

Bucharest University of Economic Studies

Faculty of Cybernetics, Statistics and Economic Informatics

Economic Informatics in English Specialization

**Graduation Thesis**

Coordinating Teacher

Asist. univ. dr. Mădălina DOINEA-ZURINI

Graduate

Ștefania-Denisa CĂLIN

Bucharest 2020



Bucharest University of Economic Studies

Faculty of Cybernetics, Statistics and Economic Informatics

Economic Informatics in English Specialization

Mood- driven travel web application

Graduation Thesis

Coordinating Teacher

Asist. univ. dr. Mădălina DOINEA-ZURINI

Graduate

Ștefania-Denisa CĂLIN

Bucharest 2020

**Declaration regarding the originality of the content and taking responsibility for it**

I hereby declare that the results presented in this paper are entirely the result of my own creation with the exception of references made to the results of other authors. I confirm the fact that any material used from other sources (magazines, books and internet websites) is clearly indicated in the paper and in the list of bibliographic references.

Contents

[1. Introduction 1](#_Toc12628640)

[2. Trip searching progress 4](#_Toc12628641)

[2.1. Trip searching online platform 11](#_Toc12628642)

[2.2. The analysis of existent software solutions 13](#_Toc12628643)

[3. Technologies Used 15](#_Toc12628644)

[3.1. HTML 15](#_Toc12628645)

[3.2. CSS 18](#_Toc12628646)

[3.3. JavaScript 22](#_Toc12628647)

[3.4. Bootstrap 26](#_Toc12628648)

[3.5. PHP 29](#_Toc12628649)

[3.6. MySQL 31](#_Toc12628650)

[3.7. phpMyAdmin 33](#_Toc12628651)

[3.8. WAMP 34](#_Toc12628652)

[4. The solution’s architecture 35](#_Toc12628653)

[4.1. Application’s description 35](#_Toc12628654)

[4.2. Database 36](#_Toc12628655)

[4.3. Diagrams 37](#_Toc12628656)

[5. Mood-Driven Travel’s implementation 41](#_Toc12628657)

[6. Conclusions 45](#_Toc12628658)

[Bibliography 46](#_Toc12628659)

[Annex 1 – The figure list 47](#_Toc12628660)

# Introduction

My graduation thesis’ main objective is the implementation of a software solution in the form of a web application for the travel industry. I have implemented an online application that helps the user to plan a new trip by providing travel ideas, according to the user’s own mood – the way he wants to feel in his future vacation.

Since Internet’s appearance, our world became increasingly connected, available and instantly accessible. The Internet has influenced the travel industry very much. Inventions like the railroads and the airplanes may have found a method over the physical barriers of travel, because of them, now we are able to reach destinations that were unreachable due to the distances and the required time, but the Internet simply transcended them.

The history of the travel industry fascinates through its ancient beginnings, its evolution from one millennium to the next and its flourish during the nineteenth century, and I will present it below.

Our travel history begins with our earliest ancestors, known to science as *Homo erectus,* who lived until about 100,000 years ago. It is known that the species originated in East Africa, but, surprisingly, fossils were discovered in 1891 in Java, an Indonesian island, at about 5,000 miles away. [1] The only plausible answer is that our ancestor gradually left Africa and colonized other parts of the world. Human migration had begun even since, and so the history of travel. Unlike *Homo erectus,* today’s humans travel much further and our willingness to travel and explore allowed us to populate the planet.

Until few centuries ago, travel was a luxury and, if not, it was only made for survival or for the need of new, of unknown territories. The second reason from above drove explorers, merchants, or the various wanderers beyond the world in which they lived, to discover the wonders of the world.

In the 19th century, humankind witnessed the greatest expansion of wealth and technology production in all its history. The barriers lowered and the costs decreased, making travel reachable to millions of people. The combination of these factors, to which we are adding the growth of disposable income, the increased conditions of the burghers determined by the rise of the middle class in many markets and the changing attitudes of people towards travel, have enabled this industry to flourish like never before.

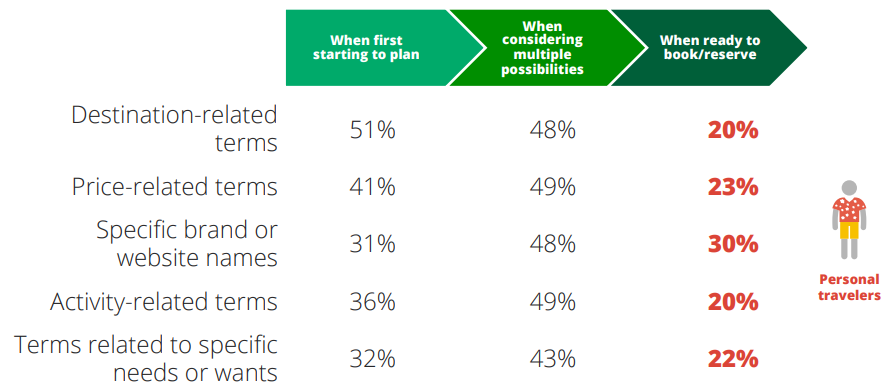
The technological evolution has changed and substantially increased the tourism industry. Today, nobody doubts its influence over the way we travel, from the holiday destination we choose, to the activities we do once we are there.

Constantly in the process to satisfy our needs and wants, it seems that we feel the need to overcome the mental boundaries of everyday existence, which constantly seems to be a “close old clothing”, and to change some details of it. One of the ways to satisfy the need of self-transcendence of a person is traveling, consisting of a process to leave something well-known and understandable for the sake of getting into something unknown until then.   
The choice of the travel parameters, like its route, destination, duration, distance, depth of spiritual immersion in a new environment, the degree of risk to life, depends only on the individual interests of the traveler. The personality of the traveler plays an important role because it depends on which values he wants to acquire – fresh sources of information, sensations, thoughts, experiences, energy, etc.

Due to technology’s impact in our world, the travel business had to adapt and redirect its resources from the physical travel agencies to the cyberspace. Web design plays an important role in visitor’s first impression on the offered services. The design can be a determining factor in whether a visitor trusts the recommendation and the information provided.   
By providing an online application, the virtual travel experience will allow the user to find out all the necessary details and to explore the sights of mountains, beaches and city walks right from the personal computer or phone. In this way, the user’s experience is enhanced, both by the vast field of information, the helpful logistical details, the useful reviews from former travelers and the tips that will help him simplify the process of searching and planning.   
Also, the user has the possibility to access and to save the preferred trips ideas anytime, unlike the face-to-face meetings with the travel companies which are time and energy consuming. The following can be achieved by using Internet, multimedia and databases: a rational and fast communication, direct contact with the market and its business partners, automation and integration of business process, and the most important – delivery and access of information.

Today, a percent of 74% travelers plan their trips on the Internet, while 13% use travel agencies to prepare the vacation for them. The internet is the top source for both business travel and leisure travel planning and, in the planning state, the searches are 48% related to the desired destination and 43% related to specific needs or wants.

The ratio changes when the trip reaches the booking state, the travel website or brand name becomes one of the most relevant search terms and it’s closely followed by price and the specific needs or wants of the person. [2]



**Figure 1.1** Search Terms that Leisure Travelers use in Planning. Source: [2]

The webpage implemented in this project has as main search terms the destination’s location and the psychological impact on the traveler, taking into account the feeling a certain trip idea transmits, which are also the most important factors, according to the studies made by Google Travel.

In the following chapters, the application’s content and development steps will be presented in detail, starting with the ways in which computer science development influences the travel market and continuing with the application’s implementation, the architecture, the technologies used, the obtained results and the conclusions based on them.

# Trip searching progress

Current approaches to the basic notions of travelling

The major changes in the world economy, materialized in significant increases of production in every country and the reduction of political and trade barriers between countries, have led to the travel business development. The number of trips and countries that participate in the travel system raised, enhancing communication, and growing the need of information with travel character.   
On another side, this development brought difficulties in knowledge and correct evaluation of the travelling business, becoming harder to compute a relevant international comparison.

This was the situation which imposed the adoption of a unitary statistical system of tourism and the development of a common terminology, proposed at the International Conference on Travel and Tourism Statistics, Ottawa, 1991. These recommendations were adopted by law in 1993, at the 19th Session of the Statistics Commission of United Nations. [3]

The adopted clarifications concerned wide range of issues that could be grouped on multiple plans, as:

* The tourism notion’s content and its forms;
* The concept of visitor and the place, duration and reason of the trip;
* Tourism industry’s content and classification of composing components;
* Classification of tourist’s activities.

The established forms of tourism are the domestic tourism (only in the country of origin), the inbound tourism (non-residents who travel in the certain country) and outbound tourism (residents of a country who travel in other countries).

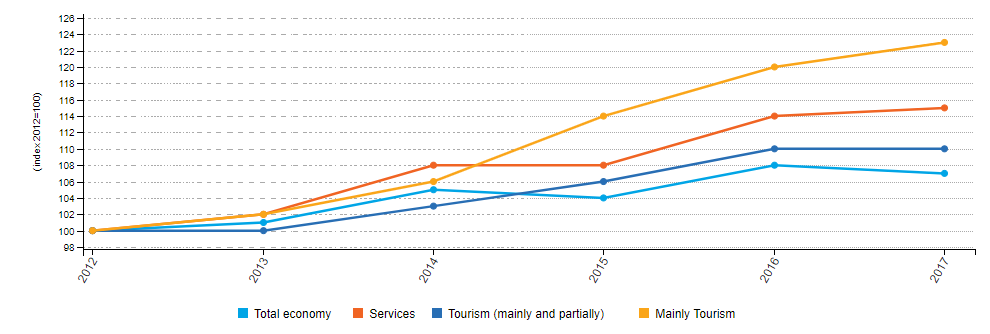
Another classification based on the previously presented paragraph is internal tourism, national tourism, and international tourism.

The statistics resulted using that classification are not 100% accurate, having as main sources: border statistics (not sufficient to determine tourism flows), data from collective accommodation (but not from private accommodation), data from international transport, household surveys and surveys at tourism attractions.

Traveling’s economic role

The traveler is the center of attention for the travel ecosystem, and the customization of the travel experience is concentrating the efforts of the stakeholders in this sector. A key factor is a better understanding of travelers’ wishes, what prompts them and the way they behave. Even so, the evolution of technology made available to the travelers has caused them to change. From oral transmission, through fairs, printing press and telephone to television and electronics, the technology of the communication market changed.

The ways of change consist also of travelers’ pro-activeness regarding contents, the relation with other travelers and the reveal that some travelers are active managers of their own business modal, not only as users or consumers, but also as producers or developers of travel business, with, of course, an economic impact.



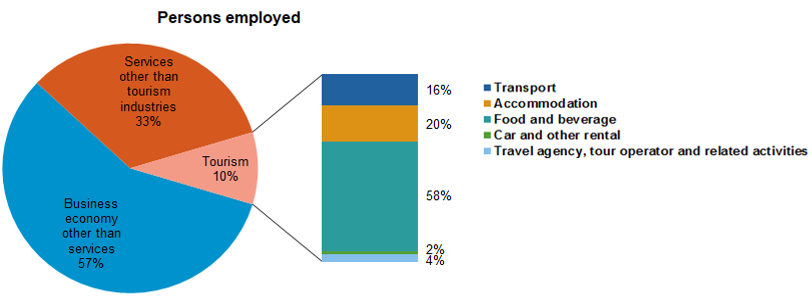
**Figure 2.1** Evolution for different sectors of the economy, 2012-2017. Source: [5]

Travelling influences a wide range of areas, like regional development, employment, education, the environment, consumer protection, health, safety, culture, new technologies, transport, finance, taxes and many more.

It all makes sense and it helps to see the big picture when taking into consideration the branches of the economy benefiting directly from the travelling, like: travel agencies, tour operators, hotels and other accommodation, restaurants and, in general, food industry, transport companies, tourist information centers, souvenir manufacturers, travel insurers, entertainment industry, maps distributors and travel equipment manufacturers.

The explanation of how travel industry keeps the economy going is that consumers tend to spend more time and, concurrently, more money when they are on a tour. One of the most important reasons is the phenomenon of the multiplier effect – how many times money spent by tourists circulates through the economy of the country.

Some countries are running their economies mostly on travelling incomes, countries like this are Switzerland and Singapore. Also, more than half (56%) of the 2.3 million enterprises in the tourism industries, were located in four of the Member States of the EU: 384 thousand in Italy, 327 thousand in France, 308 thousand in Spain and 263 thousand in Germany. [5] More and more people are travelling from one day to another, with international arrivals expected to reach 1.8 billion by the year 2030. [4]



**Figure 2.2** Number of persons employed, 2017. Source: [5]

In Europe, there are over 13 million people who are workers in the traveling industry, in 2017. Also, 13% of them were young workers, with ages between 15 and 24 years old. [5]

Travelling generates new employment avenues to the native of the country, it provides foreign exchange earnings to the destination country, it raises the living conditions of the citizens and it helps in raising the gross domestic product (GDP) of the country. Another advantages are the infrastructure development, the contribution to cultural exchange and the help on preservation.

Throughout the last ten years, the travel area had experienced a substantial travel growth. Transportation venues - airports and rest stops have specific travel function and associated meanings. In contrast, the traffic congestion that appears both in-town and out-of-town intersections is the most critical negative facet of traveling at the desired destination. Hospital and gas stations, which offer support services, have specific meanings and functions in the communities. Even if the hospital is not a place where a person goes voluntarily, being viewed as having a low demand among the tourists, it has a positive impact on the community. In a similar way, gas stations have a positive utilitarian role and are considered neutral in the social life, serving both residents and travelers. Travel industry depends to a large extent on public authorities, for example, a travel service provider can’t attract customers only through the quality of the product, but must also take into account the quality of the infrastructure, the beauty and quality of the social and natural environment, as well as the level of other services distributed on the area. [6]

Electronic technology has allowed the participants to simultaneously communicate at long distances and introduces a new vision in the field of marketing communications. Furthermore, the arrival of new generations can also bring major changes in this sector of activity, the millennials having the potential to contribute to it due to their strong digital skills, their high degree of continuous and permanent connectivity, their search for experiences and their altruistic behavior.

A dominant feature of today's travelling is the considerable level of investment made for modern technology. Travelers have high expectations for efficient services and low tolerance when it comes to the barriers of global mobility. With today’s technology and data analysis, clients must be ensured that automation does not lead to disconnections between online and in-person exchanges. The last aspect highlights how important is the privacy of the consumer and the preservation of data throughout this process.

