Journal of Science, Computing and Engineering Research (JSCER)
Volume-8, Issue-2, February 2025.
DOI: *https://doi.org/10.46379/jscer.2025.080212*

SMART MENTORING

J.Manikantha, T.Ramsai, P.Bhavani

Department of computer science and Engineering , Anurag University,Hyderbad, Telangana, India

Article Information

Received : 10 Feb 2025
Revised : 16 Feb 2025
Accepted : 20 Feb 2025
Published : 28 Feb 2025

Corresponding Author:
J.Manikantha

Abstract— This paper presents Mentoring Academy is a program which aims to integrate students of the Instituto Politécnico de Bragança in the academy through peer mentoring and peer tutoring. This program needed an application that permits its execution. Motivated by this demand, we aimed at proposing a web application that complies with the Mentoring Academy requirements. The system requirements were defined together with the program stakeholders. Based on the literature review, we opted for the technologies ASP.NET Core and MySQL, for the backend, and Angular 6, for the frontend. Subsequently, the system architecture was planned. SCRUM methodology was used in the application development, with biweekly meetings with the stakeholders to evaluate what was done and to define the efforts for the following fortnight. Eighteen features were developed and four were presented in this paper. Finally, the system was deployed on a Linux server provided by the Instituto Politécnico de Bragança, allowing it to be tested and later used by the Mentoring Academy users. Approximately 61% of the functional requirements defined in the system modeling were developed (23 out of 38) and some functionalities not previously defined were also implemented. The units tests developed for terms of components generation and HTTP request services, both exception-free. That said, the system provides students' integration, permitting them to have socio-academic inclusion and support in curricular units.

Keywords: ***web application; asp.net core; angular; mentoring; tutoring.***

Citation: J.Manikantha, T.Ramsai, P.Bhavani , "Smart Mentoring", Journal of Science, Computing and Engineering Research, 8(2), February 2025.

I. INTRODUCTION

Mentoring Academy program was created in the Instituto Politécnico de Bragança to contribute to the integration of students entering higher education for the first time, as well as contributing to their academic and personal success Mentoring Academy Official Website The Mentoring Academy program

This program counts on two coaching methods: (1) peer mentoring, which is composed of a mentor who shares experiences and helps a mentee to achieve a specific goal or entry effectively into a new environment, and (2) peer tutoring, which consists of tutors and tutees, who are also students, working together and teaching each other where the tutor helps the tutees in the learning process when they have questions.

The Mentoring Academy program was completely designed and, in order for it to be effectively launched and attend the Instituto Politécnico de Bragança student's needs, it required an application that would permit its execution. Motivated by this demand, we aimed at proposing a web application which complies with the Mentoring Academy requirements. Web applications deliver many benefits, for example: (1) easily accessible, since any computer or smartphone can be used to access it, (2) no installation required, since every computer or smartphone has a browser and (3) reachable by anyone anywhere in the world. To achieve our goal, we followed the steps of: (a) requirements elicitation, (b) architecture definition, (c) developing and (d) testing.

As a result, this work generated a web application for the Mentoring Academy program, which is a proposed solution for supporting the peer tutoring and peer mentoring coaching methods. The completion of this work permits the release of the Mentoring Academy program into the academic environment. That said, the system provides students' integration, permitting them to have socio-academic inclusion and support in curricular units.

The remainder of the paper is organized as follows: Section II presents the related works regarding peer mentoring and tutoring applications; Section III presents the requirements elicitation; Section IV presents the system architecture definition; Section V is about the development of the Mentoring Academy web application, how the methodology was used, the features and unit tests developed

and, finally, Section VI presents the conclusions of this work and possible future works.

II. RESEARCH METHODOLOGY

The related works study regarding peer tutoring and peer mentoring applications had the purpose to collect the most recent features developed in modern applications so they could serve as a basis on the web application requirements definition of this work.

