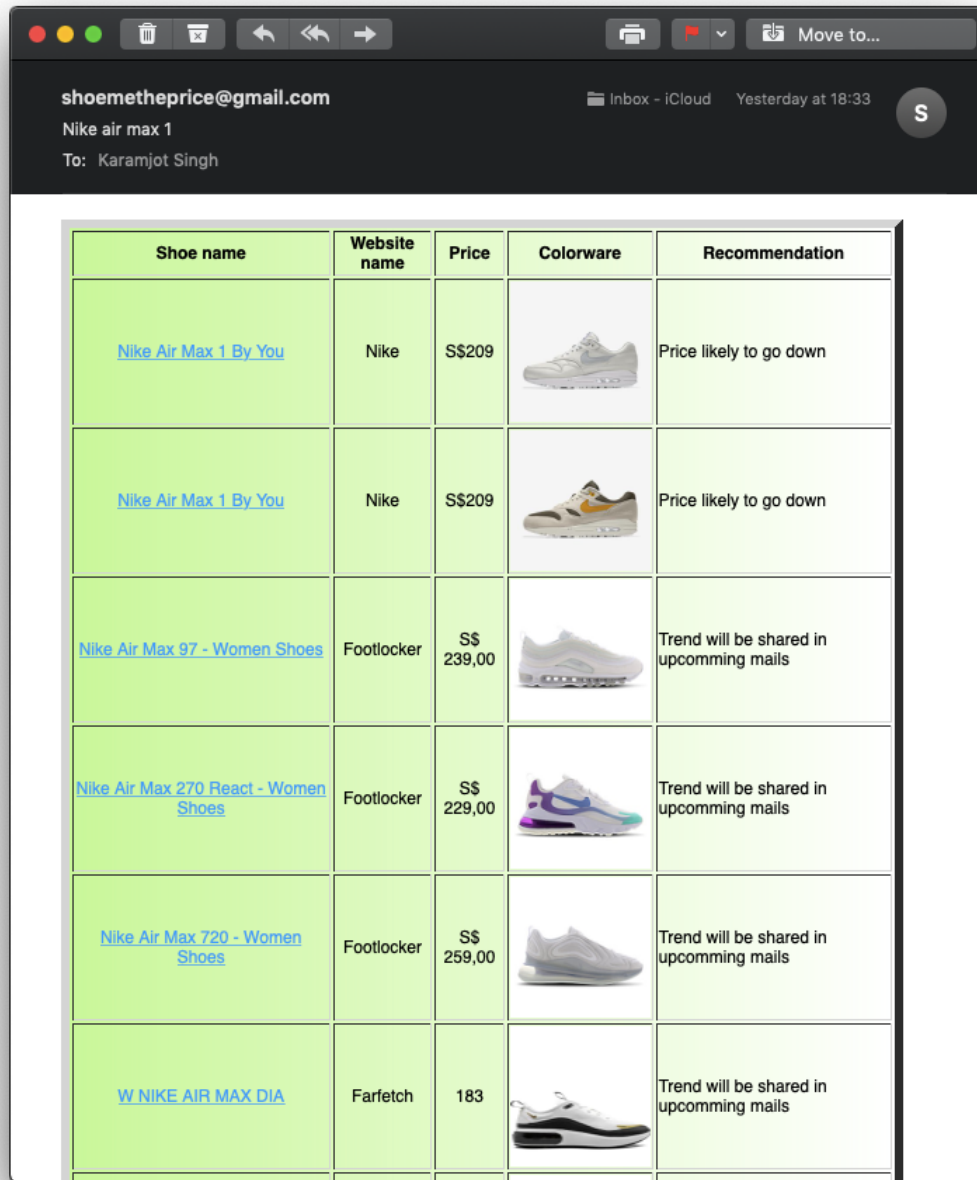


Figure B: *RPA/IPA Architecture*

TECHNOLOGICAL SPECIFICATIONS

1. **Python** is used as the programming language to develop the entire system. All individual components of codes ranging from RPA to webpage to database are integrated using python.
2. **TagUI**, which is AI Singapore's open source Robotic Process Automation (RPA) tool is used to develop the bots that perform web-scraping from several online shoe stores.
3. **Python Flask** is used to maintain server for the web application.
4. **HTML/CSS/JS** is used to develop the frontend for the web application.
5. **Postgresql** is used as the database system for storing customer information, requests as well as price history of shoes for the calculation of trends.
6. **Gmail API** is used for sending replies and receiving email from the users.

RESULTS



The screenshot shows an email client window with the address bar displaying 'shoemetheprice@gmail.com'. The email subject is 'Nike air max 1' and the recipient is 'Karamjot Singh'. The email body contains a table with five columns: 'Shoe name', 'Website name', 'Price', 'Colorware', and 'Recommendation'. The table lists six different Nike Air Max shoe models with their respective prices and recommendations.







