International Journal of Advances in Data and Information Systems
Vol. 6, No. 2, August 2025, pp. 204~215
ISSN: 2721-3056, DOI: 10.59395/ijadis.v6i2.1329

Enhancing Student Collaboration in Academic Projects Through a Content-Based Filtering Recommender System

Aldian Faizzul Anwar[1], Ririen Kusumawati[2], M. Ainul Yaqin[3], Irwan Budi Santoso[4]
[1,2,3]Magister Informatics Department, State Islamic University Maulana Malik Ibrahim Malang, Indonesia

Article Info

Article history:

Received Jun 25, 2024
Revised May 09, 2025
Accepted May 29, 2025

Keywords:

Content-Based Filtering
Recommender System
Project
Mean Average Precission

ABSTRACT

The Informatics Engineering Study Program at UIN Maulana Malik Ibrahim Malang facilitates students in developing their interests and talents through 10 academic communities that serve as forums for knowledge exchange and innovation in IT project development. However, a challenge arises in assigning suitable students to appropriate projects, resulting in many projects being completed by a limited set of students. To address this, a recommender system for academic project members was developed using the Content-Based Filtering method. This system assists project initiators in selecting competent team members based on students' prior experiences, considering the similarity between project requirements and student profiles. A dataset of 198 student-completed projects was used, with preprocessing, TF-IDF, and cosine similarity applied in the recommendation process. The system was implemented using the Flask framework with Python and HTML. Evaluation was conducted using the SUS method for usability (achieving a score of 79, categorized as excellent) and MAP for model performance across three scenarios. Scenario one (random community) scored 0.92, scenario two (same community) scored 0.79, and scenario three (comparison with actual members) scored 0.98. The results indicate that broader search scopes yield more accurate recommendations. This research contributes to the improvement of collaborative IT project in academic environments by enabling data-driven student member selection. The proposed system has the potential to be adopted by other academic institutions facing similar team formation challenges.

Corresponding Author:

Aldian Faizzul Anwar,
Magister Informatics Department,
State Islamic University Maulana Malik Ibrahim Malang,
Gajayana street No 50, lowokwaru, 65144, Malang
Email: 230605210010@student.uin-malang.ac.id

1. INTRODUCTION

According to the Minister of Manpower of the Republic of Indonesia, Ida Fauziyah, in 2023, the unemployment rate in Indonesia has reached 12%, with the dominance of bachelor's and diploma graduates [1]. He indicated that one of the causes was the mismatch between the degree and the available jobs, as well as the lack of work experience of the graduates [2]. Citing this, the role of universities is expected to be able to prepare their students to have abilities that can be used as capital in preparing for work, especially in this case informatics engineering students [3]. To prepare these skills, it is very lacking if a student only relies on academic abilities in lectures, especially students of the Informatics Engineering study program of UIN Maulana Malik Ibrahim Malang who are known as IT students of UIN Malang [4]. This can be a demand for every student

to continue to improve their hard skills, soft skills, and experience before entering the work environment [5].

The IT study program of UIN Malang itself has facilitated its students in developing their potential, interests and talents through several communities that accommodate its students in finding and developing their passion for IT [6]. In the IT study program of UIN Malang, there are 12 communities consisting of 10 academic communities and 2 non-academic communities, including MOCAP (Android), WEBBOENDER (Web), UINUX (Interface Design), MAMUD (Multimedia), ETH0 (Network), UINBUNTU (Operating System), FUN JAVA (Java Programming), GDSC (Google), DSE (Data Science), ONTAKI (Robotic), ALFATAA (Prayer), and ISC (Sports).

These communities are a forum for the exchange of knowledge, ideas, and collaboration between students or often referred to as cMOOC [7]. cMOOC education supports a wide range of knowledge and collaboration within the student academic community and provides benefits for students in improving skills and knowledge, expanding networks, and being able to share knowledge [8]. That way, it can create a collaborative and productive learning environment in order to produce innovative information technology application projects or in this case often known as *project-based learning* [9].

However, in practice, there are still problems in finding suitable project members. Many projects involve only the same group of students, so opportunities to collaborate and share experiences are uneven. To overcome this, a recommendation system is needed that is able to match students with projects based on their experience and competence. This recommendation system can assist project requesters in determining who are competent members based on student experience by analyzing and taking into account preferences and similarities between the suitability of project criteria and student experience.