Akobe, Popoola, Atayero, et al. [2] proposed an online peer tutoring application in which tutors are responsible for (1) accepting requests from tutees and in turn holding tutoringsessions for them and (2)uploading important documents to the platform for the tutees. Spanorriga, Tsiotakis, and Jimoyiannis [3] designed and implemented an e-mentoring program for primary education teachers with low teaching experience. This application had three main functionalities:

private chat between each mentor and mentee, (2) open-community space where any participant can contact one another and (3) closed group containing all the mentors so they can exchange experiences between them. Wasilewski, Nonoyama, Dale, et al. [4] developed a peer mentoring training and web-based peer support program for ventilator-assisted individuals caregivers including informatiosharing, peer-to-peer communication, and peer mentorship. The program includes (1) private chat, (2) a public discussion forum and (3) weekly moderated chats. Clemmensen and Nørbjerg [5] developed a digital peer tutoring for shop floor workers. They aimed to develop capabilities among shop floor workers to use short videos to design and document solutions to operational and collaboration issues related to assistive technologies (collaborative robots), so others could watch their videos and learn from them Thakare, Jadhav, Mane, et al. [6] designed the development of an online mentoring system to promote and encourage students to actively participate in the academic activities. They pointed out twofeatures which are (1) the upload of an assignment by the mentor and (2) the upload of its answer by each mentee. Evans and Moore [7] developed a web-based peer-tutoring system called Online Peer-Assisted Learning (OPAL) for problem-based undergraduate courses. OPAL included (1) questions in which if the students answered one correctly they would become tutors of that specific question, (2) registration of “tutoring ticket” by each tutor, containing his or her accessibility and readiness to teach the question, instruction history and duration of previous tutoring sessions, (3) video reflections from the tutors and tutees about a tutoring session and (4) answers from the students of a 5point Likert scale about the tutoring sessions received. Phiri, Meinel, and Suleman [8] proposed the design and implementation of a peer tutoring teaching platform aimed at facilitating the orchestration of tutor-led learning activities. This platform had two main features: (1) upload of documents for the tutees and (2)upload of assignments for the tutees.

TABLE I. FEATURES ENCOUNTERED IN THE RELATED WORKS.

Code Description

F-01 Advisee can request session from a specific advisor.

F-02 Advisor can accept or refuse request from advisee for a session.

F-03 Advisor can upload files to the platform for the advisees.

F-04 Advisee can download files from the platform uploaded the advisor.

Code Description

F-05 Private chat between each advisor and advisee.

F-06 Open-community space where any participant can contact one another.

F-07 Closed group containing all the advisors so they can exchange experiences between them.

F-08 Weekly moderated chats.

F-09 Assignment upload by the advisor.

F-10 Assignment resolution from the advisee.

F-11 Advisoring guided per question.

F-12 Video upload by advisors and advisees containing feedback about a session.

F-13 Evaluation of sessions with a Likert scale.

III. REQUIREMENTS ELICITATION

The requirements elicitation begins with the identification of all the web application people. Therefore, nine people were identified: (1) common user, (2) tutor, (3) tutee, (4) mentor, (5) mentee, (6) school coordinator, (7) mentor coordinator and (8) tutor coordinator.

The first person, the common user, is a student that can request tutoring and mentoring, apply himself/herself for tutor and mentor roles, participate in activities, lectures or workshops and use the e-mail.

Table II exhibits all the functional requirements regarding the common user person. The tutor and the mentor are responsible for managing their meetings with the tutees and mentees, replying meeting invitations sent by their coordinator and providing relevant material to the tutees and mentees.

TABLE II. FUNCTIONAL REQUIREMENTS OF THE WEB APPLICATION

REGARDING THE COMMON USER PERSON.

Code Description

FR-01 A common user shall be able to request tutoring.

FR-02 A common user shall be able to request mentoring.

FR-03 A common user shall be able to apply himself/herself for tutor role.

FR-04 A common user shall be able to remove his/her application for tutor role.

FR-05 A common user shall be able to apply himself/herself for mentor role.

FR-06 A common user shall be able to remove his/her application for mentor role.

FR-07 A common user shall be able to update personal user information.

FR-08 A common user shall be able to reply presence in activity, lecture or workshop.

FR-09 A common user shall be able to send, read, update and delete emails.

FR-10 A common user shall be able to receive messages sent to email group.

The tutees and mentees, on the other hand, are responsible for replying meeting invitations sent by the tutor or mentor. Table III exhibits all the functional requirements regarding these people.

