

MINISTRY OF NATIONAL EDUCATION



TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA

FACULTY OF AUTOMATION AND COMPUTER SCIENCE
COMPUTER SCIENCE DEPARTMENT

FINANCIAL EFFICIENCY BOOSTING SYSTEM FOR
CONSUMER-GRADE PRODUCTS

LICENSE THESIS

Graduate: Peter Tibor ZAVACZKI
Supervisor: Assoc. Prof. Dr. Eng. Delia Alexandrina MITREA

2019

MINISTRY OF NATIONAL EDUCATION



TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA

**FACULTY OF AUTOMATION AND COMPUTER SCIENCE
COMPUTER SCIENCE DEPARTMENT**

DEAN,
Prof. dr. eng. Liviu MICLEA

HEAD OF DEPARTMENT,
Prof. dr. eng. Rodica POTOLEA

Graduate: **Peter Tibor ZAVACZKI**

**FINANCIAL EFFICIENCY BOOSTING SYSTEM FOR
CONSUMER-GRADE PRODUCTS**

1. **Project proposal:** A financial efficiency boosting system for consumer-grade products based on an extension for the Google Chrome browser, webcrawling, a web platform and a RESTful webservice.
2. **Project contents:** *(enumerate the main component parts) Presentation page, advisor's evaluation, title of chapter 1, title of chapter 2, ..., title of chapter n, bibliography, appendices.*
3. **Place of documentation:** Technical University of Cluj-Napoca, Computer Science Department
4. **Consultants:** Assoc. Prof. Dr. Eng. Delia Alexandrina MITREA
5. **Date of issue of the proposal:** November 1, 2016
6. **Date of delivery:** July 8, 2019

Graduate: _____

Supervisor: _____



FACULTY OF AUTOMATION AND COMPUTER SCIENCE
COMPUTER SCIENCE DEPARTMENT

**Declarație pe proprie răspundere privind
autenticitatea lucrării de licență**

Subsemnatul(a) **PETER TIBOR ZAVACZKI** legitimat(ă) cu **C.I.** seria **MM** nr. **971552** CNP **1970131245031**, autorul lucrării **FINANCIAL EFFICIENCY BOOSTING SYSTEM FOR CONSUMER-GRADE PRODUCTS** elaborată în vederea susținerii examenului de finalizare a studiilor de licență la Facultatea de Automatică și Calculatoare, Specializarea **CALCULATOARE, ENGLEZA** din cadrul Universității Tehnice din Cluj-Napoca, sesiunea **IULIE** a anului universitar **2018-2019**, declar pe proprie răspundere, că această lucrare este rezultatul propriei activități intelectuale, pe baza cercetărilor mele și pe baza informațiilor obținute din surse care au fost citate, în textul lucrării și în bibliografie.

Declar, că această lucrare nu conține porțiuni plagiate, iar sursele bibliografice au fost folosite cu respectarea legislației române și a convențiilor internaționale privind drepturile de autor.

Declar, de asemenea, că această lucrare nu a mai fost prezentată în fața unei alte comisii de examen de licență.

În cazul constatării ulterioare a unor declarații false, voi suporta sancțiunile administrative, respectiv, *anularea examenului de licență*.

Data

Nume, Prenume

Peter Tibor ZAVACZKI

Semnătura

Contents

Chapter 1	Introduction - Project Context	1
Chapter 2	Project Objectives and Specifications	3
2.1	Introduction	3
2.2	Positioning	3
2.2.1	Problem Statement	3
2.2.2	Product Position Statement	4
2.3	Stakeholder and User Descriptions	4
2.3.1	Stakeholder Summary	4
2.3.2	User Summary	5
2.3.3	User Environment	5
2.3.4	Summary of Key Stakeholder or User Needs	6
2.3.5	Alternatives and Competition	6
2.4	Product Overview	6
2.4.1	Product Perspective	7
2.4.2	Assumptions and dependencies	7
2.5	Product Features	7
2.6	Other Product Requirements	8
Chapter 3	Bibliographic research	9
3.1	RESTful webservices	9
3.1.1	The Spring Framework	9
3.2	Webcrawling	9
3.2.1	Scrapy	9
3.3	Google Chrome extensions	9
3.4	Web development	9
3.4.1	Angular 7	9
Chapter 4	Analysis and Theoretical Foundation	10
4.1	Conceptual Architecture	10
Chapter 5	Detailed Design and Implementation	11