Several previous studies have discussed recommendation systems using *content-based filtering* in various contexts, such as e-commerce and education [10]. However, there has not been much research that specifically implements this approach for the formation of academic project teams in the student community environment, especially based on student experience data in the community. This is the research *gap* that we want to fill in this research. One of the relevant studies was conducted by Raharjo et al. (2022) on a job and labor recommendation system, which uses *a content-based filtering* approach with the *Word2Vec algorithm* [11]. However, the research produced an *unsatisfactory Mean Average Precision* (MAP) accuracy.

In this study, the author will use the *content-based filtering* method by comparing and calculating the similarity of the content content from the UIN Malang IT student dataset obtained from the survey results with project criteria data inputted by the user. In addition, this study will use the TF-IDF approach or *Term Frequency-Inverse Document Frequency* and *Consine Similarity* to measure the proximity between texts which is expected to increase accuracy in producing recommendations for project members.

With this recommendation system, it is hoped that it can fulfill the purpose of this research, which is to provide opportunities and experience for IT students of UIN Malang before entering the world of work, increase the equal distribution of student participation in projects, and provide recommendations for the right project members for the success of the project [9].

2. RESEARCH METHOD

2.1 Data Collection

Data collection in this study began by looking for criteria from several reference journals to be used as criteria or *content* that would be presented in the form of questions at the time of data collection. Several criteria/questions were used in the data of this study.

Table 1. Question or Criteria for Data

Question/Criteria	Answer		Source
Programmer Personality	**Oppeness to experience**: Programmers who are creative, curious, and willing to explore new ideas. Be open to new ideas and think outside the box. **Extraversion**: Tendency to talk a lot, be friendly, assertive, and have high ambitions **Conscientiousness**: Ability to be organized, responsible, and diligent in work. **Neuroticism**: Programmers who have a tendency to negative emotions, such as anxiety and mood swings **Agreeableness**: Ability to be cooperative, considerate, and empathetic towards others.		Amin *et al*., (2020)
Community	- Webboender (Website) - Mocap (Mobile) - GDSC - UINUX (uiux) - Ontaki (Robotic)	- Fun Java (Java) - ETH0 (Network) - Uinbuntu (OS) - Mamud (Multimedia) - Data Science	Academic IT Community
Programming Language	- C++ - C# - CSS - HTML - PHP - Go - JavaScript	- Dart - Swift - Kotlin - Java - JSON - Phyton - SQL	Setiadi, (2022) dan Juviler, (2022)
Role Programmer	- Frontend - Backend - Fullstack - Network engineer - Security engineer - Cloud Engineer	- UIUX Designer - Data Analsyt - Team Lead - QA engineer - UIUX Designer	Russo & Stol, (2022)
Application	- Mail Archive System - Activity Archive System - Project Management - E-commerce - Company Website - Educational Applications - Educational games	- Financial Planning Apps - IoT smarthome - Educational Robots - Sports Robots - Graphic Design - Web Design - Mobile Design -Animation	Application Sales at PT. Ekata Technology
Project Description	Description of Projects that have been done		-

Furthermore, the questions collected from these sources will be used as a reference in the process of determining whether students meet the requirements of the desired project. The data collection process was carried out through a questionnaire distributed to IT students of UIN Malang. Examples of data to be generated as Table 2.

Table 2. Data Example

Criteria/Content	Data Input
Programmer Personality	Conscientiousness
Community	GDSC
Language Programming	PHP HTML CSS
Role	Backend
Project	Company Website
Project Description	Activity archive website with activity input features disposition and performance assessment using Laravel

2.2 Design System

After determining the criteria and collecting data through a questionnaire, the results of the questionnaire data will later be used to build a student recommendation system as members of the academic community project using *content-based filtering*. The process of applying *the content-based filtering* method for recommendations is described in *Figure* 1.

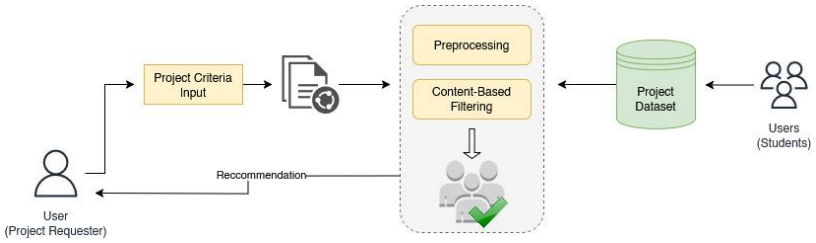

Figure 1. Design System

The illustration of this community project member recommendation system starts with the project requester inputting some of the desired project criteria. Then the criteria data will be input into the recommendation system along with the experience dataset of each student. The two data will go through two stages, namely *Preprocessing* such as *cleaning, case folding, stopwords*, *stemming* and *content-based filtering* which will calculate the similarity of some of the content of the two data. The output of this system is in the form of a list of student names as a result of the calculation *of content-based filtering similarity* in accordance with the criteria of the project that has been inputted at the beginning.

