Wafferly

وفّر لي

Prepared by

Amjad Al-Nashwan,

Bashair Al-Jarba,

Shahad Al-Barrak,

Njood Al-Shaalan

Supervised by

Dr. Afnan Al-Subaiheen

Second Semester 1441 Spring 2020

Project Submitted in Partial Fulfillment for the B.Sc. Degree in Information Technology.

STUDENT DECLARATION

Students acknowledge that they have read the above policy pertaining to plagiarism and understands that the penalty for such an act could result in disciplinary action.

Project Title:

WAFFERLY PRICE COMPARISON APPLICATION

Term of the Project: 1st & 2st Semester

Academic Year:

1441/1442 (Higri) - 2019/2020 (Gregorian)

Role	Responsibilities	Participant
Project Leader	Manage the teamwork	Amjad Al-Nashwan
System Analyst	Developing the required system model	Njood Al-Shaalan
System Analyst	Developing the required system model	Bashair Al-Jarba
System Designer	Design the interfaces Design logo	Shahad Al-Barrak
System Analyst	Develop solutions Analyze systems requirements	All team members
Project Reviewers	-Review the document. -Report comments about the document	All team members

I. ACKNOWLEDGMENTS

In the name of God, the Most Gracious, the Most Merciful. First, we thank ALLAH for his ultimate help and for the ease he always bestowed upon. Then we would like to express our deep and sincere gratitude to our supervisor, Dr. Afnan Al-Subaihin, Dr.Nora Al-Twairesh, for dedicating the time and effort to supervise our work and to meet with us, also for being there when we need them and encouraging us to be the best at what we do. Furthermore, we are deeply thankful to Dr. Hend Alrasheed for her support and all the members of the graduation project clinic. finely we are extremely grateful to our parents and friends for their love, prayers, caring and sacrifices for educating and preparing us for our future.

II. ABSTRACT (ENGLISH)

Online shopping plays an important role in the economy because of its high financial revenues, online shopping continues to grow with the spread of the internet around the world. Online shopping facilitates purchases for consumers by showing them relevant product information and giving them the convenience to buy products from around the world.

With the increase of applications that provide online shopping services, prices vary from one application to another. Consequently, many shop comparison applications are developed. This project’s goal is to provide a shop comparison application for Middle Eastern consumers with many features that facilitate for the consumer what suitable product and store to choose. Furthermore, this application provides special features, including finding synonyms when using different Arabic dialect and searching by image.

III. ABSTRACT (Arabic)

التسوق الإلكتروني يلعب دوراً مهماً في عالم الاقتصاد؛ ممّا ينعكس عليه من عوائد مادية مرتفعة، وقد استمر توسّع التسوق الإلكتروني مع انتشار الإنترنت في العالم. يسهّل التّسوق الإلكتروني للمستهلكين عمليات الشّراء عن طريق عرض المنتجات بكل تفاصيلها، وأيضا تتيح له فرصة شراء منتجات من حول العالم.

مع تزايد المواقع والتطبيقات التي تتيح للمستهلك التسوق إلكترونياً، ظهرت مشكلة تفاوت الأسعار بين هذه المواقع والتطبيقات مما أدى إلى ظهور العديد من المواقع والتطبيقات منها عالمية ومحلية للمقارنة بين أسعار هذه السلع على مختلف المتاجر المتاحة. يهدف هذا المشروع إلى توفير تطبيق مقارنة أسعار بين الأسواق والمتاجر العالمية والمحلية مع توفير العديد من المزايا التي تسهّل للمستهلك عملية اختيار المنتج والسوق/المتجر المناسب، كما يوفر أيضا خدمة البحث عن المنتجات باستخدام صورة المنتج.

Table of Contents

List of Tables

Table of Figures

CHAPTER 1 | INTRODUCTION

1. INTRODUCTION

The Saudi Ministry of Commerce is planning to start a program that pays attention to improve e-commerce by organizing all people who are affecting the e-commerce sector, according to the vision of the Kingdom of Saudi Arabia 2030 [1]. Since online shopping has been trending around the world, most of the international brands set up their online stores to satisfy their customers' needs. The result of the increased number of online stores led the companies of physical stores to bankruptcy because consumers shifted to online stores. As the New York Times stated in September 2019, that the brand Forever 21 was compelled to close around 350 stores around the world [2]. The trend of online shopping has also been increasing in the Middle East in recent years. There are statistics affirming the growth of online shopping. The Saudi Communications and Information Technology Commission declared that the percentage of online shopping increased from 37% in 2016 to 47% in 2017 and in 2018 the percentage continued to grow reaching 49.9% [3].

This shows that there is no doubt about the importance of online shopping these days. There are many reasons that contributed to the growth of online shopping: consumers save time, effort and can overcome physical limitations (such as shopping from shops located in different regions).

1.1.The problem

From the customers' point of view, they are faced with two main challenges when they intend to purchase products online. First, due to the increasing number of online stores, it became hard for them to have a full picture of all online stores that offer the products they are interested in. Therefore, customers may find it challenging to scan online stores to get informed about other stores that provide the same products which might vary in quality, price and delivery date among stores. Consumer decision-making may be guided by price, quality (i.e. rating).

Add to that the fact that online stores may change their pricing and offers regularly, making it hard for customers to know which one currently has the best offers. Consumers need better technology solutions that broaden the scope of their search in a timely manner to have a better view of the market to develop well-informed purchasing habits.

The second challenge is introduced by the fragmentation of naming the products in the Middle Eastern stores. Online stores might list the product name in English (for example

‘TV stand’), transliterated Arabic ,(‘ستاند تلفزيون’) or dialect Arabic (‘طاولة شاشة’,’دولاب’ or ’حامل تلفاز’). This makes it harder for customers to search for certain products if they are listed under different names than the one, they thought of.

Moreover, the emergence of large well-known websites negatively affected small startups. Such large online shops offer huge marketing power, making them the first choice to the customers even though startups websites might still have better offers. This poses as a drawback wherein few online shops might monopolize the market and inhibit the growth of small businesses.

Wafferly aims to help consumers by making the search process faster and more scalable. On the other hand, the market will benefit from Wafferly by giving an equal chance for the small business to appear in the search result as much as the big business.

1.2.Project Goals and Objectives

The goal of the project is to provide a unified e-commerce search application, from which consumers can search either using textual keywords or pictures; then browse the results which shall be filtered using preferred criteria.

The objectives are to:

1. Enable consumers to search in online shops by giving them access to many online shops.
2. Enable the consumer to search in current market listings using images of the required products or using descriptive keywords.
3. Help small and medium businesses get exposure and have their products and offers viewed by consumers even if consumers didn’t know these websites existed.
4. Eliminate the conflict that arises from using more than one dialect by using an Arabic linguistic model that could find synonyms of the attributes taken from image or descriptive keywords.
5. Help consumers make better purchasing decisions by comparing offers using either price or rating of shops that sell the required product.

1.3.The Solution

The solution shall be a one-stop Android and iOS application that enables the consumers to search in online shops for the desired product. The search can be done by supplying the product's name or an image of the product, and that can be done by either taking a picture or by importing it from the device's album.
Given an image, the application shall use Google Vision API [4] to analyze the image and gather its attributes, then it builds and uses a synonym lexicon that contain number of colloquial Arabic synonyms for the product.
Given the search keywords, the application shall do web scraping to search in online shops for the required product. The application shall display all relevant results (i.e. products) employing an easy-to-use, easy-to-understand interface. The consumers can then sort the online product result either by price or rating.

1.4.Project Scope

Wafferly uses the object-oriented model and it is available as an Android and iOS native application. The main target users of the application are consumers in the Middle East. The application interface language is going to be Arabic. It will allow consumers to search for a product using several ways (textual or image lookup). It will allow the consumers to search for a product in Arabic or English. The application also requires an Internet connection, and it will not include offers feature, barcode scanning, or search by voice. We will discuss the reason why it is not included in the Literature Review.

Chapter 2 | BACKGROUND

2. BACKGROUND

This project shall be delivered as a native mobile application which will be available on both Android and iOS devices. Android is an open source mobile operating system (OS) that is used by many mobile hardware manufacturers such as Samsung, Huawei, HTC and many others. On the other hand, iOS is a mobile operating system that is restricted to Apple devices. Figure 1 shows the market share of mobile devices' operating systems that are used in the Middle East[1]. It covers a year from September 2018 to September 2019. According to the statistics, Android OS and iOS are the most used in Middle East.

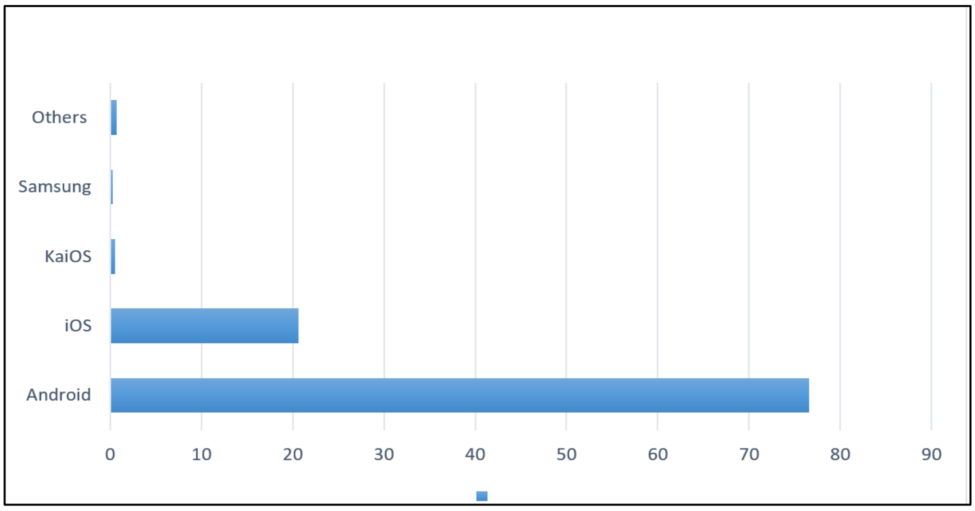


Figure 1 Statistic of mobiles and tablets operating systems market share

Android native applications are implemented in Java or Kotlin. Android application development is done using the official Android Studio. While iOS native applications are implemented in Objective-C or Swift. The development of iOS applications is done using XCode IDE.

Wafferly is going to be developed for both iOS and Android platforms as mentioned before. Having two incompatible codebases to do the same functionality, due to the use of different programing languages is an overhead. The application is going to be implemented using React Native [5] which will provide one codebase for both platforms (iOS and Android) and then it will convert the code to a native application for both platforms.

[1] Mobile & Tablet Operating System Market Share Saudi Arabia | StatCounter Global Stats https://gs.statcounter.com/os-market-share/mobile-tablet/saudi-arabia/#monthly-201809-201909-bar.

Wafferly will interface with an API provided by Google; which is Google Vision API [6]. It shall enable Wafferly, given a certain image by the user, to do image processing to help get the attributes of the product image. Furthermore, Wafferly shall enable the discovery of keyword synonyms in order to broaden the search. Finally, Wafferly will do web scraping to get search results from Google Shopping service.

This chapter will cover four sections. The first section will clarify the application domain. The second section will include the features that will be used from Google Vision API. The third section shall cover the development environment used in Wafferly, finally it will cover word synonyms.

2.1. Application Domain

Market comparison services (MCS) enable consumers to compare product prices, by enabling the consumers to search for a product and show prices from a variety of online shops [7]. They also provide different features like sorting and filtering the products to meet the consumers' needs and making the comparison process easier. The results that are displayed to the consumers are retrieved from an aggregate database of online shops that MCSs curate and maintain. The MCSs results cover most online stores, as a result consumer shall have better knowledge about market prices, and that is the main benefit of such services. Moreover, it benefits consumers by saving their time and effort [7]. Business Matters, a United Kingdom (U.K.) magazine published in July 2018, stated that 77% of consumers in the U.K. used MCS to switch their car insurance. Furthermore, the report gave reasons why consumers use these websites. According to the results depicted in Figure 2, 83% of MCSs consumers who use MCSs use them mainly to compare prices [8].
MCSs are not only useful for consumers but also for businesses as well. The small companies are the first ones to take advantage from MCSs, for example a small company sells a product with a low price, at the same time a large company sells the same product for a higher price. In the former case, the consumers will choose to buy the cheaper one or at least they will be introduced to that company. On the other hand, MCS might not provide the full picture to the consumer by showing only affiliated shops (i.e. shops that pay the MCS a fee for being featured). This might provide a biased view of the market which prevents consumers from making a well-informed decision [9].

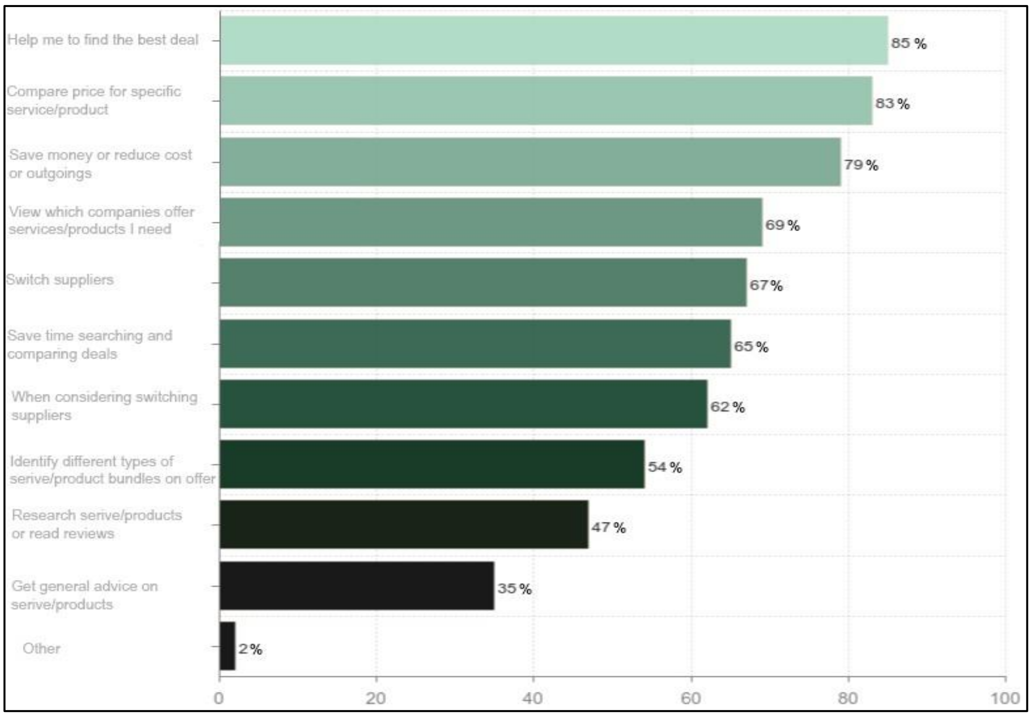


Figure 2 Why consumers use MCSs [12]

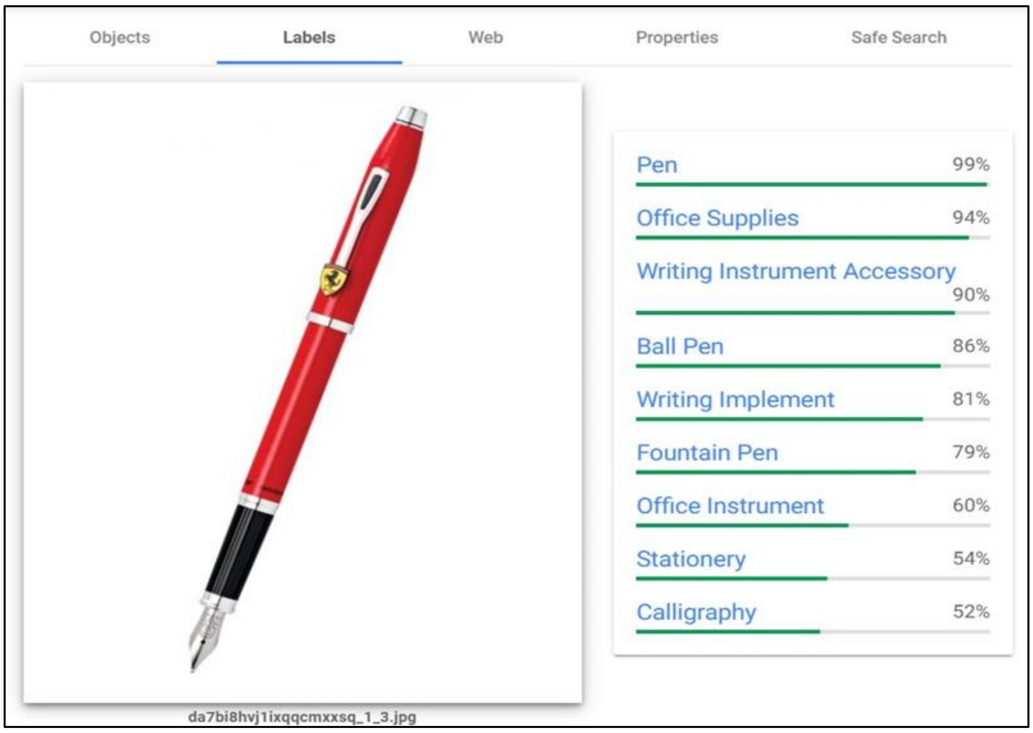
2.2.Google Vision API

Google has two Artificial Intelligent (AI) and machine learning products: autoML Vision and Vision API, the difference between these two products is that autoML Vision is a custom machine learning model which users can train, whereas the Vision API provides a model that is already trained by Google.

Google Vision is a REST API which can be invoked using an HTTP POST request. Upon sending the request, it shall include the image as base64 encoded string if it is found locally on the google cloud storage or by providing the URL of the image. When the request is successful the API shall return an HTTP response with status code 200 including the response in JSON [8]. Vision API provides several detection features such as image label detection, text detection, logo detection, object detection, detect image properties and web detection, which will be further investigated in the following sections.

2.2.1. Image labeling

This detection will get information from a picture like where it was taken, and the time shown in a picture. Furthermore, it can detect objects in a picture like the location (e.g. street, town, alley), products and animals. The labels are returned in English only, but Google provides a cloud translation to use if needed. The server returns a JSON file with these values: the description (label), score (confidence) and topicality (importance of the label in image) as shown in Figure 3 [10].



Pen	99%
Office Supplies	94%
Writing Instrument Accessory	90%
Ball Pen	86%
Writing Implement	81%
Fountain Pen	79%
Office Instrument	60%
Stationery	54%
Calligraphy	52%

Figure 3 Example of label detection

2.2.2. Logo Recognition

The tool can also detect logos in an image. However, it can only recognize popular brands. The server returns a JSON object with these values such as the description (the name of the logo) and the score (confidence) [11].

2.2.3. Text Detection (Optical Character Recognition)

This detection tool will extract text from an image. For example, given a picture of a product, the tool detects and returns what is written on that product sticker like the brand. This information returned as JSON which includes all the extracted text. This feature does support many languages including Arabic [12]

1.1.Development Environment

1.1.1. React

React is an open-source JavaScript library for building user interfaces (UIs) which is created by Facebook. It uses components and virtual document object model (VDOM) for building the view layer of webpages and mobile applications. The components are rendered into HTML by React VDOM which is responsible for updating the real DOM [12].

1.1.2. React Native

React Native is an open source framework for developing both iOS and Android applications. React Native is also created by Facebook. It uses React library for building UI components like Text and View and then they are rendered to native code by mapping them directly to the platform’s native UI building block [5].

1.1.3. Expo

Expo is a framework that has services and tools that are built on top of React Native. It allows developing iOS and Android applications using the same codebase. One of the main features that Expo provides is the ability to run a project without XCode or Android studio [13].

1.2. Google Shopping

Google Shopping is a service provided by Google, it allows the consumers to search for products and display them from different stores. On the other hand, Google Shopping API helps the sellers to broaden their reach by adding their information and the details about their products so it will be displayed in Google shopping search page [15].