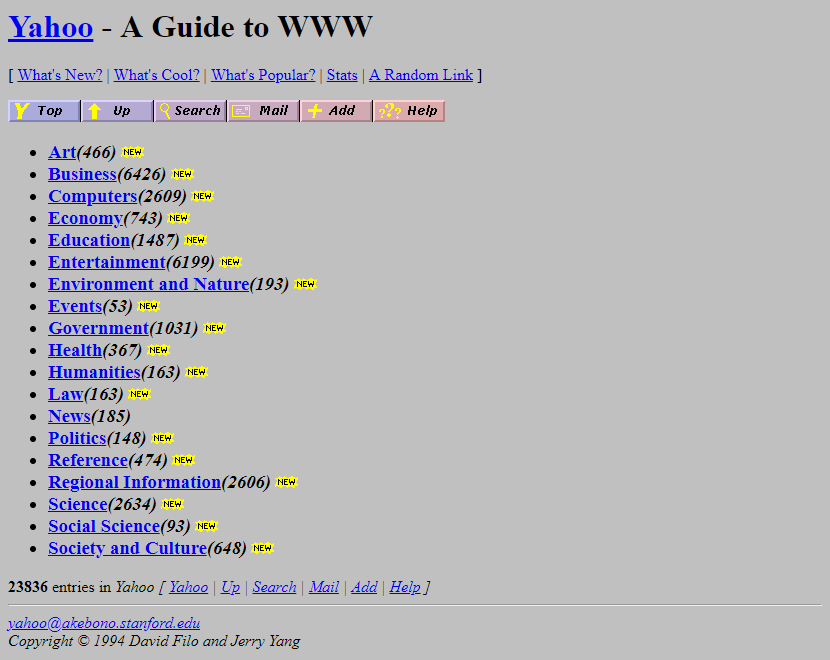
This sector of economy is sometimes the chance to relaunch after a period of crisis, other times it is the business card for a country, but, most often it is a source of environmental degradation, even if the conservation of flora and fauna is an important feature of travel industry, being the key element.

Appearance of travelling in computer software

In 1994, similar to the Yahoo style of listing of that time (see Figure 2.3), appeared the first comprehensive catalogue of hotel properties around the world which later included a component to allow bookings making – named Travelweb.com. [7]

The website was created by a small team under The Hotel Industry Switch Company’s (THISCO) John Davis. They worked with the five leading hotel chains: Hilton, Marriott, Hyatt, Starwood and Six Continents, but per total, they provided information for 90 different hotel chain brands, with over 24 000 hotel properties in 150 countries.

Later, after specializing on hotel and airline information and bookings, John Davis became a key figure in the history of online travel.



**Figure 2.3** Yahoo! website in 1994. Source: [7]

The site was formed as a showcase for the company’s activity behind the scenes, but because of its uniqueness it gained popularity over many early visitors of the web. [8] This represents the year in which the Internet became the travel industry marketing tool and the customers started surfing the net to shop for their travel industry goods.

Beginning with the year 2000, the development registered in the search engines, the transport capacities and the speed of networks have influenced the number of travelers from all over the world to use technology for planning and documenting their journeys.

Information and Communications Technologies, in general, and the Internet, in particular, started to be considered as being one of the most effective tools for boosting the travelling industry. It have been observed that the users trust the Internet to obtain advice. This is what the organizations base on when they are trying, step by step, to engage customers in online conversations, and the main reason is that they want to create a sense of trust, to improve brand image and to strengthen the commitment to the user. These days, social networks, blogs or virtual technology plays a useful role in domestic travel, communication, skills and knowledge.

Present travelling influence in computer software

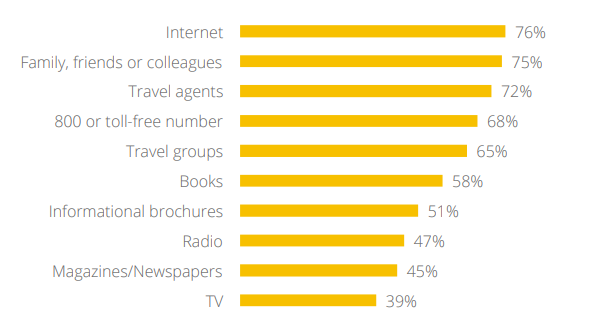
A travel website on the World Wide Web may be focused on trip fares, on travel reviews or on both of them. In 2018, approximately 587 million consumers booked travel plans online. [9]

The computers role in travel and tourism became essential. For example, when we go visit historic places and we want to find out more details about their importance, basically, their history, we do not need a guide, a person, to explain us.

Nowadays, a search on the internet should do it, or we are offered a device that is actually a computer and helps us make the tour in a virtual way. These devices also use multimedia elements like music, sound effects or floor plans to recreate a realistic representation of history.

As represented in the graphic below, the Internet occupies the first place, being the main source of information and inspiration for personal travels, the second representing people’s reviews.

Most travel websites are online travelogues or travel journals – illustrated lecture about places visited by or experiences lived of a traveler. Both types are usually created by persons who travel individually and are hosted by companies that, in most of the cases, provide this information to consumers for free.



**Figure 2.4** Importance of sources for inspiring personal travel. Source: [7]

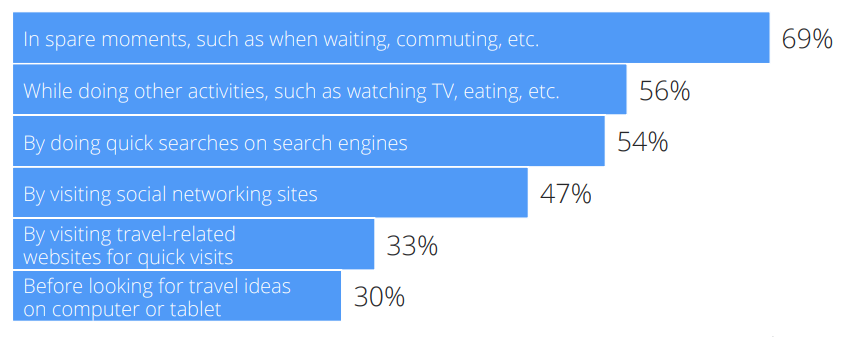
The revenue of travel hosting companies is generated through advertising or by service provided to other businesses. The styles in which they are produces and displayed are in a wide variety, often containing photography, graphics, maps and reviews that leads to acquiring unicity.

The digital facilitates deeper customer relationship and, as I stated above, higher consumer standards among travelers. The travelers have the following expectations from a website’s services: the facility in using, the informational content, the utility, the security, the rapidity in operation and the personalization.

Traveling search on different means of communication

Smartphones are often used for inspiration for journeys in “snacking moments”, before planning. This is the first step in a trip planning and it plays an important role on the user’s perception. It is mandatory for the website to have fully functionality on the smartphones, not only on the computers.

It became a tour guide, a travel agency, a map, the best restaurant locator and, perhaps, an indispensable object, being by the traveler’s side during the entire journey. In this way, the user has all the pertinent and necessary information about the trip in the palm of their hand using an application which only requires internet connection.

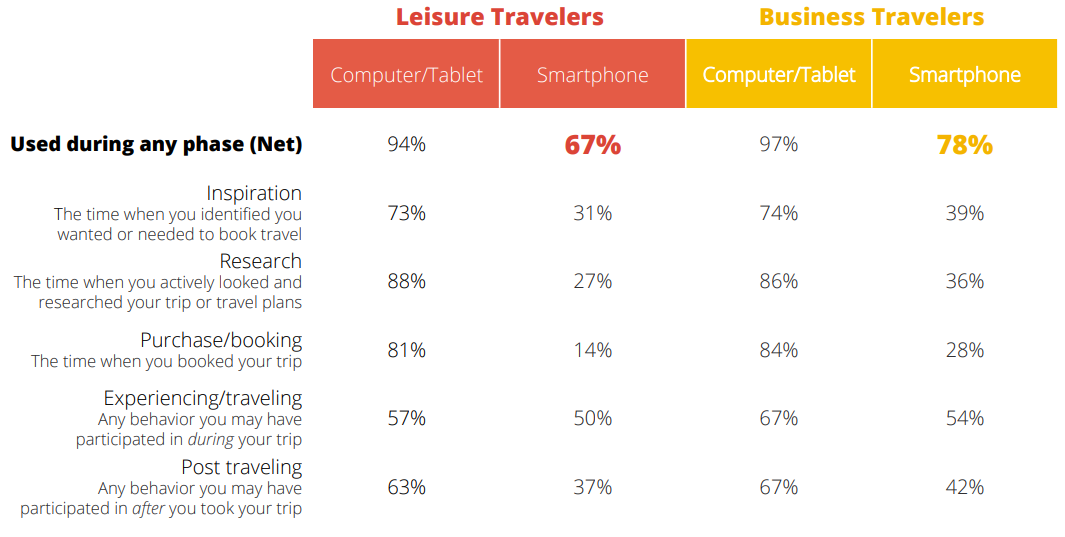


**Figure 2.5** Usage of smartphone for inspiring leisure travel. Source: [7]

It is well known that an indispensable device used on leisure travel is the smartphone.   
The wireless networks and the mobile networks allow the users to connect their communication dispositive to the global network, being widely spread across the world, in hotels, airports, pubs and many other places where people usually meet.

The 3G and 4G represent the third and, respectively, the fourth generations of mobile broadband Internet. Each generation of wireless broadband requires that the smartphone provider should make upgrades on its towers and requires that the user should upgrade his phone in order to send/receive signals through the new infrastructure.

The fast development of technology represents why there is a need to adapt corporate services and communication to all means of communication.



**Figure 2.6** Types of devices used by travelers. Source: [2]

In the graphic from above it could be clearly seen that, in both leisure and business travels, all means of communication play a huge role. If a website is not adapted for every device, the user’s experience is worsened.

Presently, the smartphone users have the possibility to choose from a variety of application that can be installed on their dispositive for speeding up communication and providing all the necessary information.

The multimedia technology started its influence over the travel field, being considered an extensive support which offers an image or a tangible experience to the customer.

Virtual tour is a new and welcoming mean of marketing, created to increase the satisfaction of experiencing in full depth and detail. It consists of a simulation of an existing location through the medium of sequential videos or images. It provides excellent alternatives to fieldwork when the time or the expenses are an issue for the user. There are several methods and techniques available in order to create a virtual tour, for example: video tours, 360o (means that you can look in any direction: to the right, to the left or even backward) or panoramic tours, still photo tours, floor plan tours. A good virtual tour should have the ability to project images from different angles and scales in order to enhance the user experience. They can also provide an experience that can be repeated, feature useful in education, to reinforce learning for students. However, they are suffering from limited navigational skills. [10]

## Trip searching online platform

The role of online platforms is to deliver benefits to both consumers and businesses by bringing together the consumer and the producer and allowing trades that would otherwise be harder to happen.

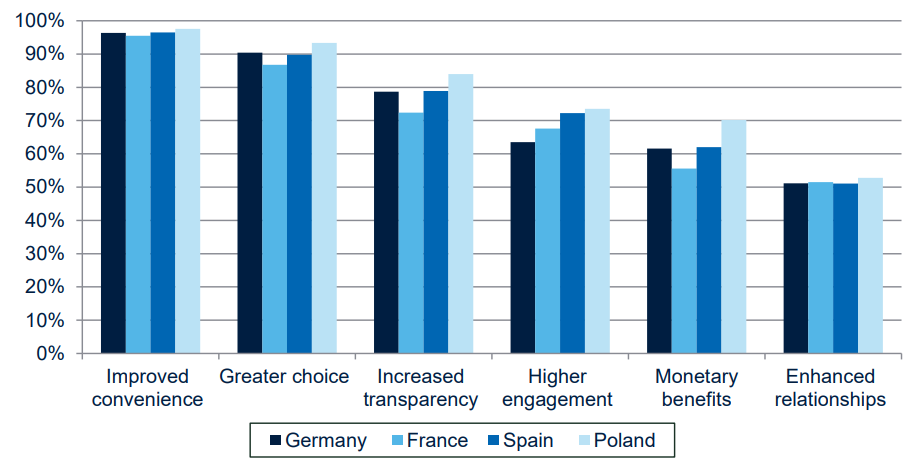
The diversity of online platforms in terms of activity, business model, sector and size is surprising, but there isn’t an apparent commonality between them. When examining the Web searching on specific domains, one user is able to increase its understanding of Web searching, to advance its knowledge of Web searcher’s information needs and to influence in a positive way the design of the Web information system. This applies the most to the travel domain. Although the extent and frequency of travel platforms, the patterns are similar across the countries.

There are 73% of American internet users that have obtained travel-related information from the Web, based on a report from the Pew Internet and American Life Project. Also, when it comes to benefits, almost all users, more accurate, 97% of them, in 4 of the most developed countries: Germany, France, Spain and Poland, think that Web searching improve convenience, gives greater choice area and increase transparency. Users estimate that information platforms have saved them 100 minutes in Poland and 50 minutes in France and Germany, in the past month.

With the dramatic growth of social media platforms that was received in the recent years, such as Instagram, about 40% of travelers said that the comments they read on social networks influenced their plans for vacation, while 50% inspired for and based their travel plans on other people’s experiences and reviews. [11]

Virtual communities become more and more influent in the travel domain, as long as the former travelers tend to create a higher degree of credibility to the user, rather than marketing messages. This kind of community helps the user to obtain information, to maintain and develop relationships and to make decisions concerning the future trip destination.

There are things that concern users, some of them being the privacy and security issues, the confusing functionality and the inappropriate content. Overall, users from all over the world are more likely to perceive benefits from than to raise concerns about the online platforms. [12]



**Figure 2.7** Consumer perceptions of benefits. Source: [11]

Just with a simple search for travel information on any major Web search engine, for example Google, Bing, Yahoo, and Yandex, the user receives billions of results. Because it was observed a high use of the Web to locate travel information, there must be an examining and understanding of how people search in order to find relevant information. When searching, a dizzying collection of booking engines, blogs, interactive maps and wikis is displayed. Compared to the last decades, a travel feels less of a lottery because we, as consumers, feel more in control.

How online platforms evolved

In the last two decades, the search engines evolved considerably. The World Wide Web’s first generation, Web 1.0 was static and unidirectional. The webpages were represented as HTML pages with a content that was rarely updated. Website’s main role was to display information for anyone anytime and to establish an online presence. It wasn’t interactive and were mainly like brochures, the visitors having as option only to visit without impact or contributions.

