

# Methods of Recommendations for Analysis of Computer Components

Oleh Veres, Pavlo Ilchuk and Olha Kots

*Lviv Polytechnic National University, Stepana Bandery str. 12, Lviv, 79013, Ukraine*

## Abstract

The article is devoted to the study of the problem of analyzing computer components for the purpose of easy construction of computers, their complete analysis and improvement of information and technical assistance to users using information technology tools.

The paper describes the process of information and technical assistance to users with various computer problems. The need for the development of a computer components analysis system for easy computer design, their complete analysis, the creation of problem analytics and ways to solve them, and the improvement of information and technical assistance to users with computer problems has been determined. An analysis of approaches to the application of the methodology and decisions regarding the analysis of computer components was carried out, as well as the methods of providing recommendations were investigated. It is best to use recommendation methods to generate a set of components offered to the user. In the case of computer components, it is more appropriate to provide recommendations for groups of users than to provide recommendations for individual users. A mixed categorical-numerical clustering method was used to search for user groups, which uses numerical rating and demographic characteristics of users, and a hybrid method for user group search was used, which is based on the sparsity coefficient of the user-subject matrix. The algorithm of operation of the hybrid recommender system, which offers computer components depending on the options of formulated user requirements, is described. A weighted hybrid mechanism was used to provide recommendations. A conceptual model of the system was designed using the tools of the UML language. C# and .NET were chosen as the source of programming for the implementation of the system application prototype, for the implementation of the functional component and the creation of the user interface. The MySQL database management system was chosen to work with the database. Two types of software product testing were conducted - interface and functionality. As a result, both types of testing did not reveal any errors or bugs. The recommendation system correctly finds the type and parameters of personal computer components with the accuracy of 87.5%.

## Keywords

Analysis, Clustering Methods, Computer Components, Methods of Recommendations, Recommendation System

## 1. Introduction

Modern information technologies are developing very rapidly. Every year, new models of computer equipment, which are better and of higher quality, replace their predecessors, ahead of them in all parameters. Accordingly, the thought arises whether the "freshly baked" technology is really the best of its kind and whether there is really a great need to buy it. This problem is best seen when using a computer or laptop. After all, they consist of almost a dozen different components, which in the end must work perfectly with each other, providing the user with maximum productivity and safety of use.

---

MoMLeT+DS 2023: 5<sup>th</sup> International Workshop on Modern Machine Learning Technologies and Data Science, June 3, 2023, Lviv, Ukraine  
EMAIL: oled.m.veres@lpnu.ua (O. Veres); pavlo.g.ilchuk@lpnu.ua (P. Ilchuk); olha.o.kots@lpnu.ua (O. Kots)  
ORCID: 0000-0001-9149-4752 (O. Veres); 0000-0003-4636-2309 (P. Ilchuk); 0000-0001-7123-3635 (O. Kots)



© 2023 Copyright for this paper by its authors.  
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).  
CEUR Workshop Proceedings (CEUR-WS.org)

The user has the opportunity to view many alternatives, compare them and choose the best option for him. Being in such a constant flow of alternative offers has led to the fact that working out useful positions on your own is not an easy task.

Recommendation systems help to solve the problem of such information overload [1]. The main task of the recommendation system is to provide personalized recommendations to the user that take into account his preferences when choosing items (goods, objects or services).

Computers available today can satisfy the requirements of any person: for work, for games, combined, for photo and video processing. When the buyer is faced with such a limitless choice, a big problem arises here - what exactly to buy and whether this or other computer will suit him. In order to solve such a dilemma, it is necessary to develop a system for analyzing computer components using the methods and technologies of recommendation systems [1].

## **2. Analysis of Recent Research and Publications**

Today, in times of war and quarantine restrictions due to the COVID-19 pandemic, the market for computer equipment and accessories is empty. Due to the actions of the so-called "repurchases", which buy up batches of all new components, their prices become higher than those specified in advance by the manufacturer.

There is a need for analysis and selection of computer components, saving time and money. It is necessary to find a solution that can select options based on the characteristics of each person or general trends. Therefore, the research and development of an information system to improve computer configuration analysis, which will use the recommendation method to offer users relevant suggestions for the selection of computer components, is very relevant.

In the 21st century, it is almost impossible to imagine life without modern technologies, to which humanity has become accustomed and dependent on them. They make life easier for us at home, at work, in education and give us the opportunity to travel. But the most common modern technology that almost every person has is a smartphone and a computer. The computer itself can be a "desktop", with a monitor and a system unit, or in the form of a laptop. A person today, using a computer, performs the most diverse tasks in terms of complexity and duration, safely for his life [2].

A personal computer consists of many parts, and together they must show maximum performance and clear operation. Unfortunately, most people do not have knowledge in the field of computer construction, the interaction of components and their functioning. That is why there are many options for solving this problem: consulting with experts, online or directly, questionnaires, using various "benchmarks" to simply show the characteristics of this or that detail [3].

Such "benchmarks" most often use a simple "yes/no" scheme. The selected part either fits your assembly or not, and then the user is forced to continue searching, not understanding whether he will be able to find something and how much time it will take. However, if you create a "benchmark" based on a recommendation system that will quickly, using a database, find the necessary data, sort it and show it to the user, such a project will be successful.

### **2.1. Review of models of recommendation systems**

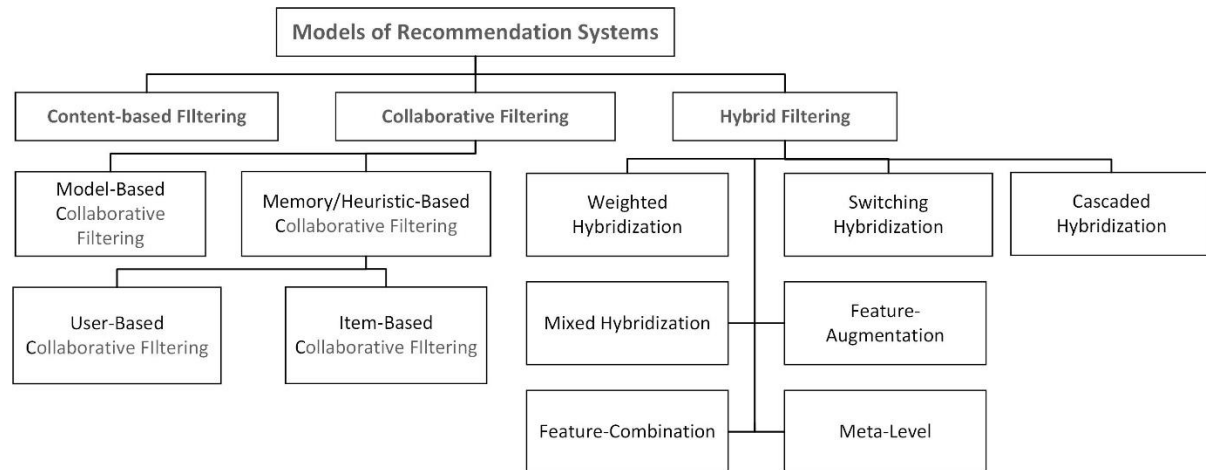
For the first *Collaborative Filtering model* proposed in the 1990s was developed, recommendation systems have been actively studied and applied in all fields of science and industry [4]. Recommendation systems are a useful technology that can alleviate the problem of information overload provided to users. It provides an evaluation of the components recommended to the user, creates a list of recommendation ratings for each user, and makes it possible to recommend products related to the user [5]. Recommendation systems are information filtering systems that provide personalized recommendations about components to a user in a service environment that can store or collect various data. Information filtering, which is mainly used in recommendation systems, is adapted to the user's preferences or offers only components that are considered useful to the user [6-9].

This technology allows the user to use as little time as possible to find any information that he is interested in. Such systems compare collected or generated data from the user and generate a list of

components to recommend to the user. It is like a kind of alternative to the search system, as it quickly helps to find data that users would not be able to find on their own [10-14].

Recommendation systems are one of the most popular applications of intelligent data analysis and machine learning in the field of Internet business. They analyze the behavior of users of the Internet service, after which they give a quantitative and qualitative assessment of the preferences of users for this or that subject. The objects of recommendations can be products in the online store, a set of sections of the Web site, media content, other users of the Web service.

Modern recommendation systems can be classified by filtering methods, namely: content-based filtering, collaborative (joint) filtering, hybrid filtering (Figure 1).



**Figure 1:** Classification of models of recommendation systems

In 1992, starting with the study of Loeb et al. [15] various models of information filtering appeared. **Content-Based Filtering**, CBF, is a method for recommending items with attributes similar to those that users like, and recommends them based on the item information [16]. That is, it is a method of recommending similar items based on information about items that have been chosen by the user in the past. Such properties for a personal computer, for example, can be the type of RAM, the size of the hard disk, the brand of the central processor, etc. The very idea of such filtering is that properties with similar content are given similar preferences by users. Also, content filtering can be used in systems where the availability of descriptive data is primarily assumed [17]. It can be filtered based on knowledge (Knowledge-Based) – offers products based on preferences and conclusions about the needs of users. It can also be filtered by certain indicators, for example, demographic (demographic filtering) – provides recommendations based on the user's demographic profile (for example, age, profession) [18]. Recommendations can then be created for different demographic groups, even combining user ratings within those groups.

The content-based filtering model only recommends data closely related to items previously rated by the user, so the system is known for its limitations as it cannot recommend new items [19]. Because of these limitations, this model is mainly used in services that recommend items or text data based on information about the item and the user's profile. The content-based filtering model uses text mining technology to identify user preferences, semantic analysis [20], TF-IDF (term-frequency, inverse document frequency) [21], neural networks [22], naive Bayes, and SVM [18].

**Collaborative filtering** is a model of information filtering that first appeared in the 1990s [23, 24]. Collaborative filtering is a model that builds a database of user preferences by using user evaluation data to predict products (items, elements) that match the user's taste, and then uses it for recommendations [15]. This model is divided into collaborative filtering based on memory (Memory/Heuristic-Based) and collaborative filtering based on the model (Model-Based) [25]. Memory-based collaborative filtering can be further divided into User-Based Collaborative Filtering and Item-Based Collaborative Filtering.