Chapter 6	Testing and Validation	12
Chapter 7	User's manual	13
Chapter 8	Conclusions	14
	Bibliography	15
Appendix A	Relevant code	16
Appendix B	Other relevant information (demonstrations, etc.)	17
Appendix C	Published papers	18

Chapter 1

Introduction - Project Context

The information age has greatly revolutionalized the lives of people all around the globe, bringing along changes that made lives easier, but also more difficult at the same time. The Internet has transformed modern society into homebodies, people who do anything from the comfort of their homes rather than stepping outdoors to complete tasks. People can do it all online : shopping, chatting, paying bills, working, learning, entertaining themselves, even ordering food. Even though life is much simpler than 50 years ago, an average person probably doesn't even know of 50% of the wonders that the World Wide Web offers them, let alone take advantage of it. That said, an increasing number of individuals use the internet every day to purchase what they need, focusing on the average Joe, who open their browser wishing to buy an item, which, in today's world can be an electronic, articles of clothing, or even groceries.

According to a statistic published in 2018, there were 1.66 billion individuals engaging in Business-to-Consumer e-commerce, which, considering an approximate 7.55 billion population count of 2017, this would mean that 21.98% of the global population bought something at least once in 2017 [1]. According to another statistic, surveying the frequency of purchases made by online shoppers, 20% of them did online shopping once a week, 25% once every 2 weeks, 31% once a month, 15% 3 to 4 times every 3 months and 10% once every 3 months [2]. All these 1.66 billion people, making an average of 2 purchases every month generated 2.3 trillion US dollars of revenue in 2017 and the total revenue generated by e-commerce transactions is predicted to hit 4.9 trillion US dollars by 2021 [3].

Humans as consumers, on the other side remained unchanged in two aspects which are relevant to our situation: one being that they will always try to seek the easiest way of doing a task and the other being the nature of constantly trying to spend less on the product they wish to buy. Unfortunately, from these perspectives the modern webshops can be a blessing as much as a burden. They allow you to purchase whatever you need and have it delivered to your doorstep with the minimal amount of physical work necessary. On the other hand they eliminate the option of good old bargaining, but given the vast number of retailers, prices are much more diverse also. And because of the fact that the customer can just as easily click a few times to buy a product from one retailer as from the other, the

greatest way for a consumer to save money on their purchase is to find the retailer with the lowest price. As such, people who intend to save some money on their online purchases end up spending valuable amounts of time looking for a product in the webshops they know, which is a very limited number from the ocean of options they would have. These people usually end up just accepting the price they find in the first few webshops to reduce the time they spend shopping. What's more, some actually trustworthy webshops might not even have a very user-friendly interface, often confusing customers when they wish to purchase a product. This means that the probability of finding the lowest price is minimal and the process is extremely time consuming, inefficient and often tiring.

Online retailers, very similarly to their offline counterparts, have a major target in their activities too, that of reaching as many consumers as possible, and creating customers out of them. This, of course can be done via advertisements, but reaching a great amount of people with an advertisement placed on a high traffic website, such as a social network, can be extremely expensive nowadays. Taking into consideration another factor, such as the possible bad reputation the website on which the advertisement appears can also negatively impact the web retailer's or the product's image in the consumer's subconscious. The final, and probably greatest threat to the advertisement based attempts to reach customers is the usage of ad-blocking software, which modify accessed websites' DOM, completely eliminating elements containing advertisements. According to some studies, 236 million desktop users actively used ad-blockig software in December 2016 and 2.3 million mobile users browsing the web from a browser that blocks advertisements by default, majorly due to the fact that 'too many ads are annoying or irrelevant' [4]. This means that many users wont even get to see the advertisement the retailer is paying for, and even if they do, it's highly probable that it is of such poor quality or so invasive that the user will just get annoyed by it and he will develop a kind of hate for it.

Chapter 2

Project Objectives and Specifications

2.1 Introduction

The purpose of this chapter is to collect, analyze, and define high-level needs and features of this license thesis. It focuses on the capabilities needed by the stakeholders and the target users, and why these needs exist.

2.2 Positioning

2.2.1 Problem Statement

Everyday more and more people are using the internet to buy what they need, from car tires to shoes, from a mobile phone to groceries. Due to the fact that the number of people shopping online increased thus increasing the demand for these opportunities, a naturally companies increased the supply by each creating one or more webshops offering the same product at different prices. This leads to the clients being lost among the options that have, and in a world where you would want to save money at every spending, a tool is necessary for the everyday webshopper to find the best price for their preferred product.

