# A Machine Learning-Based Automatic Feedback System to Teach Cybersecurity Principles to K-12 and College Students

Department of Computer Sciences and Electrical Engineering, Marshall University, Huntington, WV

Eric M. Dillon dillon221@marshall.edu
Craig Carpenter II carpenter203@marshall.edu
John Cook cook458@marshall.edu
Thomas D. Wills wills55@marshall.edu
Husnu S. Narman narman@marshall.edu

***Abstract*—Feedback is an essential part of education to help students understand and learn from their mistakes. However, while students learn new content, there is mostly no live person to provide feedback, especially in a virtual environment. Therefore, there are many software for automated code reviews to provide feedback to programming language learners. However, there are no available auto command review tools for security tools except for each tool itself and operating system suggestions. There is also no feedback tool that constructively provides feedback according to learners' experiences in security subjects while learners practice with commands. Therefore, we developed an automatic feedback system that uses machine learning to create customized student feedback on cybersecurity topics. The foundation of the software was completed and tested in 2 undergraduate introductory computer science courses. Survey results collected from users indicate that the automatic feedback system improved the learning experience of 46% of successful participants and that 77% of successful participants were interested in the continued development of the system. 88% of successful participants felt that the system taught basic command-line skills effectively.**

***Index Terms*—education, educational technology, computer-aided instruction, electronic learning, computer science education, artificial intelligence, machine learning, computer security**

## I. Introduction

The growing dependence on technology has reemphasized the need for a well-prepared and appropriately trained cybersecurity workforce [1]. This critical workforce defends against international threats to economic, societal, and governmental security [2]. This need's urgency is reflected by the work of organizations such as the National Initiative for Cybersecurity Education, which is bringing awareness to cybersecurity issues by pushing for CTE pathways, educator conferences, and learning standards at the K-12 level [3].

However, K-12 and college students often need additional assistance when learning cybersecurity and computer science topics [4]. Students from low-income backgrounds are frequently unable to pursue a STEM education due to the lack of adequate access to constructive feedback throughout their primary, secondary, and collegiate education careers. Without access to feedback, students are demotivated to continue studying complex subjects like cybersecurity. Worse, they are likely to form serious misconceptions on what these subjects entail that are difficult to correct [5]. It is expensive to recruit new teachers and retrain existing teachers to teach cybersecurity and computer science topics; thus, it becomes necessary to find an alternative solution to encourage cybersecurity students' independent learning in the face of inequality.

Although there is a huge market of cybersecurity awareness training appropriate for K-12 students, including those provided by Amazon Cybersecurity Awareness Training and the Center for Development of Security Excellence [6], there are very few programs that offer cybersecurity *software* training in an age-appropriate manner. Furthermore, although foundational knowledge in programming can be developed on sites including Khan Academy [7], FreeCodeCamp [8], and Codecademy [9], these sites do not offer lessons aimed specifically towards cybersecurity. Codecademy does notably offer lessons on the command line, but these appear to be aimed more towards general use as opposed to an offensive or defensive use.

Our *objective* in this study is to develop an Automatic Feedback System to provide the constructive feedback which is specific to each user by using machine learning algorithms and analyzing the performance of the developed system based on user experience. As a prototype, the Automatic Feedback System is intended to reflect the experience when using the command line, similar to the aforementioned lessons offered by Codecademy. However, unlike Codecademy's broader and more basic intentions for the command line, the purpose of the Automatic Feedback System is to teach command-line skills relevant to beginning cybersecurity students. It does this by providing easy-to-understand feedback throughout the learning process. Students' mistakes and successes are saved by the program in its database, and students are able to view their progress at any time. The *key contributions* are (1) the development of the machine learning-based Automatic Feedback System, (ii) testing of the developed system with first-year students whose majors are related to computing, and (iii)

analyzing the collected data to understand the effectiveness of the developed system in terms of Quality of Experience.

The results show that the Automatic Feedback System can effectively be used by students for cybersecurity and other commands while practicing over command lines.