User-based collaborative filtering is a model that compares the similarities between users by comparing the evaluation data of the same item for each user, and then creates and recommends a list

of the top N items. A user-item rating matrix is created to predict recommendations using the similarity between items. In general, memory-based collaborative filtering uses techniques such as Pearson correlation, vector cosine correlation, and KNN to create similar groups (neighborhood groups) among users and recommend items to users within the same group [26].

However, if the model does not contain enough data, three problems can arise: sparsity, cold start, and “gray sheep”. First, the sparsity problem is a problem that occurs when not enough data are available to make a recommendation [27]. Similarly, the cold start problem occurs when there is no evaluation data, i.e., the first evaluator due to the influx of new users at the beginning of the application [28]. Finally, gray sheep is a problem in which recommendation difficulties arise when the set of users whose evaluation data is similar to that of an individual user is too small [29]. To solve this problem, model-based collaborative filtering has been proposed, which evaluates or learns a model for prediction using user evaluation data [30]. Methods such as clustering, SVD, and PCA have mainly been used for model-based collaborative filtering.

Both filtering models have limitations, as the content-based filtering model relies on subject metadata, while collaborative filtering relies on user ratings of the subject. A **hybrid recommendation model** (Hybrid Filtering) was proposed to eliminate the limitations of both recommendation filtering models and to improve the effectiveness of recommendations [31]. The hybrid recommendation model is divided into seven types: weighted hybridization, hybridization with switching, cascade hybridization, mixed hybridization, feature combination, feature expansion, and meta-level (see Figure 1) [32].

Since the hybrid recommendation model is mainly designed to solve the problem of sparsity, the main goal of most studies on the hybrid recommendation model is to compensate for the lack of ranking data by integrating the information of content-based filtering and collaborative filtering models.

## 2.2. Comparison of existing computer component analysis systems

Today, the following analogues of computer component analysis systems are available on the market: CPU – Z; GPU – Z; AIDA64; Speccy. All of them are desktop applications for Windows operating systems and some of them are available as mobile applications in the Play Market or App Store. Table 1 provides a comparison of peers based on some metrics, namely: platform type, operating system, amount of memory required for installation, unification, functionality, usability, reliability, performance, usability, monthly subscription price, and rating (Table 1).

**Table 1**  
Comparison of Analogue Systems

Characteristics	Programs			
Platform	CPU-Z	GPU-Z	AIDA64	Speccy
OS	Windows	Windows	Windows, Android, IOS	Windows
Memory capacity	300 Mb	440 Mb	1,1 Gb	860 Mb
Unification	-	-	+	+
Functionality	average	average	high	average
Suitability for use	high	high	high	average
Reliability	high	high	high	average
Productivity	average	average	free	низька
Serviceability	high	high	high	high
Price	free	free	60\$	free
Rating	4.8/5	4.9/5	5/5	3.2/5

CPU-Z is a program that is based on displaying the information of computer nodes under the Windows operating system. It was created by the CPUID company to identify central processors. The company cooperates with many well-known brands - Asus Rog, MSI, GIGABYTE Aorus and is

financed by advertising from them. The main function of the program is to study and determine the technical characteristics of the processor, its properties, RAM and motherboard [33].

GPU – Z is a program from the TechPowerUp company, which borrowed the name from the CPUID company and performs the same functions, only with video adapters, namely: it deals with the identification of video cards and graphics processors, their components and properties. Also supports all versions of Windows and graphics adapters from the oldest to the latest. The program is also free, receiving income from donations and cooperation with well-known manufacturers of video adapters [34]. Main functions:

- supports for Nvidia, AMD and Intel graphics devices;
- displays information about the adapter, graphics processor and display;
- detailed reporting on the subsystem, memory: size, type, speed, bus width;
- has a GPU load test to check the configuration of PCI-Express lanes;
- can create a backup copy of the BIOS of the video card.

AIDA64, from the company FinalWire, which has been improving this program for more than twenty years, on the contrary, deals with the analysis and scanning of all components found in a personal computer. Also works with peripheral devices: mouse, keyboard and monitor. The program is paid, but offers a 30-day free period to study the full functionality. The company also cooperates with Ukraine, which is evidenced by the domain of the site and the presence of competent localization of the product. The price of a one-time purchase of the product for home use is almost \$60. Also, AIDA perfectly copes with the analysis of mobile devices on the Android or IOS system, where it is completely free [35]. Main properties: load of any computer element; quick analysis and identification of the problem; technical support service; management of disks, number of power elements and change of streams.

Speccy is a free utility that, like AIDA64, provides users with a display of all system information, as well as information about the computer's hardware. This program belongs to the British company Piriform Limited. The main task of this program is to quickly and correctly display detailed information about the processor, hard disk, RAM, graphics card and operating system. Speccy Portable is also available - a version designed to work with removable devices, i.e. USB flash drives or digital players [36].

### **2.3. Main Tasks of Information System**

Today, almost everyone has a personal computer that sooner or later will need an upgrade. A study of the trends in the modern information technology market has shown that there is a growing need for information systems that make it possible to analyze and eliminate the problem in the fastest available way, as well as to provide proposals for the cheapest high-quality improvement of the computer. Information systems that employ recommendation techniques can quickly and efficiently provide any element that a user needs, thereby saving a lot of valuable time. The urgency of developing an information system lies in the need to form a set of recommendations, taking into account the wishes of the user and optimizing the operation of the computer in general. The effectiveness of the application will improve if methods and algorithms for forecasting recommendations for user groups are developed.

The developed analysis system should, with the help of a block of recommendations, provide the user with the most accurate version of the assembly according to the parameters set by him, which will help in choosing and buying a computer. Elaboration of quantitative and qualitative evaluations of such recommendations will make it possible to provide more relevant suggestions and increase the level of accuracy of solving the user's problem. Also, the analysis of the available computer components will make it possible to identify configuration problems and propose their solution.

The main task of the recommendation system is to provide personalized recommendations to the user that take into account his preferences when choosing items (goods, objects or services). The purpose of the developed computer component analysis system is to save important human resources when choosing a computer.

The target audience of the application is personal computer users of any age. Therefore, the use of the system will be very widespread. This can be a daily quick check of your computer for any problems. Ability to create your own assembly or improve your existing one. Purchase and delivery of components directly to your home or post office.

The main goal of the implementation of the developed project is daily functioning for computer users. The possible addition of functionality is presented in each of the effect types.

Time effect – reduction of users' time spent on searching for the necessary data. Financial effect - costs of unnecessary components will be significantly reduced. Economic effect – creation of own brand, possibility of cooperation with major manufacturers of components, thus expansion of the project on the territory of Ukraine, creation of new jobs [37]. Technological effect - the system will also have an intuitive learning system, which may develop certain knowledge and skills in the field of computer engineering. Environmental effect - does not affect. The ergonomic effect is a light and easy-to-use system. Medical effect - reducing the time spent sitting in front of the monitor will save a considerable amount of health.

The system operation process consists of the following steps:

*Step 1.* The user opens the application and goes through the registration process. Registration is mandatory, for the formation of a better relevant set of recommendations and contact with users.

*Step 2.* Selection of the method of obtaining component parameters - automatic scanning of the user's personal computer or set by the user himself.

*Step 3.* Data filtering.

In order to implement filtering and presentation, it is necessary to create a recommendation system for identifying, classifying and sorting subjects (objects). Such a system will receive a user request, process it and generate a set of recommendations.

In order for a recommendation system to work properly, it needs to create and set parameters that will filter and select data based on a given query. To do this, you need to create a database that will store information about users, components and their combinations.

Therefore, it is necessary to develop an information system for the analysis of computer components using a hybrid model of recommendations. Since the Hybrid Recommendation model is mainly designed to solve the sparsity problem, the main goal is to compensate for the lack of rating data by integrating the information of the Content-based Filtering and Collaborative Filtering models. In this case, providing recommendations to groups of users is more appropriate than providing recommendations to individual users. Clustering methods are used to find groups of similar users [38-40].

## 2.4. Description of the working mechanisms of the recommendation system

### 2.4.1. Description of clustering methods for finding user groups

The formal formulation of the problem of forecasting recommendations for groups of users is as follows.

Let  $U = \{U_1, U_2, \dots, U_n\}$  – a set of vectors of user profiles;  $G = \{G_1, G_2, \dots, G_m\}$  – set of groups of parameters;  $G_i = \{U_1 G_i, U_2 G_i, \dots, U_k G_i\}$  is a set of user profiles for the group  $G_i$ .

It is necessary to make a forecast of recommendations for groups of users  $C_{G_i} = Predict(G_i)$ .

A feature of the user-subject matrix is that it contains a significant number of zero elements (see Figure 2). The number of non-zero elements does not exceed 10% of the total number of elements of the user-subject matrix. Therefore, it is advisable to use demographic characteristics of users to cluster users into groups. The main demographic attributes of users are as follows: age, gender, education, occupation, field of application. Age is a numeric attribute. Gender, education, occupation, field of application are categorical attributes [41].

Let the rating vector of the  $i$ -th user's profile be given by the following vector (1):

$$U_i = (u_{1i}, u_{2i}, \dots, u_{mi}), \quad (1)$$

where  $u_{ji}$  – is the rating of the  $j$ -th subject by the  $i$ -th user.

Let's expand this vector with the demographic attributes of the user (2):

$$U_i^{ext} = (u_{1i}, u_{2i}, \dots, u_{mi}, d_{1i}, d_{2i}, d_{3i}, d_{4i}, d_{5i}), \quad (2)$$

where  $d_{1i}, d_{2i}, d_{3i}, d_{4i}, d_{5i}$  – categorical user attributes.