Planners vs in-destination bookers

As it may have been expected, there are differences between types of travelers. There are the ones who book their activities ahead of their trip that spend 81% more on transportation and 47% more on lodging than those who wait to book in destination, making them a valuable audience, especially for companies like online travel agencies.

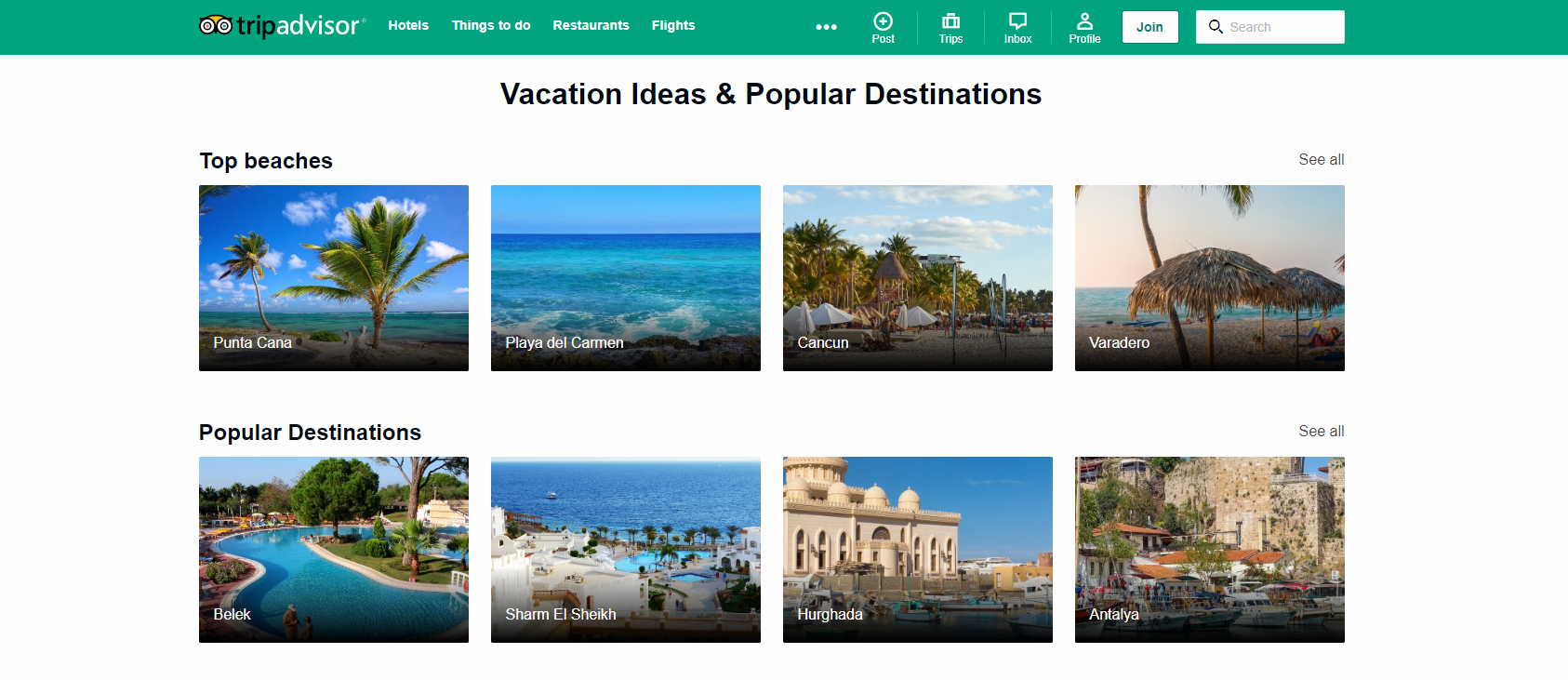
Around 50% of experiences bookings are happening once travelers arrive at the destination. Also, the majority of those in-destination searches are happening on mobile. [13] This is the main reason behind Google’s latest announce to travel brands, which consisted of a request to make websites simple, personal and mobile. These are some of the features used in order to attract, convert and retain travel customers.

After studies took place, it was observed that 14% of users are overwhelmed by the amount of travel information from the web, which explains Google’s urge for simplicity. An amount of 42% stated that the desktop computer is an easier mean of communication options for searching travel idea, than on the other interfaces. And some of the users avoid to use mobile devices for searching travel idea due to the slow connection speed. [14]

## The analysis of existent software solutions

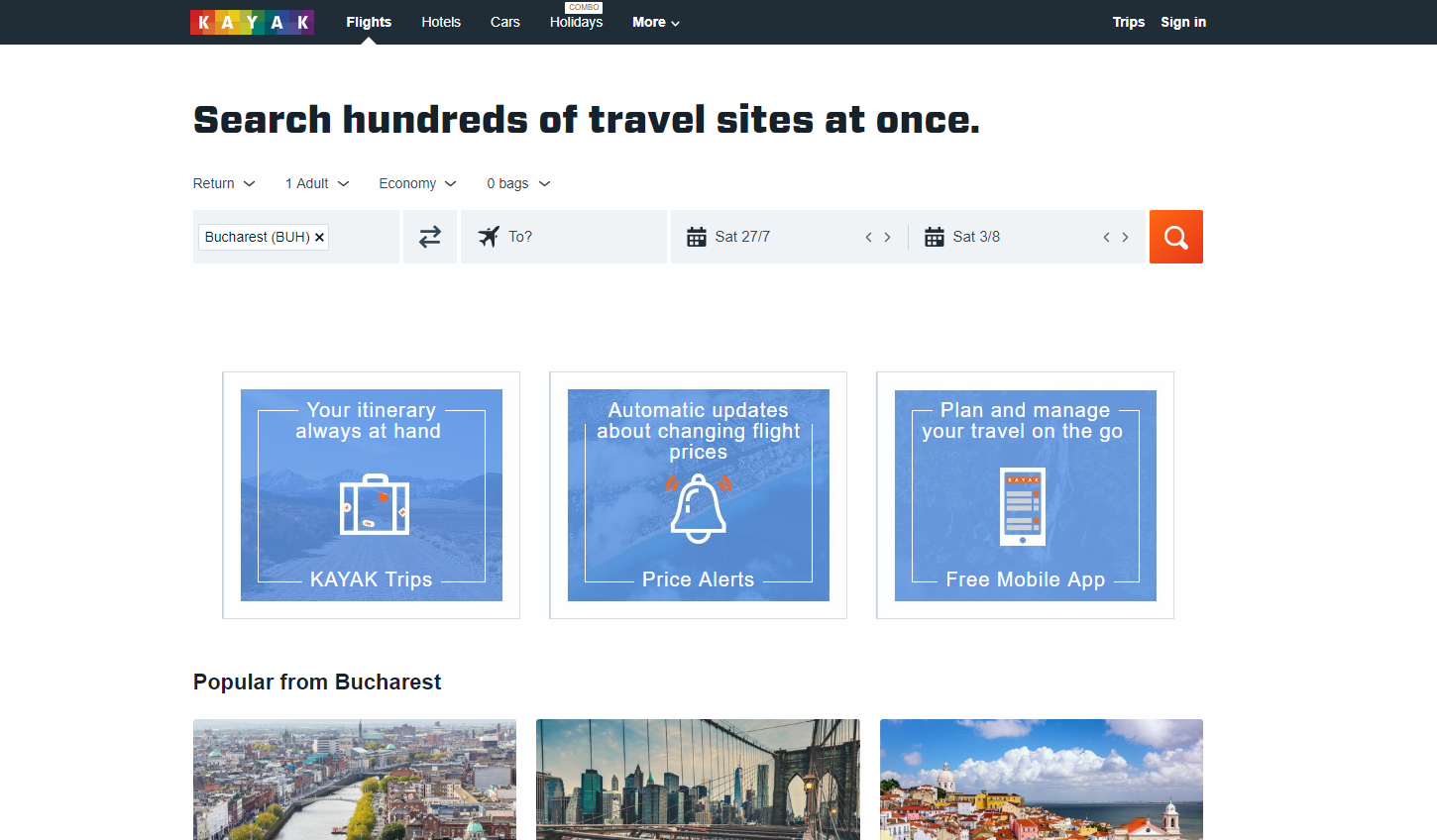
The travel industry’s development during the last decades led to a vast number of travel agencies and related websites. But, according to CleverTap, more than half of travel apps are deleted after one month.

World’s most trusted travel advices are the ones owned by TripAdvisor – the original and after 20 years, still the biggest social travel site existent on the web. The process of navigating on it was incredibly simple, consisting on the traveler’s activity to leave a review of hotels and then, the fellow travelers to find the reviews and decide whether to stay at a property or not, based on what they’ve read.



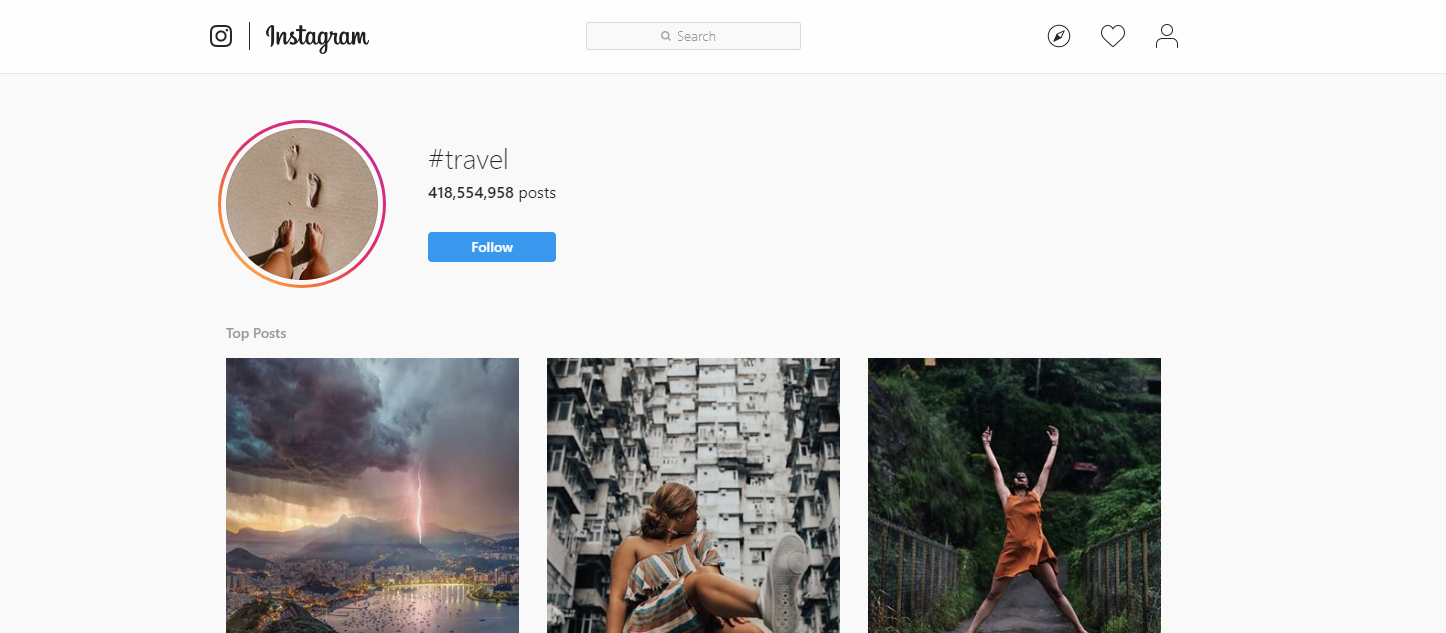
**Figure 2.8** TripAdvisor landing page. Source: https://www.tripadvisor.com

Its closest competitor is Kayak. The website is offering multiple services, like deals on flights, hotel rooms and rental cars. Its best feature is the pricing trends and the predictive algorithms on whether the prices rise or fall. Also, the travel alert services a user can benefit from after subscribing allows him to found out about travel discounts.



**Figure 2.9** Kayak landing page. Source: https://www.kayak.com

The apparition of social media created a wave of social media-fueled travelers, especially among the Instagram users. Today, influencers are everywhere on the Web and they promote different kinds of destinations. On Instagram or Facebook, pictures or videos of trips circulate rapidly and reaches a high amount of audience. Consumers also read the reviews situated in the post’s comment and are influenced by the new trends.



**Figure 2.10** Instagram travel page. Source: https://www.instagram.com

# Technologies Used

## HTML

HTML is a hypertext markup language that has become very common on the Internet. Every website on the Internet uses HTML to display information.

HTML defines the structure of the pages that you see in the browser thanks to HTML tags, the browser reads, processes them, and then displays tags to you on the screen, but in the form of HTML elements, you can even interact with some HTML elements using the mouse or keyboard.

To be precise from a formal point of view, it is correct to speak not an HTML page, but an HTML document. Your browser communicates with the web server using the HTTP protocol, sends HTTP requests and receives server responses, the body of which contains HTML.

Like the HTTP protocol, HTML was developed at CERN by Tim Berners-Lee in 1991 and was originally used by scientists to exchange scientific documents. HTML clearly defined the structure of the document and made it possible to highlight certain features of the text of the document, thanks to the simple syntax of the HTML language, it was widely spread not only in the scientific community, but also went to the masses.

At the moment, most often you can find sites using HTML 4.01, the latest HTML version is HTML 5, which all modern browsers correctly display. For each HTML document, you must specify a version, for this there is a special tag DOCTYPE, which says about the version of HTML.

The HTML language allows you to structure information in various ways: create tables, different types of lists, paragraphs, add images to a document, break a document into parts using headings and sections, create forms for interacting with users, link different HTML documents to lists.

Web designers began to look for ways to beautifully present information. Some HTML tags were not used for their intended purpose, for example, <table>. The method of layout of web documents using this tag has become so popular that it even got a separate name - tabular layout. Previously, only using this method could accurately position the elements on the page.

With tabular layout, the web page design was created directly inside the HTML document. Other tags for styling and formatting were also used there. What problems did it cause? First, the HTML code became incredibly large in length, which negatively influenced both the weight of the document and its indexing by search robots. Secondly, in order to change, for example, the color of the h1 headers on each page of the site, we had to manually work out each of them. All this took a lot of time and effort. Also to format HTML documents and to make HTML tags unique, HTML attributes were invented, but at the moment no one uses HTML attributes to design documents, as there are cascading style sheets or CSS. There are exceptions, for example, some email clients cannot work with CSS and HTML attributes are used to design emails.