The rest of the paper is organized as follows: Section II will cover the tools and software used to create the Automatic Feedback System and will observe screenshots from the prototype. Section III will discuss the post-demo questionnaire and the reasoning behind it. Section V will explain our observations and evaluate the results of this survey. Section V will report our conclusions and discuss the project's plans and what changes are necessary based on our current findings. Finally, Section VI offers thanks to those who supported this study.

## II. Evaluation of Prototype

The Automatic Feedback System has a registration and login system that enables students to have individually tailored experiences. Future versions intend to have the ability to link student accounts with instructor accounts so that the Automatic Feedback System can be used in conjunction with existing courses. The account system, as well as the user's data and information, is saved in a MongoDB database. Fig. 1 shows the login screen from the user's perspective. Its user interface design is student-friendly and inspired by design elements used by the Amazon Future Engineer program [10], Khan Academy [7], and Codecademy [9]. The colorful, geometric design deemphasizes the intimidating nature of the cybersecurity subject and thus encourages both K-12 and college students to continue with their learning. The user interface is constructed with Tkinter, a Python library for graphical user interfaces.

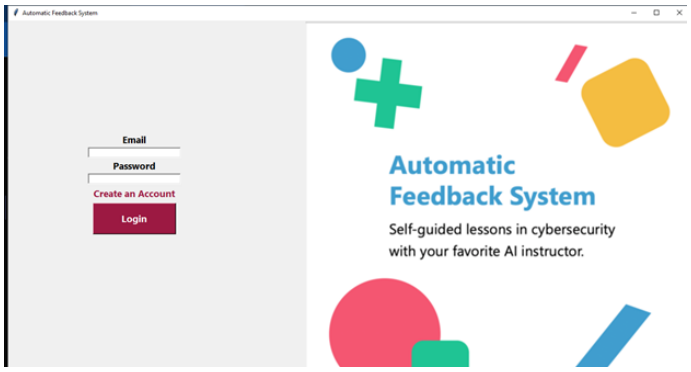


Fig. 1. Screenshot of the Automatic Feedback System login screen.

A diagnostic test has been implemented to give students insight into their performance. The diagnostic test, seen in Fig. 2, has a command line that connects to the user's system on the right side of the screen, and tasks are provided on the left side of the screen. As tasks are completed, the test will progress automatically. Students may skip questions, exit the test, or submit it at any time. This also allows the Automatic Feedback System to focus on skills a student is unfamiliar with as opposed to skills that the student is already comfortable with.

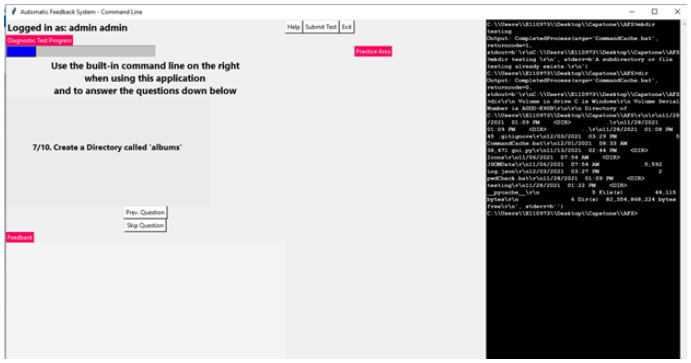


Fig. 2. Screenshot of the Automatic Feedback System diagnostic test screen.

There is an additional practice area very similar to the structure of the diagnostic test, as seen in Fig. 3. However, students receive feedback on their inputs as the program predicts what their intended command was. TensorFlow, a Python library for machine learning, was used to model an algorithm that checks student input for mistakes. This algorithm works by comparing the user's input to the expected input from the information in the provided manual. Once a mistake has been found, the system returns a prewritten explanation of the command it believes the user is attempting to use. This practice area also excludes the questions found in the diagnostic test, opting to allow the user to freely interact with the command line.