TABLE III. FUNCTIONAL REQUIREMENTS OF THE WEB APPLICATION REGARDING THE TUTOR, MENTOR, TUTEE AND MENTEE PEOPLE.

Code Description

FR-11 The tutor/mentor shall be able to schedule a meeting with the tutees/mentees.

FR-12 The tutor/mentor shall be able to update a meeting with the tutees/mentees.

FR-13 The tutor/mentor shall be able to list the meetings scheduled by him/her.

FR-14 The tutor/mentor shall be able to cancel a meeting scheduled by him/her.

FR-15 The tutor/mentor shall be able to close a past meeting by submitting a report.

FR-16 The tutor/mentor shall be able to obtain a participation

certificate.

FR-17 The tutor/mentor shall be able to reply a meeting invitation sent by the coordinator.

FR-18 The tutor/mentor shall be able to provide material to the tutees/mentees.

FR-19 The tutee/mentee shall be able to reply a meeting invitation sent by the tutor/mentor.

FR-20 The tutee/mentee shall be able to report a meeting feedback.

FR-21 The tutee/mentee shall be able to close a meeting by submitting a report. available by the tutor/mentor.The coordinators have the responsibilities of opening the mentoring matriculation, assigning a mentor/tutor to each mentee/tutee, manage the mentor and tutor roles, issue the mentoring participation certificate and schedule meetings with the mentors and tutors. Table IV exhibits all the common functional requirements regarding both people.

TABLE IV.

FUNCTIONAL REQUIREMENTS OF THE WEB APPLICATION REGARDING THE COORDINATORS PEOPLE.

Code Description

FR-23 The coordinators shall be able to open mentoring matriculation.

FR-24 The coordinators shall be able to assign mentor to mentee and tutor to tutee manually.

FR-25 The coordinators shall be able to assign or remove mentor or tutor role.

FR-26 The coordinators shall be able to automatically assign mentor to mentee and tutor to tutee.

FR-27 The coordinators shall be able to issue mentoring participation certificate.

FR-28 The coordinators shall be able to list meetings between mentors and mentees or tutors and tutees.

FR-29 The coordinators shall be able to schedule a meeting with mentors and tutors.

FR-30 The coordinators shall be able to update meeting scheduled by him/her.

FR-31 The coordinators shall be able to cancel meeting scheduled by him/her.

FR-32 The coordinators shall be able to list meetings scheduled by him/her.

FR-33 The coordinators shall be able to close meeting scheduled by him/her.

Finally, there are some functional requirements regarding the system, which is responsible for notifying the interested participants about

(1) session scheduling,

(2) session cancellation,

(3) mentoring certificate issuance,

(4) tutoring certificate issuance and

(5) activity, lecture or workshop certificate issuance.

Table V exhibits all the functional requirements regarding the system.

TABLE V. FUNCTIONAL REQUIREMENTS OF THE WEB APPLICATION REGARDING THE SYSTEM.

Code	Description
FR-34	The system shall be able to notify session scheduling to interested parties.
FR-35	The system shall be able to notify session cancellation to interested parties.
FR-36	The system shall be able to notify when a mentoring certificate is issued.
FR-37	The system shall be able to notify when a tutoring certificate is issued.
FR-38	The system shall be able to notify when an activity, lecture or workshop certificate is issued.

IV. THEORY AND CALCULATION

3.1 Theoretical Foundation

1. Social Learning Theory: The project is based on the social learning theory, which suggests that learning occurs through observation, imitation, and reinforcement.

2. Mentorship Theory: The project also draws from mentorship theory, which emphasizes the importance of guidance, support, and feedback in the learning process.

3. Personalized Learning: The project incorporates personalized learning principles, which involve tailoring the learning experience to individual students' needs, interests, and abilities.