Shoe name	Website name	Price	Colorware	Recommendation
Nike Air Max 1 By You	Nike	S\$209		Price likely to go down
Nike Air Max 1 By You	Nike	S\$209		Price likely to go down
Nike Air Max 97 - Women Shoes	Footlocker	S\$ 239,00		Trend will be shared in upcoming mails
Nike Air Max 270 React - Women Shoes	Footlocker	S\$ 229,00		Trend will be shared in upcoming mails
Nike Air Max 720 - Women Shoes	Footlocker	S\$ 259,00		Trend will be shared in upcoming mails
W NIKE AIR MAX DIA	Farfetch	183		Trend will be shared in upcoming mails

Figure C: An example response from the agent

As seen in the sample response above, the user requested a price query for “Nike air max

1” shoes. The agent replied with a dynamic response mail, after scraping the details for the same from various websites, providing the URL (on shoe-names), website name, price, image and a buying recommendation - for that particular shoe. The shoes for which prediction data for the past 15 days is not present, a generic response is sent as the recommendation. For the available ones, an upward or downward trend prediction is returned. Based on this reply, the user has the flexibility to choose a particular website to purchase the required shoe after an instant comparison.

CONCLUSION

We’ve successfully created a service to bridge a gap in the market for active price comparison of shoes of interests to users. The entire process for the application does not require any human intervention. This was possible entirely due to the use of intelligent robots developed with the knowledge and experience gained in our recent courseware on ‘Intelligent Process Automation’.

Through the course of this project we’ve come to understand that the scope of intelligent bots can be extended to various other tasks in various processes which can reduce human intervention thereby reducing valuable resource cost and time .

FUTURE ENHANCEMENTS

Immediate Additions - These are some functionalities that can enhance the market value of the product instantly

1. Support for more websites as robots, like stockx.com, adidas.com.
2. Support for more channels like Chatbots, Google Assistant voice search etc.
3. Scraping websites (for all shoes on popular sites) routinely to maintain data for the prediction model, irrespective of user searches (right now, data is being added on user searches).
4. Coupon/offer extraction from sites.
5. Multi-agent design for handling requests from various channels parallely, rather

than the current sequential approach. This would allow almost instant replies and will mitigate the response time - making it independent of the number of pending requests in the system

Prospective Additions -

1. Support for additional merchandise search - like bags, clothes etc.
2. Improved prediction model
3. Image-based search - if a user is unable to name a particular shoe, but has image of it, the user can send this image to our SMTP agent & the agent can look-up for similar shoes on various shopping websites, and reply the user with appropriate response.

REFERENCES

1. TagUI RPA Library for Python - <https://github.com/kelaberetiv/TagUI>
2. GMail API - <https://developers.google.com/gmail/api>
3. Cover Photo - <https://thesolesupplier.co.uk/release-dates/nike/off-white-nike/off-white-x-nike-air-vapormax-white/>

APPENDIX 1 : USER GUIDE

To see this system in action, please follow the below steps. Once all one-time activities are handled correctly, the whole application can be run by just a simple command on command line interface.

System Requirements -

A PC running macOS/Windows or Linux operating system with :

- A. Python3 (this system will be used as a server for execution of bots).
- B. A python env on above machine based on file: requirements.txt
- C. PostgreSQL - <https://www.postgresql.org/download/>
- D. PostgreSQL server application -

1. Recommended for macOS : **PostgresApp**(Free) - <https://postgresapp.com>
2. Recommended for Windows/Linux :
 - a) **DataGrip**(Free) - <https://www.jetbrains.com/datagrip/>
 - b) **pgAdmin**(Free) - <https://www.pgadmin.org/download/>
- E. Google Chrome - <https://www.google.com/chrome/>

Environment Setup (one-time process) -

1. Download and unzip the project to a path on disk **without any spaces**, like '~/.development/ISA-IPA-2019-10-07-IS01FT-GRP-Kickers-Shoe_Me_The_Price'.
2. Setup the PostgreSQL server and note the **username** and **password**. If using remote server, please note **host**, **port** as well.
3. Add this information to file: **configurations.ini** located in the home folder, under the **[DATABASE]** tag.
4. Create a new python virtual env OR using an existing, install dependencies from the requirements.txt file - **pip3 install -r requirements.txt**

Application execution -

1. **Activate** python env (if using a virtual env for this application)
2. **Activate** postgres server from the server application
3. **Open** terminal/command-line interface
4. **Goto** home folder of the app - ISA-IPA-2019-10-07-IS01FT-GRP-Kickers-Shoe_Me_The_Price/
5. **Execute** - "python3 initiate.py"

Example flows -

Once the application is running, choose any of the two channels for shoe price utility -

a. Web-interface -

- i. Go to <http://127.0.0.1:5000/>. This acts as a front-end where the user can request a query to the application. A form will be presented (figure A).

