Pyo, a Chatbot Assistant for Introductory Programming Students

Conference Paper · November 2022

DOI: 10.1109/SIIE56031.2022.9982349

4 authors:

Gustavo Carreira

1 PUBLICATION **20** CITATIONS

SEE PROFILE

Leonardo Soares

Federal Institute for Education Science and Technology of Pernambuco

37 PUBLICATIONS **180** CITATIONS

SEE PROFILE

Antonio Jose Mendes

University of Coimbra

164 PUBLICATIONS **2,456** CITATIONS

SEE PROFILE

Hugo Gonçalo Oliveira

University of Coimbra

167 PUBLICATIONS **1,490** CITATIONS

SEE PROFILE

Pyo, a Chatbot Assistant for Introductory Programming Students

Gustavo Carreira
CISUC, DEI
University of Coimbra
Coimbra, Portugal
gusmcarreira@hotmail.com

Leonardo Silva
CISUC, DEI
University of Coimbra
Coimbra, Portugal
lses@dei.uc.pt

António José Mendes
CISUC, DEI
University of Coimbra
Coimbra, Portugal
toze@dei.uc.pt

Hugo Gonçalo Oliveira
CISUC, DEI
University of Coimbra
Coimbra, Portugal
hroliv@dei.uc.pt

2022 International Symposium on Computers in Education (SIIE) | 978-1-6654-7464-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/SIIE56031.2022.9982349

***Abstract*—As a result of the technological advancements in our society, the number of students enrolling in Computer Science and related courses continues to rise. Programming is regarded as a difficult subject to learn, and as the size of classes and the number of students enrolled in online courses continues to increase, providing individualized support to determine where the difficulties of a student lie may appear impossible, resulting in frustration, lack of motivation, and an increase in dropout rates. Chatbots can mitigate this problem as it can provide simultaneous support to multiple students 24 hours a day, seven days a week, and by allowing introverted students to express questions more comfortably. This paper proposes the implementation of a chatbot denominated Pyo aiming to provide assistance to novice programmers. It was evaluated how students enrolled in an online introductory programming course interacted with Pyo in order to draw conclusions about the effectiveness of the chatbot's features. The results indicate that Pyo facilitates student learning; however, more straightforward explanations of its functionalities were requested.**

***Index Terms*—chatbot, conversational agent, artificial intelligence, pattern-matching, Rasa, introductory programming, Python**

I. Introduction

With the accelerating pace of the technological evolution in our society, the search for individuals with related knowledge increases. In fact, the demand for professionals in computer and information science is projected to rise by 22% over the next decade [1].

Given this growth, it is not surprising that enrollment in the course of computer science has not only increased but, as stated by the British Computer Society, has had the biggest increase of any UK university subject in 2022 [2].

However, such high numbers are also accompanied by a high drop-out rate, as per stated by the British Higher Education Statistics Agency, with a percentage of 33 citing the difficulty of the course as the main cause [8].

According to previous researchers [4], [13], [19], introductory programming courses are notoriously challenging. Novices frequently demonstrate misconceptions and other obstacles that impede their ability to learn and progress [19], therefore, it is vital for professors to comprehend a student's faulty conceptualization, in order to assist them in achieving an accurate one.

Diverse factors contribute to the challenges encountered by students learning to program. Quian and Lehman [19] identify several, including the increase in cognitive load as the exercises become more complex. This can impose an obstacle for a novice who has yet to have the syntax and concepts of a language solidified and, therefore, may not be able to have the capacity to trace more complex programs [3]. Additionally, flawed mental models, meaning a wrong internal representation created to simulate the behavior of a program, are another source of difficulties. Ma et al. [15] discovered that, by the end of their first year, the majority of novice programming students held non-viable mental models and performed significantly worse than the minority of students who held correct models.

To identify the various causes of a student's issues, it is of tremendous benefit to provide individualized support to each student. However, that is not always possible due to the introversion of a student, size class, or the online format.

The evolution of intelligent tutoring systems, particularly chatbots, has prompted researchers to utilize this technology as a means to solve the problem of insufficient personalised support for inexperienced programmers [6], [9].

This work aims to make a contribution to the underdeveloped field of chatbot assistants in the domain of introductory programming. The chatbot, named Pyo, is a Portuguese chatbot developed to assist students in introductory programming, specifically learning the Python language. The intention behind the agent is that it contains the ability to assist students when their algorithm produces an error, as well as offer guidance on the exercise, and definition of concepts.

The following research question guided this investigation: *How do introductory programming students respond to having a chatbot as an assistant?*. An exploratory research was conducted to determine which features a chatbot, aiming to facilitate the learning of programming, should have, as well as to evaluate how users engage with it. The data from 22 students, enrolled in an online introductory programming

This work is funded by national funds through FCT, within the scope of the project CISUC (UID/CEC/00326/2020) and by European Social Fund, through the Regional Operational Program Centro 2020.

course, were analyzed in order to answer the research question.

The remainder of this work is structured as follows: Section II discusses chatbots in the education domain. Section III introduces the objectives and characteristics of the conversational agent in greater detail. The evaluation process is described in section IV, which is followed by the presentation and discussion of its results (section V). In section VI, the limitations of the evaluation are discussed. Section VII concludes with a brief summary of the conclusions drawn from the developed work and future considerations.

II. Literature Review

According to Dale [7] a chatbot is "any software application that engages in a dialog with a human using natural language". This technology can handle the queries of several students simultaneously, offering constant support, and comfort to students who may fear being judged.