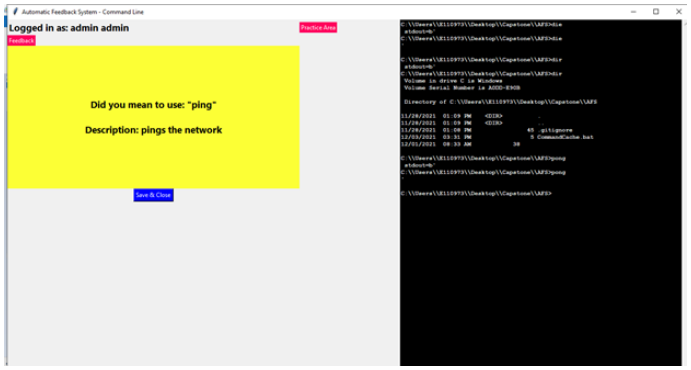


Fig. 3. Screenshot of the Automatic Feedback System practice area screen.

## III. Survey Methodology

The Automatic Feedback System was tested in two groups that constituted separate in-person sections of an introductory computer science course at Marshall University. The testing was administered on November 29, 2021, for session one, and November 30, 2021, for session two.

Participants were provided with an executable and a readme file. The former contained the program itself, whereas the latter contained information on how to access the survey. Students were instructed to create an account, log in to the system, and complete the diagnostic test. After completion, they were to fill out a survey responding with their opinions.

Participants were asked the following questions. Note that questions requesting personally identifying information (name, email, etc.) are excluded for the privacy of all participants. Additionally, for those who were unable to log in successfully, the survey concluded after completing question 2. Otherwise, participants continued with the remaining questions.

1) Before using the Automatic Feedback System, how familiar were you with the Windows command line?
2) Were you able to log into the program successfully?
3) What concepts presented by the Automatic Feedback System did you feel were explained well?
4) What concepts presented by the Automatic Feedback System did you feel were presented poorly or that could be improved on?
5) Disregarding their quality, what concepts presented by the Automatic Feedback System did you think were interesting?
6) On a scale of 1-5, how would you rate your understanding of the purpose of the Automatic Feedback System?
7) On a scale of 1-5, how would you rate the user-friendliness of the Automatic Feedback System?
8) On a scale of 1-5, how would you rate the educational benefits of the Automatic Feedback System?
9) On a scale of 1-5, would you be interested in continuing to use the Automatic Feedback System (either in its current form or in a more improved form)?
10) On a scale of 1-5, do you feel the Automatic Feedback System is unique compared to other comparable sites and programs (Khan Academy, Codecademy, etc.)?
11) On a scale of 1-5, how would you rate your overall experience using the Automatic Feedback System?
12) Do you have any additional comments, questions, or concerns?

- Questions 1 and 6 - 11 were quantified on a five-point scale, where 1 represents a highly negative perception, and 5 represents a highly positive reception.
- Question 2 was responded to with the answers "Yes, and I completed the diagnostic test," "Yes, but I did not complete the diagnostic test," "No, but I successfully created an account," and "No, and I did not receive a response regarding a successful account creation."
- Questions 3 - 5 was a multi-selection list including the options of "navigating the command line," "cybersecurity basics," and "systems administration."
- Question 12 served as a free-response section.

Some reasoning behind some of these questions is necessary. Questions 3 and 4 ask the inverse of the same idea in order to minimize the impact created when a positively- or negatively-phrased question results in an answer that does not accurately reflect the feelings of the respondent [11]. As seen in Section III, this resulted in one of the responses having a contradictory answer, indicating a more complicated perception of that idea.

The answers for questions 3 - 5 were chosen based on a group consensus of what was felt to be the main concepts presented by the Automatic Feedback System at the time of testing. Note that these are broad, elementary ideas due to the limited time of the testing and the low amount of experience that the testing group had with these concepts.

## IV. Results

The vast majority of participants were unfamiliar with the Windows command line. Only 30% reported an average to above-average familiarity with its use, while the remaining 70% reported no prior use or very little prior use (see Fig. 4). In other words, these participants were ideal to evaluate how students who are unfamiliar with a program will interact with the Automatic Feedback System learning software.

