



People 's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University of Shaheed Hama Lakhdar-ElOued-
Faculty of Exact Sciences



Project End Of Study Presented for the Diploma of

ACADEMIC LICENSE

Domain: Mathematics and computer science

Department : Computer Science

Specialty: Information Systems

Theme:

Design and implementation of a price comparator under Android

Presented By:

Soltana Tehani

Baha ELhadda

Cheniba Khadidja

Proposed and followed by: Mr. Medileh Saci

Discussed in / /2018

In front of the jury :

Mr.....

MCA President

Mr.

MAA Rapporteur

University year :2017/2018

Table of Contents

<i>General Introduction:</i>	1
------------------------------------	---

Chapter 01 : Project Presentation

1	Introduction:.....	3
2	Problematic:.....	3
3	General description of our application:.....	3
4	Purpose of the application:.....	3
5	Software Development Life Cycle Models and Methodologies:..	4
5.1	Types of Software Development Life Cycle:.....	4
5.2	Methodology and formats adopted:.....	5
5.2.1	Waterfall Model:.....	5
6	Conclusion:.....	6

Chapter 02: Analysis and specification of requirements

1	Introduction:.....	8
2	Search engine:.....	8
2.1	Definition:.....	8
2.2	Why people use search engine:	8
2.3	The purpose of a search engine:.....	8
2.4	Types of Search Engines:.....	9
3	Comparison shopping engines (CSEs):.....	10

3.1	What are comparison shopping engines (CSEs)?	10
3.2	How does a comparison shopping engines (CSEs) work?	10
3.2.1	Getting data from merchant's feeds:	11
3.2.2	Crawling:	11
3.2.3	API integration:	12
4	API (Application Programming Interface):	12
4.1	What is an API? (Application Programming Interface):	12
4.2	What Do APIs Do?	13
4.3	Common Methods of API:	13
4.3.1	Authentication and Authorization:	13
4.3.2	Common Methods of API Authentication:	13
5	Application System Modeling:	14
5.1	What is System Modeling:	14
5.1.1	What is UML:	14
5.1.2	UML diagram types:	15
5.1.3	Advantages of UML :	15
6	Mobile Applications :	16
6.1	Mobile Device :	16
6.1.1	What Is a Mobile Device?	16
6.1.2	Mobile Device Types:	17
6.2	Mobile Applications:	17
6.2.1	What is Mobile Applications?	17

6.2.2	History of Mobile Applications:.....	17
6.2.3	Applications mobiles Types:.....	18
6.3	Integrated Development Environment (IDE):.....	18
6.3.1	What is (IDE)?.....	18
6.3.2	History of (IDEs):.....	19
6.3.3	Benefits of (IDEs):.....	19
6.3.4	Different types of (IDEs):.....	20
6.4	Android Operating System:.....	21
6.4.1	What is Android:.....	21
6.4.2	History of Android:.....	22
6.4.3	Why does Android look different on each phone?.....	22
6.4.4	Android versions , code name and API:.....	22
6.4.5	What are the advantages of Android?.....	23
7	Conclusion:.....	24

Chapter 03: Conception and Modeling

1	Introduction:.....	26
2	The application architecture:.....	26
3	System Modeling:.....	27
3.1	Use Case diagram:.....	27
3.2	Sequence Diagram:.....	27
3.3	Class diagram:.....	29

3.3.1	Description of the class model:.....	29
3.3.2	Transition to relational model:.....	29
3.3.3	Class diagram modeling:.....	30
4	Conclusion:.....	30

Chapter 04: Implementation

1	Introduction:.....	33
2	Platforms and Utilities:.....	33
2.1	Hardware:.....	33
2.2	Software:.....	34
3	Presentation of the technologies used:.....	35
3.1	XML:.....	35
3.2	JSON:.....	35
3.3	JAVA:.....	35
3.4	SQL:.....	36
3.5	Software developer's kit (SDK):.....	36
3.6	Java Development Kit (JDK):.....	36
3.7	Retrofit:.....	36
4	Presentation of the application:.....	37
5	Conclusion :.....	46
	<i>General Conclusion</i> :.....	47
	<i>References</i> :.....	48

Figures list:

Figure 1: Software Development Life Cycle.....	4
Figure 2: Waterfall Model.....	5
Figure 3 : Android versions , code name and AP.....	23
Figure 4: Application Architecture	26
Figure 5:Use Case diagram for the visitors of the application.....	27
Figure 6:Sequence Diagram for the application system.....	28
Figure 17: <i>Database schema</i>	30
Figure 7: Start page.....	37
Figure 8: Home page.....	38
Figure 9:Product Details page	39
Figure 10: Wbsite page	40
Figure 11:search result page.....	41
Figure 12: lowest price page	42
Figure 13: Highest price page	43
Figure 14: Home page (horizontal)	44
Figure 15: product details 1 (horizontal).....	45
Figure 16: product details 2 (horizontal).....	46

Tables list:

Table 1: Application Architecture steps.	26
Table 2:Description of the class model	29
Table 3:Transition to relational model	30
Table 6:Development Machine specification.....	33
Table 7:Used Application.....	34

Abstract

Due to the multiplicity of specialized sites in e-commerce domain and shopping online and as an attempt to keep up with the developments in it. We developed an Android application to facilitate the remote buying process. Which is kind of a search engines that searches for products based on the user's desire then display them on organized and arranged form according to the price on one page without effort

Keywords: search engine. Compare prices. Android.

ملخص

نظرا لكثرة المواقع المختصة في مجال التجارة الالكترونية والتسوق عبر الانترنت ومحاولة لمواكبة التطورات الحاصلة فيه قمنا بتطوير تطبيق اندرويد يقوم بتسهيل عملية الشراء عن بعد حيث انه عبارة عن محرك بحث مصغر يعمل على البحث عن المنتجات استنادا لرغبة المستخدم وعرضها له منظمة ومرتبة حسب السعر في صفحة واحدة بدون جهد او عناء
الكلمات المفتاحية: محرك بحث . مقارنة الاسعار . اندرويد.

Résumé

En raison de la multiplicité des sites spécialisés dans le domaine du commerce électronique et des achats en ligne et comme une tentative de suivre les développements dans ce domaine. Nous avons développé une application Android pour faciliter le processus d'achat à distance. ce qui est une sorte de moteur de recherche qui recherche des produits en fonction du désir de l'utilisateur, puis les affiche sous forme organisée et arrangée en fonction du prix sur une page sans effort

Mots-clés: moteur de recherche. Comparer les prix. Android.

General Introduction

What we see in this days from the evolution in the information and communications technology has led to the development of international economy .This evolution led to the emergence of an important aspect of well-being , that known as an electronic commerce, which has greatly facilitated online shopping. But by the passing of the time the number of the users in this domain (sellers and buyers) has significantly increased which has made it difficult for the shoppers to pick the product that they want it, and the reason why is because there are many shopping sites and different product prices.

Therefore, in order abreast this developments and to solve this problem which we just presented we came with this project, which is an application that facilitates online shopping, where we presented in the form of a study, divided it into four chapters. Each one of them has dedicated to a specific step. The first chapter is " Project Presentation" when we talked about our app by giving the general idea about it and in the second chapter "Analysis and specification of requirements" we defined and analyzed the most important elements to this project then there's the third chapter "conception and modeling" where we dedicated it for the step Which comes before development step which is concerned in system modeling.

In the end we come up with the final chapter "Implementation" which is the most important chapter because it contains the result of our study.

Chapter 01

Project Presentation

Chapter 01 : Project Presentation

1 Introduction:

This chapter will be dedicated to defining our application .By giving the general description about it , mentioning its purpose and the Methodology and formats adopted to build it.

2 Problematic

By the increasing of the shopping websites, therefore the increase of the different offers for the products which make us wonder about the easiest way for online shopping, how can we get a better price and can we search and compare between the offers by our own? Is there any alternative?

3 General description of our application

Our project is an Android application we create it to help shoppers to find the best deals and best price of product to come up with one answer for all the ambiguous questions that occurs to their minds, which is having an application for comparing the prices. This application searches in the shopping websites on the desired product and display all the available suggestions for the customer, and that by comparing and organizing the results according to the customer preferences by this the clients get what they want in the easiest ways.

4 Purpose of the application

- Presenting the different and available products
- Facilitating the process of searching for suitable and better prices
- Saving time and effort

Chapter 01 : Project Presentation

- The quick and detailed scan for results

5 Software Development Life Cycle Models and Methodologies

Software development life cycle (SDLC) is important for the software project success, the good software engineer should have enough experience and knowledge to prefer and choose one model than another based on the project context.

Therefore, it may be required to choose the right SDLC model according to the specific concerns and requirements of the project.

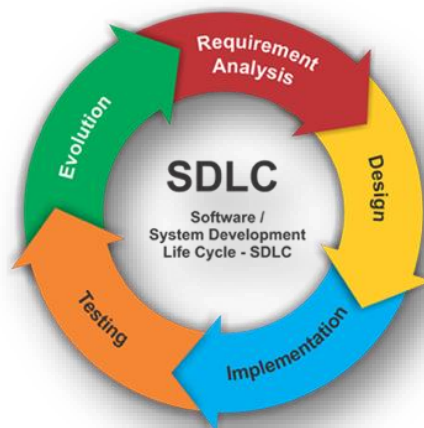


Figure 1:Software Development Life Cycle

5.1 Types of Software Development Life Cycle

- Waterfall Model
- V-Shaped Model
- Evolutionary Prototyping Model
- Spiral Method (SDM)

Chapter 01 : Project Presentation

- Iterative and Incremental Method
- Agile development

5.2 Methodology and formats adopted:

5.2.1 Waterfall Model:

5.2.1.1 Description:

The Waterfall Model is a linear sequential flow. In which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirement. The waterfall approach is the earliest approach and most widely known that was used for software development

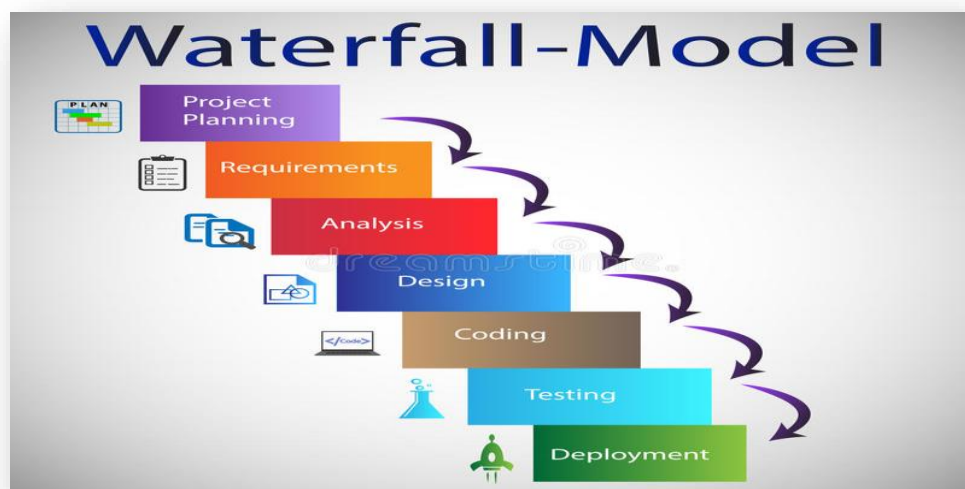


Figure 2:Waterfall Model

Chapter 01 : Project Presentation

5.2.1.2 When can we use the waterfall model

Projects which not focus on changing the requirements, for example, projects initiated from request for proposals (RFPs), the customer has a very clear documented requirements

5.2.1.3 Waterfall model advantages

- Easy to explain to the users.
- Structures approach.
- Stages and activities are well defined.
- Helps to plan and schedule the project.
- Verification at each stage ensures early detection of errors/misunderstanding.
- Each phase has specific deliverables.¹

6 Conclusion

After we had known the general idea about our application we have to start analyzing and specifying the requirements to build it and this what we will see in the next chapter.

