DESIGN AND IMPLEMENTATION OF E-COMMERCE APPLICATION WITH DISPLAY SYSTEM OF PRODUCTS IN SOCIAL NETWORKS

by

BOUTSAKIS MATTHAIOS

Bachelor degree: University of Crete, 2012

A THESIS

submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

DEPARTMENT OF APPLIED INFORMATICS
AND MULTIMEDIA

SCHOOL OF APPLIED TECHNOLOGY

TECHNOLOGICAL EDUCATIONAL INSTITUTE OF CRETE

·2019·

Approved by:

Major Professor
Costantinos Aivalis

Abstract

E-commerce applications are beneficial either for buying or selling products through Internet. Users from every place in the world can enter to an e-commerce application at any time and buy online products both quickly and efficiently. Users' purchases can be delivered to their place of residence or even buy them from available branches of the e-shop application. Due to these advantages, a store that makes such application can undoubtedly raise the number of its customers.

From the very first beginning of Internet history, many internet technologies were developed because they had as a purpose to make online activities and transactions as fast as possible. As a result, new web technologies were developed in order to make Internet faster and efficient for the users and also accessible to everyone. For this purpose, we focused on the growth of Rich Internet Applications. Such applications make most of the work on client's side, while they make all the transfers with the server side in such a transparent way that they give the impression to the end user that they are desktop applications.

The purpose of this thesis proves that this possibility of friendly applications can be extended to e-commerce applications too. So an e-commerce application will be designed and implemented and it will contain payment system. The customer can pay with the following ways: PayPal, pay on delivery, cash and deposit. It consists of two web applications: a customer interface system and a management system of products, customers and orders. The implementation will be done in Java language, with Google Web Toolkit which is a web development tool for Rich Internet Applications. System will also offer the possibility to the administrator of the e-commerce application to display selected products from the application through periodic posts to Facebook. The automated posts will take place in time frames that will be determined through a special interface of the application. Finally, it will include summary comparative research among RestFB, Spring Social Facebook and Facebook4J and it will choose the most suitable API for the implementation of the Facebook interface. MYSQL is used as a database.

Therefore, the basic questions that we want to answer with this thesis work are how well GWT can respond in the implementation of e-commerce application and how well Social Media can respond for the promotion of e-commerce products.

Table of Contents

List of Figures

List of Tables

Acknowledgents

Firstly, I would like to thank Mr. Costantinos Aivalis, professor of the Department of Applied Informatics and Multimedia of TEI Crete, not only for the assignment of this thesis but also for his undeniable help and support.

Additionally, I would like to thank my family members for their support and courage that they were constantly offering me in order to fulfill my master degree.

Finally, I would like to state how grateful I am for all the people who helped me to complete this thesis.

Chapter 1
Introduction

Internet

Internet is a huge world computer network that connects smaller and local computer networks to each other. Tim Berners-Lee, who is founder of the WWW Consortium, gives the following definition of the Internet: "It's a bit like a postcard with a simple address on it. If you put the right address on a packet, and give it to any computer which is connected as part of the Net, each computer would figure out which cable to send it down next so that it would get to its destination. That's what the Internet does. It delivers packets—anywhere in the world; normally well under a second."

"The ARPANET was the precursor to the Internet. Developed by the Department of Defense in conjunction with several universities; the goal of ARPANET was to allow university-based researchers working for the Defense Department to share information with their colleagues in other U.S. cities. That first 'login' message was sent from UCLA to Stanford; soon after, two more universities, UC Santa Barbara and University of Utah, joined the ARPANET system." [1]

"By 1971, there were about 30 universities participating in ARPANET. The public's first glimpse of the new computer network was in 1972, at the International Computer Communication Conference, where conferees latched onto a handy new application called electronic mail. The Internet, as we know it, was well on its way." [1]

"In the late 70s, Dr.Vinton Cerf was finalizing a common language for communicating between computers called TCP/IP (Transmission Control Protocol/Internet Protocol). Within the TCP/IP language, a protocol called FTP (file transfer protocol) became dominant, allowing users to access files on a remote computer. FTP allowed networks outside ARPANET to connect with the network initiated at UCLA as well as other networks beside ARPANET." [1]

"By 1984, the network had grown to include 1,000 host computers. The National Science Foundation was one of the first outside institutions hoping to connect to this body of information. Other government, non-profit, and educational institutions followed. Initial attempts to catalogue this rapidly expanding system of networks were simple. Among the first was

Archie, a list of FTP information created by Peter Deutsch at McGill University in Montreal. However, the greatest innovation in the Internet was still to come, brewing in an MIT laboratory in Cambridge, Mass. " [1]

"The World Wide Web, or the Web, is often confused with the Internet. In fact, it is just one part of the Internet, along with email, videoconferencing, and streaming audio channels." [1]

"In 1989, Tim Berners-Lee, now a scientist at the Massachusetts Institute of Technology, introduced a new system of communicating on the Internet which used hyperlinks and a user-friendly graphical interface. His slice of the Internet pie became known as the World Wide Web. " [1]

Tim Berners-Lee defines web as "an abstract (imaginary) space of information". He also says, "On the Net, you find computers—on the Web, you find documents, sounds, videos, information. On the Net, the connections are cables between computers; on the Web, connections are hypertext links. The Web exists because of programs which communicate between computers on the Net. The Web could not be without the Net. The Web made the net useful because people are really interested in information (not to mention knowledge and wisdom!) and don't really want to have knowledge about computers and cables. "[1]

Today, new web technologies grew in order to make Internet faster, more efficient and accessible to everyone. For this purpose, there was growth of Rich Internet Applications. Such kinds of applications make most of the work on client’s side, while they make all of the transfers with the server side in a transparent way.

Summary

One of the technologies which belongs to these new web technologies is Google Web Toolkit (GWT) and it was recommended by Google. Firstly, GWT offers all the necessary tools not only for implementing applications with absolutely friendly interfaces but also for the exploitation of all the possibilities that the Internet evolution provides both to programmers of web applications and unfamiliar users too. Secondly, GWT simplifies the implementation of web applications, which are designed to satisfy every requirement with the use of complete, powerful and efficient developed tools. In this thesis work, we present Web 2.0 (which is the result of the rapid growth of web applications and infrastructures), Rich Internet Applications that are a tendency for the satisfaction of the requirements of the expanded audience of the internet by the interfaces of applications and finally AJAX technology on which the core of GWT is based. Furthermore, we present the design and implementation of e-commerce application with payments system through credit cards and PayPal and also another application which contains an automated display system of products in social networks such as Facebook. Here we need to identify the most suitable API for the implementation of the Facebook interface among RestFB, Spring Social Facebook and Facebook4J. Finally, we present the conclusions that were drawn from the implementation of this application.

Motivation

Basic motivation for the implementation of this thesis work is the acceptance of having Rich Internet Applications and Social Media in web communities. Nowadays, the tendency regarding the development of applications is to create web application interfaces to communicate in a transparent way with servers and give the impression to the end user that they are desktop applications. As a result, different patterns and methodologies have been recommended in order to implement such applications. One of these applications is Google Web Toolkit that is based on Java program language. It’s a new approach but it has great development so it provides programmers with high possibilities which are documented from different sources. Social Media have been developed and have been used all over the world in current years and as a matter of fact many companies want to expand through them. Especially Facebook is used by many users and also by companies that want to expand and be advertised. So the basic questions that we want to answer with this thesis are how well GWT can be a successful tool in the implementation of an e-commerce application and how well Social Media can be successful in the promotion of e-commerce products.

Scope

This thesis aims to present the possibilities offered by the eclipse software development environment in combination with the Google Web Toolkit (GWT). For this purpose, the development of a web application, which was implemented using the advantages of GWT, is analyzed. Also, after a comparative research we choose the most suitable API for the implementation of the Facebook interface. Its purpose is to provide the administrator of the e-commerce application the possibility to display selected products from the application through periodic posts on Facebook.

Chapter 2

Rich Internet Applications

Web 2.0

The Web 2.0 is an evolution of the World Wide Web that occurred as an impact of the combination of the widespread use of the Internet by the majority of the worldwide population with the positive effects of the implementation of broadband in most countries of the universe. The core idea of Web 2.0 refers to the end users' perception that they interact with applications installed on their computers the moment when they handle web applications.

Firstly, we must define what Web 2.0 is. "Web 2.0 refers to the second generation of the Web, wherein interoperable, user-centered web applications and services promote social connectedness, media and information sharing, user-created content, and collaboration among individuals and organizations." [2]

Some of the benefits of Web 2.0 are the following:

- Acceleration
- Good User Interaction
- Simple and great way of publicizing things
- Cheap way of advertising things
- Several solutions for single problem
- Easy way to collaborate business
- Leveraging Business
- Easy Online Presence [3]

The main technologies that are included according to Mike Wolcott are: "

- **AJAX:** An acronym derived from 'Asynchronous JavaScript and XML'. AJAX is an important underlying technology used to create interactive Web applications. Ajax enables Web 2.0 sites to behave dynamically, so that they feel more like computer programs than static web pages. Like the way Google Maps lets you drag elements across the page? That's AJAX in action.

- **Atom:** A format for the syndication of online content, atom functions as a newer alternative to RSS (described below).

- **Blog:** It originally derived from the word ‘weblog’, a blog is a website of simple content created with inexpensive self-publishing tools. Blogs are the backbone of Web 2.0's democratic spirit.
- **Mash-ups:** Websites or applications that combine content from either one source or more.
- **RSS:** Shorthand for ‘Real Simple Syndication’, RSS is a protocol that enables computer users to receive content from their favorite providers whenever the content is updated. Instead of having to remember to visit a website in order to read a favorite column, you can watch a video, or even listen to an audio program. RSS lets a user subscribe to the content so it's delivered automatically. The flow of content that the user receives is called ‘RSS feed.’
- **Social media:** A generic term used to describe Web-based tools that harness the power of collaboration and group interaction. This can take many forms, from the personal web pages of MySpace to the virtual worlds of Second Life or even the professional networking of LinkedIn.
- **Tags:** User-generated keywords are used to describe online content. Tags benefit both humans and search engines by enabling them to find relevant and related information.
- **Wikis:** A dynamic Web document that allows users to add, change, or edit the content which is displayed on the page.
- **XML:** It is an abbreviation for ‘Extensible Markup Language’. XML is not only a programming code for online data that preserves the structure but it also formats a digital document regardless of which application is used to read it. XML is a significant enabling technology for RSS feeds." [4]

The use of Web 2.0 technologies has developed services that web applications offer since it has raised their response speed. Many companies like Google, Amazon, e-bay, Wikipedia, Yahoo etc. were pleased with the results of Rich Internet Technologies and they used them in their applications to improve the user experience and this is one of the reasons they attract so many users in their websites. [5]

Rich Internet Applications

Rich Internet Applications are considered as an approach of developing web applications with the aim of providing high level interoperability and support of all kinds of contents that can be transmitted through Internet. In the first place, we must define the meaning of Rich Internet Applications. "Rich Internet Applications (RIAs) are web applications, which use data that can be processed both by the server and the client. Furthermore, the data exchange takes place in an asynchronous way so that the client stays responsive while continuously recalculating or updating parts of the user interface. On the client, RIAs provide a similar look-and-feel as desktop applications and the word ‘rich’ means particularly the difference to the earlier generation of web applications. RIAs are basically characterized by a variety of interactive operating controls, the possibility of on-/offline use of the application, and the transparent usage of the client and server computing power and of the network connection."[6]

Traditional web applications have lots of limitations concerning usability and interactivity of their user interfaces and for this reason Rich Internet Applications (RIAs) came up to solve these limitations.[6] "This type of applications offers a series of advantages, e.g. they provide a richer and more efficient graphical interface resembling desktop applications. RIAs are complex web applications based on thick client architecture, asynchronous communication, and a great variety of user interface widgets. On the one hand, the client will manage data and processes; on the other hand, network traffic is reduced by the improved communication. Both innovations allow for additional user interface features increasing the usability and the interaction possibilities of the users.

The development of RIAs, which are more complex software than web applications, requires also a more complex development process. Requirements engineering has to cope with the variety of features that RIAs can offer to the users. The design of RIAs has to consider different types of architectures and communication and desktop-like user interfaces. Implementation of RIAs is based on the integration of specific libraries that support the different user interface features. "[6]

Apart from better user experience, processing user interactions, asynchronous communication and reduced network traffic, some other advantages of RIAs on desktop applications are the following: "

- simplicity of installation or even often no installation steps
- safety for users, because the service is less vulnerable to threats (malware)
- availability for users on any machine connected to Internet
- independency from the hardware platform or operating system
- easiness of updates and changes that are immediately available for all users. "[7]

"However, before deciding to build an application in form of RIA, their disadvantages must be pointed out and understood:

- requirements for additional software (plug–ins/RIA frameworks)
- lower speed than desktop applications, because code is interpreted and converted into a native code on client machine
- security restrictions, forbidding access to critical resources
- Rest fully on server and the quality of network connection". [7]

AJAX

AJAX is a collection of different technologies that are used for the development of web applications. AJAX is analyzed into "Asynchronous JavaScript and XML". We can create web sites by using dynamic features with AJAX. It also looks better and effective enough. It offers the possibility to update some part of the content of a webpage without forcing the webserver to request again and retrieve the webpage. [8] The data retrieved using AJAX mechanisms may be stored in a text file, an XML file or a database. Most of the time the retrieval of content demands to call the script of a dynamic web programming language. Technologies that AJAX contains are the following:

- HTML
- CSS
- JavaScript
- XML and XLST
- XMLHttpRequest Object
- Document Object Model (DOM)

The XMLHttpRequest is an object that JavaScript supports. It offers all the functions, features and events that are necessary for the complete operation of a process that is based on AJAX mechanisms. Firstly, it was created and used by Microsoft, then it was used many times by Google and now it is supported by W3C.

Here is an example of how XMLHttpRequest works:

```
"var request = new XMLHttpRequest()
request.open(‘GET’,url,true)
request.send(null)
request.onreadystatechange=stateChanged()
function stateChanged() {
        if (request.readyState == 4) {
           alert(request.responseText)
           }
        }" [9]
```

In the following figure we show the operation of AJAX mechanism compared to classic method of content request:

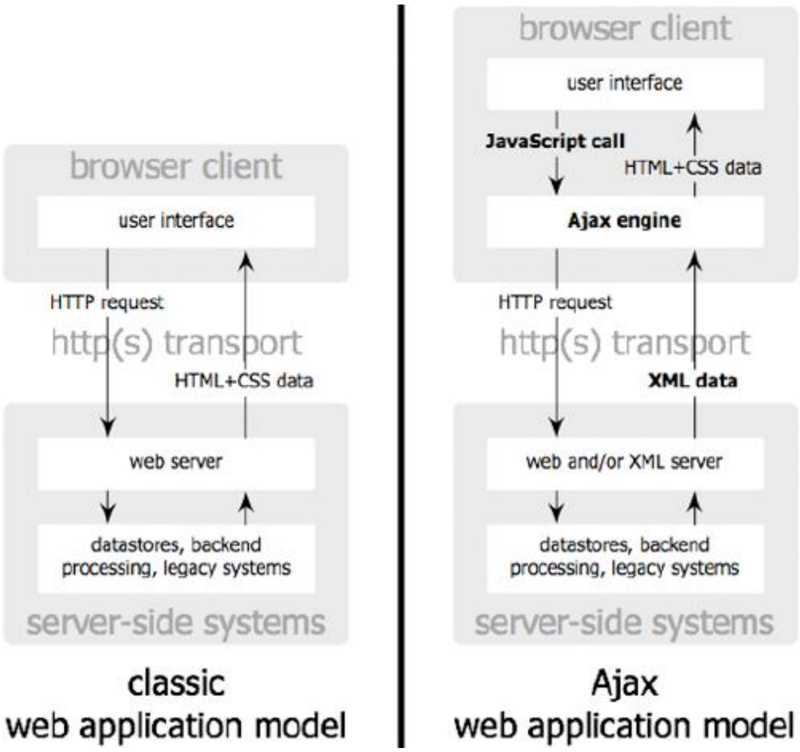


Figure 1: AJAX functionality

Chapter 3

Google Web Toolkit

What is GWT

GWT was presented by Google on May 18th, 2006 as a set of development tools, programming utilities, and widgets that let you create rich Internet applications in a completely different way that you had done with traditional internet applications before. [10]

GWT is based on Java. So it allows you to write your application in Java, and then the source is compiled to be highly optimized in JavaScript that runs across all major browsers. [11] Its main function is to create dynamic webpages from source code written in Java. It's an open code framework which is possible to develop applications that use AJAX technology and Java code to be transformed to HTML/JavaScript and is optimized concerning the possibility to be localized by search engines. Source code creates dynamic webpages with incorporated functions that are set to run in client side and the call of AJAX functions activates servlet calls. It provides programmer with a set of tools that supports synchronous or asynchronous communication, for handling XMLHttpRequest objects and transforming data in XML and JSON formats.

Incorporation of GWT technology in web applications fixes the following disadvantages of AJAX technology:

- It needs knowledge of at least two programming languages, JavaScript and one server side language (java, php, .net).
- Server call mechanism is difficult to be used
- JavaScript presents vulnerabilities in development of high scale applications

It has been used for the development of well-known webpages like Apache Wave, Google Base, Google Checkout, Google Moderator, Google Web, Google AdSense, gogrid.com, contactoffice.com, whirled.com etc.

The process of developing a GWT web application includes: "

- Module Descriptors: A module descriptor is a configuration file in the form of XML that is used to configure a GWT application
- Public Resources: These are the various files referenced by GWT module, such as Host HTML page, CSS or images. The location of these resources

can be configured using <public path=‘path’/> element in module configuration file. By default, it is the public subdirectory underneath where the Module XML File is stored. When the application is compiled into JavaScript, all the files that can be found on the public path are copied to the module's output directory.

- Client-side Code: This is the actual Java code which is written for implementing the business logic of the application. With this, GWT compiler translates into JavaScript, which will eventually run inside the browser.
- Server-side Code: This is the server side part of the application and it is optional very much. If we don’t do any backend processing within the application, then this part is not needed. However, if there is some processing required at backend and the client-side application interacts with the server, then these components will have to be developed. " [12]

In the following picture you can see how GWT works:

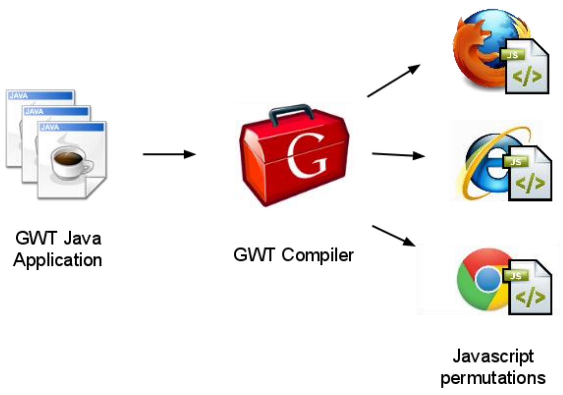


Figure 2: GWT functionality

In GWT the following parts are included: "

- GWT Java to JavaScript compiler: This is the most important part of GWT which makes it a powerful tool for building RIAs. GWT compiler is used to translate all the application code written in Java into JavaScript.
- JRE Emulation library: Google Web Toolkit includes a library that emulates a subset of the Java runtime library. The list includes java.lang, java.lang.annotation, java.math, java.io, java.sql, java.util and java.util.logging
- GWT UI building library: This part of GWT consists of many subparts which include the actual UI components, RPC support, History management, and many more.
- GWT Hosted Web Browser: It lets you run and execute GWT applications in hosted mode, where code runs as Java in the Java Virtual Machine without compiling to JavaScript. " [12]

GWT includes a set of libraries which can add high level functionality in webpages. These can be seen in the following table:

Table 1: GET packages

Name	Description
Com.google.http.client	It is used for the asynchronous communication with XMLHttpRequest
Com.google.gwt.user.client	It is used with the most famous browsers for the communication.
Com.google.gwt.user.client.ui	It offers possibilities for creation of graphical interfaces for the applications being developed.
Com.google.gwt.user.client.rpc	They are used for the RPC communication between client and server.
Com.google.gwt.user.server.rpc	They are used for the RPC communication between client and server.
Com.google.gwt.xml.client	It’s used for parsing from and to XML form.
Com.google.gwt.xml.json	It’s used for parsing from and to JSON form.
Com.google.gwt.juni.client	It’s used in code control.

Basic advantages of GWT are the following: "

- GWT provides easy integration with Junit and Maven.
- Being Java based, GWT has a low learning curve for Java Developers.
- GWT generates optimized JavaScript code, produces browser's specific JavaScript code by itself.
- GWT provides widgets library which supplies most of the tasks that are required in an application.
- GWT is extensible and custom widget based which can be created to cater to various application needs.
- GWT applications can run on all major browsers and smart phones including Android and iOS based phones/tablets. " [12]

Disadvantages of GWT are the following: "

- Not indexable: Web pages which are generated by GWT would not be indexed by search engines because these applications are generated dynamically.
- Not Degradable: If the application user disables JavaScript, then user will just see the basic page and nothing else.
- Not Designer's Friendly: GWT is not suitable for web designers who prefer using plain HTML with placeholders for inserting dynamic content at a later point in time. " [12]

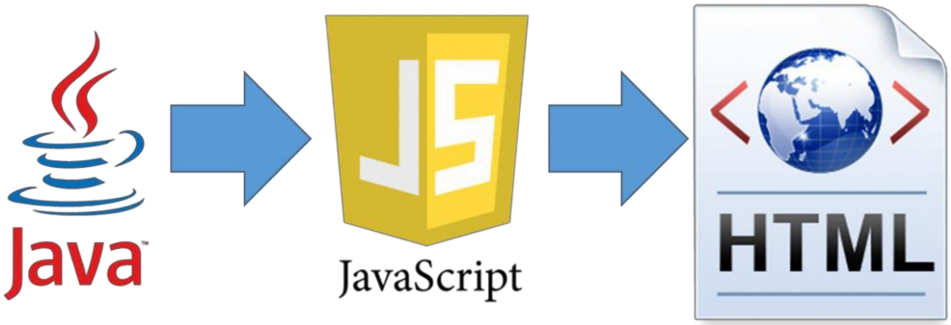


Figure 3: Process of developing applications with GWT

Widgets, Panels and Events

GWT contains a complete set of classes that describe interface elements and also events to them. It is possible to produce high level results in a transparent way for developers by using these interfaces of web applications. Interface elements are divided into two main categories:

- Panels: It is basically the substrate on which the interface elements will be placed and the users use them to interact with the application in order to operate. There is a wide variety of widgets of this form in a set that can respond to any interface layout requirement.
- Widgets: They consist of the communication channel between application and user. GWT has all those widgets that are possible on a webpage, accompanied by a complete set of properties.

Communication between application and user is accomplished with the creation of events that are triggered either by user actions or by responses of the application. The events that GWT supports are also capable of responding to any action that may be required during operation of a web application. These actions are detected with Listener Events which are Java objects that give the behavior for the detection and reaction to events, in the object that is assigned. Thus, a corresponding Event Listener needs to be applied for every widget in which the events that are applied lead to a corresponding change of the application behavior.

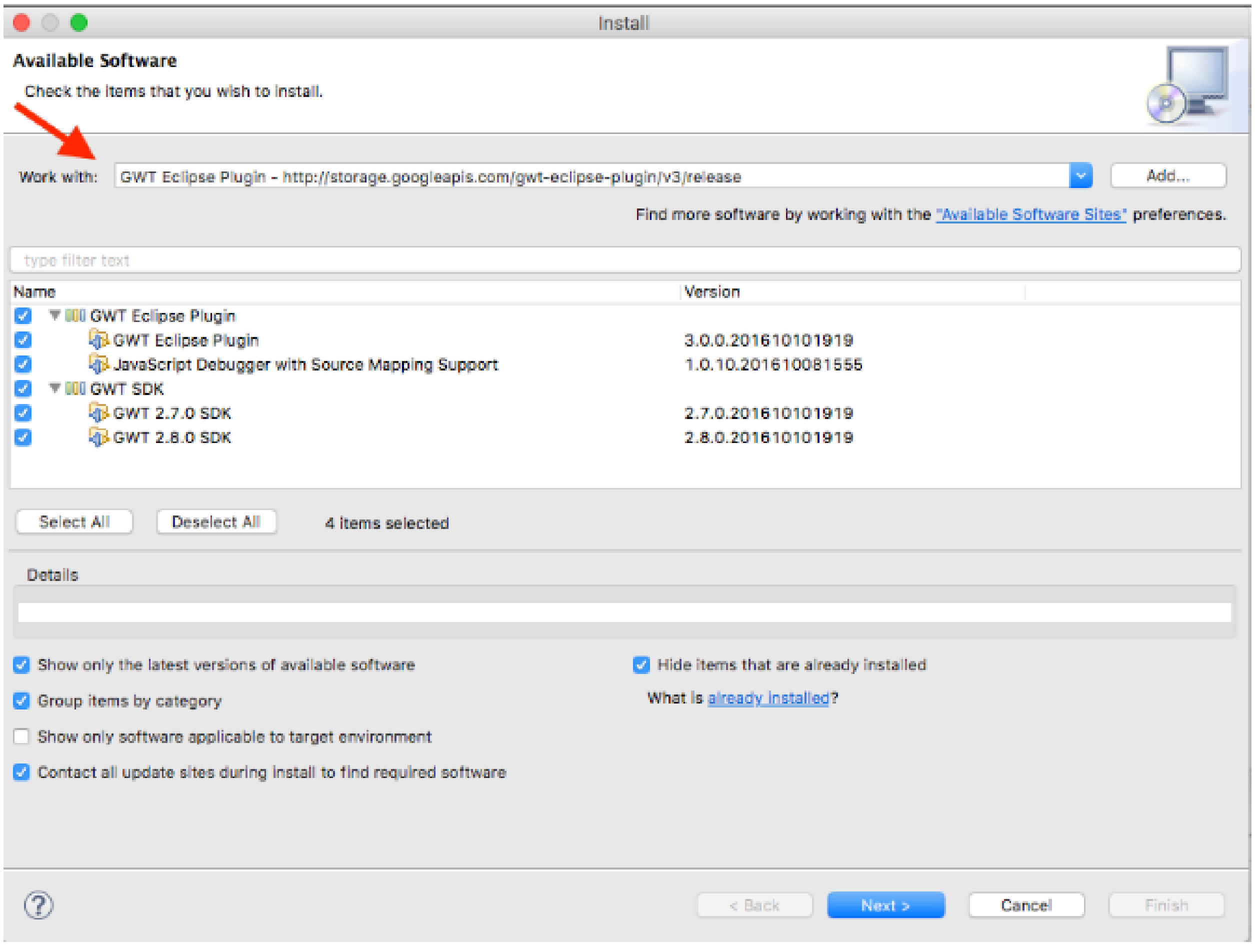
Client-Server Communication

Communication between client and server in applications which are being developed with GWT is fulfilled with the use of AJAX technology. The use of AJAX can be accomplished mainly with two ways. In the first mode, an object named RequestBuilder is used and it installs a connection by sending an asynchronous service request, specifying the HTTP method to be used (GET or POST). The call of the sending method is accompanied by a parameter object RequestCallback, which determines how the received response will be faced and it either results from an integrated process by the server or it comes by an error (onResponseReceived () and onError () methods respectively). However, the second technique is based on RPC calls. A Java Interface that extends the class Remote Service needs to be defined and it contains all the methods of the RPC service. Then a class must be created for the implementation of this interface and that extends class RemoteServiceServlet. Simultaneously an interface, which is used on client side for calling service methods, must be defined and it will include an additional argument for maintaining response (AsyncCallback type). As a result, the procedure is executed with the creation of an RPC service, the development of an AsynchCallback object which handles the result and finally the call of the method which determines the reaction to the received response. This method was used in this application for the communication with the database.

GWT installation to eclipse

Appropriate plugins which are adapted in the Eclipse applications development platform are provided in order to use the possibilities of GWT. Their installation is done through a relatively simple process. This process ends with their configuration which must be done carefully. The process is the following:

The hyperlink "Install the GWT Eclipse Plugin" is selected in link address https://cloud.google.com/eclipse/docs/migrating-gpe and then we click on "GWT Eclipse Plugin Download and Install". After we press the install button to download GWT plugin. In the form the user needs to register the website from which the files that consist the plugins will be retrieved. For the development of this application, installation was done from the website http://gwt-plugins.github.io/documentation/gwt-eclipse-plugin/Download.html [13]. So "Install Now" button was clicked and the following link was registered in the displayed window.



Available Software

Check the items that you wish to install.

Work with: GWT Eclipse Plugin - http://storage.googleapis.com/gwt-eclipse-plugin/v3/release

Find more software by working with the "Available Software Sites" preferences.

Name	Version
☑ GWT Eclipse Plugin	
☑ GWT Eclipse Plugin	3.0.0.201610101919
☑ JavaScript Debugger with Source Mapping Support	1.0.10.201610081555
☑ GWT SDK	
☑ GWT 2.7.0 SDK	2.7.0.201610101919
☑ GWT 2.8.0 SDK	2.8.0.201610101919

4 items selected

Details

☑ Show only the latest versions of available software

☑ Group items by category

□ Show only software applicable to target environment

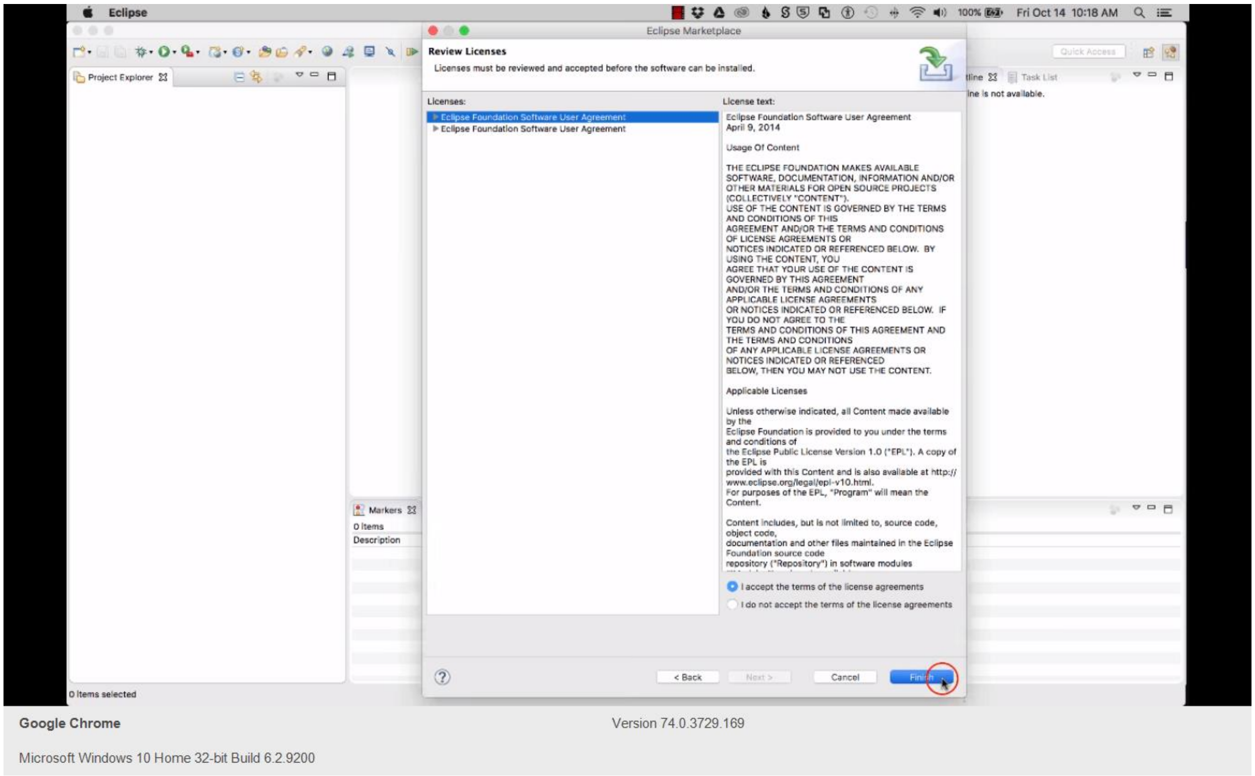
☑ Contact all update sites during install to find required software

☑ Hide items that are already installed

What is already installed?