3.2 System Design and Calculation

System Components

1. Student Profiling: A database to store student information, including academic records, interests, and learning styles

2. Mentor Profiling: A database to store mentor information, including expertise, experience, and communication styles.

3. Matching Algorithm: A algorithm to match students with mentors based on their profiles.

4. Communication Platform: A platform for students and mentors to communicate, share resources, and track progress

3.3 Matching Algorithm Calculation

1. Student-Mentor Matching: The algorithm calculates the similarity between student and mentor profiles using a weighted sum of the following factors:

- Academic compatibility (30%)
- Interests and hobbies (20%)
- Learning style and preferences (20%)
- Communication style and preferences (30%)

2. Matching Score: The algorithm generates a matching score for each student-mentor pair, ranging from 0 to 100.

3. Optimal Matching: The algorithm selects the mentor with the highest matching score for each student.

3.4 System Evaluation Calculation

1. Student Satisfaction: A survey to measure student satisfaction with the mentoring program, using a scale of 1 to 5.

2. Mentor Satisfaction: A survey to measure mentor satisfaction with the mentoring program, using a scale of 1 to 5.

3. Academic Performance: A comparison of students' academic performance before and after participating in the mentoring program.

4. Program Effectiveness: A calculation of the program's effectiveness in improving student outcomes, using a combination of the above metrics.

V. SYSTEM ARCHITECTURE

The web application architecture describes the interactions between applications, databases and middleware systems on the web. As any typical web application, there are two different codes (sub-programs) running side-by-side:

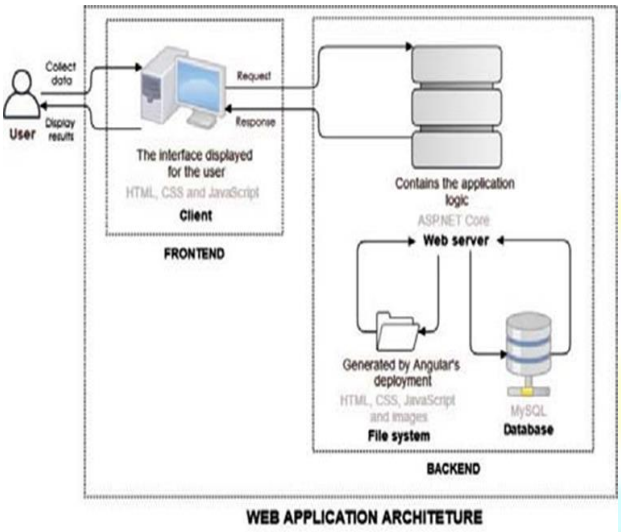
(1) client-side code, which is the code that is in the browser and responds to user inputs.

(2) server-side code, which is the code that is on the server and responds to the Hypertext Transfer Protocol (HTTP) requests.

The model chosen for the Mentoring Academy web application components is the “one web server, one database”, which is the simplest as well as the least reliable web app component model. Such a model uses a single server as well as a singledatabase.

The web application architecture chosen is the single page applications (SPAs), which allows dynamic interactions by means of providing updated content to the current page instead of loading the entire page after any user interaction. SPAs are designed in a way so that they request for most necessary content and information.

Figure 1 illustrates the interactions between user, client, web server, file system and database. It begins when the user interacts with the interface over the browser starting a HTTP request. This request is sent to the web server, which resolves it consulting the file system and the database and replies a HTTP response. This HTTP response may contain static files (HTML, CSS, JavaScript and images) or data from the database. The static files served by the file system are generated by the Angular' deployment.



VI. RESULTS AND DISCUSSION

4.1 Results

1. Students' Integration: The system provides students' integration, permitting them to have socio-academic inclusion and support in curricular units.

2. Components Generation: The unit tests developed for the web application proved promising in terms of components generation.

3. HTTP Request Services: The unit tests also proved promising in terms of HTTP request services, both exception-free.

4. Mentor, Mentee, Tutor, and Tutee Tables: The system provides tables with summed information and export to Excel functionality.

5. Graphs and Reports: The system provides graphs about assigned mentees and reunions per week.

6. Meeting Invitation Acceptance and Decline: The system allows meeting invitation acceptance and decline through personal email notification.

7. Import New Users: The system allows importing new users recently enrolled in the institution.

Quantitative Results

1. Functional Requirements Developed: 23 out of 38 functional requirements defined in the system modeling were developed (61%).

2. Unit Tests: The unit tests developed for the web application proved promising in terms of components generation and HTTP request services.

Qualitative Results

1. Improved Students' Integration: The system provides students' integration, permitting them to have socio-academic inclusion and support in curricular units.

2. Enhanced Mentorship Experience: The system provides a platform for mentors and mentees to interact, share resources, and track progress.