The problem of	looking for the cheapest offer for a product a client wishes to buy
affects	everyday people doing their shopping on-line
the impact of which is	excessive time spent looking for an offer, ending in unsatisfactory results
a successful solution would be	easy to use easily accessible able to track the prices of a set of products across multiple webshops easy to extend to cover other webshops

2.2.2 Product Position Statement

The Financial Efficiency Boosting System (FEBS) comes as a solution to the problem presented in the previous section by the use of webcrawling, a web platform and a Google Chrome extension.

For	customers of webshops
who	need a tool to simplify their search for a good price
The FEBS	is a system which tracks a set of products on online marketplaces
that	stores current data about products and tells the user about the lowest price of the product they are looking at
unlike	compari.ro
The FEBS	will be more accessible

2.3 Stakeholder and User Descriptions

2.3.1 Stakeholder Summary

Name	Description	Responsibilities
Webshop customer	Person who engages in on-line shopping	Find the product they wish to buy and/or track
Online merchant	Retailers of products who have their products tracked in the FEBS	Have the data about their products as clear and accessible as possible
FEBS developer	Person who creates and maintains the FEBS	Create, improve and offer technical support for the FEBS

2.3.2 User Summary

Name	Description	Responsibilities	Stakeholder
Unregistered customer	Person who rarely engages in online shopping	Access the application to find a good offer	Webshop customer
Registered customer	Person who frequently engages in online shopping	Access the application to find a good offer and add products to their favourites list	Webshop customer

2.3.3 User Environment

2.3.3.1 Users

The application is public and the number of users may fluctuate based on the time of the day/week/month/year. The total number of supported users depends on the server.

2.3.3.2 Time limits

The application should be available all the time, except for maintenance downtimes or unpredictable/uncontrollable downtimes such as power outages. A user can use the application anywhere from a minute if he finds an ideal price to a large time period if they wish to add a product to their favourites list to have it in an easily accessible location.

2.3.3.3 Collaboration

The application is used by a single person, anything a user performs with the system should not influence other users' experience.

2.3.3.4 Infrastructure

The application will be accessible from web browsers. For full functionality a desktop based Google Chrome will be necessary, as this is the only browser to which an extension will be developed for further ease of use.

2.3.4 Summary of Key Stakeholder or User Needs

Need	Priority	Concerns	Current solution	Proposed solution
Support a multitude of domains	0	Customers, Merchants	Crawler for each domain	Analyze and implement a crawler for each new domain
Easy to use interface	1	Customers	Web platform	Google Chrome extension which recognizes its environment
Have an up to date database of items	1	Customers, Merchants	Crawlers for each domain	Run the crawling task at regular intervals

2.3.5 Alternatives and Competition

There are similar tools currently available. One of those is *compari.ro*, which also lists the different webshops and prices for an item, but the fact that a client would have to actually access the website technically makes it harder to use than accessing a Google Chrome extension that is always present and provides data at one click distance.

2.4 Product Overview

The Financial Efficiency Boosting System should provide an easy to use way to users to see the best offer for their desired product. This concept can be seen in figure 2.1.

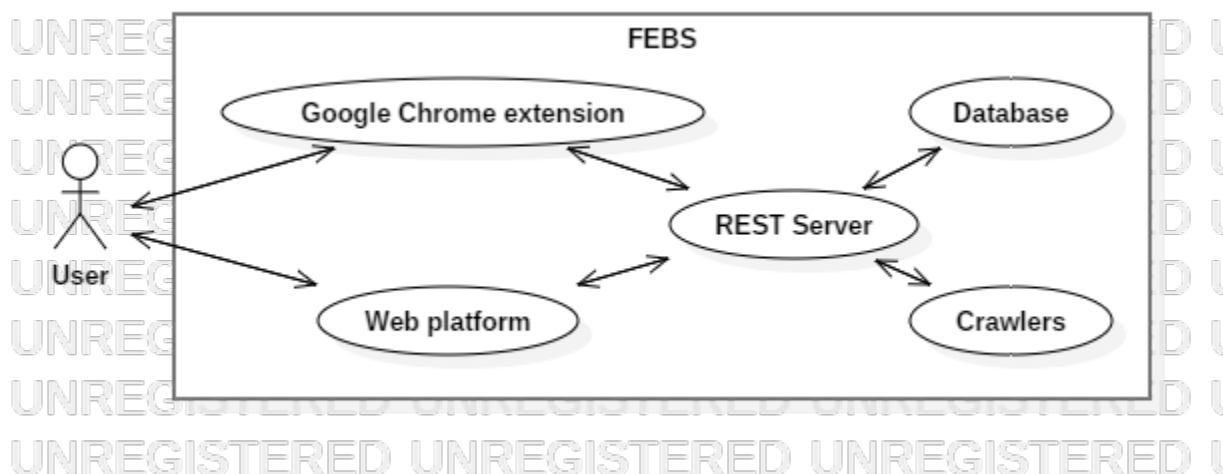


Figure 2.1: The FEBS's system diagram

2.4.1 Product Perspective

Compared to some competitors, such as compari.ro or PriceMon, FEBS aims to further ease the interaction of the user with the system by offering a front-end through a Google Chrome extension. FEBS is a standalone application thus it is not part of a larger system.