1.3.Word Synonyms

Synonyms are words that have the same meaning to another words. In Wafferly application we need Synonyms to expand the search keywords, and eliminate the fragmentation
of the language used to search for and describe products. So, we considered incorporating a component in our system to detect word’s synonyms. The investigated methodologies are AraVec, WordNet and crowed-sourcing an Arabic synonyms lexicon.

2.7.1 AraVec

AraVec is an open source Word2Vec. Word2Vec is a word embeddings model that discovers the words that have the same meaning, by assigning them a similarity score that increases if the words appear in the same context. AraVec provides an Arabic model that is trained on different domains like Twitter and Wikipedia [16].

2.7.2 WordNet

WordNet is a lexical database that links word in semantic relation, WordNet is freely and publicly available and supports more than 200 language. The main relation is synonym that links words based in their concept and meaning [17].

2.7.3 Crowed-Sourcing

Crowed-Sourcing is a process of engaging groups of individuals for a specific goal. It is useful in solving problems by involving information and opinion from the group. Its benefit lies in the speed to reach those group via internet, also the difference in the way of thinking of individuals which can lead to good solutions, that could take longer if solved computationally. However, there are a problem when using Crowed-Sourcing techniques: as the number of individuals increases, noise may be introduced to the data since not all human input is reliable. So, to solve this problem it necessary to control it with weights which represent the agreement of human participants; the threshold of acceptable weight is typically adjusted according to the noise and volume of participants [18].

Chapter 3 | LITERATURE REVIEW

2. LITERATURE REVIEW

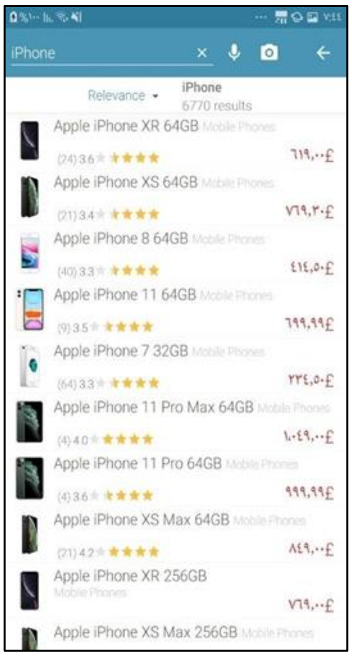
This chapter includes a review of different market comparison services (MCSs). Throughout this chapter, MCSs have been discovered by searching relevant keywords like ‘price comparison’ and ‘market price comparison’. The obtained results of the search phase were huge since it had a lot of websites and applications that provide the same service. Therefore, the results got reduced to the mobile applications which we intend to do as well, so it will provide a fair comparison between our services and their services. The list could be helpful in knowing the application domain, set of user-expected features, used technologies and several drawbacks and to take them into account while designing and implementing the application. It will be useful to know what the trending features are and see if Wafferly is going to meet the consumers’ expectations.

In the following sections, we review the final list of competitive and similar applications, highlighting how they are different from Wafferly.

2.1.International Services

2.1.1. PriceSpy

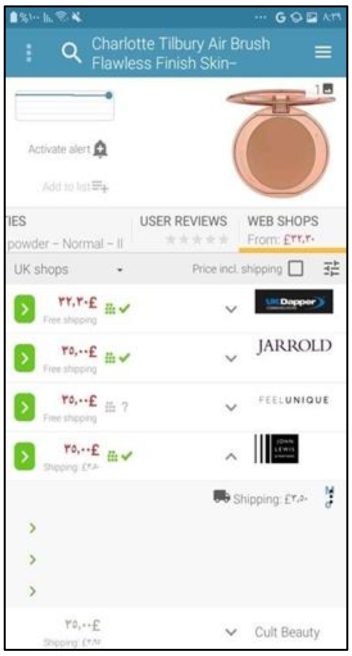
The application PriceSpy provides product comparisons for more consumers by targeting online stores in Sweden, Norway, Finland, Great Britain, Ireland, New Zealand, France, and Italy. Figure 6 shows the homepage of the application which displays the Categories (e.g. Daily Deals, Home & Garden and Kids & Toys). PriceSpy has the ability to search for products by either entering the product's name, searching by voice or scanning a barcode but it requires internet connection to search. The result of the search will display a list of products (see Figure 4). The details on the products are divided into five sections, the first section is physical shops which display the shops that provide the product and their locations if any. The second section includes web shops that provide the product including the prices with the shipping fee (see Figure 5). The third section includes consumers' reviews about the product; whereas the fourth section shows the product properties; and the last section shows other similar products. The search results can be filtered by category, by price range, by product rating range, by shop rating range or by stores. Moreover, PriceSpy features offers, which can be filtered by name, popularity, lowest price, highest price, price drop.
On the other hand, PriceSpy does not provide search by image. Also, it does not filter by the delivery date [19].


iPhone

Relevance ▾ iPhone 6770 results

Apple iPhone XR 64GB Mobile Phones	(24) 3.6 ★★★★	٦١٩,٠٠£
Apple iPhone XS 64GB Mobile Phones	(21) 3.4 ★★★★	٧٦٩,٣٠£
Apple iPhone 8 64GB Mobile Phones	(40) 3.3 ★★★★	٤١٤,٥٠£
Apple iPhone 11 64GB Mobile Phones	(9) 3.5 ★★★★	٦٩٩,٩٩£
Apple iPhone 7 32GB Mobile Phones	(64) 3.3 ★★★★	٢٣٤,٥٠£
Apple iPhone 11 Pro Max 64GB Mobile Phones	(4) 4.0 ★★★★	١,٠٤٩,٠٠£
Apple iPhone 11 Pro 64GB Mobile Phones	(4) 3.6 ★★★★	٩٩٩,٩٩£
Apple iPhone XS Max 64GB Mobile Phones	(21) 4.2 ★★★★	٨٤٩,٠٠£
Apple iPhone XR 256GB Mobile Phones		٧٦٩,٠٠£
Apple iPhone XS Max 256GB Mobile Phones		

Figure 6 Homepage


Charlotte Tilbury Air Brush Flawless Finish Skin–

Activate alert

Add to list

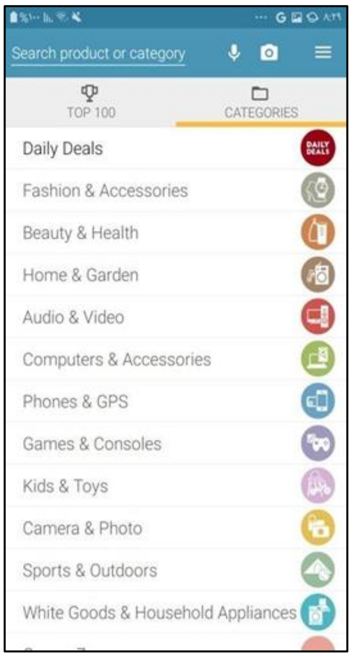
IES powder – Normal – II | USER REVIEWS ★★★★★ | WEB SHOPS From: £٢٢,٣٠

UK shops ▾ Price incl. shipping ☐

٢٢,٣٠£ Free shipping	✓	UKDapper
٢٥,٠٠£ Free shipping	✓	JARROLD
٢٥,٠٠£ Free shipping	?	FEELUNIQUE
٢٥,٠٠£ Shipping: £٢,٥٠	✓	JOHN LEWIS
٢٥,٠٠£ Shipping: £٢,٨٠		Cult Beauty

Shipping: £٢,٥٠

Figure 5 Review an item


Search product or category

TOP 100 | CATEGORIES

- Daily Deals
- Fashion & Accessories
- Beauty & Health
- Home & Garden
- Audio & Video
- Computers & Accessories
- Phones & GPS
- Games & Consoles
- Kids & Toys
- Camera & Photo
- Sports & Outdoors
- White Goods & Household Appliances

Figure 4 Result of the search

2.1.2. Idealo

Idealo provides products comparison for Germany, France, Austria, Italy, Spain, and the U.K. Figure 7shows the homepage of the application which displays the last search viewed for the consumer, suggestion and top deals. As in PriceSpy, Idealo provides search for products by entering the product's name, search by voice or scan a barcode. The search result can also be filtered and sorted, but when compared with PriceSpy, filtering in Idealo is limited to filtering the result by price range, by category or by consumers' reviews. Sorting in Idealo is the same as PriceSpy. Idealo differs from PriceSpy with the details of the product in the five sections that are mentioned above, where Idealo provides seven sections, the first section includes the variants for a product (e.g. if the product is iPad the variants will be similar iPads with different memory, size and color etc.), the second section shows the stores that provide the product, the payment method and delivery duration as shown in Figure 9. The third section provides the specification of a product. The fourth section includes consumers' reviews about the product in the form of average star ratings. The fifth section provides expert reviews about the product. The sixth section as showing in Figure 8 includes price history which represents the price changes in a graph, it can represent it in one month, three months and six months, it also calculates the average price, lowest price and the highest price. The last section provides similar products.

On the other hand, Idealo does not provide image searching, it also does not provide sorting by delivery date. And like PriceSpy it does not search for products in the independent retailer on social media [20].

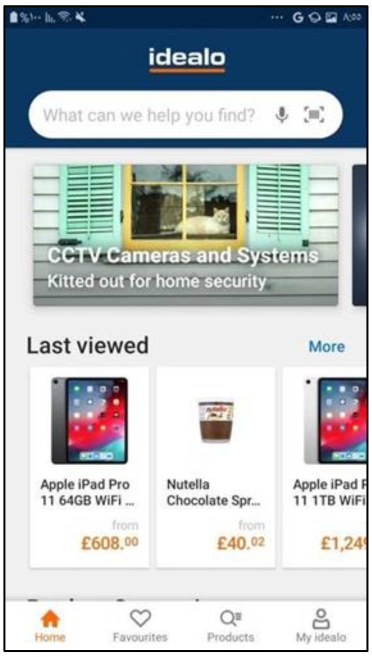


Figure 7 Homepage

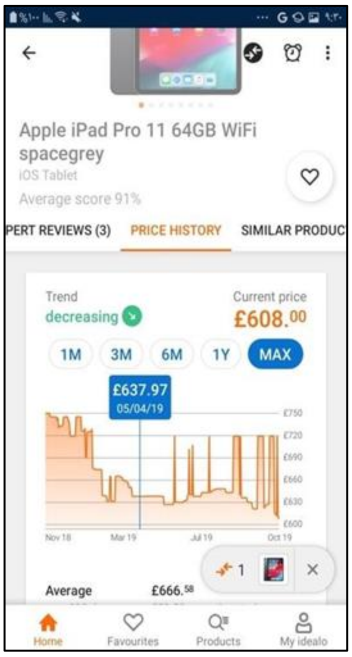


Figure 8 price history

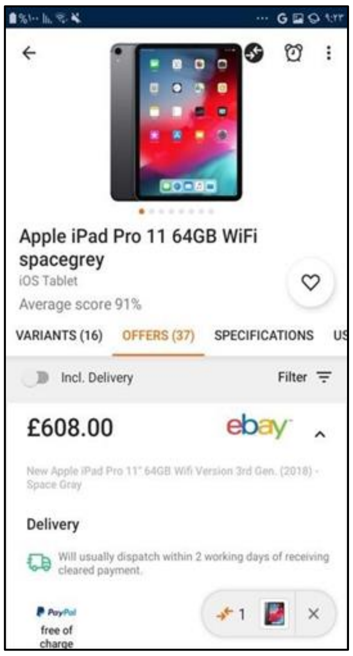

Apple iPad Pro 11 64GB WiFi spacegrey
iOS Tablet
Average score 91%
VARIANTS (16)
OFFERS (37)
SPECIFICATIONS
Incl. Delivery
Filter
£608.00
ebay
New Apple iPad Pro 11" 64GB Wifi Version 3rd Gen. (2018) - Space Gray
Delivery
Will usually dispatch within 2 working days of receiving cleared payment.
PayPal
free of charge

Figure 9 Review an item

2.1.3. Scrooge

Scrooge is only slightly different from the above applications; it provides price comparison for U.K. online stores. The homepage of Scrooge is a with a search field. As in PriceSpy and Idealo, Scrooge can search for products by entering the product's name and search by voice as shown in Figure 10, where it is lacking to search by image. The filter of the results depends on the type of product (e.g. when the products are clothes the filter will be by color, by gender, etc.) as shown in Figure 12 and the sort can be done by price and popularity. The view for details of product is the same as Idealo except that scrooge provides more detailed price history, which shows a graph of the price changing in one month, four months and six months. On the other hand, Idealo does not provide image searching, also it does not provide sort by delivery date [21].

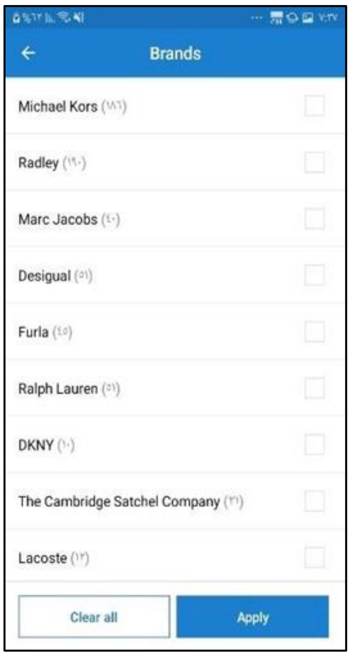


Figure 12 Filter by brand

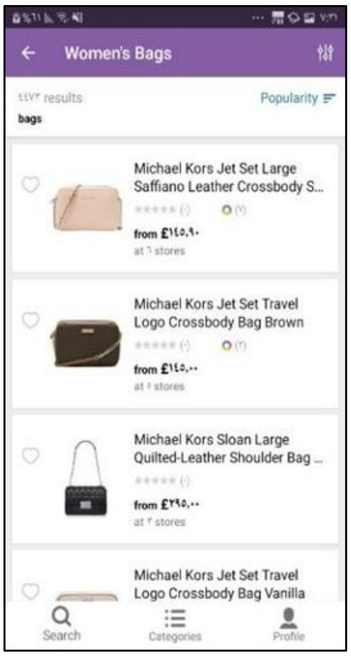

Women's Bags
results
bags
Popularity
Michael Kors Jet Set Large Saffiano Leather Crossbody S...
Michael Kors Jet Set Travel Logo Crossbody Bag Brown
Michael Kors Sloan Large Quilted-Leather Shoulder Bag ...
Michael Kors Jet Set Travel Logo Crossbody Bag Vanilla
Search
Categories
Profile

Figure 11 Result of the voice search

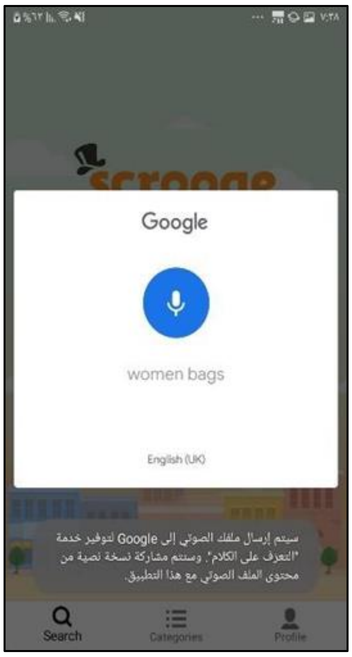


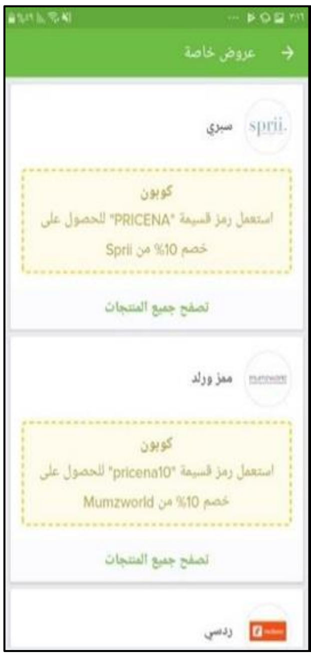
Figure 10 Voice search

2.2.Local Services

2.2.1. Pricena

Pricena application provides product comparison and its main target are Middle Eastern online consumers. The home page of the application displays the recent price drop for the products the consumer is interested in and the products that are popular in user's country Also, it provides recommendations for products from different categories. Figure 13Figure 15shows the section of the offers from different online stores (i.e. coupons).

The application provides several ways for the consumer to search for a product. Consumer can search for a product by entering the product's name, importing/taking its picture (see Figure 13) or by scanning the barcode but it's like the previous application as it requires internet connection to search. The result of the search will be a list of products. The results can be filtered either by the brand as shown in Figure 16, store, category or price range. After the consumer selects a product, the stores that sell the product will be displayed to the consumer sorted by the cheapest price to the most expensive. It also shows if the product is currently available in these stores. Pricena doesn't only display the online stores that sell the product but also displays the nearest stores to the consumer's location.



عروض خاصة
سبري
كوبون
استعمل رمز قسيمة "PRICENA" للحصول على خصم 10% من Sprii
تصفح جميع المنتجات
ممز ورلد
كوبون
استعمل رمز قسيمة "pricena10" للحصول على خصم 10% من Mumzworld
تصفح جميع المنتجات
ردسي

Figure 15 Available offers

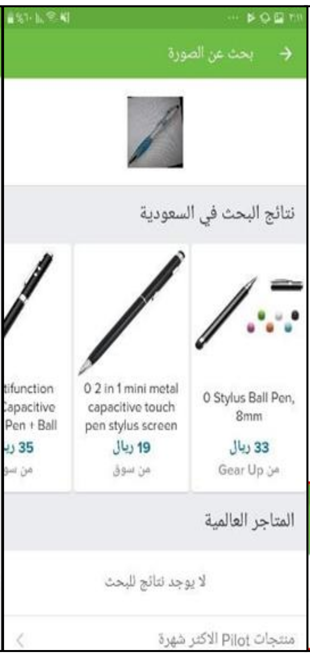
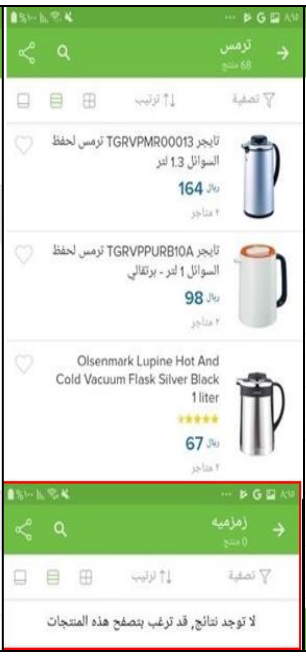


Figure 13 Result of image search



ترمس
68 منتج
ترتيب
تصفية
تايجر TGRVPMR00013 ترمس لحفظ السوائل 1.3 لتر
164 ريال
تايجر TGRVPPURB10A ترمس لحفظ السوائل 1 لتر - برتقالي
98 ريال
Olsenmark Lupine Hot And Cold Vacuum Flask Silver Black 1 liter
67 ريال
زمزمية
0 منتج
لا توجد نتائج قد ترغب بتصفح هذه المنتجات

Figure 14 Fragmentation of words

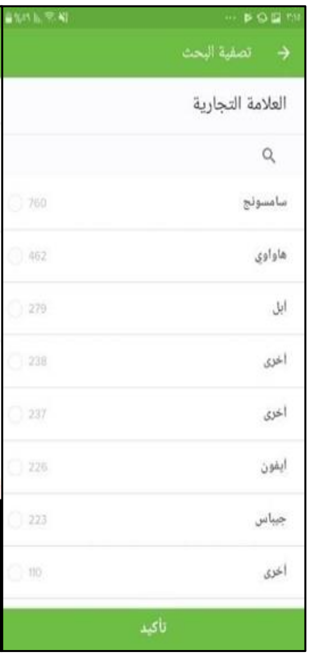


Figure 16 Filter by brand

But there are some features that are not provided by Pricena including search by voice and sorting the results based on date of delivery. According to Figure 14, Pricena might suffer from the keyword fragmentation problem. For example, when the consumer searches for (‘ترمس’), the word got translated in English to (‘Thermos’) correctly; but when the consumer to searches for (‘زمزمية’), there were no results even though the latter has the same meaning as the former and it is widely used by the Middle Eastern society [22] .