2.3 Content-Based Filtering

2.3.1 Preprocessing

Data or text preprocessing is the stage of selecting raw data to be processed in each document [16]. The main purpose of this data preprocessing stage is to process data in raw text for *cleaning* or removing punctuation and symbols [17], *Case Folding* or converting capital letters to lowercase, *Stopwords* or deleting common words [16], and *Stemming* or change the word to its root word form until it is ready to be analyzed according to the method used to improve the accuracy of the recommendations on the system.

2.3.2 TFIDF

TF-IDF is a technique in which the words in a document will be changed in numerical form to give weight to each word that appears to indicate the importance of the word in a document [11]. The first stage is to calculate the TF (*Term Frequency*) value or the number of words or terms in the document. The TF calculation uses the formula in equation 1.

$$TF\ (t,D) = \frac{frequency\ of\ term\ (t,D)}{maxOthers(t,D)} \quad (1)$$

After calculating TF, then calculating IDF (*Inverse Document Frequency*) with the formula in equation 2. The logs used in the TF-IDF library in sklearn are natural logarithms, which can also be written as Ln or e-based logarithms with e = 2.718281828459 [18].

$$IDF\ (t,D) = Log(\frac{N}{df(t)}) \quad (2)$$

Information:
IDF (t,D) = IDF value for the keyword (term) t in document D.
N = Total number of documents
df (t) = the number of documents containing the keyword (term) t

The next step is to calculate the weight or W. This weight calculation is used to find out the weight of each document. The greater the value of W, the higher the level of similarity of the document to the keyword [19]. The weight calculation is found in equation 3.

$$W = TF \times (IDF + 1) \quad (3)$$

Information :
W = weight of each word
TF = The number of occurrences of terms in the document
IDF (t,D) = IDF value for the keyword (term) t in document D.

2.3.3 Cosine Similarity

The calculation of similarity between each content in the study uses *the cosine similarity algorithm*. From this calculation, a similarity score will be obtained to the extent that the project criteria are similar to the student experience. The *cosine similarity* algorithm is shown in equation 4 [20].

$$\cos\theta_{(A,B)} = \frac{A \times B}{\|A\| \times \|B\|} \quad (3.4)$$

Information:
A = the weight of TF-IDF in document 1
B = TF-IDF weight on document 2

2.4 System Usability Scale

The SUS test in this study will use 10 questions using Indonesian which refers to the SUS testing questions conducted by john brooke. The question is as shown in Table 3 [21].

Table 3. Question of *System Usability Scale*

No	Question	1	2	3	4	5
1.	"I think that i would like to use this system frequently"					
2.	"I found the systemunnecessarily complex"					
3.	"I thought the system was easy to use"					
4.	"I think that i would need the support of a technical person to be able to use this system"					
5.	"I found the various functions in this system were well integrated"					
6.	"I thought there was too much inconsistency in this system"					
7.	"I would imagine that most people would learn to use this system very quickly"					
8.	"I found the system very cumbersome to use"					
9.	"I felt very confident using the system"					
10.	"I needed to learn a lot of things before I could get going with this system"					

From these several questions, the SUS score will be assessed as follows:

1. The scale used is strongly *disagree* to strongly *agree* with a value of 1-5.
2. For odd-numbered questions calculated by means, the value of the user's response is subtracted by a value of 1.
3. For even numbered questions calculated by means, a value of 5 is subtracted by the user's response value.
4. Add up the response values that have been calculated in steps 2 and 3, and then multiply by a value of 2.5.

The results of this calculation will convert the value range to between 0-100, which is then interpreted into *an adjective rating* to further clarify the level of *usability of the system* after a survey, such as Figure 2 [22].