Today, thanks to the existence of CSS, it is possible to separate the design of the page from its content, as well as speed up the work process and significantly reduce the size of the HTML document. Those who have been creating websites for a long time have to get rid of old habits and learn to perceive HTML only as a markup language designed for structuring and organizing data. It is easier for beginners to learn new principles of creating web pages due to the lack of views on HTML as a tool for making pages look attractive. CSS is now responsible for this.

To structure HTML pages we can use the following tags:

* h1-h6 (heading) – these tags are intended to denote headings. Using these tags is very convenient to separate the text. For better understanding, imagine a book with chapters and subchapters. The title of the book is h1, the subchapters are h2, parts of the subchapters are h3, etc. It is better to place heading tags sequentially.
* p (paragraph) – This tag is used to designate paragraphs of text.
* ol, ul (ordered list, unordered list)– these tags are a handy tool for labeling lists (navigation links, items in the text, sequential list, etc.).
* dl (definition list) – this tag in conjunction with the <dt>, <dd> tags is used when creating a definition list, where <dt> is the definition term, <dd> is the definition description.
* div (division) – this is a block element that can be used to select a fragment of a document, as well as to logically combine several elements. With CSS, you can give the <div> block the necessary look and positioning, but the <div> itself does not change the appearance of the document.
* span (span) – the role of this tag is similar to <div>. But <div> is a block element, and <span> is a lowercase element. For example, if you need to change the style of a single word inside the <p> tag, you wrap this word into the <span> tag, add an id or class attribute with the name of the selector, and then assign the desired style to it in CSS.
* HTML5 semantic tags – to better structure your HTML document, use new tags to help you better describe the content. To distinguish the site header, navigation menu and footer from other content is hard, if all of them are marked with <div> tags. HTML5 tags such as <header>, <nav>, <footer> and others can help us to do this. They do not affect the appearance, but they help navigate browsers and search robots that have come to the site.

HTML5 is not a follower of the hypertext markup language, but a new open platform designed to create web applications that use audio, video, graphics, animation, and more.

The developer has a lot of new tools to improve the user interface: from more meaningful tags and improved cross-site and inter-window communications to animation and improved multimedia support.

The HTML5 standard has appeared quite recently and is still under development. Its main difference from the markup languages ​​of past generations of HTML 4.01 and XHTML 1.1 is the presence of specifications for working with multimedia applications, while maintaining the ease of reading code for humans and clarity of performance for computers and devices (the so-called user agents).

The main features that HTML5 brought into the development of web applications and interfaces include:

* **New improved markup of documents** – thanks to new markup elements, creating HTML5-based documents becomes faster and more quality, which leads to lower costs for creating layouts for web pages and applications. The semantics of the page increases - search engines automatically recognize where on the page is navigation, and where content is.
* **Drawing on the page** – HTML5 defines the canvas tag as a raster graphics canvas that can be used to display diagrams, computer games, or display other images on the fly. The canvas itself is a rectangle on the page in which you can draw what you want with JavaScript. HTML5 defines a set of functions called the “Canvas API” for drawing shapes, outlines, creating gradients, and transforming.
* **New form elements** – thanks to HTML5, developers have new features that were previously available only when using complex JavaScript libraries. Validation of the entered data is now happening "on the fly" right in the browser, which simplifies filling out the form.
* **Audio and video streaming support** – now you can easily embed media files directly into a web page without using “heavy” technologies like Adobe Flash or Microsoft Silverlight. Moreover, the HTML5 specifications allow direct control over the playback of these files, which can be useful, for example, when synchronizing video and subtitles to it.
* **Cross Platform Support** – HTML5 specifications are suitable for various user agents, which can be not only computer browsers, but also various portable devices. HTML5 creates various applications for smartphones, mobile phones, and home gaming consoles of the current generation.
* **Error processing** – documents may not always contain the correct syntax, but HTML5-compatible browsers, like their predecessors, use markup error analysis algorithms in documents to build the correct object model (DOM). A clear definition of requirements for user agents is made in order to achieve compatibility between browsers from different manufacturers. As well as the requirements for the syntax of the markup of documents in order to correctly display them in different browsers.
* **Geolocation** – some web applications with the user's permission may transmit data of his location. There are several ways to determine your position: by IP address, connection to a wireless network, cellular operator or via GPS equipment.

This is not all the features that HTML5 offers to developers and users. However, with all its merits, like all new technologies, HTML5 faces some problems. Most outdated browsers do not work with HTML5, and all specific tags are simply ignored, thus the display of pages in different versions of the browser may differ significantly. Since HTML5 is still under development, modern browsers can support it to varying degrees too.

## CSS

CSS (Cascading Style Sheets) is a formal language used to describe the appearance of a document created using a markup language (HTML, XHTML, and XML).

The purpose of CSS is to separate what defines the appearance of the page from its content. If a document is created only using HTML, then it defines not only each element, but also the way it is displayed (color, font, block position, etc.).

If cascading style sheets are connected, HTML describes only the sequence of objects. And for all their properties is responsible CSS.

In HTML, it is enough to prescribe a class, without listing all the styles each time.

Such technology:

* provides a relatively simple and fast development, because once created design can be applied to many pages
* Increases flexibility and ease of editing – you can just edit the CSS for a certain element and the design for this element changes everywhere
* makes the code simpler by reducing the repeatability of the elements so it is easier to read for programmers and search bots
* speeds up loading time, because CSS can be cached when first opened, and only structure and data are read in subsequent ones
* increases the number of visual solutions for presenting content
* provides the ability to easily apply different styles to one document (for example, to create an adapted version for mobile devices or special styles for the visually impaired)

So the cascading tables serve not only to translate the design, but also drastically change the approach to site building, simplifying the work of developers and providing implementation flexibility.

The need to develop CSS was recognized by the W3C consortium in the 1990s.

In 1996, the CSS1 standard was adopted, allowing you to change the font parameters, color, text attributes, alignments, and indents.

In 1998, CSS2 was released, which added the possibility of using block layout, sound tables, generated content, pointers, and paged media.

CSS3 version significantly increased the possibilities of styles: it became possible to create animated elements without using JavaScript, there was support for anti-aliasing, shadows, gradients, etc. The specification was divided into modules, each of which began to develop separately. Since 2011, the development of CSS4 modules has begun.

CSS can be described in simple terms as a set of rules that describe how an element should look. A rule consists of a selector and a block of declarations. In the Figure 3.1 you can see the example of the CSS rule.

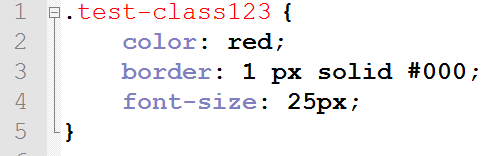


**Figure 3.1** Example of the CSS rule.

The selector points on the element which style properties we want to describe in CSS for later applying them to it. The selector can be any tag that has to be formatted (size, color, etc.).

If you need to set different styles for a tag or apply one for different elements, the classes are used. In Figure 3.2 you can see the syntax of the style declaration for the classes. The class name is given using Latin letters, it may contain an underscore or a hyphen.

If you do not specify a tag, and start recording with ".class", then you can use the rule for any tag. If you list several classes for one tag, all described styles will be applied to it.



**Figure 3.2** Example of declaring styles for the elements with class "test-class123".

The ids specify the unique name of the element to change the style or select using a script. The name of the identifier consists of letters of the Latin alphabet, it is permissible to use a hyphen and underscore.

To apply an identifier to a specific element, you have to set a unique id name and then in CSS put # sign and the id name of the element without spaces between them.

The declaration block consists of “property: value” pairs (always written with a colon) placed in curly braces. Entries end with a semicolon. CSS is insensitive to tabs, spaces, case. The choice of method of recording (indented column or just a line) is left to the discretion of the developer. If for one selector different values are written for one property, then priority is given to the lower entry.

There are several ways to link the Cascading Style Sheets to your HTML page for applying the declared styles:

* inside the tag using the style attribute. There is no need to specify a selector
* add a <style> tag with the type = "text / css" attribute
* connect an external style sheet: <link rel = "stylesheet" href = "path to style.css" type = "text / css" />

The third method is the most popular and recommended, because it allows you to fully enjoy the benefits of the separation of form and content provided by CSS.

CSS3 has revolutionized the web development world, because it brought a lot of new functionalities. This technology continues to evolve and introduce new features. CSS3 allows developers to re-style the background and borders of HTML elements in new ways.

border-radius is a useful property added in CSS3 that allows you to round the corners of the blocks. By default, they are rectangular, and this is not always necessary in design. Previously, such rounding was realized with the help of additional background images. Well, now we can rejoice that everything is so much simplified.

Now it is enough to write “border-radius: 35px;”. The larger the value is, the greater the rounding will be applied. It is not necessary to set it in pixels - it is also possible in percent or em-units. Also depends on the size of the block. If the element is square, then it can be turned into a circle. For this you need to set a rounding of 50% or half the pixels of its width. You can round each individual corner. To do this, use the rules of border-top-left-radius, border-bottom-right-radius, etc. Another option to define the values ​​for each corner separately is to write four values ​​in border-radius. The values ​​are alternately set for the upper left, upper right, lower left, and lower right corners.

Of course, since the property is part of the CSS3 specification, it’s wise to use it with vendor prefixes for cross-browser compatibility. Perhaps you yourself know them. These are –moz, -o, -ms and –webkit for support in Mozilla, Opera, IE and Chrome respectively, or rather, in older versions of these web browsers, because new browsers already understand the property without prefixes. These same prefixes can be used with all other properties from the CSS3 version.

**Linear and radial gradients** is one more part of the big CSS3 innovations about which you could write a separate book, because gradients have a lot of different parameters. But they are not created as a new property, they only expand the capabilities of the background property. To use them, it is enough to write in the background or background-image property:   
linear-gradient() or radial-gradient().

For example “background: linear-gradient (to right, aqua, yellow);” will set the smooth transition from the aqua color to the yellow color from left to right.

The “opacity” property allows you to set the transparency of the desired element. Values from 0 to 1 are accepted, where 1 is full opacity. With a value of 0 the element is not visible on the page, but it remains in its place. It must be said that the opacity acts on the element as a whole, and not just on the background. If there were text in our block, it would also become unreadable, and this should not be allowed. Therefore, text boxes usually use the “rgba” color mode as an alternative. Opacity is convenient to use for hiding / appearing elements.

**Shadows** were also added in CSS3. To add them, 3 properties are provided, and each has its own characteristics. All shadows have common settings, such as horizontal offset, vertical, blur, and color. Stretching is set only for box-shadow.

* box-shadow - usually used for blocks.
* text-shadow - shadow for text. It has the same parameters as the previous one, but it does not specify a stretch. It is worth noting that blur is also an optional parameter for both properties, it can be omitted.
* filter: drop-shadow() - perhaps, this property is the least known, and it is worst supported by browsers. In fact, if you try to apply it without prefixes, then, most likely, it will not work.

**Transformations** are another separate group of new css3 properties. Rather, the parameters and not the property itself give for the developers many possibilities. The transform works in much the same way as the filter, but has more complete support. Main transformation features:

* Move element horizontally / vertically, using transform: translate / translateY (for vertical displacement).
* Rotate elements to the desired number of degrees - transform: rotate (45deg). Values ​​from 0 to 360 and negative are accepted.
* Tilt items using skew. Also set in degrees.
* The ability to control block sizes using scale. You can set only scaleX or scaleY to increase the element only in width or height, respectively. Usually used to implement with smooth transitions.

**Smooth transitions** are also very important for animating objects in CSS3. To implement them, it is enough to use one property — transition, which can indicate the time during which the styles change. Usually set 0.5-2 seconds, although it is possible in milliseconds.In fact, the transition supports many more parameters that can be defined, but it can take too much text to explain each one.

## JavaScript

Like any programming language, the main task of JavaScript is to create a sequence of actions that will lead to a certain result. These may be “if-then” conditions, loops that create a specific sequence of actions, mathematical calculations, etc. The most important thing is that all these operations can be performed on web pages, in a browser window. Moreover, JavaScript can work without connecting to the Internet.

JavaScript is an interpreted programming language designed to interact with web pages. JavaScript is an implementation of ECMAScript and consists of three parts:

* The kernel (ECMAScript) is the core JavaScript functionality.
* Document Object Model (DOM) - for working with the contents of web pages.
* Browser Object Model (BOM) - to interact with the browser.

In browsers, by default, special software is built in, called the JavaScript interpreter, this is done so that the browser can execute code written in JavaScript. Usually, JavaScript is called client language, thus emphasizing that the script is executed on the client computer in the browser, and not on the web server.

Here are some examples that demonstrate the capabilities that can be obtained using JavaScript:

* **Mathematical operations:** On web pages very often there is a need to make certain calculations. For example, there are two text fields and you need to output the sum of two numbers in the third text field, which are entered in the first two. Using JavaScript, you can create a calculator and place it on a web page. Another situation, you take some text string on a web page and you want to increase its size by 1.5 times. This can also be done with the help of mathematical calculations, by multiplying the current size by 1.5. There are a lot of tasks that require calculations in practice. JavaScript allows you to do all this.
* **Data processing in HTML-forms without connecting to the Internet and without using server side programming languages:** JavaScript allows you to verify that all required fields are filled and the data that they contain corresponds to the desired format (for example, if digits are required, then there must be digits and there should be no letters). Before sending data to the server, they are pre-tested by JavaScript. This reduces the load on the server.
* **Interactions with a user and events:** Different effects may appear on a web page, depending on what actions the user performs. For example when you click on the button you want to hide or show some elements on the page, or the appearance of a pop-up window when the mouse cursor went outside the browser window, or background dimming and fade-out effects. All of these actions and effects can be implemented using JavaScript.
* **Interactions with HTML elements on the page and managing of their content and styles:** When a certain event occurs (for example, a mouse click or any other), you can change the appearance (CSS styles) of elements on the page. You can also add some HTML tags or attributes to them, also when a certain event occurs.
* **Create and read cookies, retrieve visitor’s computer data:** A cookie can be used for session tracking, authenticating, remember specific information about the user like his name, password, last visited date etc. This is useful for websites and users both, if not used to steal privacy.