2.2.2. KSA Price

KSA Price is another local service application that provides product comparison for Saudi online stores. It differs from Pricena on the homepage since KSA Price doesn’t support Arabic in its interface and it has a special button in the homepage for the refresh. Also, it doesn’t provide recently viewed products on the homepage as is done by Pricena. And it also requires internet connection to search. Moreover, it doesn't enable searching in Arabic. Figure 19 shows the search results after searching with the keywords "سماعة" also "كتاب" and Figure 18 shows the search result for after entering English keywords. The application focuses on electronic products.
KSA Price only provides search by product name. The result of the search can be filtered by the price range and it shows only one store that sell the product. Figure 17 shows the offers page that includes the catalogue of different supermarkets while showing where these offers are available, these offers can be filtered by city. Furthermore, it doesn't allow the customer to favorite products they need. also, KSA price does not offer a way to give feedback about the product [23].

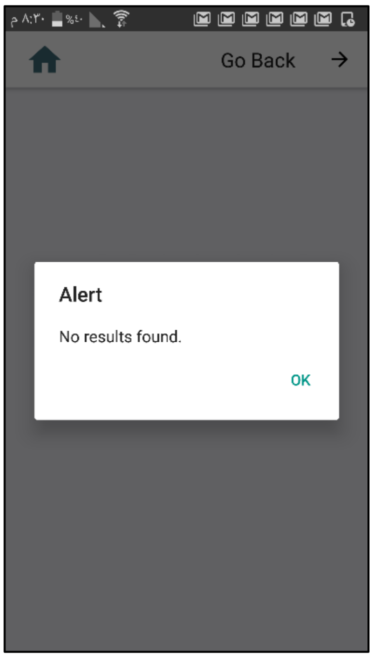


Figure 19 Result of the Arabic search

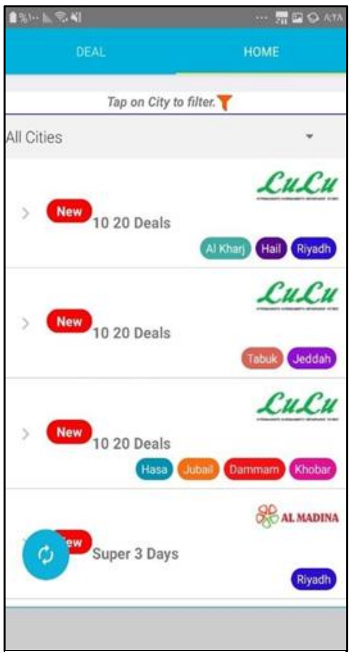


Figure 17 Available offers

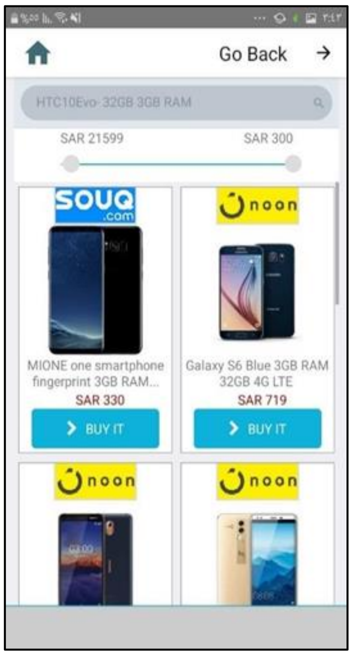

Go Back
HTC10Evo 32GB 3GB RAM
SAR 21599
SAR 300
MIONE one smartphone fingerprint 3GB RAM...
SAR 330
BUY IT
Galaxy S6 Blue 3GB RAM 32GB 4G LTE
SAR 719
BUY IT

Figure 18 Result of the English search

2.2.3. Kanbkam

Kanbkam application provides product comparison for the Middle East and North Africa online stores. The homepage of the application shows the daily deals and the different categories it has, such as Mobile & Tablets, Kids & Babies, Cameras and Grocery. The homepage interface supports both Arabic and English languages. Also, the homepage has a sidebar that contains all categories, button to change the interface language, deals, and the top discounts. As in KSA Price the search for a product can only be done by entering its name however, Kanbkam provides a search in Arabic. The search result can be sorted by popularity, low to high price, high to low and recently updated. Also, the results can be filtered by category, brand, price range, seller and price drop. Figure 20 shows the offers and the current drop in the price for each product. Furthermore, in the product details page, Kanbkam provides an extra feature which is a diagram showing the increases and decreases in the product price (see Figure 21). Moreover, it shows a table that has the lowest and the highest prices also recent price increases. It can be visited by its website or using the mobile application which is supported in Android platform and it required internet connection [24].

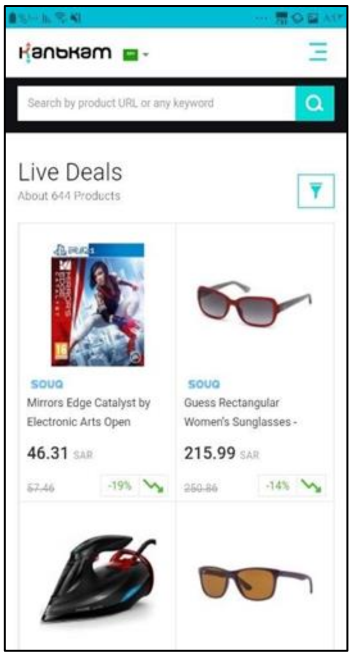

Kanbkam
Search by product URL or any keyword
Live Deals
About 644 Products
souq
Mirrors Edge Catalyst by Electronic Arts Open
46.31 SAR
57.46
-19%
souq
Guess Rectangular Women's Sunglasses -
215.99 SAR
250.86
-14%

Figure 20 Available offers

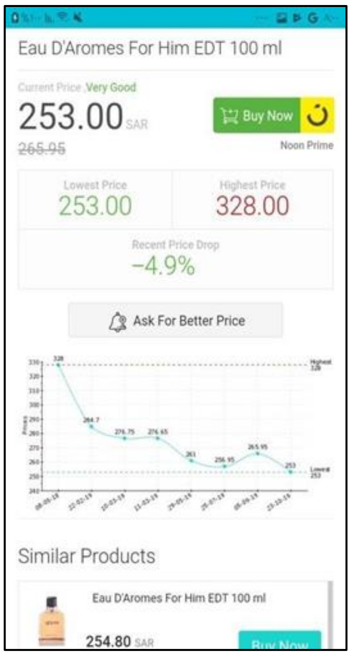
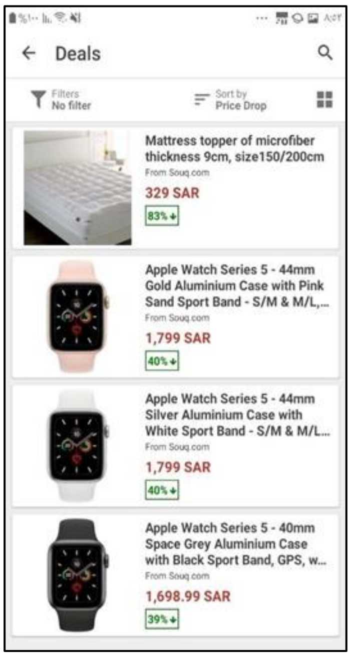


Figure 21 Diagram of price

2.2.4. Yaoota

Yaoota is also one of the local applications that provide products comparison for the Saudi Arabia, Kenya, Nigeria and United Arab Emirates online stores. Its homepage is similar to Kanbkam application, it displays the categories of the products, but it doesn’t show the daily deals. As shown in Figure22 deals are displayed in a separate page. Yaoota provides a registration page to allow customers to enter their reviews and rate for a product, the customers can reach their reviews and rating by moving to the 'reviews' section in the sidebar. also, the sidebar shows a profile name and photo. Furthermore, Yaoota provides the ability to favorite certain products, the customers can also reach their favorites via the 'favorite' section in the sidebar. Moreover, the sidebar includes category and settings section. the setting section consists of Yaoota logo and its version also it contains a bar to change the county and the language. Another feature that is similar to Kanbkam is sorting the results of the search. Sorting can be done also based on the price, popularity, price drop, and the relevance to the search keywords. Moreover, the results in Yaoota can be also filtered by category, stores and price range. As in KSA Price and Kanbkam, Yaoota only searches by the product name it does not provide search by image, voice, or barcode. And it also requires internet connection [25].



Mattress topper of microfiber thickness 9cm, size150/200cm
From Souq.com
329 SAR
83%

Apple Watch Series 5 - 44mm Gold Aluminium Case with Pink Sand Sport Band - S/M & M/L,...
From Souq.com
1,799 SAR
40%

Apple Watch Series 5 - 44mm Silver Aluminium Case with White Sport Band - S/M & M/L...
From Souq.com
1,799 SAR
40%

Apple Watch Series 5 - 40mm Space Grey Aluminium Case with Black Sport Band, GPS, w...
From Souq.com
1,698.99 SAR
39%

Figure22 Available offers

All the previous applications KSA price, Kanbkam, Yaoota, do not sort the result by the delivery date.

Table 1 : Applications Comparison

Logo	Name	Supported interface languages (AR, EN)	Search by voice	Search by image	Barcode scanning	Provide offers
pricena search · compare · save	Pricna	Arabic English	×	√	√	√
PriceSpy	Pricespy	English	√	×	√	√
idealo	Idealo	English	√	×	√	×
	Scrooge	English	√	×	×	×
KSA price	Ksaprice	English	×	×	×	√
kanbkam	Kanbkam	Arabic English	×	×	×	√
yaoota!	Yaootah	Arabic English	×	×	×	√
	Wafferly	Arabic	×	√	×	×

2.3.Discussion

After analyzing the locally provided applications: Pricena, scrooge, PriceSpy, Idealo, KSA Price, Kanbkam and Yaoota, it is clear that there is no single application that combines most features to enhance the consumer's experience and meet consumer's satisfaction. The comparison between Wafferly and all the other applications can be seen in Table 1. In fact, we found that 4 out of the 7 reviewed applications did not support search by voice, whereas, 6 out of 7 of the applications did not use search by images. Therefore, we decided that using the search by images feature is very important since it facilitates the searching process. The image processing feature will allow the consumer to take a picture of the product or import it to the platform. It will be more convenient for the consumer since the consumer does not have to have the product in front of him/her to physically scan the barcode. It will also help with solving the problem of dialect differences in product listings among the online store directory. In fact, only 3 out of 7 of the applications use barcode scanning which proves that the feature has alternatives that could give the same result, for instance Wafferly uses search by image which almost gives results in the same manner. Also, 2 out of 7 of the applications did not use 'show offers' which will be helpful to include it in Wafferly, but due to the time limitation of the project, we cannot currently include this feature in Wafferly.

Chapter 4 | SYSTEM ANALYSIS AND DESIGN

3. SYSTEM ANALYSIS

This chapter explains the system of Wafferly and how it works, first it describes the user characteristics which will discuss the different attributes of the consumers of the system, then it will cover the specific requirements of the system: functional and nonfunctional Finally, this chapter shall discuss the modeling of the system, which includes: the use case diagram, use case description, sequence diagrams, and finally, the conceptual and class diagrams.

3.1.User Characteristics

Wafferly is aimed to assist consumers who need to find a shop that provides the cheapest price for a product. Mainly the type of users which Wafferly will serve are the ones with interest in finding a product at a lower price, or different preferred criteria such as the highest rating. The application will have one type of user which is the consumer who will be using the application to search for a product. Consumers only need a basic knowledge of using mobile applications and can read/write Arabic. The website will be in Arabic.

3.2.Specific Requirements

The requirements were gathered from a survey (see APPENDIX A) that gathered over 276 responses. The first section of the survey focused on the requirements related to the online store. We observed that 68.1% respondents (188 out of 276) sometime purchase online. Also, we perceived that 48.2% respondents (133 out of 276) always search in different online stores for a cheaper price. Furthermore, we asked if they need an application that compare product in a different online store, 94.2% respondents (188 out of 276) said yes. We have also asked specific questions to take a deep view of the solution. So, we asked if users want the application to provide a search by image feature, 87.7% respondents (242 out of 276) said yes. Also, we asked if they choose a product based on its rating, 76.8% respondents (212 out of 276) said yes. In addition to the findings of the survey, the requirement was collected and finalized by conducting the domain analysis and literature review. The application has some constraints which are: Internet connection and basic knowledge of using mobile devices.

3.2.1. Functional Requirements

1 The consumer shall be able to search for a product by descriptive keywords in different stores.

1.1 The system shall allow the consumer to enter keywords.

1.2 The system shall look for the synonyms of the keywords.

1.3 The system shall display all the matching products from online stores based on the product synonyms.

2 The consumer shall be able to search for a product by its image in different stores.

2.1 The system shall allow the consumer to import/take an image.

2.2 The system shall get the textual attributes of the image.

2.3 The system shall display all the matching products based on image attributes.

3 The consumer shall able to sort the result by price.

3.1 The system shall allow the consumer to sort the results based on price.

3.2 The system shall display the list of the sorted results.

4 The consumer shall be able to sort the result by consumer ratings.

4.1 The system shall allow the consumer to sort the results based on ratings.

4.2 The system shall display the list of the sorted results.

3.2.2. Non-Functional Requirements

1. **Reliability:**

1.2 The system shall be able to recover within no more than one hour and continue working normally.

2. **Usability:**

The user shall be able to learn how to use the system within no more than 2 minutes.

3. **performance:**

3.1 The response for the system shall take no longer than 5 seconds [26]

4. Availability:

4.1 The system shall be available 90% of the time.
4.2 The system shall be accessible from anywhere in the world via the Internet.

5. Security:

5.1 The system database shall be secure.

3.3.System Models

3.3.1. Use Case Diagram

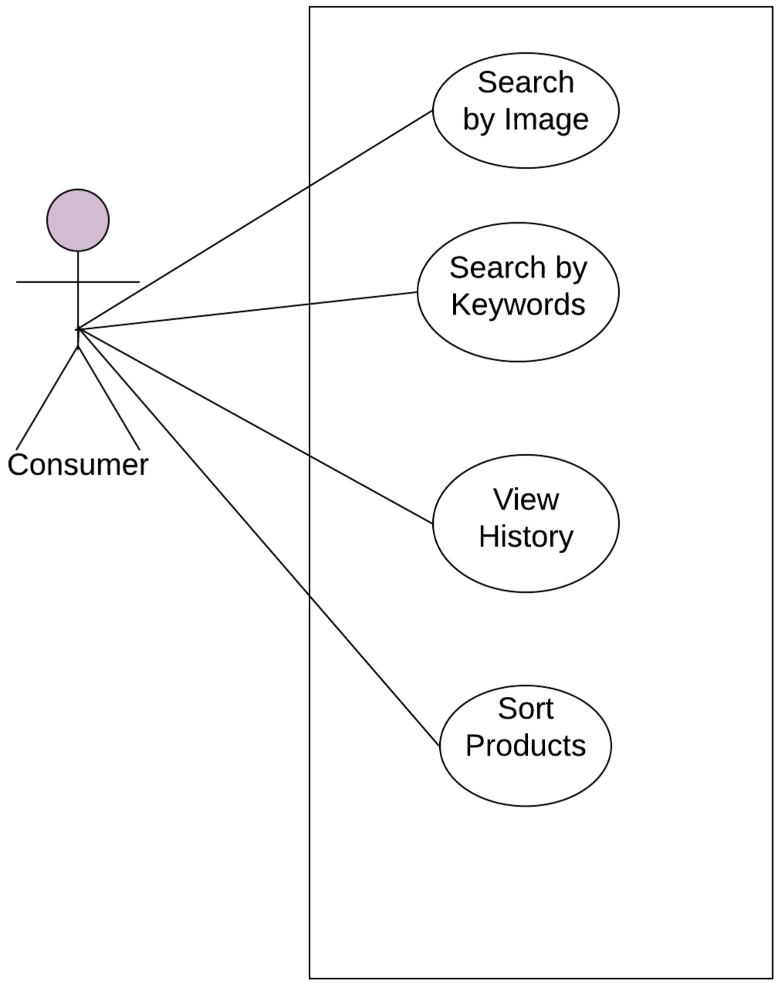


Figure 23 Use case diagram.

3.3.2. Use Case Description

The consumer's use cases:

Table 2: Search by keyword

Use case name	Search by keyword.	**Actor**	Consumer
Purpose	To allow the consumer to search for a product by keywords.		
Cross reference	R1		
Type	Primary, Essential		
Typical course of events	**Actors**	**System**	
	1- The use case starts when the consumer wants to search for a product by its keywords.		
	2- The consumer types keywords of a product	3- The system takes the keywords.	
		4-System searches for the word's synonyms	
		5-The system displays a list of the sorted product and their details.	
Pre-conditions	none		
Alternatives	none		

Table 3: Search by image

Use case name	Search by image	**Actor**	Consumer
Purpose	To allow the consumer to search for a product by its image.		
Cross reference	R2		
Type	Primary, Essential		
Typical course of events	**Actors**	**System**	
	1- The use case starts when the user wants to search for a product by its image.		