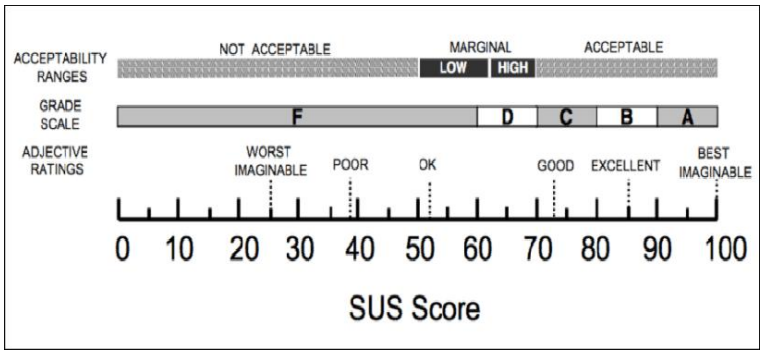

Figure 2. SUS Score

2.5 Mean Average Precission

The *Mean Average Precission* or MAP method is a metric used to measure the performance of a model that performs a document or information search task. The MAP method is suitable for algorithms whose output returns an item's ranking order, where each item can be hit (relevant) or missed (irrelevant) by the user [23]. The result of this study is a recommendation in the form of similarity values and student names. Then from the results of the recommendations, an evaluation will be carried out with MAP.

There are three stages in MAP, namely calculating precision, *calculating mean precision*, and calculating MAP values. The first stage, *precission* is calculated by dividing the number of relevant documents by all the documents displayed by the system. Then the *mean precission* is calculated by dividing the total *precission* value by the number of relevant documents. In the last stage, MAP is obtained by dividing the total *mean precission* (MP) by the number of experiments. The MAP value is said to be good if a system produces a value close to 1 [24].

$$MAP@N = \frac{1}{|U|} \sum_{u \in U} (AP@N)_u \tag{5}$$

Information:

MAP@N = Mean average precission value for N number of recommended items for all users
N = Number of recommended items
U = User group

3. RESULTS AND DISCUSSION

This chapter explains the results or implementation and discussion of the system that has been designed in the research methodology chapter.

3.1. Data Collecting

Data was obtained from an online survey to UIN Malang Informatics Engineering students from semesters 1-7 through Google Form. The survey was disseminated from January 5 to February 17, 2024 and was carried out in stages starting from semester 1 to semester 7. From this time vulnerability, as many as 198 project data for the final project of the UIN Malang Informatics engineering student course were obtained which would later be used as the main data on the system that was inputted into the database.

3.2. Database

Database implementation is a figurean in creating databases on the system that is built, the creation of this database uses MySQL which functions to support the system to be built.

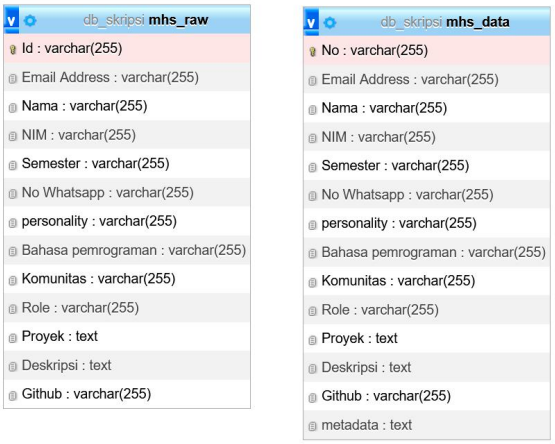

Figure 3. Database Implementation

The "mhs_raw" table contains data from project experiences that have been carried out by students such as email columns, Name, NIM, Semester, Whatsapp No, personality, Programming language, Community, Role, Github which is varchar data type, Project and Description columns which are text data types. In the second table, Table "mhs_data" also has the same columns as Table "mhs_raw" but in Table "mhs_data" there are metadata columns of type data text which contain a combination of columns Personality, Programming Language, Community, Role, Project, Description. Once the database is created in the localhost MySql, it is configured so that the database is connected to the system. On the database connection python using the library 'from flask_mysqldb MySQL import`.

3.3 System Implementation

The recommendation system in this study was developed as a website using the Flask framework. Flask is a *framework* that can integrate python to process data and html, css, javascript for the display or visualization of the data used [25]. Here are some views of the system that has been created.

3.3.1 Landing Page

On the landing page there is a navbar that leads to several other pages and also the title of the research conducted by the author under which there is a button to go directly to the recommendation page as in Figure 4.