Chapter 02

*Analysis and specification of
requirements*

Chapter 02: Analysis and specification of requirements

1 Introduction

Since our project is an Android application designed to comparing prices, in this chapter we will talk about price comparison engines android operating system and about everything is related to them and our project at the same time and all this for the purpose of understanding the theme more and more.

2 Search engine:

2.1 Definition:

A search engine is a web site that collects and organizes content from all over the internet. Those wishing to locate something would enter a query about what they'd like to find and the engine provides links to content that matches what they want. ²

2.2 Why people use search engine:

Generally, people use search engines for one of three things:

- ✚ Research
- ✚ Shopping
- ✚ Entertainment. ³

2.3 The purpose of a search engine:

The purpose of a search engine is to extract requested information from the huge database of resources available on the internet. Search engines become an important day to day tool for finding the required information without knowing where exactly it is stored.

Chapter 02: Analysis and specification of requirements

2.4 Types of Search Engines:

Search engines are classified into the following three categories based on how it works.

➤ Crawler based search engines:

All crawler based search engines use a crawler or bot or spider for crawling and indexing the new content to the search database. There are four basic steps which every crawler based search engines follow before displaying any sites in the search results.

- Crawling.
- Indexing.
- Calculating Relevancy.
- Retrieving the Result.

➤ Human powered directories:

Human powered directories also referred as open directory system depends on human based activities for listings. The process of indexing is explained as below:

1. Site owner submits a short description of the site to the directory along with category it is to be listed.
2. Submitted site is then manually reviewed and added in the appropriate category or rejected for listing.
3. Keywords entered in a search box will be matched with the description of the sites.

This means the changes made to the content of a web pages are not taken into consideration as it is only the description that matters.

➤ Hybrid search engines

Hybrid Search Engines use both crawler based and manual indexing for listings the sites in search results. Most of the crawler based search.

Chapter 02: Analysis and specification of requirements

engines like Google basically uses crawlers as a primary mechanism and manual screening as a secondary mechanism ⁴

3 Comparison shopping engines (CSEs):

3.1 What are comparison shopping engines (CSEs)?

Comparison shopping engines (CSEs, price comparison websites, comparison shopping websites) are channels gather information about products and services, their prices, descriptions, features, etc., and then show that combined information on one page as a response to visitors' search request These are some examples of comparison price engines:

- Google Shopping
- NexTag
- PriceGrabber
- Shopping.com
- Shopzilla
- Become
- Bing Shopping Campaigns
- Pronto
- BizRate
- Amazon Product Ads

3.2 How does a comparison shopping engines (CSEs) work?

To compare data, you need to get it first. Getting and updating data is probably the most important and challenging task for price comparison engines. So how do they collect the product data? Basically, there are

Chapter 02: Analysis and specification of requirements

three options for gathering various product information for price comparison engines.

3.2.1 Getting data from merchant's feeds:

Price comparison websites can establish an agreement with a store or a company so that they can get the data directly from retailers' web feeds. What is more, they can also become an affiliate partner and get a referral commission.

Positives:

- You don't need to implement any technology to get in.

Negatives:

- It demands a lot of manual work.
- Inability to make real-time updates.
- Getting the needed information from thousands of merchants separately will get you nowhere.
- Managing information coming in multiple formats from multiple sources is a hell.

3.2.2 Crawling:

Using crawling bots, which regularly gather information from websites for further processing. Thus, it is possible to get accurate information about products, which is available on the site. However, many websites have special anti-crawling mechanisms, which block users that make too many requests.

Positives:

- You can get rich and reliable data of a product.
- You don't depend on merchants to provide you with the data.

Chapter 02: Analysis and specification of requirements

Negatives:

- Expensiveness.
- Anti-crawling mechanism on websites, restrict from gathering info.
- Difficulties in writing a crawler for each of the websites.
- Inability to always get an up-to-date data, as crawling takes quite a long time.

3.2.3 API integration:

You can integrate with shopping carts and get all the data needed for accurate product comparison via API requests: Prices, variants, colors, images, descriptions, etc.

Positives:

- It saves a lot of time on development;
- Allows to integrate comparison price engines with multiple shopping carts easily.
- Allows to gather products' images, descriptions, variants, and options from stores, and process it within your system.

Negatives:

- It is not free. ⁵

4 API (Application Programming Interface):

4.1 What is an API? (Application Programming Interface):

API is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other. Each time you use an app like Facebook, send an instant message, or check the weather on your phone, you're using an API.⁶

Chapter 02: Analysis and specification of requirements

4.2 What Do APIs Do?

A website uses a URL address to make a call to a server and pull up a webpage in a browser. APIs also facilitate calls to a server, but they execute it more simply. They connect the web, allowing developers, applications, and sites to tap into databases and services (or, assets)—much like open-source software. ⁷

4.3 Common Methods of API:

To understand the most common method of Application Programming Interface we have to know what is Authentication and Authorization first.

4.3.1 Authentication and Authorization:

- ✚ Authentication is when an entity proves an identity. In other words, Authentication proves that you are who you say you are.
- ✚ Authorization is an entirely different concept, though it is , Authorization is when an entity proves a right to access. In other words, Authorization proves you have the right to make a request.

4.3.2 Common Methods of API Authentication:

4.3.2.1 HTTP Basic Authentication:

One solution is that of HTTP Basic Authentication. In this approach, an HTTP user agent simply provides a username and password to prove their authentication. This approach does not require cookies, session IDs, login pages, and other such specialty solutions, and because it uses the HTTP header itself, there's no need to handshakes or other complex response systems.

Chapter 02: Analysis and specification of requirements

4.3.2.2 API Keys:

API Keys were created as somewhat of a fix to the early authentication issues of HTTP Basic Authentication and other such systems. In this approach, a unique generated value is assigned to each first time user, signifying that the user is known. When the user attempts to re-enter the system, their unique key is used to prove that they're the same user as before.

4.3.2.3 OAuth:

OAuth is not technically an authentication method, but a method of both authentication and authorization.

In this approach, the user logs into a system. That system will then request authentication, usually in the form of a token. The user will then forward this request to an authentication server, which will either reject or allow this authentication. From here, the token is provided to the user, and then to the requester. Such a token can then be checked at any time independently of the user by the requester for validation, and can be used over time with strictly limited scope and age of validity.⁸

5 Application System Modeling:

5.1 What is System Modeling:

5.1.1 What is UML:

UML, short for Unified Modeling Language, is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.⁹

Chapter 02: Analysis and specification of requirements

5.1.2 UML diagram types:

1. Activity diagrams: which show the activities involved in a process or in data processing.
2. Use case diagrams: which show the interactions between a system and its environment.
3. Sequence diagrams: which show interactions between actors and the system and between system components.
4. Class diagrams: which show the object classes in the system and the associations between these classes.
5. State diagrams: which show how the system reacts to internal and external events.¹⁰

5.1.3 Advantages of UML :

➤ Visual Representation:

A UML diagram is a visual representation of the relationships between classes and entities in a computer program. A class is an object in programming that organizes similar variables and functions in one location. To understand a program, it is essential to understand what each class object does the information it stores and how it relates to other classes in the program. By showing this information in a diagram, it is easy to understand and visualize a program's relationships.

➤ Readability and Re-usability:

A UML diagram is beneficial in that it is very readable. The diagram is meant to be understood by any type of programmer and helps to explain relationships in a program in a straightforward manner. Traditionally, to understand a program, a programmer would read the code directly. This could be thousands or millions of lines of code in very large programs. Having a UML diagram helps to quickly illustrate those relationships.

Chapter 02: Analysis and specification of requirements

Additionally, by using a diagram to show the code running in a program, a programmer is able to see redundant code and reuse portions of code that already exist rather than rewrite those functions.

➤ **Standard:**

UML is the current standard for programming in object-oriented programming languages. When creating classes and other objects with relationships between each other, UML is what is used to visually describe these relationships. Because it is used as a standard, it is widely understood and well known. This makes it easy for a new programmer to step into a project and be productive from day one.

➤ **Planning Tool:**

UML helps to plan a program before the programming takes place. In some tools used to model UML, the tool will generate code based on the classes set up in the model. This can help reduce overhead during the implementation stage of any program. Additionally, a UML model diagram is easy to change, whereas reprogramming a section of code can be tedious and time-consuming.¹¹

6 Mobile Applications :

6.1 Mobile Device :

6.1.1 What Is a Mobile Device?

A mobile device is a small, handheld computing device usually outfitted with a display screen and input method (either touchscreen or miniature keyboard). Many portable mobile devices have operating systems that can run applications (or “apps”). Apps make it possible for mobile devices and phones to be used as gaming devices, media players, calculators, navigators, and more.¹²

Chapter 02: Analysis and specification of requirements

6.1.2 Mobile Device Types:

There are four primary mobile devices:

- ❖ Feature phones:
- ❖ Multimedia phones:
- ❖ Smartphones
- ❖ Tablets

6.2 Mobile Applications:

6.2.1 What is Mobile Applications?

A mobile app is a software application developed specifically for use on small, wireless computing devices, such as smartphones and tablets, rather than desktop or laptop computers.

Mobile apps are sometimes categorized according to whether they are web-based or native apps, which are created specifically for a given platform. A third category, hybrid apps, combines elements of both native and Web apps. As the technologies mature, it's expected that mobile application development efforts will focus on the creation of browser-based, device-agnostic Web applications.