The screenshot shows a web browser window with the address bar displaying '127.0.0.1'. The page has a blue header with the text 'Show Me The Price'. Below the header, the main content area is white and contains a form titled 'Key in some details to subscribe'. The form includes an email input field with the value 'karamjotsingh@me.com', a label 'Email', and a green underline. Below this is a section for shoe names, with two chips: 'Nike Air Max 1' and 'Nike Vapourmax', each with an 'X' icon to remove it. A label 'Name your shoe' is positioned below the chips. Further down is a 'Select Shoe Size' section with a dropdown menu showing the value '9'. To the right of the size dropdown are gender selection options: 'Male' and 'Female', with a radio button selected for 'Male'. To the right of the gender options is a 'Weekly Updates' section with a toggle switch set to 'No'. At the bottom left of the form is a 'Submit' button.

Figure D : Web interface of the application

- ii. Complete and submit the form. One can enter more than one shoe names. Press **Enter** after key-ing in every shoe-name- to make shoe-name chips. On completion, confirmation screen (Figure B) will be shown.

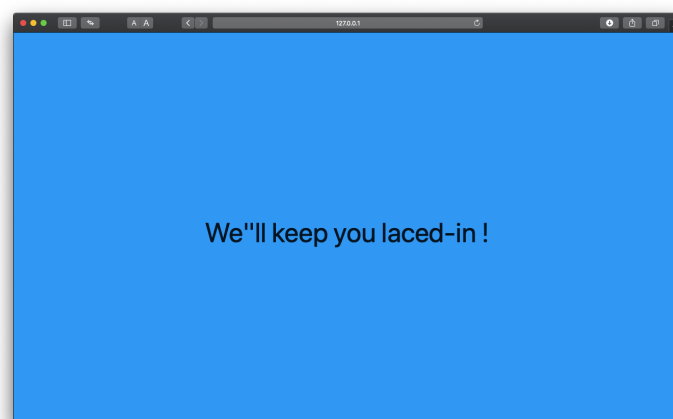


Figure E: Successful submission of request

- iii. Once the request is successful and correct, the bots will execute on the hosting machine and will mail the request user with appropriate details in response.

b. Mail service -

- i. Compose & send a mail with the following details -
 - 1. To - shoemetheprice@gmail.com
 - 2. Subject - **[Gender] Shoe_Name_1, Shoe_Name2...**
 - a. E.g. 1 [M] Nike Air Max 1
 - b. E.g. 2 [F] Nike Joyride, Nike Vapourmax
 - c. E.g. 3 [M] Nike Free

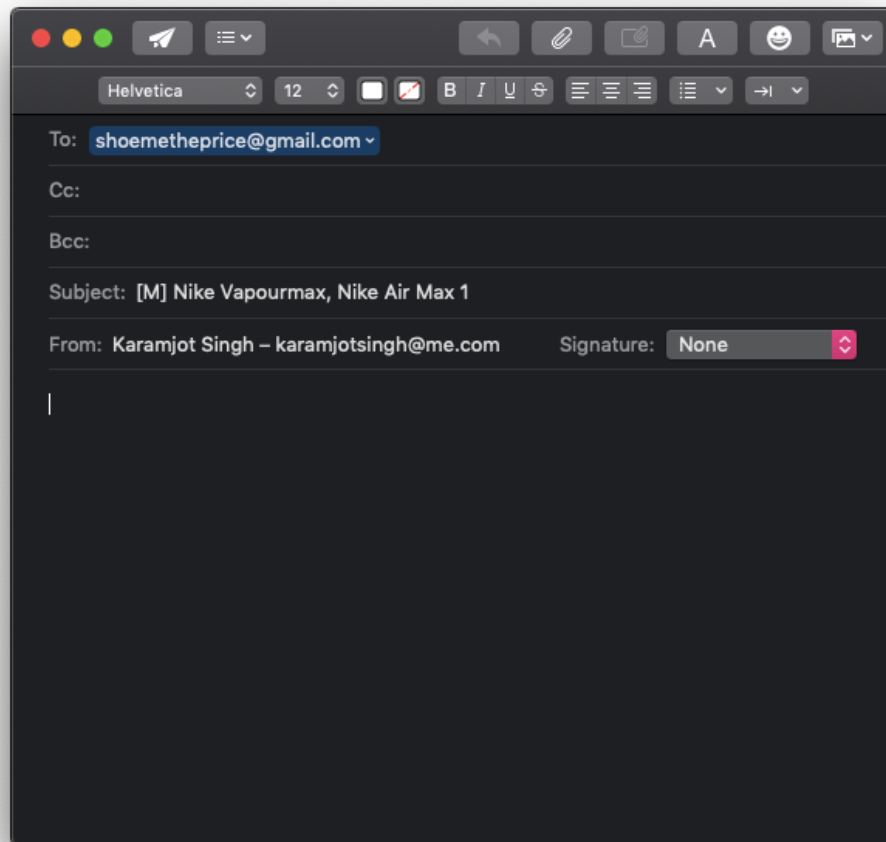


Figure F: Sample mail request

- ii. Once the request is successful and correct, the bots will execute on the hosting machine and will mail the request user with appropriate details.