	2- The user selects the search by image option and upload/take an image of a product.	3- The system takes the image and extracts its attributes.
		4- The System search using the extracted attributes.
		5 - The system displays a list of the sorted product and their details.
Pre-conditions	none	
Alternatives	Step 5: If the result does not match the image that was uploaded, the user can search by keywords	

Table 4: Sort the result

Use case name	Sort the result	**Actor**	Consumer
Purpose	To allow the consumer to sort the products by their price or rating.		
Cross reference	R3, R4		
Type	Primary, Essential		
Typical course of events	**Actors**	**System**	
	1- The use case starts when the consumer wants to sort the products.	2- The system displays the sort option.	
Pre-conditions	The consumer must search for a product.		
Alternatives	none		

3.3.3. Sequence Diagrams

Search by keyword:

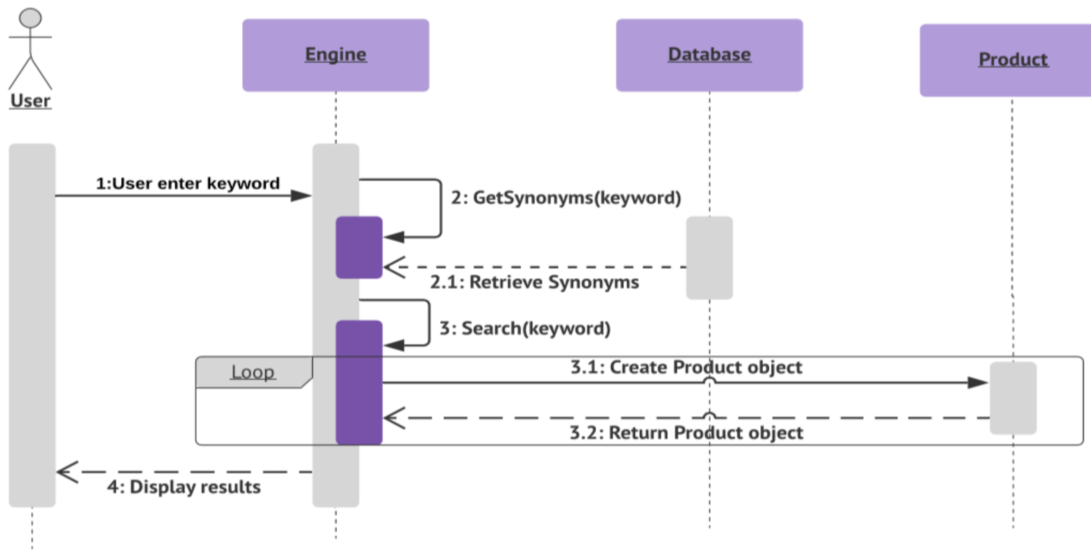


Figure 24: Search

Extract image attribute:

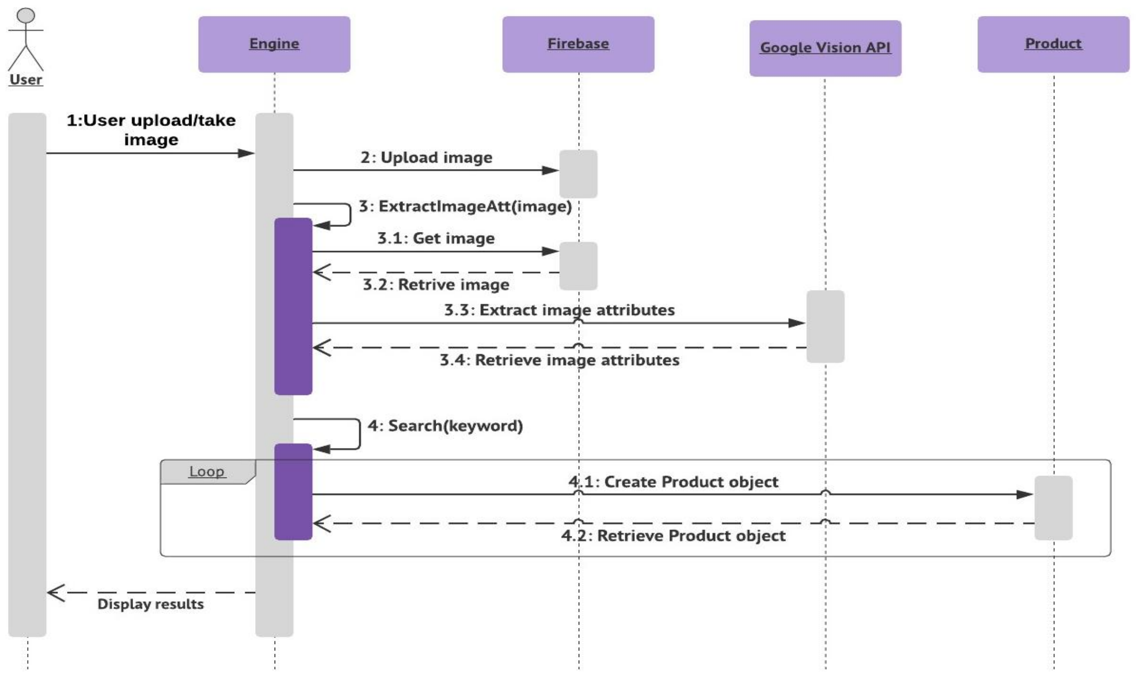


Figure 25 Search by image

Sort the result:

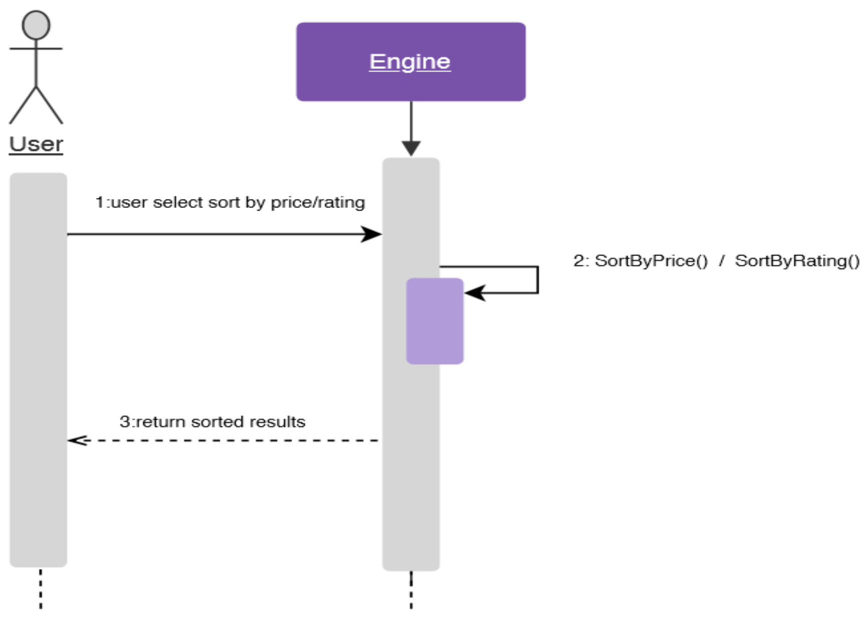

User
Engine
1:user select sort by price/rating
2: SortByPrice() / SortByRating()
3:return sorted results

Figure 26 Sort the result

3.3.4. Conceptual Diagram

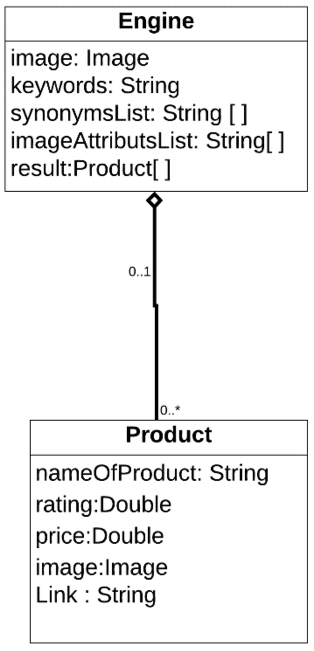


Figure 27 Conceptual Diagram

3.3.5. Class Diagram

Class diagram shows the classes and the relationship between them. It also shows the attributes and the methods in each class. The Figure 28 shows the class diagram of Wafferly application.

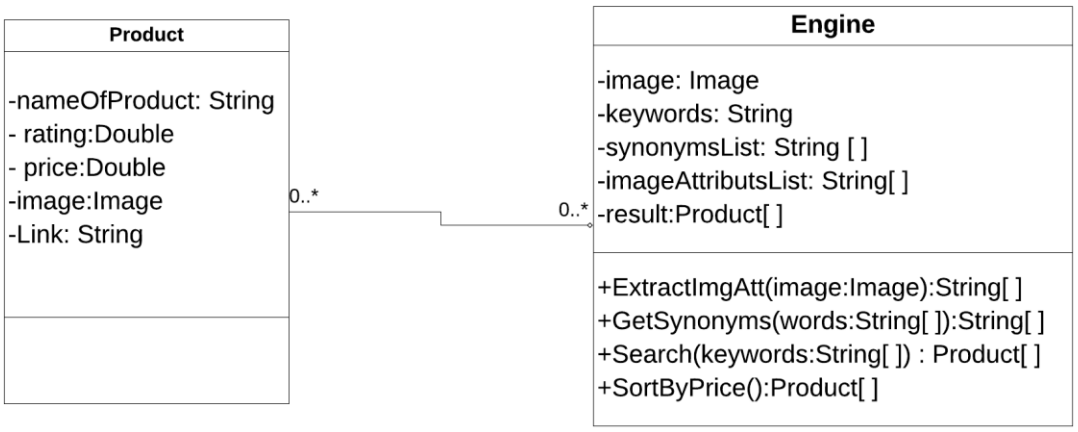

Product
-nameOfProduct: String
- rating:Double
- price:Double
-image:Image
-Link: String
0..*
0..*
Engine
-image: Image
-keywords: String
-synonymsList: String [ ]
-imageAttributsList: String[ ]
-result:Product[ ]
+ExtractImgAtt(image:Image):String[ ]
+GetSynonyms(words:String[ ]):String[ ]
+Search(keywords:String[ ]) : Product[ ]
+SortByPrice():Product[ ]

Figure 28 Class Diagram

3.4.System Design

3.4.1. Architectural design

Wafferly, in its design, follows a Model-View-Controller (MVC) architecture. The application starts with the view which is the user interface which is implemented using Expo. When the user searches for products by keywords the view will trigger an HTTP request to the server. The controller receives the entered keywords and prepares them before sending them to the 'get synonyms' function which is a part of the model. The search function will start after getting the new keywords list from the get synonyms function. The search function will return the search results of products as a JSON object back to the controller which will send it back to the view.

On the other hand, when the user searches by image, the view will upload the image to a server (Firebase), then it will trigger an HTTP request to the search controller on the server. The controller receives the image's URI and sends it to Google Vision to analyze the image and extract the image attributes. The list of attributes is then forwarded to the search function as keywords which will operate as described earlier.

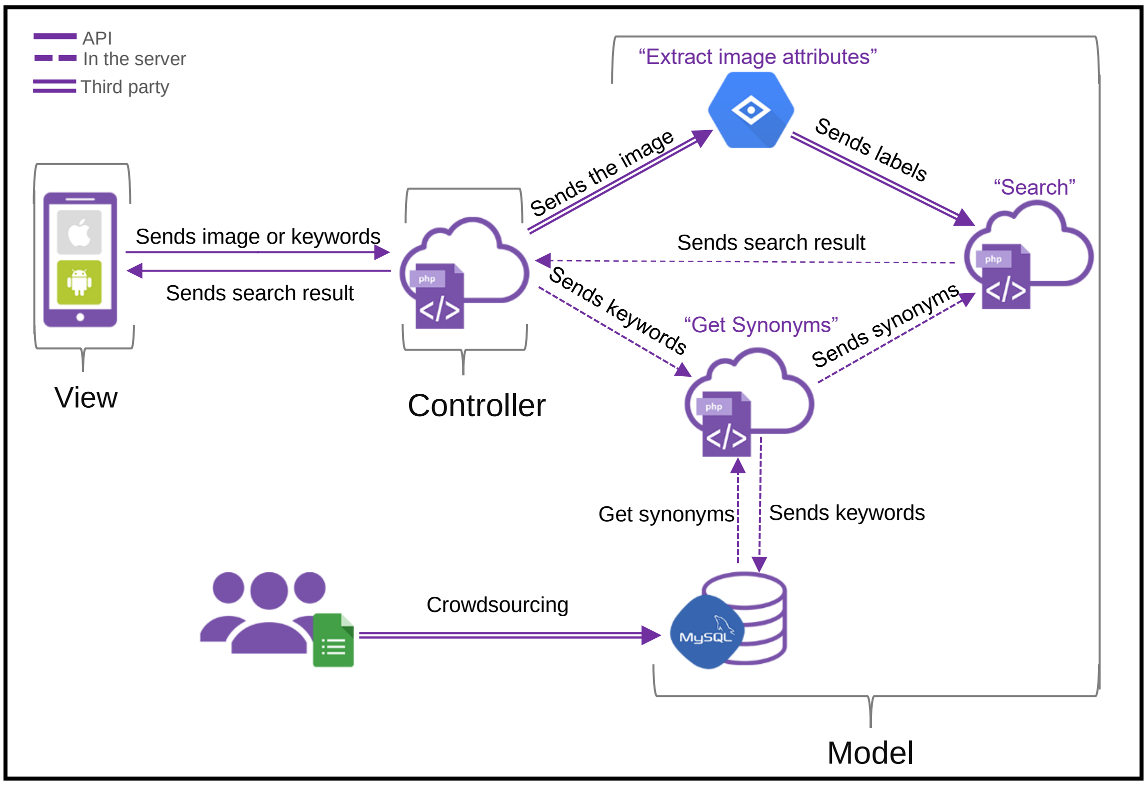


Figure 29 System architecture

3.4.2. Component Design

1. Search by keywords

Classification: Function

Definition: It enables the consumer to search for a product by keywords.

Responsibilities: R1, R2

Pre-condition: None

Pseudo Code:

```
BEGIN
DISPLAY the search page
                GET the synonyms
                START the Search
                DISPLAY Result
        ENDIF
 END
```

2. Search by image

Classification: Function

Definition: It enables the consumer to search for a product by an image.

Responsibilities: R2

Pre-condition: None

Pseudo Code:

```
BEGIN
        DISPLAY the search page
        IF the consumer clicks on the search by an image button
                DISPLAY upload image options
                        IF the consumer selects import from album
```

```
                    DISPLAY album
                    READ the image
                ENDIF
                ELSE IF the consumer selects take an image
                    DISPLAY camera
                    READ the image
                    ENDIF
            EXTRACT the image attributes
            DISPLAY result
    ENDIF
END
```

3. Get Synonyms

Classification: Function

Definition: It enables the system to search for keyword's synonyms.

Responsibilities: R1

Pre-condition: None.

Pseudo Code:

```
BEGIN

        IF the consumer submits inputs
            READ the keywords
            IF keywords in the database
                FIND the synonyms
                SEND synonyms
            ELSE
              ADD keyword to database
              SEND keyword
            ENDIF
        END
```

4. Sort the result

Classification: Function

Definition: It enables the consumer to sort products by price and rating.

Responsibilities: R3, R4

Pre-condition: The consumer must choose a product.

Pseudo Code:

```
BEGIN
    DISPLAY the search result
    IF the consumer selects a sort option
        WHILE (search result not empty)
            SORT product by the selected option
        END WHILE
        DISPLAY products sorted by the selected option
    ENDIF
END
```

3.4.3. User Interface Design

User Interface Hierarchy:

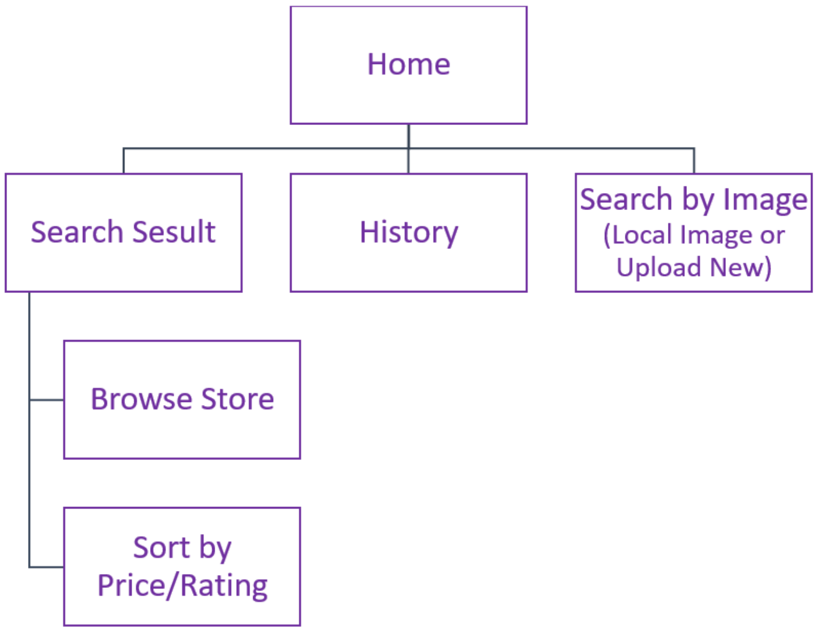


Figure 30 Interface Hierarchy

User Interface Description:

Home page:

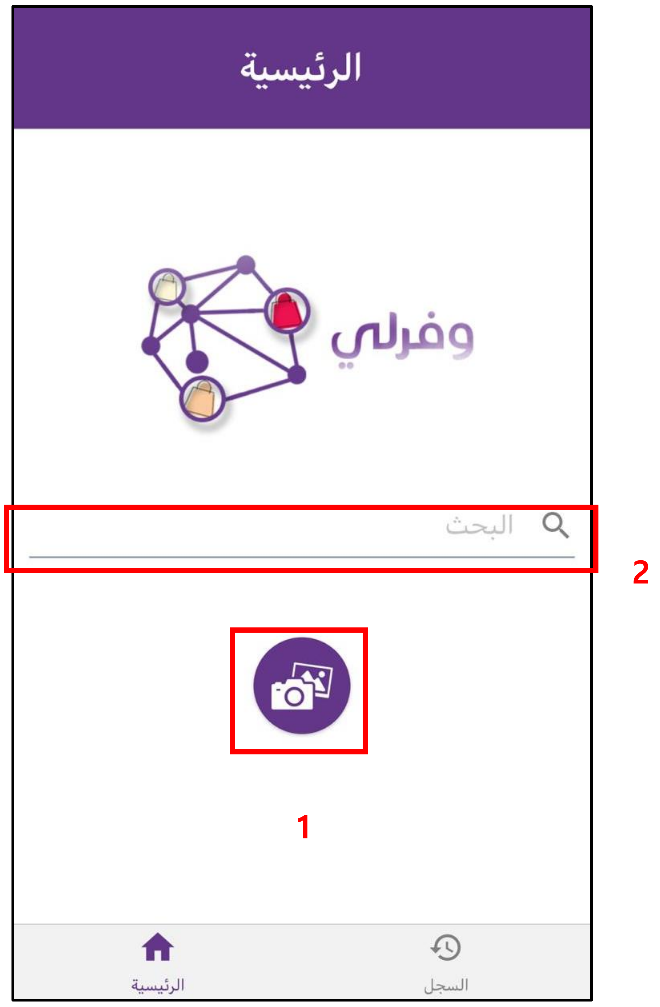
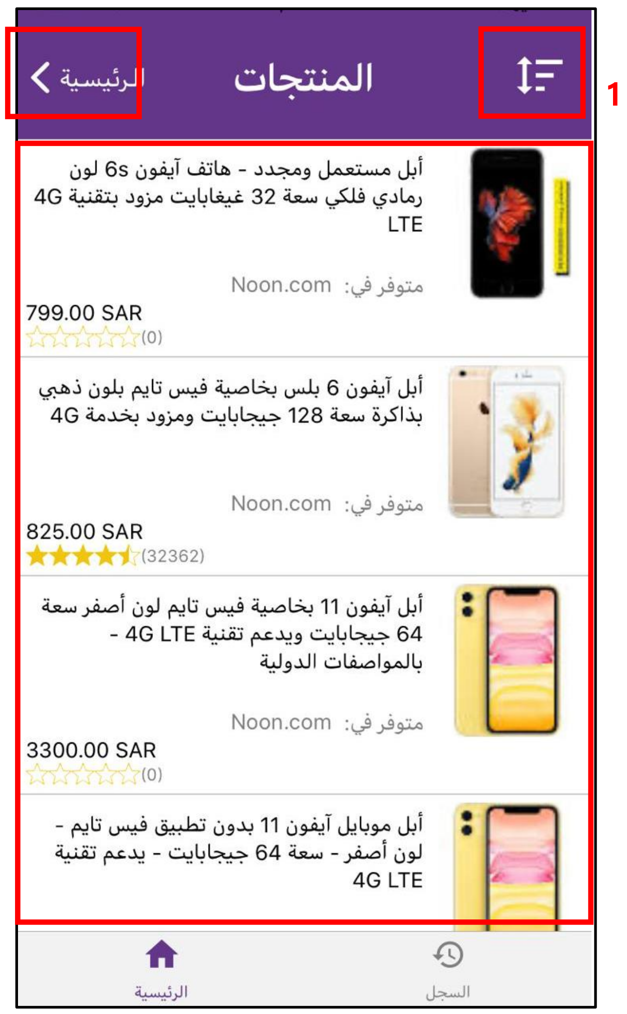


Figure 31 Home Page

Table 5: Home page Description

No.	Description
1	Buttons to search either by keywords or by image
2	Text field to enter keywords

Search result page:



أبل مستعمل ومجدد - هاتف آيفون 6s لون رمادي فلكي سعة 32 غيغابايت مزود بتقنية 4G LTE
متوفر في: Noon.com
799.00 SAR
☆☆☆☆☆(0)

أبل آيفون 6 بلس بخاصية فيس تايم بلون ذهبي بذاكرة سعة 128 جيجابايت ومزود بخدمة 4G
متوفر في: Noon.com
825.00 SAR
★★★★☆(32362)

أبل آيفون 11 بخاصية فيس تايم لون أصفر سعة 64 جيجابايت ويدعم تقنية 4G LTE - بالمواصفات الدولية
متوفر في: Noon.com
3300.00 SAR
☆☆☆☆☆(0)

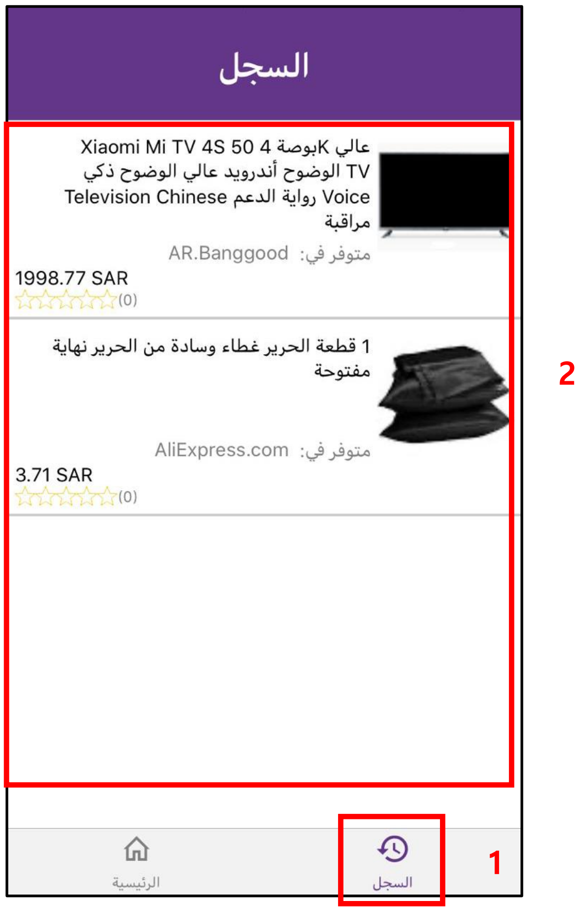
أبل موبايل آيفون 11 بدون تطبيق فيس تايم - لون أصفر - سعة 64 جيجابايت - يدعم تقنية 4G LTE

Figure 32 Search Result Page

Table 6: Search result page Description

No.	Description
1	Button to sort the results.
2	Section to display the products.
3	Button to go back to the home page.