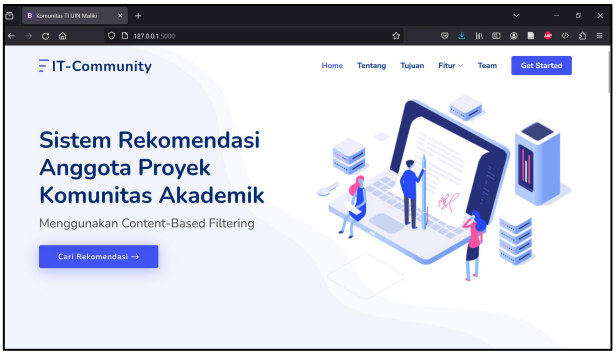

Figure 4. Landing Page

3.3.2 Data Page

This page will display all the data that has been registered or entered into the system database in the form of a Table. The data displayed on this page is raw data obtained from google forms which are then exported to the database, as shown in Figure 5.

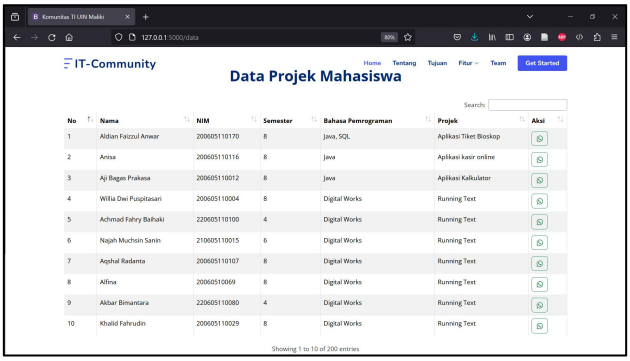

Figure 5. Data Page

3.3.3 Recommendation Page

The recommendation page is the main feature of the system developed in this study, which is to search for project recommendations. On this page, users can directly input some of the project criteria they want which are displayed in form or form-select.

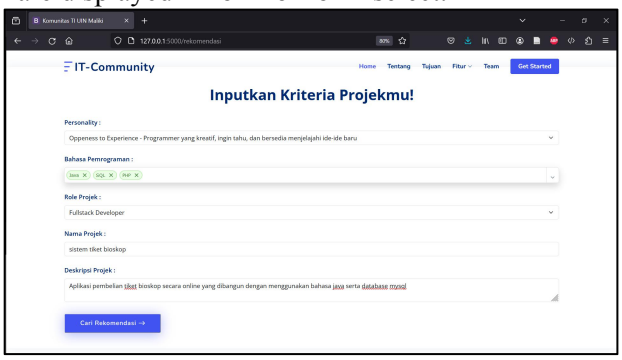

Figure 6. Recommendation Input Page

After the user inputs the desired project criteria and clicks *the* search button for recommendations button, the system will process for a moment and will provide recommendations based on the calculation of the similarity level calculated using *cosine similarity*. The system will display the top 4 recommendations according to the input project as shown in Figure 7.

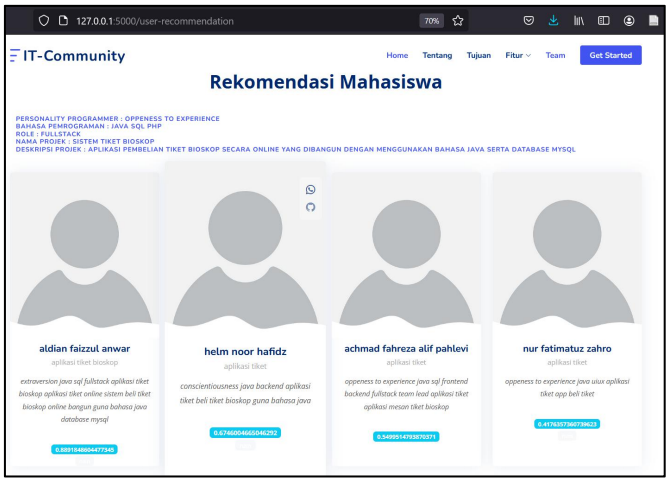

Figure 7. Recommendation Result

As in Figure 7, after the user inputs the desired project criteria, it will display 4 recommendations with the highest *similarity* value. On this page, you will display some information related to the students recommended by the system such as names, projects that have been created, project descriptions, *similarity* scores shown with light blue highlights.

In addition, if the user agrees with the results of the recommendation, the user can follow up by contacting the recommended student's *whatsapp* contact by clicking on the whatsapp logo in the upper right corner of the recommendation item display.

3.4 System Usability Scale

To test the functionality of the system that has been created, especially the recommendation feature in this study uses *the System Usability Scale* (SUS) method. The test was carried out through questionnaires and direct interviews with 25 respondents using *purposive sampling methods or techniques*. The following are the results of the SUS test.