Notes -

1. It is advisable to let the server machine execute the flows unaltered, meaning not to perform actions on the system which is being used as the server for this application. Also, advisable to not alter the chrome browser execution window.
2. Running the application from home folder is mandatory. Firstly, please reach the project's home folder & only then run the application.

APPENDIX 2 : INDIVIDUAL PROJECT REPORT

Karamjot Singh - A0198470U

1. Personal Contribution -

- Project design & structure – Initially provided this idea to the team, which on careful analysis of complexity and business value, was finalized for the group project. Upon selection, designed the general flow strategies of the app, various packages and modules of the app, various flows of the applications, their interactions (& segregation for code maintainability). Consolidated these with the team in discussions.
- Market Analysis – Initial examination of the trends in the market for shoe 'craze'. Looked into various factors like the current players in the market, their website/channel's user experience, trends on google over the years - which allowed us to narrow down the popular sites, channels and functionalities for the application that would be relevant to the potential users.
- Creation of RPA robots – Created RPA robots for Nike, Under Armour, Puma & Adidas. For simplicity of the project, only a few robots out of the total, were kept in the final app.
- Designed & developed code for the functionalities - Automated project env and database creation scripts, DB updates methods, maintaining the data for the application, parallel processing of scripts of different channels - one script handles all channels, handling to mailing bot - receiving and filtering relevant emails

through gmail API, timely calling and handling of robots for instant responses, design for the prediction model with regards to our application, sending response to the user for all channels.

2. What have I learnt -

- Application of RPA package such as TagUI to harness data from websites and automate mundane processes.
- Creation of prediction logic for time series problems.
- A refresh on designing & developing python MVC applications, and integrating RPA & IPA techniques to build intelligent systems.
- Python design patterns and exception handling.

3. Application of knowledge and skills in other situations -

- Planning to expand this application to include even more RPA/IPA features - as the ones mentioned in the future enhancements of this project.
- Inclusion of cloud API models to the same problem, such as object classification and labelling through Google cloud API for image-based search enhancement.

Kartik Chopra - A0198483L

4. Personal Contribution -

- I developed 2 RPA bots for the projects, Nike and Footlocker.
- I developed the time series prediction logic for price trends.
- Along with this, I developed several modules such as bot_controller that handles all the bots, price_prediction that predicts the trend of each shoe from each website as well as daily_script module that ensures that the shoe requests are processed on a daily basis as well as handling of any new shoe request that comes in. I also developed several functions in db_updates that help with the manipulation of data in the database.

5. What have I learnt -

- While working on this project, I gained in depth knowledge of TagUI Python.

- I learnt about time series analysis to a level to apply it to simple problems such as this one.
- I also learnt how an end-to-end system is developed.
- I also gained a lot of insight about what an RPA is what an IPA is and how it can be used to solve repetitive problems with some intelligence.

6. Application of knowledge and skills in other situations-

- TagUI python is very flexible and can be applied to any domain problem. We developed an RPA to solve the problem of searching for shoes but this can be extended to any kind of commodity that one searches online (clothes, bags, consoles, laptops, electronics).
- An RPA bot can be developed to replace any repetitive task and I believe that I would develop an RPA/IPA bot that can help me in my personal life such as handling of several accounts, or updating my budget list to an excel sheet.

Tarun Rajkumar- A0198522X

1. Personal Contribution -

- Developed RPA bots for JDsports.com.sg and Farfetch.com.sg
- Built the UI for the website.
- In addition I also created the dataset required for the initial price prediction by scraping data and then engineering the data to extend the number of samples.
- I also explored the possibility of scaling the project by using the concept of ORM for the project using sqlalchemy.

2. What have I learnt -

- While gaining familiarity with TagUI, I also understood that there are a lot of edge cases including and not limited to considering parameters such as browser window size, usage of additional meta tags, IFrames et cetera.
- Data engineering, as to what goes on behind the scenes of generating datasets.
- Extension of RPA in processes not limited to just web scraping.

3. Application of knowledge and skills in other situations-

- Over the process of understanding TagUI, I'd likely explore the feasibility of pairing this to more intelligent automation in the future.
- I've been actively working on a side project that helps me fill up job forms based on input labels for me.