To simplify the description of the method, let us denote the vector  $U_i^{ext}$  by vector (3):

$$X_i = \{x_{1i}, x_{2i}, \dots, x_{ni}\}. \quad (3)$$

We get a mixed vector of the user's profile, which contains numerical and categorical values (2). Clustering of mixed vectors of user profiles is carried out using the method of mixed clustering [40]. The method of mixed clustering is based on the calculation of the density of placement of mixed vectors of user profiles and determines the number and position of cluster centers. The density is defined as the number of vectors of user parameters that are in the neighborhood of radius  $d_c$  near each user in formulas (4) and (5).

$$\rho_i = \sum_{j=1}^N f(d_{ij} - d_c), \quad (4)$$

where  $d_{ij}$  is the distance between the  $i$ -th and  $j$ -th vectors of user parameters;  $d_c$  – limit value;  $N$  is the number of users.

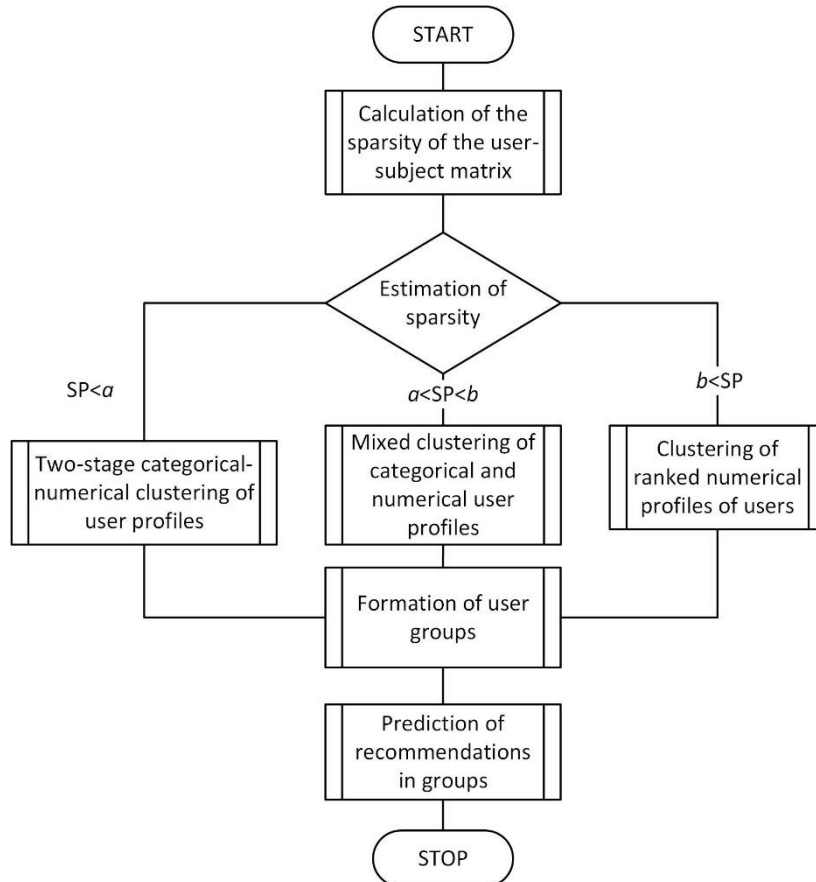
$$f(x) = \begin{cases} 1, & x = d_{ij} - d_c \leq 0 \\ 0, & x = d_{ij} - d_c > 0 \end{cases} \quad (5)$$

Search results for user groups depend on the sparsity of the user-item matrix. The sparseness of the user-subject matrix is calculated by formula (6):

$$SP = \frac{N_{Users}}{N_{Items}}. \quad (6)$$

where  $SP$  is the number of non-zero elements of the user-subject matrix;  $N_{Users}$  – number of system users;  $N_{Items}$  – the number of items in the system.

Sparsity of the user-item matrix is used in the hybrid method of searching for user groups [40]. The structural diagram of the method is presented in Figure 2.



**Figure 2:** Block diagram of a hybrid clustering method for finding groups of similar users

The hybrid clustering method for finding user groups involves the following methods: a modified method of non-hierarchical clustering of numerical vectors of user profiles, which is based on the  $k$  means method; method of mixed clustering of categorical and numerical vectors of user profiles; two-stage method of categorical-numerical clustering. The selection of the method is carried out by estimating the sparsity of the user-subject matrix and two parameters  $a$  and  $b$  (upper and lower sparsity threshold values). For low sparseness, a modified method of non-hierarchical clustering of numerical vectors of user profiles is used, which is based on the  $k$  means method. At an average value of sparsity, the method of mixed clustering of categorical and numerical vectors of user profiles is used. For high sparseness, a two-stage method of categorical-numerical clustering is used.

At the first stage, categorical clustering of vectors of demographic profiles of users is carried out and groups of users that are similar in their demographic characteristics are built. At the second stage, numerical clustering of user profile vectors, which contain numerical ratings of subjects, is carried out.

A modified ROCK method (A Robust Clustering Algorithm for Categorical Attributes) [42] is used for categorical clustering of vectors of demographic profiles of users. ROCK is an agglomerative hierarchical algorithm for clustering categorical attributes. The modified ROCK method requires less time for calculation and solves the problem of "false" clusters at the final stage of clustering.

## 2.4.2. Methods of Recommendations in the Computer Components Analysis System

In order to avoid certain limitations of "pure" recommendation systems and to minimize the problems created by these methods in providing recommendations, hybrid methods are used [43-47]. The idea is that a combination of algorithms will provide more accurate and efficient recommendations than a single algorithm, since the disadvantages of one algorithm can be overcome by another algorithm.

If the user independently submits the parameters of components (items), then we use the method of content filtering to generate a recommendation based on the database of leading manufacturers of components. To sort and position the product from the best to the worst, according to the given parameters, you need to create a set of links of all brands, types and characteristics of computer components for the last ten years. The databases of mass developers of computer components Intel, Nvidia, AMD and Kingston will help us in this. All of these developers have a larger market share than their competitors, so they have more sales and more accurate statistics. The set will consist of about 50 types of computer parts of each type [3].

In the case when the user needs to analyze the available components, we use the method of collaborative filtering to generate a set of recommendations. The collaborative filtering model has been adopted and used as a recommendation model more often than content-based filtering. However, despite the development of collaborative filtering, the scalability problem and the sparsity problem [36] have not been solved, so there is a limitation that the accuracy of the recommendation is reduced.

Therefore, the weighted hybrid method calculates the prediction score as the results of all the recommendation approaches, treating them as variables in a linear combination. Suppose that there are  $k$  recommendation approaches to be combined using a weighted strategy, the prediction score of user  $m$  to subject  $i$  can be calculated as:

$$p_{m,i} = \sum_f^k \sigma_f p_{m,i}^{(f)}, \quad (7)$$

where  $\sigma_f$  – is the weight of the algorithm  $p_{m,i}$ .

Since two recommendation approaches will be combined, then  $k = 2$ . Then we get:

$$p_{m,i} = \sigma_1 \times p_{m,i}^{(1)} + (1 - \sigma_1) \times p_{m,i}^{(2)} \quad (8)$$

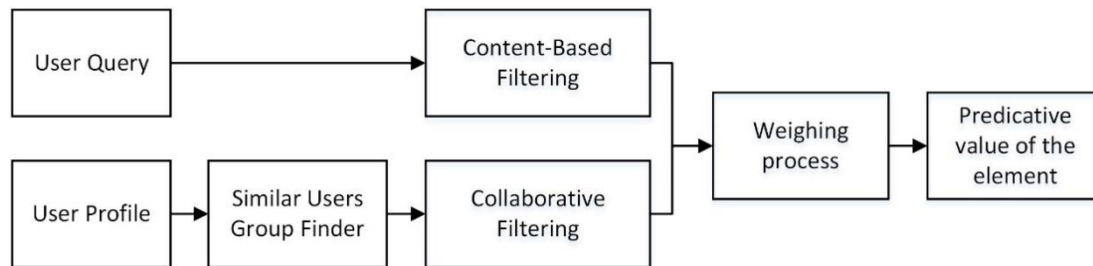
And the optimized weight can be obtained using calculations [47]:

$$\sigma_1 = \frac{\sum_m \sum_i (p_{m,i} - p_{m,i}^{(2)}) (p_{m,i}^{(1)} - p_{m,i}^{(2)})}{\sum_m \sum_i (p_{m,i}^{(1)} - p_{m,i}^{(2)})^2}. \quad (9)$$



The hybrid approach is applied in the computer component analysis system at the step of forming the final set of recommendations (Figure 3). Here, the results of different recommendations are combined by integrating the estimates of each of the methods used according to the linear formula (8).

Initially, recommendations from collaborative filtering and content filtering are given equal weight. As the predictions are confirmed or denied, the weights are adjusted [47]. The advantage of the weighted hybrid is that all the strengths of the recommendation system are used during the process of providing them in a simple way.



**Figure 3:** Weighted hybrid recommendation method

## 2.5. Conceptual model of the recommendation system

The proposed methods and models will be taken into account when developing a conceptual model of the computer components analysis system. The conceptual model of the information system provides identification of its various entities and their possible interaction [48-52]. When developing the conceptual model of the system, UML (Unified Modeling Language) diagrams are built, which are designed to simplify the understanding of the information system project being created [53]. Diagrams increase project support and facilitate documentation development.

System development involves modeling a set of requirements for its operation, namely: business, functional, non-functional and user requirements. The result of creating requirements is a UML-diagram of use cases (Use Case Diagram) for displaying the structural diagram of the behavior of functional and user requirements, which is presented in Figure 34. It is determined that there are three roles in the system. Roles "User" and "Guest", which are related by the generalization relationship, and "Administrator".

"User" is a person who has successfully logged into the system. "User" has access to the entire system and can use all the presented system functionality. There are associative links between the options of use and the "User". Unlike the "User", the "Guest" has the right only to get acquainted with the functions of the system. However, after authorization, the "Guest" receives all the rights of the "User".

The "user" has the opportunity to create a profile – based on the received data, initial recommendations will be made. Obviously, the "User" gets a certain "freedom" in his actions. He can use the analyzer of his own computer, which will detect outdated components and offer better quality and, most importantly, the most suitable parts. If the user wants a completely new computer, then he can use the assembly designer, which selects components according to the user's request, or an already selected part of the computer. Next, the system either shows the availability of the selected part at the points of sale closest to the user, or offers to assemble yourself or send to you the entire assembly.