3. Increased Efficiency: The system automates several tasks, such as meeting invitation acceptance and decline, and import new users.

4.2 Discussion

smart Mentoring project has demonstrated promising results in providing a platform for students' integration, mentorship, and socio-academic inclusion. The system's ability to automate tasks, provide graphs and reports, and facilitate communication between mentors and mentees has shown potential in enhancing the mentorship experience.

Implications

1. Improved Student Outcomes: The system's ability to provide personalized support and guidance to students may lead to improved academic outcomes and increased student satisfaction.

2. Increased Efficiency: The automation of tasks and provision of graphs and reports may reduce the administrative burden on mentors and administrators, allowing them to focus on more critical tasks.

3. Scalability: The system's design and architecture may enable it to be scaled up to accommodate larger numbers of students and mentors.

Limitations

1. Limited Functional Requirements: The system currently only addresses 61% of the functional requirements defined in the system modeling, leaving room for further development and improvement.

2. Lack of Data Mining and Analysis: The system does not currently provide data mining and analysis capabilities, which may limit its ability to provide insights into the effectiveness of the mentorship program.

Future Directions

1. Develop Remaining Functional Requirements: The remaining 28 functional requirements should be developed and tested to ensure the system meets all the necessary requirements.

2. Integrate Data Mining and Analysis: The system should be integrated with data mining and analysis capabilities to provide insights into the effectiveness of the mentorship program.

3. Conduct Evaluation Studies: Evaluation studies should be conducted to assess the impact of the Smart Mentoring system on student outcomes and mentorship experiences.

4.3 Comparison with Recent Developments

1. personalized Learning: Recent developments in mentoring platforms, such as DreamBox and MentorNet, also focus on personalized learning experiences for students.

2. Data-Driven Insights: Many recent mentoring platforms, including iMentor and eMentor, utilize data-driven insights to inform mentoring relationships and improve student outcomes.

3. Online Platforms: The majority of recent mentoring platforms, such as MentorHub and (link unavailable), utilize online platforms to facilitate mentoring relationships and provide resources for mentors and mentees.

Differences

1. Integration with Academic Systems: The Smart Mentoring project integrates with academic systems to provide a seamless experience for students and mentors, whereas many recent mentoring platforms operate independently.

2. Focus on Socio-Academic Inclusion: The Smart Mentoring project places a strong emphasis on socio-academic inclusion, recognizing the importance of social relationships in academic success.

3. Automated Matching Algorithm: The Smart Mentoring project utilizes an automated matching algorithm to pair mentors and mentees, whereas many recent mentoring platforms rely on manual matching processes.

Unique Contributions

1. Holistic Approach: The Smart Mentoring project takes a holistic approach to mentoring, recognizing the interconnectedness of academic and social relationships.

2. Contextualized Support: The project provides contextualized support for mentors and mentees, taking into account the unique needs and circumstances of each individual.

3. Scalability: The Smart Mentoring project is designed to be scalable, allowing it to be easily adapted and implemented in various academic settings.4.4 Challenges and Future Work

A. Mentor and mentee connection

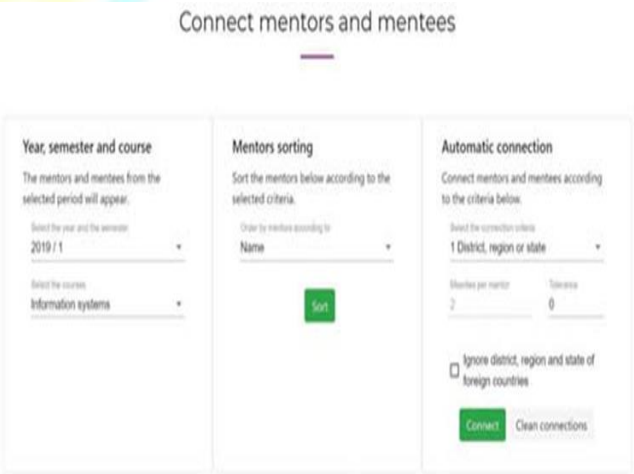
The following figures exhibit the screenshots about the mentor and mentee connection functionality, which complies the FR-24 and FR-26. Figure 2 shows the page which is responsible

for the manual and automatic connection between mentors and mentees made by the school and mentor coordinators. The first part of the page is composed of three boxes:

(1) the first box is responsible for the year, semester and course selection, which will filter all shown mentors and mentees;

(2) the second box is responsible for sorting the mentors, which offers the options “Name” and “Quantity of assignments”;

(3) the third box is responsible for the automatic connection between mentors and mentees.