6.2.2 History of Mobile Applications:

If you go back to the history of the mobile applications, then you can clearly figure out that a few Java games, a calculator or monthly calendar were all that came under the category of mobile apps. However, the first smart phone was announced for the general use by IBM in 1993 that was equipped with the features like calculator, world clock, calendar and contact book. The BlackBerry Smartphone released in 2002 was the next major achievement in the field of mobile application development and it was marked by BlackBerry Limited, formerly known as Research In

Chapter 02: Analysis and specification of requirements

Motion Limited (RIM) and integrated with the innovative concept of wireless email.¹⁴

6.2.3 Applications mobiles Types:

- ❖ Native Applications.
- ❖ Mobile web Applications.
- ❖ Hybrid Applications.

6.3 Integrated Development Environment (IDE):

6.3.1 What is (IDE)?

An integrated development environment (IDE) is an application that facilitates application development. IDEs are designed to encompass all programming tasks in one application. Therefore, IDEs offer a central interface featuring all the tools a developer needs, including the following:

- Code editor: This feature is a text editor designed for writing and editing source code. Source code editors are distinguished from text editors because they enhance or simplify the writing and editing of code.
- Compiler: This tool transforms source code written in a human readable/writable language into a form executable by a computer.
- Debugger: This tool is used during testing to help debug application programs.
- Build automation tools: These tools automate common developer tasks.

In addition, some IDEs might also include the following:

- Class browser: This tool is used to examine and reference the properties of an object-oriented class hierarchy.

Chapter 02: Analysis and specification of requirements

- Object browser: This feature is used to examine the objects instantiated in a running application program.
- Class hierarchy diagram: This tool allows the programmer to visualize the structure of object-oriented programming code.

6.3.2 History of (IDEs):

Before IDEs, developers wrote their programs in text editors. They would write and save an application in a text editor; then run the compiler, taking note of the error messages; then go back to the text editor to revise the code. In 1983, Borland Ltd. acquired a Pascal compiler and released it as TurboPascal, which featured, for the first time, an integrated editor and compiler. While TurboPascal launched the idea of an integrated development environment, many believe Microsoft's Visual Basic (VB), launched in 1991, was the first real IDE. Visual Basic was built on the older BASIC language, which was a popular programming language throughout the 1980s. With the emergence of Visual Basic, programming could be thought of in graphical terms, and significant productivity benefits emerged.

6.3.3 Benefits of (IDEs):

The overall goal and main benefit of an integrated development environment is improved developer productivity. IDEs boost productivity by reducing setup time, increasing the speed of development tasks, keeping developers up to date and standardizing the development process.

- Faster setup: Without an IDE interface, developers would need to spend time configuring multiple development tools. With the application integration of an IDE, developers have the same set of

Chapter 02: Analysis and specification of requirements

capabilities in one place, without the need for constantly switching tools.

- **Faster development tasks:** Tighter integration of all development tasks improves developer productivity. For example, code can be parsed and syntax checked while being edited, providing instant feedback when syntax errors are introduced. Developers don't need to switch between applications to complete tasks.
- **Continual learning:** Staying up to date and educated is another benefit. For instance, the IDE's help topics are constantly being updated, as well as new samples, project templates, etc. Programmers who are continually learning and current with best practices are more likely to contribute value to the team and the enterprise, and to boost productivity.
- **Standardization:** The IDE interface standardizes the development process, which helps developers work together more smoothly and helps new hires get up to speed more quickly.

6.3.4 Different types of (IDEs):

There are a variety of different IDEs, catering to the many different ways developers work and the different types of code they produce. There are IDEs that are designed to work with one specific language, cloud-based IDEs, IDEs customized for the development of mobile applications or for HTML, and IDEs meant specifically for Apple development or Microsoft development.

- **Multi-Language IDEs**

Multi-language IDEs, such as Eclipse, NetBeans, Komodo, Aptana and Geany, support multiple programming languages.

- **IDEs for Mobile Development**

Chapter 02: Analysis and specification of requirements

There are IDEs specifically for mobile development, including PhoneGap and Appcelerator's Titanium Mobile.

Many IDEs, especially those that are multi-language, have mobile-development plugins. For instance, Eclipse has this functionality.

- HTML IDEs

Some of the most popular IDEs are those for developing HTML applications. For example, IDEs such as HomeSite, DreamWeaver or FrontPage automate many tasks involved in web site development.

- Cloud-Based IDEs

Cloud-based IDEs are starting to become mainstream. The capabilities of these web-based IDEs are increasing rapidly, and most major vendors will likely need to offer one to be competitive. Cloud IDEs give developers access to their code from anywhere. For example, Nitrous is a cloud-based development environment platform that supports Ruby, Python, Node.js and more. Cloud9 IDE supports more than 40 languages

- IDEs Specific to Microsoft or Apple

These IDEs cater to those working in Microsoft or Apple environments:

- IDEs for Specific Languages

Some IDEs cater to developers working in a single language. These include CodeLite and C-Free for C/C++, Jikes and Jcreator for Java, Idle for Python, and RubyMine for Ruby/Rails.¹⁷

6.4 Android Operating System:

6.4.1 What is Android:

Android OS is a Linux-based open source operating system formed in 2007 by “the consortium of the initial members of the Open Handset

Chapter 02: Analysis and specification of requirements

Alliance (OHA) such as Google. It's recognized mostly for being mainly written in Java. ¹⁸

6.4.2 History of Android:

Here is the presentation of the android history in a sequence:

- 1) Initially, Andy Rubin founded Android Incorporation in Palo Alto, California, United States in October, 2003.
- 2) In 17th August 2005, Google acquired android Incorporation. Since then, it is in the subsidiary of Google Incorporation.
- 3) The key employees of Android Incorporation are Andy Rubin, Rich Miner, Chris White and Nick Sears.
- 4) Originally intended for camera but shifted to smart phones later because of low market for camera only.
- 5) Android is the nick name of Andy Rubin given by coworkers because of his love to robots.
- 6) In 2007, Google announces the development of android OS.
- 7) In 2008, HTC launched the first android mobile. ¹⁹

6.4.3 Why does Android look different on each phone?

Android doesn't look different on every device, but it does have a number of different versions. Android is open-source, which means that manufacturers are free to customize the software and make it their own.

6.4.4 Android versions , code name and API:

Each new version of Android gets a code name based on consecutive letters of the alphabet. The most recent version is known as Marshmallow because it is the Android M release. Previous versions have included Lollipop, KitKat, Jelly Bean and Gingerbread.

Chapter 02: Analysis and specification of requirements

Let's see the android versions, codenames and API Level provided by Google:

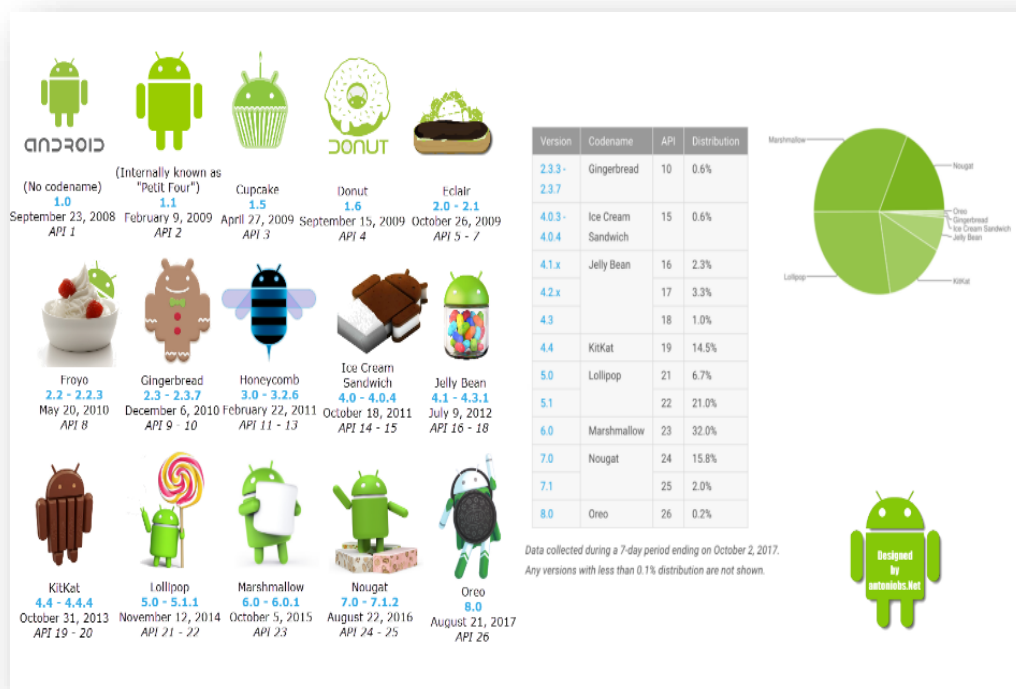


Figure 3: Android versions, code name and API

6.4.5 What are the advantages of Android?

Choice. For example, if you want iOS, you have a choice of iPhone, iPhone or iPhone. If you go for Android there are stacks of great devices to choose from. Those devices are often cheaper than the equivalent Apple devices.

Android's choice isn't just about hardware. It's about everything else too. Android is incredibly easy to customize, both in terms of how it looks and how it works, and the various app stores aren't as tightly controlled as its rivals' stores, like Apple.²⁰

Chapter 02: Analysis and specification of requirements

7 Conclusion:

So everything mentioned in this chapter is the most important requirement we need to build our application. Now, we will start the process of building the system and the first step that will be used is conception and modeling, this is what we will see in the next chapter

Chapter 03

Conception and Modeling

Chapter 03: Conception and Modeling

1 Introduction:

In this chapter we will describe the system by showing the general architecture of the app then we will specify the actions (use cases) that the system should or can perform , show the object interactions arranged in time sequence and describe the structure of the system.