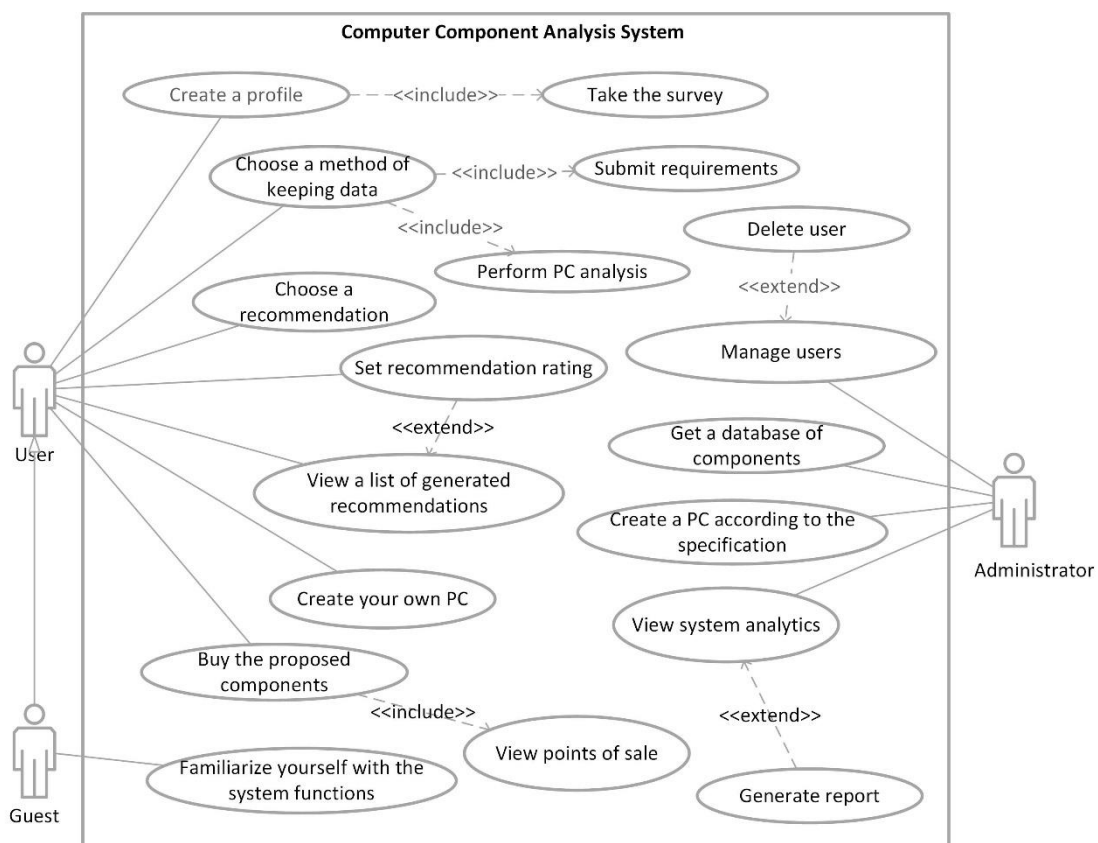
If the method of obtaining the parameters is automatic, then first a request is sent to the database of the personal computer to obtain data about the components and their characteristics. The formed list is the input data for generating recommendations.

If the system detects new modern components, the work of which is maximally productive and does not require any replacement, the system will notify about this immediately after analyzing the personal computer.

Accordingly, the following usage options were formed:

- Create a profile;

- Pass the survey;
- Choose a method of receiving data;
- Send requirements;
- Perform PC analysis;
- Choose a recommendation;
- View the list of generated recommendations;
- Set recommendation rating;
- Create your own PC;
- Buy the proposed components;
- View points of sale;
- Get to know the system functions.



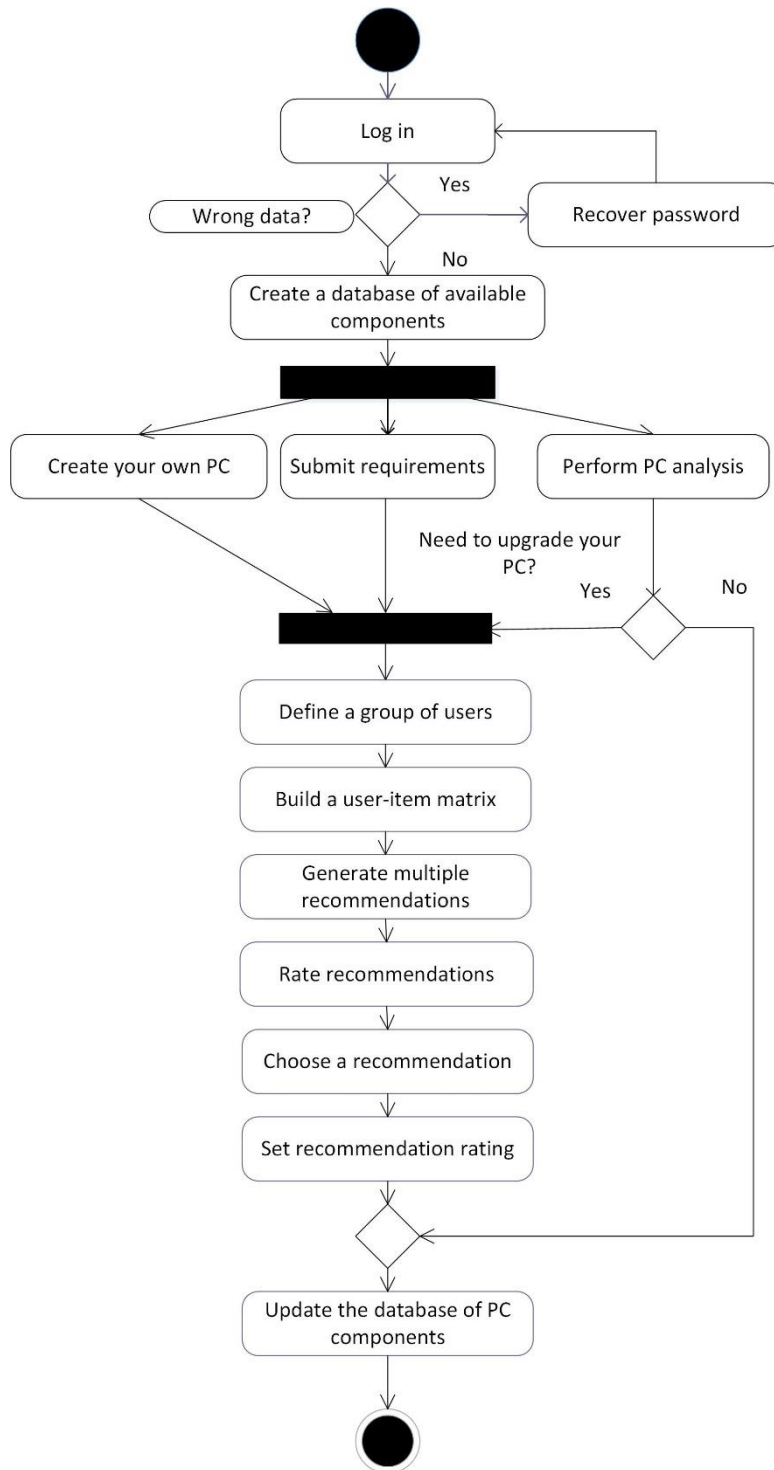
**Figure 4:** Use case diagram

The dynamic behavior of the system can be described with the help of an Activity Diagram (Figure 5).

As soon as the user successfully enters the system, the following actions will be available to him:

- “Create your own PC” – the user can use the assembly designer, which, based on the recommendation system, accurately selects the components to the given request of the user, or an already selected part of the computer. Next, the system either shows the availability of the selected part at the points of sale closest to the user, or offers to assemble yourself or send to you the entire assembly.
- “Send requirements” – the user specifies a list of components and their characteristics, which are input data for generating recommendations based on the database of components of leading manufacturers.
- “Perform PC analysis” – the method of obtaining parameters is automatic, the recommendation system first sends a request to the personal computer database, from where all the necessary data, in the form of components and their characteristics, is sent back, and

then the recommendation system sends a request to the software database provisioning, searching, filtering and sorting is done there, sent back and then sent to the user.



**Figure 5:** Activity diagram

- “Define user group” – it is better to generate recommendations for a user group, so we use clustering methods to search for user groups.
- “Build a user-subject matrix” – an up-to-date "user-subject" matrix is required for the functioning of recommendation algorithms.

- “Generate a set of recommendations” – depending on the received parameters, a set of recommended computer components is generated.
- “Rate a recommendation” – the user can rate any recommendation, regardless of whether he is in the process of implementing it, has already completed it, or has never implemented it.
- “Choose a recommendation” – with this action, the user informs the system that he is ready to perform the recommended update of computer components. The system, in turn, provides the user with the necessary content and feedback related to the selected recommendation.
- “Set recommendation rating” – a list of recommendations that can be marked with a high level of popularity among users during a fixed period of time. This category includes recommendations with the highest ratings, with the highest concentration of user activity in a given time period, that is, those where the system has recorded a lot of interest from a variety of users. Any recommendation can be rated by the user, regardless of whether he is in the process of implementing it, has already completed it, or has never implemented it. This will speed up the process of cutting off irrelevant recommendations to the user.
- “Update database of PC components” – after the user completes the update of components, the system will record this in the system directory of the computer.

**State diagram.** Figure 6 shows the state diagram for the designed system. The system behavior model is illustrated from the point of view of the main functioning process, which consists in the formation and provision of recommendations for the user.

A state diagram is used to model the dynamic aspects of the system. Behavior is considered from the point of view of the sequence of states through which a certain modeled element passes during its existence in response to some events from outside or inside the system. A state is a combination of certain conditions that satisfy some period of life of a system element while it performs a certain activity or waits for a certain event.

The process involves users who have successfully logged in with their account. When creating a user profile, a person provides general information about himself. After that, the mandatory step is to complete the questionnaire. The purpose of these steps is to gather some initial information about the user, as well as to start the process of forming a user portrait.