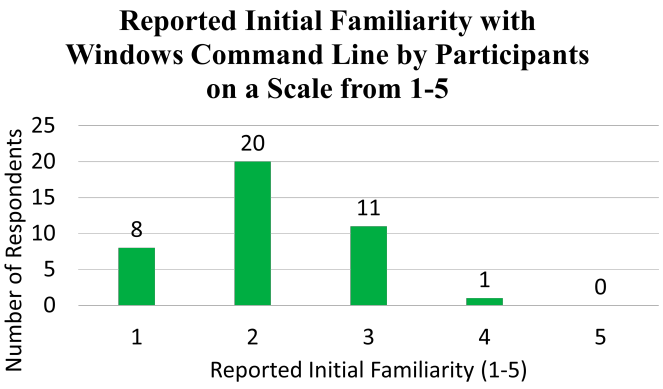


Fig. 4. Graph counting how the 40 participants reported their initial perception of their familiarity with the Windows command line.

14 out of 40 participants, or about 39% of all participants, were unable to register an account (see Fig. 5). Note that all participants who created an account were able to log in to the system. Attempts to mitigate this issue, including providing access to an existing demo account and additional on-site support. After multiple attempts, it was found that the issue was related to multiple failed attempts to connect to the database, preventing any information from being sent or received by the participant. Consequently, only the 26 participants unaffected by this error were asked further questions.

Out of the 26 successful participants, 23 of them felt that the program guided them on how to navigate the command line well (see Fig 6). Despite the target of the prototype being aimed at cybersecurity basics and systems administration, very few of these participants reported that these concepts were

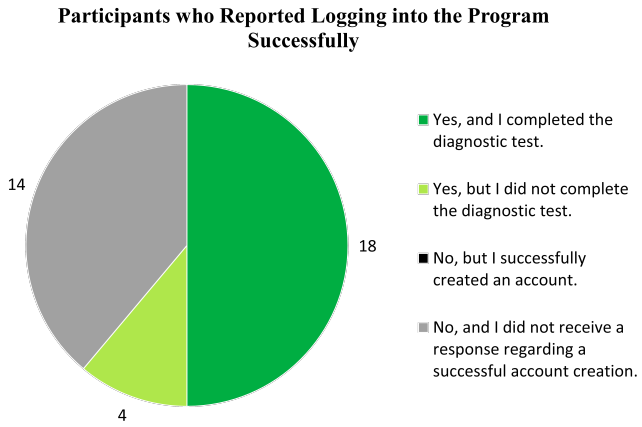


Fig. 5. Graph showing whether each of the 40 participants was able to log in to the Automatic Feedback System.

implemented well (4 and 6, respectively). This is likely due to confusion on how the command line directly relates to cybersecurity and systems administration.

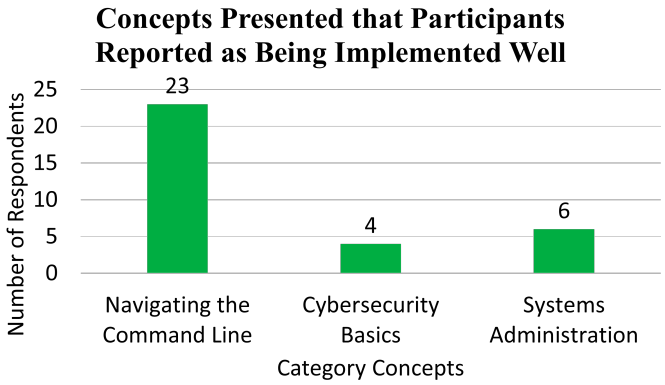


Fig. 6. Graph counting how the 26 successful participants reported their positive feelings of implementation of different content areas.

As discussed in Section IV, this question was followed up by a similar question but focused on what participants reported went wrong as opposed to what they felt went right (see Fig. 7). Participants overwhelmingly felt (18 out of 26) that cybersecurity basics were not covered well at this time, with systems administration trailing right behind (12 out of 26). Interestingly, this means that at least some participants reported that they felt the cybersecurity basics were presented both successfully *and* poorly. Whether this is an indication of mixed feelings or confusion towards the cybersecurity element of the Automatic Feedback System or a mistake made when reporting their responses is unclear.