2 The application architecture:

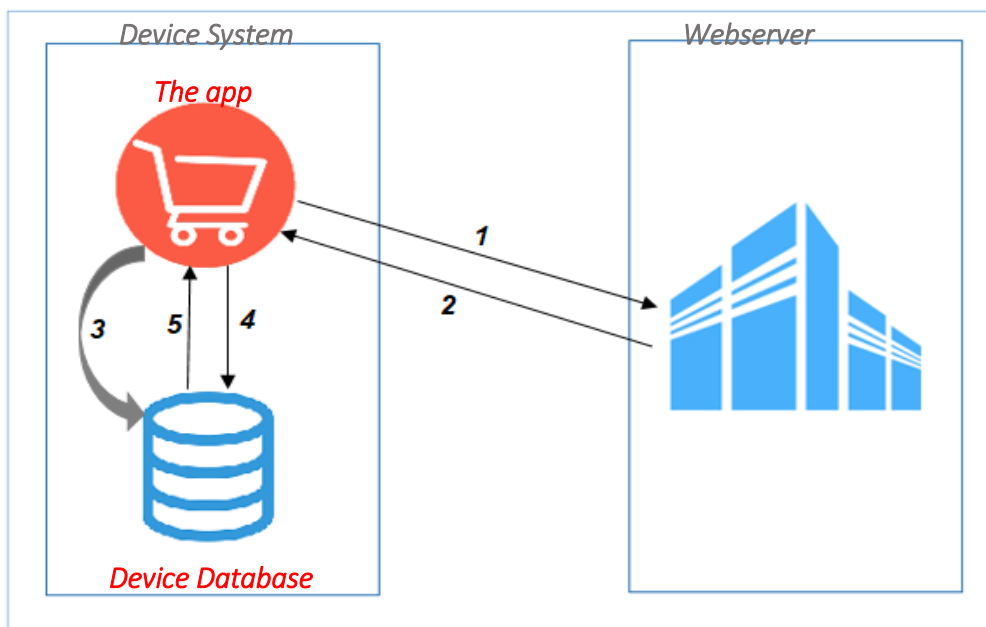


Figure 4: Application Architecture

Action number	Action description
1	Request the products
2	Response
3	Store incoming products
4	Request products from the local database
5	Return the available products

Table 1: Application Architecture steps

Chapter 03: Conception and Modeling

3 System Modeling:

For the System Modeling we will use the Unified Modeling Language (UML):

3.1 Use Case diagram:

A UML use case diagram is the primary form of system/software requirements for a new software program under developed. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (such as UML). A key concept of use case modeling is that it helps us design a system from end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.²¹

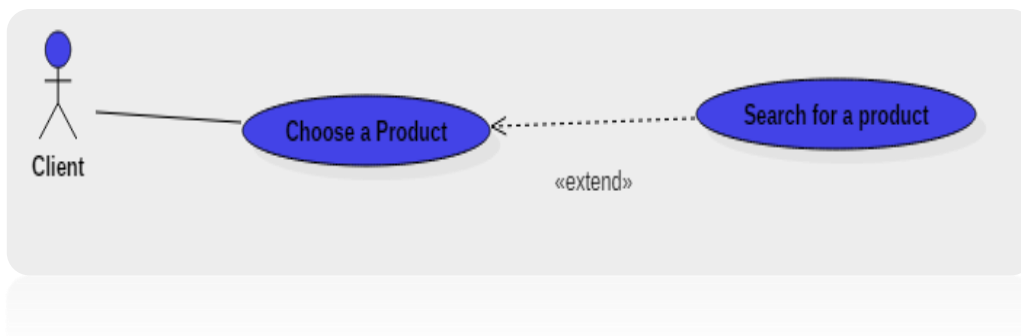


Figure 5: Use Case diagram for the visitors of the application

3.2 Sequence Diagram:

The Sequence Diagram models the collaboration of objects based on a time sequence. It shows how the objects interact with others in a particular scenario of a use case. With the advanced visual modeling capability, you can create complex sequence diagram in few clicks.²²

Chapter 03: Conception and Modeling

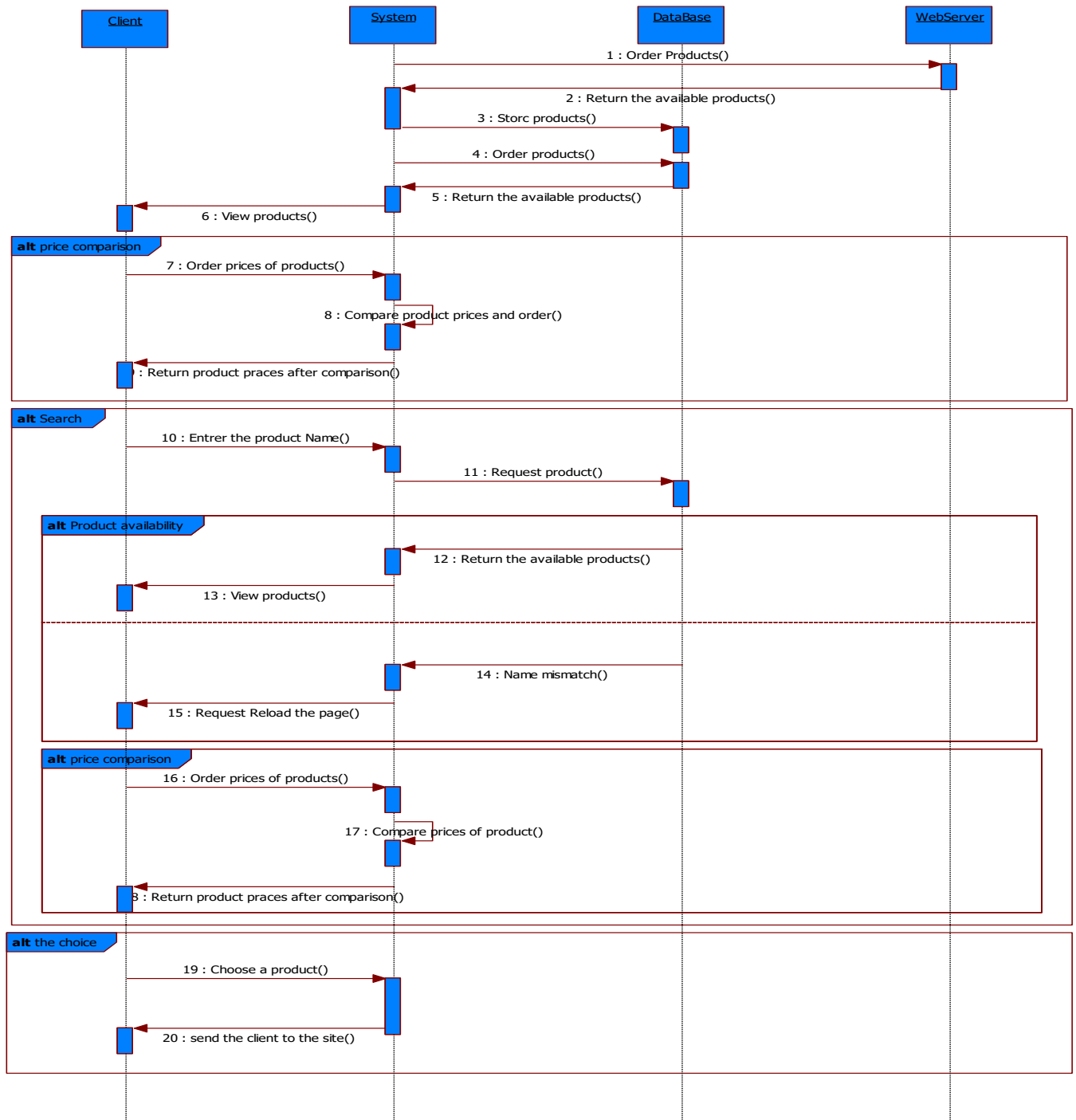


Figure 6: Sequence Diagram for the application system

Chapter 03: Conception and Modeling

3.3 Class diagram:

Class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.²³

3.3.1 Description of the class model:

Class	Attributes	Type
Product	_id	Int
	Product_Name	String
	Product_Price	String
	Shipping_Price	String
	Shipping_Cost	String
	Currency	String
	Site_Url	String
	IMG_Name	String

Table 2:Description of the class model

3.3.2 Transition to relational model:

Object model	Relational model
Class	Table
Simple type attribute	Column
Complex type attribute	Foreign key column
Instance	T-uplet

Chapter 03: Conception and Modeling

ID	primary key
Association	Foreign key or link table
Heritage	Identical primary key on multiple tables

Table 3: Transition to relational model

6.1.3 Class diagram modeling

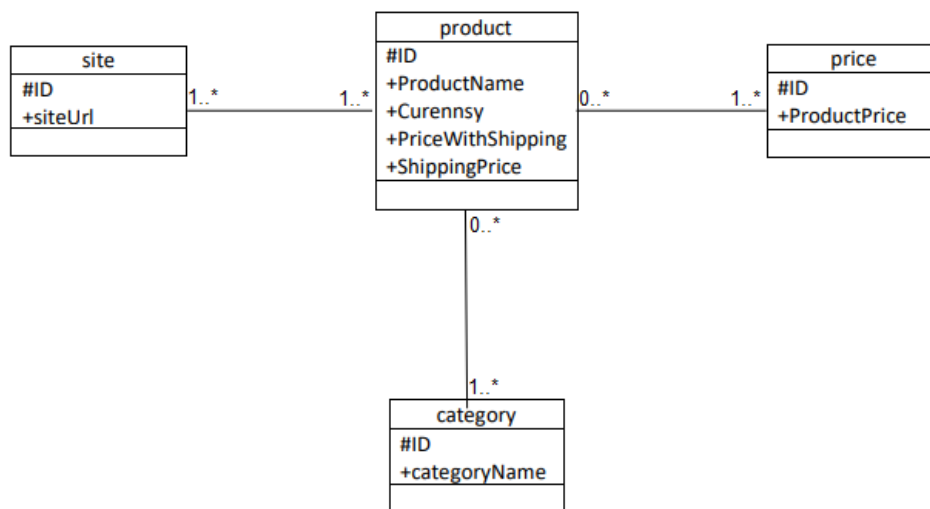


Figure 17: Database schema

7 Conclusion

After we have described the system carefully and identified the most important tasks, we have nothing left to do except programming and this will be included in the last chapter

Chapter 04

Implementation

Chapter 04: Implementation

1 Introduction

The step in this chapter is the last. In the sense of development and also the most important step in the application development .It will collect the different elements that we will need to get the application run .and in the end of this chapter we will show the different parts of our work by some screenshots

2 Platforms and Utilities:

This application is developed and tested on several platforms (machines + Operating Systems). Their characteristics are presented in the following tables:

2.1 Hardware

The Machine we used to develop this application configured as follows:

Machine: FUJITSU(LIFEBOOK A514)		
Component	Name & Version	specifications
Operating System	Windows 10 Home	64-bit
Processor	Intel(R)	Core(TM) i3-4005U @ 1.70 GHz
Memory	DDR3-1600	Capacity: 4.00 GB
Hard Disk	TOSHIBA MQ01QBF050	Capacity: 500 GB
Graphics	Intel(R) HD Graphics	—
Network Interface Card		—

Table 6:Development Machine specification

Chapter 04: Implementation

2.2 Software

The implementation of the application was also done using a set of utilities they come in this table:

Software	Name & Version	Description
	Android Studio 1.5.1	official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. ²⁴
	Genymotion 2.12.0	one of the most stable Android virtual machines that help in testing and debugging the android apps. ²⁵
 VirtualBox	Oracle VM VirtualBox 5.2.6	a cross-platform virtualization application. ²⁶
	StarUML	open source software modeling tool that supports UML (Unified Modeling Language). ²⁷
	SQLite 3.10.1	SQLite is a software library that provides a relational database management system. ²⁸