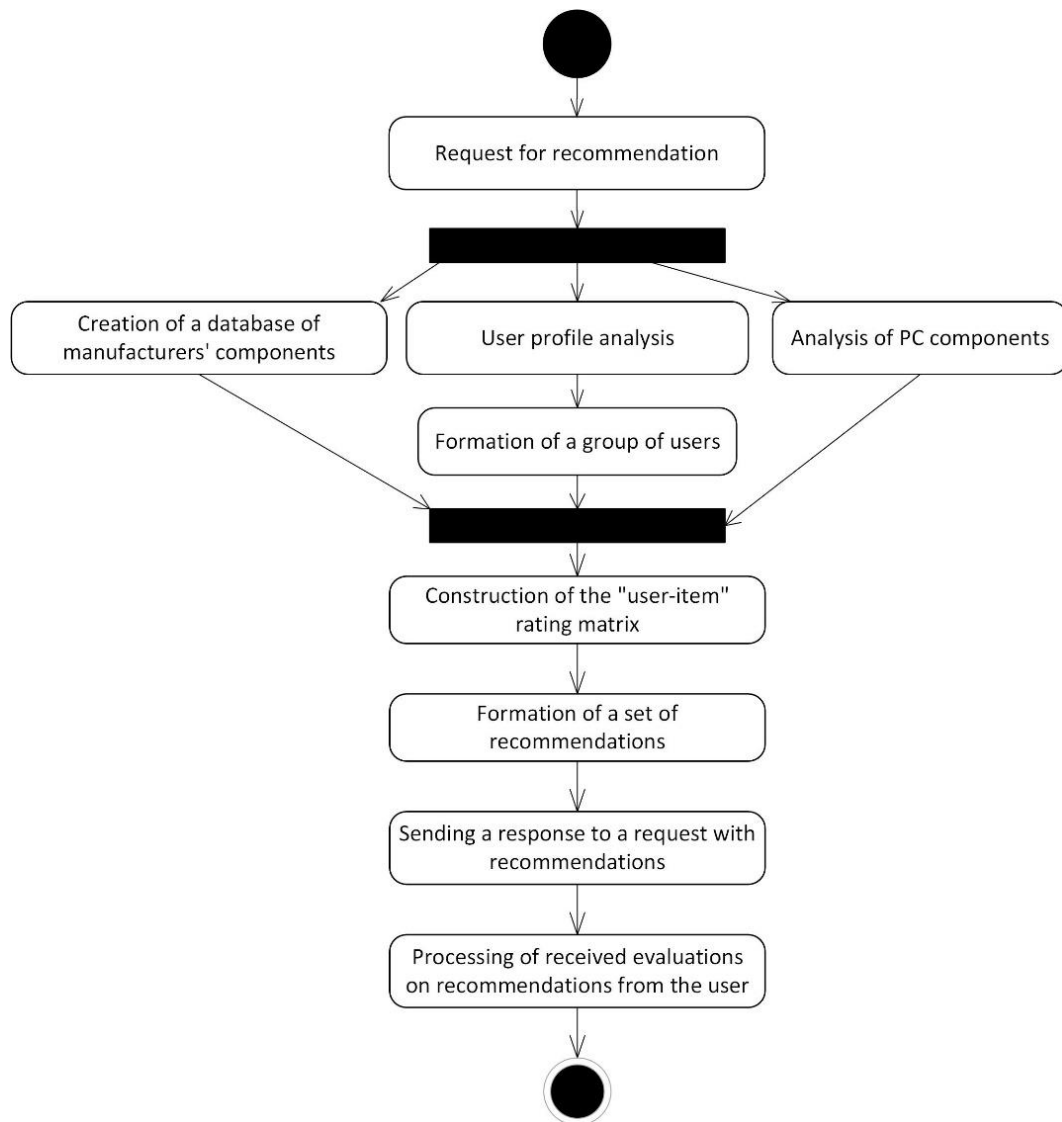
The recommendation system needs to have as much information as possible about the user in order to provide informed recommendations. During the user's first interaction with the system, the recommendation system will read data from the computer, and also take into account the parameters specified by the user himself. This will allow the recommendation system to obtain basic data to initiate computer construction or parts selection.

The use of this technique in the recommendation system will make it possible to more fully understand the contents of the computer, and it is possible to predict the further preferences of other users.

In order to receive a recommendation for computer components, the user needs to send a corresponding request. The system starts processing the request by analyzing the chosen way of obtaining parameters – automatic scanning from the user's personal computer or set by the user himself. If the user independently submits the parameters of components, the system creates a database of components from leading manufacturers. To sort and position the product from the best to the worst, according to the given parameters, you need to create a set of links of all brands, types and characteristics of computer components in recent years. Based on the user profile, we determine the group of users for whom the recommendation will be generated. For the methods of generating recommendations to work, it is necessary to build a matrix of evaluations (ratings) of the user-subject.

Then the recommendation methods search for similar elements (meaning the similarity of users and certain activity). What is important is that the user should receive new recommendations, which would not be repeated with already completed ones or those with low ratings. Then the results are formed in the form of a recommendation and sent by the system to the user.

It is necessary to define the main classes of the developed system, with their methods and attributes (Figure 7).



**Figure 6:** State diagram

The Computer Component Analysis System will be the main class that will start the program and will have the following attributes: System Interface and Server. We will use the Initialization method to show the actions of the user interface.

The interface of the system will be a program with attributes: Server, “Analyzer” button, “Build” button, where the Server itself will process button presses, and the buttons, in turn, will be responsible for the method of obtaining the component. Two methods: “Analyzer” and “Build” button click processing.

The server is the main object of the class that starts and manages subprocesses. The attributes are as follows: Analyzer – sending a request to the database; Build – returning the result from the database; Recommendation system – receives, processes and returns the name of the object; Return. The server has such class methods as Process the submitted request – starts the process of the recommendation system; Handling the interaction process between the recommendation system and the database; Create a list – recommendation components; Send request – send a ready report to the user. The “Analyzer” is an interface of a “smart system” with the method “Make an analysis”. “Construction” – the interface for searching and building data from the database using the “Search in the database” method. The Attribute recommendation system contains a list of custom settings for the system, and has a “sampleSort” method that checks and produces a result. The recommendation system with the same attribute as parameter list and a “ShowReport” method that returns analytics as text. The “Return” has a Subprocess attribute that is used to output the results. Computer Component Analysis System and Interface, Interface and Server, Server and Analysis, Server and Construction, Server and

Recommendation System, Server and Return are the main relationships between classes. The type of all relations is aggregation. Modeling system objects in the form of UML diagrams defined the main classes of the system, their attributes and methods, which are presented in the class diagram.

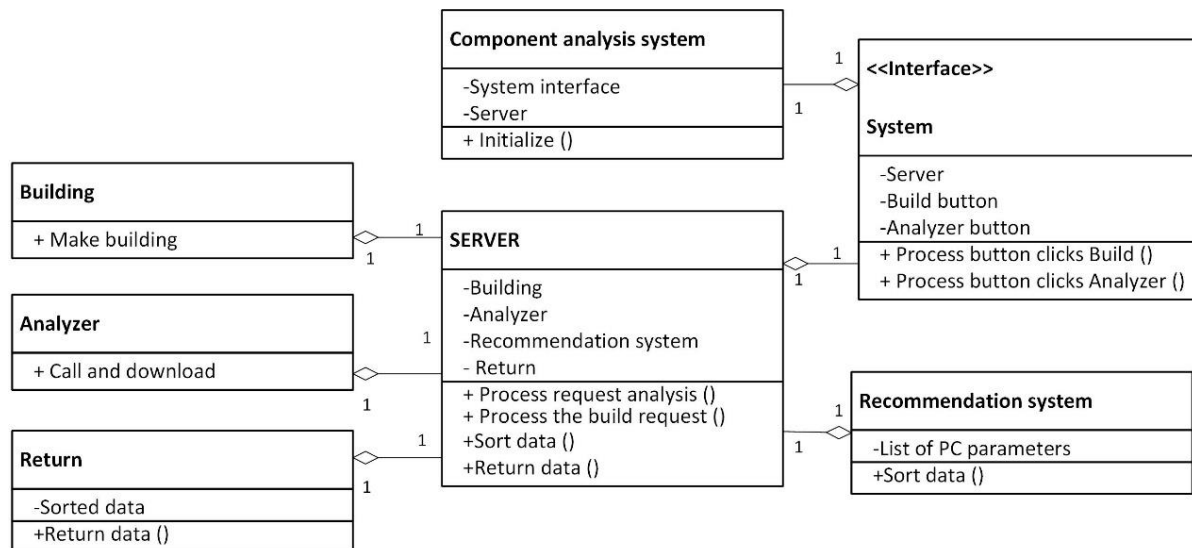


Figure 7: Class diagram

## 2.6. Means for the system prototype implementation

In order to facilitate automatic analysis and quick access to the software, it was decided to create the first version as a "desktop" application.

The development of both the user interface and the functional part will be more successful using C#. After all, according to the solution of the task - we need to create a "desktop" application with a recommendation system, for this we will need the .NET Framework environment. As for .NET frameworks and libraries, they are officially designed and assigned to work on Windows operating systems. Therefore, they will show the most effective performance and quality of work during use.

The CoreCLR library will help periodically free memory by removing objects from it that are no longer needed, thanks to a built-in "garbage collector".

A large number of libraries, frameworks, applications for machine learning and the creation of small artificial neural networks allow to solve most of the needs for creating such programs. And since this program is developed by Microsoft, its libraries will work perfectly with the direct libraries of a personal computer, which is what our program needs for productive work. CoreFX is a library that will make it possible to seamlessly connect one database to another, in particular thanks to the System.IO component and System.Collections.

The MySQL database management system was chosen to work with the database. Data processing is carried out using C# tools. The dataset is generated from the Core libraries by importing files created from MySQL into the application using the settings and the Directory and Entity libraries.