When told to disregard the quality of the implementation, participants were interested in systems administration but not cybersecurity basics despite both concepts having generally identified as being poorly implemented (see Fig. 8). As mentioned earlier, this may be due to students being unfamiliar with how the Windows command line relates to cybersecurity. However, it may also be due to inexperience with the topic, a pre-established lack of interest (as many of these students are taking the introductory computer science course as part of requirements for another program as opposed to a direct interest in cybersecurity), or another unidentified factor. Regardless, the lack of interest in cybersecurity is something that needs to be addressed in future iterations of the Automatic Feedback System.

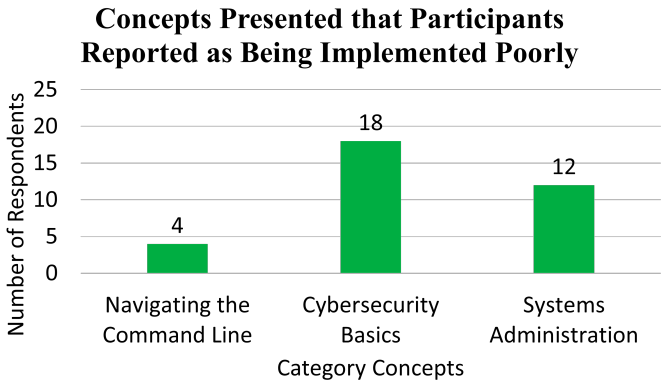


Fig. 7. Graph counting how the 26 successful participants reported their negative feelings of implementation of different content areas.

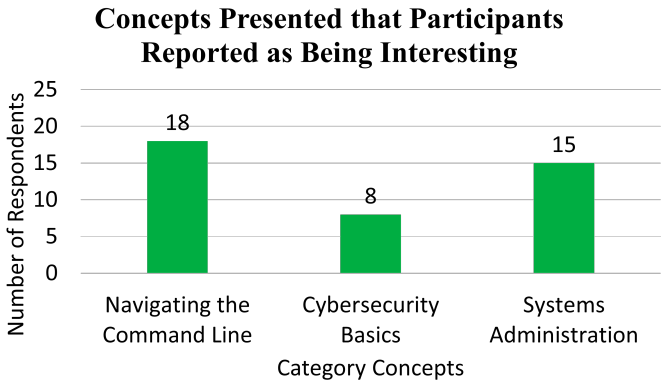


Fig. 8. Graph counting how the 26 participants who logged in successfully reported their interest in different content areas addressed by the Automatic Feedback System.

Participants reported a mixed understanding of the Automatic Feedback System's purpose (see Fig. 9). This may be due to a lack of clarity when initially describing the Automatic Feedback System to the participants, as they were not given information about the prototype in advance to the testing session being held. Furthermore, the poor perception of the software's teaching of cybersecurity basics may have also confused students on what the main goal of the Automatic Feedback System was.

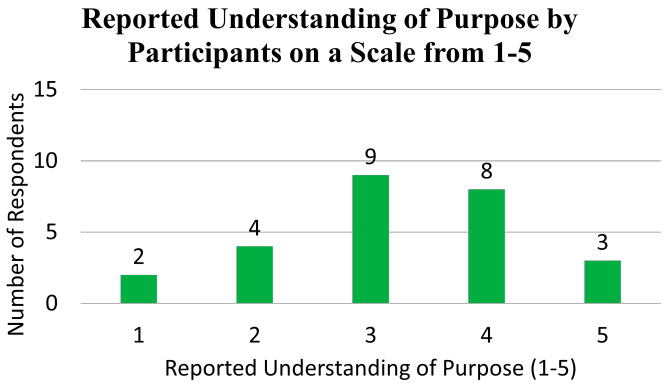


Fig. 9. Graph counting how the 26 participants who logged in successfully reported their understanding of the purpose of the Automatic Feedback System on a scale from 1 to 5.