Table 4. SUS Score

Respondent	Calculation Score										Sum	SUS Value (total x 2.5)
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
Student 1	3	0	4	1	3	3	3	3	4	3	27	68
Student 2	4	3	4	3	4	3	4	3	3	4	35	88
Student 3	3	3	3	2	3	3	2	3	3	3	28	70
Student 4	4	2	4	2	4	3	4	2	4	2	31	78
Student 5	3	3	3	2	3	3	3	3	3	3	29	73
Student 6	3	3	3	3	4	4	4	3	4	3	34	85
Student 7	3	3	3	2	3	3	3	3	2	3	28	70
Student 8	4	3	4	3	4	2	4	3	2	3	32	80
Student 9	3	3	3	3	4	2	4	3	3	3	31	78
Student 10	4	2	2	2	2	3	2	2	2	2	23	58
Student 11	3	3	3	3	3	2	3	3	3	3	29	73
Student 12	3	3	3	3	3	2	3	3	3	3	29	73
Student 13	4	4	4	1	4	4	4	4	3	4	36	90
Student 14	3	3	3	3	4	3	4	3	3	3	32	80
Student 15	3	3	3	3	3	3	3	3	3	3	30	75
Student 16	4	4	4	3	3	4	3	4	3	4	36	90
Student 17	4	3	3	3	4	3	4	3	3	3	33	83
Student 18	4	3	4	3	4	4	4	3	3	3	35	88
Student 19	3	2	4	2	3	3	3	2	3	2	27	68
Student 20	4	3	4	3	3	3	3	3	3	3	32	80
Student 21	3	3	3	2	4	4	4	3	4	3	33	83
Student 22	4	3	4	3	4	2	4	3	2	3	32	80
Student 23	4	4	4	1	4	3	4	4	3	4	35	88
Student 24	4	4	4	4	3	4	3	4	3	4	37	93
Student 25	4	2	4	2	4	3	4	2	4	2	31	78
Total Value												**1963**
SUS Average Score (total score/n)												**79**

3.5 Mean Average Precission

The *Mean Average Precision* or MAP test in this study was carried out by comparing the results of the recommendations produced by the system with actual data or data of project members that have been carried out in the informatics engineering study program of UIN Malang. The following are the results of the MAP calculation from the test results.

Table 5. Similarity Result

Data	Similarity Score			
	Recommendation 1	Recommendation 2	Recommendation 3	Recommendation 4
Document 1	0.6827	0.6339	0.3823	0.2720
Document 2	0.7397	0.2911	0.1247	0.1110
Document 3	0.6832	0.5152	0.2060	0.1228
Document 4	0.5538	0.4339	0.3824	0.3647
Document 5	0.4725	0.3867	0.2953	0.1947

Table 6. MAP Result

Data	Similarity Score				AP
	Recommendation 1	Recommendation 2	Recommendation 3	Recommendation 4	
Document 1	1	1	1	1	1
Document 2	1	0	0	0	1
Document 3	1	1	0	0	1
Document 4	1	1	0	0.75	0.91
Document 5	1	1	0	0	1
		Total			4.91
MAP = $\frac{Sum\ value\ of\ AP}{Sum\ Document}$ x 100%					**98%**

4. DISCUSSION

The results of the study show that the *content-based filtering recommendation system* with the TF-IDF approach and *cosine similarity* is able to produce relevant recommendations, as evidenced by a MAP score of 0.98. This shows the system can identify a match between the project and the student effectively. However, this system has limitations, such as data coverage that is limited to one institution and has not considered a collaborative approach in providing recommendations. Additionally, the SUS score indicates the user experience can still be improved.

Compared to previous studies such as Raharjo et al. (2022) which used Word2Vec and produced low accuracy, the TF-IDF approach in this study gave better results. The implications of these results suggest that the system can help the process of building project teams in an academic environment and support project-based learning more optimally.

5. CONCLUSION

The recommendation system that was successfully created in this study used several contents/criteria to determine the results of recommendations such as programmer personality, programming language, programmer role, project name, and project description. These criteria will affect the process of calculating similarity or cosine similarity produced by the recommendation system.

In the test using *the System Usability Scale* or SUS method which was carried out face *to face* to respondents, this system received a score of 79. This figure was obtained from the assessment of 25 respondents who were informatics engineering students, as shown in Figure 8.