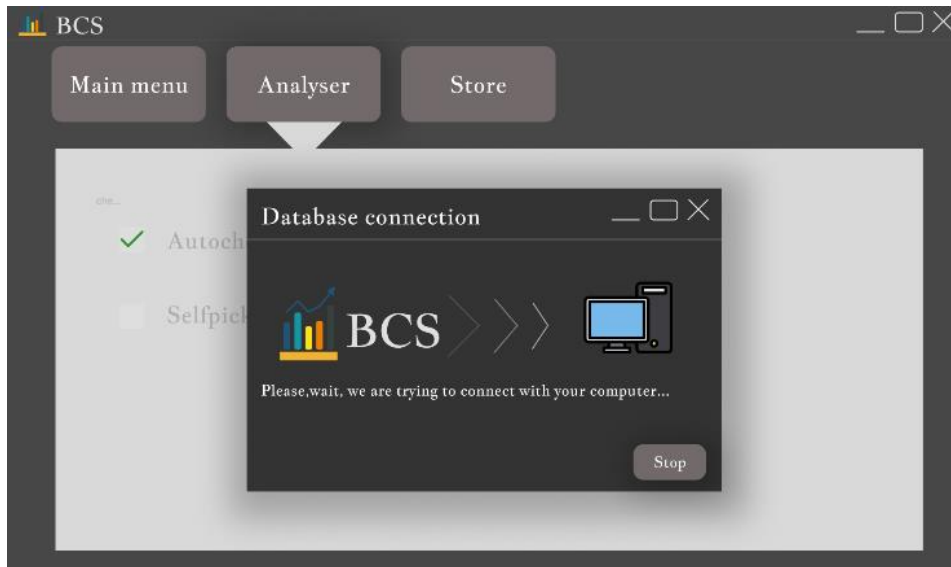
## 3. Analysis of the obtained results

Software testing is one of the important processes of technical research, which is designed to determine the quality of the product in relation to the area in which it is to be used. Also, testing includes the process of finding errors, bugs or other defects, and basic testing of software tools for the purpose of evaluation.

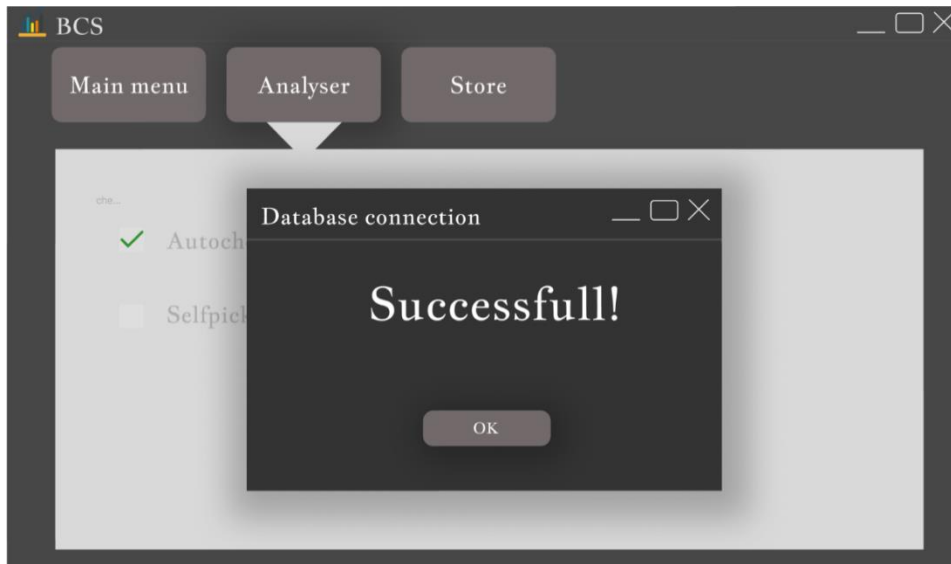
Since the number of test attempts even for simple software components is innumerable, the essence of testing is to conduct possible tests based on the availability of time and resources. There will be only

two types of testing: functional, by checking the work of the main driver - the recommendation system, and user interface testing, which is usually performed simply while using the program.

First, we will test the main functional component of the recommendation system. With the given parameters, you can test how the system will filter the data. Figure 8 shows the process of loading and connecting the MySQL database with the C# libraries, or in the case of automatic analysis, the user simply authorizes the program to work with his personal computer.



**Figure 8:** The process of interaction between the program and the PC



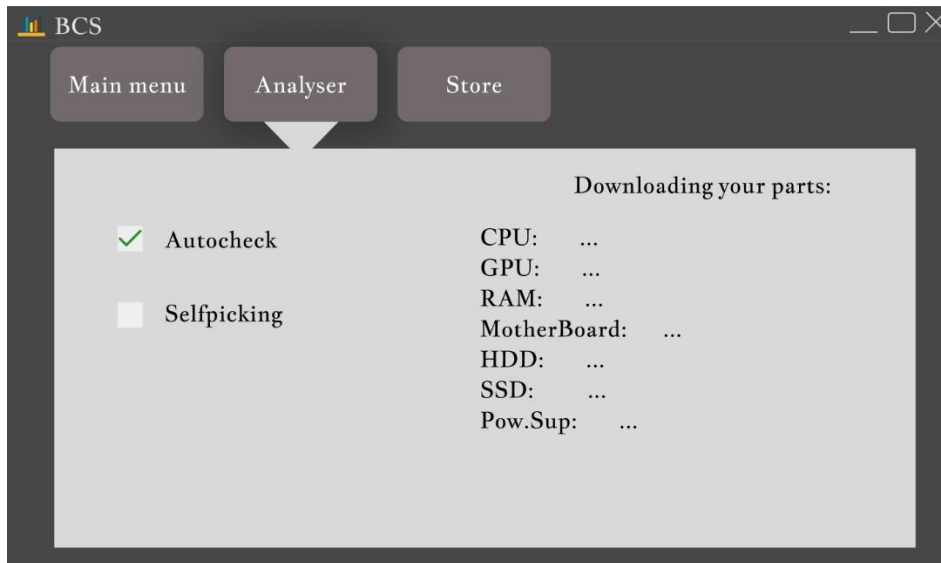
**Figure 9:** Confirmation of interaction

In Figure 9 the confirmation of the achievement of the interaction of the software and the computer, or the user parameters, is presented.

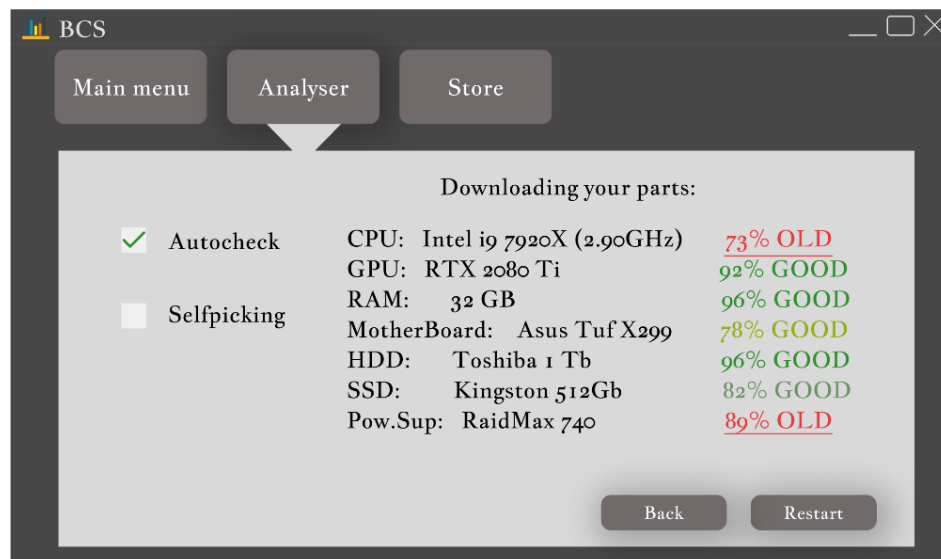
The next step is to process the received data and send a request by the user to receive appropriate recommendations for available computer components, followed by the implementation of this request by the recommendation system. This action is presented in Figure 10.

In Figure 11 the final result is shown, i.e. the received list with outdated and new components.

According to the obtained result, the recommendation system determined that the central processor is outdated, and shows that with a probability of 73% it should be replaced with a new one.



**Figure 10:** Implementation of the request



**Figure 11:** Results

The same situation with the power supply unit, 89% notes that it needs to be replaced urgently because it can lead to the failure of all other components because they are all connected to the power supply unit. Taking everything into account, it can be understood that the system determines and analyzes the given requests quite well.

After conducting functional testing, not many errors or bugs were found,

Let's move on to the analysis of the user interface or UI.

The entire user interface built using C# and .NET. Open the application and get to the main menu shown in Figure 12.

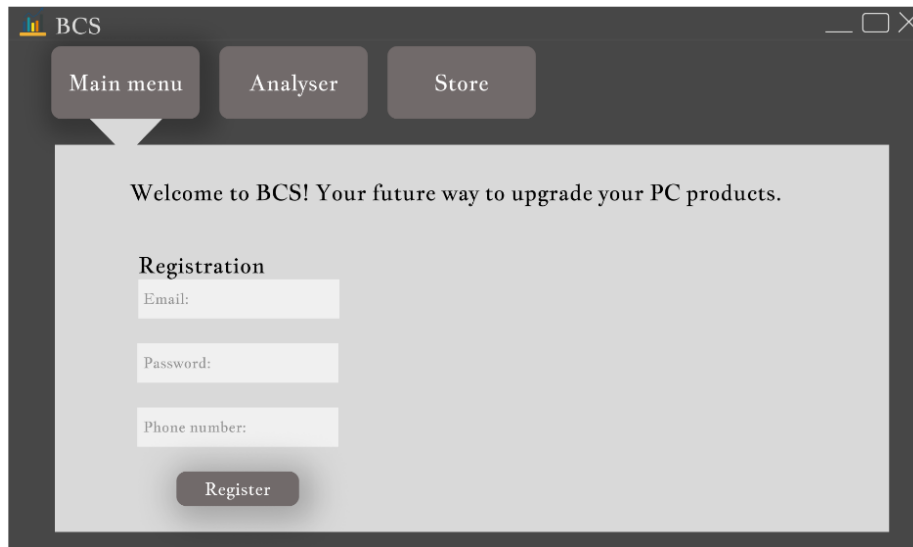
Next, it is necessary to carry out authentication and confirmation, which is sent to the mail to the user (Figure 13).