Table 7:used Applications

Chapter 04: Implementation

3 Presentation of the technologies used:

3.1 XML

"Extensible Markup Language". XML is used to define documents with a standard format that can be read by any XML-compatible application. The language can be used with HTML pages, but XML itself is not a markup language. Instead, it is a "met language" that can be used to create markup languages for specific applications. For example, it can describe items that may be accessed when a Web page loads. Basically, XML allows you to create a database of information without having an actual database. While it is commonly used in Web applications, many other programs can use XML documents as well.²⁹

3.2 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language, Standard ECMA-262 3rd Edition - December 1999. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.³⁰

3.3 JAVA

Java is a programming language that produces software for multiple platforms. When a programmer writes a Java application, the compiled code (known as byte code) runs on most operating systems (OS),

Chapter 04: Implementation

including Windows, Linux and Mac OS. Java derives much of its syntax from the C and C++ programming languages.³¹

3.4 SQL

Stands for "Structured Query Language", and can be pronounced as either "sequel" or "S-Q-L." It is a query language used for accessing and modifying information in a database. Some common SQL commands include "insert", "update" and "delete".³²

3.5 Software developer's kit (SDK):

A software developer's kit (SDK) is a set of programs used by a computer programmer to write application programs. Typically, an SDK includes a visual screen builder, an editor, a compiler, a linker, and sometimes other facilities.³³

3.6 Java Development Kit (JDK):

A Java Development Kit (JDK) is a program development environment for writing Java applets and applications. It consists of a runtime environment that "sits on top" of the operating system layer as well as the tools and programming that developers need to compile, debug, and run applets and applications written in the Java language.³⁴

To complete our project we used :**jdk1.8.0_171**

3.7 Retrofit:

Retrofit is a type-safe HTTP client for Android and Java. With Retrofit we can compose the HTTP connection easily through a simple expressive interface just like an Api document. Besides the elegant syntax it provides, it's also easy to incorporate with different library.³⁵