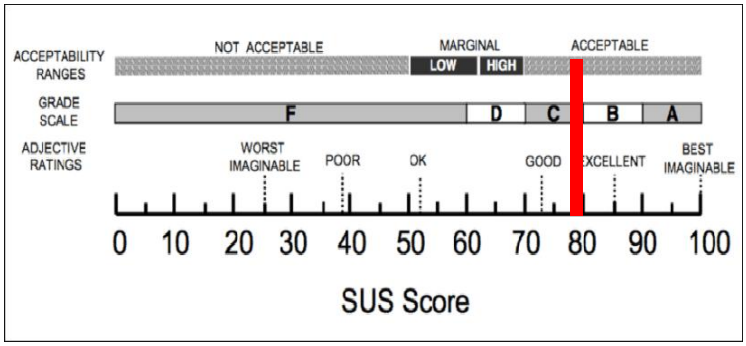

Figure 8. SUS Score

Based on *the acceptability range,* or the range of acceptance to users, this system is included in the category *of Acceptable* or acceptable with *marginal high*. This shows that users in general can accept well and feel satisfied with the usability of the system that has been built. Based on *the Grade Scale* this system gets *a grade of C* close to B, This shows that although the system is acceptable to the user but there are still some improvements that are considered to be improved for the ease of use of the system for the user.

To further validate whether the results of the recommendations generated by the system using *content-based filtering* are appropriate or not, a third test was carried out that compared the project members of the system recommendations with the members of the actual data project or

those that had been carried out by previous informatics engineering students. And the results in the third scenario got a score of 98% in the test. The test was carried out by looking at whether the names of students who appeared as project member records in each document were the same as the names of students in the actual project data. If the name of the recommendation that appears is the same as the actual data, then the results of the recommendation are considered relevant. So that it produces a final MAP score of 98% or out of 20 names in the actual project, 12 of which are relevant or the same.

As a follow-up, further research is recommended to explore *hybrid* methods by combining *content-based filtering* and *collaborative filtering* to improve the accuracy and personalization of recommendations. In addition, it is necessary to develop a more intuitive user interface (UI/UX) so that the system can be used more widely by lecturers and students in academic contexts and other collaborative projects.

REFERENCES

[1] A. (VOI) Nurjani, "PHK dan Pengangguran Bakal Terus Bayangi Indonesia di 2025." 2025.

[2] G. Grehenson, "Menaker: 12 Persen Pengangguran di Indonesia Didominasi Lulusan Sarjana dan Diploma | Universitas Gadjah Mada." Jogja, 2023. [Online]. Available: https://www.ugm.ac.id/id/berita/23493-menaker-12-persen-pengangguran-di-indonesia-didominasi-lulusan-sarjana-dan-diploma

[3] H. Setiarini, H. Prabowo, S. Sutrisno, and H. C. Gultom, "Pengaruh Soft Skill Dan Pengalaman Magang Kerja Terhadap Kesiapan Kerja Student Melalui Motivasi Kerja Sebagai Variabel Intervening (Studi Kasus Pada Student Feb Universitas Pgri Semarang)," *EKOBIS J. Ilmu Manaj. dan Akunt.*, vol. 10, no. 2, pp. 195–204, 2022, doi: 10.36596/ekobis.v10i2.941.

[4] H. Magdalena, "Sistem Pendukung Keputusan untuk Menentukan Pemberian Beasiswa di STMIK Atma Luhur," *J. Buana Inform.*, vol. 1, no. 1, 2022, doi: 10.24002/jbi.v3i2.325.

[5] Y. Ratuela, O. Nelwan, and G. Lumintang, "Pengaruh Hard Skill, Soft Skill dan Efikasi Diri Terhadap Kesiapan Kerja Pada Student Akhir Jurusan Manajemen FBE UNSRAT Manado," *J. EMBA*, vol. 10, no. 1, pp. 172–183, 2022.

[6] T. Suprihatin and E. A. Setiowati, "Dukungan Komunitas Belajar dan Ketekunan Akademik Pada Student," *Psychopolytan J. Psikol.*, vol. 5, no. 1, pp. 20–25, 2021, doi: 10.36341/psi.v5i1.1713.

[7] R. Kusumawati, P. Setyosari, I. N. S. Degeng, and S. Ulfa, "Designing a cmooc for academic communities to support awareness of scaffolding procedure on sharing knowledge," *J. Eng. Sci. Technol.*, vol. 16, no. 4, pp. 3576–3587, 2021.

[8] H. Praherdhiono, R. Kusumawati, E. A. Pramono, and A. Atmoko, "Strengthening MOOCs of academic community through scaffolding electronic automation," *Int. J. Innov. Creat. Chang.*, vol. 5, no. 4, pp. 431–446, 2019.