Figure 4: Install plugin on eclipse (1)

The required component here is "Google plugin for eclipse". The other features are optional but if we want, we can also click them. Then, we review the features we want to install and we click "Next". Then we read the license agreements and then we select "I accept the terms of the license agreements" as shown in the picture below.



Review Licenses

Licenses must be reviewed and accepted before the software can be installed.

Licenses:

- Eclipse Foundation Software User Agreement
- Eclipse Foundation Software User Agreement

License text:

Eclipse Foundation Software User Agreement
April 9, 2014

Usage Of Content

THE ECLIPSE FOUNDATION MAKES AVAILABLE SOFTWARE, DOCUMENTATION, INFORMATION AND/OR OTHER MATERIALS FOR OPEN SOURCE PROJECTS (COLLECTIVELY "CONTENT"). USE OF THE CONTENT IS GOVERNED BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. BY USING THE CONTENT, YOU AGREE THAT YOUR USE OF THE CONTENT IS GOVERNED BY THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT AND THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW, THEN YOU MAY NOT USE THE CONTENT.

Applicable Licenses

Unless otherwise indicated, all Content made available by the Eclipse Foundation is provided to you under the terms and conditions of the Eclipse Public License Version 1.0 ("EPL"). A copy of the EPL is provided with this Content and is also available at http://www.eclipse.org/legal/epl-v10.html. For purposes of the EPL, "Program" will mean the Content.

Content includes, but is not limited to, source code, object code, documentation and other files maintained in the Eclipse Foundation source code repository ("Repository") in software modules

● I accept the terms of the license agreements

○ I do not accept the terms of the license agreements

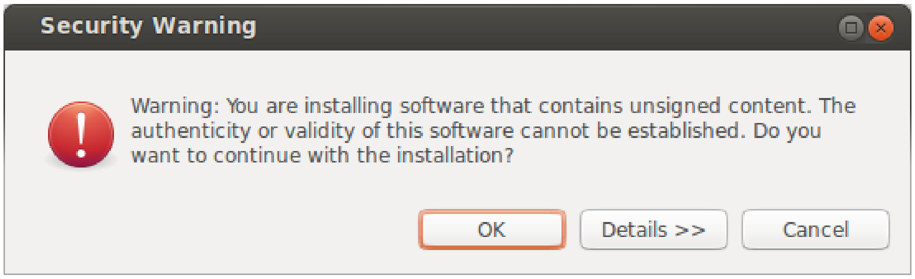
Google Chrome

Version 74.0.3729.169

Microsoft Windows 10 Home 32-bit Build 6.2.9200

Figure 5: Install plugin on eclipse (2)

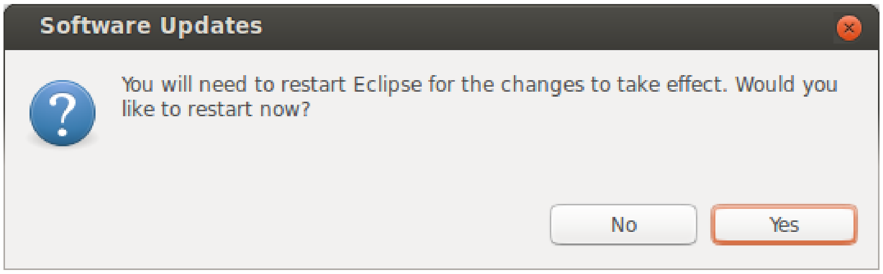
Then we click on the security warning.



Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?

Figure 6: Install plugin on eclipse (3)

Finally we will be asked if we would like to restart eclipse. We click "Restart Now".



Software Updates

You will need to restart Eclipse for the changes to take effect. Would you like to restart now?

Figure 7: Install plugin on eclipse (4)

After this, we will be ready to build App Engine Web Applications.

Chapter 4

Design of the application

Determination of requirements

Non-functional requirements

Application must be accessible through the network.

- Functionalities of the application should be completed with low complexity interfaces since it will also be used by users who are not familiar with computers.
- Response of the system must be immediate to the users' actions
- The storage of data which is managed by the application should be done with safety.
- Users should not be free since it can be possible to enter free text to the interfaces.

Functional requirements

The users who will use the application will be:

- Guests
- Customers
- Employee

Every category of users will do the following functionalities:

- Guests
  - Sign up
  - View information data of the company
  - View product list
  - View categories list
  - View product details
  - Product purchase
  - View evolution of order

    - Show payment methods
    - View cart
- Customers
    - Login
    - View information about the company
    - View product list
    - View categories list
    - View product details
    - Product purchase
    - View evolution of order
    - Show payment methods
    - View cart
    - Logout
- Employee
    - Login
    - Administrate company’s information
    - Creation / Process product
    - Creation/Process category
    - Orders view
    - Process of evolution order
    - Logout
    - Users process

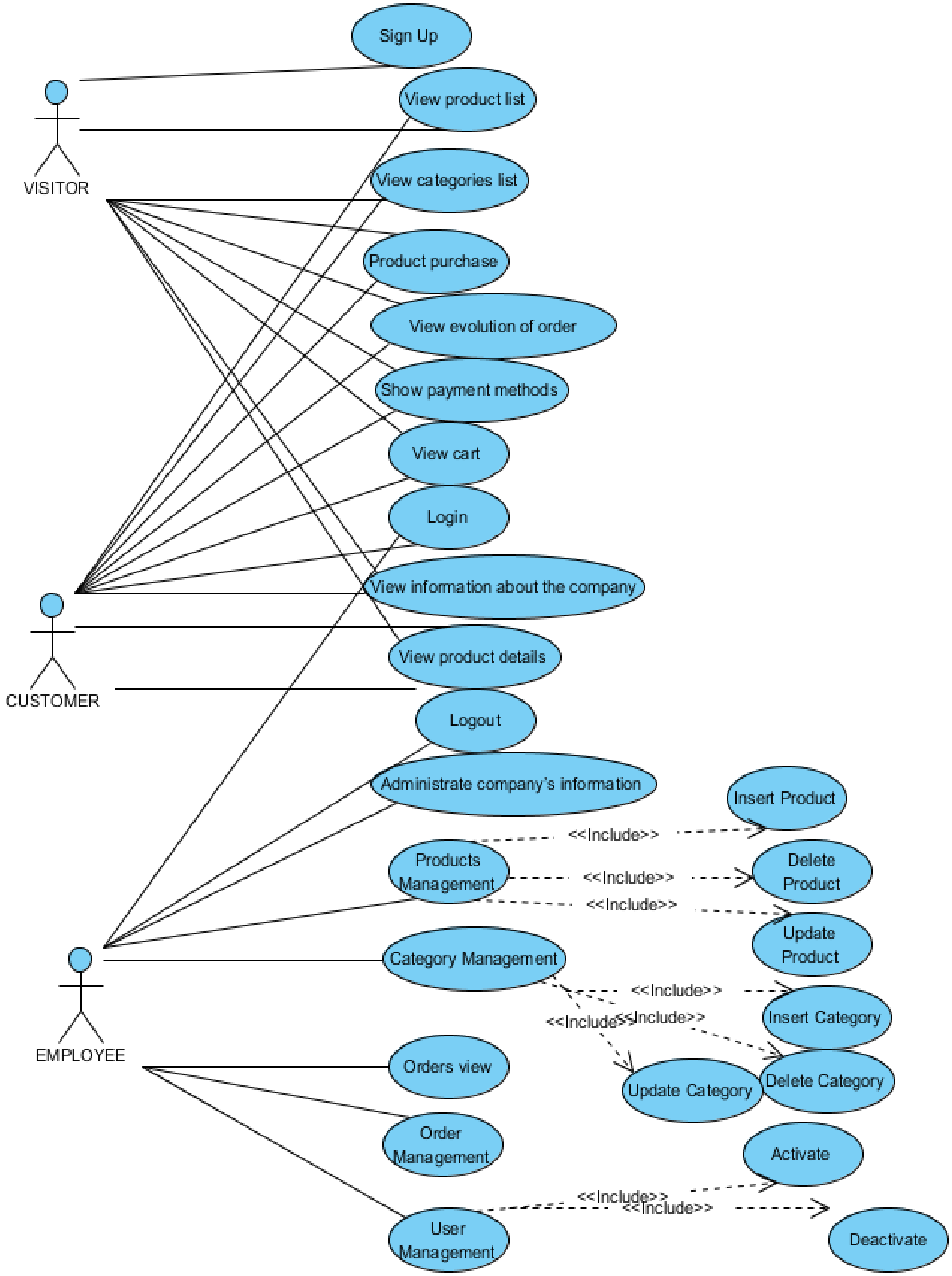


Figure 8: Use case diagram

Database Design

The solution of the relational database will be chosen for storing application data. Database is an organized collection of associated data that are used by the functions of one or more applications. The use of a database for the storing information is characterized by data universality, their division in such a way that ensures storage of redundant data as least times as possible, their availability from all users and finally the processes according to the policy access that is applied. Use of database has the following advantages : "

- **Improved data security**: The more users access the data, the greater the risk of data security breaches. Corporations invest considerable amounts of time, effort, and money to ensure that corporate data are used properly. A DBMS provides a framework for better enforcement of data privacy and security policies.
- **Improved data sharing:** DBMS helps create an environment in which end users have better access to more and better-managed data. Such access makes it possible for end users to respond quickly to changes in their environment.
- **Better data integration:** Wider access to well-managed data promotes an integrated view of the organization’s operations and a clearer view of the big picture. It becomes much easier to see how actions in one segment of the company affect other segments.
- **Minimized data inconsistency:** Data inconsistency exists when different versions of the same data appear in different places. The probability of data inconsistency is greatly reduced in a properly designed database.
- **Improved data access:** DBMS makes it possible to produce quick answers to ad hoc queries. From a database perspective, a query is a specific request issued to the DBMS for data manipulation—for example, to read or update the data. Simply put, a query is a question, and an ad hoc query is a spur-of-the-moment question. The DBMS sends back an answer (called the query result set) to the application.
- **Improved decision making:** Better-managed data and improved data access make it possible to generate better-quality information, on which better decisions are based. The quality of the information generated depends on the quality of the underlying data. Data quality is a comprehensive approach to promoting the

accuracy, validity, and timeliness of the data. While the DBMS does not guarantee data quality, it provides a framework to facilitate data quality initiatives.

- **Increased end-user productivity:** The availability of data, combined with the tools that transform data into usable information, empowers end users to make quick, informed decisions that can make the difference between success and failure in the global economy. " [14]

Disadvantages of database are the following: "

- **Increased costs:** Database systems require sophisticated hardware and software and highly skilled personnel. The cost of maintaining the hardware, software, and personnel required to operate and manage a database system can be substantial. Training, licensing, and regulation compliance costs are often overlooked when database systems are implemented.
- **Management complexity:** Database systems interface with many different technologies and have a significant impact on a company's resources and culture. The changes introduced by the adoption of a database system must be properly managed to ensure that they help the company's objectives advance. Given the fact that database systems hold crucial company data that are accessed from multiple sources, security issues must be assessed constantly.
- **Maintaining currency:** To maximize the efficiency of the database system, you must keep your system current. Therefore, you must perform frequent updates and apply the latest patches and security measures to all components. Due to the fact that database technology advances rapidly, personnel training costs tend to be significant. Vendor dependence. Given the heavy investment in technology and personnel training, companies might be reluctant to change database vendors. As a consequence, vendors are less likely to offer pricing point advantages to existing customers, and those customers might be limited in their choice of database system components.
- **Frequent upgrade/replacement cycles:** DBMS vendors frequently upgrade their products by adding new functionality. Such new features often come bundled in new upgrade versions of the software. Some of these versions

> require hardware upgrades. Not only do the upgrades themselves cost money, but it also costs money to train database users and administrators to use and manage the new features properly. " [14]

As we can see from the presentation of the advantages and disadvantages of databases, their use seems to be advantageous for the operation of applications that support any kind of information systems. The greater the importance of the stored data is, the greater the tendency for their use is. Their recent development has helped available database implementations exist. These implementations cover all possible requirements and meet the cost budget constraints. As a consequence, almost all web applications are supported by relational database management systems.

The design of the relational database includes the development of the conceptual model, the development of the relational model and also the normalization of the model. The design process is included in the following paragraphs.

Conceptual Model

The entities that participate in the system are the following:

Table 2: User Entity

USER: User of the e-shop	
User id	Unique code of the user
First Name	Name of the user
Surname	Surname of the user
Address	Address of the user
Postal Code	Postal Code of the user
City	City of stay of the user
Mobile number	Mobile phone number of the user
Home number	Home phone number of the user
Email	E-mail of the user that is unique
Username	Unique username of the user
Password	Password of the user
Payment Type	Payment Type of the user
User Type (Administrator or user)	User Type of the e-shop account
Creation date	Creation date of the account

Table 3: Product Entity

Product: Product of the e-shop	
Product id	Unique code of the product
Description	Description of the product
Constructor	Constructor of the product
Product Category	Category that the product belongs
Image	Image of the product
Price	Price of the product
SEO (Search Engine Optimization	Keywords used in search engine to find the product
Unit in stock	How many units of product are available
Availability	Check if the product is available or not
Creation date	Date when the product is created
Date modified	Date when it was modified

Table 4: Order Entity

Order: Order made by a customer	
Order id	Unique order code of the customer
Order date	Order date of the customer
Branch receipt	Branch that the customer received his order
Payment Type	Payment Type of the order
Evolution order step	In which step is customer’s order (waiting or received)
Customer	The customer who makes the order

Table 5: Branch Entity

Branch: Branches of the e-shop	
Branch id	Unique code of the branch
Address	Address of the branch
Postal Code	Postal Code of the branch
City	City of the branch
Telephone number	Phone number of the branch
Creation date	Date that the branch was created

Table 6: Product Orders Entity

Product Orders: Product Orders by customers	
Order Id	Id of the order
Product Id	Id of the product
Quantity	Quantity of the product
Price	Price of the product

Table 7: Payment type entity

Payment Type: The way that customer pays	
Type Id	Unique id of the payment type
Payment Type	The type in which customer pays
Payment Account	Payment Account of the customer

Table 8: Category entity

Category: Category of products	
Category Id	Unique id of the category
Name	Name of the category
Description	Description of the category

Relations between entities are described in the following table:

Table 9: Entities Relations

Entities	**Description of relation**
USER ORDER	An order can be done by a user and an order can belong only to one user
ORDER BRANCH	A purchase can be done only from a branch while from a branch can be done many purchases.
ORDER PAYMENT TYPE	An order can have one payment type while a payment type can have many orders
USER PAYMENT TYPE	A user can have one payment type while a payment type can have many users
PRODUCT CATEGORY	A product can belong to a category while in a category can belong many products
ORDER PRODUCT ORDERS	An order can have many products that are ordered while a product that is ordered can belong to many orders
PRODUCT PRODUCT ORDERS	A product can belong to many orders

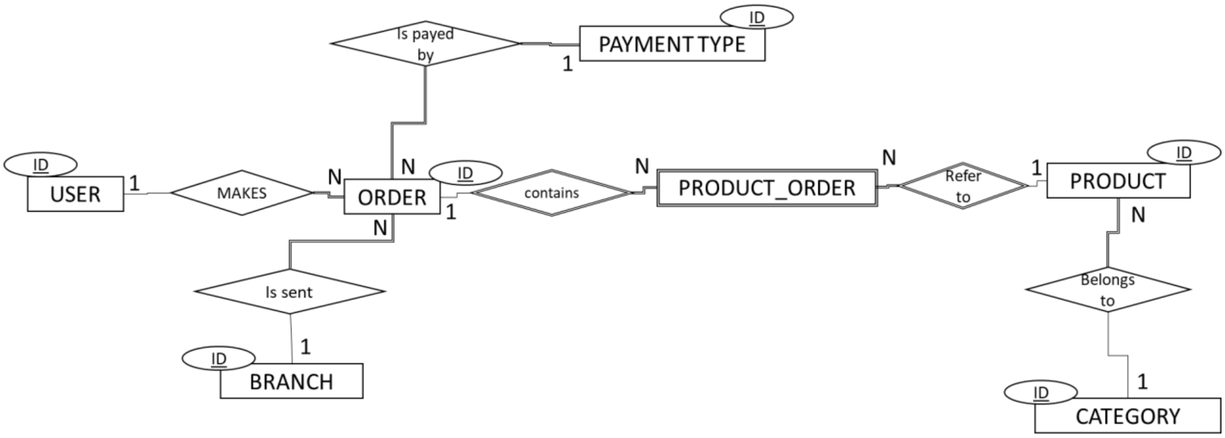


Figure 9: Entity-Relationship diagram

Relational Model

From the conceptual model occurs the following relational model:

Table 10: User Table

USER: User of the e-shop	
Id	int(11) unique
First Name	varchar(50)
Surname	varchar(50)
Address	varchar(100)
Postal Code	varchar(100)
City	varchar(100)
Mobile number	varchar(20)
Home number	varchar(20)
Email	varchar(150) unique
Username	varchar(20) unique
Password	varchar(500)
Payment Type	int(100) (Foreign Key in Customer Payment Type table)
User Type (IsAdmin)	tinyint(1)
Creation date	date

Table 11: Product Entity

Product: Product of the e-shop	
Id	int(11) unique
Title	varchar(100)
Description	varchar(100)
Product Category	int(11) (foreign key in category table)
Image	varchar(100)
Price	double(10,2)

SEO (Search Engine Optimization)	varchar(50)
Unit in stock	int(100)
Availability	tinyint(1)
Creation date	date
Date modified	date

Table 12: Order Entity

Order: Order made by a customer	
Id	int(11) unique
date	date
Branch receipt	int(100) (Foreign Key in Branch table)
Payment Type	int(50) (Foreign Key in Customer Payment Type table)
order step	varchar(50)
customer	int(100) (Foreign Key in User table)

Table 13: Branch Entity

Branch: Branches of the e-shop	
Id	int(11) unique
Address	varchar(100)
Postal Code	varchar(100)
City	varchar(100)
Telephone number	varchar(20)
Creation date	date

Table 14: Product Orders Entity

Product Orders: Product Orders by customers	
Order Id	int(11) unique
Product Id	int(11) (unique and foreign key in product table)
Quantity	Int(100)

Table 15: Payment type Entity

Payment Type: The way that customer pays	
Id	int(11) unique
Payment Type	varchar(50)
Payment Account	varchar(50)

Table 16: Category Entity

Category: Category of products	
Id	int(11) unique
Name	varchar(50)
Description	varchar(150)

In the next picture we can see the relational database model:

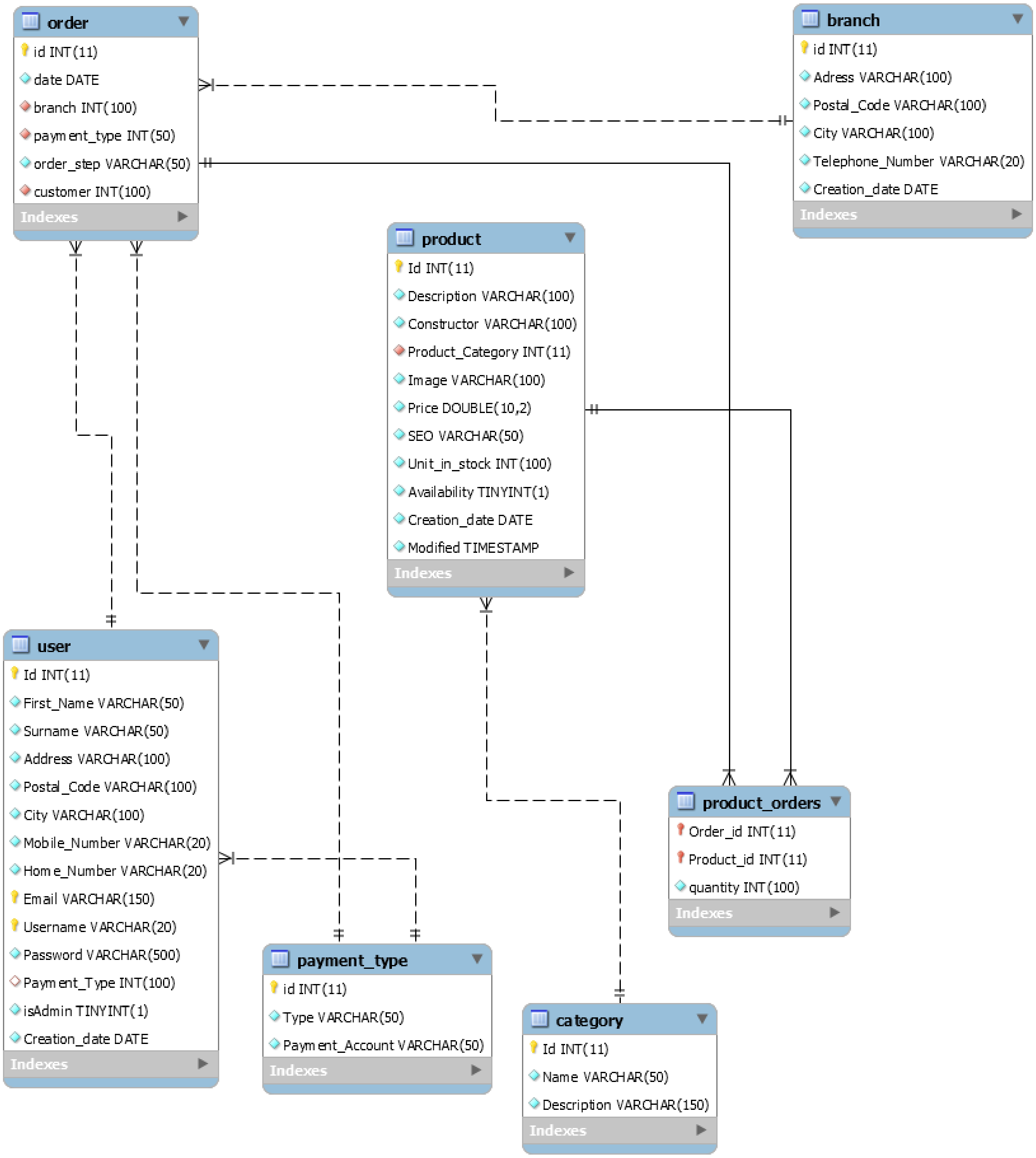


Figure 10: Relational Database Model

Structure of classes

In the structure of this applications we have the following object classes that participate:

Table 17: User Class

USER: User of the e-shop	
Id	int(11) Primary Key
First Name	varchar(50)
Surname	varchar(50)
Address	varchar(100)
Postal Code	varchar(100)
City	varchar(100)
Mobile number	varchar(20)
Home number	varchar(20)
Email	varchar(150) unique
Username	varchar(20) unique
Password	varchar(500)
Payment Type	int(100) (Foreign Key in Customer Payment Type table)
User Type	tinyint(1)
Creation date	date
Methods	
User	Constructor who creates an object of User type
setCustomerId(int customerId):void	Assigns value customerId to the CustomerId method
getCustomerId():int	Returns the value of the CustomerId method
setFirstname(String firstname):void	Assigns value firstname to the Firstname method
getFirstname():String	Returns the value of the Firstname method
setSurname(String surname):void	Assigns value surname to the Surname method
getSurname():String	Returns the value of the Surname method
setAddress(String address):void	Assigns value address to the Address method
getAddress():String	Returns the value of the Address method
setPostalCode(String postalCode):void	Assigns value postalCode to the PostalCode method
getPostalCode():String	Returns the value of the PostalCode method
setCity(String city):void	Assigns value city to the City method
getCity():String	Returns the value of the City method
setMobileNumber (String mobileNumber):void	Assigns value mobileNumber to the MobileNumber method
getMobileNumber(): String	Returns the value of the MobileNumber method

setHomeNumber(String homeNumber):void	Assigns value homeNumber to the HomeNumber method
getHomeNumber ():String	Returns the value of the HomeNumber method
setEmail(String email):void	Assigns value email to the Email method
getEmail():String	Returns the value of the Email method
setUsername(String username):void	Assigns value username to the Username method
getUsername ():String	Returns the value of the Username method
setPassword(String password):void	Assigns value password to the Password method
getPassword ():String	Returns the value of the Password method
setPaymentType (PaymentType paymentType):void	Assigns value paymentType to the PaymentType method
getPaymentType(): PaymentType	Returns the value of the PaymentType method
setUserType(String userType):void	Assigns value userType to the UserType method
getUserType():string	Returns the value of the UserType method
setCreationDate(String creationDate):void	Assigns value creationDate to the CreationDate method
getCreationDate():string	Returns the value of the CreationDate method

Table 18: Product Class

Product: Product of the e-shop	
Id	int(11) Primary Key
Title	varchar(100)
Description	varchar(100)
Product Category	int(11) (foreign key in category table)
Image	Varchar(100)
Price	double(10,2)
SEO (Search Engine Optimization)	varchar(50)
Unit in stock	int(100)
Availability	tinyint(1)
Creation date	date
Date modified	date
Methods	
Product	Constructor that creates an object of Product type
setProductId(int productId):void	Assigns value productId to the ProductId method
getId():int	Returns the value of the ProductId method

setTitle(String title):void	Assigns value title to the Title method
getTitle(): String	Returns the value of the Title method
setDescription(String description):void	Assigns value description to the Description method
getDescription(): String	Returns the value of the Description method
setProductCategory (Category productCategory): void	Assigns value productCategory to the ProductCategory method
getProductCategory(): Category	Returns the value of the ProductCategory method
setImage(String image):void	Assigns value image to Image method
getImage():String	Returns the value of the Image method
setPrice(double price):void	Assigns value price to Price method
getPrice():double	Returns the value of the Price method
setSEO(String SEO):void	Assigns value SEO to the SEO method
getSEO ():String	Returns the value of the SEO method
setUnitsInStock(int unitsInStock):void	Assigns value unitsInStock to the UnitsInStock method
getUnitsInStock ():int	Returns the value of the UnitsInStock method
setAvailability(boolean availability):void	Assigns value availability to the Availability method
getAvailability():boolean	Returns the value of the Availability method
setCreationDate(String creationDate):void	Assigns value creationDate to the CreationDate method
getCreationDate():string	Returns the value of the CreationDate method
setModified(String modified):void	Assigns value modified to the Modified method
getModified():String	Returns the value of the Modified method

Table 19: Order Class

Order: Order made by a customer	
Id	int(11) Primary Key
date	date
Branch	int(100) (Foreign Key in Branch table)
Payment Type	int(50) (Foreign Key in Customer Payment Type table)
Order step	varchar(50)
Customer	int(100) (Foreign Key in User table)
Methods	

Order	Constructor that creates an object of Order type
setOrderId(int orderId):void	Assigns value orderId to the OrderId method
getOrderId():int	Returns the value of the OrderId method
setOrderDate(String orderDate):void	Assigns value orderDate to the OrderDate method
getOrderDate():string	Returns the value of the OrderDate method
setBranchReceipt (Branch branchReceipt):void	Assigns value branchReceipt to BranchReceipt method
getBranchReceipt(): Branch	Returns the value of the BranchReceipt method
setPaymentType (PaymentType paymentType):void	Assigns value paymentType to PaymentType method
getPaymentType(): PaymentType	Returns the value of the PaymentType method
setEvolutionOrder Step(string evolutionOrderStep): void	Assigns value evolutionOrderStep to EvolutionOrder Step method
getEvolutionOrder Step ():string	Returns the value of the EvolutionOrder Step method
setCustomer (Customer customer):void	Assigns value customer to Customer method
getCustomer(): Customer	Returns the value of the Customer method

Table 20: Branch Class

Branch: Branches of the e-shop	
Id	int(11) Primary Key
Address	varchar(100)
Postal Code	varchar(100)
City	varchar(100)
Telephone number	varchar(20)
Creation date	date
Methods	
Branch	Constructor that creates an object of Branch type
setBranchId(int branchId):void	Assigns value branchId to the BranchId method
getBranchId():int	Returns the value of the BranchId method
setAddress(String address):void	Assigns value address to the Address method
getAddress():String	Returns the value of the Address method

setPostalCode(String postalCode):void	Assigns value postalCide to the PostalCode method
getPostalCode():String	Returns the value of the PostalCode method
setCity(String city):void	Assigns value city to the City method
getCity():String	Returns the value of the City method
setTelephoneNumber (String telephoneNumber): void	Assigns value telephoneNumber to the TelephoneNumber method
getTelephoneNumber() :String	Returns the value of the TelephoneNumber method
setCreationDate(String creationDate):void	Assigns value creationDate to the CreationDate method
getCreationDate(): String	Returns the value of the CreationDate method

Table 21: Payment type class

Payment Type: The way that customer pays	
Type Id	int(11) Primary Key
Payment Type	varchar(50)
Payment Account	varchar(50)
Methods	
Payment Type	Constructor that creates an object of Customer Payment type
setTypeId(int typeId):void	Assigns value typeId to the TypeId method
getTypeId():int	Returns the value of the TypeId method
setPaymentType(string paymentType):void	Assigns value paymentType to PaymentType method
getPaymentType(): string	Returns the value of the PaymentType method
setPaymentAccount(string PaymentAccount):void	Assigns value PaymentAccount to PaymentAccount method
getPaymentAccount() :string	Returns the value of the PaymentAccount method

Table 22: Category class

Category: Category of products	
Id	int(11) Primary Key
Name	varchar(50)
Description	Varchar(50)
Methods	

Category	Constructor that creates an object of Category
setCategoryId(int categoryId):void	Assigns value categoryId to the CategoryId method
getCategoryId():int	Returns the value of the CategoryId method
setName(string name):void	Assigns value name to the name method
getName():string	Returns the value of the name method
setDescription(string description):void	Assigns value description to the description method
getDescription(): string	Returns the value of the Description method