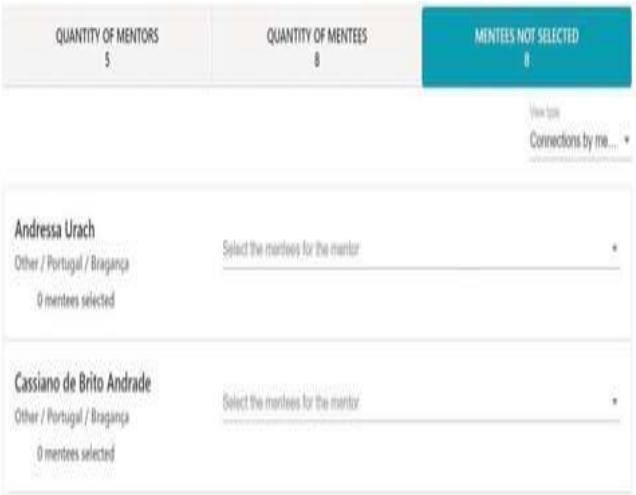


The second part of the page, presented by Figure 3 displays three important information: quantity of mentors, quantity of mentees and quantity of mentees that are not assigned to any mentor. Below it, there is a combo box which contains three view types:

(1) connection by mentor, which is the selected one in the figure,

(2) mentor table and

(3) mentee table. The first view, shown by the figure, contains each mentor in a box, with their personal information (gender, country and city) and quantity of mentees selected for each one. The combo box on each mentor's box contains the mentees which the coordinator can manual or automatic assign them to the mentors.



The automatic connection offers three rules, which are

(1) district, region or state,

(2) country and

(3) spoken language. The coordinator can select how many rules he/she wants, in any order he/she wants. The number that appears in the beginning of the rule represents the order of which the user selected the rule, so when the user clicks on the "Connect" button, the system will automatically connect mentors and mentees following these rule orders.

At first, the system respects a maximum quantity of mentees per mentor, which is shown in the third box. If the coordinator needs a bigger quantity of mentees per mentor, he/she can put how much more he/she needs to add in the "Tolerance" field. In case the coordinator wants to apply the "District, region or state" rule only to portuguese students, he/she can check the "Ignore district, region or state of foreign countries" checkbox. If the coordinator needs to restart the connection, he/she can click on the "Clean connection" button to undo all the selections.

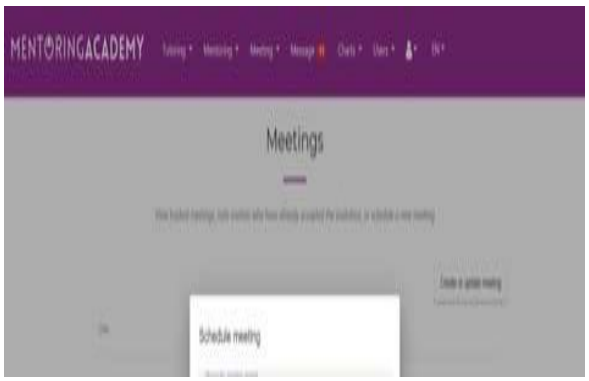
B. Schedule meeting

The following figures present the screenshots about the schedule meeting functionality, which complies with the FR11 and FR-29. Figure 4 shows the pop-up that appears once the user requests to create or update a meeting. The popupoffers three operations:

(1) create meeting,

(2) update meeting and

(3) cancel meeting. This section will show the "Create meeting"functionality



C. Graphs

It was developed for the Mentoring Academy system two graphs to help coordinators keep track of the program's status more easily. The first graph is about the assigned mentees, like shown in Figure 6. The coordinator can filter the mentees by their countries and the year of when they made part of the mentoring program, and the graph will show, for each country, the quantity of mentees not assigned to any mentor yet in a red bar and the quantity of mentees that were assigned to a mentor

D. System notification

The following figures exhibit the screenshots about system notifications regarding meeting scheduling and cancelling, which complies with the FR-34 and FR-35. Figure 8 shows a screenshot of the email sent to a user's personal email about a meeting scheduled by the "Coordenador de Mentor" on the day

"January 1st of 2020" at 9 o'clock.