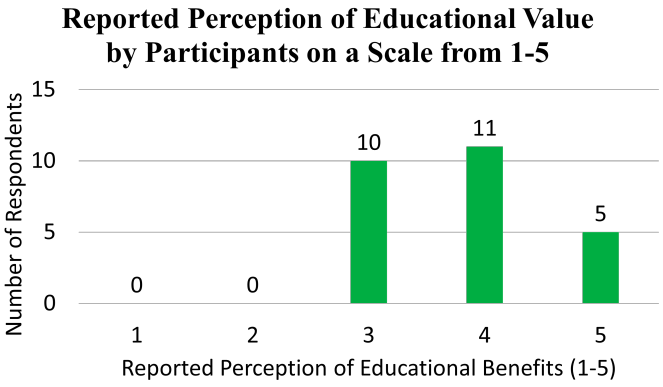


Fig. 11. Graph counting how the 26 participants who logged in successfully rated the educational benefits of the Automatic Feedback System on a scale from 1 to 5.

The mean of the reported user-friendliness was 3.4 (see Fig. 10), indicating the perception was broadly average to above average. As described in Section II, the user interface was directly inspired by that offered by Codecademy and Khan Academy. By basing the software's user interface on existing designs, students who have used these sites in the past were more likely to recognize the general format of the Automatic Feedback System. However, the attractiveness of the software–due to the generally rudimentary nature of Tkinter–was a notable detractor from receiving a definitely positive response as opposed to a more neutral response,

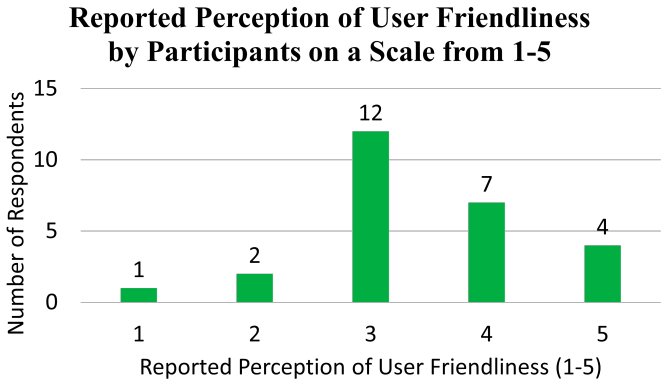


Fig. 10. Graph counting how the 26 participants who logged in successfully rated the user-friendliness of the Automatic Feedback System on a scale from 1 to 5.

All students responded with an average to the high perception of the educational value of the Automatic Feedback System, with an average score of 3.8 overall (see Fig. 11). The feedback that was returned to students was corrective on their mistakes and acted as a secondary instructor, which was well-received by students in this regard.

The mean of 3.5 among all participants regarding participants' desire to continue using the system (see Fig. 12) is deceptive compared to the previous questions. Continued use was a very mixed topic, with students reporting themselves to be severely disinterested or extremely interested in continuing their use with the program. This may be due to a lack of understanding of the purpose of the program, the lack of interest in learning cybersecurity topics, or the low perception of the implementation of cybersecurity topics. Consequently, all of these topics need to be addressed before further testing can begin.

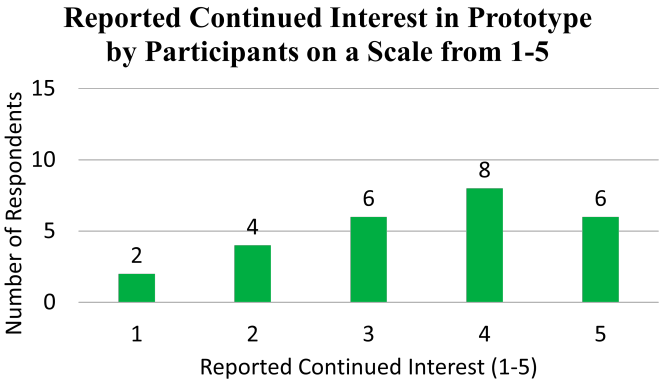


Fig. 12. Graph counting how the 26 participants who logged in successfully reported their interest in continuing to use the Automatic Feedback System on a scale from 1 to 5.