To complete our project we used: **retrofit: 2.3.0**

Chapter 04: Implementation

4 Presentation of the application:

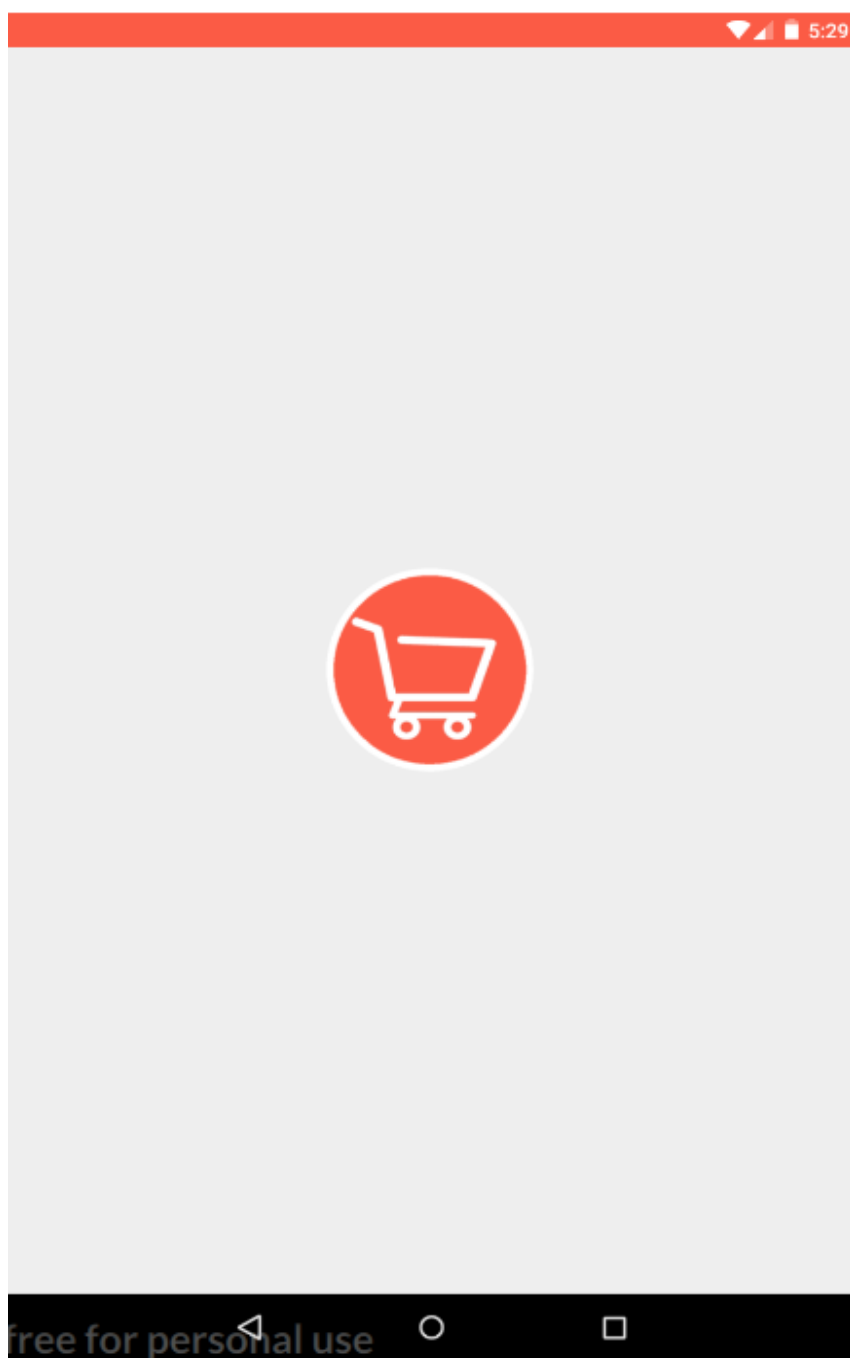


Figure 7: Start page

Chapter 04: Implementation

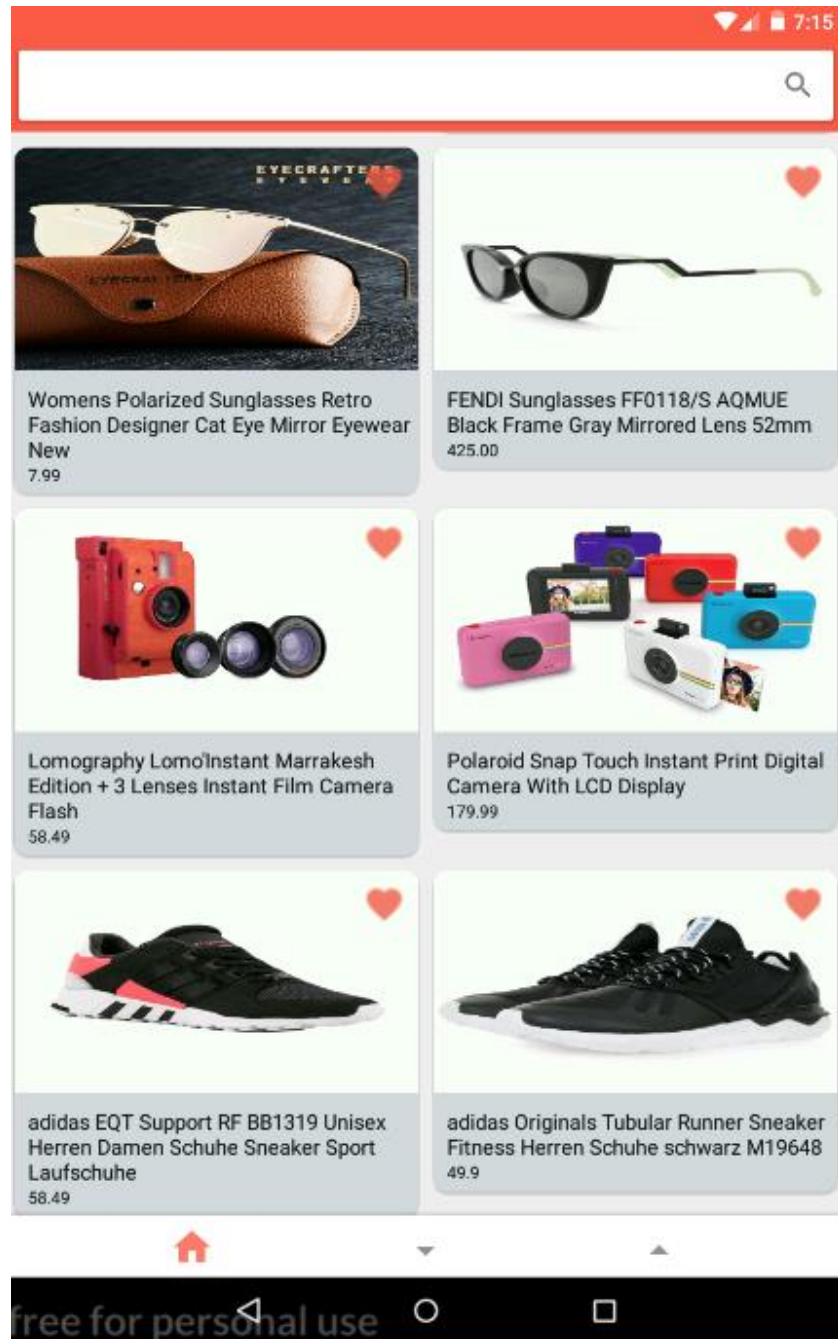


Figure 8: Home page

Chapter 04: Implementation

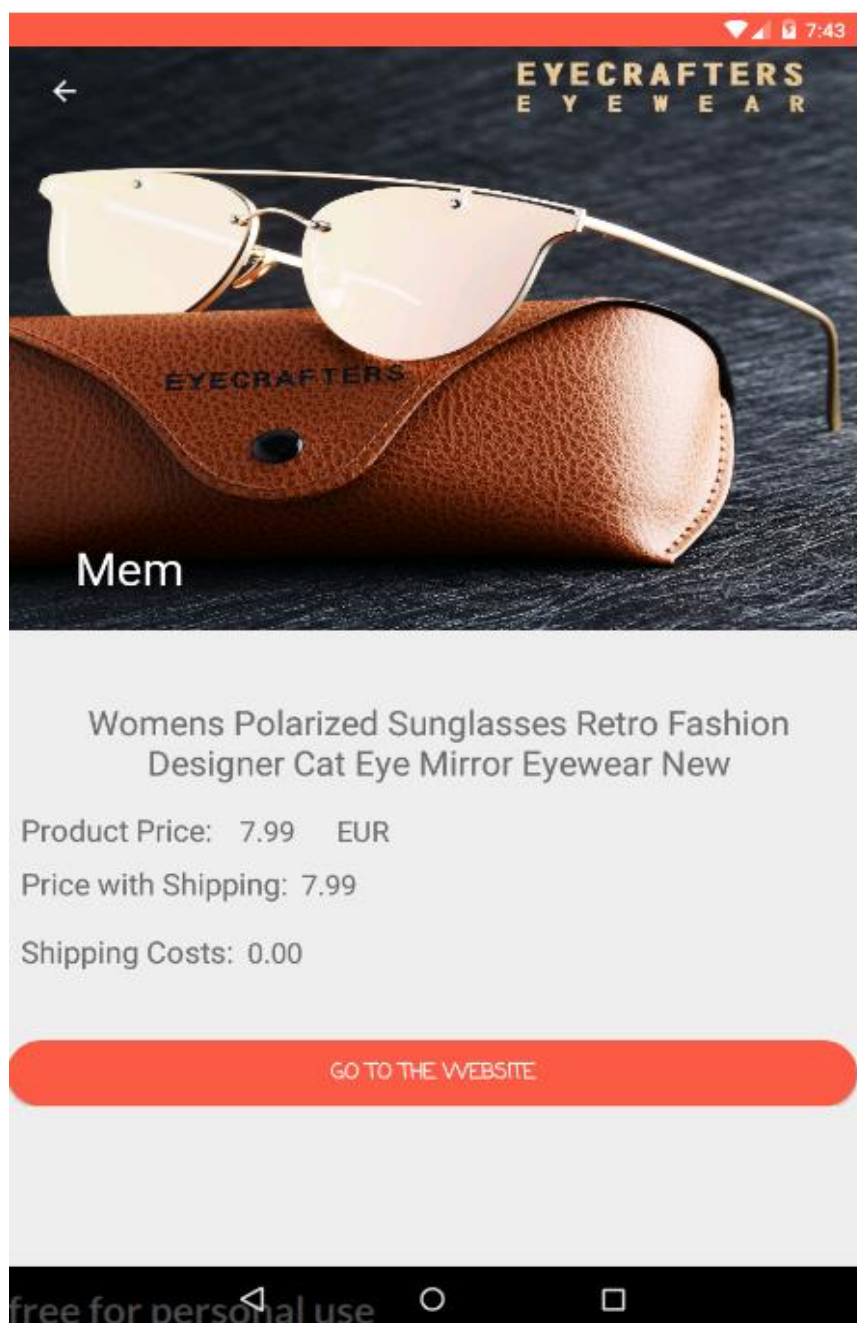


Figure 9: Product Details page

Chapter 04: Implementation

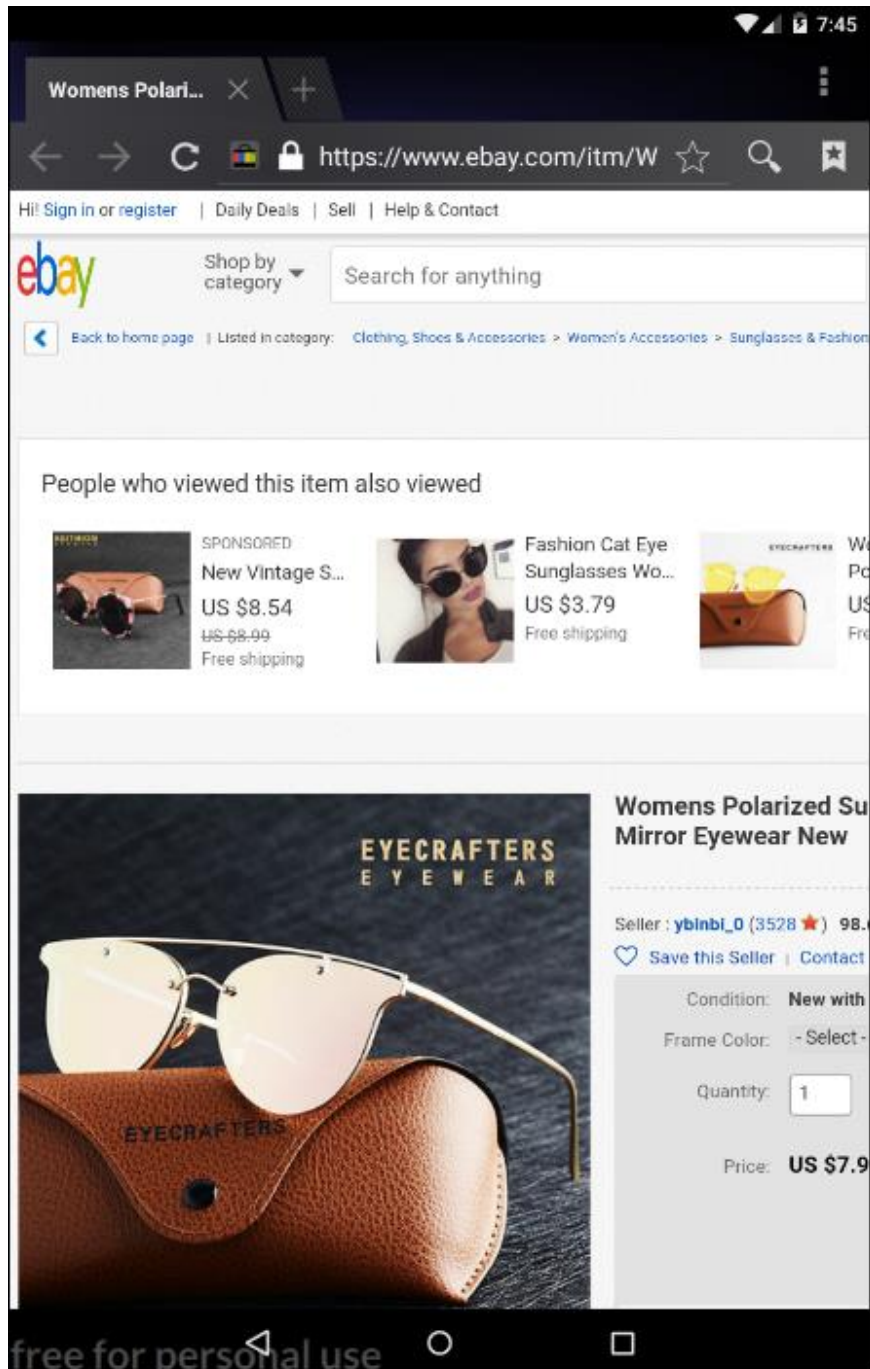


Figure 10: Web site page

Chapter 04: Implementation

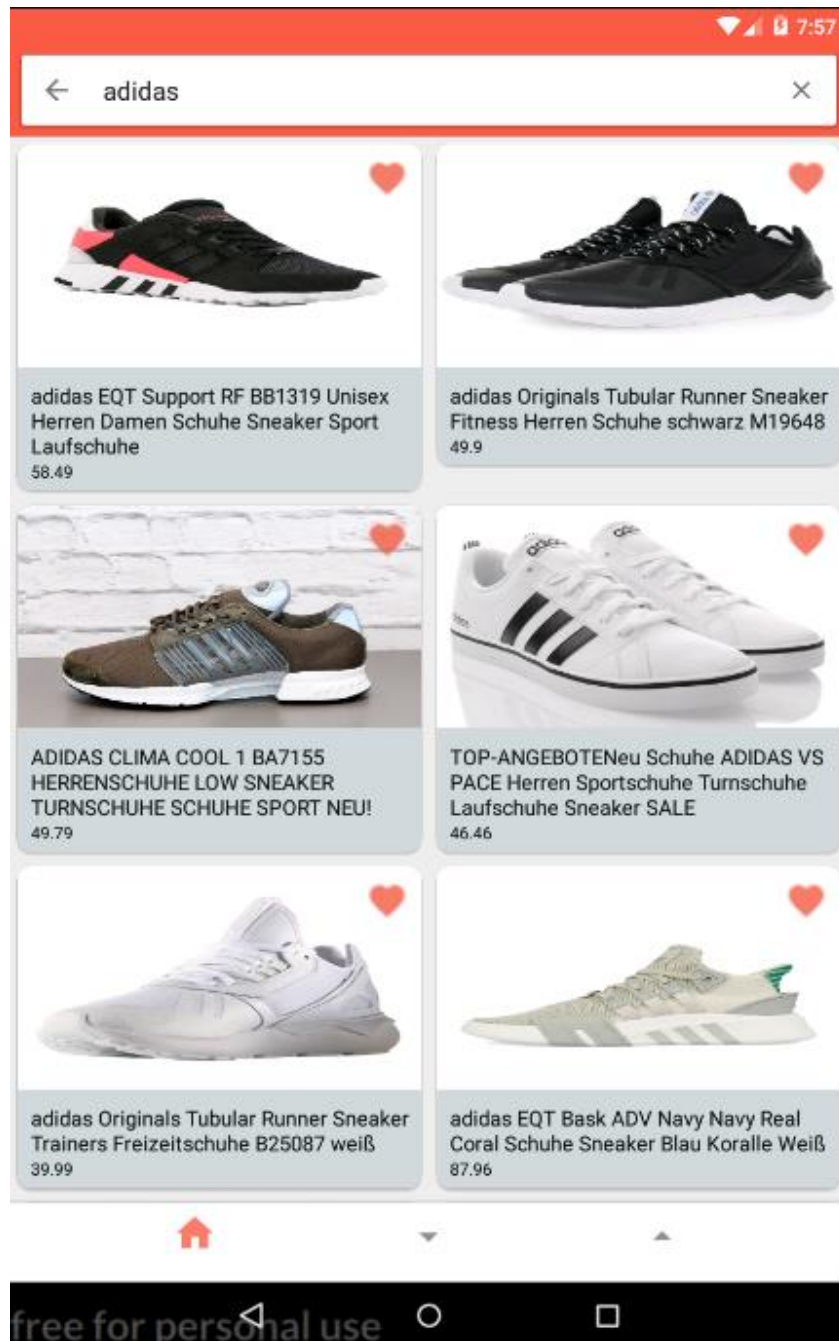


Figure 11: Search result page

Chapter 04: Implementation



Figure 12 : lowest price page

Chapter 04: Implementation

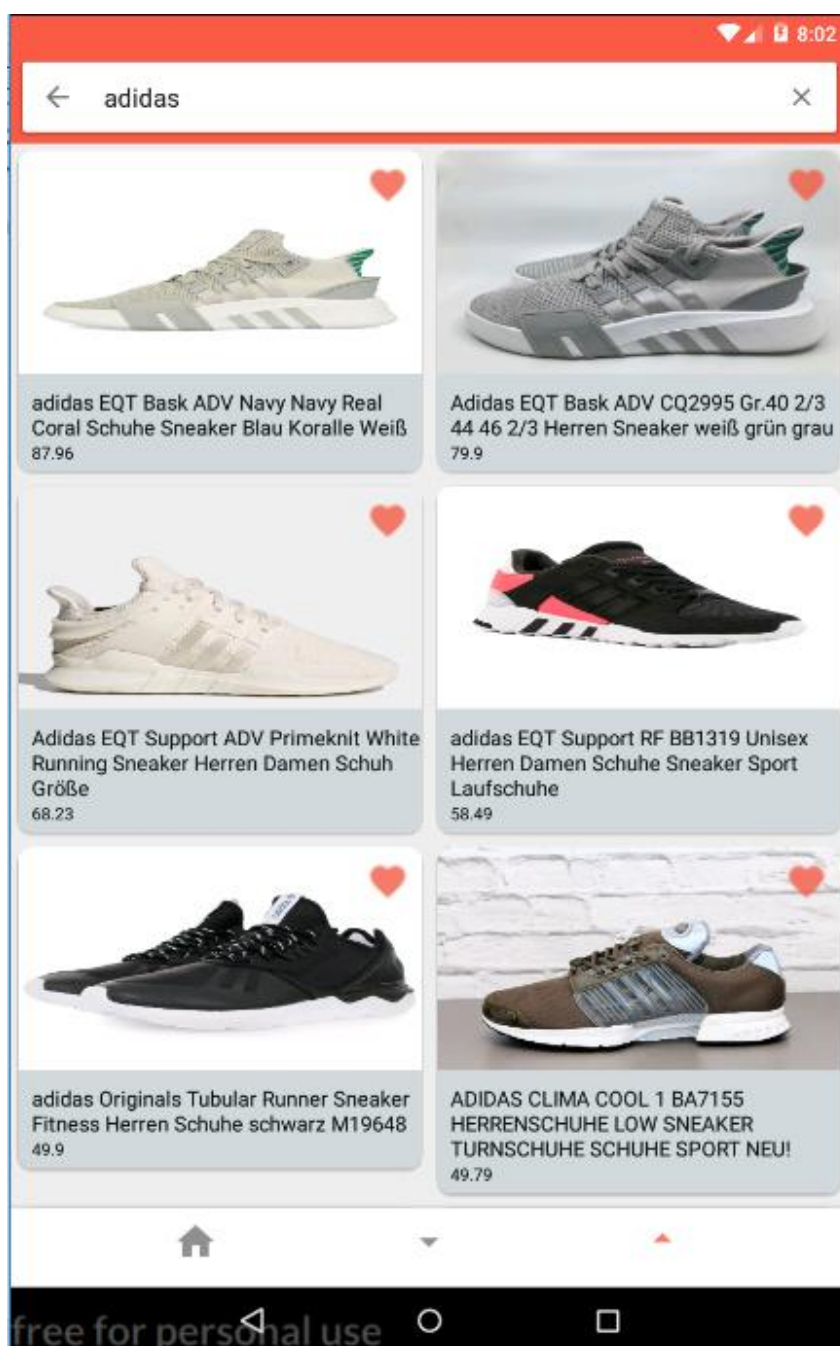


Figure 13: Highest price page

Chapter 04: Implementation

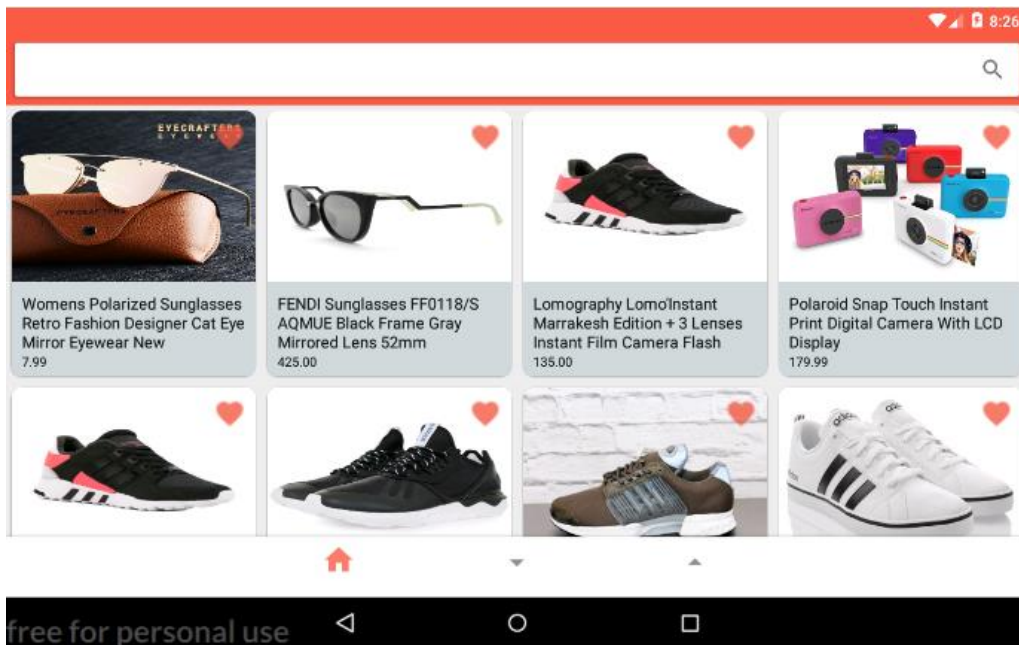


Figure 14: Home page (horizontal)

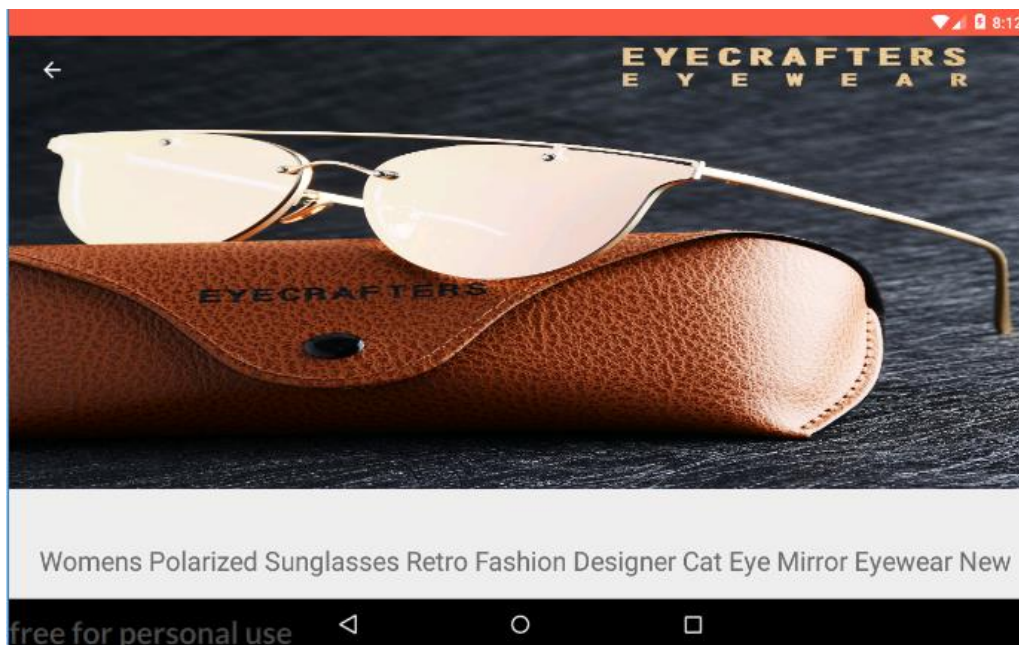


Figure 15: Product details 1 (horizontal)

Chapter 04: Implementation

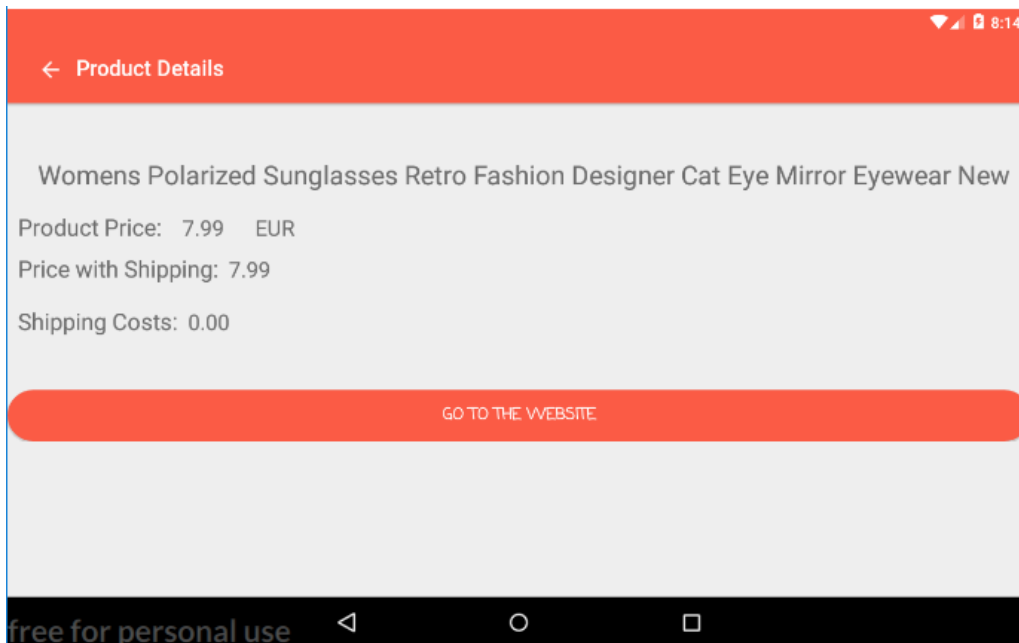


Figure 16: Product details 2 (horizontal)

As you have seen, our project has several basic interfaces that we will explain right now:

Home page : which contains a list of different products, search bar to enter the names of the products to be searched and a bottom bar for arranging the products in case you want to arrange them

Product details page : It comes after the customer has chosen a product which shows the product details, with an explanation picture of it, and it gives the possibility to access to the site where the product exists by "GO TO THE WEBSITE" button

Search result page: which is the page that contains the search result for the desired products

Chapter 04: Implementation

5 Conclusion

Now that we have finished this chapter we can say that we have finished building our application and we reached our goal and we solved the main problematic that we faced in the beginning of this project.

General Conclusion

This study was provided for the purpose of facilitating the online shopping and resolving the problem of selecting products by shoppers .

Through this study we have faced some difficult problems (Like the difficulty of getting free API) but most of them have been solved to produce an application that is easy to use and compatible with a number of devices and that's because the generality of the operating system that the app was developed by.

After all,we can say that we have solved a huge part of the problem that we talked about at the beginning of this study and that the purpose of the project was reached.