Table 23: Quantity class

Category: Category of products	
Methods	
Item	Constructor that creates an object of Item
Item(product p, int q):void	Assigns values p and q (quantity of product) to the Item method

Table 24: Database class

Connection with database (MySQLConnection): Class for the database connection	
Variables	
Link for the database (url) String	Server link in which the database runs
Name (db) String	Name of the database
Driver url String	Url that is used for the connection with database
Username String	Username for the connection to database
Password String	Password for the connection to database
Connection (conn) Connection	Connection to database
Methods	
Connect()	Constructor.It creates an object of Database type
Disconnect()	Constructor. It creates and object of Database type
ArrayList<branch> getBranches()	Retrieves and returns from database the list of all branches
ArrayList<Item> getItems (int x)	Retrieves and returns from database the list of all items
ArrayList<Category > getCategories()	Retrieves and returns from database the list of all categories
ArrayList<Product> getProducts()	Retrieves and returns from database the list of all products
ArrayList<Order> getOrders()	Retrieves and returns from database the list of all orders

ArrayList<Order> getOrders(String from,String until)	Retrieves and returns from database the list of all orders with the specified criteria
ArrayList<Order> getOrders(String from,String until, User attribute)	Retrieves and returns from database the list of all orders with the specified criteria
ArrayList <Customer> getCustomers()	Retrieves and returns from database the list of all customers

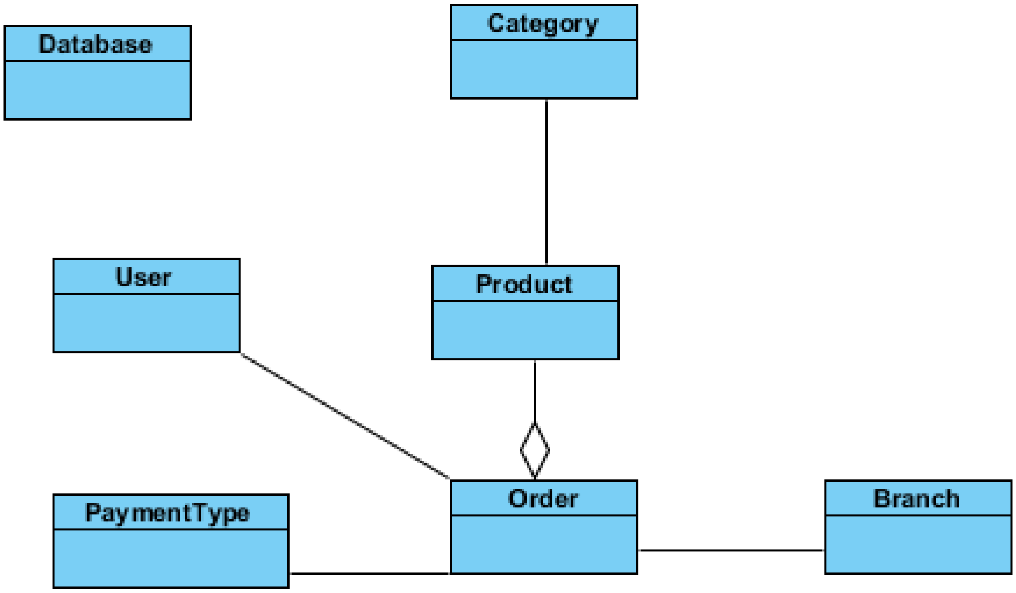


Figure 11: Database diagram

Table 25: Greeting Service Methods

Greeting Service methods	
String add2Cart(String addrequest)	It makes a request to add a product to cart and it adds the product to cart
String getCart()	It returns the products that are added to cart
String removeFromCart()	It removes all the products from cart
String logout()	It logouts a user from the system
String createCategory(String name,String description)	It creates a category that contains a name and a description
String upDateCategory(String name,String description)	It updates a category that contains a name and a description

String deleteCategory(String name,String description)	It deletes a category
String getAllCategories(String x)	It Retrieves and returns from database the list of all categories
String login(String Username, String Password)	It validates that the credentials of the parameters correspond to a user of the system and returns back that user.
String createCustomer(String firstname,string surname,string address,string postalCode,string city,string mobileNumber,string homeNumber,string email,string username,string password,string paymentType,string paymentDetails)	It registers a new customer with elements of the parameters.
String upDateCustomer(String firstname,string surname,string address,string postalCode,string city,string mobileNumber,string homeNumber,string email,string username,string password,string paymentType,string paymentDetails)	It updates a customer with these parameters.
String deleteCustomer(int customerId)	It deletes a customer.
String getAllCustomers(String x)	It retrieves and returns from database the list of all customers
String getThisCustomer(int customerId)	It retrieves and returns from database the customer with the given Id
String getCurrentCustomers(int x)	It retrieves and returns the given number of customers from database
String createProduct(String Title,String Description,int categoryId,String image,double Price,string SEO,int UnitsInStock,bolean Availability,string creationString,string Modified)	It creates a new product with these elements of the parameters

String updateProduct(String Title,String Description,int categoryId,String image,double Price,string SEO,int UnitsInStock,bolean Availability,string creationString,string Modified,int productId)	It updates the product with these elements of the parameters.
String getAllProducts(String x)	It retrieves and returns from database the list of all products.
String deleteProduct(int productId)	It deletes a product with the given Id.
String getAllBranches(String x)	It retrieves and returns from database the list of all branches.
String getThisBranch(int branchId)	It retrieves and returns from database the branch with the given Id.
String getAllProducts0(String x,int y)	It retrieves and returns from database the list of all products in the selected category.
String getThisProduct(int productId)	It retrieves and returns from database the product with the given Id.
String searchProduct(int categoryId,string term)	It retrieves and returns the product with in the selected category putting a term for search.
String searchOrders(String from,String until)	It retrieves and returns the orders for an administrator with these dates of the parameters.
String searchClientOrders(String from,String until)	It retrieves and returns the orders for a client with these dates of the parameters.
String insertOrder(int branch)	It inserts an order to a specified branch.
String upDateOrder(int orderId,String status)	It updates an order with the selected Id and the selected status.
String fbPublish(String file,String message,String link,String filename)	It’s created the Facebook publication mechanism with the given parameters.
String fbPublishOptions(String pageId,String pageName,String AccessToken,String period)	It returns the options that the administrator has put for the mechanism to work.

Functionalities Design

Functionalities that are included in the application are the following:

Login

A form is displayed to the user for entering his username and password. When these are submitted, they are transferred to the server and are checked for validation. In case that credentials are correct, user's details are retrieved and returned to the client, and in the negative case an empty user is returned. If user is an administrator, the administrator options are visible, if it’s a registered user, the options of the visible users are visible, while in the negative case an Error message is displayed.

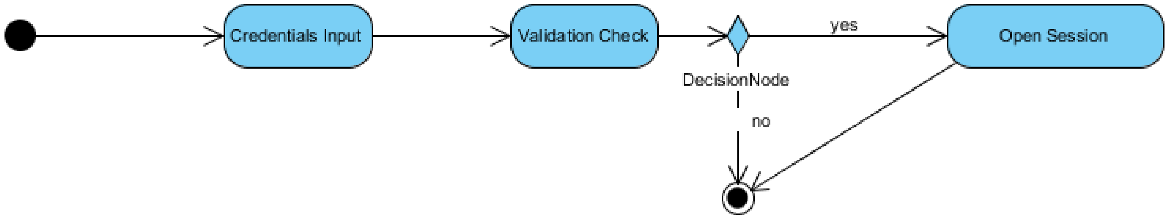


Figure 12: Login functionality

View information data of the company

It concerns basic information about the e-shop, branches that the e-shop has and their telephone numbers for contact.

View Information

Figure 13: View information functionality

Sign up

It displays the form with the fields that must be filled in by the new user in order to register himself. After completing the form, fields are sent to the server where they are checked for correctness. In a positive case, they are saved in the database and a message of success is displayed to the client's screen whereas in the negative case an error message is displayed.

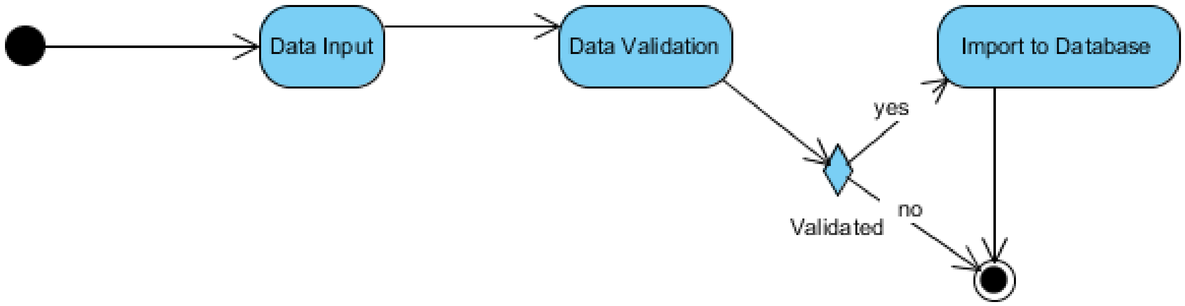


Figure 14: Sign up functionality

View product list

It concerns the display of the products that a user can buy.

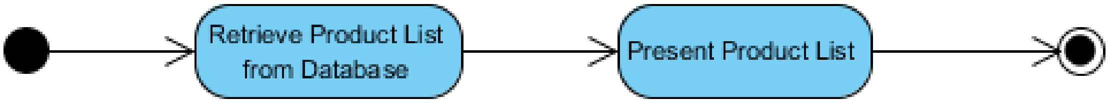


Figure 15: Product list presentation functionality

View category list

It concerns the display of the categories of products.

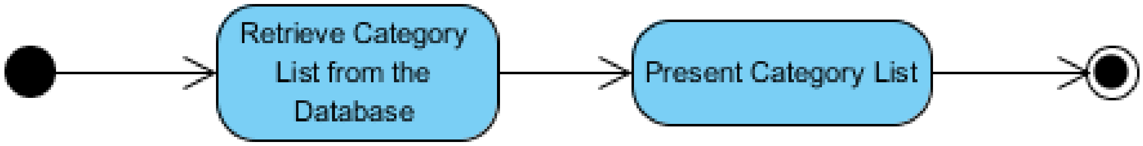


Figure 16: Category list presentation functionality

View product details

The information of the products is retrieved from the database and is sent to the client for presentation to the user.

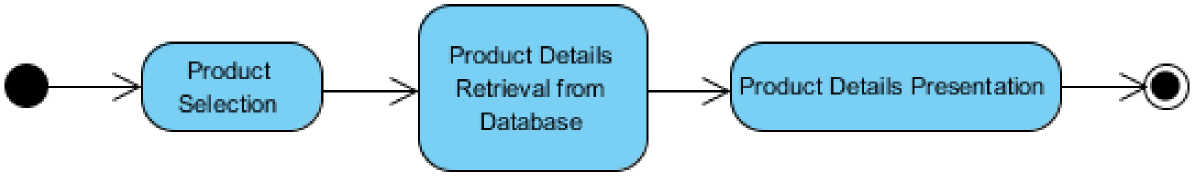

Product Selection
Product Details Retrieval from Database
Product Details Presentation

Figure 17: Product details presentation functionality

Product Purchase

The user makes the purchase of the product that he wants by filling the appropriate fields and making the payment process with the desired way (credit card, PayPal). So the payment process is made by taking the appropriate amount of money from the payment account of the user. Also the order of the products is sent to the server and if the purchase is successful, then a success message is returned and the order of the products is displayed on the screen.

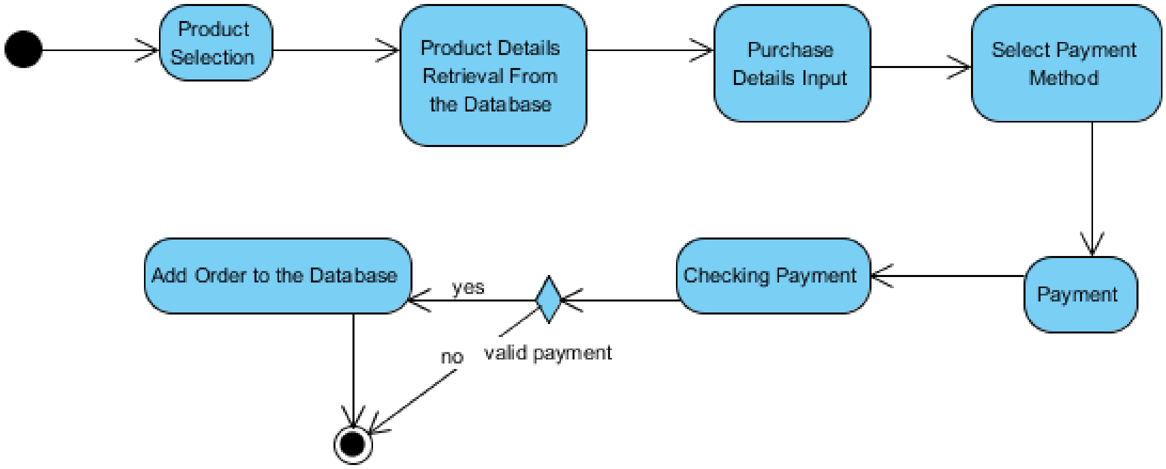


Figure 18: Product purchase functionality

View evolution of order

It concerns the display of the evolution order step.

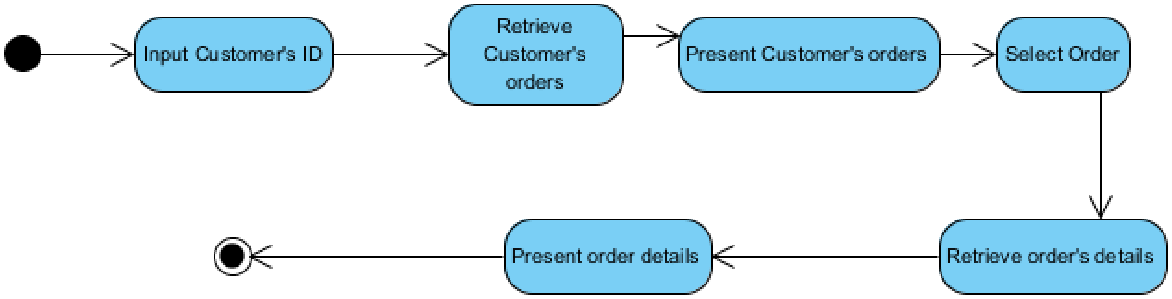


Figure 19: Evolution of order presentation functionality

Show payment methods

It concerns the display of the payment methods for a user.

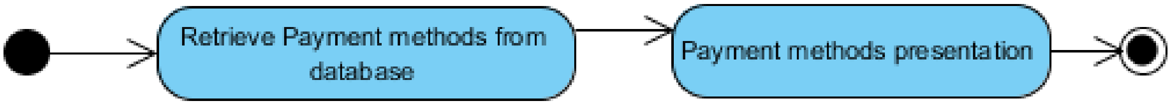

Retrieve Payment methods from database
Payment methods presentation

Figure 20:Payment method presentation functionality

View cart

It concerns the display of the order of the products for purchase.

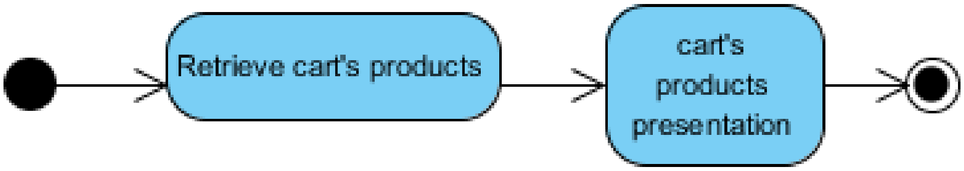


Figure 21: Cart presentation functionality

Process of information data of the company

Information data of the company are processed by the administrator. Updated information data are transferred to the server in order to be updated.

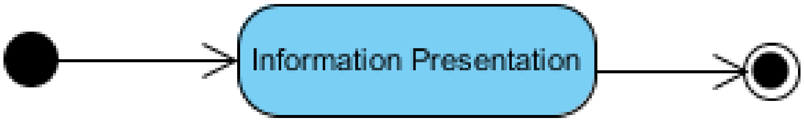


Figure 22: Company’s information presentation functionality

Users' process

Users' data are retrieved from the database and are returned to client for process. Updated users data are collected and transferred to the server for validation and in positive case, database is updated. In case of error, appropriate message is sent.

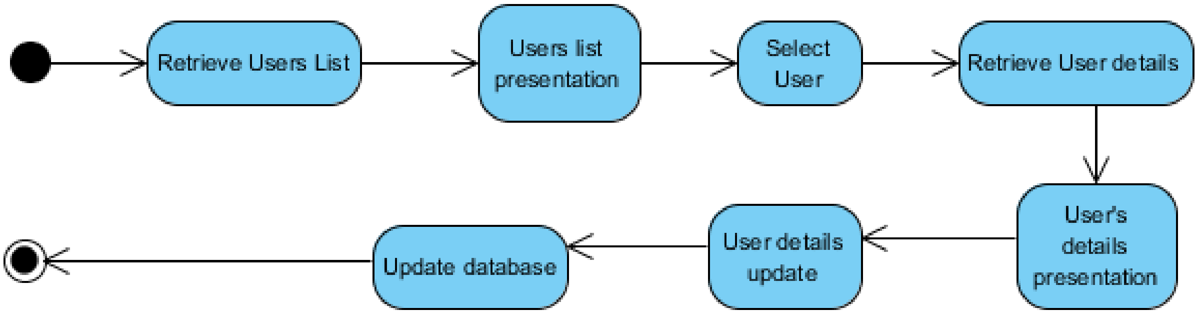


Figure 23: Users' process functionality

Registration of new product

Product data are collected and transferred to the server for validation and in positive case data are saved in the database. In case of error, an appropriate message is sent.

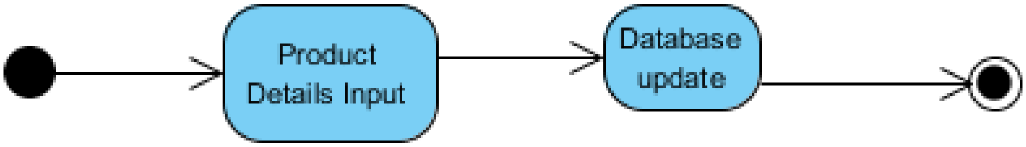


Figure 24: Product registration functionality

Registration of new category

Category data are collected and transferred to the server for validation and in positive case data are saved in the database. In case of error, an appropriate message is sent.

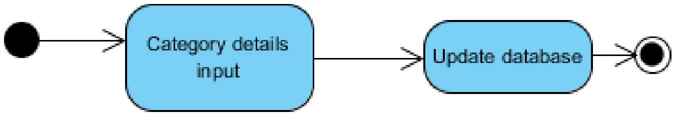


Figure 25: Category registration functionality

Product update

Product code is selected, sent to the server and product data are retrieved from the database and returned to the client. Data are updated and sent back to the server to update the database. Finally, a message appears about the success or failure of the process.

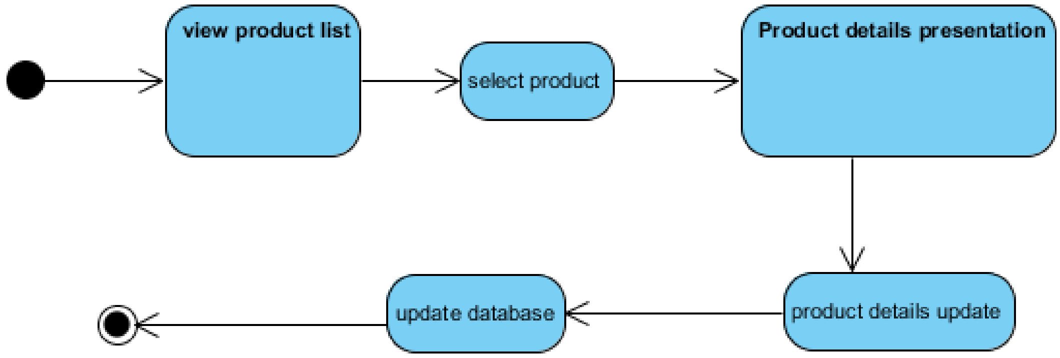


Figure 26: Product details' update functionality

Category update

Category code is selected, sent to the server and category data are retrieved from the database and returned to the client. Data are updated and sent back to the server to update the database. Finally, a message appears regarding the success or failure of the process.

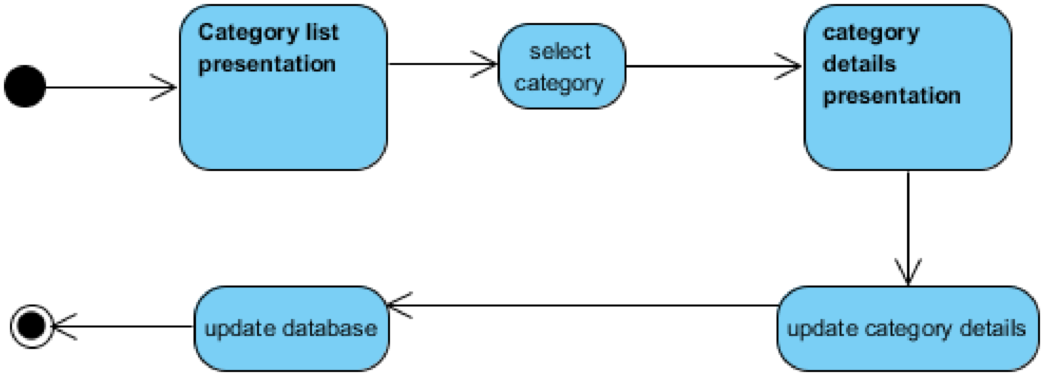


Figure 27: Category details' update functionality

Orders view

It concerns the display of the orders to the administrator.

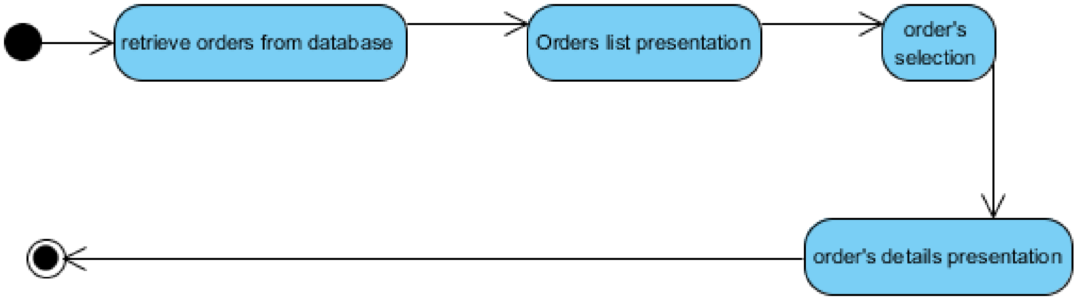


Figure 28: Orders presentation functionality

Process of evolution order

Evolution order status is retrieved from the database and is returned to client for process. Updated status is collected and transferred to the server for validation and in positive case, database is updated. In case of error, appropriate message is sent.

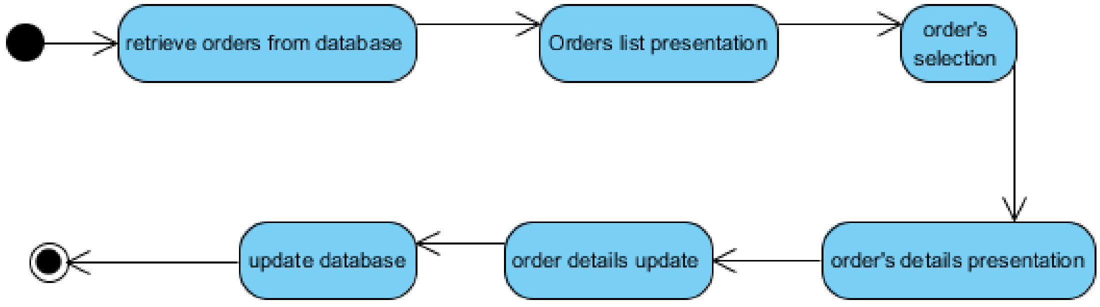


Figure 29: Process of evolution order functionality

Interfaces Design

Here we describe the application interfaces specifications. The interfaces were divided into two categories, those which are visible by users (simple visitors and the registered users) of the application and those handled by the administrators of the e-shop. Choices of functions will be made in a way that gives the end user the feeling that he moves from function to function by remaining in the same application.

User

Template

The application interface structure consists of four main sections:

- A banner on the top of the screen containing the e-shop logo and session information.
- The application's main menu
- The main content that refers to the selected functionality. It also contains a secondary menu if it is necessary.
- A section at the bottom of the screen containing general information, social network and contact links.

Home Page

There will be a welcome message to the e-shop application in the home page and a list of branches details of the e-shop including the phone numbers of each of them.

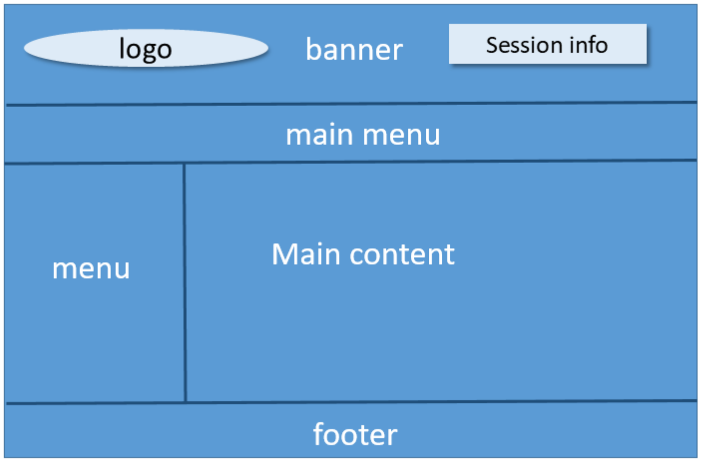


Figure 30: Home page

User registration

A form which contains appropriate fields for the registration of the user's records is displayed in this interface. The user will be able to enter free text in most fields and there will be a choice field for the user to choose his way of payment. Also the field that will be used to enter the password will hide the text they contain. In case of date registration a date box will appear to the user to choose date of registration. After having completed the fields, the user clicks on an appropriate button, so their correctness is checked and an appropriate message is displayed. Appropriate messages will be shown in case they report a registration error. If no error occurs, then the data are entered in the database.

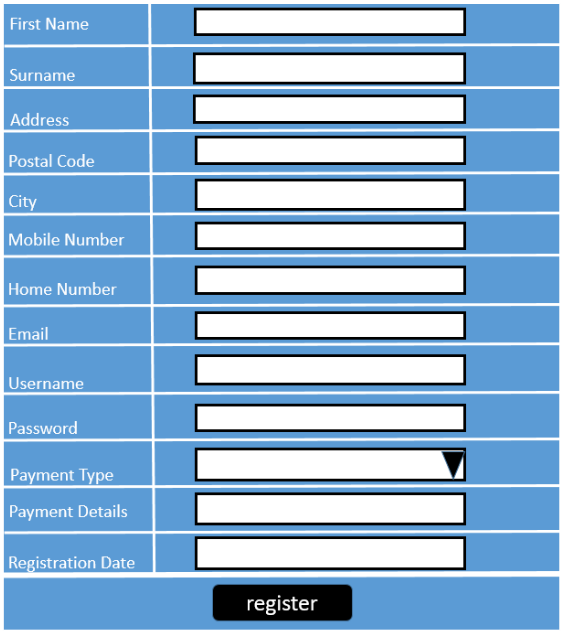


Figure 31: User registration page

Products

On this screen products will be presented in categories with description details, price and unit in stock and on each product there will be a field for putting units of that product on the cart for purchase.

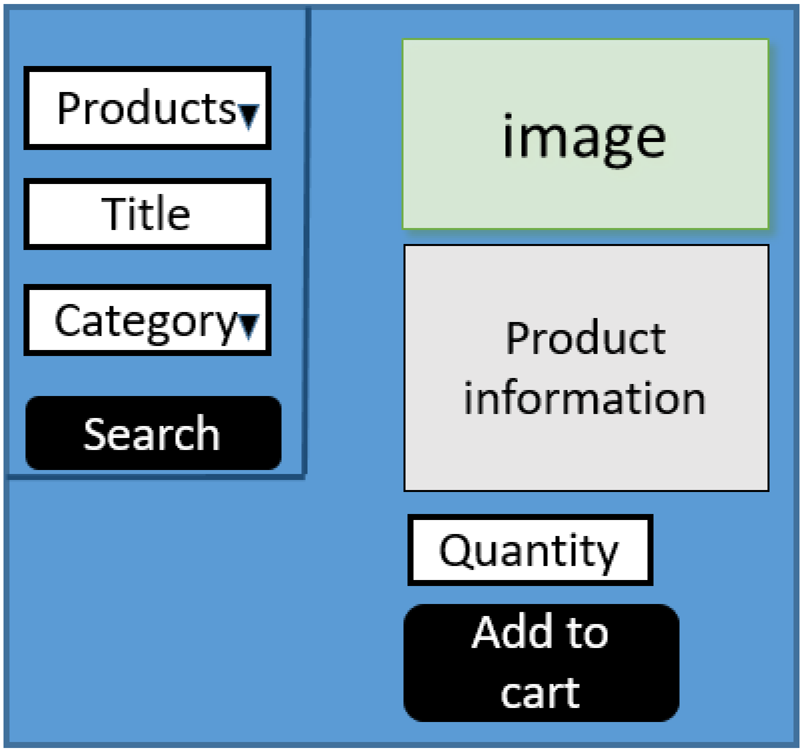


Figure 32: Product search Page

Purchase completion

On this screen, the user can choose the branch where he will receive his order. When this is completed, there is a send order button that the user presses to complete the purchase. If the user chooses PayPal payment type on the right side of the screen, there will be a PayNow button to enter to PayPal site and pay. However, if user chooses credit card or cash as payment types, the user will just press the send order button and the purchase will be completed.