History Page:



السجل
Xiaomi Mi TV 4S 50 4 عالي Kبوصة TV الوضوح أندرويد عالي الوضوح ذكي Voice رواية الدعم Television Chinese مراقبة
متوفر في: AR.Banggood
1998.77 SAR
(0)
1 قطعة الحرير غطاء وسادة من الحرير نهاية مفتوحة
متوفر في: AliExpress.com
3.71 SAR
(0)
2
الرئيسية
السجل
1

Figure 33: History Page

Table 7: History page Description

No.	**Description**
1	Button to view the products that were visited in the past.
2	Section to display the products that have been visited.

3.4.4. Data design

This section will cover the data design of Wafferly application, in fact Wafferly did not need to store data permenantly except in the Get Synonyms component. First when we collect the synonyms using Crowed-sourcing technique and save those synonyms into the database, another use for it is when the user entered keyword to search, the database is searched for the keyword if it found, then search for its synonyms, if the keyword not found it is inserted to the database.

ER Diagram

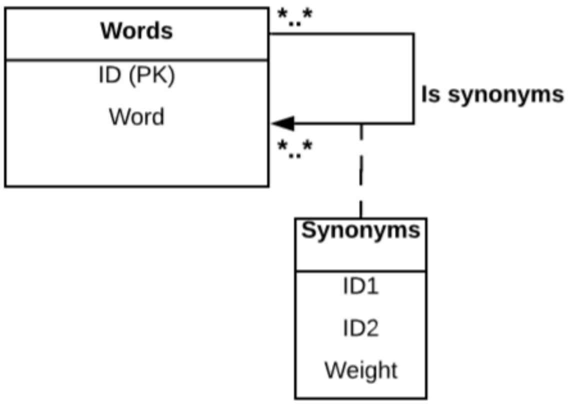


Figure 34: ER Diagram

RELATIONAL SCHEMA

Words(ID, word)
PRIMARY KEY:ID

Synonyms(ID1,ID2,Weight)
FOREIGN KEY: ID1,ID2

DATA DICTIONARY

Table 8: Data Dictionary

Entity Name	Attribute	Description	Data type	Length	Null	Multi Value	Default Value	Range	PK
Words	ID	Id for each word	INT	11	No	No	False	-	Yes
	word	Word text	varchar	40	No	No	False	-	No
Synonyms	ID1	Id for the first word	INT	11	No	No	False	-	No
	ID2	Id for the second word	INT	11	No	No	False	-	No
	Weight	Wight for IDs	INT	11	No	No	False	-	No

Table 9: Relationship Dictionary

Entity Name	Multiplicity	Relationship	Entity Name	Multiplicity
Words	*	Is Synonym	Words	*

Chapter 5 | System Implementation and Integration

This chapter will cover system implementation and integration of Wafferly application, it will mention the hardware and software tools of the application, implementation details, and integration of the system and what the issues that we faced and how we resolved them.

5.2 Hardware and Software Tools:

5.2.1 **Hardware**:

Table 10 shows the hardware needed to complete Wafferly application requirements.

Name	**Description**	**Used for**	**Price**
SaharaNet server	Company that provides web hosting	For uploading the application's back-end source code, for a centralized storage for all app users.	288 SAR

Table 10

Furthermore, Table 11 show the hardware uses for developing Wafferly application.

Device	**Quantity**	**Description**
Laptop	4	To write and develop all the requirement for the application
Samsung mobile	1	To compare the applications in literature review
iPhone	1	To install the similar applications and compare them in literature review. Also, to test the application

Table 11

5.2.2 **Software**:

Table 12 show the software needed for developing Wafferly application.

Software	**Developer**	**Version**	**Description**
WhatsApp	WhatsApp Inc.	2.20.31	For text communication among team members.
MS Office Word	Microsoft		To write project document.
Lucidchart	Lucid Software	2020	To draw use cases, sequence, class diagrams
Sublime Text	Sublime HQ	3.2.2	For developing the application.
react native	Facebook	0.61	For programming both iOS and Android with the same codebase.
Expo	Facebook	36.0.0	For programming iOS on a Windows machine.

Software	Developer	Version	Description
Node.js	Various	13.11.0	To be able to use Expo and React Native.
Skype	Skype Technologies	8.56.0.102	For live communication among team members.
Zoom	Cisco Systems	4.6	For live communication among team members.
Visual studio	Microsoft	1.42.1	For developing the application
Google drive	Google	2020	For sharing documents among team members.
Anaconda Navigator	Anaconda, Inc.	1.9.12	For programing Python code

Table 12

5.3 Implementation Details

As described in section 4.4.1. The view includes the interfaces implemented in Expo. The interfaces communicate with the controller which in turn prepares user input before sending it to the search function. User input can be prepared in two ways: If it’s an image, the controller will interface with the Google Vision API and retrieve the image’s extracted attributes; these attributes will then be used as search keywords moving forward.
The second possibility is that the user entered textual keywords, the controller will then send these keywords to the synonym’s finder, which will look for crowed-sourced synonyms that have enough weight. The initial keywords in addition to the newly found synonyms are then combined to form a new list of keywords which will then be used by the search function.
The search function searches for products using these keywords by interfacing with Google Shopping and will then retrieve a list of products. The view will then receive this list of products then shows them to the user sorted by price: cheapest to most expensive. The view will enable the user to sort the returned result by rating or price, also it will give the user the ability to browse any recently viewed products.

5.3.1 Front-End:

The front-end of Wafferly consists of six screens implemented using Expo. ‘React navigation’ library was used to navigate between screens. The first screen displayed to the user is the home screen which contains the input field and buttons to open the camera screen/album screen for the user to start the search. After the user starts searching either by an image of the product or keywords, the second screen will display the search results.

The search result screen sends the user’s input via HTTP request to the server and list the retrieved products. Then the user will be able to open the sort screen which will enable the user to sort the retrieved list either by price or product rating.
Any products the user tapped on will be saved in the local storage of the device then displayed in the history screen. The history screen is limited to save up to fifteen products. When the number of products exceeds fifteen the older once will be overwritten with the most recently visited products.

Challenges:

The main challenge with the front-end implementation was using Expo since we never dealt with it. Another challenge since we are using Expo, we must upload images to Firebase first then it can be sent to the Google Vision API. We have resolved this issue by using Firebase for temporary image upload. Firebase was the suitable solution since it easily integrates with the Google Vision API.

5.3.2 Back-End:

The back-end files are implemented using PHP. The main components of the back end are ‘Get Synonyms’, ‘Search’ and the ‘Extract Image Attributes’ functions.

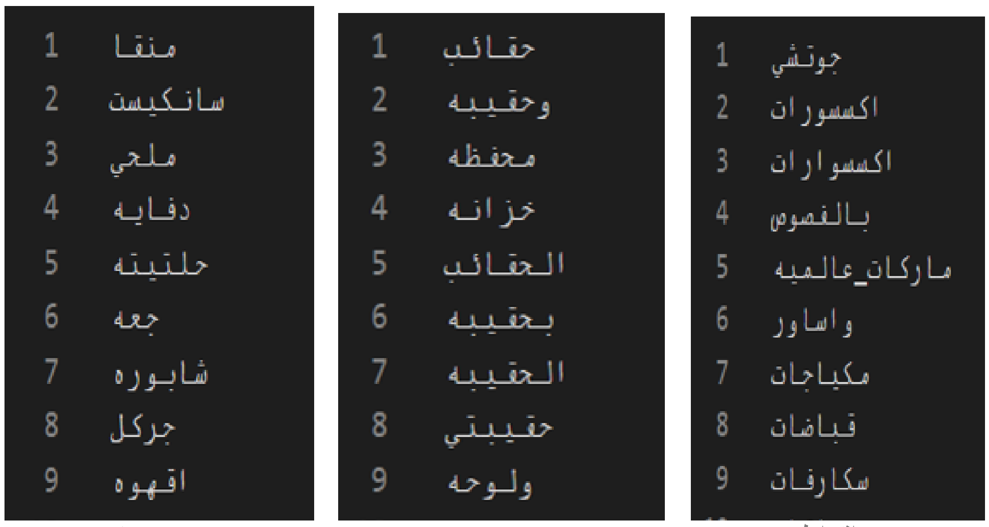
Get Synonyms:

The idea of the Get Synonyms function is to get as many synonyms as possible of the keyword that the user entered to expand the search result. The keyword should be in Arabic to get its synonyms. Synonyms are found by querying a crowed-sourced lexicon of Modern Standard Arabic words and their synonyms. the list of the synonyms is saved in a database table (as described in section Data design).
The synonyms were collected by first collecting a seed of keywords scraped from online stores including Souq and Noon and storing them in the database schema shown in section (4.4.4). Then, we built a dynamic questionnaire that randomly selects words from the database and asks users to enter their synonyms in their opinion. The code then checks if the entered synonyms are already associated with that word; if yes, increment the association weight by 1; if not, add that word as a synonym to the original word with initial association weight of 1. When an association weight reaches 3, these two words are considered synonyms by our system.

So, when the ‘Get Synonyms’ function receives the keyword entered by the user in the search function, its checks if the keyword is Arabic or not, if it is an Arabic keyword, it checks if the keyword is in the synonyms database or not: if yes, then searches for its synonyms (other words with association weight equal or above three). It then sends the keyword and its synonyms to the search function. However, if the keyword was not found in the database, it is inserted in the synonyms database. This will add the word to our future crowd-sourcing component (i.e. Users will be able to see this word in the dynamic form that collects synonyms).
Keyword cleaning: since the database have a static format, the ‘Get Synonyms’ function, after if it checks if the keyword is an Arabic, also unifies these characters and replace them as follows: ‘أ’, ‘إ’, ‘آ’ to ‘ا’, ‘ة’ to ‘ه’, ‘ئ’ to ‘ي’, ‘ؤ’ to ‘و’.

Challenges:
As mentioned, the idea of synonyms revolved around finding the largest number of synonyms for the product, thus expanding the field of search. Hence, we started to search for a model to provide this service, we found AraVec and WordNet, but we faced some problems with them, so we chose to crowd-source a lexicon for the synonyms.

First, we considered using the AraVec model. AraVec is an open source Word2Vec model written in Python. AraVec provides two models built using two different datasets: Twitter and Wikipedia. We used Twitter dataset which has two types: full_grams_cbow_100_twitter and full_grams_cbow_300_twitter. After we used this model on both types, we noticed that it brings up similar words, not synonyms. Figure 34 and Figure 35 show the output of AraVec, the inputs words were “غلاية” and “حقيبة” and “اقراط”, from left to right, Figure 34 shows the output full_grams_cbow_100_twitter dataset and Figure 35 shows the output of full_grams_cbow_300_twitter. These words do not fulfill our purpose since they will lower the accuracy of the search results.



```
1   منقا
2   سانكيست
3   ملحي
4   دفايه
5   حلتيته
6   جعه
7   شابوره
8   جركل
9   اقهوه
```

غلاية

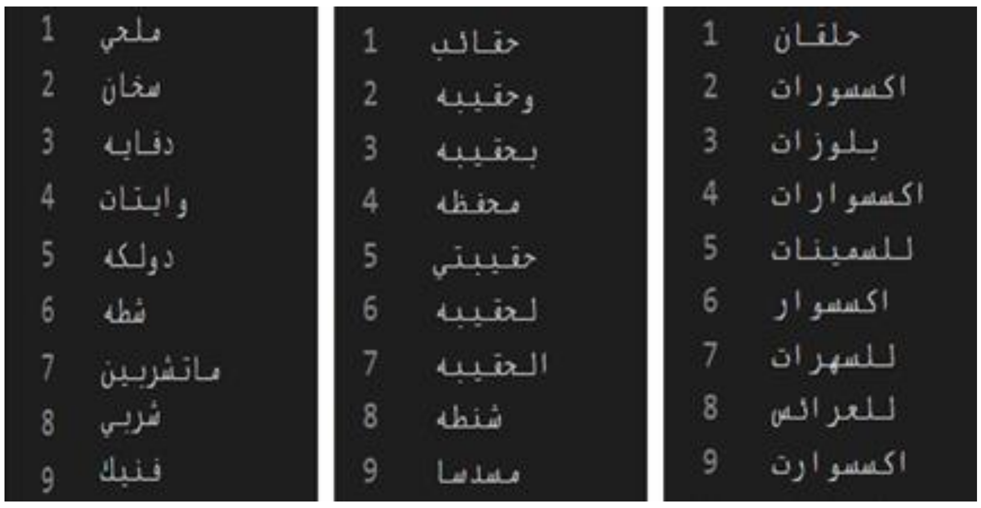
```
1   حقائب
2   وحقيبه
3   محفظه
4   خزانه
5   الحقائب
6   بحقيبه
7   الحقيبه
8   حقيبتي
9   ولوحه
```

حقيبة

```
1   جونشي
2   اكسسورات
3   اكسسوارات
4   بالفصوص
5   ماركات_عالميه
6   واساور
7   مكياجات
8   قباضات
9   سكارفات
```

اقراط

Figure 35: Output of full_grams_cbow_100_twitter



```
1   ملحي
2   سخان
3   دفايه
4   وايتان
5   دولكه
6   شطه
7   ماتشربين
8   شربي
9   فنبك
```

غلاية

```
1   حقائب
2   وحقيبه
3   بحقيبه
4   محفظه
5   حقيبتي
6   لحقيبه
7   الحقيبه
8   شنطه
9   مسدسا
```

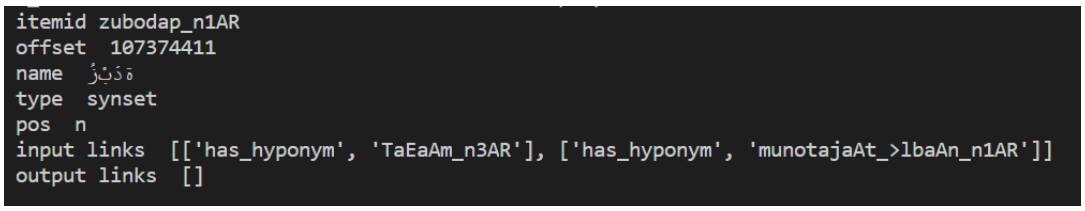
حقيبة

```
1   حلقان
2   اكسسورات
3   بلوزات
4   اكسسوارات
5   للسمينات
6   اكسسوار
7   للسهرات
8   للعرائس
9   اكسسوارت
```

اقراط

Figure 36 Output of full_grams_cbow_300_twitter

When AraVec did not help us, we investigated using WordNet. WordNet is a lexical database which contains large words with their synonyms and a simple definition of the word. But the problem is that its words are not colloquial Arabic, also its output is transliterated Arabic, it contains a specific number of words so we must restrict with just these words. As shown in Figure. As shown in Figure 36, the input was "زبدة" the output was "TaEaAm" and "mountajaAt_>labAn" and that in Arabic language "'طعام" and "منتجات ألبان".



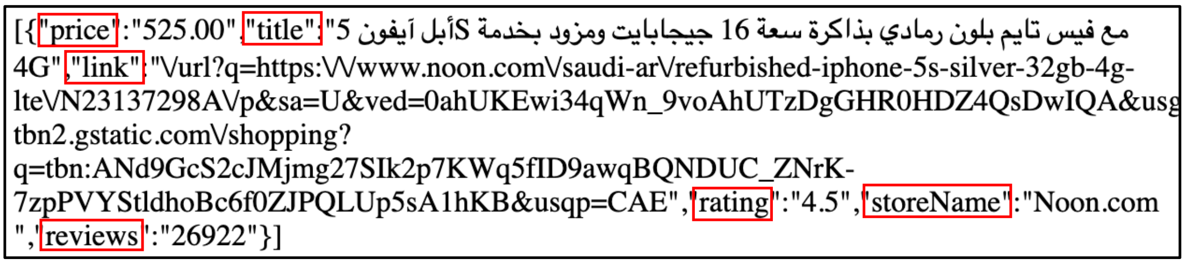
```
itemid zubodap_n1AR
offset  107374411
name  زُبْدَة
type  synset
pos  n
input links  [['has_hyponym', 'TaEaAm_n3AR'], ['has_hyponym', 'munotajaAt_>lbaAn_n1AR']]
output links  []
```

Figure 37: Output of WordNet

After we encountered these problems, we have opted to build a synonyms lexicon as mentioned before.

Search:

For the ‘Search’ function implementation, the function receives the keywords and their synonyms from the ‘Get Synonyms’ function. Then the ‘Search’ function makes an HTTP request with the synonyms included into the URL to Google Shopping using cURL in PHP. This returns a Google Shopping page to be scrapped. The ‘Search’ function then parses it using Simple HTML DOM library to convert the returned page to a DOM representation to get the products’ information.



[{"price":"525.00","title":"أبل آيفون 5S مع فيس تايم بلون رمادي بذاكرة سعة 16 جيجابايت ومزود بخدمة 4G","link":"\/url?q=https:\/\/www.noon.com\/saudi-ar\/refurbished-iphone-5s-silver-32gb-4g-lte\/N23137298A\/p&sa=U&ved=0ahUKEwi34qWn_9voAhUTzDgGHR0HDZ4QsDwIQA&usg
tbn2.gstatic.com\/shopping?
q=tbn:ANd9GcS2cJMjmg27SIk2p7KWq5fID9awqBQNDUC_ZNrK-
7zpPVYStldhoBc6f0ZJPQLUp5sA1hKB&usqp=CAE","rating":"4.5","storeName":"Noon.com
","reviews":"26922"}]

Figure 38 Returned Json

The function will then create a new product object and assigns its attributes with the retuned information. Each product object is appended to an array which is then encoded as a json object to be returned to the front end. Figure 38 shows the product after getting the information from Google Shopping page and converting it to JSON.