References

- 1:** Mohamed Sami “Software Development Life Cycle Models and Methodologies” 15March 2012.Web. 5. May.2018 . <https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/>
- 2:** JAMES KIMMONS « What is a Search Engine?” September 08, 2017.Web 5.May.2018 .<https://www.thebalancesmb.com/search-engine-2867354>
- 3:** dummies “WHY PEOPLE USE SEARCH ENGINES: RESEARCH, SHOPPING, AND ENTERTAINMENT”.Web. 5.May.2018 . <http://www.dummies.com/web-design-development/search-engine-optimization/why-people-use-search-engines-research-shopping-and-entertainment/>
- 4:** Editorial Staff “What are Different Types of Search Engines?” Mar 9, 2018 .Web.5.May.2018.<https://www.webnotes.com/what-are-different-types-of-search-engines/>
- 5:** Anastasia Kholod “ How do Price Comparison Engines Gather their Data “ 9 October 2017.Web.5.May .2018 <https://api2cart.com/ecommerce/price-comparison-engines-gather-data/>
- 6:** Mulesoft “What is an API? (Application Programming Interface)” .Web .5.May.2018.<https://www.mulesoft.com/resources/api/what-is-an-api>
- 7:** CAREY WODEHOUSE “Intro to APIs: What Are APIs and What Do They Do?” .Web.5.May.2018. <https://www.upwork.com/hiring/development/intro-to-apis-what-is-an-api/>

8: KRISTOPHER SANDOVAL “3 Common Methods of API Authentication Explained” FEBRUARY 6, 2018 .Web. 5.May 2018 [HTTPS://NORDICAPIS.COM/3-COMMON-METHODS-API-AUTHENTICATION-EXPLAINED/](https://nordicapis.com/3-common-methods-api-authentication-explained/)

9: visual-paradigm “What is Unified Modeling Language (UML)?” WeB .5.MAY 2018. [HTTPS://WWW.VISUAL-PARADIGM.COM/GUIDE/UML-UNIFIED-MODELING-LANGUAGE/WHAT-IS-UML/](https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/)

10: Ian Sommerville “Ch5- Software Engineering 9” Sep 7, 2010 .WEB .5.MAY.2018 [HTTPS://WWW.SLIDESHARE.NET/SE9/CH5-5147965](https://www.slideshare.net/SE9/CH5-5147965)

11: mICHELLEfRIESEN “List of Advantages of UML” .web.5.may.2018. [HTTPS://WWW.TECHWALLA.COM/ARTICLES/LIST-OF-ADVANTAGES-OF-UML](https://www.techwalla.com/articles/list-of-advantages-of-uml)

12: LDS Media Library “Mobile Devices” .web .5.may.2018. [HTTPS://WWW.LDS.ORG/MEDIA-LIBRARY/ACCESSING-MEDIA-MOBILE?LANG=ENG](https://www.lds.org/media-library/accessing-media-mobile?lang=eng)

13: instapage “What is a Mobile App Landing Page? Chapter 3” .web .5.may.2018 . [HTTPS://INSTAPAGE.COM/WHAT-IS-A-MOBILE-APP-LANDING-PAGE-CHAPTER-3](https://instapage.com/what-is-a-mobile-app-landing-page-chapter-3)

14: Mehul Rajput “Tracing the History and Evolution of Mobile Apps” November 27, 2015 .web.5may.2018 . [HTTPS://TECH.CO/MOBILE-APP-HISTORY-EVOLUTION-2015-11](https://tech.co/mobile-app-history-evolution-2015-11)

15: thinkmobiles “What are the popular types and categories of apps” .web.5.may.2018 . [HTTPS://THINKMOBILES.COM/BLOG/POPULAR-TYPES-OF-APPS/](https://thinkmobiles.com/blog/popular-types-of-apps/)

16: Mallikarjun M K “Different Types of Mobile Applications – Native, Hybrid and Web Apps” February 20, 2017 .Web .5.May.2018 . <https://blog.trigent.com/different-types-of-mobile-applications-native-hybrid-and-web-apps/>

17: veracode “WHAT IS AN INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)?”.web.5.may.2018.[HTTPS://WWW.VERACODE.COM/SECURITY/INTEGRATED-DEVELOPMENT-ENVIRONMENTS](https://www.veracode.com/security/integrated-development-environments)

18: Angela Stringfellow “What is Android OS?” November 8, 2017 .web .5.May.2018.
[HTTPS://WWW.TCCROCKS.COM/BLOG/WHAT-IS-ANDROID-OS/](https://www.tccrocks.com/blog/what-is-android-os/)

19: javatpoint “History of Android” .web.5.may.2018.
[HTTPS://WWW.JAVATPOINT.COM/ANDROID-HISTORY-AND-VERSIONS](https://www.javatpoint.com/android-history-and-versions)

20: Cory Schmidt “What is Android? Here is a complete guide for beginners” jun.24,2016.web.5.may.2018. [HTTPS://WWW.ANDROIDPIT.COM/WHAT-IS-ANDROID](https://www.androidpit.com/what-is-android)

21: VISUAL PARADIGM “WHAT IS USE CASE DIAGRAM?”.WEB.12.MAY.2018.
[HTTPS://WWW.VISUAL-PARADIGM.COM/GUIDE/UML-UNIFIED-MODELING-LANGUAGE/WHAT-IS-USE-CASE-DIAGRAM/](https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/)

22: VISUAL PARADIGM “WHAT IS UNIFIED MODELING LANGUAGE (UML)?”.WEB.12.MAY.2018. [HTTPS://WWW.VISUAL-PARADIGM.COM/GUIDE/UML-UNIFIED-MODELING-LANGUAGE/WHAT-IS-UML/#SEQUENCE-DIAGRAM.](https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/#sequence-diagram)

23: VISUAL PARADIGM “WHAT IS CLASS DIAGRAM?”.WEB.12.MAY.2018.
[HTTPS://WWW.VISUAL-PARADIGM.COM/GUIDE/UML-UNIFIED-MODELING-LANGUAGE/WHAT-IS-CLASS-DIAGRAM/](https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/)

24: DEVELOPER” MEET ANDROID STUDIO” WEB.12.MAY.2018.
[HTTPS://DEVELOPER.ANDROID.COM/STUDIO/INTRO/](https://developer.android.com/studio/intro/)

25: JAY SARASWAT" WHAT IS GENYMOTION AND WHY IT IS THE BEST ANDROID EMULATOR?" NOVEMBER 28, 2017. WEB.12.MAY.2018.

[HTTPS://WWW.WEBMOBRIL.COM/WHAT-IS-GENYMOTION-AND-WHY-IT-IS-THE-BEST-ANDROID-EMULATOR/](https://www.webmobril.com/what-is-genymotion-and-why-it-is-the-best-android-emulator/)

26: [VIRTUALBOX ."CHAPTER01" WEB.12.MAY.2018.](https://www.virtualbox.org/manual/ch01.html)

[HTTPS://WWW.VIRTUALBOX.ORG/MANUAL/CH01.HTML](https://www.virtualbox.org/manual/ch01.html)

27: [METHODSANDTOOLS" STARUML - OPEN SOURCE UML TOOL"](http://www.methodsandtools.com/tools/staruml)

WEB.12.MAY.2018.[HTTP://WWW.METHODSANDTOOLS.COM/TOOLS/STARUML.PHP](http://www.methodsandtools.com/tools/staruml.php)

28: [SQLITETUTORIAL" WHAT IS SQLITE" WEB.12.MAY.2018.](http://www.sqlitetutorial.net/what-is-sqlite/)

[HTTP://WWW.SQLITETUTORIAL.NET/WHAT-IS-SQLITE/](http://www.sqlitetutorial.net/what-is-sqlite/)

29: [TECHTERMS" XML". WEB.12.MAY.2018.](https://techterms.com/definition/xml)

[HTTPS://TECHTERMS.COM/DEFINITION/XML](https://techterms.com/definition/xml)

30: [JSON.ORG "INTRODUCING JSON". WEB.12.MAY.2018.](https://www.json.org/)

[HTTPS://WWW.JSON.ORG/](https://www.json.org/)

31: [TECHOPEDIA"JAVA". WEB.12.MAY.2018.](https://www.techopedia.com/definition/3927/java)

[HTTPS://WWW.TECHOPEDIA.COM/DEFINITION/3927/JAVA](https://www.techopedia.com/definition/3927/java)

32: [techterms"sql" WEB.12.MAY.2018.](https://techterms.com/definition/sql)

[HTTPS://TECHTERMS.COM/DEFINITION/SQL](https://techterms.com/definition/sql)

33: [Margaret Rouse" software developer's kit \(SDK\)". WEB.12.MAY.2018.](https://whatistechtarget.com/definition/software-developers-kit-sdk)

[HTTPS://WHATIS.TECHTARGET.COM/DEFINITION/SOFTWARE-DEVELOPERS-KIT-SDK](https://whatistechtarget.com/definition/software-developers-kit-sdk)

34: [Margaret Rouse" java developer's kit \(SDK\)". WEB.12.MAY.2018.](https://www.theserverside.com/definition/java-development-kit-jdk)

[HTTPS://WWW.THESERVERSIDE.COM/DEFINITION/JAVA-DEVELOPMENT-KIT-JDK](https://www.theserverside.com/definition/java-development-kit-jdk)

35: [Jintin" Retrofit 2".3. December.2017. WEB.12.MAY.2018.](https://medium.com/@jintin/retrofit-2-b0c80d5caadf)

[HTTPS://MEDIUM.COM/@JINTIN/RETROFIT-2-B0C80D5CAADF](https://medium.com/@jintin/retrofit-2-b0c80d5caadf)

Abstract

Due to the multiplicity of specialized sites in e-commerce domain and shopping online and as an attempt to keep up with the developments in it. We developed an Android application to facilitate the remote buying process. Which is kind of a search engines that searches for products based on the user's desire then display them on organized and arranged form according to the price on one page without effort

Keywords: search engine. Compare prices. Android.

ملخص

نظرا لكثرة المواقع المختصة في مجال التجارة الالكترونية والتسوق عبر الانترنت ومحاولة لمواكبة التطورات الحاصلة فيه قمنا بتطوير تطبيق اندرويد يقوم بتسهيل عملية الشراء عن بعد حيث انه عبارة عن محرك بحث مصغر يعمل على البحث عن المنتجات استنادا لرغبة المستخدم وعرضها له منظمة ومرتبطة حسب السعر في صفحة واحدة بدون جهد او عناء
الكلمات المفتاحية: محرك بحث . مقارنة الاسعار . اندرويد.

Résumé

En raison de la multiplicité des sites spécialisés dans le domaine du commerce électronique et des achats en ligne et comme une tentative de suivre les développements dans ce domaine. Nous avons développé une application Android pour faciliter le processus d'achat à distance. ce qui est une sorte de moteur de recherche qui recherche des produits en fonction du désir de l'utilisateur, puis les affiche sous forme organisée et arrangée en fonction du prix sur une page sans effort

Mots-clés: moteur de recherche. Comparer les prix. Android.