The uniqueness of the prototype was responded to broadly average to above-average, with a mean of 3.2 (see Fig. 13). As the current state of the prototype is similar to the command line tutorial offered by Codecademy, it is understandably seen as an alternative to this course as opposed to its unique structure. As the program develops, the major intent is to connect the Automatic Feedback System with prominent and freely-available cybersecurity software, which is not offered by other programs.

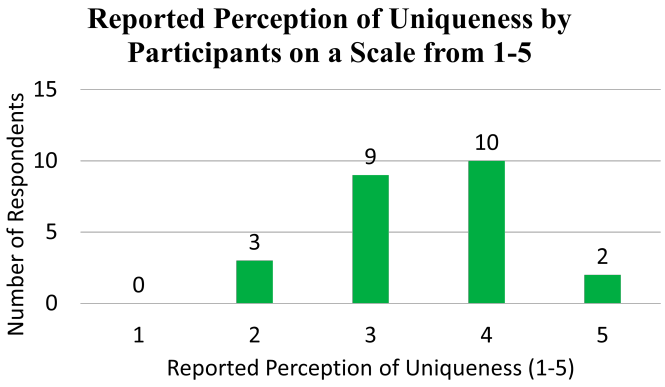


Fig. 13. Graph counting how the 26 participants who logged in successfully rated the uniqueness of the Automatic Feedback System on scale from 1 to 5.

Overall, students responded well to the Automatic Feedback System with a mean of 3.4 (see Fig. 14). Despite a low self-reported interest in and perception of the Automatic Feedback System's cybersecurity topics, this above-average response indicates an overall success in the program. Only two students indicated a below-average response, while one student indicated being highly satisfied with the experience. This success may seem surprising, but given the positive reception of the user-friendliness, educational value, uniqueness, as well as the strong implementation of the command line navigation topics, it is reasonable to say that students found many elements of the Automatic Feedback System appealing. Once the changes to the program described in Section V are made, this value can be further improved.

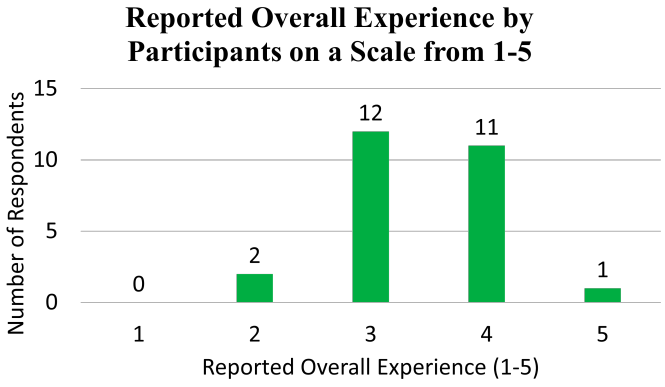


Fig. 14. Graph counting how the 26 participants who logged in successfully reported their overall experience with the Automatic Feedback System on a scale from 1 to 5.

In addition to the numerical feedback, participants were offered the chance to provide comments about their experiences. Notably, there were conflicting opinions on the implementation of the feedback. One participant noted that the diagnostic test provides suggestions for commands seemingly out of context, which "[doesn't help you to] pick up much... knowledge... an explanation on what [the commands] are used for [would be] helpful." However, other participants stated that these brief notes that "show you what the commands do and how to execute them" were helpful. The mixed response to the feedback itself matches greatly with the response regarding the implementation of various topics. It seems that this may be an issue of the quality of the implementation of topics and should include topic explanations in addition to practicing platform and that an easier-to-understand description of the commands is critical to ensure this software is beneficial to various users.