After that, the program offers us to go to the "Computer construction" section and analyze the user's personal computer, or if desired, enter the parameters manually (Figure 14).

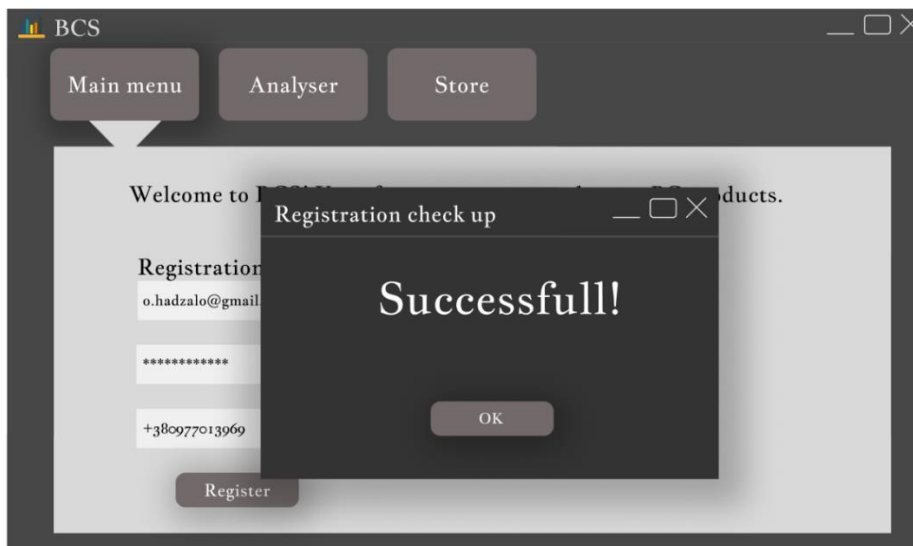
After that, the recommendation system receives the request, processes it and presents it in the form of an analytical list, which is the final process.

As a result of the review of the user interface, no defects or any serious errors were found.





**Figure 12:** Main menu



**Figure 13:** Registration process

The process of creating parameters and functions for a recommendation system is quite lengthy, as the system has to process input and output data. Figure 15 shows the process of automatic processing of parameters that the system downloads from a personal computer.

As a result, after comparing the result of the system and the existing parameterization of this personal computer, the system recognized 7 out of 8 components, which in terms of percentage of accuracy will be 87.5%, which is shown in Figure 15.

The global goal is achieved because it involves creating a simple user interface for computer users. Taking into account that a temporary interface is provided, a recommendation system is created, the goal can be considered fulfilled within the first version of the project.

The general purpose of providing users with sorted data, either automatically or upon their request, is fully achieved with the help of a recommendation system.

At the stage of the first version of the software, the implementation of the project, that is, the deployment, is not difficult. The system consists of an interface and a driving part, which is written using the C# programming language. A database represented by MySQL must also be involved.

To combine all these components into one functioning mechanism, Core libraries for C# and internal .NET libraries are used.

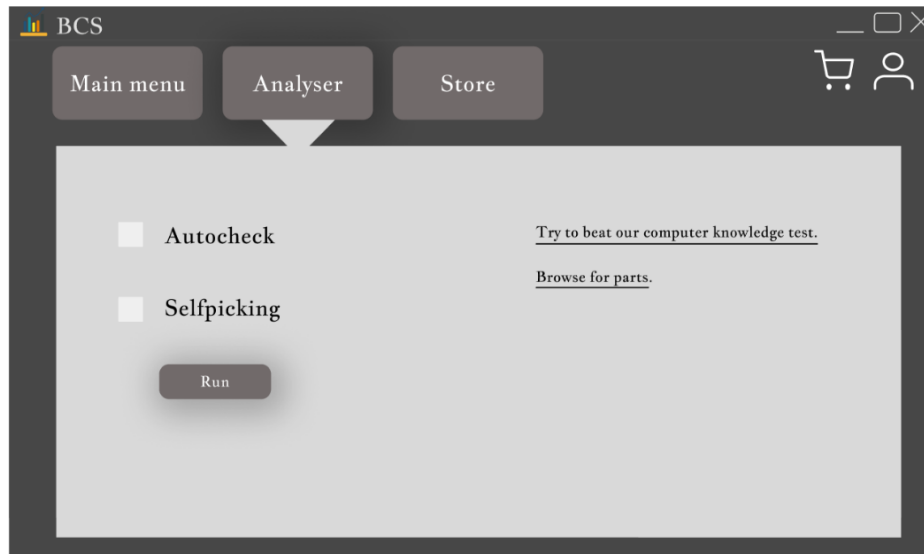


Figure 14: "Analyser" menu

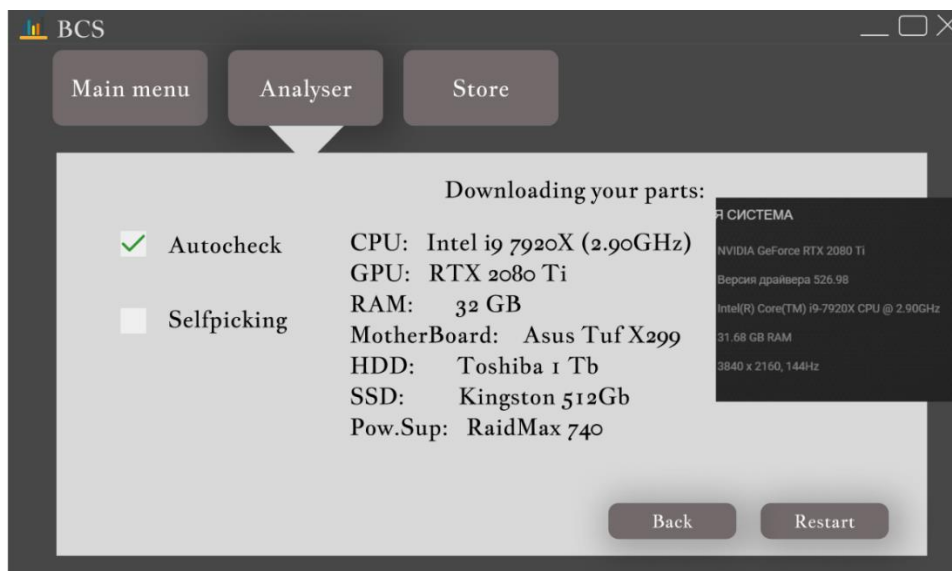


Figure 15: Automatic parameter processing

To access the application, you need to download BestCompabilitySystem to your PC. The deployment of the application will take place when the main menu of the program is loaded. From there, the user can navigate within their interest and search for a solution to the problem.

Access to the developed project is provided only if the application is downloaded. You also need to go through the authentication system and connect to the computer database. So far, the program implements the analysis according to the specified parameters and the parameters that the project independently determined when contacting the database.

#### 4. Conclusions

To develop an information system for the analysis of computer components, the main methodologies and solutions for the analysis of computer components were analyzed. In order to focus on the relevance and competitiveness of the developed product, a market analysis was conducted and four applications with a similar solution to the problem were identified. Methods and technologies for generating a set of recommendations for the end user were also investigated. It has been determined that providing

recommendations to groups of users is more appropriate than providing recommendations to individual users. The mixed categorical-numerical method was used to search for groups of users taking into account demographic characteristics. A hybrid method of searching for user groups has been developed, which includes the method of numerical non-hierarchical clustering, the method of mixed categorical-numerical clustering, and the two-stage clustering method. The two-stage method performs categorical clustering at the first stage, numerical clustering at the second stage. Taking into account the research results, the optimal type of information system was chosen for the implementation of the proposed solution, namely: a hybrid recommendation system. The system uses a weighted hybrid mechanism to provide recommendations, which allows for more accurate recommendations.

The work of the recommendation system is aimed at analyzing the main active components and generating various best options of computer components to solve the user's problem based on the received personal data of the user and computer settings. The recommendation system will take care of the most effective combination of components that will be offered to the user.

Two types of software product testing were conducted: interface and functionality. As a result, both types of testing did not reveal any errors or bugs. Statistical data were shown that the recommendation system correctly finds the type and parameters of components on a given personal computer with an accuracy of 87.5%.

Further research will be aimed at the complete implementation and testing of the Desktop prototype of the system, as well as the development of a suitable version of the Web application and possibly also a mobile version.

## 5. References

- [1] P. Melville, V. Sindhvani, Recommender Systems, In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning and Data Mining. Springer, Boston, MA, 2017. doi: 10.1007/978-1-4899-7687-1\_964.
- [2] V. A. Pavlysh, L. K. Glinenko, N. B. Shakhovska, Fundamentals of information technologies and systems: a textbook, Lviv, 2018.
- [3] Best benchmarks software of 2023, 2023. URL: <https://www.techradar.com/best/best-benchmarks-software>.
- [4] M. Jalili, S. Ahmadian, M. Izadi, P. Moradi, M. Salehi, Evaluating collaborative filtering recommender algorithms: a survey, IEEE 6 (2018) 74003-74024. doi: 10.1109/ACCESS.2018.2883742.
- [5] D. Bollen, B. P. Knijnenburg, M. C. Willemsen, M. Graus, Understanding choice overload in recommender systems. In Proceedings of the fourth ACM conference on Recommender systems (2010) 63-70. doi: 10.1145/1864708.1864724
- [6] O. Veres, P. Ilchuk, O. Kots, Y. Levus, Information System for Leisure Time-Management in Quarantine Conditions, International Scientific and Technical Conference on Computer Sciences and Information Technologies (2022) 156-159.
- [7] O. Veres, P. Ilchuk, O. Kots, Y. Levus, O. Vlasenko, Recommendation System for Leisure Time-Management in Quarantine Conditions, CEUR Workshop Proceedingsthis Vol-3312 (2022) 263-282.
- [8] O. Veres, P. Ilchuk, O. Kots, Intelligent information system for remote customer service, CEUR Workshop Proceedingsthis Vol-2917 (2021) 283-299.
- [9] O. Bulut, D. C. Cormier, J. Shin, An intelligent recommender system for personalized test administration scheduling with computerized formative assessments, Front. Educ. 5:572612 (2020). doi: 10.3389/educ.2020.572612.
- [10] K. Falk, Practical recommender systems, Simon and Schuster, 2019.
- [11] A. Beheshti, S. Yakhchi, S. Mousaeirad, S. M. Ghafari, S. R. Goluguri, M. A. Edrisi, Towards cognitive recommender systems, Algorithms 13(8) (2020). doi: 10.3390/a13080176.
- [12] M. Atas, A. Felfernig, S. Polat-Erdeniz, A. Popescu, T. N. T. Tran, M. Uta, Towards psychology-aware preference construction in recommender systems: Overview and research issues, Journal of Intelligent Information Systems 57 (2021) 467-489. doi: 10.1007/s10844-021-00674-5.