If the user clicks on the "ACCEPT" button, he/she will be redirected to the page shown by the Figure 9, which confirms that he/she accepted the invitation successfully..

VII. CONCLUSION

Academy requirements. The methodology adopted was divided As a result, this work generated a web application for the Mentoring Academy program, which is a proposed solution for supporting peer tutoring and peer mentoring coaching methods. The completion of this work permits the Mentoring Academy program to be released into the academic environment.

That said, the system provides students' integration, permitting them to have socio-academic inclusion and support in curricular units. The units tests developed for the web application proved promising in terms of (1) components generation and (2) HTTP request services, both exception-free. In addition, approximately 61% of the functional requirements defined in the system modeling were developed (23 out of 38) and some functionalities not previously defined were also developed, like (a) mentor, mentee, tutor and tutee tables with summed information and export to Excel functionality, (b) graphs about assigned mentees and reunions per week, (c) meeting invitation acceptance and decline through the personal email notification and (d) import new users recently enrolled in the institution. Therefore, this works left 28 functional requirements undone. Thus, as future works, these functional requirements could be developed and tested. In addition, there are some studies that could be made on the system regarding data mining over the peer mentoring and tutoring accomplishments in terms of (1) successful matchup between "mentor and mentees" and "tutor and tutees" and (2) academic success after participating in the Mentoring Academy program, in order to measure the program results in the academic environment.

REFERENCES

[1]. Mentoring Academy Official Website the mentoring academy program, https://mentoringacademy.ipb.pt/, Dec. 2019.

[2]. D. Akobe, S. Popoola, A. Atayero, O. Oseni, and S. Misra, "A web framework for online peer tutoring application in a smart campus," International Conference on Computational Science and Its Applications, Jun. 2019.

[3]. C. Spanorriga, P. Tsiotakis, and A. Jimoyiannis, "Ementoring and novice teachers' professional development: Program design and critical success factors," EdMedia and Innovate Learning, 2018.

[4]. M. Wasilewski, M. Nonoyama, C. Dale, D. McKim, J. Road, D. Leasa, R. Goldstein, and L. Rose, "Development of a web-based peer support program for family caregivers of ventilator-assisted individuals living in the community: Protocol for a pilot randomized controlled trial," JMIR Research Protocols, Feb. 2019.

[5]. S. Thakare, S. Jadhav, I. Mane, S. Pawar, and A. Kulkarni, "Online mentoring system (an online mentorstudent system)," International Journal of Engineering Trends and Technology, Jan. 2019.

[6]. DC Jullie Josephine, J Sudhakar, T Helan Vidhya, R Anusuya, G Ramkumar,"An Improved Multi class Breast cancer classification and Abnormality Detection based on Modified Deep Learning Neural Network Principles", Deep Learning in Biomedical Signal and Medical Imaging, CRC Press, Taylor and Francis, 2024.

[7]. M. Evans and J. Moore, "Peer tutoring with the aid of the internet," British Journal of Educational Technology, Jan. 2013.

[8]. Phiri, C. Meinel, and H. Suleman, "Peer tutoring orchestration: Streamlined technology-driven orchestration for peer tutoring," Proceedings of 9thInternational Conference on Computer Supported Education, Apr. 2017.

[9]. R. S. Vignesh, A. Kumar S, T. M. Amirthalakshmi, P. Delphy, J. R. Arunkumar and S. Kamatchi, "An Efficient and Intelligent Systems for Internet of Things Based Health Observance System for Covid 19 Patients," 2023 International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF), Chennai, India, 2023, pp. 1-8, doi: 10.1109/ICECONF57129.2023.10084066.

[10]. T. Clemmensen and J. Nørbjerg, " digital peer-tutoring'. early results from a field evaluation of a 'ux at work' enhancing learning format," Cardiff Ubiquity Press, 2019.