The release of the ECMAScript 2015 standard, informally called ES6, made significant adjustments to familiar things. The appeared functional is a superset of the language, aimed at solving actual problems. Creating arrow functions, as well as const, let and classes became a surprise for many. This standard made possible applying some new methodologies in JavaScript programming:

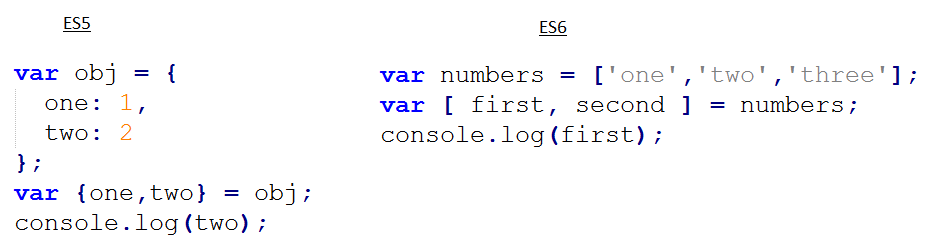
* **Destructive assignment or restructuration:**

In Figure 3.3 you can see the same code written in two different standards. In ES5, in order to transfer object properties to a variable, you need to write var one = obj.one. Such an operation must be performed with each property of the object.

In the case of the new standard, as you can see, everything is much simpler, and the code is much shorter.

In the example below, I created two variables at the same time, and they assigned themselves two properties of the object in pairs.

Moreover, this operation can be done in one line, which is very convenient when you have a large object with an array of properties. Destructive assignment works with arrays too.



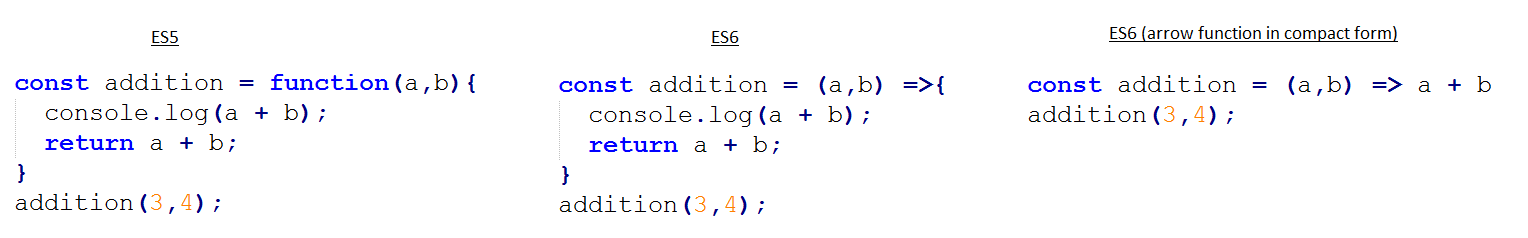
**Figure 3.3** Destructive assignment in ES5 and ES6

The same approach can be used when working with functions. In addition, so-called “default parameters” are supported. It is possible to specify what value the function argument will receive if we did not pass any value to it during the call.

It should be noted that when transmitting any value other than undefined, including an empty string, zero or null, the parameter is considered to be passed, and the default value is not used.

* **Arrow functions:**

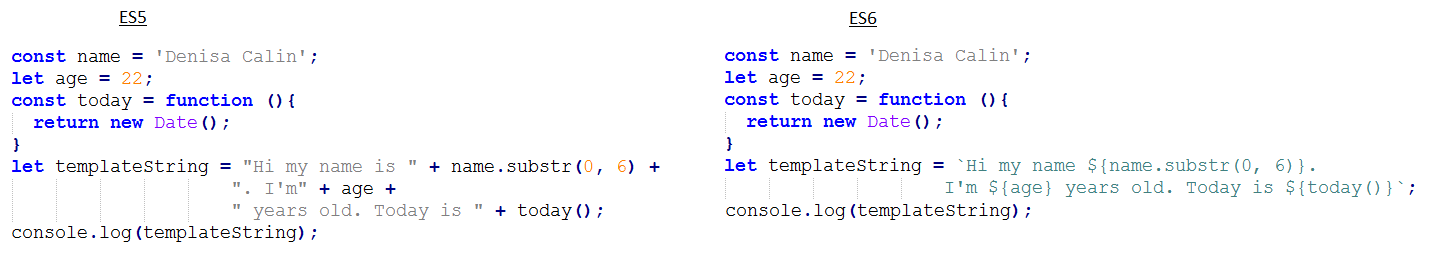
In fact, they are syntactic innovations and allow you to record the function itself much shorter. The syntax using () => at first seems a bit unusual, but the more you use it, the more you will like it, as the code with it gets shorter and cleaner, as it can be seen in Figure 3.4.



**Figure 3.4** Three different methods of writing the same function

* **Template strings:**

Web developers often deal with strings when they need to join strings and variables, and as a result get one big string. Often this code is too cumbersome, with a considerable number of spaces and plus signs for concatenation. Now everything is much better, as seen in Figure 3.5. Template strings save us from all this cumbersome code. A template string is created using inverse brackets `, and in order to put a variable into a string, you must put a $ sign in front of it and wrap it with curly braces. You can also wrap lines without any hyphens. In addition, you can call the function right inside the curly braces.



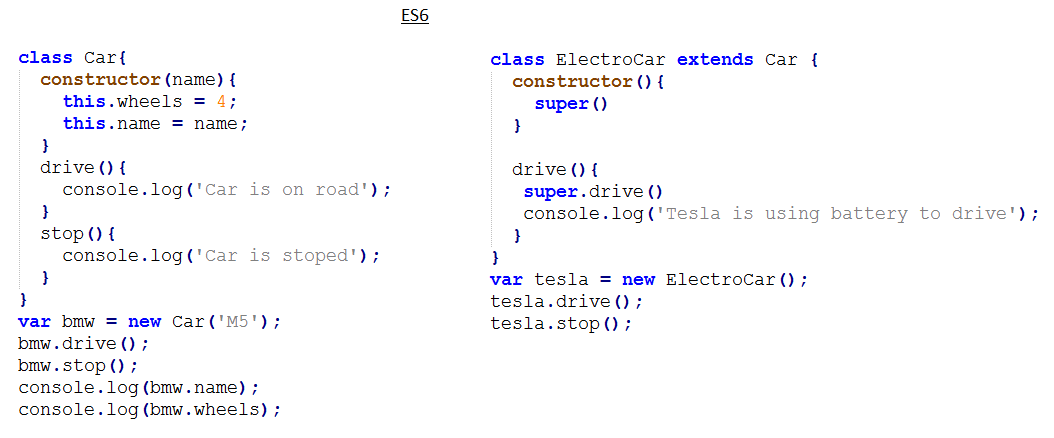
**Figure 3.5** Joining strings with variables in ES5 and ES6

* **Classes in ES6:**

Finally, the classes, which simplify the understanding of not clear for all prototypical inheritance, appeared in JavaScript. In fact, the internal implementation has not changed, but the visually new approach looks much more obvious.

You declare a class using clearer syntax, shown in Figure 3.6. Inside the class, you can set the constructor and other methods. If you need to inherit a class from the parent or base class, you can use the “extend” keyword.

Previously, if you had to call the method of the parent from the heir, you had to resort to all sorts of tricks that looked strange and were not always clear. Now everything is much easier. Finally, you can use the parent method call via super().



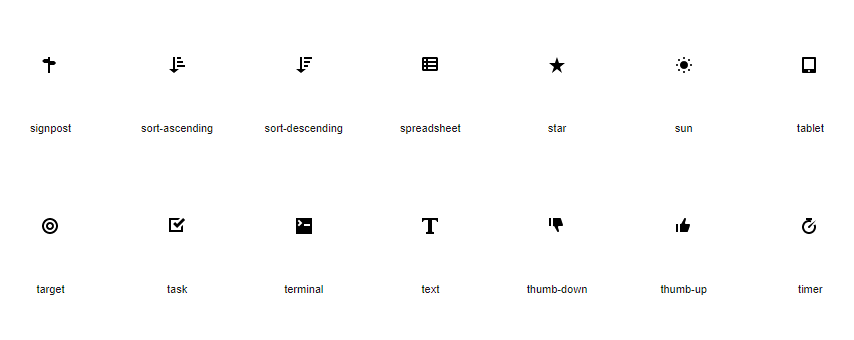
**Figure 3.6** An example of creating a class and its child class in ES6

## Bootstrap

The Bootstrap front-end framework is a set of ready-made tools for developing Internet solutions. In other words, while creating a website, there is no need to think about typography, responsive design, icons, indents, and other trifles that make up a truly beautiful design. You just need to download the package of tools Bootstrap and start using it.

After installing the bootstrap we get the following features:

* Ready CSS model with responsive tables and grids. You don’t have enough practice to create modern websites - it’s okay, the bootstrap has taken care of almost everything and all you have to do is read the documentation and start creating.
* Beautiful typography and icons for your site, shown in Figure 3.7. All indents between words, lines and paragraphs are already thought out. Your site will look equally good on both big screens and smartphones. Icons will help revive the interface and make it much more attractive and interesting for the user.



**Figure 3.7** Examples of the Bootstrap icons

Interactive solutions using JavaScript. Tooltips, interactive windows, drop-down lists, slider - in general, all you need to make the site alive and interesting to the user. All these solutions already have ready code, you only need to copy it to the site.

Not the last question that a web developer should ask himself is support of various browsers. The site must be correctly displayed in visitors’ browsers. And if most of the users use Internet Explorer 8, then you will have to develop a solution that will not scare customers. Bootstrap has already bothered about this and supports Internet Explorer 8th version. With the rest of the browsers, there is usually no such problem, because they automatically try to be updated as soon as possible.

To connect the framework files to the HTML file, you need to download the necessary framework from getbootstrap.com and copy its content into the project. After that, the developer must connect the files that he plans to use. The most popular files in bootstrap layout are:

* bootstrap.css / bootstrap.min.css. These files are an uncompressed and compressed version of the CSS code. For a project that is already running, bootstrap.min.css is usually connected. By using a compressed file, the download speed is improved. If the developer needs to view the code in the file, then he should connect bootstrap.css.
* bootstrap.js / bootstrap.min.js. Such versions of the file, but with scripts.
* The glyph icons files in the fonts folder. This folder contains more than 200 font icons.

The listed files are a typical set of frameworks. However, the developer has much more features. He can change frameworks according to his requirements.

The main element of Bootstrap is an adaptive grid. This element determines the value of this framework. The grid allows you to quickly create responsive templates. Most importantly, developers do not need knowledge and understanding of media queries. The layout of the site Bootstrap takes responsibility for the implementation of adaptability. To understand how to work with an adaptive grid, you can imagine it in the form of an html-table with rows and cells. There are two grid options:

* completely flexible
* with a defined maximum width

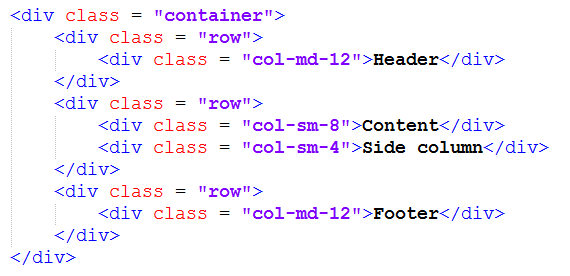
In the second case, the container class is used, which allows you to assign the required width in pixels to the general block. This block will not increase more than the specified value.  
The bootstrap grid also uses container-fluid. The presence of this class indicates that the grid is flexible and does not have any size restrictions. Consequently, users will be able to see the site in full width, regardless of the size of the monitor.

According to experienced developers, the most important issue related to Bootstrap is the nuances of the 12-column system. Knowledge of these nuances opens up the way for developers to responsive adaptive layout.

Layout on Bootstrap provides the using of 4 classes designed to change the parameters of the blocks:

* lg - for large displays with a width of more than 1199 pixels (PC);
* md - for displays of average size up to 1199 pixels wide (laptops, netbooks);
* sm - for small displays up to 991 pixels wide (tablet devices);
* xs - for the smallest displays up to 768 pixels wide (smartphones).

In this system, in order to set the width, it is customary to indicate numbers from 1 to 12.



**Figure 3.8** Example of Bootstrap grid

Any experienced web programmer will understand this markup from Figure 3.8 without any problems.

First of all, it is necessary to create a header that, in fact, may not be in the bootstrap grid. This is due to the fact that the header, as a rule, occupies 100% of the site width. In this example, I still keep the header in the “container”.

Studying the above code, you could have a question why the class col-md-12 is necessary. This class is required to send the following message to the browser: on devices belonging to the md class (laptops and netbooks), the grid width should be 12 out of 12 columns. In other words, 100% of the row width should be used here. On devices belonging to the class lg width will also be 100%. This is because the values for large displays are determined by the principle of inheritance. For smaller displays, this rule does not apply.

Therefore, if the developer specified col-xs-6, the width of the columns on any devices will be equal to 50%. If the developer specified col-lg-6, then this width will be relevant only for large screens. About this feature should know any web developer who uses an adaptive layout Bootstrap 4. The code that uses classes col-sm-8 and col-sm-4 sends the following message to the browser: the width of the content block will be equal to 66% on all screens except the extra-small ones. But on the smallest displays, the default is 100% width.

This kind of code is not just an example. In practice, this particular markup is used very often. As for the side column, its width will be 33% on small, medium and large displays. Consequently, on an extra-small screen, the default width will be 100%.

Among other features bootstrap can be noted the presence of the so-called **Responsive-utilities**. Thanks to these utilities, the developer has the ability to hide or, conversely, open blocks at the desired width. Thus, the developer has the opportunity to hide the side blocks on the displays of smartphones, add new layout elements on the screens of personal computers, etc.

In order to use the capabilities of the Responsive utilities, add classes to the selected block. To hide a block, use the class hidden. In Figure 3.9 you can see the example of its usage. In this example, the hidden-xs class is required to hide the footer on small displays. On devices with a large screen size, the footer will always be visible.



**Figure 3.9** Usage of the "hidden" class in Bootstrap

Reverse example: if a small block needs to be shown on a small display, the web developer should specify the class visible-xs-block. This code means that users will see the block exclusively on the displays of smartphones.

Classes of Responsive-utilities are written as follows:

* according to the programmer’s task he can use visible or hidden class if he has to show or, respectively, to hide the element
* to specify the display parameters, abbreviations of classes are used xs, sm, md and lg
* when specifying visible, you need to add some value: inline, inline-block or block

## Firebase

Firebase is a software development platform, representing a complete Backend-as-a-Service solution. The platform is both for mobile and web-based applications and includes services for building, testing and managing them. It has many different features that can plug right into client applications, like: Firestore, Cloud Storage, Cloud Functions, Hosting and Authentication.