## V. Conclusion and Future Plans

The prototype of the Automatic Feedback System was successfully completed and tested in an introductory computer science class at Marshall University. Survey results indicate that students were highly satisfied with the concept of an Automatic Feedback System but were not interested in using it for cybersecurity. This may have been impacted by the low number of references to cybersecurity topics. Consequently, an overhaul of the lecture content is needed before future testing can be conducted.

Additionally, the program was unattractive and a potential source of security issues, as seen by multiple warnings from antivirus software on participants' systems. Future iterations will investigate the Automatic Feedback System being hosted as a web application, which will enable the use of design-oriented frameworks including React, Bootstrap, and Angular.

Furthermore, participants' inability to register or log in to the program hindered testing and created inaccessibility. The issue was found to be due to the way in which the MongoDB database, which stores both account information and interaction data, was being accessed. To eliminate this issue, alternative account log-in solutions will be investigated, including those offered by Amazon Cognito.

The indicated interest in the prototype's future development from 77% of successful participants indicates a demand for this solution. 46% of successful participants indicated a response that equated to an actively improved educational experience. However, as 88% participants indicated that they felt the system taught command-line skills effectively, further implementation of the system is necessary to bring the system to an overall satisfactory level.

Moreover, the system should be improved to be used by users with disabilities because one of the typical limitations of many e-learning platforms and self-learning tools is the lack of such features [12]. Primarily, we want our program to be used by visually impaired users. Therefore, in future work, we will add voice features that assist visually impaired users [13] in using the system.

## VI. Acknowledgment

We would like to thank the Marshall University Research Foundation for funding the continued development of this

project with the Undergraduate Creative Discovery and Research Awards. We would also like to thank Dr. Wook-Sung Yoo for his guidance and support.

## REFERENCES

[1] F. B. of Investigation, "2020 internet crime report," Federal Bureau of Investigation, Tech. Rep., Jan. 2021. [Online]. Available: https://www.ic3.gov/Media/PDF/AnnualReport/2020_IC3Report.pdf

[2] D. J. Trump, "America's cybersecurity workforce," 2019, volume 84, executive order 13870.

[3] U. D. of Education, "Stem webinar: Cybersecurity education 101," 2020. [Online]. Available: https://www.ed.gov/sites/default/files/documents/stem/cybersecurity-slides.pdf

[4] Google, "Searching for computer science: Access and barriers in u.s. k-12 education," Google, Tech. Rep., 2015. [Online]. Available: https://services.google.com/fh/files/misc/searching-for-computer-science_report.pdf

[5] ——, "Images of computer science: Perceptions among students, parents, and educators in the u.s." Google, Tech. Rep., 2015. [Online]. Available: https://services.google.com/fh/files/misc/images-of-computer-science-report.pdf

[6] (2021, Oct) Free and low cost online cybersecurity learning. National Institute of Standards and Technology. [Online]. Available: https://www.nist.gov/itl/applied-cybersecurity/nice/resources/online-learning-content

[7] (2021, Dec) Khan academy homepage. Khan Academy. [Online]. Available: https://www.khanacademy.org/

[8] (2021, Dec) Freecodecamp homepage. FreeCodeCamp. [Online]. Available: https://www.freecodecamp.org/

[9] (2021, Dec) Codecademy homepage. Codecademy. [Online]. Available: https://www.codecademy.com/

[10] (2022, April) Amazon future engineer homepage. Amazon. [Online]. Available: https://www.amazonfutureengineer.com/

[11] R. Tourangeau, L. J. Rips, and K. Rasinski, *The psychology of survey response*. Cambridge: Cambridge University Press, 2000.

[12] M. Laabidi, M. Jemni, L. Jemni Ben Ayed, H. Ben Brahim, and A. Ben Jemaa, "Learning technologies for people with disabilities," *Journal of King Saud University - Computer and Information Sciences*, vol. 26, pp. 29–45, 2014.

[13] M. Batshaw, G. L. Nancy Roizen and, L. G. Anthony, L. W. Desch, L. Kenworthy, and L. Pellegrino, *Children with Disabilities*. Brookes Publishing, July 2012.