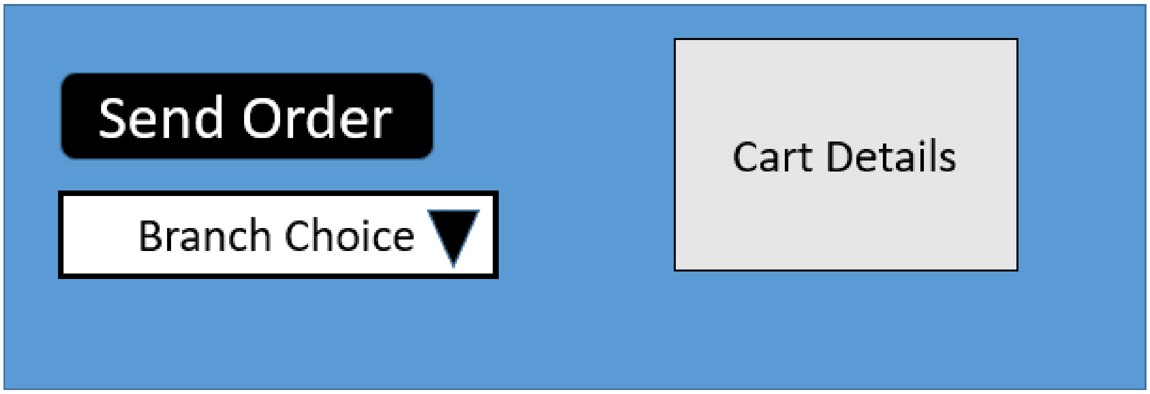


Figure 33: Purchase completion Page

Update Registration

In this form the customer can change his records which he has already registered in the registration form or delete his account in the e-shop application. After the customer changes the fields that he wants, he can press the Update button for the data to be updated or immediately press the delete button once he enters in the update registration form. An appropriate message is displayed in case of success; otherwise an error message is displayed.

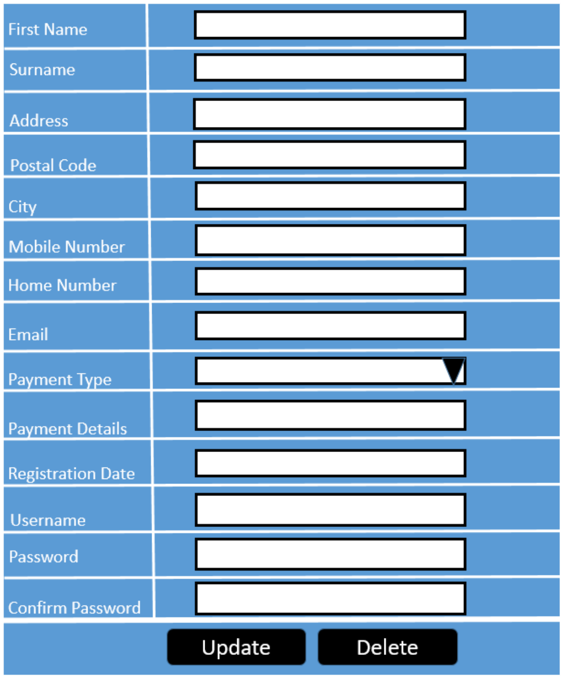


Figure 34: Update Registration Page

Login

A form is displayed to the user in order to enter his username and password. When these are submitted, they are transferred to the server and they are checked for validation. In case that credentials are correct, user's details are retrieved and returned to the client, and in the negative case an empty user is returned. If the user is an administrator, administrator options are visible, if he’s a registered user, options of the visible users are visible, whereas in the negative case an error message is displayed.

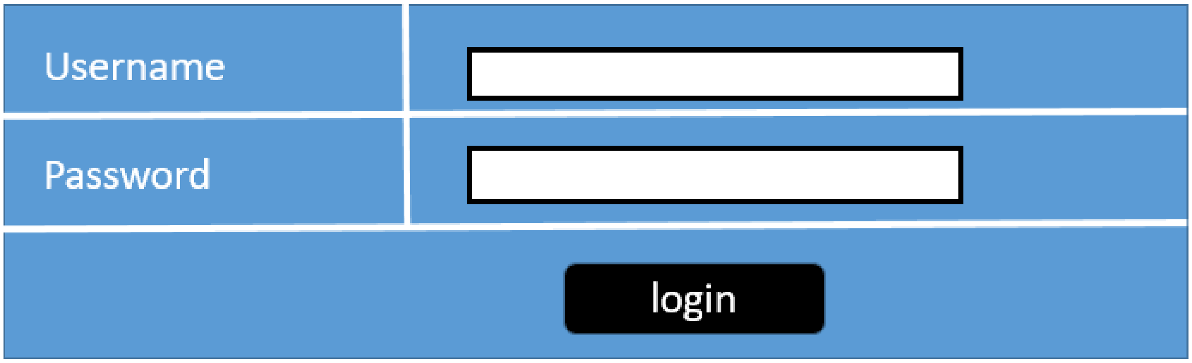


Figure 35: Login Page

Logout

User who has logged as a registered user to the site and wants to logout from his account, he clicks on an appropriate button to terminate his connection. With the termination of his connection logout button disappears and the login button is displayed instead.

Administrator

User registration

User registration can be done also by the administrator except for the customer of the e-shop.

Product registration

The administrator can register a new product by recording its details in an appropriate form. The fields, in which product data can be entered, are free text, numbers and date checkbox. After completing the registration, an appropriate button is clicked and data are entered in the database. If the entry is successful, an appropriate message appears otherwise an error message appears.

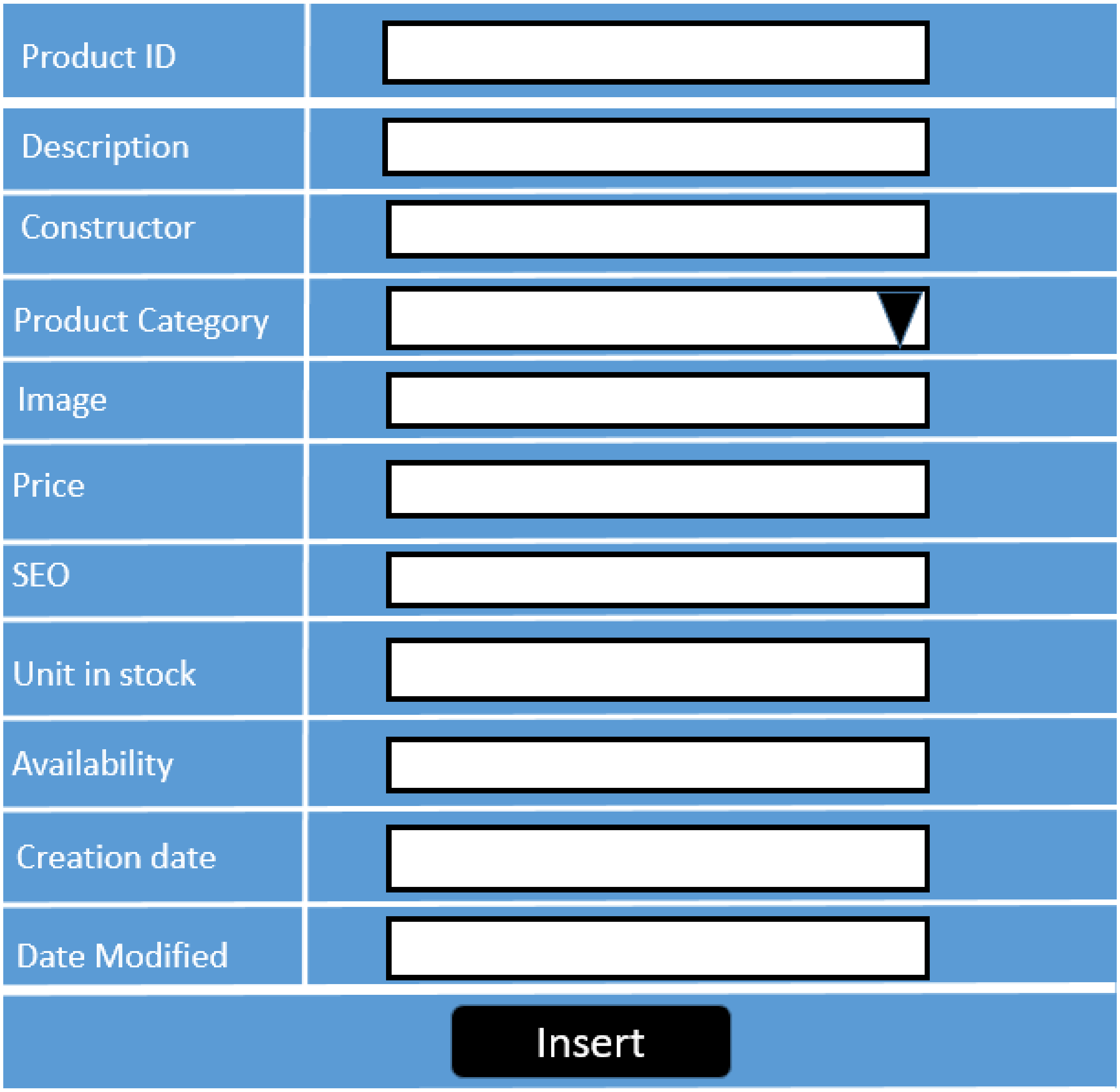


Figure 36: Product registration page

Category registration

The administrator can register a new category by recording its details in an appropriate form. The fields in which product data can be entered are free text. After the registration is completed, an appropriate button is clicked and data are entered in the database. If the entry is successful, an appropriate message appears, otherwise an error message appears.

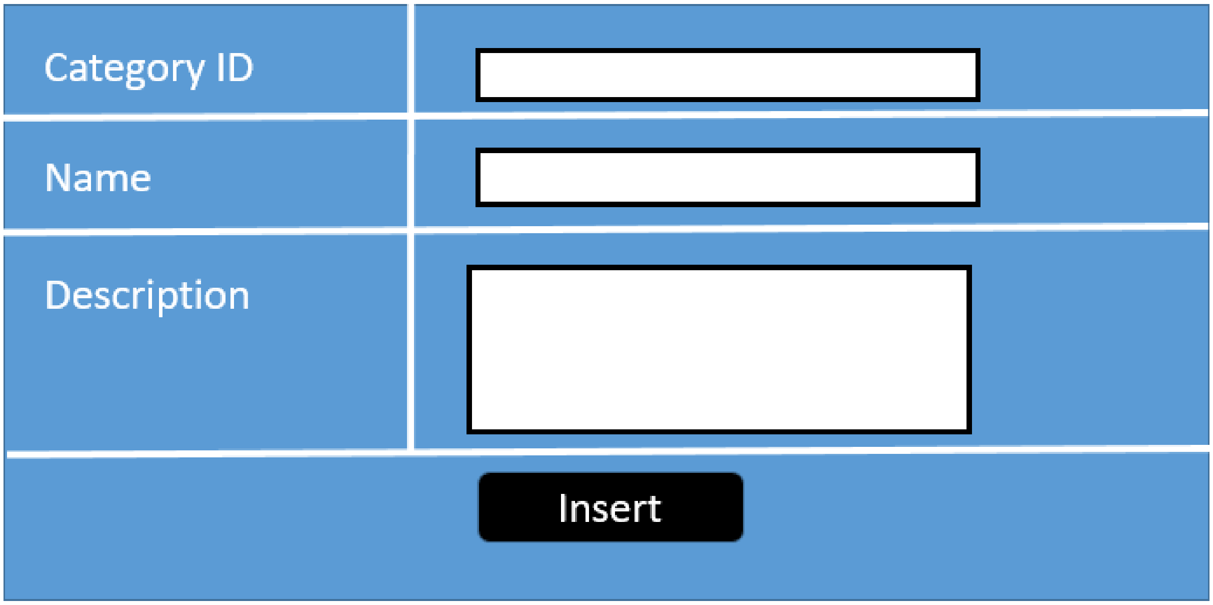


Figure 37: Category registration page

Update product details

The administrator selects the product that he/she wants. There the availability of the product is displayed (checkbox will be ticked if the product is available). Once the administrator has defined the availability of the product, he will click on an appropriate button so that changes will be recorded in the database. If the updates are successful, a success message is displayed, otherwise an error message is displayed.

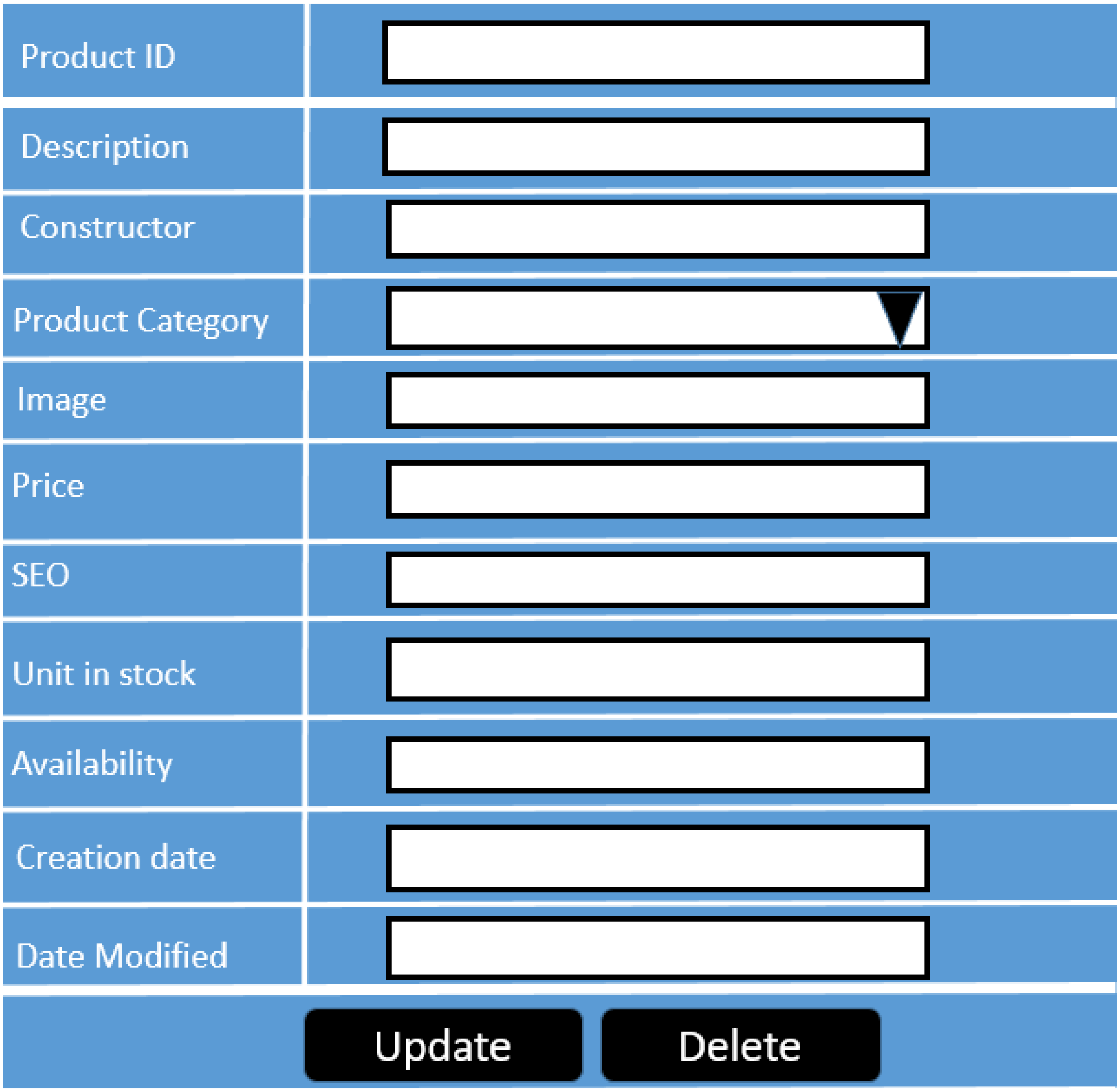


Figure 38: Product update page

Update category details

The administrator selects the category that he wants through combo box. Then the elements of the category are displayed in fields of free text which the user can modify. If the updates are successful, a success message is displayed otherwise an error message appears.

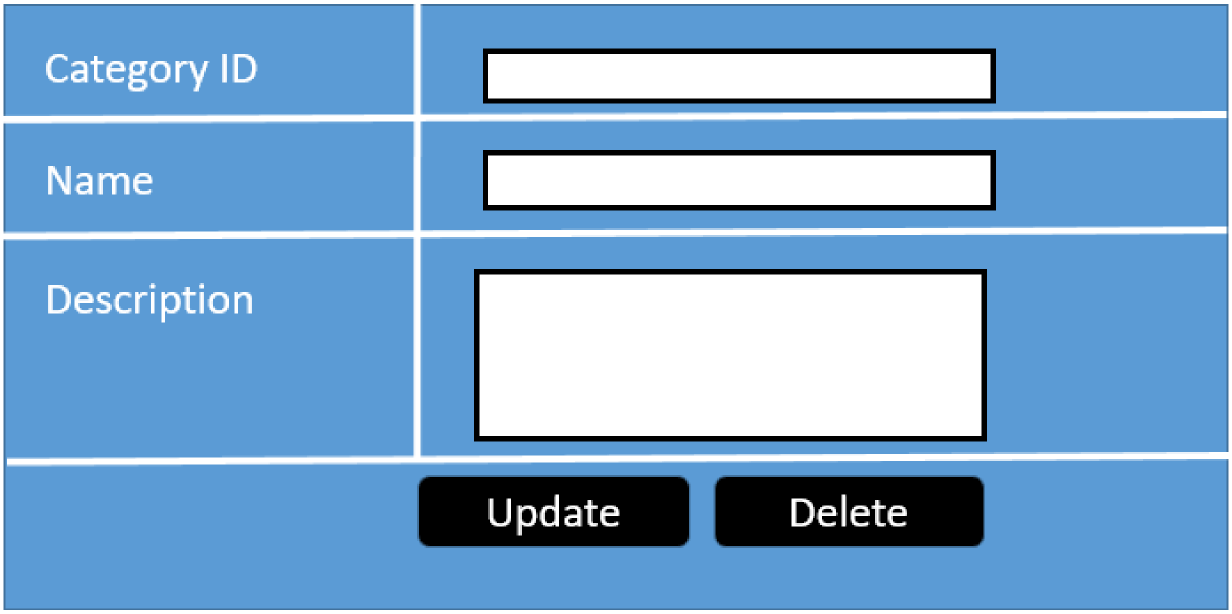


Figure 39: Category update page

Orders search

In this interface, the user will select the status of the order from the combo box, as well as from two calendars, the period when the orders are searched. Clicking on an appropriate button will show the orders that match with the given criteria or an error message will appear on a red background.

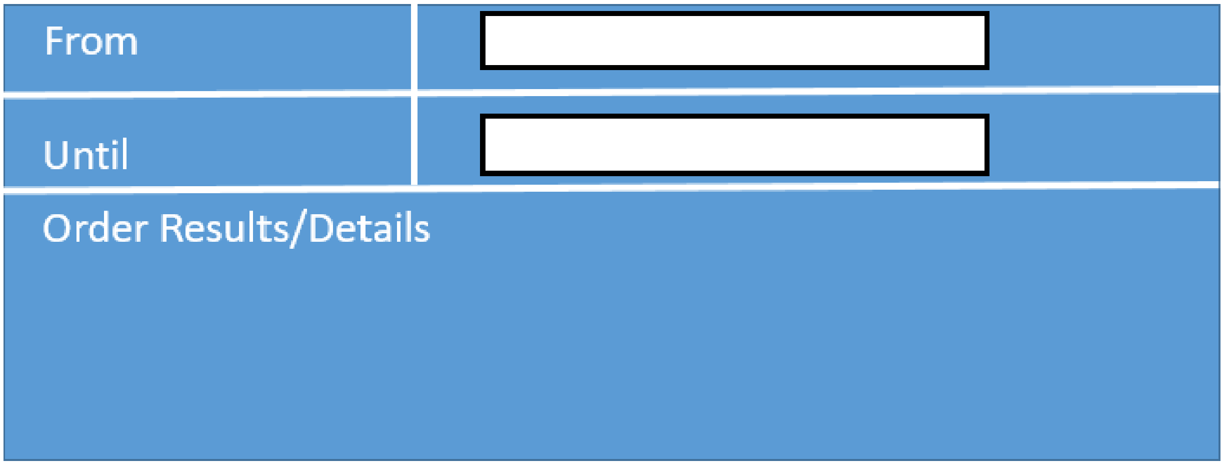


Figure 40: Order search page

Login

Administrators and registered users can login from the same interface. However, successful login both opens and displays available options for administrators.

Logout

It is done in the same way just like registered users do.

Facebook publication mechanism

The current application provides a mechanism for publishing the content of products to the e-shop’s Facebook page. The RestFb API was chosen to be used for implementing it. This API handles the Facebook social graph for creating or grapping information derived from its study.

Social Graphs

"Social networks are popular platforms for interaction, communication and collaboration between friends."[15] "Social networks such as MySpace and Facebook provide communication, storage and social applications for hundreds of millions of users. Users join, establish social links to friends and leverage their social links to share content, organize events, and search for specific users or shared resources. "[15]

In social networking applications, each user has an account. This account records information about the person who makes his basic profile. This information varies according to the orientation of the application but there is a basic set that is found in all cases. Such data are demographic and social. More specialized categories of data might refer to economic, political, and other various kinds of preferences. In addition to the data the user himself registers. His profile also includes information that is generated by the interpretation of user interfaces. Profile data, in any way that is stored, remain a valuable research base for many different directions.

Graphs which are also known as social networking graphs are used for graphing and studying user profiles [15].Generally, the study of graphs is used in many scientific fields for research purposes. Some typical examples can be graphs that represent data networks, neural networks, or entity associations. Social graphs represent the associations between members of a social network [15]. So it is a diagram illustrating the interconnections among people, groups and organizations. The term is also used to describe a person's social network. When it is displayed on the two-dimensional level, it appears as a set of network nodes linked to lines [15]. Each node, which can also be referred to as acting, is in fact a data file that compiles its profile. The link lines, which are used to map the relationships among them, are referred to as edges, bonds or interdependencies. Due to the fact that modern social networks are used by millions of people’s profiles, groups or organizations and multiple interconnections, social graphs are highly complex. This makes it quite difficult to analyze and study them. Moreover, their characteristic is that they are dynamic, as the edges can be adhered at a high rate. In most social networks, the graphs that represent them are oriented since the associations among the node members can be two-way. Specialized software is used to process them. The object of their process can be their structure, the way they evolve in time and space and their durability which is their ability to maintain their consistency and structure after removing nodes or edges.

Spring Social Network

"Spring Social Facebook is an extension of Spring Social that enables web users to connect applications with Facebook's Graph API." [16] It consists of the following features: "

- A service provider and a connection factory for using Spring Social's connection framework,
- An API binding for Facebook's Graph API,
- A real-time update controller for handling real-time update callbacks from Facebook
- A disconnect controller to handle disconnect callbacks from Facebook
- Much more…"[16].

Spring social API provides users with the opportunity to connect to applications through their Facebook account. It also provides the administrators of the application with the ability to send post to Facebook. It is easy to be embodiment to spring web applications. Additionally, it is characterized by its simple and its developers-friendly interfaces’ functionality.

Facebook4J

"Facebook4J is an unofficial Java library for the Facebook Graph API." [17] It is written in pure Java programming language and it can be added to applications working on any Java Platform version 5 or later [17]. It is not dependent to any other external library [17]. Developers have just to add Facebook4J to their applications. Application administrators can make the following actions using this API:

- Publishing a message
- Publishing a link
- Getting News Feed
- Add “Like” to publications
- Publishing a comment
- Searching the social graph
- Search for Events
- Search for Groups
- Search for Places
- Search for Check-ins
- Search for Locations
- Executing Queries to Social Graph [17]

RestFb

"RestFB is a simple and flexible Facebook Graph API client written in Java." [18] It is minimal and developers have to use just two methods: "

- a method to fetch information from
- a method to publish information." [18]

Developers do not have to handle any code. They only have to add restFB.jar file to their application’s class path and then configure simple components in order to gain APIs functionality. RestFb simplicity was its strong advantage which concerned the chosen decision for the API to be used for the publication mechanism.

RestFB usage

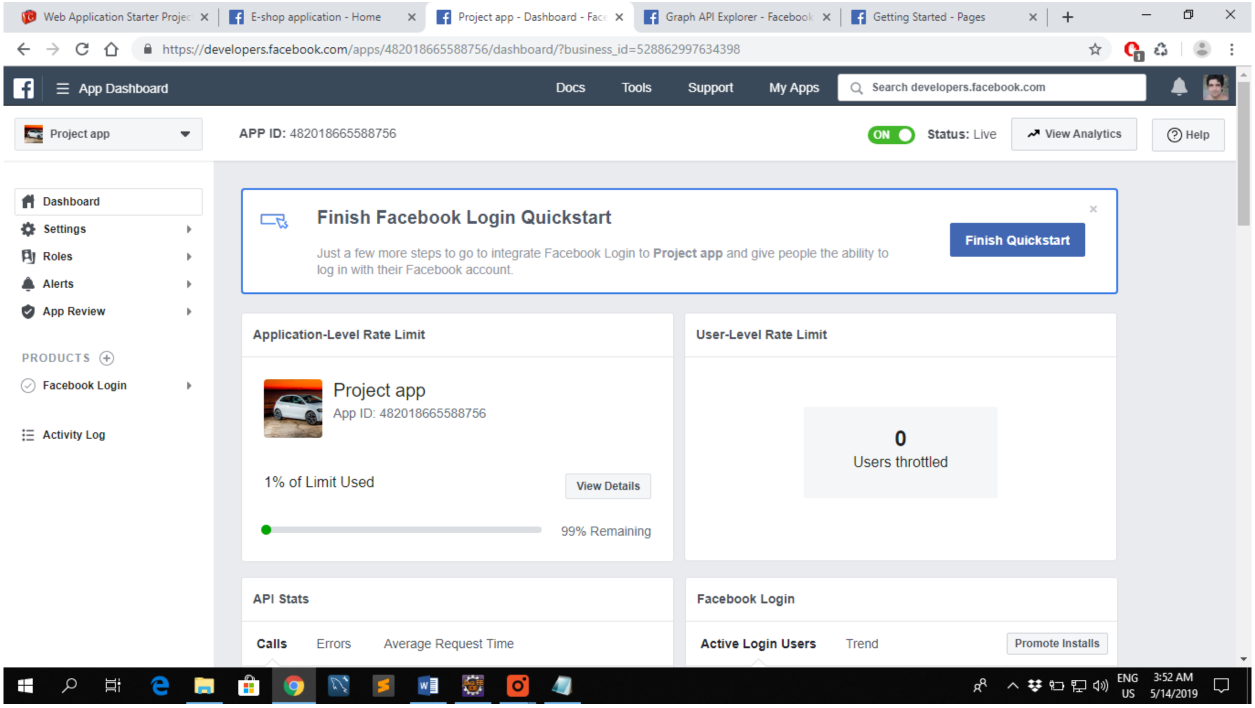
To implement the publishing mechanism through the company's Facebook page, a class was created that makes a link to the administrator for the company’s Facebook page. It also includes a method that records a post. This post includes a message and an image. This particular method is called:

- When a new product is registered.
- When product data is updated.

There is also a servlet. While it is being loaded, it is being refreshed periodically (the period is being defined by the administrator to the Facebook Service Configuration section) by sending a new publication to the Facebook page concerning a randomly chosen product.

Facebook application

A Facebook application was made for obtaining access token and therefore being able to post as a page. The application was named Project App.



https://developers.facebook.com/apps/482018665588756/dashboard/?business_id=528862997634398

APP ID: 482018665588756

Status: Live

Finish Facebook Login Quickstart

Just a few more steps to go to integrate Facebook Login to **Project app** and give people the ability to log in with their Facebook account.

Application-Level Rate Limit

Project app
App ID: 482018665588756

1% of Limit Used

99% Remaining

User-Level Rate Limit

0
Users throttled

Figure 41: Facebook application page

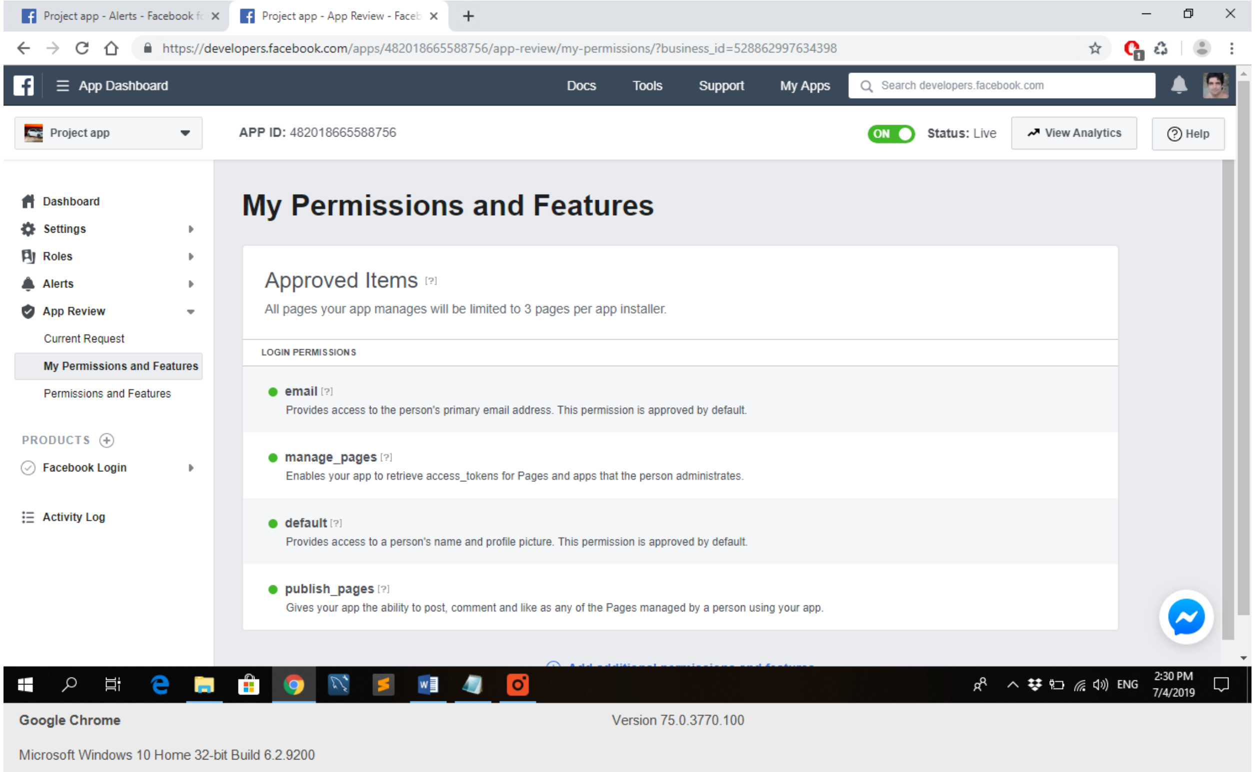
In order to be able to post products through Facebook page, two attributes (manage pages and publish pages) were needed to be approved by Facebook.

When the application was created both Email and default attribute were default attributes but it was necessary to be approved by both manage_pages and publish_pages in order to be able to post products on Facebook. There were difficulties in order these items to be approved because the Facebook team asked for specific instructions to be followed in order to have a

Facebook post approved. It should be written a detailed description about the e-shop application and sent to Facebook.