All together, these different Firebase services provide clients with a whole complete backend infrastructure for their websites, so they do not need to go and create one themselves from scratch. By allowing a rapid application development, the client gets rid of the common development challenges and focuses on features and user experience.

The client controls the backend services through his own Firebase console for each different project. From the frontend, the client connects the backend via the Firebase SDK and that allows him to communicate with all the Firebase services that he needs. [15]

In the beginning, the client needs to have an initialized Firebase project, like in the example below, after adding the specific Firebase products to be used.



**Figure 3.10** Example of an initialized Firebase project

* + 1. **Cloud Authentication**

Firebase authentication allows clients to implement authentication flow into their frontend applications or websites. The users can sign up, login, access data and have profiles.

The authentication flow works in the following way:

First, the developer has a form on the frontend, like a login form, or a signup form used for capturing users’ credentials. Then, the credentials are sent securely to the server via a login method or a signup method provided by the firebase SDK.

On the server firebase validates these credentials and then sends back an authentication token to the browser. After that, the developer can access data in the frontend from this token such as the username or the email of the user that just logged in or signed up.

When requests are made to the firebase server after login, for example to the Firestore or to the Cloud Functions, then this token is sent along for the ride, to the services that has access to that information.

When a request is made to change Firestore data, Firestore will be able to look at the auth token of that request and then to protect data based on that user token. For example, the developer wants to protect sensitive database data from any user who is not an admin, and he can tell that from the token that comes along for the ride.

Also, once a user is logged in to developer’s application, the firebase persists the user’s state, meaning that refreshing the page will not mean that he logged out, he will still be logged in.

* + 1. **Cloud Firestore**

Cloud Firestore is a real time, NoSQL cloud database that is used to store and sync data for web apps, completely managed, serviced and scaled automatically by google firebase.

Firebase itself is a NoSQL database, meaning that it doesn’t use tables and columns and rows and is schema-less, which means there aren’t any database-level restrictions around what kind of data is put at any point in time in the database and it is easily iterable. It follows a pattern similar to MongoDB, which is the most widely adopted no sequel database in the world.

When a Firestore database is created, data is split into collections, and inside each of those collections, documents are stored, or, in other words, fields. To each document, it will be given a unique identifier and each document would look something like a JavaScript object or an object in any other programming language, where it has different key-value pairs. Those documents, similar to JS objects, support many different data types, from simple strings and numbers, to complex, nested objects. Every document has a unique id which can be created by the user or be automatically generated.

Subcollections could be created within those documents, in order to build hierarchical data structures that are scalable depending on the growth of the database.

Another aspect about using Cloud Firestore is that all database queries could be made from the front-end, not needing server-side code.   
Also, the authentication system is directly tied to the database, and be managed by writing expressive statement that define the backend security logic for the app. For example, the client could secure a certain set of documents that only a logged in user should see.

In code, the first thing that needs to be done is a reference to Firestore, then a reference to the collection and then a reference to the specific document ID. That reference is used to retrieve the document, listen to it in real time or update and delete it.

To retrieve it, “get” should be called, which is an asynchronous operation that return a promise that resolves with an actual document data saved in Firestore. If instead of “get”, “onSnapshot” function is called, instead of a promise, is returned a real-time stream that can be listened to with a callback function and, every time the document changes, it is going to emit a new document to that function and all of the users that have view rights on the document, are going to be notified at the exact moment of the change.

To update it, the client could use a form with multiple inputs, and control every time an input is changed. In order to do that, a reference to the document is made, but this time the client calls “update” and passes it the information that the user had typed into the form. When data is updated from the app, Firestore does optimistic updates or latency compensation. If a real-time listener is used on the data, it will update the view immediately, so the user doesn’t have to wait for a server to respond, which might have taken a second or two, and potentially hurt the user experience.

To delete it, the “delete” method is used. This method does not delete the documents within its subcollections, which could still be access by reference. A complete deletion of the document and all the documents within its subcollections is done manually, by retrieving every individual document or by putting documents in smaller batches, in case of larger collections, to avoid out-of-memory errors.

* + 1. **Cloud Storage**

Firebase Storage lets you store and retrieve user-generated content, for example uploaded images, audio and video, all without the need of a server. In order to use that product, the client needs “storage.ref(‘path’)”. The function “storage” gets the storage API and by using “ref” the client passes the arbitrary nesting that he wants to use, together, they will create the folder structure in the storage bucket, which is also tied to the firebase project.

After that, the reference “put(file)” is called for uploading that file to Firebase Storage and it returns an upload task. Using this task, the client can subscribe to any state changes happen. These state changes are represented by functions: upload progress, upload or security error and the complete function, when the file has finished uploading.

# The solution’s architecture

## Application’s description

For my graduation thesis, I have developed a web application which gives the users the possibility to select and to search ideas for their future trips based on the desired mood and location.

The results of my research, exposed in the previous chapters, indicated the fact that the websites from the travel area are the main source of inspiration for travelers worldwide.   
With an ample range of options, users that search for a journey idea are confused by the quantity of information found on different websites. The search flow would be similar to the following one: the user searches for idea, when he founds something that looks interesting for him, he selects the idea, and then, he wants to find out about possible activities at the selected destination. After that he starts looking for formal travelers’ reviews and experiences, and analyzes them. If he is satisfied by his research’s results and he’s still interested about it, he returns and adds the trip idea to his browser’s bookmarks or hopes to remember it.

The objective of my application is to help the traveler to identify more efficiently a trip idea that fits its personality and its wishes. Based on my research, the highest number of searches was related with the following two fields: location and specific needs or wants, resulting that this is the reason behind the choice I made - to filter the travel destinations based on location and traveler’s moods.

Taking into consideration that the user experience should be personal, the interface displays to the user the following options: to register if he does not have an account, and after that he can log in with specified credentials. The user obtains access to view and search travel ideas and many other features, after authentication. In order to help the user manage his preferences, he has the possibility to add ideas to his wish list, and, because preferences can change, to delete them. To improve the process of decision, reviews from former travelers are displayed.

In the previous chapters I argued the importance of a responsive design, meaning that it should adapt on every mean of communication, no matter the screen size and the resolution. The website I have designed is fully responsive, having elements that are stylized depending on the type of device (mobile, tablet, netbook, laptop, computer and 4K screens).

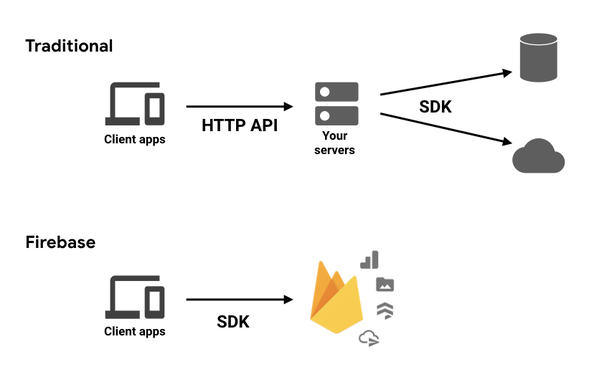
Its design is intuitive and so is its functionality, realized by adding representative icons and names for different page sections. According to studies presented above, user experience is improved if he doesn’t access an overloaded and overfilled user interface.

## Database

For developing my application, I have used Firebase, as I previously stated in the above chapter. Its ability to handle real-time updates between devices, to be readily available everywhere, to process very large database faster than existing solutions, and its data protection and relatively lightweight implementation made it suitable for use in the various online platforms.

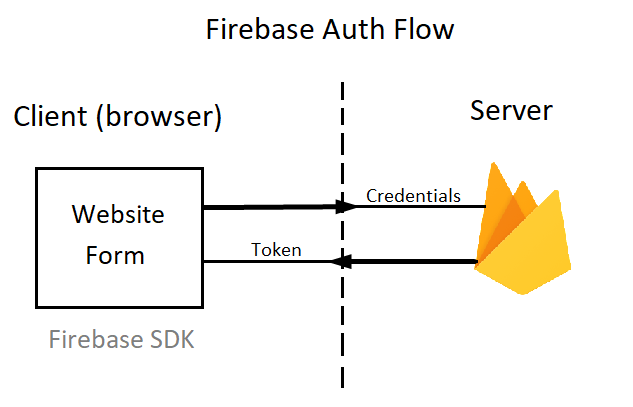
Google Firebase Database is a NoSQL database with out of the box API connectors and wrappers for query purposes. Rather than building REST API in the traditional way of connecting the client to the database, with Firebase I used the SDK to do the same purpose. As a result, I cut the development time by removing the API component.

The traditional way has an inefficient architecture, due to the speed of updates to get new sets of data on the client side. Using Firebase, a client can be automatically triggered for refresh via callbacks as soon as an update was made in the database. Instead of HTTP request, I used data synchronization.



**Figure 4.1** Firebase Database vs Traditional Database

Firebase also comes with a built-in authentication module that supports Google, Facebook, Twitter, GitHub, Yahoo, Microsoft, Apple and the basic username and password. I have done the integration of this module in my app through their SDK, using the email and password authentication. Also, a confirmation email could be sent to the user after registration, as well as an email for password reset or email address changes. The user could use a two-factor authentication, like SMS verification. The authentication flow could be seen in the image below.



**Figure 4.2** Firebase Authentication Flow

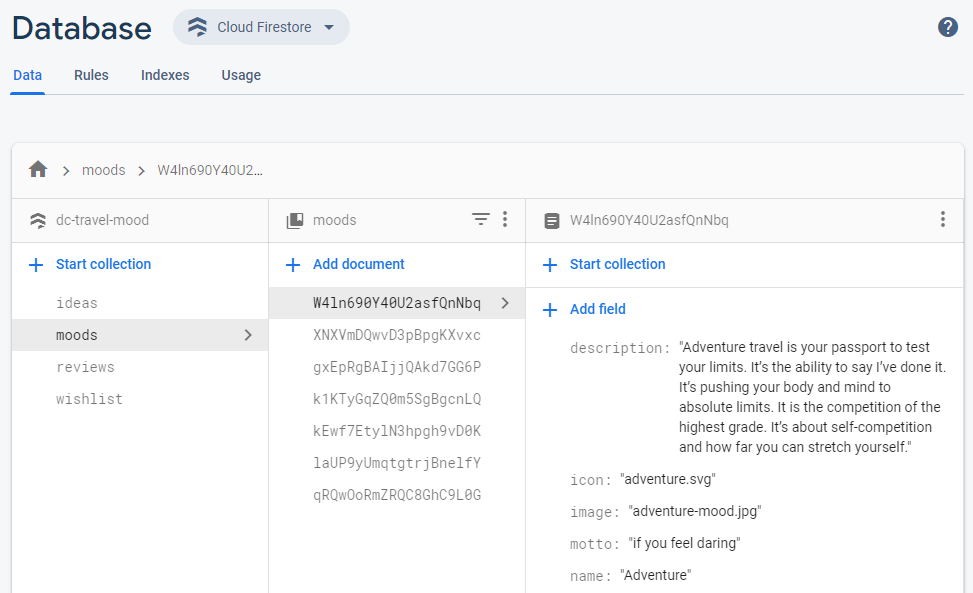
The users’ database has multiple features on Firebase console to be used by developer, like searching for a certain user account in order to reset his password, disable or delete his account.



**Figure 4.3** Firebase Authentication Console

All data used in the app is saved in the Firestore, in collections, as described below:

* Moods – contains information such as the type of mood, motto, the representative icon (saved in Firebase Storage), a short description to help user identify the correlation between his wishes and the filter, and the background image(saved in Firebase Storage) which has a visual impact on the user;
* Ideas – intended for the information regarding travel ideas, which contains the travel idea name, the corresponding mood, the location, a short description to inform the user about the details of the trip, an expectations field that contains different tips and tricks that could help the user to bring the best out of the travel experience and a counter, for statistics about which idea has the highest number of views, therefore, being the most popular;
* Reviews – has as scope the storing of users’ feedback about a certain destination and is linked to the users’ and ideas’ collections by id, in order to make a connection. It also includes the date on which the user wrote the review;
* Wishlist – is entirely made of links to user account and travel idea, having as purpose to store the interconnections between the travel ideas preferred by the user and the user himself.



**Figure 4.4** Firestore Console

## Diagrams

This chapter has the role to give the full particulars of the project and to reveal the design methodology that will lead to the implementation of the web page. The web application should optimize the process of searching trip ideas, provide information anytime, on any device, for better client experiences.

It is designed for 2 types of users: Client and Administrator. The client represents the common visitor of the web, that can create/access an account and view data. The administrator has rights of add, update and delete data displayed on the website, as well as to hide functionality for unauthenticated users. The administrator can manage the database both from the Firebase Console and from the application’s frontend. For better security, the users’ credentials are managed only from the Firebase Authentication Console.

The client part of the web application we discussed about consists of:

* Sign Up form
* Log In form
* Homepage

The unregistered client could see website info, for example contact details, terms and conditions, privacy policy and follow on different platforms, but no other pages are available. In order to obtain full access to the website, the client should authenticate.

After authentication, the menu is visible, and the following are displayed:

* Different filters and search bar to help him/her identify the preferred trip destination (travel moods, travel experiences).
* List of destination ideas page

After choosing, a list of travel destinations will be displayed.

* Certain destination idea page

After a destination is selected, more information will be provided (place, description, expectations), as well as a comment section where useful reviews from former travelers should be found.