Challenges:

In the beginning we decided to use SerpAPI which is a website that provides a variety of APIs, including YouTube Search API, Yahoo Search and Google Search API. We used their Google Shopping API which is provided under the Google Search API to get the product information from Google Shopping since Google does not provide an API for Google Shopping. The API had issues with the returned json as it was not stable and kept changing the attribute values, so it was not reliable enough and did not meet our minimum requirements. Also, the images of the products were returned in a webp format which is not supported on iOS devices.

After facing these problems, we decided to create our own API (search function) using web scraping.

Another challenge was getting the delivery date and the brand of the product.

Each store has this information in different DOM paths, so we needed to customize the information retrieval based on the store. The delivery date was also formatted differently by different online stores.

We tested getting the delivery date for two stores, using custom DOM paths for each store and parsing the delivery date string, to test the efficacy of retrieving the information. When the function was tested, it took more than two minutes to get the search result which significantly reduced the performance of the application though it was only implemented for two stores. Because of that, we decided to drop sort by ‘Delivery date’ function because of the latency it causes.

For the brand information it was not included in the Google Shopping page also we could not get the information from the website because of that we could not implement the ‘Filter by brand’ function.

Search by image:

We used Google vision API, the API extracts many attributes from the image such as label detection, face detection, text detection, landmark detection, and logo detection. In order to extract the description and general labels of the image with their confidence, we use the label detection method of the Vision API. Additionally, we used logo detection within the search image since we need to search not only the description of the image but also any apparent logo. Finally, we used text detection which will extract any text in the image in several languages including Arabic and English.

To have the best result for the image processing we tested 50 images from different categories including electronics, Accessories, T-shirts, kid's stuff and dresses. The test was done by sending each image to the Google Vision API, after that we took the average of the manually verified and accepted images labels. The result after doing all the 50 images is confidence threshold equal to 91.122%.

Challenges:

When implementing the ‘Extract Image Attributes’ function, we faced an issue which is how to send the image from the front end to the back end. So, we used Firebase as a third party to upload the image to the web in order to be able to send it to the Google Vision API.

5.3.3 Code

In this section we will cover the most important code for Wafferly application and how it is work.

Function Name	Search
Description	Allow user to search for a product by entering the keywords and found synonyms of the product.

```php
include 'simple_html_dom.php';

$ProductList = array();

$page = 0;

// Loops over the search result pages

for ($i = 0; $i < 7; $i++) {

//the URL which will return the search result page from google shopping

    $base = 'https://www.google.com/search?q='.$keywords.
'&start='.$page.'&tbm=shop&biw=1422&bih=642&gl=sa&hl=ar-SA';

//HTTP request to google shopping page

    $curl = curl_init();

    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);

    curl_setopt($curl, CURLOPT_HEADER, false);

    curl_setopt($curl, CURLOPT_FOLLOWLOCATION, true);

    curl_setopt($curl, CURLOPT_URL, $base);

    curl_setopt($curl, CURLOPT_REFERER, $base);

    curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);

    $str = curl_exec($curl);

    curl_close($curl);

// create new instance of simple_html_dom

    $html = new simple_html_dom();


// convert the returned page to html dom

    $html->load($str);

//Loops through the results

    foreach ($html->find('.u30d4') as $container) {

// create new instance of product
```

```
        $product = new Product();

        if ($container->find('.rgHvZc', 0)) {

            $product->set_title($container->find('.rgHvZc',     0)->find('a',     0)-
>plaintext);

            $product->set_link($container->find('.rgHvZc',      0)->find('a',      0)-
>{'href'});

            $product->set_price(price($container->find('.dD8iuc   .HRLxBb     ',   0)-
>plaintext));

            $product->set_image($container->find('div.oR27Gd',0)->find('img',0)-
>{'src'});

            $product->set_rating(rating($container->find('div.m0amQc', 0)));

            $product->set_storeName(storeName($container));

            $product->set_reviews(reviews($container->find('div.d1BlKc', 0)));

            $Del=Delevery($product->get_storeName(),$product->get_link());

            $product->set_deliveryDate($Del[0]);

            $product->set_deliveryDateReact($Del[1]);

            $ProductList[] = $product;

        } //end if

    } //end foreach

//update page number

    $page += 20;

} // end for

//encode the products array to json format

$json = json_encode(SortByPrice($ProductList));

echo $json;
```

Function Name	Extract image attributes
Description	Allow user to search for a product by uploading/taking an image of the product.

```
require 'autoload.php';

// import the library of Google Vision API and Firebase library

use Kreait\Firebase\Factory;
use Kreait\Firebase\ServiceAccount;
use Google\Cloud\Storage\StorageClient;
use Google\Cloud\Vision\VisionClient;

// Authentication using keyfile to use API
$vision = new VisionClient(['keyFile' => json_decode(file_get_contents("ivory-signer-
267012-56b31bc7521e.json"), true)]);

// Receive image path in Firebase from front end
$json = json_decode(file_get_contents('php://input'),true);

// Authentication to use Firebase storage
$storage = new StorageClient([ 'projectId' => 'ivory-signer-267012']);

// Select the storage bucket
$bucket = $storage->bucket('ivory-signer-267012.appspot.com');

// select the image that have been upload to the Firebase
$object = $bucket->object($imageUri);
$familyPhotoResource = $object;
```

```
// Select the detections that will be used to image
$image = $vision->image($object,
[
'LABEL_DETECTION',
'LOGO_DETECTION',
"TEXT_DETECTION”]);

// Submit image to API for analysis
$result = $vision->annotate($image);

// Receive the detected labels, logos and text as array
$labels = $result->labels();
$logos = $result->logos();
$fullText = $result->text();

$syn = array();
$re="";$logo = "";$l=array();$logore="";$re2="";

if ($logos){

//Loop through logos array
foreach ($logos as $logo){

// Select the logos that have score from 90 and above
if(($logo->info()['score'] * 100)>=90){

// Select the name of logo
$re2= ucfirst($logo->info()['description']);
array_push($l,$re2);

}//end if
}// end loop

$logos2="";
$counter = 0;
foreach ($l as $logo2){
if( $counter == count( $l ) - 1) {
$logos2.=$logo2;
}else {
$logos2.=$logo2." ";}
$counter = $counter+1;
}

}// end if


if($labels){

//Loop through labels array
foreach ($labels as $label){

// Select the labels that have score from 90 and above
if(($label->info()['score'] * 100)>=90){

// Select the labels name
$re = ucfirst($label->info()['description']);
array_push($syn,$re);

}// end if
} // end loop
}// end if
$keywords2="";
$counter = 0;

// Add labels name to a single string
foreach ($syn as $keyword){
if( $counter == count( $syn ) - 1) {
$keywords2.=$keyword;
}else {
$keywords2.=$keyword." ";}
$counter = $counter+1;
}


$syn = array();

if($fullText){

// Loop through text array
foreach ($fullText as $text){
$re = ucfirst($text->info()['description']);
array_push($syn,$re);
}}


$text="";
$i=0;

// loop to select just the first two text
for ($i=0; $i<2 ;$i++){
if( $i == 1) {
$text.=$syn[$i];
}else {
$text.=$syn[$i]." ";}
}

$syn2 = explode(" ",$text);
```

```
$text="";

for ($i=0; $i<4 ;$i++){
if( $i == 3) {
$text.=$syn2[$i];
}else {
$text.=$syn2[$i]." ";}
}

echo Search($keywords2,$text,$logos2);


?>
```

5.4 System Integration

In this section we will explain how we integrate system components with each other. Also, we will cover any issues that we faced and how we resolved them.

In Wafferly the integration process was done in two parts. The first part was among the functions. The second was among the application and third-party libraries.

'Get synonyms' function is integrated with the database by sending the keyword to the database to check if this keyword has synonyms to be returned to the function or to add this keyword in the database in case of no synonyms were found. Then the 'Get synonyms' is integrated with 'Search' function by sending the synonyms of the keywords.

Another integration was between front end (Home Screen) and Firebase, the integration was based on sending an image in BLOB format to Firebase to upload the image then getting the ID of the uploaded image from Firebase storage. The Home Screen will send the ID of the image to 'Extract image attribute' function. The 'Extract image attribute' is integrated with Firebase by sending the ID of the image to retrieve it and use it with Google Vision API to get its attributes by Google Vison start the attributes extraction.

In the following subsection is a more thorough explanation to the components which we had some issues with.

5.4.1 Server:

We were searching for the most suitable server to host Wafferly. First the application was integrated with Google Cloud since we thought it was the most suitable one to use. But there was a problem which is the region of the server, the returned results were different from the one that we got when we tested the returned results locally, none of the stores which are available in the Middle East were returned in the result even when we changed the HTTP header it is still returned non-localized results.

Sahara net was used instead of Google Cloud; it is a hosting company with servers located in the Middle east which provides a more accurate result than before but it still not 100% accurate. Sahara net was used instead of Google Cloud; it is a hosting company with servers located in the Middle east which provides a more accurate result than before but it is still not 100% accurate.

5.4.2 Firebase:

Firebase is a Google platform to improve and grow mobile and web applications, it provides many services including a database storage solution. The application needs a storage solution in order to store the image temporarily while the application uses the Google Vision API. So, we decided to choose Firebase because it is more flexible than others and easy to use also, it’s strongly supported by Google.

Chapter 6 | SYSTEM TESTING

This chapter will cover different types of testing of Wafferly system; to ensure that the system is free of errors. Many types of tests were applied: unit, integration, regression, performance, stress, and user acceptance testing.

6.1 Unit Testing:

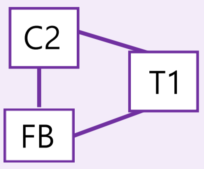
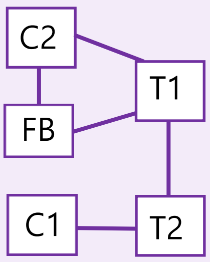
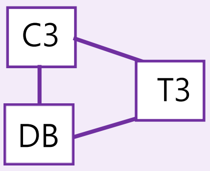
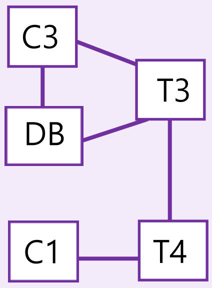
Table 13: Unit Testing

Users	Component Name	Component Description	Is working as expected
Customer	Component 1	Search	Pass
	Component 2	Search by image	Pass
	Component 3	Get synonyms	Pass
	Component 4	Sort by price	Pass
	Component 5	Sort by users rating	Pass

6.2 Integration and Regression Testing:

In Wafferly we did a bottom-up integration testing where we started testing the component at the lowest level first and when we assured that the result is as expected we integrated it with the component at the higher level. We showed the sequence of the integration testing in Table 14.

Table 14: Integration Testing

Test Sequence				
Test No.	Test 1	Test 2	Test 3	Test 4
Components	Integrate component 2 (C2) with Firebase (FB)	Integrate Test1(T1) with Component 1 (C1)	Integrate Component 3 (C3) with Database (DB)	Integrate Test3 (T3) with Component 1(C1)
Graphs				

Description	Test Extract image attribute with the Firebase	Test Search with Test1 components	Test get synonyms with the Database	Test Search with Test3 components
Results	Pass	Pass	Pass	Pass

6.3 Performance and Stress Testing:

Performance testing is used in Wafferly system to check the performance like scalability and speed. Moreover, stress testing is a subset of the performance testing and it checks the behavior of the system like stability. A test on the speed of getting the result was carried out as shown in Table 15.

Table 15: Performance and Stress Testing

Measures	**Search by keywords**	**Search by image**
Time	00:06:08 sec	00:022:52 sec
Number or products	36	40
Description	1-Read the user's keywords 2- get the synonyms of the keywords 3- start the search	1-Read the user's image 2- upload to Firebase 3- get the attributes of the image 3- start the search

6.4 User Acceptance Testing:

User acceptance testing used in Wafferly application as last step in software testing process, in this step the end-users of the software test it to make sure that the software handles the requirements, also to get feedback from the end-users to make them satisfied. The target users of Wafferly are the consumers of online shops, who have a variety of backgrounds and ages.

Table 16: User Acceptance Testing 1

Tester Name	Rand			
Task	**Number of errors**	**Time needed**	**User's feedback**	**Completion status**
Search	0	00:07:66 sec	Need to show button to start search	Completed
Search by image	0	00:09:51 sec	-	Completed
Sort by price	0	00:06:72 sec	-	Completed
Sort by users rating	0	00:04:07 sec	-	Completed

Table 17: User Acceptance Testing 2

Tester Name	Reema			
Task	**Number of errors**	**Time needed**	**User's feedback**	**Completion status**
Search	0	00:16:27 sec	-	Completed
Search by image	0	00:05:55 sec	-	Completed
Sort by price	0	00:04:30 sec	-	Completed
Sort by users rating	0	00:04:30 sec	-	Completed

Table 18: User Acceptance Testing 3

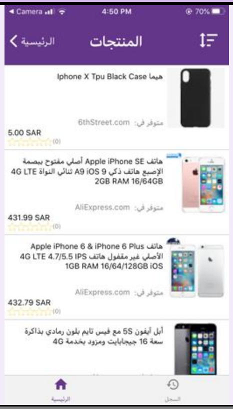
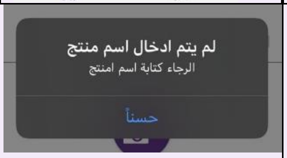
Tester Name	Mohammed			
Task	**Number of errors**	**Time needed**	**User's feedback**	**Completion status**
Search	0	00:05:60 sec	-	Completed
Search by image	0	00:06:65 sec	-	Completed
Sort by price	0	00:05:73 sec	-	Completed

Sort by users rating	0	00:04:05 sec	-	Completed

6.5 Test Cases:

All the test case pass (Get Synonyms and search) but for the search by image Some of the test cases fail since we are limited by the Google Vision API performance. For example, uploading bad resolution image in table 18 fail. Also, there were some input (keywords/images) that produced less accurate results than others.

Table 19: Search

Function	Search				
Description	This test cases shows different scenarios that a consumer may experience when trying to search for specific product.				
Test Case	**Description**	**Action**	**Expected Result**	**Actual Result**	**Result**
Enter keywords	The user enters keywords	The user enters keywords in the input filed and click search	The application will search using the entered keywords and redirect the user to the result page	 ◀ Camera 4:50 PM 70% الرئيسية المنتجات Iphone X Tpu Black Case هيما متوفر في: 6thStreet.com 5.00 SAR (0) هاتف Apple iPhone SE أصلي مفتوح ببصمة الإصبع هاتف ذكي iOS 9 A9 ثنائي النواة 4G LTE 2GB RAM 16/64GB متوفر في: AliExpress.com 431.99 SAR (0) هاتف Apple iPhone 6 & iPhone 6 Plus الأصلي غير مقفول هاتف 4G LTE 4.7/5.5 IPS 1GB RAM 16/64/128GB iOS متوفر في: AliExpress.com 432.79 SAR (0) أبل آيفون 5S مع فيس تايم بلون رمادي بذاكرة سعة 16 جيجابايت ومزود بخدمة 4G الرئيسية السجل	Pass
Leave the filed empty	The user presses the search button without writing any keywords		An alert message that prompt the consumer to enter keywords	 لم يتم ادخال اسم منتج الرجاء كتابة اسم امنتج حسناً	Pass

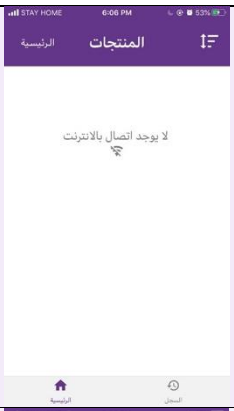
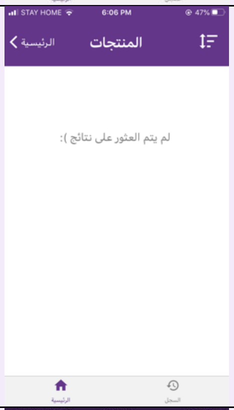
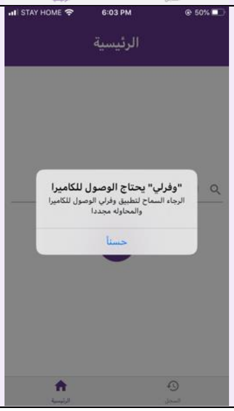
Search by keywords with no internet connection	The user tries to search by keyword without internet connection	The user enters keywords in the input filed and click search	The application will display a message telling the consumer that there is no internet	 لا يوجد اتصال بالانترنت	Pass
No result was found	The user search for product and no result was found	The user enters/upload keywords/image in the input filed and click search	The application will display a message telling the consumer that no result was found	 لم يتم العثور على نتائج):	Pass
No camera/album access	The user tries to take/select a picture of a product	The user clicks on the upload image button	An alert message that prompt the consumer to allow access to the camera/album	 "وفرلي" يحتاج الوصول للكاميرا الرجاء السماح لتطبيق وفرلي الوصول للكاميرا والمحاوله مجددا حسنا	Pass

Table 20: Get Synonyms

Function	Get Synonyms				
Description	Test the ability of get synonym function to fetch the synonym from the database				
Test Case	**Description**	**Action**	**Expected Result**	**Actual Result**	**Result**

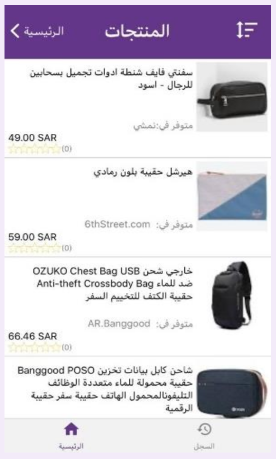
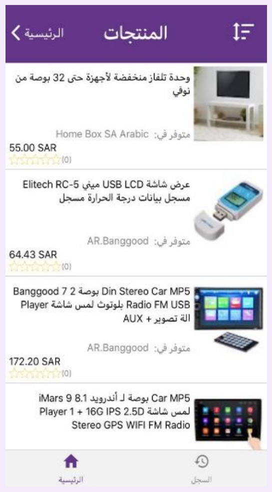
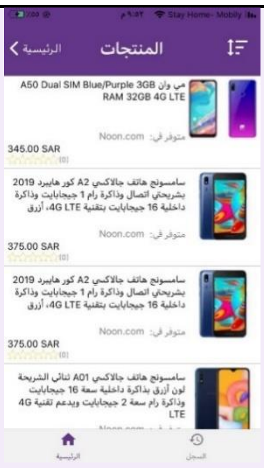
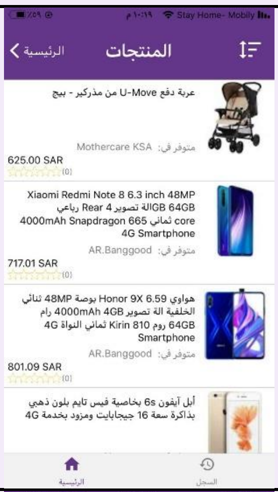
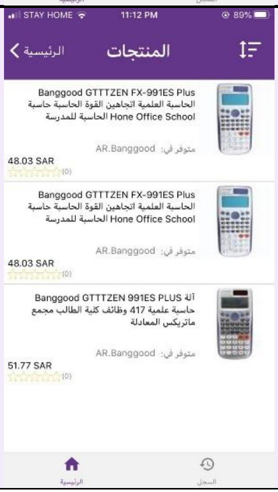
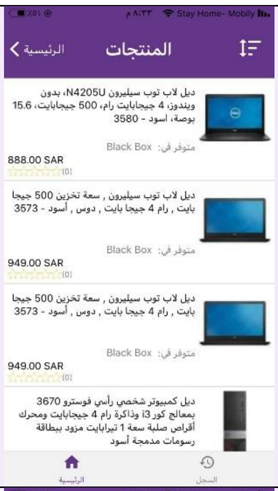
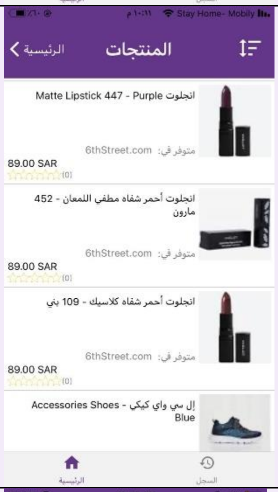
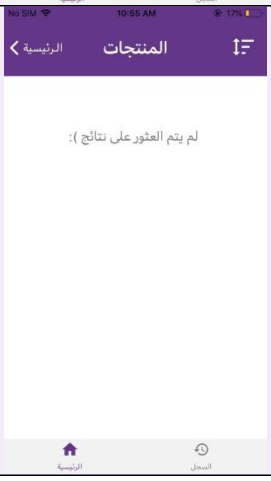
Enter an Arabic keyword That have synonym in database like “حقيبة”	The user enters an Arabic word that is in the synonym’s lexicon.	Enter the keyword	The search will cover the keyword and its synonyms: "حقيبة"and"شنطة"	 الرئيسية المنتجات سفنتي فايف شنطة ادوات تجميل بسحابين للرجال - اسود متوفر في:نمشي 49.00 SAR (0) هيرشل حقيبة بلون رمادي متوفر في: 6thStreet.com 59.00 SAR (0) OZUKO Chest Bag USB خارجي شحن Anti-theft Crossbody Bag ضد للماء حقيبة الكتف للتخييم السفر متوفر في: AR.Banggood 66.46 SAR (0) Banggood POSO شاحن كابل بيانات تخزين حقيبة محمولة للماء متعددة الوظائف التليفونالمحمول الهاتف حقيبة سفر حقيبة الرقمية الرئيسية السجل	Pass
Enter an Arabic keyword That don’t have synonyms in database (or not with sufficient weight) like “شاشة”	The user enters an Arabic word That is not in the synonyms lexicon.	Enter the keyword	The search will cover just the keyword that was entered.	 الرئيسية المنتجات وحدة تلفاز منخفضة لأجهزة حتى 32 بوصة من نوفي متوفر في: Home Box SA Arabic 55.00 SAR (0) Elitech RC-5 ميني USB LCD عرض شاشة مسجل بيانات درجة الحرارة مسجل متوفر في: AR.Banggood 64.43 SAR (0) Banggood 7 2 بوصة Din Stereo Car MP5 Player بلوتوث لمس شاشة Radio FM USB الة تصوير + AUX متوفر في: AR.Banggood 172.20 SAR (0) iMars 9 8.1 بوصة لـ أندرويد Car MP5 Player 1 + 16G IPS 2.5D لمس شاشة Stereo GPS WIFI FM Radio الرئيسية السجل	Pass