- [13] Z. Abbasi-Moud, H. Vahdat-Nejad, J. Sadri, Tourism recommendation system based on semantic clustering and sentiment analysis, *Expert Systems with Applications* 167 (2021) 114324. doi: 10.1016/j.eswa.2020.114324.
- [14] F. Amato, V. Moscato, A. Picariello, F. Piccialli, SOS: a multimedia recommender system for online social networks, *Future generation computer systems* 93 (2019) 914-923. doi: 10.1016/j.future.2017.04.028.
- [15] S. Loeb, D. Terry, Information filtering, *Communications of the ACM* 35(12) (1992) 26-28. doi: 10.1145/138859.138860.
- [16] I. Cantador, M. Fernández, D. Vallet, P. Castells, J. Picault, M. Ribiere, A multi-purpose ontology-based approach for personalised content filtering and retrieval, *Advances in Semantic Media Adaptation and Personalization* (2008) 25-51.
- [17] J. L. Herlocker, J. A. Konstan, L. G. Terveen, J. Riedl, Evaluating collaborative filtering recommender systems, *ACM Trans. Inf. Syst.* 22 (2004) 5-53.
- [18] G. Geetha, M. Safa, C. Fancy, D. Saranya, A hybrid approach using collaborative filtering and content based filtering for recommender system, *Journal of Physics: Conf. 2018, April, Series 1000* 012101 (2018). doi:10.1088/1742-6596/1000/1/012101.
- [19] J. Salter, N. Antonopoulos, Cinema Screen recommender agent: combining collaborative and content-based filtering. *IEEE Intelligent Systems* 21(1) (2006) 35-41. doi: 10.1109/MIS.2006.4.
- [20] B. Li, G. Li, J. Xu, X. Li, X. Liu, M. Wang, J. Lv, A personalized recommendation framework based on MOOC system integrating deep learning and big data, *Computers and Electrical Engineering* 106 (2023) 108571. doi: 10.1016/j.compeleceng.2022.108571.
- [21] H. H. Lathabai, A. Nandy, V. K. Singh, Institutional collaboration recommendation: An expertise-based framework using NLP and network analysis, *Expert Systems with Applications* 209 (2022) 118317. doi: 10.1016/j.eswa.2022.118317.
- [22] K. Shivaram, P. Liu, M. Shapiro, M. Bilgic, A.Culotta, Reducing Cross-Topic Political Homogenization in Content-Based News Recommendation, in: *Proceedings of the 16th ACM Conference on Recommender Systems* (2022) 220-228. doi: 10.1145/3523227.3546782.
- [23] D. Goldberg, D. Nichols, B. M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry. *Communications of the ACM* 199235(12) (1992) 61-70. doi: 10.1145/138859.138867.
- [24] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, Grouplens: An open architecture for collaborative filtering of netnews, in: *Proceedings of the 1994 ACM conference on Computer supported cooperative work* (1994): 175-186. doi: 10.1145/192844.192905.
- [25] S. H. Park, S. P. Han, Empirical analysis of the impact of product diversity on long-term performance of recommender systems, in: *Proceedings of the 14th Annual International Conference on Electronic Commerce* (2012) 280-281. doi: 10.1145/2346536.2346592.
- [26] M. Nilashi, O. Ibrahim, K. Bagherifard, A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques, *Expert Systems with Applications* 92 (2018) 507-520. doi: 10.1016/j.eswa.2017.09.058.
- [27] A. Abdelwahab, H. Sekiya, I. Matsuba, Y. Horiuchi, S. Kuroiwa, Alleviating the sparsity problem of collaborative filtering using an efficient iterative clustered prediction technique, *International Journal of Information Technology & Decision Making* 11(01) (2012) 33-53. doi: 10.1142/S0219622012500022.
- [28] G. B. Martins, J. P. Papa, H. Adeli, Deep learning techniques for recommender systems based on collaborative filtering, *Expert Systems* 37(6) (2020) e12647. doi: 10.1111/exsy.12647.
- [29] R. Alabdulrahman, H. Viktor, Catering for unique tastes: Targeting grey-sheep users recommender systems through one-class machine learning, *Expert Systems with Applications* 166 (2021) 114061. doi: 10.1016/j.eswa.2020.114061.
- [30] A. Ansari, S. Essegai, R. Kohli, Internet recommendation systems (2000). doi: 10.1509/jmkr.37.3.363.18779.
- [31] J. Basilico, T. Hofmann, Unifying collaborative and content-based filtering, in: *Proceedings of the twenty-first international conference on Machine learning* (2004). doi: 10.1145/1015330.1015394.
- [32] E. Çano, M. Morisio, Hybrid recommender systems: A systematic literature review, *Intelligent Data Analysis* 21(6) (2017) 1487-1524. doi: 10.3233/IDA-163209.
- [33] CPU-Z System information software. URL: <https://www.cpuid.com/softwares/cpu-z.html>.

- [34] GPU-Z Main settings and features. URL: <https://www.techpowerup.com/gpuz/>.
- [35] AIDA64 Extreme settings. URL: <https://www.aida64.com/products/aida64-extreme>.
- [36] Speccy main projects and opportunities, 2023. URL: <https://www.ccleaner.com/speccy>.
- [37] O. Veres, P. Ilchuk, O. Kots, Data Science Methods in Project Financing Involvement, in: International Scientific and Technical Conference on Computer Sciences and Information Technologies 2 (2021) 411-414. doi: 10.1109/CSIT52700.2021.9648679.
- [38] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, Knowledge-based systems 46 (2013) 109-132. doi: 10.1016/j.knsys.2013.03.012.
- [39] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE transactions on knowledge and data engineering 17(6) (2005) 734-749. doi: 10.1109/TKDE.2005.99.
- [40] M. V. Lobur, M. E. Schwartz, Y. V. Stech, Models and methods of forecasting recommendations for collaborative recommender systems, Bulletin of the Lviv Polytechnic National University. Series: Information systems and networks 901 (2018) 68-75. URL: <https://science.lpnu.ua/sites/default/files/journal-paper/2019/feb/15581/181912maket-68-75.pdf>.
- [41] J. Masthoff, Group modeling: Selecting a sequence of television items to suit a group of viewers, Personalized Digital Television: Targeting Programs to individual Viewers (2004) 93-141. doi: 10.1007/1-4020-2164-X\_5.
- [42] S. Guha, R. Rastogi, K. Shim, ROCK: A robust clustering algorithm for categorical attributes. Information systems 25(5) (2000) 345-366. doi:10.1016/S0306-4379(00)00022-3.
- [43] R. Burke, Hybrid web recommender systems, The adaptive web (2007) 377-408. URL: [https://link.springer.com/chapter/10.1007/978-3-540-72079-9\\_12](https://link.springer.com/chapter/10.1007/978-3-540-72079-9_12).
- [44] H. Ko, S. Lee, Y. Park, A. Choi, A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields, Electronics 11(1) 141 (2022). doi: 10.3390/electronics11010141.
- [45] U. Javed, K. Shaukat, I. A. Hameed, F. Iqbal, T. M. Alam, S. Luo, A review of content-based and context-based recommendation systems, International Journal of Emerging Technologies in Learning (iJET) 16(3) (2021) 274-306. URL: <https://www.learntechlib.org/p/219036/>.
- [46] G. Nápoles, I. Grau, Y. Salgueiro, Recommender system using Long-term Cognitive Networks, Knowledge-Based Systems 206 (2020) 106372. doi: 10.1016/j.knsys.2020.106372.
- [47] W. Lin, Y. Li, S. Feng, Y. Wang, The optimization of weights in weighted hybrid recommendation algorithm, in: 13th International Conference on Computer and Information Science (ICIS) (2014, June) 415-418. doi: 10.1109/ICIS.2014.6912169.
- [48] J. Johnson, GUI bloopers 2.0: common user interface design don'ts and dos, Elsevier, 2007.
- [49] A. Parush, Conceptual design for interactive systems: designing for performance and user experience, Morgan Kaufmann, 2015.
- [50] E. de Schipper, R. Feskens, J. Keuning, Personalized and Automated Feedback in Summative Assessment Using Recommender Systems, Frontiers in Education 6 (2021). doi: 10.3389/educ.2021.652070
- [51] O. Veres, N. Kunanets, V. Pasichnyk, N. Veretennikova, R. Korz, A. Leheza, Development and Operations-the Modern Paradigm of the Work of IT Project Teams, in: 14th International Conference on Computer Sciences and Information Technologies (CSIT) (2019) 103-106. doi: 10.1109/STC-CSIT.2019.8929861.
- [52] L. Halkiv, I. Kulyniak, N. Shevchuk, L. Kucher, T. Horbenko, Information Support of Enterprise Management: Diagnostics of Crisis Situations, in: Proceedings of the 11th International Conference on Advanced Computer Information Technologies, ACIT 2021 (2021) 309-312. doi: 10.1109/ACIT52158.2021.9548354.
- [53] OMG® Unified Modeling Language® (OMG UML®). URL: <https://www.omg.org/spec/UML/2.5.1/PDF>.