[9] N. Maudi, "Implementasi Model Project Based Learning Untuk Meningkatkan Kemampuan Komunikasi Matematis Siswa," *JPMI (Jurnal Pendidik. Mat. Indones.*, vol. 1, no. 1, p. 39, 2016, doi: 10.26737/jpmi.v1i1.81.

[10] R. Habibi and M. D. Albanna, "Analisis Sistem Rekomendasi Pada Job Recommendation Berdasarkan Profil LinkedIn Menggunakan Cosine Similarity," *J. Tek. Inform.*, vol. 14, no. 3, pp. 118–122, 2022.

[11] P. N. Raharjo, A. Handojo, and H. Juwiantho, "Sistem Rekomendasi Content Based Filtering Pekerjaan dan Tenaga Kerja Potensial menggunakan Cosine Similarity," *J. Invra*, vol. 10, no. 2, pp. 1–6, 2022.

[12] A. Amin *et al.*, "The impact of personality traits and knowledge collection behavior on programmer creativity," *Inf. Softw. Technol.*, vol. 128, no. September, p. 106405, 2020, doi: 10.1016/j.infsof.2020.106405.

[13] T. Setiadi, "8 Bahasa Pemrograman yang Wajib Dikuasai di 2022 untuk Melejitkan Karier S1 Sistem Komputer S." 2022. [Online]. Available: https://sistem-komputer-

s1.stekom.ac.id/informasi/baca/8-Bahasa-Pemrograman-yang-Wajib-Dikuasai-di-2022-untuk-Melejitkan-Karier/e7ca85eebdf1d8c4c36bc09851649db8698d2bbc
[14] J. Juviler, "12 of the Best Programming Languages to Learn in 2022." 2022.
[15] D. Russo and K. J. Stol, "Gender Differences in Personality Traits of Software Engineers," *IEEE Trans. Softw. Eng.*, vol. 48, no. 3, pp. 819–834, 2022, doi: 10.1109/TSE.2020.3003413.
[16] M. Fajriansyah, P. P. Adikara, and A. W. Widodo, "Sistem Rekomendasi Film Menggunakan Metode Content Based Filtering," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 6, pp. 2188–2199, 2021, [Online]. Available: http://e-journal.uajy.ac.id/20600/
[17] N. M. A. J. Astari, Dewa Gede Hendra Divayana, and Gede Indrawan, "Analisis Sentimen Document Twitter Mengenai Dampak Virus Corona Menggunakan Metode Naive Bayes Classifier," *J. Sist. dan Inform.*, vol. 15, no. 1, pp. 27–29, 2020, doi: 10.30864/jsi.v15i1.332.
[18] Harishamzah, "Perbandingan Perhitungan Bobot TF-IDF secara Manual dan Menggunakan Python," *Medium*. 2020. [Online]. Available: https://medium.com/bisa-ai/perbandingan-perhitungan-bobot-tf-idf-secara-manual-dan-menggunakan-python-377392a165c6
[19] F. B. A. Larasati and H. Februariyanti, "Sistem Rekomendasi Product Emina Cosmetics Dengan Menggunakan Metode Content - Based Filtering," *J. Manaj. Inform. dan Sist. Inf.*, vol. 4, no. 1, p. 45, 2021, doi: 10.36595/misi.v4i1.250.
[20] van der Leeuw, "Assessing scientific progress through peer review using cosine similarity," 2022.
[21] J. Brooke, "SUS : A Retrospective," no. June, 2020.
[22] A. Bangor, P. Kortum, and J. Miller, "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale - JUX," *Journal of User Experience*, vol. 4, no. 3. pp. 114–123, 2021. [Online]. Available: https://uxpajournal.org/determining-what-individual-sus-scores-mean-adding-an-adjective-rating-scale/
[23] H. H. Arfisko and A. T. Wibowo, "Sistem Rekomendasi Film Menggunakan Metode Hybrid Collaborative Filtering Dan Content-Based Filtering," *e-Proceeding Eng.*, vol. 9, no. 3, pp. 2149–2159, 2022.
[24] M. I. Hasan, "Information Retrieval System artikel kesehatan menggunakan pembobotan tf. idf dan Latent Semantic Indexing," 2018, [Online]. Available: http://etheses.uin-malang.ac.id/12546/
[25] H. L. Walingkas and P. O. N. Saian, "Penerapan Framework Flask pada Pembangunan Sistem Informasi Pemasok Barang," *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 7, no. 2, pp. 227–234, 2023, doi: 10.35870/jtik.v7i2.729.