* Travel Planning Details
* Wish list page

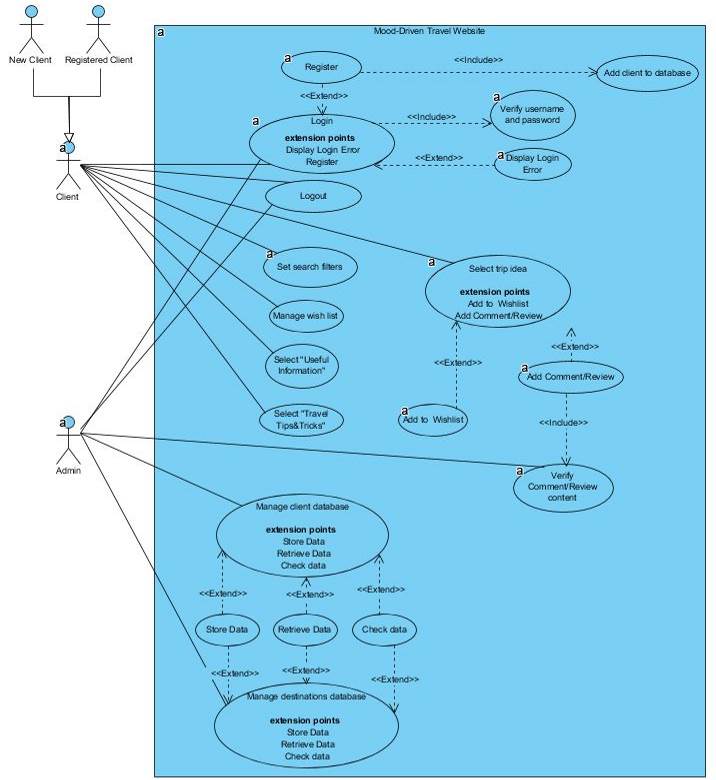
If the client likes the trip idea, he could add it to wish list. After that, after every log in, the client can keep track of his/her favorite holiday ideas.

* Most read trips ideas – displayed on homepage based on number of views
* Resources
* Account info – where users have the possibility of updating email and password.
* Log out option

The administrator has as tasks to:

* Manage the users’ collection from Firebase Authentication Console
* Manage the destinations’ collection – both from front-end and Firestore Console
* Manage the moods’ collection – both from front-end and Firestore Console
* Verify the comments/reviews content from Firestore Console

The detailed Use Case Diagram, that shows the relations between the actors of this system and the system itself, the types of users (new and existent) and administrator, their permissions, that are set from Firebase Console, on the different functionalities of the platform implemented for searching ideas for the different kind of trips based on the user’s current mood.

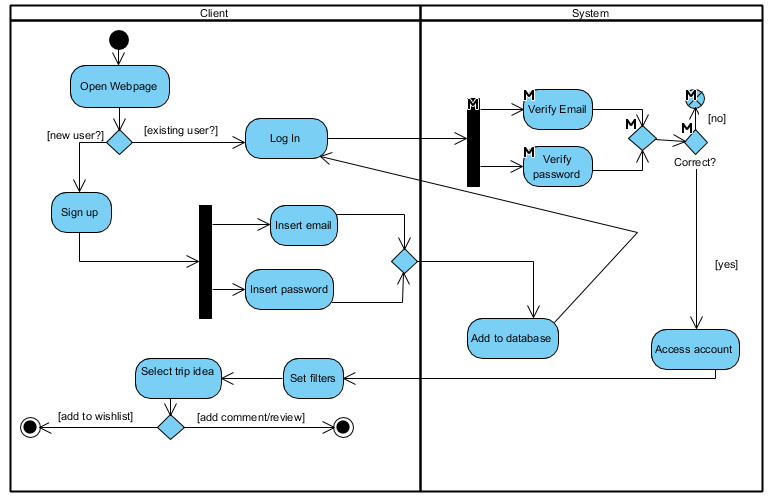


*………………*

*………*

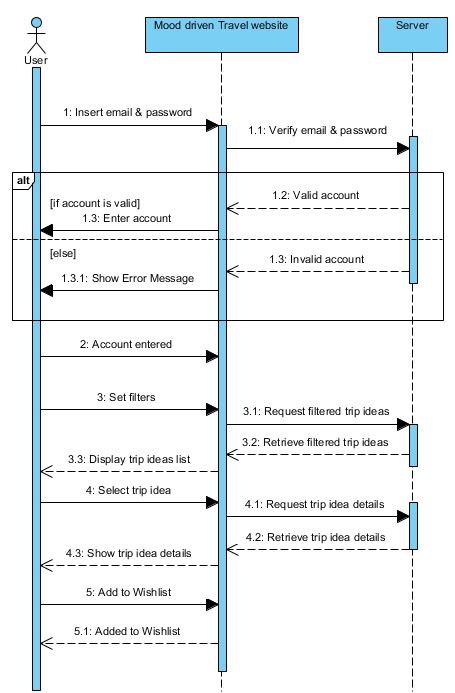
**Figure 4.5** Use case diagram

The next one is the Activity Diagram, which represents the activity flow in which the system is portrayed. The figure 4.4 shows how the activities are organized in order to allow the user to perform an adding to wish list.



**Figure 4.6** Activity diagram

In the Figure 4.5 it is displayed the sequence diagram that presents the steps a user must make in order to add a trip idea to his/her wish list.



A user interacts with the website and, implicitly, with the server which verifies the correctness of account details, consisting of the email and the password, and which returns the requested data.  
After that, the user can use all the functionalities provided by the travel web service. The filtering options are available only for the authenticated users. The common visitors have no access over any if the data. There are some extra features for which it is worth authenticating, like adding the trip idea to the personal wish list page and posting a review concerning a certain destination. The possibility to share your opinion with other travelers adds the personal and social part to the application.

**Figure 4.7** Sequence diagram

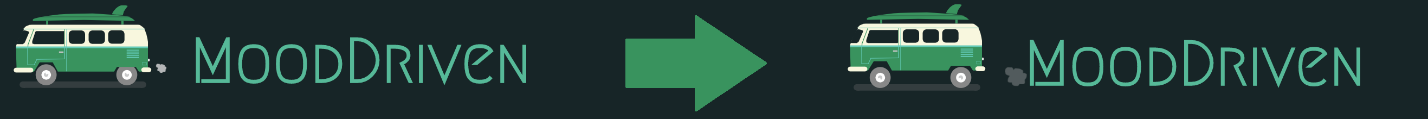
# Mood-Driven Travel’s implementation

My web application’s main purpose is to offer a simplified search for all the steps a traveler has to make when planning a journey. It offers a user-friendly interface and, after authentication, it displays ideas of travel based on some filters, the necessary preparations for obtaining the most of a travel experience, the supposed expectations a traveler has when he first visits a location and the possibility to store the preferred ideas on the personal wish-list which further could be accessed by him.

The front-end part of the application was implemented using HTML5 – for the skeleton, CSS – for the design and JavaScript – for the interactivity obtained by selecting HTML elements and adding event listeners on them.

I have used HTML5 semantic tags to make the code structure clearer and more logical. As a result, the header, navigation menu, the main section and the footer content are distinguishable, being marked with the appropriate tags – <header>, <nav>, <main> and <footer>. When marked with <div> tag having a class that specifies the section, the code is harder to read, both by the developer himself and by other developers.

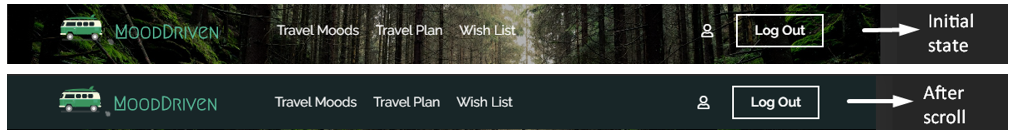
The power of scalable vector graphics animation (with the <svg> tag) was used to obtain the logo of my website. The logo is represented by a hippie van that appears to be moving thanks to CSS keyframes feature. The keyframes modifies the car’s position by changing the bottom property per each fraction of time, as it can be observed in Figure 5.2.



**Figure 5.1** Logo transition

The header component also has some effects that make it more attractive for the users’ eyes. In the initial state it is positioned on the top of the page and its background is transparent, so you can entirely see the background image of the hero element which is displayed on the full screen width and full screen height of the user’s device.

When user starts scrolling down the page’s content – the header’s styles are changing with the visible transition of 0.5 seconds. The background color is added with a box shadow and its position is changed from absolute to fixed which makes it to stay all the time on the top of the screen, as shown in the Figure 5.3.



**Figure 5.2** Header transition

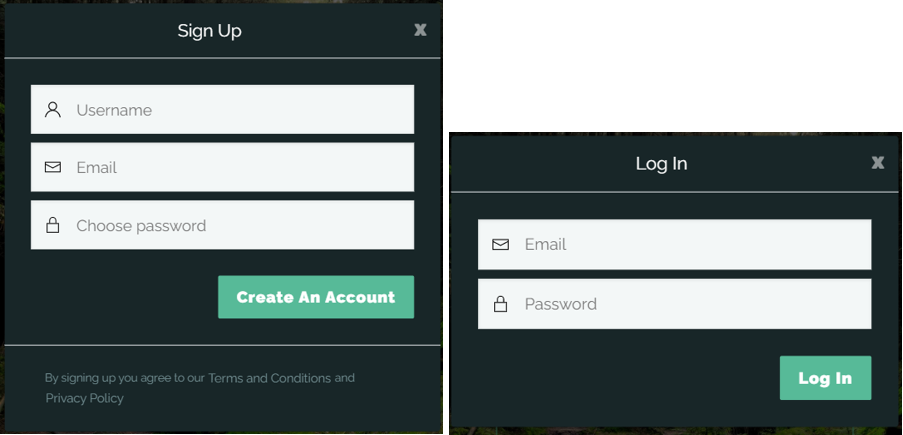
The Firebase Authentication gives the developer the ability to specify whether a registered user state should be cleared when the window closes, cleared when the page is reloaded or should be persisted indefinitely until explicit command, when user signs out. For my application I used the behavior, where the user’s session persists even after browser closing. That approach is convenient because the user is not required to re-login every time after closing the page on the same device. When a guest, an unregistered user, accesses the website from his browser, most of the functionality and all the data are unavailable.

If the user is not authenticated, in the menu appear 2 buttons, stylized with Bootstrap and CSS. One button triggers the Sign-up modal on click, with the following fields: username, email and password. When the user wants to create an account, Firebase Authentication functions are called, verifying the uniqueness of the email and the password length. In the modal’s footer, is a notice about term and conditions agreement and privacy policy, both triggering overlaying modals, to make the user aware of the use of personal info, according to GDPR.

If registration succeeds, the user is saved in the database and is logged in.

If the Log in button is clicked, the corresponding modal is shown and already registered user enters his credentials - email and password, and the Firebase Auth verifies the correctness.   
For both modals, implemented using the Bootstrap modal plugin, corresponding authentication errors appear.

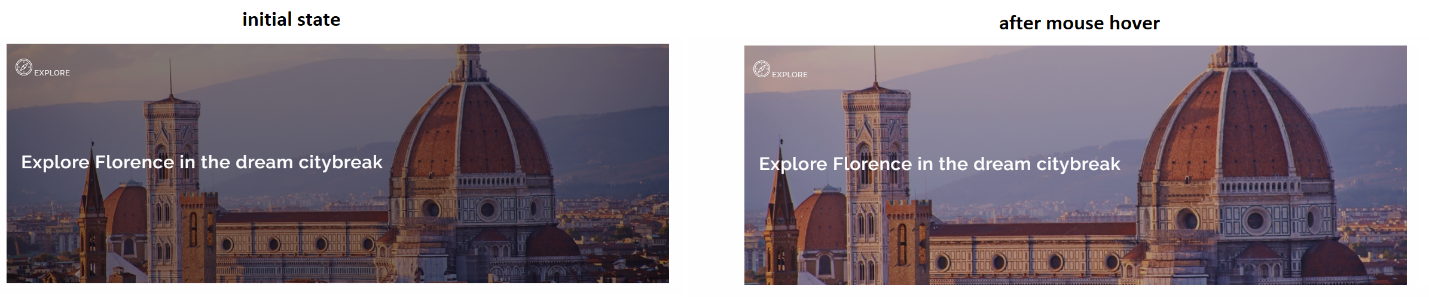
I preferred to use modals for the authentication part because reloading the page after just clicking on the log in respectively sign up button from the menu would take too much time if my approach would have been to make two separate HTML pages.



**Figure 5.3** Sign Up / Log In forms

After the authentication step is passed, the user has access to a multitude of features and data. In the menu, the buttons described above are replaced with two other buttons: one that triggers a modal – Account information and one for Log Out. By accessing the Account Information modal, the user has as options to change his email or his password, which are saved in the database after user confirms these changes.

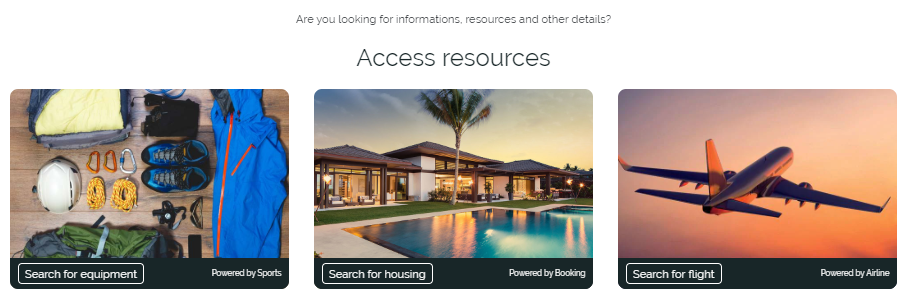
The “Most read travel ideas” section has as components items of different sizes having background image with dark faded overlay and one more overlay that contains the travel mood type with the corresponding icon and trip idea’s title. On hover, the faded overlay gradually disappears by increasing its opacity and the background image is zoomed in. The result of this effect can be seen in Figure 5.4.



**Figure 5.4** Most read travel ideas section transition

The ideas in this section depend on the numbers of users that accessed them, being generated only the ones with the highest ranking, stored in the database as a counter.

In the last section from the webpage’s main components, it is situated a carousel element which displays resources that could be accessed by the user. Each resource is placed in a carousel item. By clicking on one of them, the new tab with a resource’s official website will be opened. These items are animated by sliding the whole list with an interval of 5 seconds. Also, there are two buttons to make it possible to navigate between them manually. This feature was implemented using Bootstrap 4 and the results can be seen in Figure 5.5.