In addition it should be recorded and sent to Facebook a detailed video description of how a product of the e-shop application would be posted through Facebook mechanism on the page of Facebook. So Facebook page should be made in advance to show how the product would be posted on this page. Since Facebook mechanism could not work without the approval of Facebook team, a product on Facebook page was posted manually to show the process. The exact way of Facebook posting step by step from the beginning should be shown; otherwise the attributes are not approved. So, all the instructions given from Facebook team should be followed very carefully and without mistakes in order to be approved immediately and with no time loss. Facebook team could see a wrong detail in the process and reject the request for the attributes for only that detail.

After both of these attributes were approved, a Facebook page of the e-shop application was made for posting products of this application.


Project app - Alerts - Facebook f
Project app - App Review - Faceb
https://developers.facebook.com/apps/482018665588756/app-review/my-permissions/?business_id=528862997634398
App Dashboard
Docs
Tools
Support
My Apps
Search developers.facebook.com
Project app
ON
Status: Live
View Analytics
Help
Dashboard
Settings
Roles
Alerts
App Review
Current Request
My Permissions and Features
Permissions and Features
PRODUCTS
Facebook Login
Activity Log
2:30 PM
7/4/2019
ENG

APP ID: 482018665588756

My Permissions and Features

Approved Items [?]

All pages your app manages will be limited to 3 pages per app installer.

LOGIN PERMISSIONS

● email [?]
Provides access to the person's primary email address. This permission is approved by default.

● manage_pages [?]
Enables your app to retrieve access_tokens for Pages and apps that the person administrates.

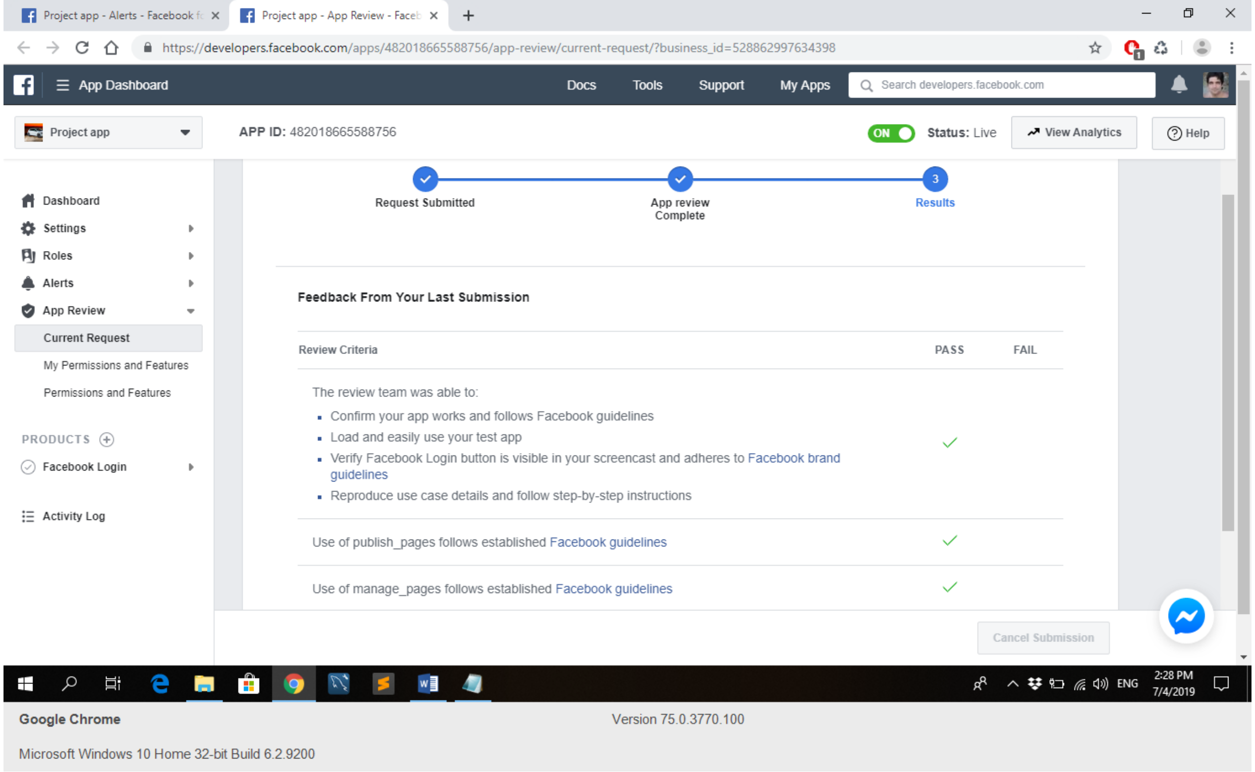
● default [?]
Provides access to a person's name and profile picture. This permission is approved by default.

● publish_pages [?]
Gives your app the ability to post, comment and like as any of the Pages managed by a person using your app.

Google Chrome Version 75.0.3770.100

Microsoft Windows 10 Home 32-bit Build 6.2.9200

Figure 42: Approved attributes and their description



APP ID: 482018665588756

Status: Live

Feedback From Your Last Submission

Review Criteria	PASS	FAIL
The review team was able to: ▪ Confirm your app works and follows Facebook guidelines ▪ Load and easily use your test app ▪ Verify Facebook Login button is visible in your screencast and adheres to Facebook brand guidelines ▪ Reproduce use case details and follow step-by-step instructions	✓	
Use of publish_pages follows established Facebook guidelines	✓	
Use of manage_pages follows established Facebook guidelines	✓	

Google Chrome Version 75.0.3770.100

Microsoft Windows 10 Home 32-bit Build 6.2.9200

Figure 43: Results of approving attributes request

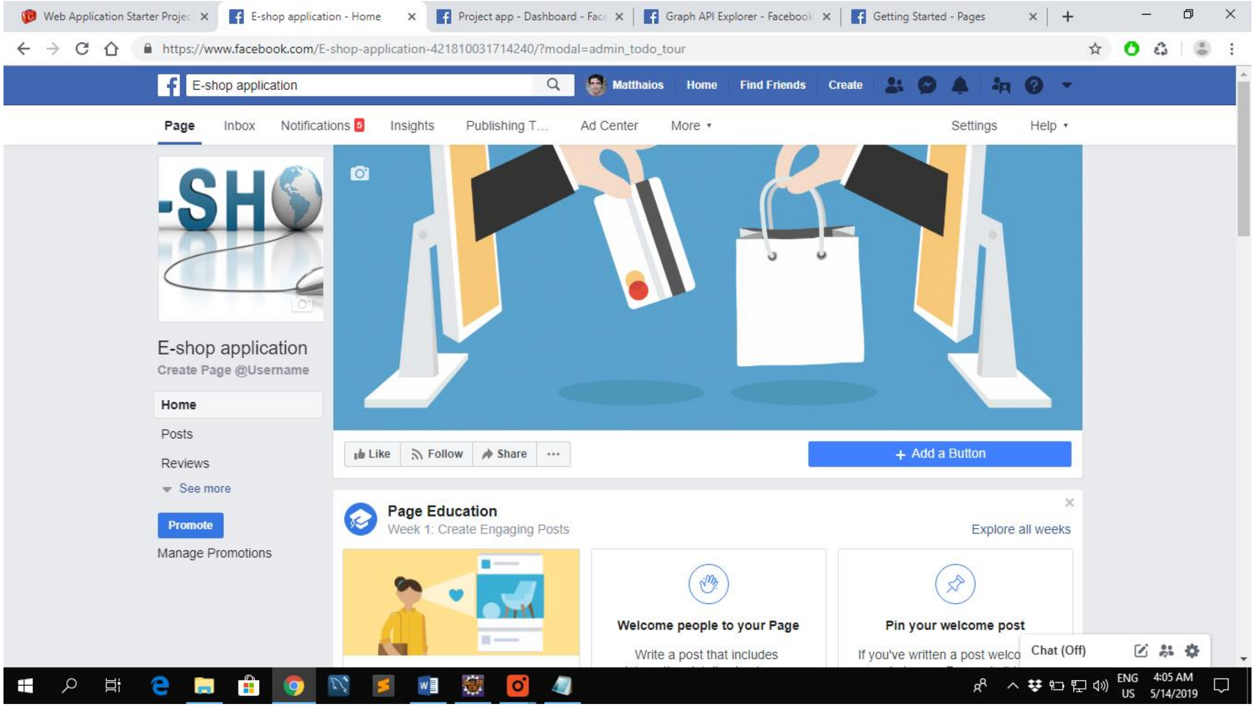
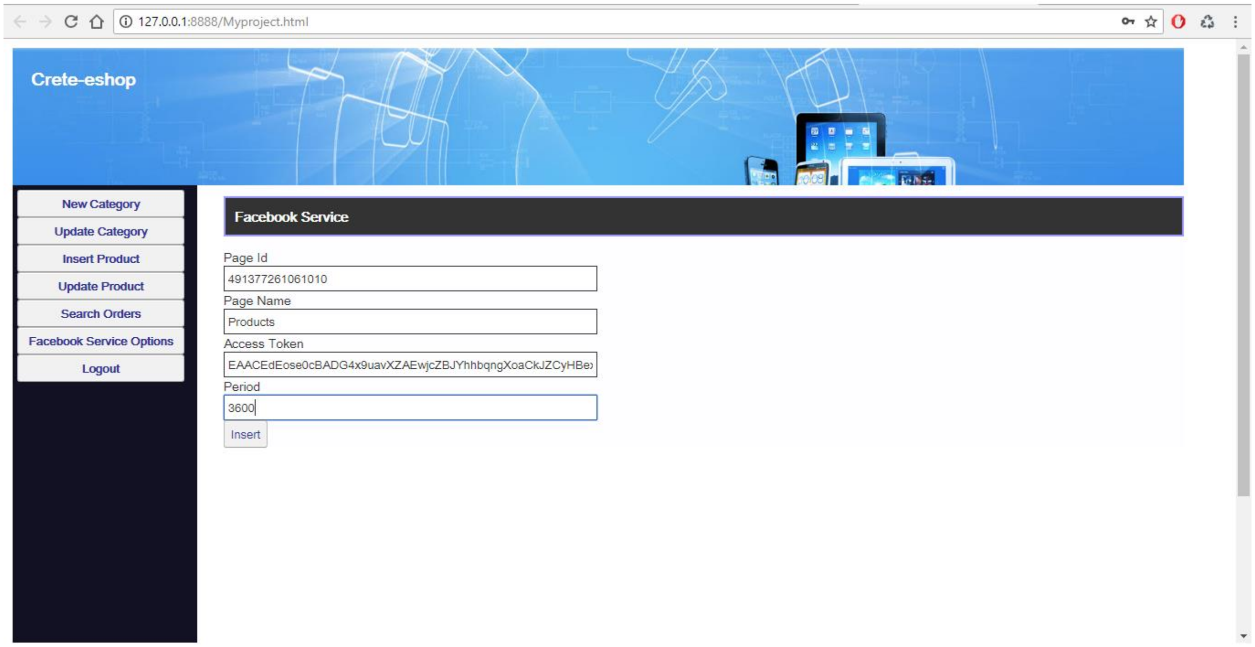


Figure 44: Facebook page for posting products

Administrator can set the Facebook publishing mechanism parameters in the e-shop application.



Page Id
491377261061010
Page Name
Products
Access Token
EAACEdEose0cBADG4x9uavXZAEwjcZBJYhhbqngXoaCkJZCyHBe
Period
3600

Figure 45: Facebook service options page

The steps that the administrator has to follow in order to post a product on Facebook page are the following:

1. Click on Facebook Service Options button.

2. Type the APIs parameters to the appearing form. The fields that must be completed are the Id of the Page, Name of the page, Access Token and period which is the frequency that products are uploaded on Facebook mechanism.

Here we give a more detailed description of these fields.

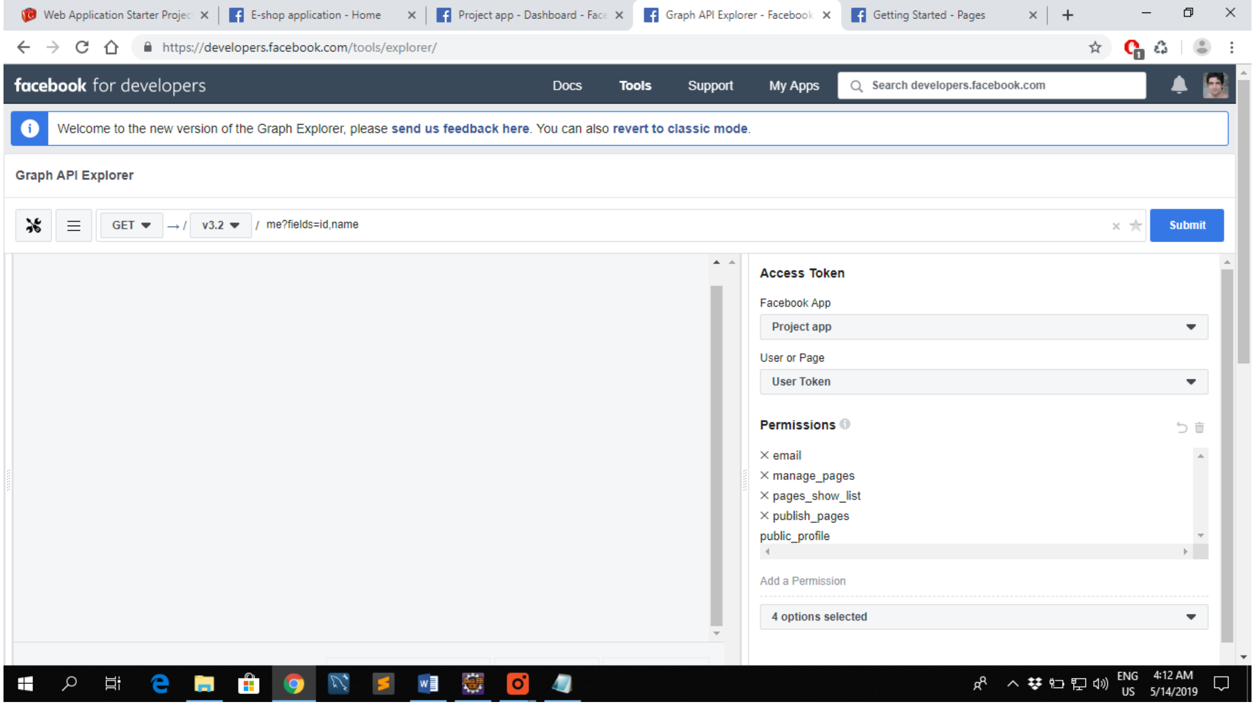
Page Id: The Id of the Facebook page that was created for posting products.

Page Name: The name of the Facebook page.

Access Token: It is obtained through Facebook for developers' site where the Facebook application was made for obtaining access token.

Period: The time interval between two posts on Facebook, for example every 3600s, a product on Facebook page will be posted. It is counted in seconds.

3. On Facebook for developers' site we go to Docs-> Tools->Graph API Explorer. We meet there the following screen.


Web Application Starter Projec
E-shop application - Home
Project app - Dashboard - Face
Graph API Explorer - Facebook
Getting Started - Pages
https://developers.facebook.com/tools/explorer/
facebook for developers
Docs
Tools
Support
My Apps
Search developers.facebook.com
Submit

Welcome to the new version of the Graph Explorer, please **send us feedback here**. You can also **revert to classic mode**.

Graph API Explorer

GET → / v3.2 / me?fields=id,name

Access Token

Facebook App: Project app

User or Page: User Token

Permissions

- × email
- × manage_pages
- × pages_show_list
- × publish_pages
- public_profile

Add a Permission: 4 options selected

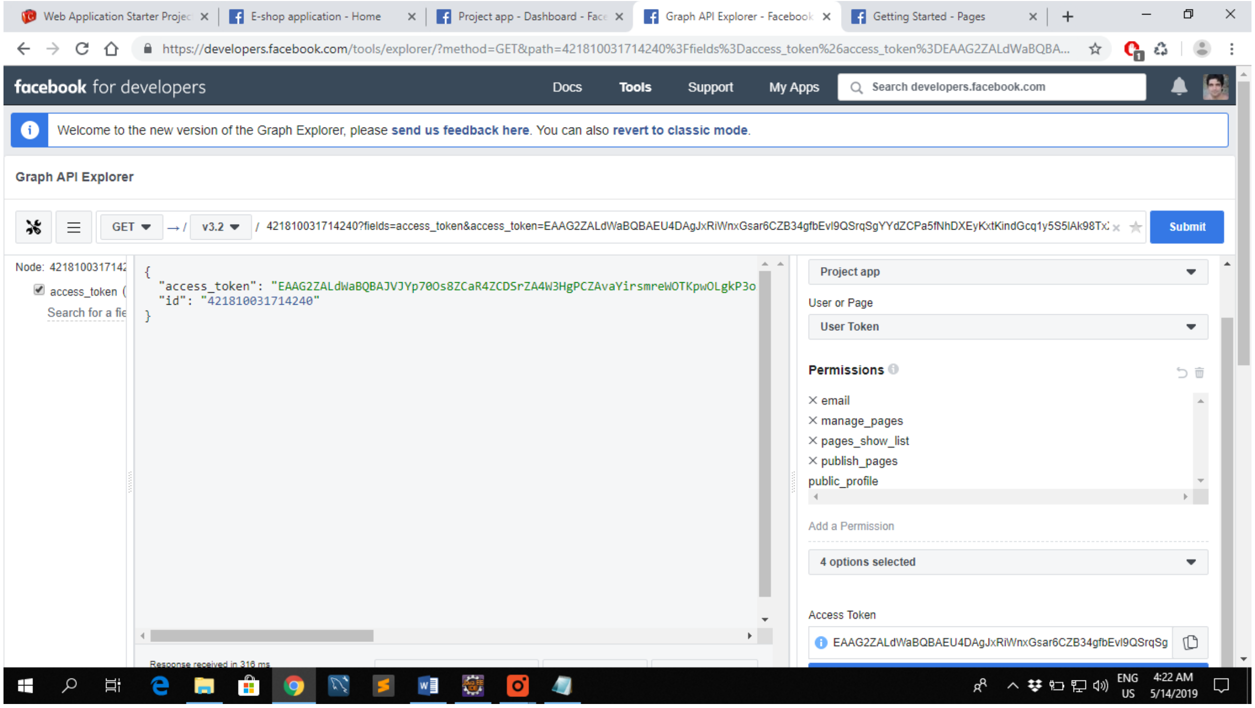
Figure 46:Graph API on Facebook for developers' site (1)

4. From that screen we obtain the page access token. On the top bar after v3.2 we write the syntax command: your-page-id?fields=access_token&access_token=your-user-access-token

We replace variables your-page-id= 421810031714240 and your-user-access-token= EAAG2ZALdWaBQBAEU4DAgJxRiWnxGsar6CZB34gfbEvl9QSrqSgYYdZCPa5fNhDXEyKxtKindGcq1y5S5lAk98TxXIXkuSRVlyyYoerZB1g482LTUtpnLh8jBmDCNv9eu7DsvXZBIZCd88MRVddCGgzBKXzDJ5IdQYvQdMZBmZCrTUvldpZCoCCoNkf8jNEYm4oGYZD

5. We hit submit button

6. From the screen below we see the page access token that corresponds to the page Id.



Welcome to the new version of the Graph Explorer, please **send us feedback here**. You can also **revert to classic mode**.

Graph API Explorer

GET → / v3.2 / 421810031714240?fields=access_token&access_token=EAAG2ZALdWaBQBAEU4DAgJxRiWnxGsar6CZB34gfbEvl9QSrqSgYYdZCPa5fNhDXEyKxtKindGcq1y5S5lAk98Tx

Node: 421810031714

☑ access_token

```
{
  "access_token": "EAAG2ZALdWaBQBAJVJYp70Os8ZCaR4ZCDSrZA4W3HgPCZAvaYirsmreWOTKpwOLgkP3o
  "id": "421810031714240"
}
```

Project app

User or Page: User Token

Permissions

× email

× manage_pages

× pages_show_list

× publish_pages

public_profile

4 options selected

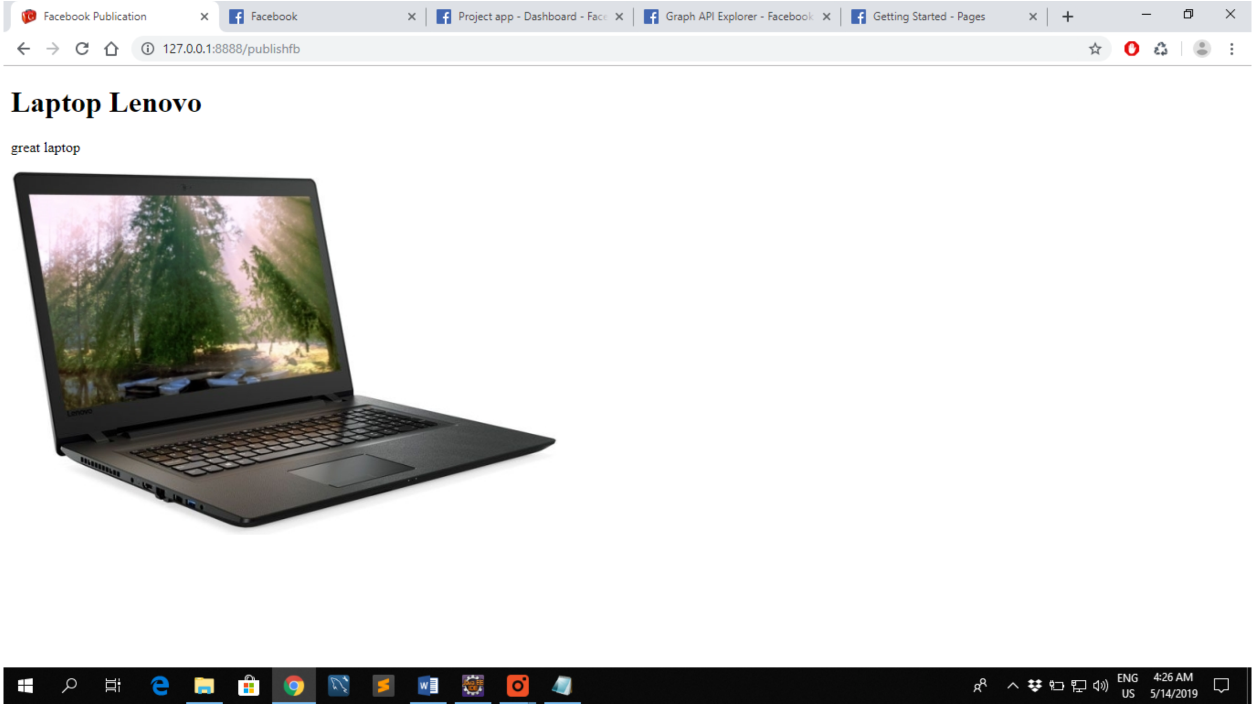
Access Token: EAAG2ZALdWaBQBAEU4DAgJxRiWnxGsar6CZB34gfbEvl9QSrqSg

Web Application Starter Projec
E-shop application - Home
Project app - Dashboard - Fac
Graph API Explorer - Facebook
Getting Started - Pages
https://developers.facebook.com/tools/explorer/?method=GET&path=421810031714240%3Ffields%3Daccess_token%26access_token%3DEAAG2ZALdWaBQBA...
facebook for developers
Docs
Tools
Support
My Apps
Search developers.facebook.com
Search for a fie
Add a Permission
Submit
Response received in 316 ms
ENG
US
4:22 AM
5/14/2019

Figure 47:Graph API on Facebook for developers' site (2)

7. On the web browser the Administrator sends request to get the publishfb link: page /publishfb

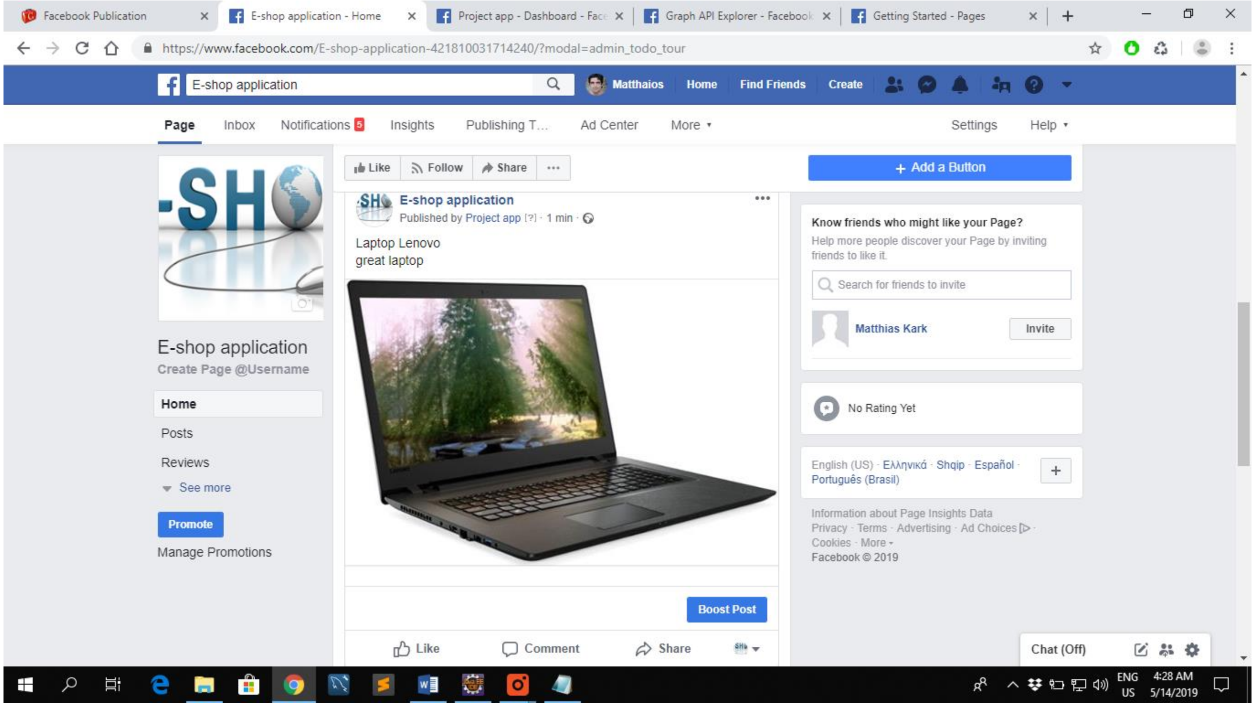
8. After entering in publishfb link the same time the product that we see in the following screen is posted on Facebook page of the application



Laptop Lenovo

great laptop

Figure 48: Screen that displays the product image that is posted on Facebook page


Facebook Publication
E-shop application - Home
Project app - Dashboard - Face
Graph API Explorer - Facebook
Getting Started - Pages
https://www.facebook.com/E-shop-application-421810031714240/?modal=admin_todo_tour
E-shop application
Matthaios
Home
Find Friends
Create
Page
Inbox
Notifications 8
Insights
Publishing T...
Ad Center
More ▾
Settings
Help ▾
Like
Follow
Share
+ Add a Button
E-shop application
Create Page @Username
Home
Posts
Reviews
See more
Promote
Manage Promotions
Know friends who might like your Page?
Help more people discover your Page by inviting friends to like it.
Search for friends to invite
Matthias Kark
Invite
No Rating Yet
English (US) · Ελληνικά · Shqip · Español · Português (Brasil)
Information about Page Insights Data
Privacy · Terms · Advertising · Ad Choices · Cookies · More ·
Facebook © 2019
Boost Post
Like
Comment
Share
Chat (Off)
ENG US 4:28 AM 5/14/2019

E-shop application

Published by Project app [?] · 1 min ·

Laptop Lenovo
great laptop

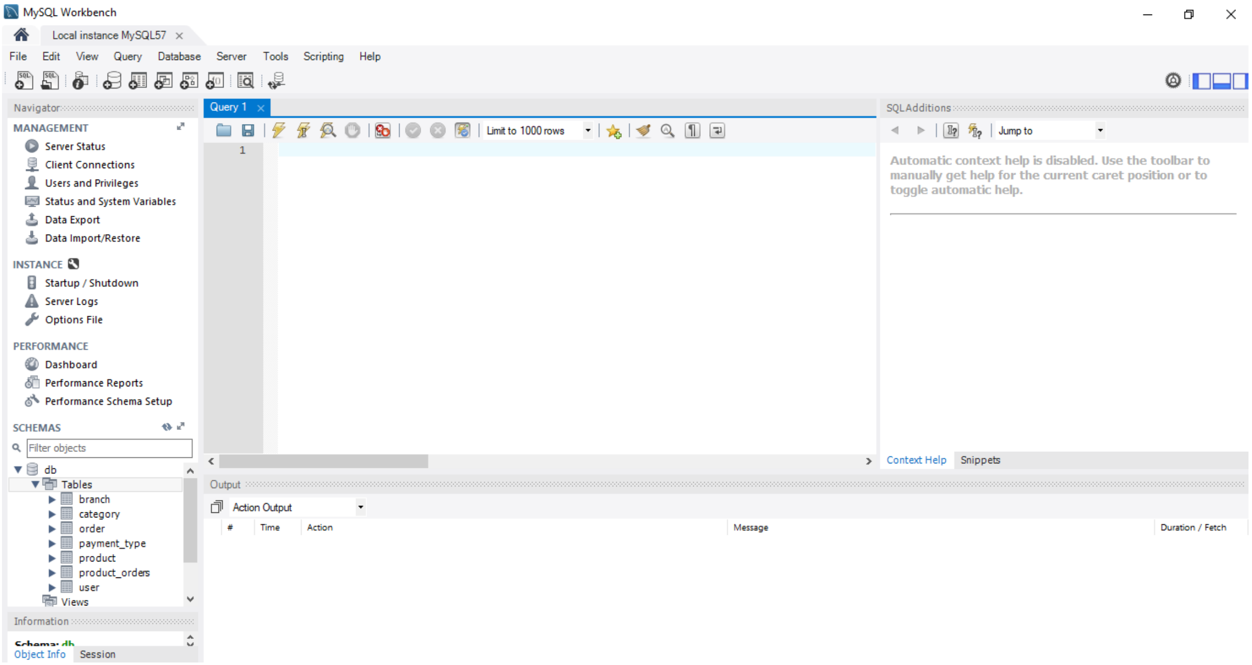
Figure 49:Posted product on Facebook page

Chapter 5

Implementation-Usage of the application

Implementation of database

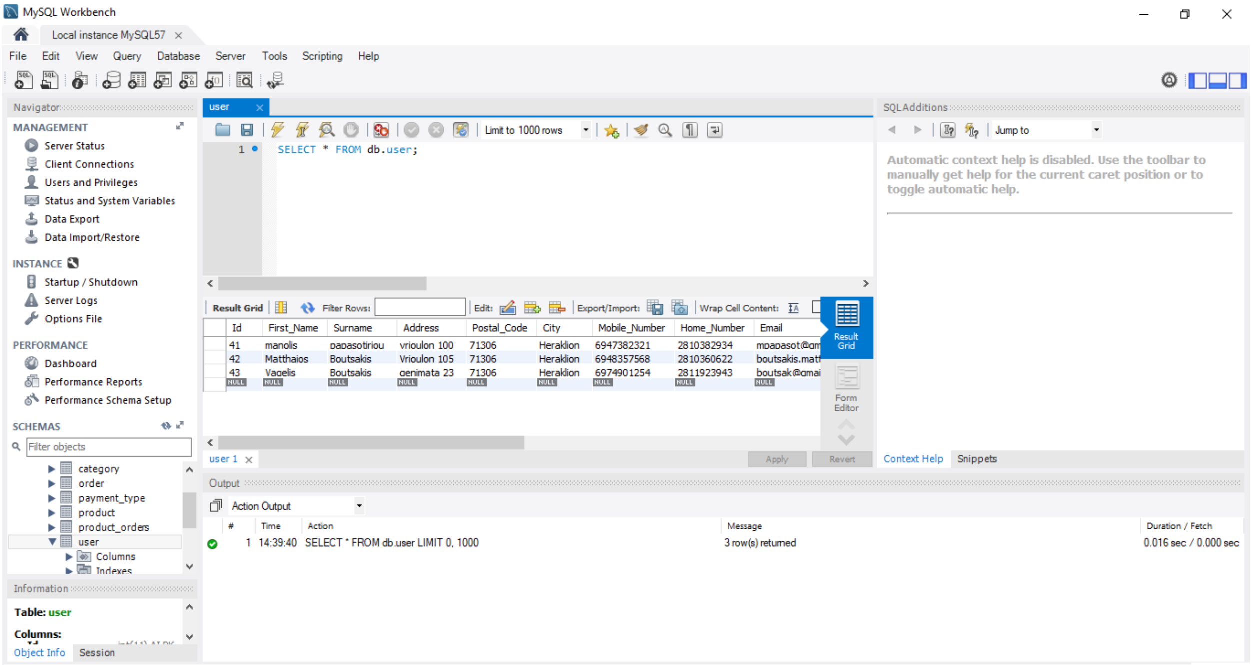
MYSQLWorkbench is the tool that we used in order to manage the database. "It is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single, seamless environment for the MySQL database system. It provides a graphical tool for working with MYSQL servers and databases. MySQL Workbench is available in two editions, the Community Edition and the Standard Edition. The Community Edition is available free of charge. The Standard Edition provides additional Enterprise features, such as database documentation generation, at low cost."[20].



Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Figure 50:MYSQL Workbench database

This is the interface of MYSQL Workbench. The database (db) for our application is left on the bottom.



```
SELECT * FROM db.user;
```

Id	First_Name	Surname	Address	Postal_Code	City	Mobile_Number	Home_Number	Email
41	manolis	papasotiriou	vrioulon 100	71306	Heraklion	6947382321	2810382934	mpapasot@gm
42	Matthaios	Boutsakis	Vrioulon 105	71306	Heraklion	6948357568	2810360622	boutsakis.matt
43	Vagelis	Boutsakis	genimata 23	71306	Heraklion	6974901254	2811923943	boutsak@gmai
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
#  Time      Action                              Message             Duration / Fetch
1  14:39:40  SELECT * FROM db.user LIMIT 0, 1000  3 row(s) returned   0.016 sec / 0.000 sec
```

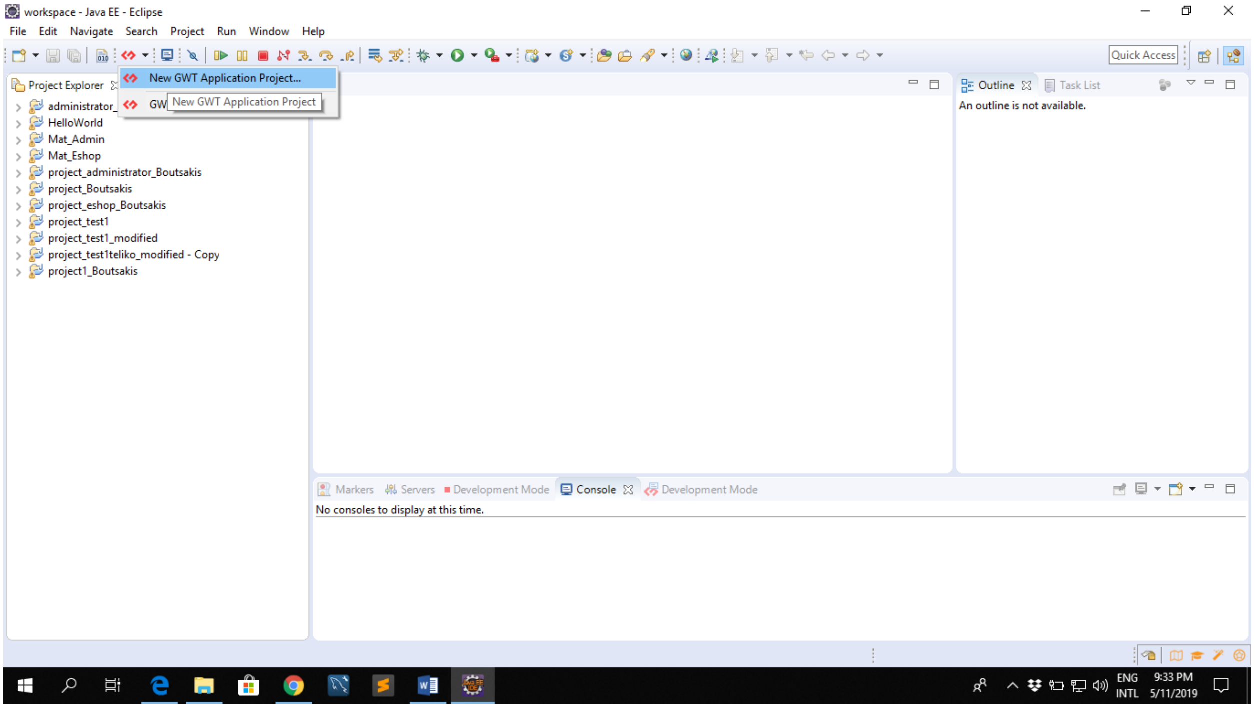
Figure 51:MYSQL User table example

This is an example of how a table is shown in MYSQL Workbench.

Implementation of application

Function Implementation

The eclipse software development environment, supplied with the necessary plugins of GWT, was used for the implementation of the functions of the application. The application was split into two projects so that administrator and clients have different webpages for the application and also different interfaces. So in the administrator application only the administrator can login whereas in the client application registered clients and visitors can have access. The database for the two projects is common. As a result, the process for creation of the projects and their structure are the same. We just name them differently so they can be separated. So we show how we create the administrator project. We click on GWT plugin → New GWT Application Project…

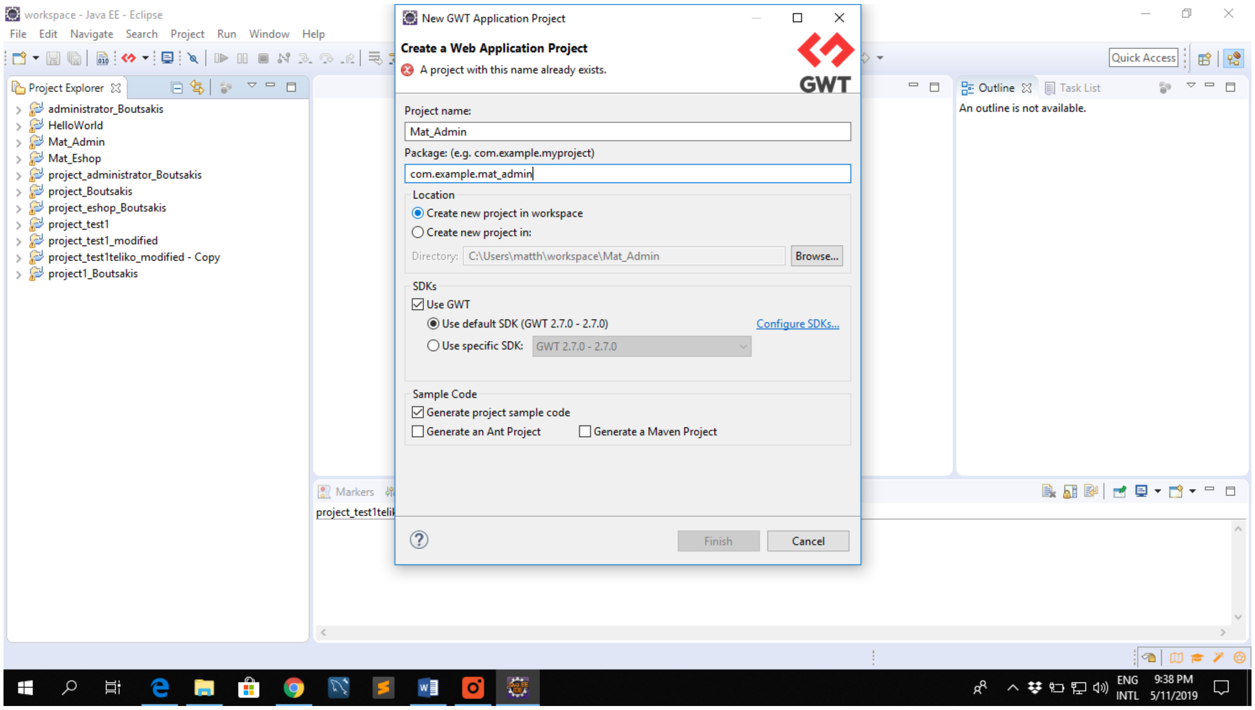


An outline is not available.

No consoles to display at this time.

Figure 52:Click on GWT plugin

Then it appears an alert for completing the settings of the project.



An outline is not available.

Create a Web Application Project

A project with this name already exists.

Project name: Mat_Admin

Package: (e.g. com.example.myproject) com.example.mat_admin

Location

● Create new project in workspace

○ Create new project in:

Directory: C:\Users\matth\workspace\Mat_Admin

SDKs

☑ Use GWT

● Use default SDK (GWT 2.7.0 - 2.7.0)

○ Use specific SDK: GWT 2.7.0 - 2.7.0

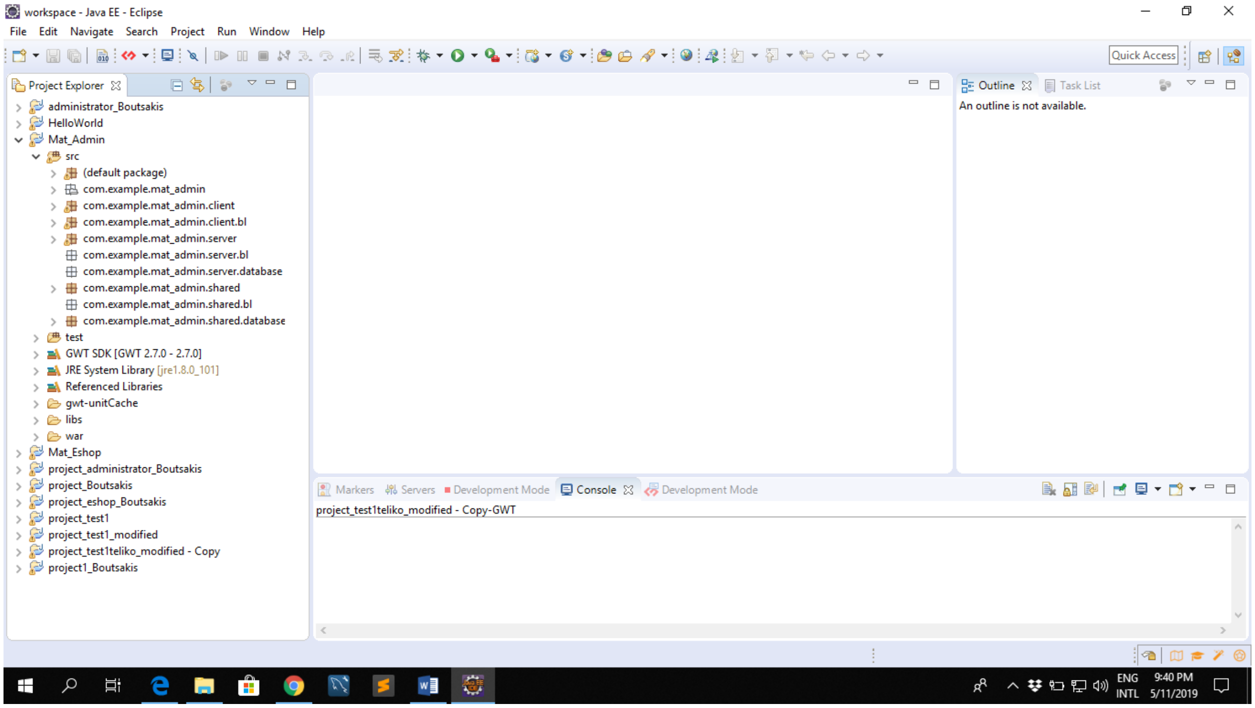
Sample Code

☑ Generate project sample code

□ Generate an Ant Project □ Generate a Maven Project

Figure 53:Complete GWT settings

After having completed the project settings, user clicks on finish button to complete the creation of the project. The created files have the structure that is shown in the following picture on the left side:



```
administrator_Boutsakis
HelloWorld
Mat_Admin
  src
    (default package)
    com.example.mat_admin
    com.example.mat_admin.client
    com.example.mat_admin.client.bl
    com.example.mat_admin.server
    com.example.mat_admin.server.bl
    com.example.mat_admin.server.database
    com.example.mat_admin.shared
    com.example.mat_admin.shared.bl
    com.example.mat_admin.shared.database
  test
  GWT SDK [GWT 2.7.0 - 2.7.0]
  JRE System Library [jre1.8.0_101]
  Referenced Libraries
  gwt-unitCache
  libs
  war
Mat_Eshop
project_administrator_Boutsakis
project_Boutsakis
project_eshop_Boutsakis
project_test1
project_test1_modified
project_test1teliko_modified - Copy
project1_Boutsakis
```

An outline is not available.

project_test1teliko_modified - Copy-GWT

Figure 54: Files structure

Package with client ending includes the functions that are done in client side. On the contrary, server ending includes the functions that are done in server side. The most important files that are created in client side are Mat_Admin.java. Mat_Admin.java includes all interface elements that are available to the end user for the interaction and the execution of the application functions. In this file Mat_Admin class is defined. Mat_Admin class implements Entrypoint interface which determines the application behavior. During the execution of the application, method onModuleLoad() is executed and it determines the structure of the interfaces but it also calls listeners for the responses to the user events.

The following files are included in WAR file:

- Administrator.html: It’s the webpage that contains the application, in which the interfaces that are defined from Mat_Admin file are put.
- Administrator.css: Defines the styles of the webpage.

- fbconf.conf: It saves the parameters that the administrator puts for Facebook posts of the products (page ID, page name, access token, period)

All the functions of the application perform transactions with the database sending queries. These queries are managed by the database via servlets. So appropriate RPC calls need to be developed by using a callback handler in client side to communicate with the database. Therefore, from the very first beginning, MySQL's capabilities should be added to Java installation. This is accomplished by adding the appropriate database driver. In this case the MySQL Java DB Connector is compatible with the installed version of MySQL. It should be added to the Eclipse class path and also to the class path of the webserver. The steps to add the connector to these class paths are the following:

1. Right-click to the Eclipse project.
2. Click on Properties in the menu that appears.
3. Go to Java Build Path.
4. Add External Jar.
5. Find and select the MySQL connector.

The same file is also copied to WEB-INF -> LIB.

In the next step an interface for asynchronous callbacks has to be developed in the client package. This interface will expand the com.google.gwt.user.client.rpc.RemoteService interface. It will include all the methods which are connected in some way to the database. For each class of objects that these methods handle, these classes must extend serialized class that implements all the methods that are connected to the database. These classes are included in the client part of the application. In asynchronous interface methods it will not return anything (void) and it must be included a parameter of AsyncCallback object.

Each Serialized class should include its features. It should set and get methods, and definitely a default constructor. The server package develops the serialized class which includes the methods that handle transactions with the database. It also extends the RemoteServiceServlet class and implements the asynchronous client interface. Database class in client side contains the connection characteristics of the database as well as the constructor on which this connection is established. In the EntryPoint an rpc provider is required for call and connection for each operation that needs to be done with the database. In the server side there is also the Facebook connector class for connection with restfb API. Methods for post and connection on Facebook interface are included in serialized class. Restfb API should be added to the eclipse class path and it is copied also to WEB-INF -> LIB. It was used restfb-2.19.0.jar.

Implementation of interfaces

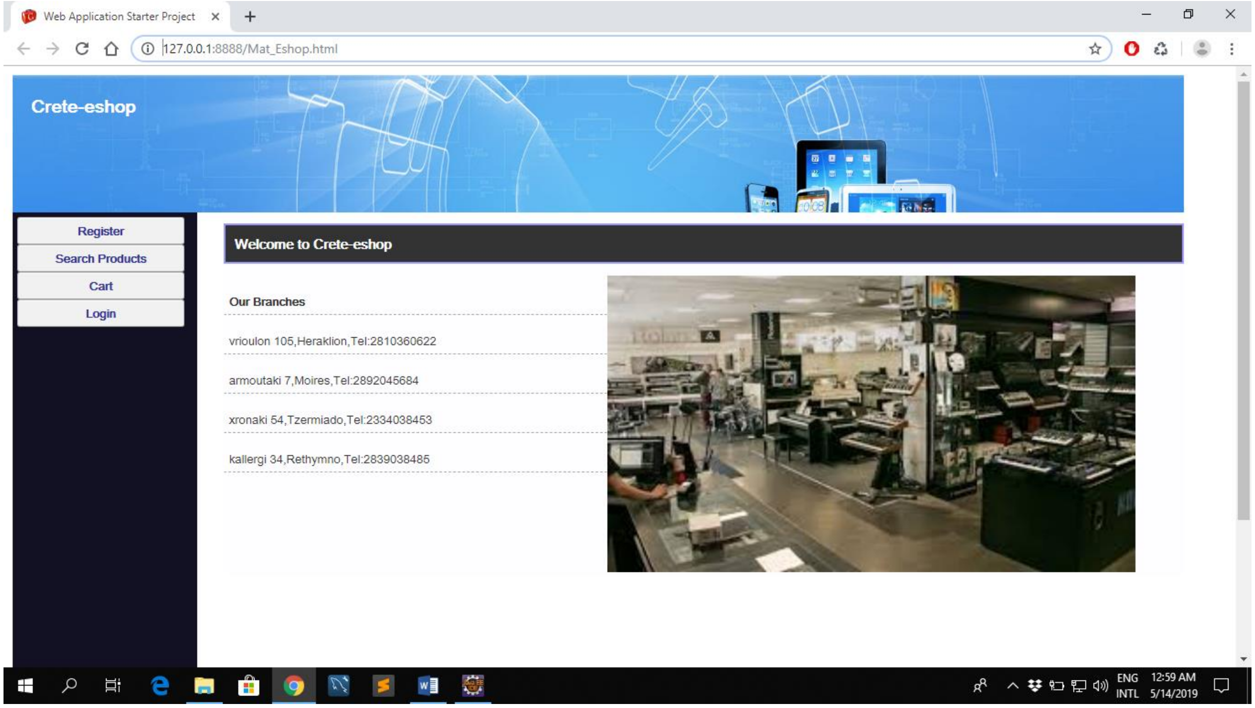
The interfaces of the application were designed using Widget class of GWT. It is the base class for most user-interface objects. It adds support for receiving events from the browser and it is added directly to panels. Some of the important widgets are label widget, html widget, image widget and anchor widget but it can be extended to many other kinds of use like button widget, checkbox widget, RadioButton widget, listbox widget etc.

We can go to the webpage either through running the application from eclipse or entering to the webpage online through web-browser in order to enter to the application. The webpage for the customers or visitors is http://datalab.stef.teicrete.gr:8080/mat_eshop and the webpage for the administrator is http://datalab.stef.teicrete.gr:8080/mat_admin

Usage of the application

Visitor

Upon entering the application, the homepage is presented to visitors.



Welcome to Crete-eshop

Our Branches

vrioulon 105,Heraklion,Tel:2810360622

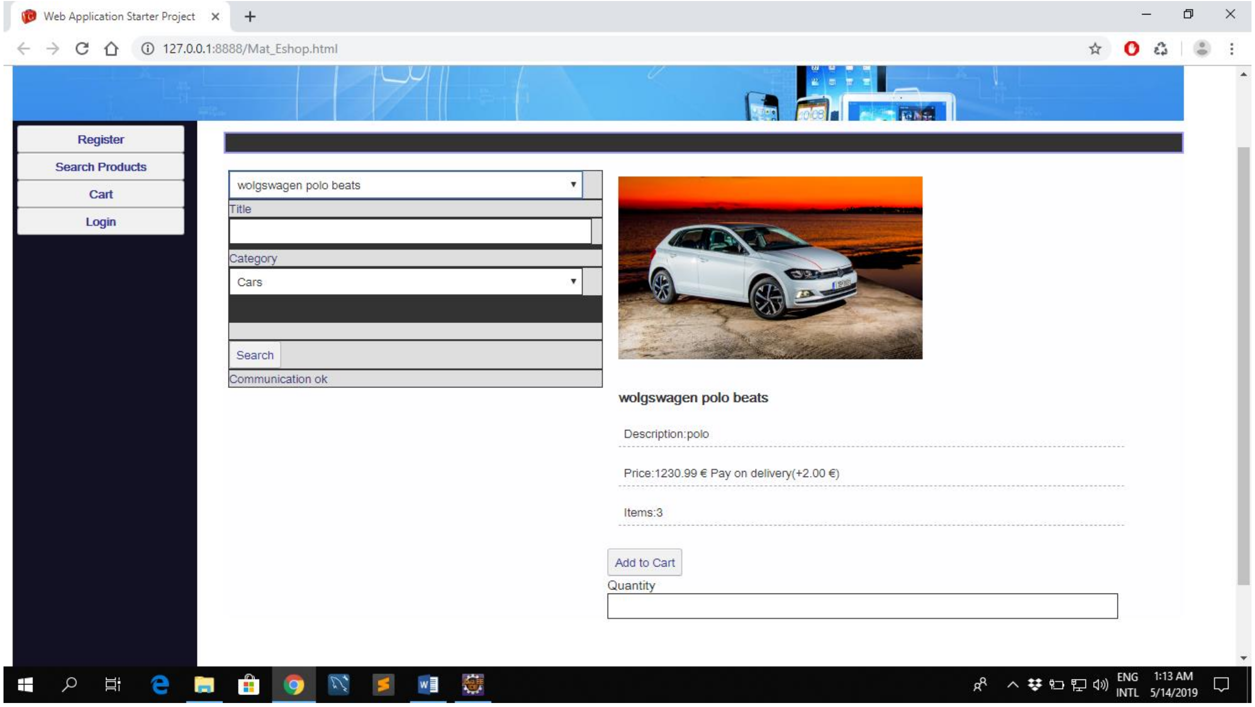
armoutaki 7,Moires,Tel:2892045684

xronaki 54,Tzermiado,Tel:2334038453

kallergi 34,Rethymno,Tel:2839038485

Figure 55: E-shop home page for visitors

The customer can search for products that particularly interest him. For the product search the criteria they can use are the product category and its title (part or whole). By clicking on the Search button, the ListBox at the top of the page includes all the products that meet the criteria.



wolgswagen polo beats

Title

Category

Cars

Communication ok

wolgswagen polo beats

Description:polo

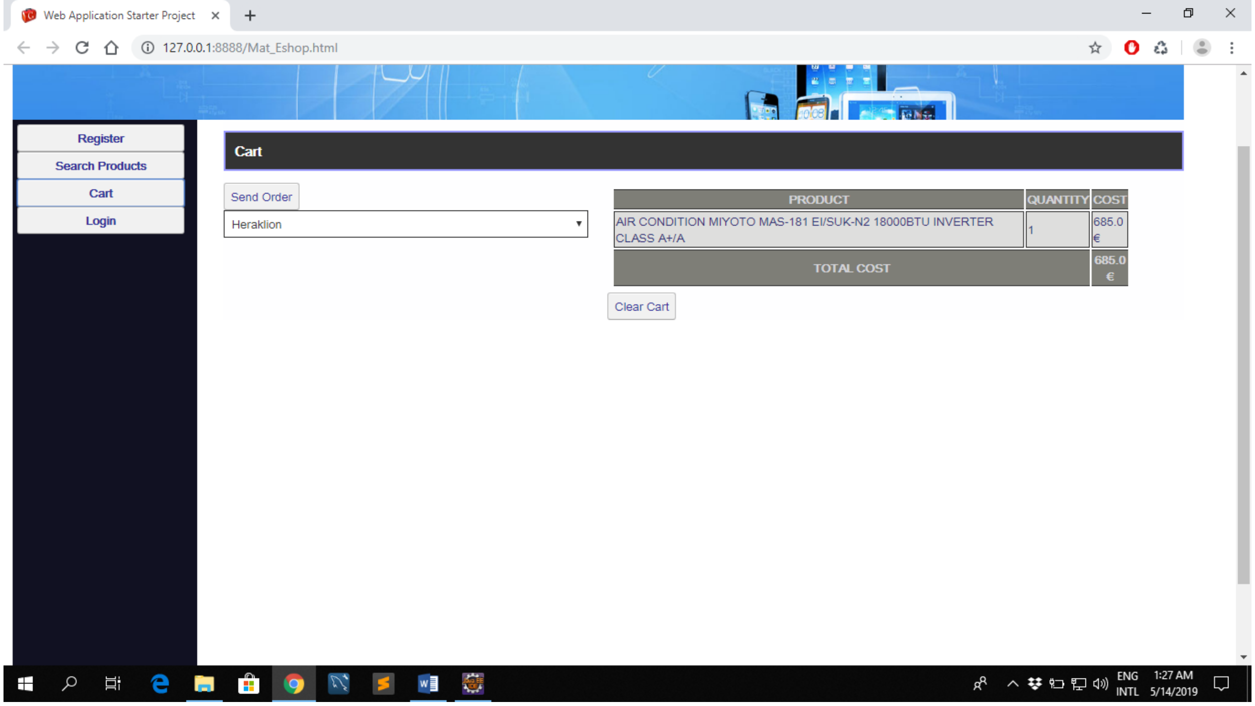
Price:1230.99 € Pay on delivery(+2.00 €)

Items:3

Quantity

Figure 56: Search products page for visitor

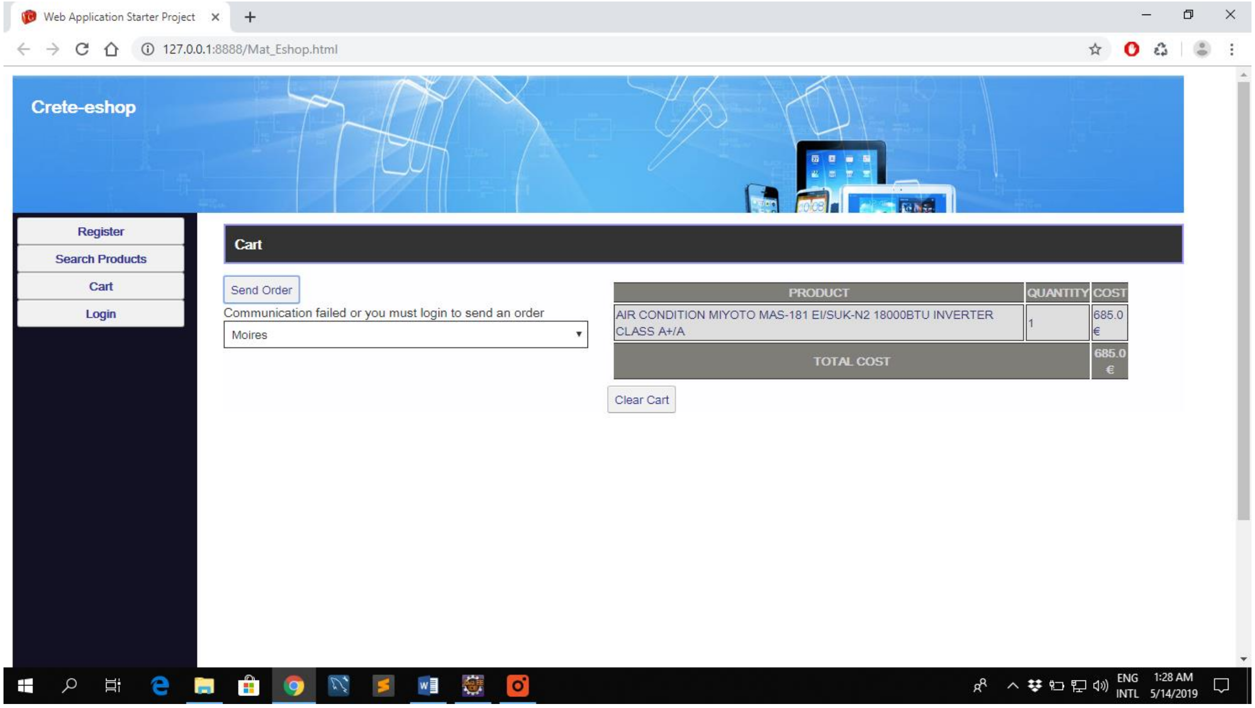
When users select the product they are interested in, the product details are shown on the right side of the screen. At the bottom users can add the product to the shopping cart by filling in the desired quantity and by pressing the Add to Cart button. In order to view the contents of the shopping cart, the users have to click the Cart button. The contents of the shopping cart are then displayed on the right side of the screen. If they have completed the desired add-ons in the shopping cart, they can select the desired pickup branch and click on the Send Order button on the left of the screen.



Heraklion

PRODUCT	QUANTITY	COST
AIR CONDITION MIYOTO MAS-181 EI/SUK-N2 18000BTU INVERTER CLASS A+/A	1	685.0 €
TOTAL COST		685.0 €

Figure 57: View cart page for visitor (1)



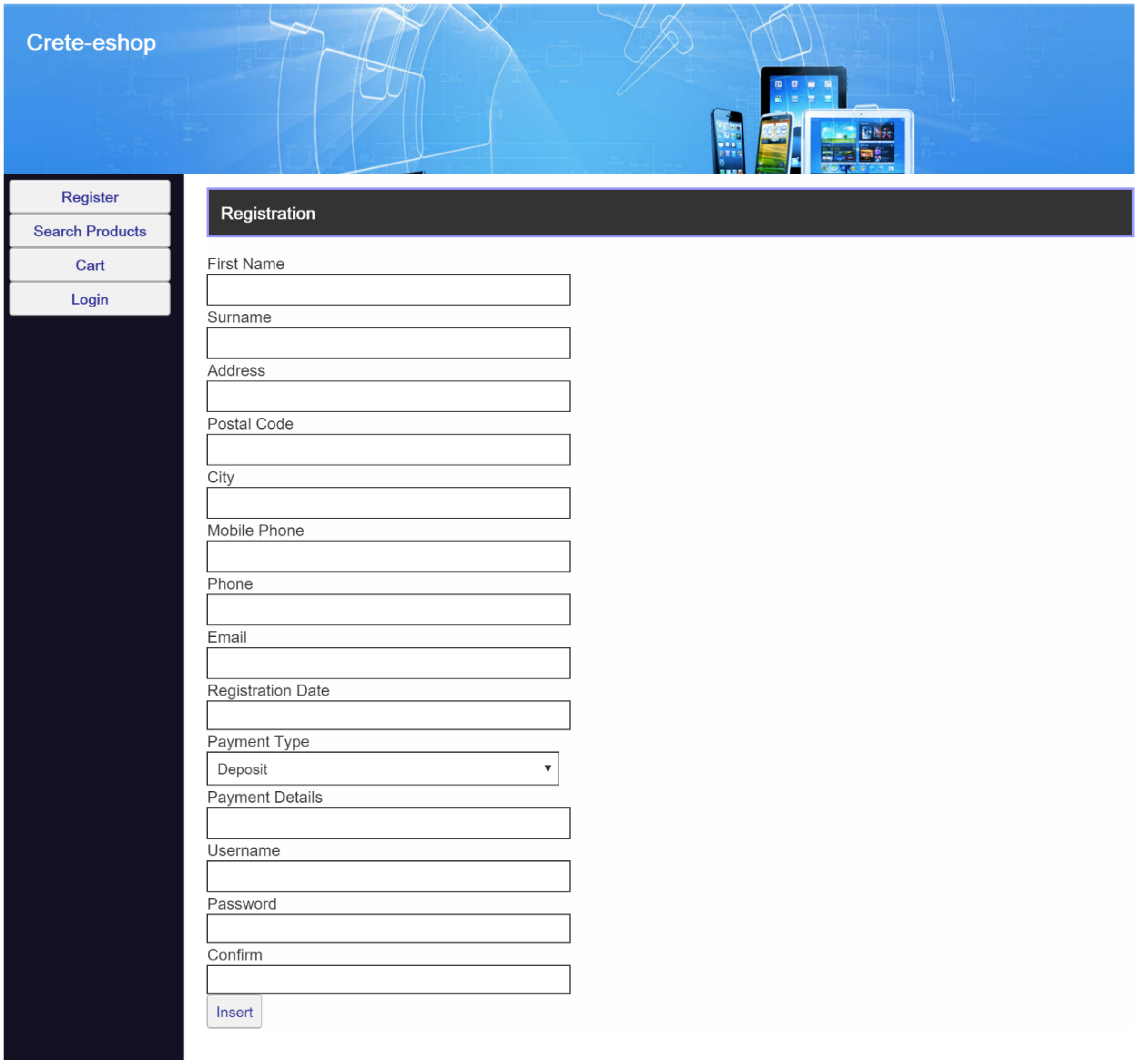
Communication failed or you must login to send an order

PRODUCT	QUANTITY	COST
AIR CONDITION MIYOTO MAS-181 EI/SUK-N2 18000BTU INVERTER CLASS A+/A	1	685.0 €
TOTAL COST		685.0 €

Figure 58: View cart page for visitor (2)

The visitors can’t make a purchase but they can just search for products. So when they want to send an order, a message that says "Communication failed or you must login to send an order" appears. So users must login in order to make their purchase.

By clicking the Register button, a visitor can register in the application as a customer. In this interface a form containing appropriate fields for the registration of the records for the user is displayed. The user is able to enter free text in most fields and there is a choice field to choose his way of payment. Also the field that is used to enter the password hides the text it contains. In case of date registration, a date box appears to the user to choose date of registration. After the completion of the fields, the user clicks on an appropriate button, so their correctness is checked and an appropriate message is displayed. Appropriate messages are shown if they report a registration error. If no error occurs, the data are entered in the database.

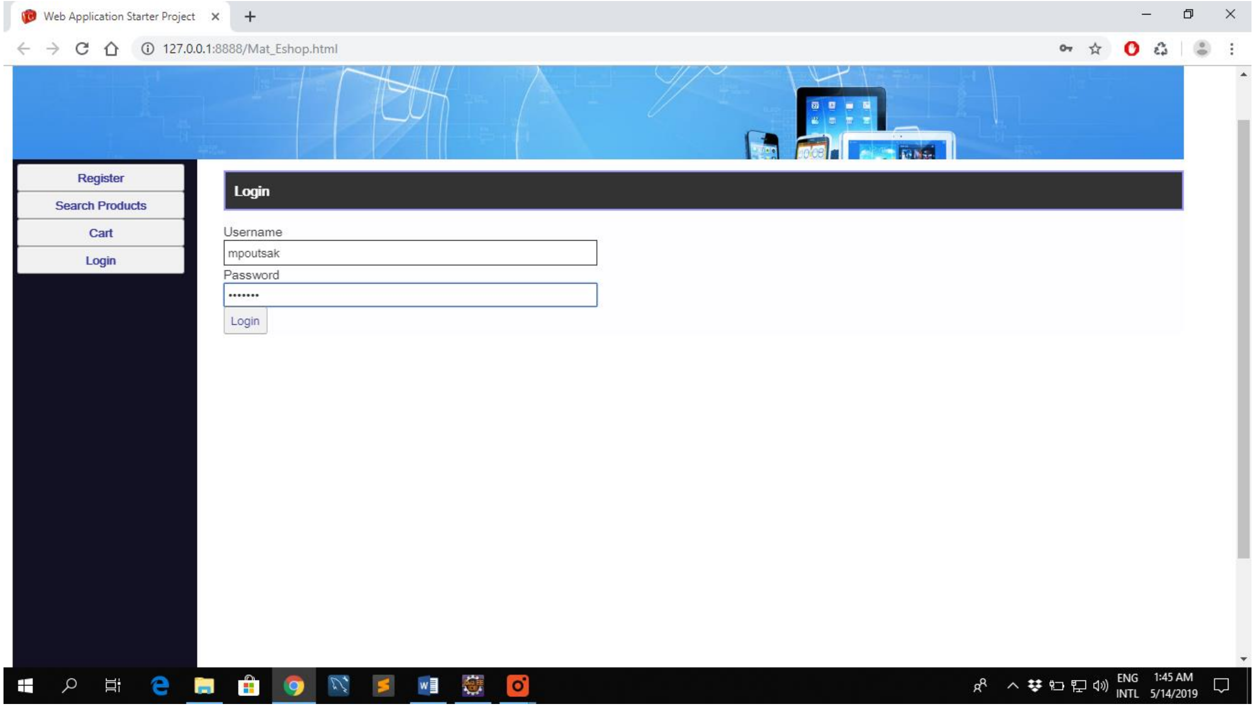


Payment Type
Deposit

Figure 59: Register page

Customer

A customer can enter the application by clicking Login. Then, he/she will need to fill in a username, a password and click the Login button.


Web Application Starter Project
127.0.0.1:8888/Mat_Eshop.html
Register
Search Products
Cart
Login
ENG
INTL
1:45 AM
5/14/2019

Login

Username

mpoutsak

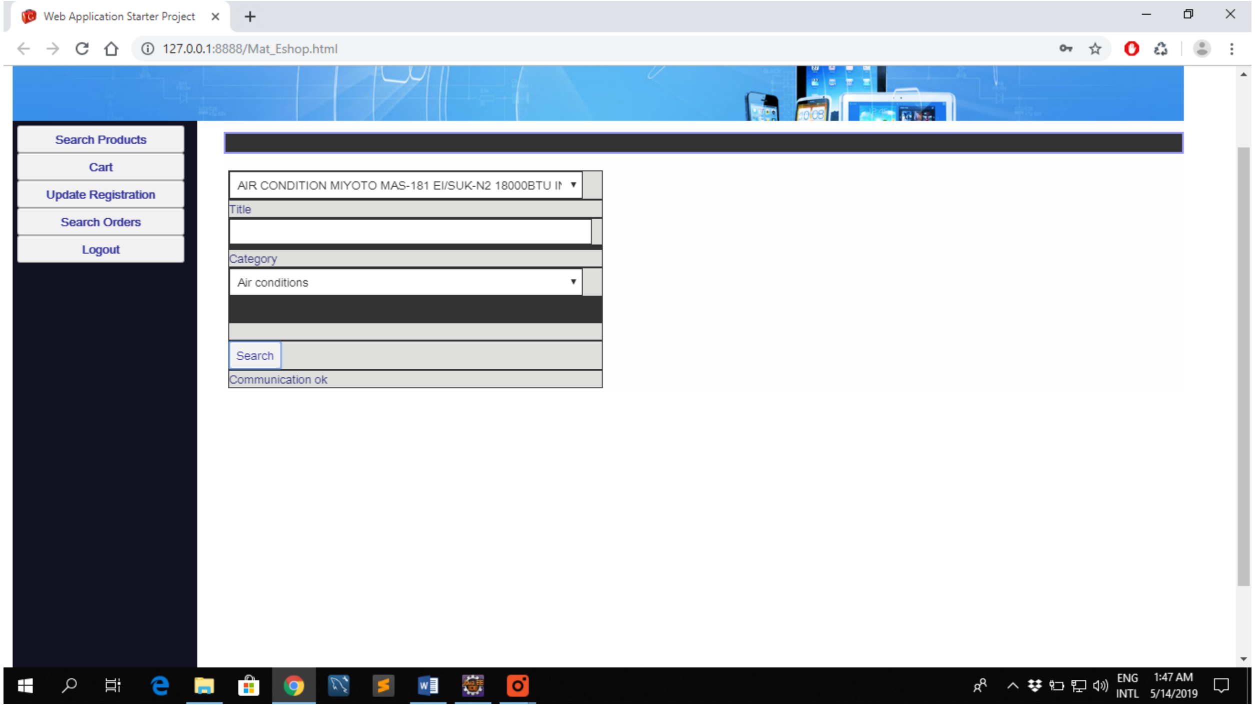
Password

•••••••

Login

Figure 60: Login page for customers

After the entry, the customer can search for products that interest him/her. For the product search the criteria they can be used are the product category and its title (part or whole). By clicking on the Search button the ListBox at the top of the page includes the products that meet the criteria.



AIR CONDITION MIYOTO MAS-181 EI/SUK-N2 18000BTU IN ▾

Title

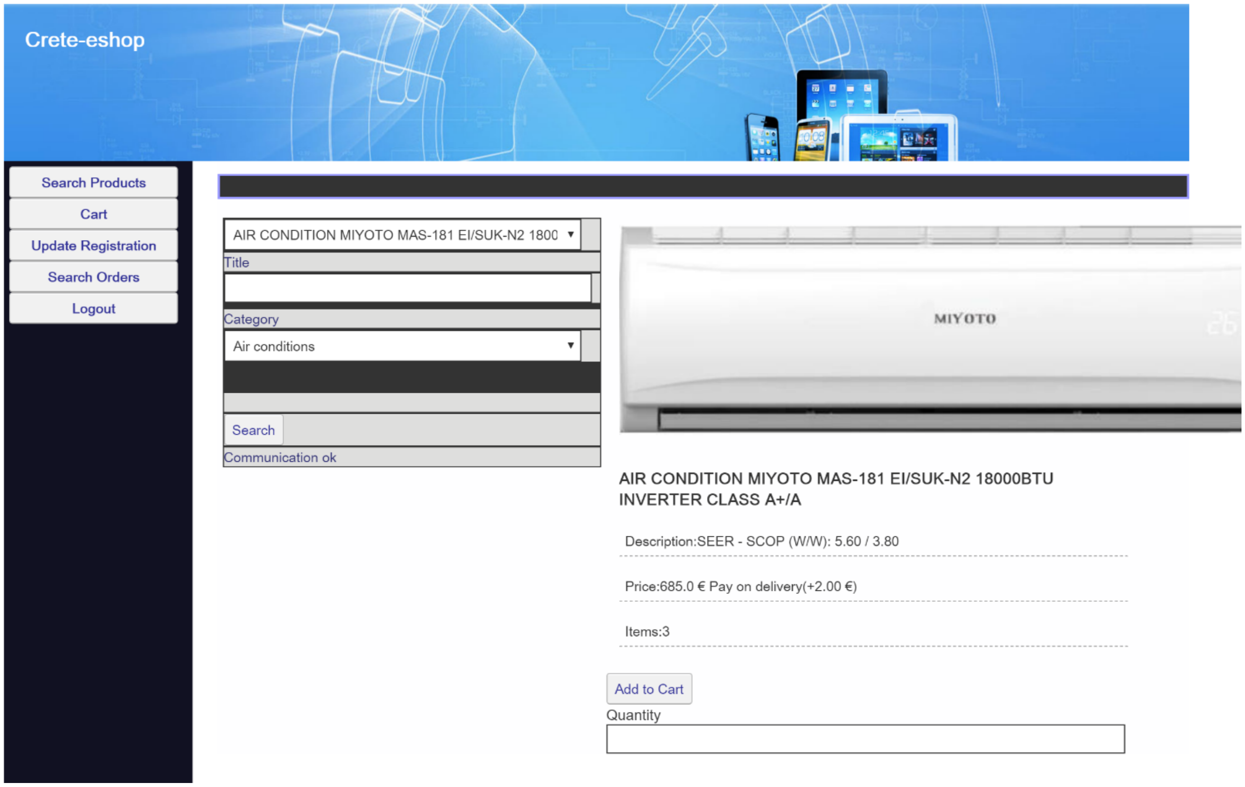
Category

Air conditions ▾

Communication ok

Figure 61: Search products page for customers (1)

When users selects the product that they are interested in, the product details are shown on the right side of the screen. At the bottom, the users can add the product to the shopping cart by filling in the desired quantity and by pressing the Add to Cart button.



AIR CONDITION MIYOTO MAS-181 EI/SUK-N2 1800

Title

Category

Air conditions

Communication ok

AIR CONDITION MIYOTO MAS-181 EI/SUK-N2 18000BTU INVERTER CLASS A+/A

Description:SEER - SCOP (W/W): 5.60 / 3.80

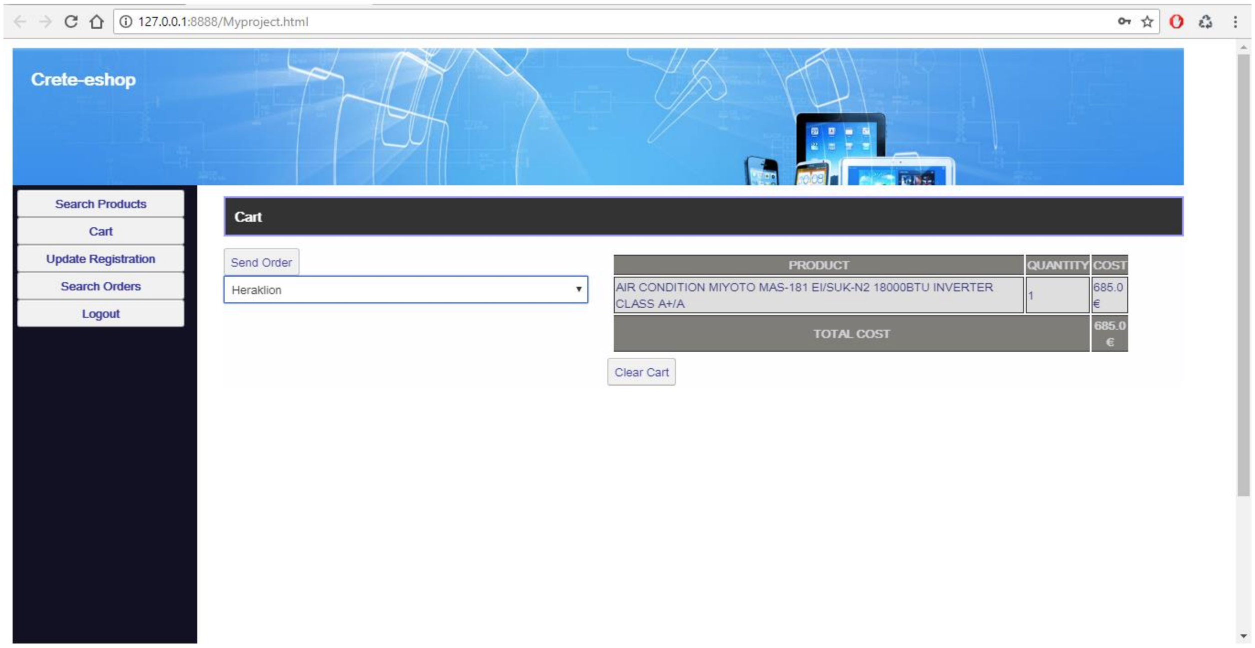
Price:685.0 € Pay on delivery(+2.00 €)

Items:3

Quantity

Figure 62: Search products page for customers (2)

In order to view the contents of the shopping cart, the user has to click the Cart button. The contents of the shopping cart are then displayed on the right side of the screen. If the users have completed the desired add-ons in the shopping cart, they can select the desired pickup branch and click on the Send Order button on the left of the screen.

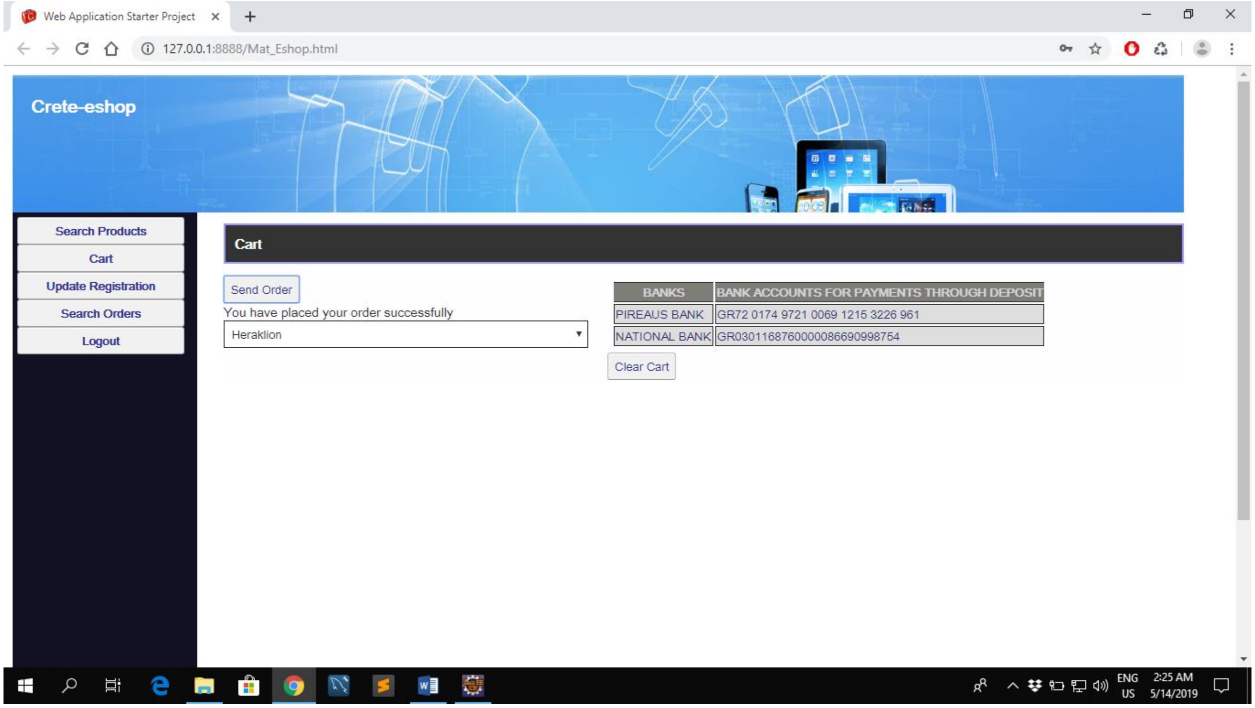


Heraklion

PRODUCT	QUANTITY	COST
AIR CONDITION MIYOTO MAS-181 EI/SUK-N2 18000BTU INVERTER CLASS A+/A	1	685.0 €
TOTAL COST		685.0 €

Figure 63: View cart page for customers (1)

If the users choose Paypal payment type, there will be a PayNow button to enter to Paypal site on the right side of the screen and they will be able to pay. Otherwise if the users choose deposit payment type, there will be a matrix with the payment accounts of the e-shop that the customer can deposit the money. However if the customers want to pay on delivery by cash or credit cards, the users will have just to press the send order button and the purchase will be completed.



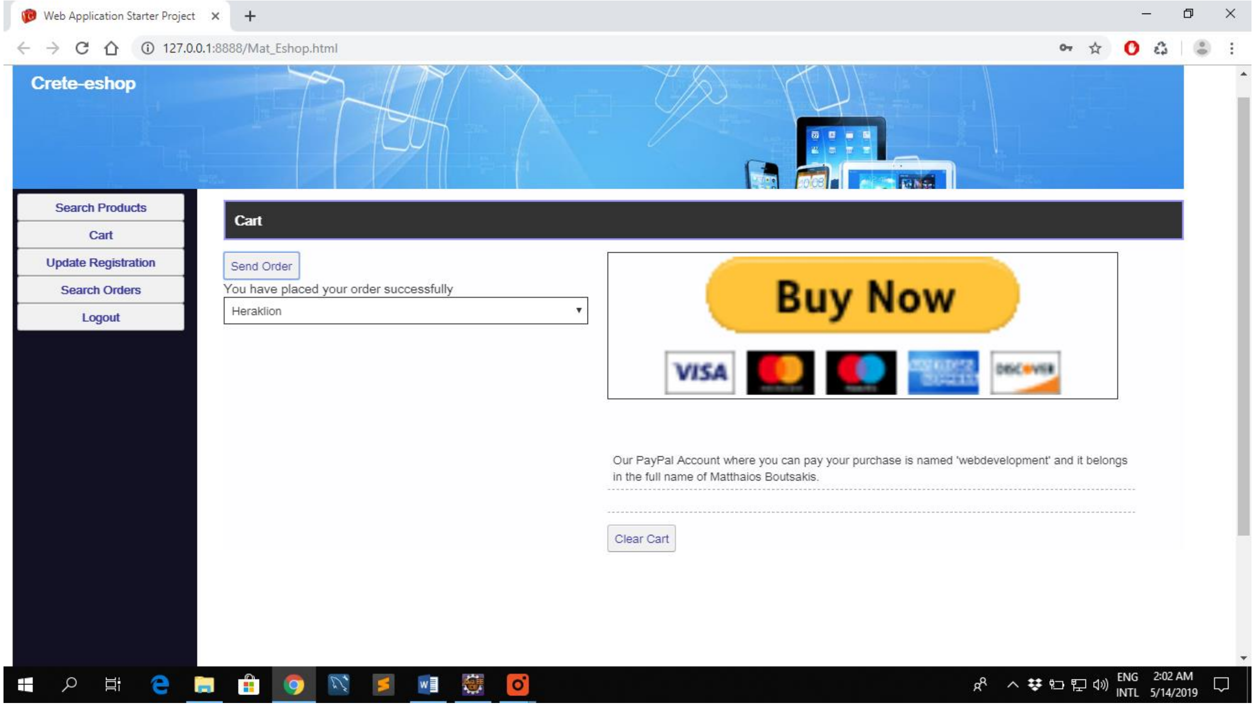
Cart

You have placed your order successfully

Heraklion

BANKS	BANK ACCOUNTS FOR PAYMENTS THROUGH DEPOSIT
PIREAUS BANK	GR72 0174 9721 0069 1215 3226 961
NATIONAL BANK	GR0301168760000086690998754

Figure 64: View cart page for customers (2)



Cart

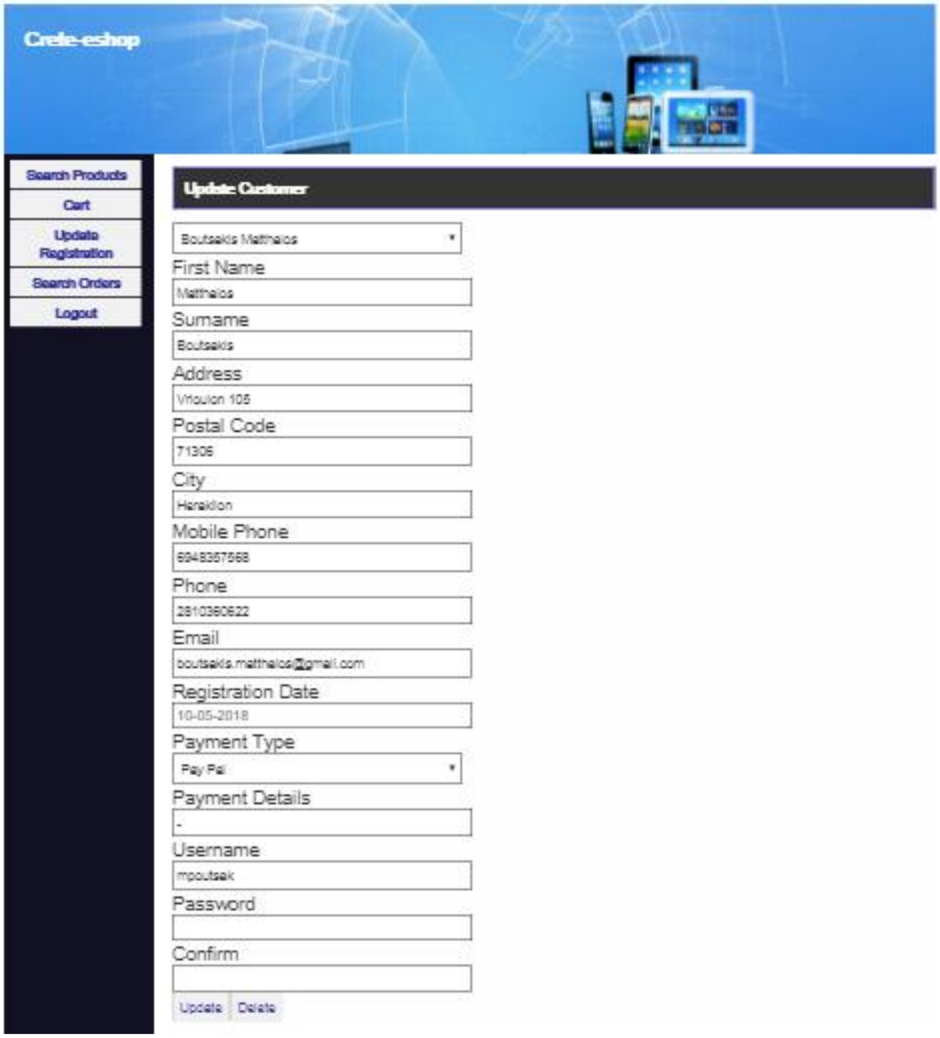
You have placed your order successfully

Heraklion

Our PayPal Account where you can pay your purchase is named 'webdevelopment' and it belongs in the full name of Matthaios Boutsakis.

Figure 65: View cart page for customers (3)

Customers can modify their profile details by clicking the Update Registration button. At the provided screen, they can modify the current entries and click to the Update or Delete button for updating or deleting them.



Boutsakis Matthaios

First Name: Matthaios

Surname: Boutsakis

Address: Vrioulon 105

Postal Code: 71306

City: Heraklion

Mobile Phone: 6948357568

Phone: 2810360622

Email: boutsakis.matthaios@gmail.com

Registration Date: 10-05-2018