2.4.2 Assumptions and dependencies

A client is assumed to have a constant internet connection while using the application.

For the server side:

Database	MySQL
Programming language support	Java 8, Python 3, Angular 7
Frameworks, packages	for Java: Spring framework 2.1.3 or larger, MySQL connector 8.0.13 , Commons codec 1.12 or larger for Python: Scrapy 1.6.0 or larger, requests 2.22.0 or larger for Angular: Angular CLI 7.2.4
Miscellaneous	Node.js, Node Package Manager (NPM)

For the client side:

Web browser	Google Chrome 75.0.3770.100
-------------	------------------------------------

2.5 Product Features

1. Simplistic web platform

The web platform should be as simple as possible so that the user can use and navigate it with maximum ease, while still offering the necessary features to accomplish the necessary.

2. Context recognition

The Google Chrome extension should automatically check if the user opens a link to a supported domain, in which case it automatically activates in case the user accessed the URL of a supported product it can show different things if the webshop sells the product at the cheapest price or not: a message stating that the user is looking at the best offer or a link to the product page of the webshop with the cheapest price.

3. Data Monitoring

The system tracks changes about the supported products and refreshes the already available information and each distinct running of the respective crawlers.

2.6 Other Product Requirements

1. Visually minimalist interface

The front-end of the application should be as clutter-free as possible, thus improving intuitiveness and ease of use.

2. Usability

The application should require close to no input from the user, as it should recognize its working environment to automatically help the user find the cheapest offer. In case user input is necessary, it should be as intuitive as possible to do this, such as interacting with a search bar on the web platform.

3. Performance

The system's performance is measured in the client side response time, which should be at most 5 seconds for most operations.

4. Availability

Besides maintenance downtimes, which should take at most 2 hours each week, preferably in a time period when there is as little user activity as possible, there shouldn't be any availability issue.

5. Scalability

The application should be able to serve many concurrent users without introducing too much stress on the system.

6. Maintainability

The system should be easily maintained, as most problems could be solved by either rebooting one of the services, or fixing an outdated webcrawler.

7. Extensibility

The application should be easily extensible by adding supported products, adding URLs from supported domains to the existing products to gather data about them, or create further crawlers to support more domains.

Chapter 3

Bibliographic research

3.1 RESTful webservice

3.1.1 The Spring Framework

3.2 Webcrawling

3.2.1 Scrapy

3.2.1.1 Spiders

3.2.1.2 Items and Item Pipelines

3.3 Google Chrome extensions

3.4 Web development

3.4.1 Angular 7

Chapter 4

Analysis and Theoretical Foundation

4.1 Conceptual Architecture

Chapter 5

Detailed Design and Implementation

Chapter 6

Testing and Validation

Chapter 7

User's manual

Chapter 8

Conclusions

Bibliography

- [1] J. Clement, Nov 29, 2018. [Online]. Available: <https://www.statista.com/statistics/251666/number-of-digital-buyers-worldwide/>
- [2] ——. [Online]. Available: <https://www.statista.com/statistics/664770/online-shopping-frequency-worldwide/>
- [3] A. Orendorff. [Online]. Available: <https://www.shopify.com/enterprise/global-ecommerce-statistics#1>
- [4] [Online]. Available: <https://www.statista.com/topics/3201/ad-blocking/>

Appendix A

Relevant code

```
/** Maps are easy to use in Scala. */
object Maps {
  val colors = Map("red" -> 0xFF0000,
                   "turquoise" -> 0x00FFFF,
                   "black" -> 0x000000,
                   "orange" -> 0xFF8040,
                   "brown" -> 0x804000)

  def main(args: Array[String]) {
    for (name <- args) println(
      colors.get(name) match {
        case Some(code) =>
          name + " has code: " + code
        case None =>
          "Unknown color: " + name
      }
    )
  }
}
```

Appendix B

Other relevant information
(demonstrations, etc.)

Appendix C

Published papers