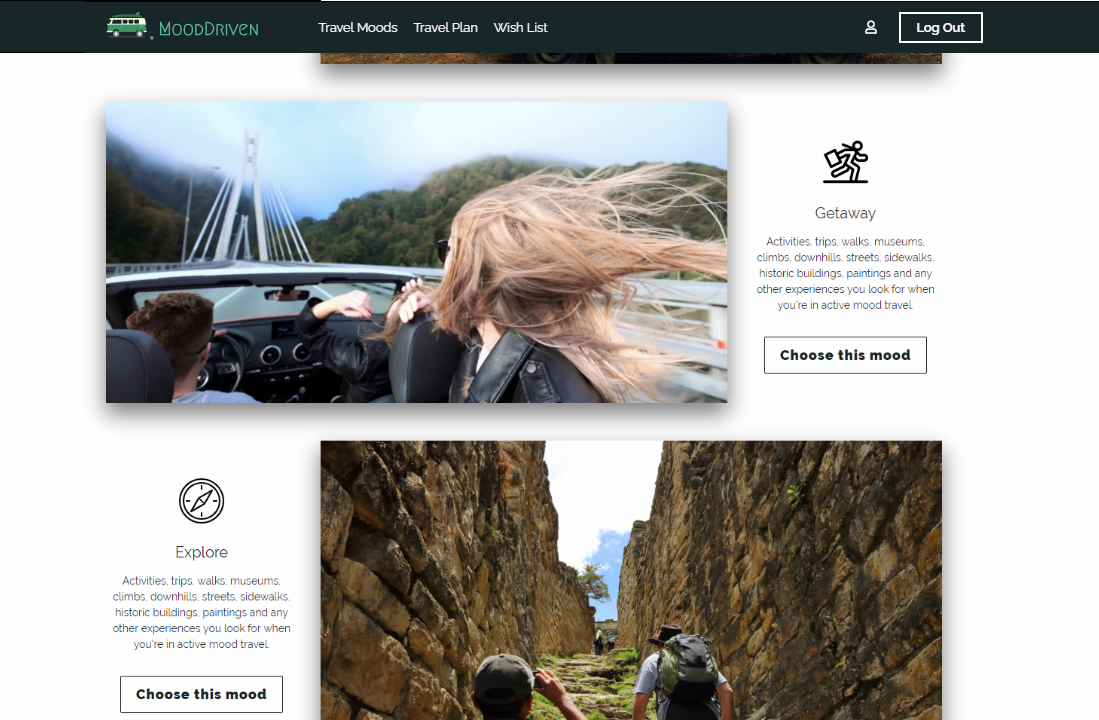
**Figure 5.5** Resources carousel

In the footer section could be found contact information, a list of travel moods with corresponding links, about us section with the modals displaying the terms and conditions and the privacy policy, and two social links, to Instagram and Facebook.

After authentication, the menu is populated with three more options: Travel Moods, Travel Plan and Wish list. Travel Moods has a dropdown attached which is displayed on hover, containing the name of the travel mood and its motto, taken from the Firestore. When this option is clicked, the Travel Moods page is opened. On that page is generated a list of travel moods, of course, from JavaScript, which is populated from the Firestore collection named “moods” with the name, icon, background and description. A “Choose this mood” button is in each item of the list, with a link to that mood.

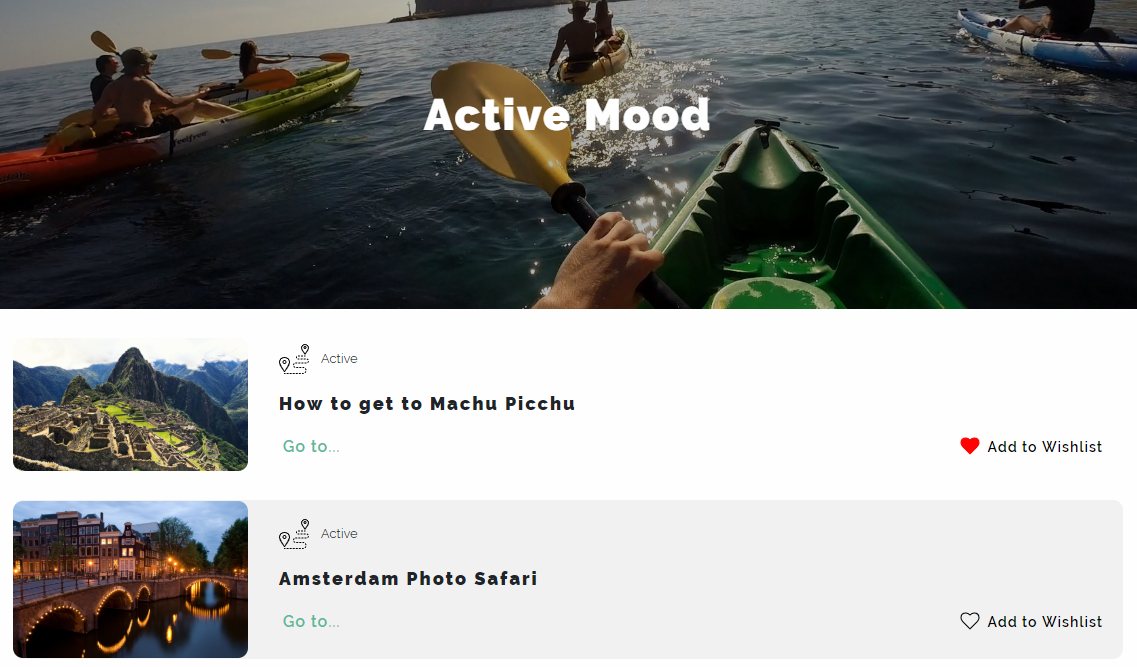
The admin, recognized at authentication by Firebase Authentication, has the right to add, edit and delete moods and ideas from the frontend. On his dedicated page, he has a form to add a mood which is saved in the database and attached, without refresh, to a list with similar design as that displayed on the Travel Moods page. The difference is that when hovering on an item, he sees 2 options: to edit or to delete it. The same functionality is for managing ideas.

There are 6 travel moods: Active, Adventure, Getaway, Explore, Relax and Romantic. The aspect of a section of the page is shown in Figure 5.6.



**Figure 5.6** Travel Moods page

After selecting the desired travel mood, from that list or from the menu’s dropdown, the user is redirected to the list of ideas with the respective mood. He has as options two buttons: to go to the travel idea’s full specification page by clicking on “Go to…” button, or to add the idea to wish list, and, as well, to remove it from there. When user presses the button “Add to wish list”, the event onclick runs the function assigned to it. This function creates request with the data about the selected idea and sends it to the Firebase, which stores the information about the user’s preferred idea to the database and establishes a relation between them. The ideas that are in the wish list, are marked with a filled red heart. By clicking multiple times on that button, the icon toggles between filled and unfilled, and the data are added and deleted from the Firestore.



**Figure 5.7** Search based on travel mood filter

The “Travel Plan” option from the menu displays a page with information about all the aspects to be considered when planning a vacation, the necessary steps and some recommendations.

The wish list page consists of a list with all ideas selected by the user and for each added idea, a link to its page and a delete button are present.

At log out, the session is closed, and the user is redirected to the initial homepage for unauthenticated users.

# Conclusions

As a conclusion for my graduation thesis, I am going to present the strong points of the application, some improvements that can be made to this website, but also the degree of completion of the application.

The solution I implemented manage to achieve its initial purpose, namely to provide a place where people that want to travel could find ideas that are made for their psychical improvement and could inform about the necessary steps in trip planning. The user is able to sign up and to log in successfully. After authentication, more pages are available to use, and he has the option to search traveling ideas and save the ones that seemed suitable for his interests. Also, he can enter on the “Trip Planning” page and discover a detailed list of steps to follow when planning.

The implemented solution was adapted for all types of smart devices, giving the user the possibility to access it and have a nice experience with full functionality, indifferent of the mean of communication he has at his disposal.

The attractive design and animated visual effects improve the user’s experience who might maintain his attention for a longer period of time. The most part of the improvement would be made based on user’s feedback and on their requirements.

A drawback consists in the limited data from the travel ideas collection that would improve if a travel agency would provide some recent and up-to-date information from the tourism sector. The application could be managed by a travel agency if the booking part is implemented, in order to allow the user to obtain the trip reservation and not only the proposal.

An improvement that would help the application’s evolution is founding real partnership, as none of the partnerships presented on the webpage is existent or a valid source of information, being simulated.

To conclude, an online travel page is a great source of inspiration among travelers and tourism is a continuous expanding domain, permitting all types of improvements. My opinion is that this type of platform can attract numerous persons in search of a real travel experience they would always remember about. Therefore, every travel website should also be based on user’s inner living, and not only a profitable business.

# Bibliography

|  |  |
| --- | --- |
| [1] | P. S. Stephen Gosch, Premodern Travel in World History, London: Routledge Taylor&Francis Group, 2007. |
| [2] | Google Travel, "The 2014 Traveler's Road to Decision," June 2014. [Online]. Available: https://www.thinkwithgoogle.com. [Accessed 12 January 2019]. |
| [3] | J. W. Daniela Turcu, Economia turismului, Timisoara: Editura Eurostampa, 2008. |
| [4] | T. O. Chris Lee, "Blurtit," Blurtit, June 2006. [Online]. Available: https://science.blurtit.com. [Accessed 23 03 2019]. |
| [5] | Statistics Explained, "Eurostat," Eurostat, Luxembourg, 2016. |
| [6] | M. S. M. D. A. W. David Snepenger, "Meanings and Consumption Characteristics of Places at a Tourism Destination," in *Journal of Travel Research Vol.45*, New York, Sage Publications, 2007, pp. 310-321. |
| [7] | K. May, "How 25 years of the Web inspired the travel revolution," The Guardian, 12 05 2014. [Online]. Available: https://www.theguardian.com. [Accessed 20 05 2020]. |
| [8] | W. D. Chalmers, On the origin of the species Homo touristicus, Bloomington: iUniverse, 2011. |
| [9] | B. Vidal, "Tourism and Technology: How Tech is Revolutionizing Travel," August 2018. [Online]. Available: https://www.wearemarketing.com. [Accessed 20 February 2019]. |
| [10] | B. Baader, "Virtual Tourism: the future of travel?," Prezi, 21 04 2015. [Online]. Available: https://prezi.com/. [Accessed 05 03 2020]. |
| [11] | Google Travel, "The 2013 Traveler's Road to Decision: Affluent Insights," January 2014. [Online]. Available: https://www.thinkwithgoogle.com. [Accessed 27 03 2019]. |
| [12] | Oxera Consulting LLP, "Benefits of online platforms," Oxera, Brussels, 2015. |
| [13] | J. Delago, "What travel marketers should know about people searching for experiences," May 2019. [Online]. Available: https://www.thinkwithgoogle.com. [Accessed 20 May 2019]. |
| [14] | K. May, "PhocusWire," 21 October 2013. [Online]. Available: https://www.phocuswire.com. [Accessed 22 May 2019]. |
| [15] | Firebase, Inc., "Firebase," Google, [Online]. Available: https://firebase.google.com/. [Accessed 15 05 2020]. |
| [16] | Refsnes Data, "W3Schools," Refsnes Data, 01 2019. [Online]. Available: https://www.w3schools.com. [Accessed 26 02 2019]. |
| [17] | J. Jung, "The Conversation," Edith Cowan University, March 2017. [Online]. Available: http://theconversation.com. [Accessed 28 March 2019]. |
| [18] | D. East, "The Firebase Blog," Firebase, 06 07 2016. [Online]. Available: https://firebase.googleblog.com/. [Accessed 12 06 2020]. |
| [19] | S. Pelling, "Build Web Apps with Vue JS 2 & Firebase," Udemy, 2018. [Online]. Available: https://www.udemy.com/. [Accessed 05 03 2020]. |
| [20] | K. Taulien, "Complete Web Developer Course," Udemy, 06 2018. [Online]. Available: https://www.udemy.com/. [Accessed 15 03 2019]. |

# Annex 1 – The figure list

[**Figure 1.1** Search Terms that Leisure Travelers use in Planning. Source: [2] 3](#_Toc43768460)

[**Figure 2.1** Evolution for different sectors of the economy, 2012-2017. Source: [5] 5](#_Toc43768461)

[**Figure 2.2** Number of persons employed, 2017. Source: [5] 6](#_Toc43768462)

[**Figure 2.3** Yahoo! website in 1994. Source: [7] 8](#_Toc43768463)

[**Figure 2.4** Importance of sources for inspiring personal travel. Source: [7] 9](#_Toc43768464)

[**Figure 2.5** Usage of smartphone for inspiring leisure travel. Source: [7] 10](#_Toc43768465)

[**Figure 2.6** Types of devices used by travelers. Source: [2] 11](#_Toc43768466)

[**Figure 2.7** Consumer perceptions of benefits. Source: [11] 13](#_Toc43768467)

[**Figure 2.8** TripAdvisor landing page. Source: https://www.tripadvisor.com 14](#_Toc43768468)

[**Figure 2.9** Kayak landing page. Source: https://www.kayak.com 15](#_Toc43768469)

[**Figure 2.10** Instagram travel page. Source: https://www.instagram.com 15](#_Toc43768470)

[**Figure 3.1** Example of the CSS rule. 20](#_Toc43768471)

[**Figure 3.2** Example of declaring styles for the elements with class "test-class123". 21](#_Toc43768472)

[**Figure 3.3** Destructive assignment in ES5 and ES6 25](#_Toc43768473)

[**Figure 3.4** Three different methods of writing the same function 26](#_Toc43768474)

[**Figure 3.5** Joining strings with variables in ES5 and ES6 26](#_Toc43768475)

[**Figure 3.6** An example of creating a class and its child class in ES6 27](#_Toc43768476)

[**Figure 3.7** Examples of the Bootstrap icons 28](#_Toc43768477)

[**Figure 3.8** Example of Bootstrap grid 29](#_Toc43768478)

[**Figure 3.9** Usage of the "hidden" class in Bootstrap 30](#_Toc43768479)

[**Figure 3.10** Example of an initialized Firebase project 31](#_Toc43768480)

[**Figure 4.1** Firebase Database vs Traditional Database 36](#_Toc43768481)

[**Figure 4.2** Firebase Authentication Flow 37](#_Toc43768482)

[**Figure 4.3** Firebase Authentication Console 37](#_Toc43768483)

[**Figure 4.4** Firestore Console 38](#_Toc43768484)

[**Figure 4.5** Use case diagram 40](#_Toc43768485)

[**Figure 4.6** Activity diagram 40](#_Toc43768486)

[**Figure 4.7** Sequence diagram 41](#_Toc43768487)

[**Figure 5.1** Logo transition 42](#_Toc43768488)

[**Figure 5.2** Header transition 43](#_Toc43768489)

[**Figure 5.3** Sign Up / Log In forms 44](#_Toc43768490)

[**Figure 5.4** Most read travel ideas section transition 44](#_Toc43768491)

[**Figure 5.5** Resources carousel 45](#_Toc43768492)

[**Figure 5.6** Travel Moods page 46](#_Toc43768493)

[**Figure 5.7** Search based on travel mood filter 47](#_Toc43768494)