Table 21: Search by Image

Function	Search by image					
Description	Test the ability of search by image function to extract labels and search					
Test Case	**Description**	**Action**	**extracted labels**	**Expected Result**	**Actual Result**	**Result**

Upload product image	Upload in a Galaxy mobile	Press upload from gallery icon	**labels:** Mobile phone Gadget, Communication Device , Portable communications device **Text**: SAMSUNG	Galaxy mobile	 المنتجات الرئيسية مي وان A50 Dual SIM Blue/Purple 3GB RAM 32GB 4G LTE متوفر في: Noon.com 345.00 SAR (0) سامسونج هاتف جالاكسي A2 كور هايبرد 2019 بشريحتي اتصال وذاكرة رام 1 جيجابايت وذاكرة داخلية 16 جيجابايت بتقنية 4G LTE، أزرق متوفر في: Noon.com 375.00 SAR (0) سامسونج هاتف جالاكسي A2 كور هايبرد 2019 بشريحتي اتصال وذاكرة رام 1 جيجابايت وذاكرة داخلية 16 جيجابايت بتقنية 4G LTE، أزرق متوفر في: Noon.com 375.00 SAR (0) سامسونج هاتف جالاكسي A01 ثنائي الشريحة لون أزرق بذاكرة داخلية سعة 16 جيجابايت وذاكرة رام سعة 2 جيجابايت ويدعم تقنية 4G LTE الرئيسية السجل	Pass
Upload product image	Upload Baby product	Press upload from gallery icon	**labels:** Baby carriage product, White Black Baby Product **logo**: none **Text:** none	Baby stroller	 المنتجات الرئيسية عربة دفع U-Move من مذركير - بيج متوفر في: Mothercare KSA 625.00 SAR (0) Xiaomi Redmi Note 8 6.3 inch 48MP رباعي Rear 4 الة تصوير GB 64GB 4000mAh Snapdragon 665 ثماني core 4G Smartphone متوفر في: AR.Banggood 717.01 SAR (0) هواوي Honor 9X 6.59 بوصة 48MP ثنائي الخلفية الة تصوير 4GB 4000mAh رام 64GB روم Kirin 810 ثماني النواة 4G Smartphone متوفر في: AR.Banggood 801.09 SAR (0) أبل آيفون 6s بخاصية فيس تايم بلون ذهبي بذاكرة سعة 16 جيجابايت ومزود بخدمة 4G الرئيسية السجل	Pass (with low accuracy)
Take an image of the product	Take calculator image	Press camera from gallery icon	**labels:** Calculator Office equipment **Text**: CASIO fx-991ES PLUS NATURAL-UPAM TWO	Calculator	 STAY HOME 11:12 PM 89% المنتجات الرئيسية Banggood GTTTZEN FX-991ES Plus الحاسبة العلمية اتجاهين القوة الحاسبة حاسبة Hone Office School الحاسبة للمدرسة متوفر في: AR.Banggood 48.03 SAR (0) Banggood GTTTZEN FX-991ES Plus الحاسبة العلمية اتجاهين القوة الحاسبة حاسبة Hone Office School الحاسبة للمدرسة متوفر في: AR.Banggood 48.03 SAR (0) Banggood GTTTZEN 991ES PLUS آلة حاسبة علمية 417 وظائف كلية الطالب مجمع ماتريكس المعادلة متوفر في: AR.Banggood 51.77 SAR (0) الرئيسية السجل	Pass

Take an image of the product	Take laptop image	Press camera from gallery icon	**Label:** laptop, laptop part, netbook **Text**: Dell **Logo**: Dell	Dell laptop	 المنتجات الرئيسية ديل لاب توب سيليرون N4205U، بدون ويندوز، 4 جيجابايت رام، 500 جيجابايت، 15.6 بوصة، اسود - 3580 متوفر في: Black Box 888.00 SAR (0) ديل لاب توب سيليرون , سعة تخزين 500 جيجا بايت , رام 4 جيجا بايت , دوس , أسود - 3573 متوفر في: Black Box 949.00 SAR (0) ديل لاب توب سيليرون , سعة تخزين 500 جيجا بايت , رام 4 جيجا بايت , دوس , أسود - 3573 متوفر في: Black Box 949.00 SAR (0) ديل كمبيوتر شخصي رأسي فوسترو 3670 بمعالج كور i3 وذاكرة رام 4 جيجابايت ومحرك أقراص صلبة سعة 1 تيرابايت مزود ببطاقة رسومات مدمجة أسود الرئيسية السجل	Pass
Take inaccurate image	Take bad resolution image	Press camera from gallery icon	No label	No result	 المنتجات الرئيسية Matte Lipstick 447 - Purple انجلوت متوفر في: 6thStreet.com 89.00 SAR (0) انجلوت أحمر شفاه مطفي اللمعان - 452 مارون متوفر في: 6thStreet.com 89.00 SAR (0) انجلوت أحمر شفاه كلاسيك - 109 بني متوفر في: 6thStreet.com 89.00 SAR (0) Accessories Shoes - إل سي واي كيكي Blue الرئيسية السجل	Pass (with low accuracy)
Upload a black image	Upload black image	Press upload from gallery icon	**Label:** black text **Logo:** none **Text**: none	No result	 المنتجات الرئيسية لم يتم العثور على نتائج): الرئيسية السجل	Pass

Chapter 7| CONCLUSION

7. Conclusion

7.5.Summary

In conclusion, Wafferly helps consumers find out whether the product they want to buy is available at a lower price by searching various online shops.
As mentioned in the first chapter the number of the online stores has been increasing thus making it difficult for consumers to know about the products in each online store and having an overview of the price rates.
The background section provided a brief introduction of some of the technologies that were used or considered to build the application. Furthermore, the application domain was illustrated more deeply to have a clear picture of the price-comparison websites domain. This was followed by an explanation of the APIs the application uses. Lastly the storage that the application uses was described.
The literature review chapter outlined some similar applications to review the features that are typically expected of services in this application domain. This overview provided a clear view and set of goals to meet the consumers' expectations in this domain.
For the system analysis and design chapter, it included the application requirements and illustration of the main functions of the system in the form of diagrams. The design of the interfaces was built with user experience (UX) in mind.
For the System Implementation and Integration chapter, it covers the hardware and software tools used in the development of Wafferly application and implementation details including back-end and front-end with their challenges. Also, in this chapter, we mention the most important code for Wafferly. lastly, we cover the system integration.
The final chapter is system testing which covers different types of testing such as unit testing integration and regression testing performance and stress testing user acceptance testing and test cases.
We hope Wafferly will achieve its goals and fulfill the consumer needs and be one of the popular in the Middle East applications that provide an essential service to the community by aiding consumers to take more well-informed spending decisions.

7.6.Local impact

Wafferly aims to help the Middle East consumers in saving their money by providing a complete picture of the price rates of certain products provided by online stores. It compares the online stores that are in the Middle East. Through providing this service, the application seeks to support the saving culture of Middle East-consumers. The application also aims to provide an easy comparison process by providing an easy and understandable search interface.

7.7.System limitation and future work

The main limitation of Wafferly is the result of the search which is caused by being limited to what Google Shopping will return. The scope of the search inWafferly also is limited on the stores which are available in Google Shopping. The interface language Wafferly supports is Arabic only. In the future we intend to work on the limitations of Wafferly by providing our own API for the sellers as what Google Shopping did, to help us get more information about the products and avoid the overhead of scraping. We will also broaden our scope by supporting different languages for the interface and by covering more stores, including the stores on Instagram.

References

[1] M. O. COMMERCE, "MINISTRY OF COMMERCE STRATEGIC PLAN 2019 - 2021," [Online]. Available: https://mci.gov.sa/ar/About/sp/Pages/default.aspx. [Accessed 16 3 2020].

[2] T. N. Y. Times, "Forever 21 Bankruptcy Signals a Shift in Consumer Tastes," [Online]. Available: https://www.nytimes.com/2019/09/29/business/forever-21-bankruptcy.html. [Accessed 24 10 2019].

[3] ه. الاتصالات, "ارتفاع نسبة التسوق عبر الإنترنت في المملكة إلى 49.9% بنهاية 2018," وزارة الاتصالات وتقنية المعلومات, 23 10 2019. [Online]. Available: https://www.mcit.gov.sa/ar/media-center/news/148549. [Accessed 23 10 2019].

[4] G. Cloud, "Detect image properties | Cloud Vision API Documentation," 17 10 2019. [Online]. Available: https://cloud.google.com/vision/docs/detecting-properties.

[5] R. Native, "A framework for building native apps using React," 2 12 2019. [Online]. Available: https://facebook.github.io/react-native/ Accessed: 2019-12-02.

[6] G. Cloud, "Vision AI | Derive Image Insights via ML | Cloud Vision API | Google Cloud," [Online]. Available: https://cloud.google.com/vision/#industry-leading-accuracy-for-image-understanding. [Accessed 17 10 2019].

[7] "Digital comparison tools market study Final report," The National Archives, London, 2017.

[8] G. Cloud, "Vision AI | Derive Image Insights via ML | Cloud Vision API," 17 10 2019. [Online]. Available: https://cloud.google.com/vision/#industry-leading-accuracy-for-image-understanding.

[9] B. M. Magusara, "How Comparison Websites Impact Businesses," 3 4 2018. [Online]. Available: https://www.business.com/articles/comparison-websites-business-impact/. [Accessed 17 10 2019].

[10] G. Cloud, "Detect Labels | Cloud Vision API Documentation | Google Cloud," [Online]. Available: https://cloud.google.com/vision/docs/labels. [Accessed 17 10 2019].

[11] G. Cloud, "Detect logos | Cloud Vision API Documentation |," [Online]. Available: https://cloud.google.com/vision/docs/detecting-logos. [Accessed 17 10 2019].

[12] g. cloud, "Detect text in images," [Online]. Available: https://cloud.google.com/vision/docs/ocr.

[13] "React – A JavaScript library for building user interfaces," [Online]. Available: https://reactjs.org/. [Accessed 2 12 2019].

[14] E. Documentation, "What is Expo?," [Online]. Available: https://docs.expo.io/versions/v35.0.0/. [Accessed 2 12 2019].

[15] datafeedwatch, "What is Google Shopping?," [Online]. Available: https://www.datafeedwatch.com/academy/google-shopping. [Accessed 2020].

[16] D. S.-. F. Sukkar, "Arabic Sentences Classification via Deep Learning," *International Journal of Computer Applications,* p. 3, 2018.

[17] w. net, "what is wordnet," [Online]. Available: https://wordnet.princeton.edu/. [Accessed 2020].

[18] M. HARGRAVE, "Crowdsourcing," investopedia, 8 Jul 2019. [Online]. Available: https://www.investopedia.com/terms/c/crowdsourcing.asp. [Accessed 2020].

[19] G. play, 11 3 2020. [Online]. Available: https://play.google.com/store/apps/details?id=se.prisjakt.pricespy&hl=ar.

[20] G. play, 11 3 2020. [Online]. Available: https://play.google.com/store/apps/details?id=de.idealo.android&hl=ar.

[21] G. play, 11 3 2020. [Online]. Available: https://play.google.com/store/apps/details?id=uk.co.scrooge&hl=en_US.

[22] G. play, 11 3 2020. [Online]. Available: https://play.google.com/store/apps/details?id=com.naddad.pricena&hl=en_US.

[23] G. play. [Online]. Available: https://play.google.com/store/apps/details?id=com.ksaprice.

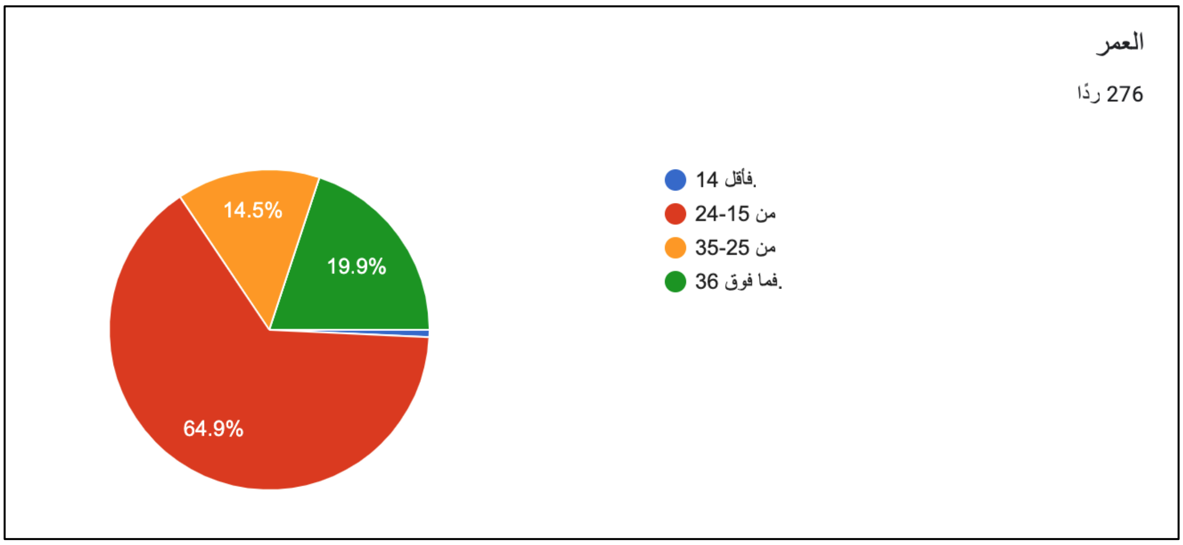
[24] G. play, 11 3 2020. [Online]. Available: https://play.google.com/store/apps/details?id=com.icloudit.kanbkam&hl=ar.

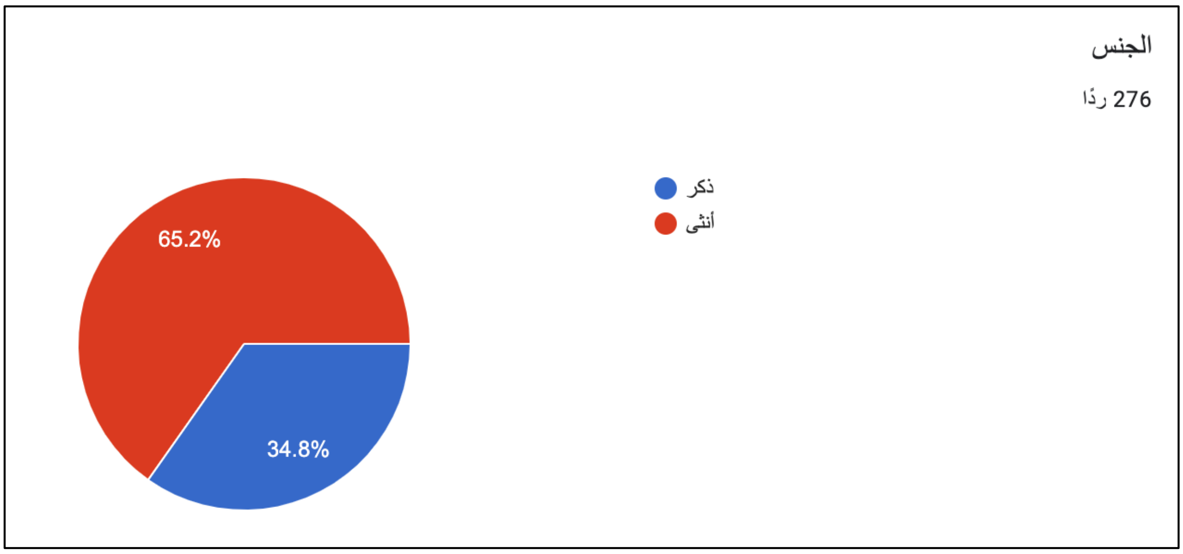
[25] G. play, 11 3 2020. [Online]. Available: https://play.google.com/store/apps/details?id=com.flyingelephantlab.yaoota&hl=en_US.

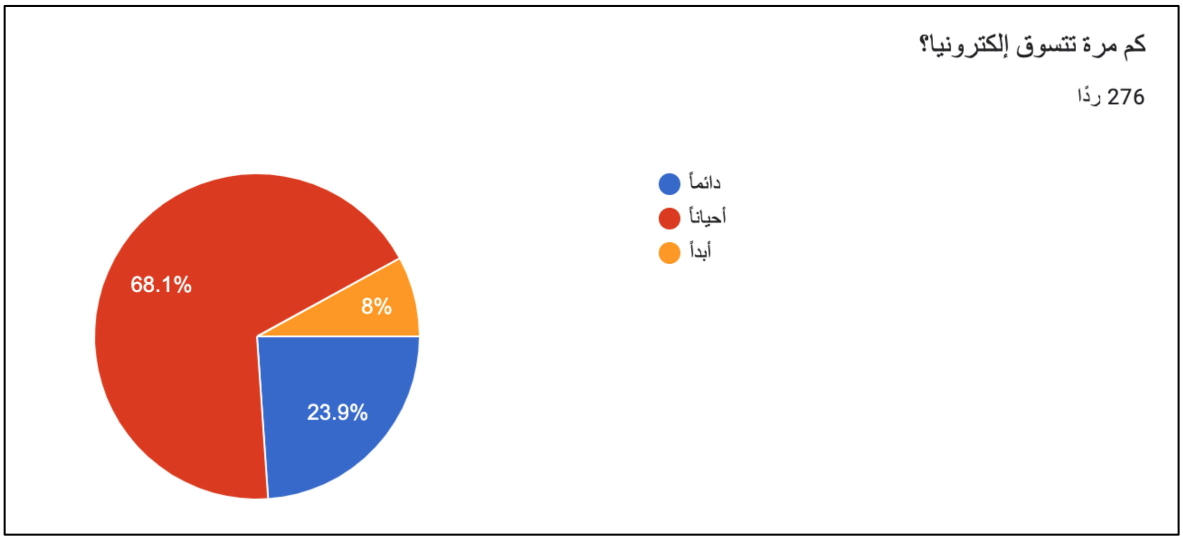
[26] 1202Performance, "How to write Performance Requirements with Exampl," [Online]. Available: http://www.1202performance.com/atricles/how-to-write-performance-requirements-with-example/. [Accessed 03 12 2019].