Payment Type: Pay Pal

Payment Details: -

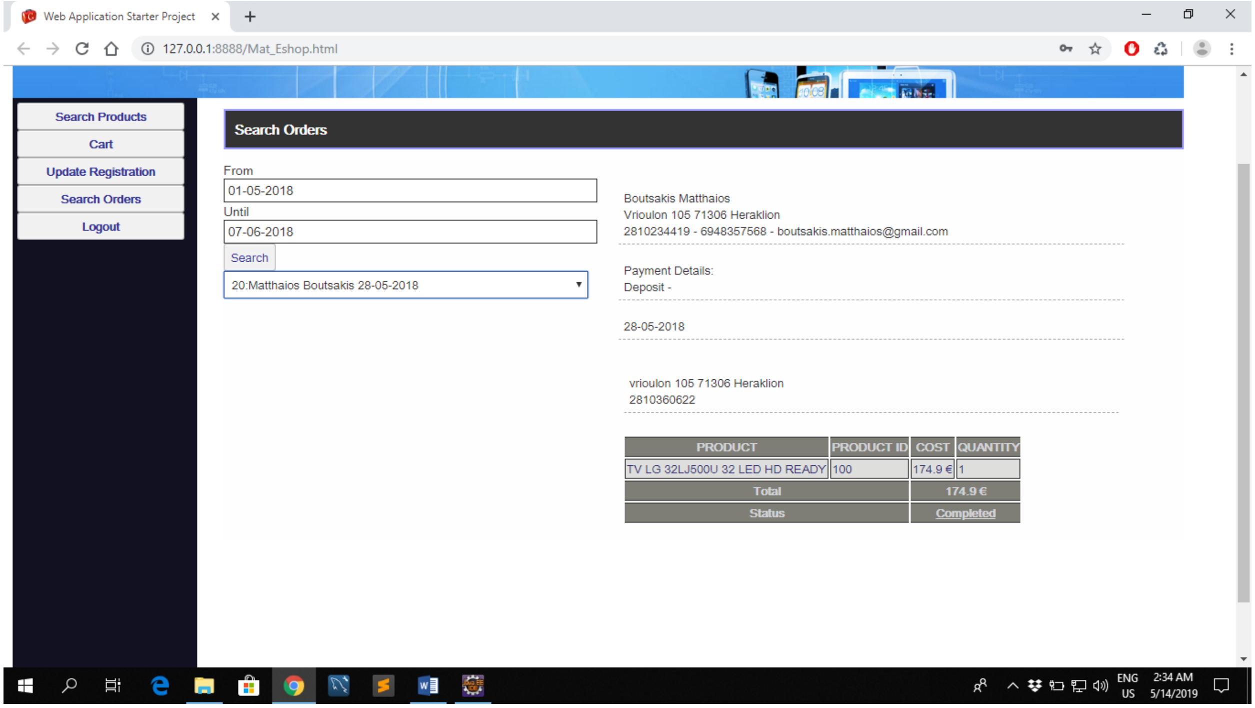
Username: mpoutsak

Password:

Confirm:

Figure 66: Update registration page for customers

The customers can check their orders by clicking the Search Orders button. On the screen that appears, they should select the range of dates to search for orders. After clicking on the search button, the ListBox is filled with the corresponding orders.



From
01-05-2018
Until
07-06-2018
20:Matthaios Boutsakis 28-05-2018

Boutsakis Matthaios
Vrioulon 105 71306 Heraklion
2810234419 - 6948357568 - boutsakis.matthaios@gmail.com

Payment Details:
Deposit -

28-05-2018

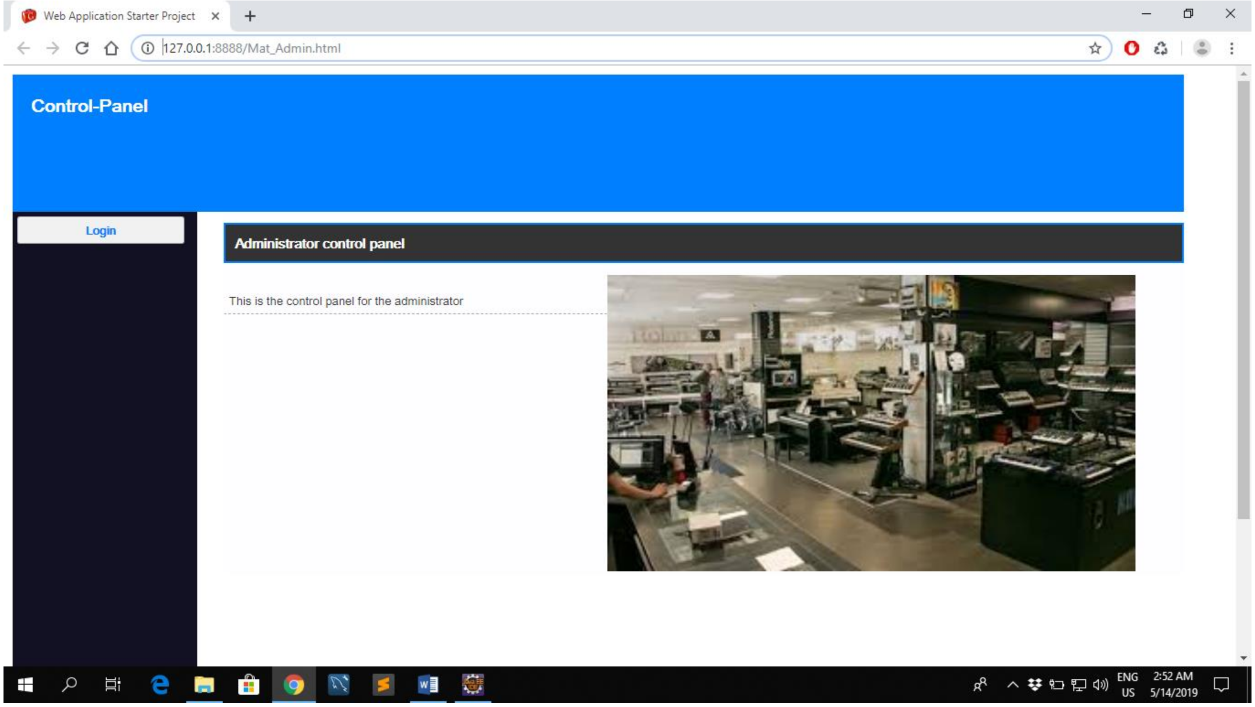
vrioulon 105 71306 Heraklion
2810360622

PRODUCT	PRODUCT ID	COST	QUANTITY
TV LG 32LJ500U 32 LED HD READY	100	174.9 €	1
Total		174.9 €	
Status		Completed	

Figure 67: Search orders page for customers

Administrator

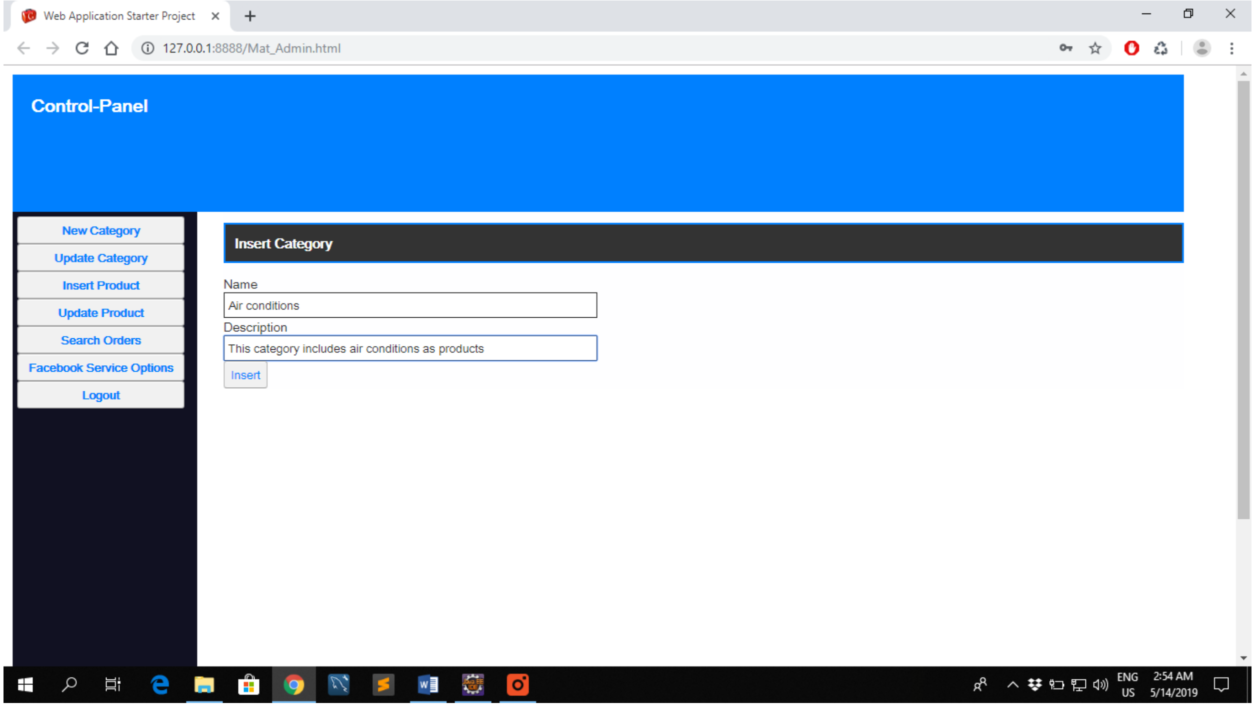
The administrator makes login from a different panel than the customer.


Web Application Starter Project
127.0.0.1:8888/Mat_Admin.html
Control-Panel
Login
Administrator control panel
ENG US
2:52 AM
5/14/2019

This is the control panel for the administrator

Figure 68: E-shop home page for administrator

For inserting a product category, the administrator should click the New Category button. On the screen that appears, he has to type the category details and click the Insert button.



Insert Category

Name

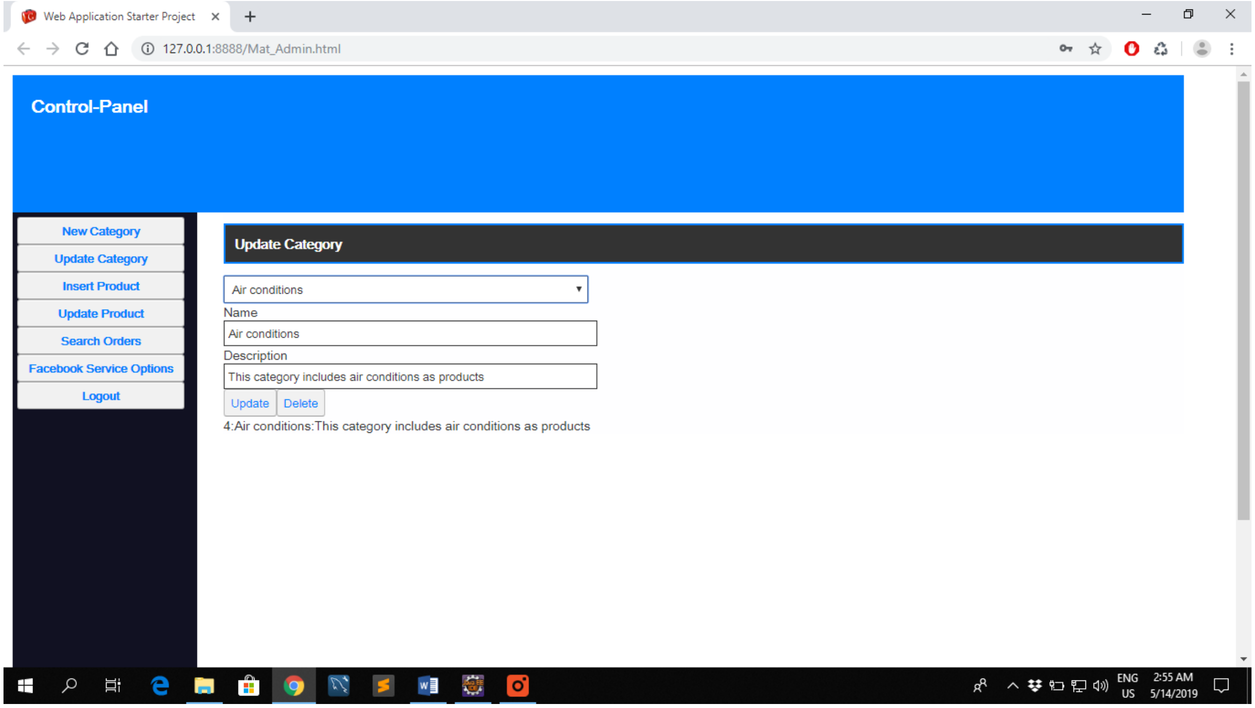
Air conditions

Description

This category includes air conditions as products

Figure 69: Insert category page

In order to update or delete a product category, the administrator should click the Update Category button. In the displayed screen he/she selects the desired category. By selecting it, the current category items are being displayed. The administrator can change these details and click on the Update button or click on the Delete button either to update or even delete it.



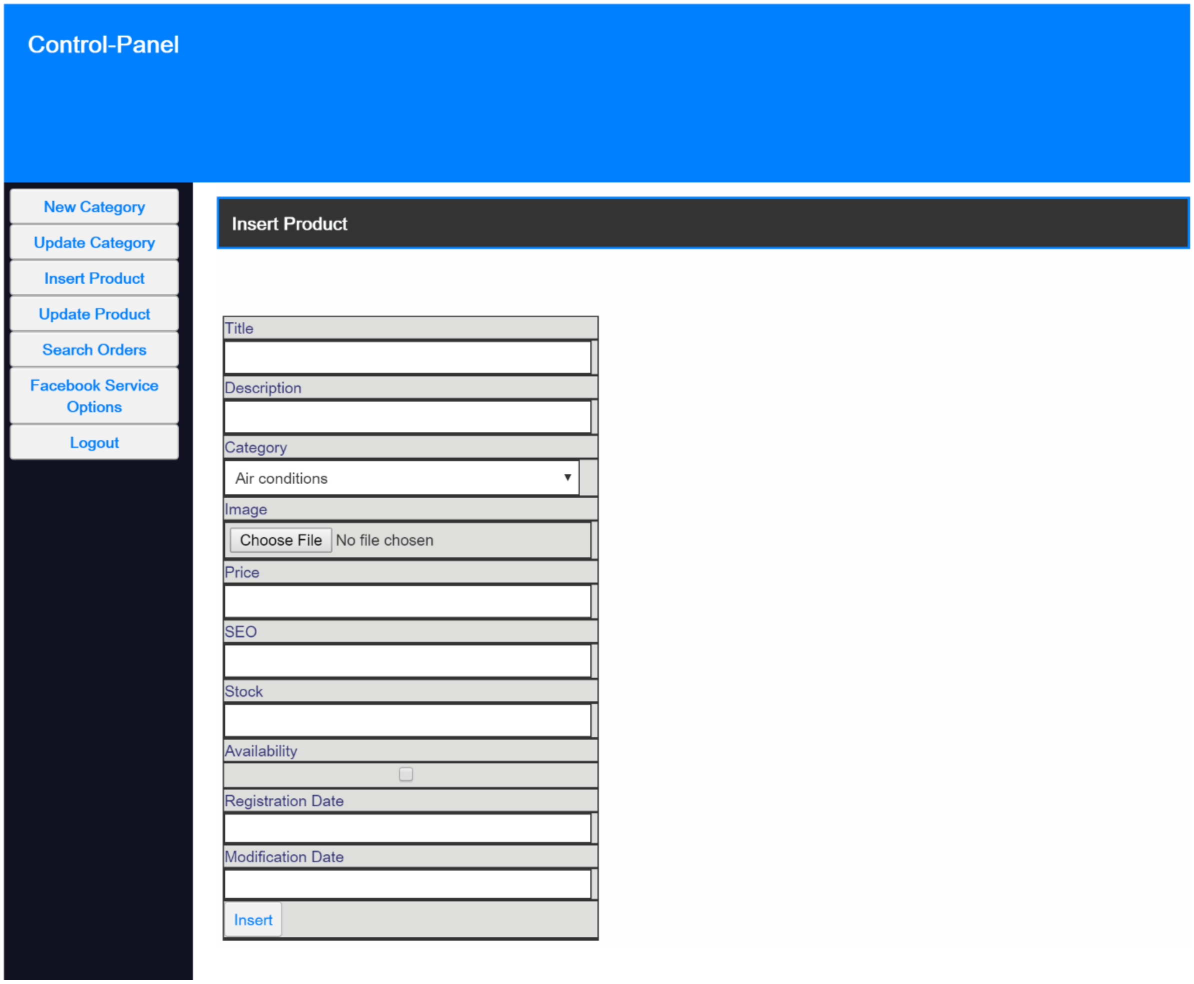
Air conditions

This category includes air conditions as products

4:Air conditions:This category includes air conditions as products

Figure 70: Update category page

To register a product, the administrator must click on the New Product button. On the screen that appears, he/she has to enter the product details and click on the Insert button.



Insert Product

Title

Description

Category: Air conditions

Image: Choose File No file chosen

Price

SEO

Stock

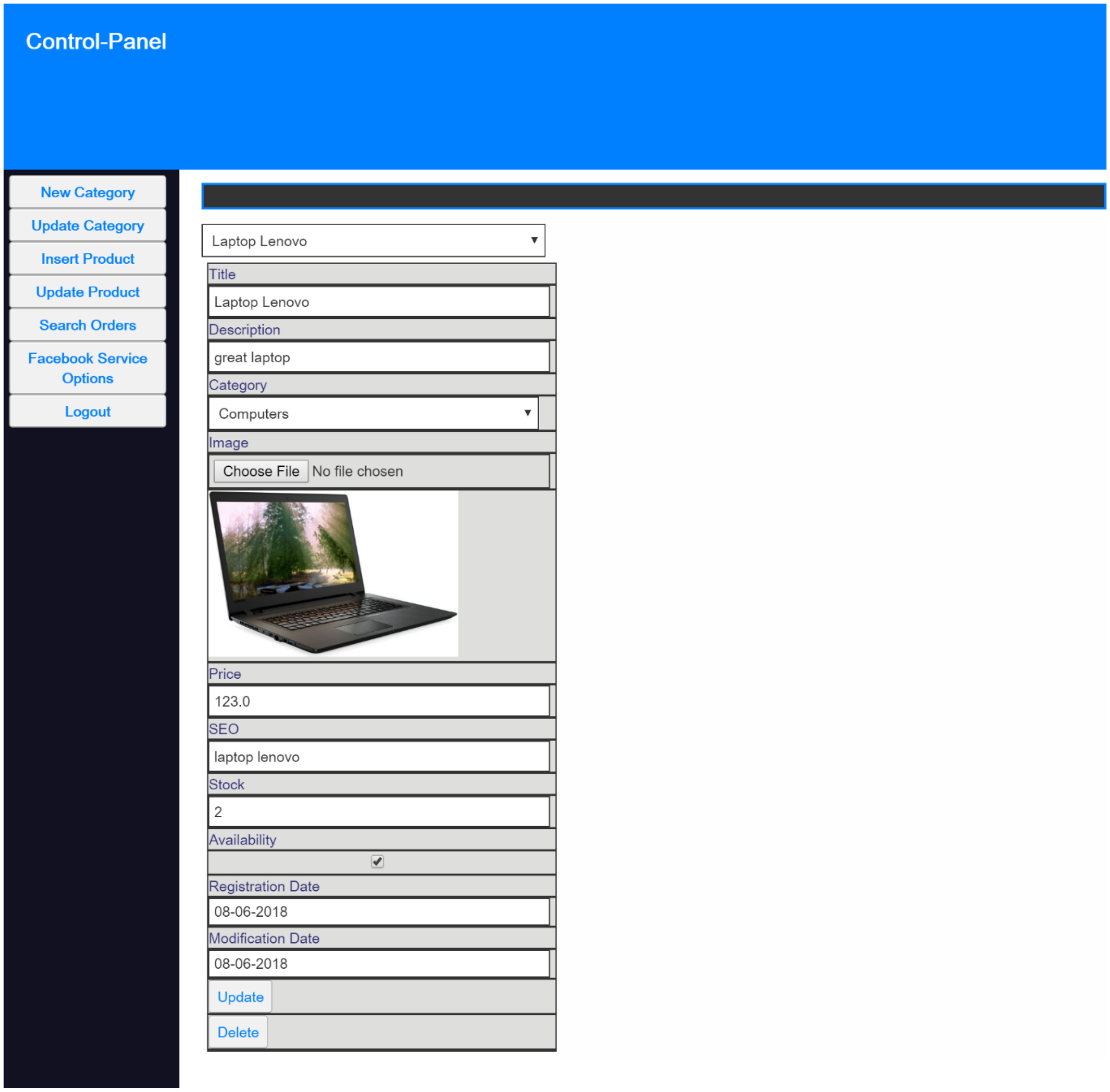
Availability □

Registration Date

Modification Date

Figure 71: Insert product page

In order to update or delete a product, the administrator should click the Update Product button. On the screen that appears, he/she has to select the desired product. The current product details are being displayed. The administrator can change these details and click on the Update button or click Delete to update or delete it respectively.



Laptop Lenovo ▾

Title: Laptop Lenovo

Description: great laptop

Category: Computers ▾

Image: No file chosen

Price: 123.0

SEO: laptop lenovo

Stock: 2

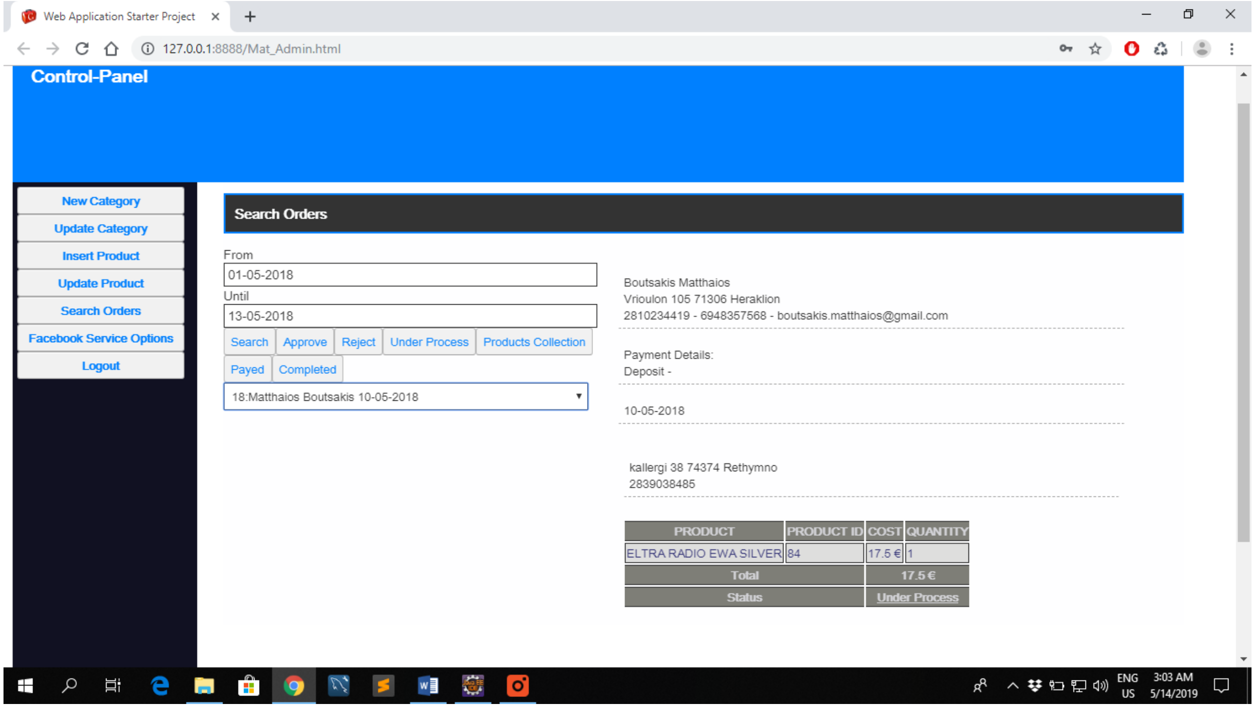
Availability: ☑

Registration Date: 08-06-2018

Modification Date: 08-06-2018

Figure 72: Update product page

The users can check incoming orders by clicking the Search Orders button. On the screen that appears, they should select the range of dates to search for orders. After clicking on the search button, the ListBox is filled with the corresponding orders.



From: 01-05-2018

Until: 13-05-2018

18:Matthaios Boutsakis 10-05-2018

Boutsakis Matthaios
Vrioulon 105 71306 Heraklion
2810234419 - 6948357568 - boutsakis.matthaios@gmail.com

Payment Details:
Deposit -

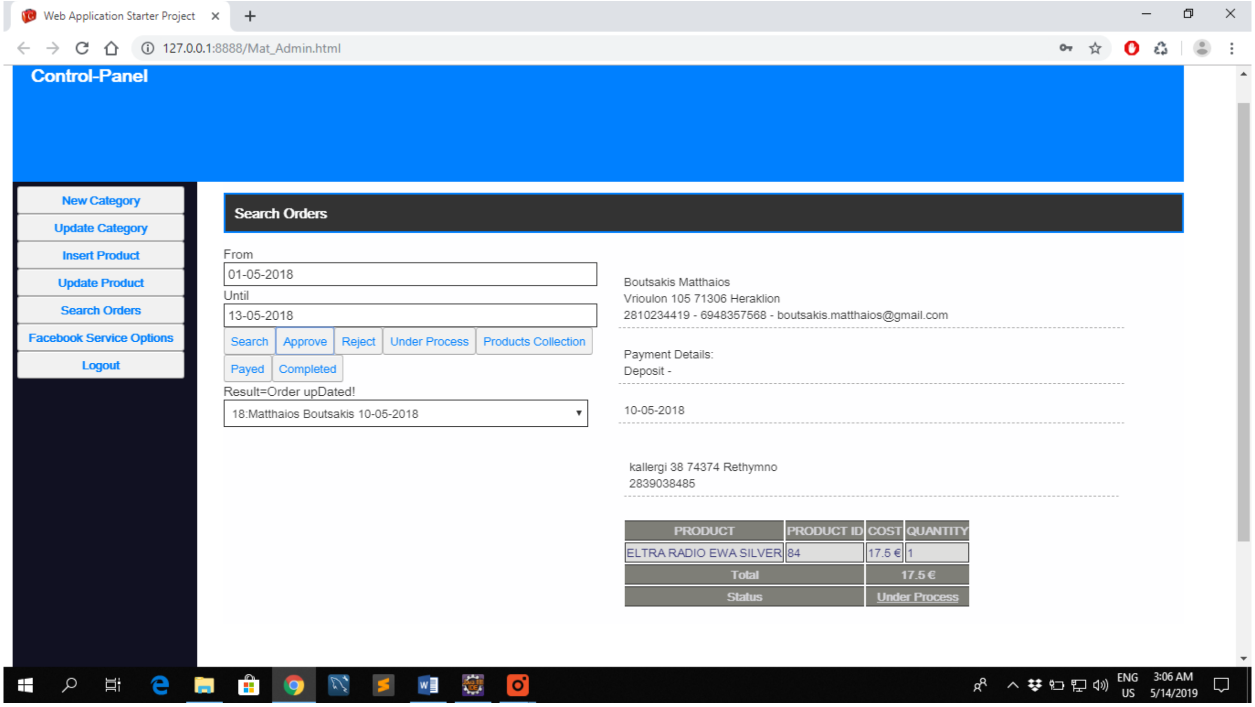
10-05-2018

kallergi 38 74374 Rethymno
2839038485

PRODUCT	PRODUCT ID	COST	QUANTITY
ELTRA RADIO EWA SILVER	84	17.5 €	1
Total		17.5 €	
Status		Under Process	

Figure 73: Search orders page (1)

Then the users can select the order of their interest by clicking on it. The order details appear on the right side of the screen. They can approve it by clicking the Send button or dismiss it by clicking the Reject button.



From: 01-05-2018

Until: 13-05-2018

Result=Order upDated!

18:Matthaios Boutsakis 10-05-2018

Boutsakis Matthaios
Vrioulon 105 71306 Heraklion
2810234419 - 6948357568 - boutsakis.matthaios@gmail.com

Payment Details:
Deposit -

10-05-2018

kallergi 38 74374 Rethymno
2839038485

PRODUCT	PRODUCT ID	COST	QUANTITY
ELTRA RADIO EWA SILVER	84	17.5 €	1
Total		17.5 €	
Status		Under Process	

Figure 74: Search orders page (2)

Chapter 6
Conclusions

The web application which was implemented in this thesis work was designed and implemented by using Rich Internet Applications. Their difference with traditional internet applications is that web application interfaces, which are implemented by using RIAs communication in a transparent way with servers, give the impression to the end user that they are desktop applications and they are friendly and with a better user experience for every user. The navigation for one tab to another is done without loading. Thus, it makes it to look like a desktop application because we have tab changing. For the design of the application, Unified Modeling Language (UML) model was used during the implementation phase.

Rich Internet applications have many patterns for the implementation of applications. One of them is Google Web Toolkit (GWT). Its possibilities can be incorporated in the Eclipse development environment. Programmers can implement different kinds of RIAs with this tool. Its basic advantage is that a programmer has to implement a web application using only Java code. During implementation it is needed in just a few cases to implement html, javascript, css code etc. Also GWT has all libraries or scripts that are needed to provide functionality and aesthetic to the application interfaces. As a matter of fact, external search is not needed to be done to download them. Undoubtedly, this is a big advantage for programmers who are familiar with java language because they don’t need to implement the application by using different programming languages and different development environments.

Plugins, which are available in Eclipse, provide powerful mechanisms for compiling and debugging. So programmers can compile and run the application just with some clicks in the development environment and they don’t need to write commands just like they do in other environments. Also Eclipse's auto-fill capabilities can be used while writing java code as well as suggestion engines for fixing errors. These mechanisms release developers from searching for the appropriate expressions and commands every time and therefore making them much more flexible and fast in implementing applications. Additionally, since java language is used in GWT, all advantages of object oriented programming can be used because java is an object oriented programming language. In this way, each application interface was implemented by using classes and objects. An important tool that GWT provides is GWT designer for the design of interfaces. It contains a complete toolbox that includes all possible interface elements and gives the possibility of high-level presentation of application functions.Consequently, designing interfaces is done easily and quickly and in a way that the programmer is allowed to control how the interface will look like immediately.

On the other hand, the basic disadvantage of GWT is that it can be used only for medium or high level applications due to its complexity. There are times when the programmer tries to run the application and it can't run due to bugs' presentation in all programs that it should not normally exist. This forces programmer to delete application from Eclipse and install it there again on the condition that zip file of the application should be saved somewhere in the computer. So if the programmer needs to implement a low level complexity application, he/she needs to search for other programming tools that are suitable for implementing such applications.

For uploading products of the application on social media (Facebook), RestFB API tool was used. It was used because it was the simplest tool comparing with Facebook4J and Spring Social Facebook. It is minimal and developers need to use only two methods, a method to fetch information from and a method to publish information. A mechanism for automated and periodic posts on Facebook was used and the administrator can fill Facebook mechanism settings in a specific interface in the application for that purpose. He completes page id, page name, access token and period of posting. After activating the Facebook product link, this link is posted on the Facebook page of the application and after the period when the link was set, it is posted again showing a different product. A picture of the product is shown and the title and a description of the product in the image description field on Facebook are written. Therefore due to its simplicity, RestFB is the best API tool for posting content on a Facebook page. It can be used for every kind of web application and it is the best tool for promoting the application on social media.

Chapter 7

References

[1] R. V. Siasoco, "Infoplease," Sandbox Networks, [Online]. Available: https://www.infoplease.com/history-internet.

[2] David W. Wilson, Xiaolin Lin,Phil Longstreet and Saonee Sarker, "Web 2.0: A Definition, Literature Review, and Directions for Future Search," in *Seventeenth Americas Conference on Information Systems*, Detroit,Michigan, 2011.

[3] "Ukessays," 2 March 2018. [Online]. Available: https://www.ukessays.com/dissertation/examples/information-technology/web-2-0-can-be-described.php.

[4] M. WOLCOTT, "CBS NEWS," CBS Interactive Inc., 1 May 2008. [Online]. Available: https://www.cbsnews.com/news/what-is-web-20/.

[5] T. O’Reilly, "What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software," in *communications & strategies*, Sebastopol, Mpra, 2007, p. 17.

[6] Marianne Busch and Nora Koch, "Rich Internet Applications State-of-the-Art," Ludwig-Maximilians-Universität München, Munchen, 2009.

[7] Wieslaw Pietruszkiewicz and Dorota Dzega, "The Practical Aspects of Rich Internet Application Development and Quality Factors: RIA–based Decision Support System," *The Practical Aspects of RIA Development and Quality Factors*, vol. N115 020 31/0713.

[8] Dr. SamratVivekanand Omprakash Khanna and Mr Mijal Mistry, "IMPACT OF AJAX IN WEB APPLICATIONS," *International Journal of Advanced Engineering Technology* , vol. III, no. 1, pp. 144-145, 2012.

[9] U. Priss, "Uta Priss," 2013. [Online]. Available: https://www.upriss.org.uk/wp/w10.pdf.

[10] EdBurnette, Google Web Toolkit-Taking the pain out of Ajax, Raleigh,North Carolina,Dallas,Texas: Fridays, 2006.

[11] Daniel Guermeur and Amy Unruh, Google App Engine Java and GWT Application Development, Birmingham: Packt Publishing, 2010.

[12] Google Web Toolkit, tutorialspoint, 2015.

[13] "GWT Plugin Documentation," 3 November 2016. [Online]. Available: http://gwt-plugins.github.io/documentation/gwt-eclipse-plugin/Download.html.

[14] "DBMS," *Equizine-Equitec digital magazine,* 22 Octomber 2012.

[15] CHRISTO WILSON, ALESSANDRA SALA, KRISHNA P. N. PUTTASWAMY and BEN Y. ZHAO, "Beyond Social Graphs: User Interactions in Online Social Networks," *ACM Transactions on the Web,* vol. 6, no. 17, 2012.

[16] S. Social, "Spring," Pivotal Software, 2019. [Online]. Available: https://projects.spring.io/spring-social-facebook/.

[17] "Facebook4J," [Online]. Available: https://facebook4j.github.io/en/index.html.

[18] "restfb," Mark Allen and Norbert Bartels, 2019. [Online]. Available: https://restfb.com/.

[19] MySQL Database Management System, tutorialspoint, 2018.

[20] "Oracle," 2010. [Online]. Available: https://www.oracle.com/technetwork/mysql-hands-on-lab-403032.pdf.