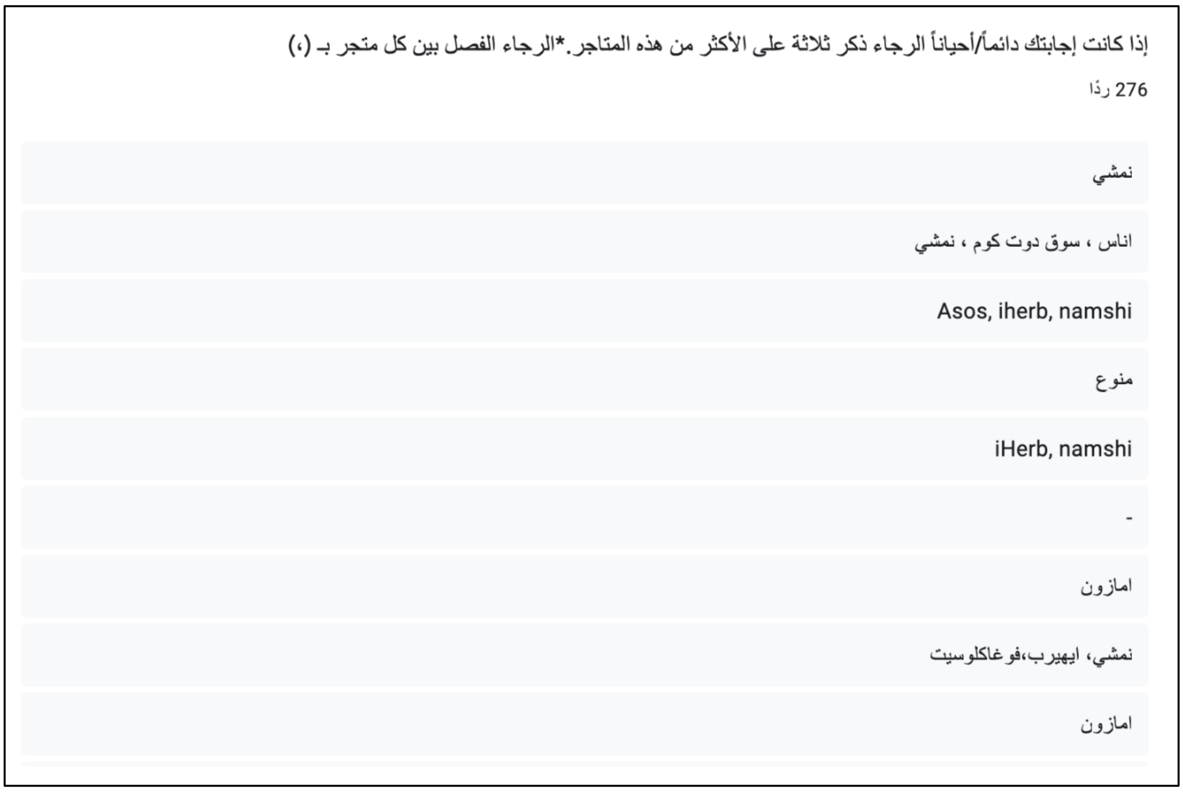
APPENDIX

Appendix A - Questionnaire









إذا كانت إجابتك دائماً/أحياناً الرجاء ذكر ثلاثة على الأكثر من هذه المتاجر.*الرجاء الفصل بين كل متجر بـ (،)

276 ردًا

نمشي

اناس ، سوق دوت كوم ، نمشي

Asos, iherb, namshi

منوع

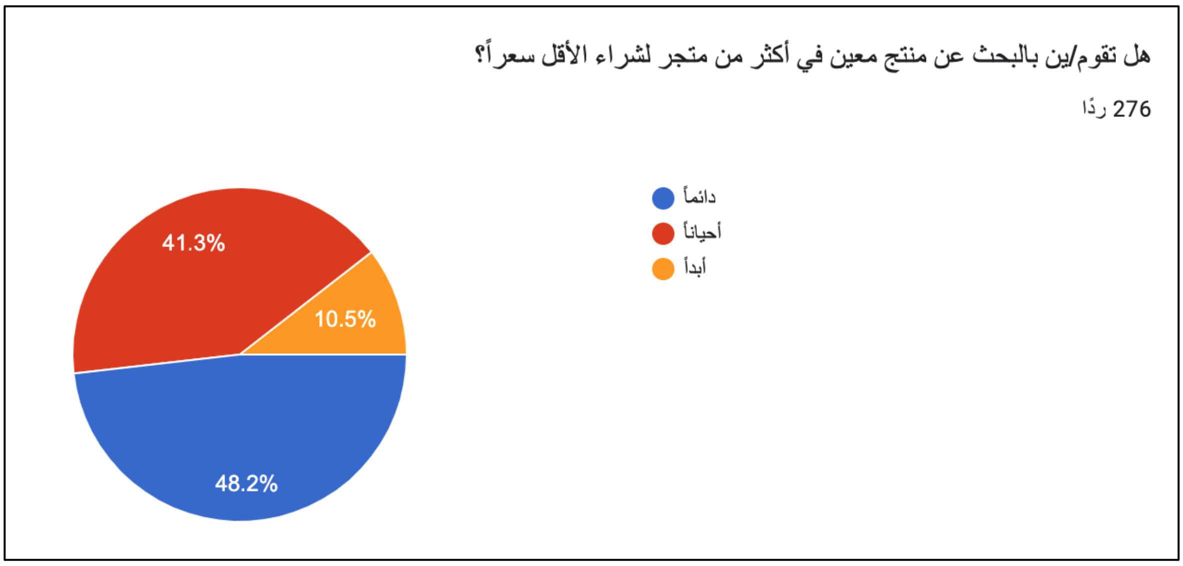
iHerb, namshi

-

امازون

نمشي، ايهيرب،فوغاكلوسيت

امازون

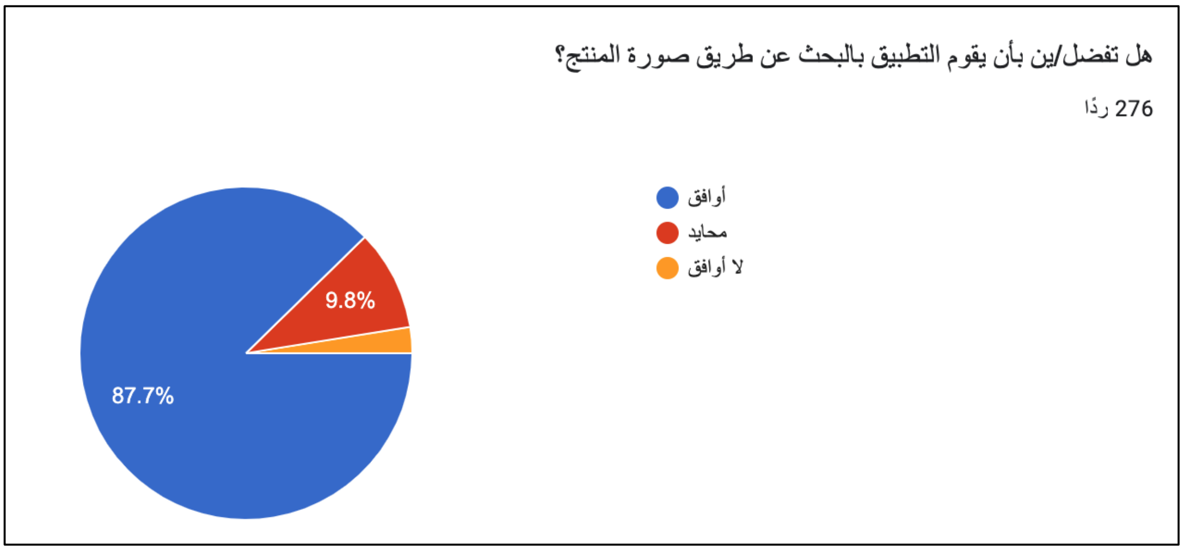


هل تقوم/ين بالبحث عن منتج معين في أكثر من متجر لشراء الأقل سعراً؟

276 ردًا

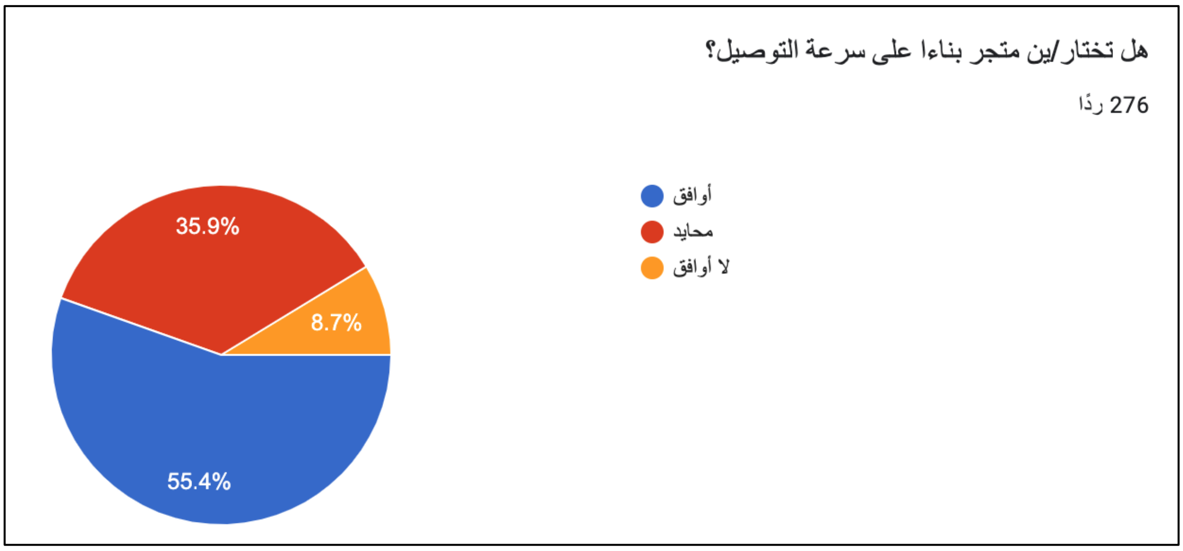
هل تفضل/ين بأن يقوم التطبيق بالبحث عن طريق صورة المنتج؟

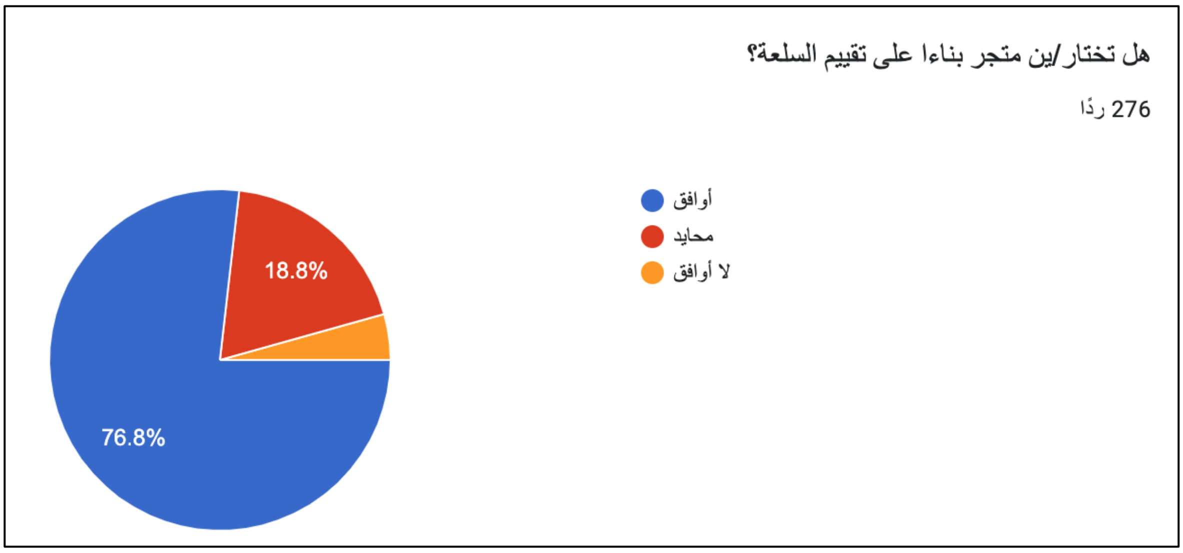
276 ردًا

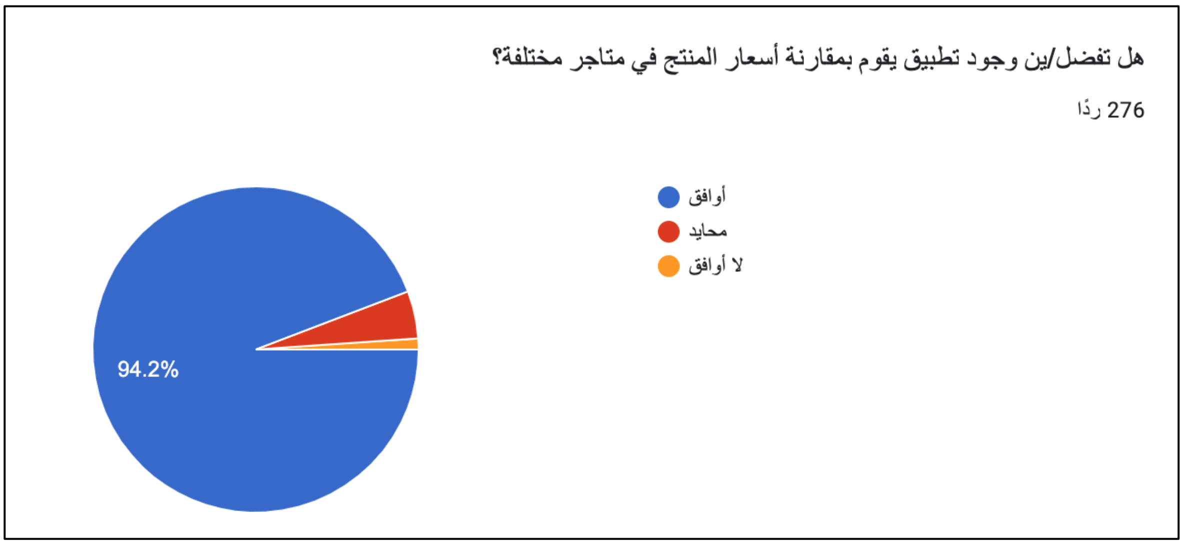


هل تختار/ين متجر بناءا على سرعة التوصيل؟

276 ردًا



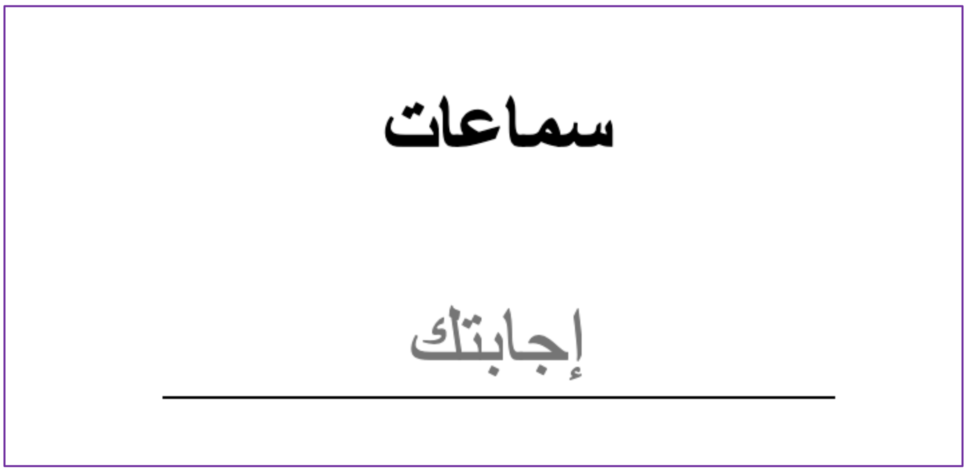




Survey of Wafferly

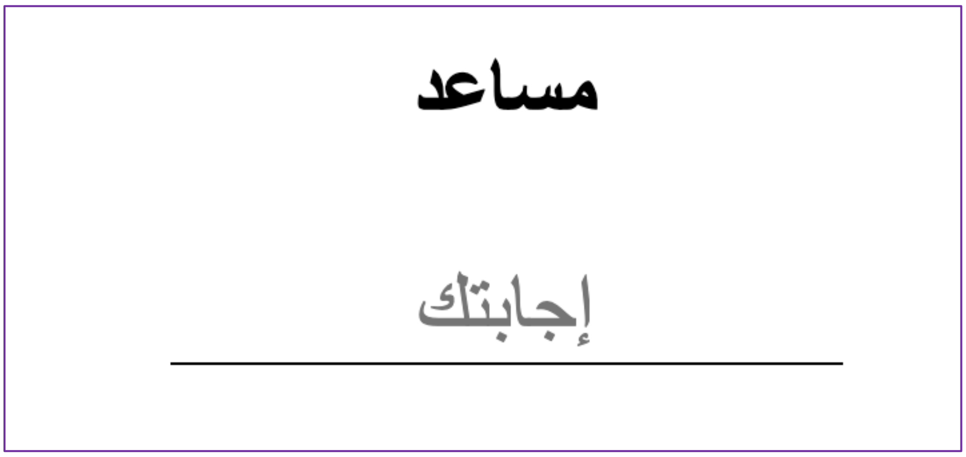
https://forms.gle/N6L1QDUWKz5H9QCQ6

Appendix B – Crowdsourcing form



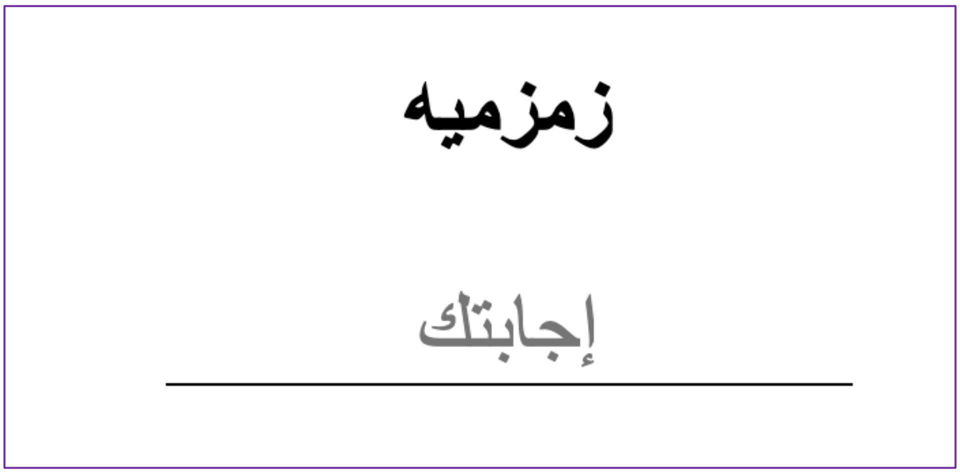
سماعات

إجابتك



مساعد

إجابتك



زمزميه

إجابتك

عدسات

إجابتك

____

طقم

إجابتك

____

Survey of get synonym:

http://wafferlyksu.com/Form.php