Initially, chatbots relied on rudimentary pattern-matching techniques [7] but, as improvements were made in domains such as Natural Language Processing (NLP) and artificial intelligence, chatbots' ability to hold a conversation in both text and voice format improved substantially [16]. This progress became evident in the early 2010s with the rise of the well-known Big Four voice-driven assistants: Apple's Siri, Microsoft's Cortana, Amazon's Alexa and Google's new Assistant [7]. However, even though they have been at the vanguard of this technology, they are far from being the only ones, as the commercial interest of this technology continuous to increase [16].

A chatbot can serve numerous purposes, including chatting, amusement, question answering, booking management, and educational purposes [10]. In this section, an analysis on how chatbots have been used in the education domain is conducted.

When it comes to the education domain, it is common to see chatbots with a clear service focus assisting on Frequently Asked Questions (FAQs). The University of Murcia in Spain, for instance, has implemented Lola with the purpose of reducing the volume of student enrolment-related questions during enrolment periods [17].

However, there is a significant increase in the number of chatbots whose aim focuses on the learning aspect of a specific topic. Used for both informal and formal education, these conversational agents serve the purpose of interacting and helping the students as a human educator would [18].

Researchers have discovered that engaging with chatbots increases students' interest in learning, which has further prompted the creation of educational chatbots [12]. Recent examples whose objective is knowledge acquisition include the work of Shorey et al. [20], where a virtual agent was created to imitate a patient, in order to teach nursing students how to speak more effectively, and ChatBot [5], a conversational agent developed with the purpose of teaching Chinese.

The search for relevant literature on chatbots in the field of introductory programming did not provide a plethora of results but did uncover some relatively recent work. Python-Bot [6], for instance, was implemented as a standalone application to teach the Python programming language. The bot comprises three different functionalities, explanation of concepts, meeting scheduling with a human educator, and predefined concept-training questions.

Edubot [22] is an example of another chatbot implemented for this domain, which was created to assist students beginning their programming journey with MATLAB. The chatbot provides responses to queries regarding the fundamentals of this language.

Similarly, Lectures' Apprentice [11] provides definitions to multiple introductory concepts of the Java programming language, as well as give out examples of exercises and its solutions.

According to the analysis, chatbots tend to be designed for question-answering purposes in which the interactions are initiated by the student. While one of Pyos' features aims to answer questions on introductory concepts, it also strives to be proactive by asking student whether they need assistance when committing an error or if they want help on the exercise after a certain amount of time is spent on the same subject.

III. Pyo Pedagogical Design

The objectives of this study include the creation of a chatbot (Pyo) using Rasa[1] and the Python programming language. Rasa is an open-source framework based on machine learning that facilitates the process of building complex chatbots and their integration on websites.

Pyo's pedagogical approach was grounded within a Vygotskian perspective of the zone of proximal development [21]. It is seen as a support for students to perform tasks they would have difficulties doing independently. For example, novice programmers often struggle planning their program development, and without proper support their academic success could be impacted [14]. Thus, the features of the Pyo provide the support to overcome known barriers in the programming learning process, such as with problem-solving actions and error correction.

The chatbot consists of three primary aspects: exercise assistance, error guidance, and concept definitions.

1) Exercise Assistance: The exercise assistance feature prompts the student to arrange, in natural language, the skeletal components of a possible answer. This support is intended to provide the student with an understanding of the concepts involved in the exercise without revealing the precise Python syntax required to implement them.

This functionality is triggered. This trigger is sent two minutes after the student has begun an exercise involving the creation of an algorithm, at which point it sends the student the message shown at the top of figure 1(a) in light blue. The requirement that a minimum of two minutes must elapse before a student may request assistance on an exercise is meant to encourage the student to first consider and analyze the exercise.

When a student requests help, they are asked to organize a scrambled list of the constituent parts of a possible answer. In

[1] https://rasa.com

figure 1(b) the buttons at the bottom of the image represent the answer's components. In the end, whether the student is correct or not, the chatbot provides them with the correct answer so they can visualize it as often as necessary (bottom of fig. 1(c)).

The translation of code into natural language steps is conducted using Pythons' ast[2] library. This module enables the transformation of Python code into abstract syntax trees, with each node representing a distinct Python class. This library enables the analysis of code by iterating the tree to determine, for example, whether a conditional, variable, repetition, or function is being declared.

2) Error Guidance: When beginning the process of learning a subject, it is normal to make several mistakes; programming is no different. Beginner programmers typically produce a large number of flawed algorithms at first, and if their conceptual understanding is not solid, moving on to more complex tasks that require them to produce more complex solutions, can lead to frustration and a lack of motivation.

The error guidance feature is intended to assist students in identifying and correcting syntax errors in their programs without providing the exact solution. Specifically, when a student submits code containing an error, as highlighted in red in fig. 2, the student will receive a message requesting assistance (top fig. 2 in light blue).

The guidance consists in a dialogue composed of predetermined multiple-choice questions designed to lead the student to the solution (bottom of fig. 2). The student is only required to select the answer they believe is correct; the chatbot will then send a formal explanation of the error. Multiple-choice questions provide examples of the given error.

3) Concept Definitions: As indicated in the introduction, an increase in cognitive load as tasks become more complex is one of the causes of difficulty. This can be problematic for novice programmers who lack a firm grasp of the concepts and, as a result, forget how to implement them. Each of the topics integrated into the exercises is defined by Pyo. In fig.3, a student requests the syntax of an if statement (in green), and the chatbot responds with the answer (in blue) and an example (in orange).

Occasionally, Pyo is not be able to predict the next action, or the student is unsure of what to ask. In light of this, the chatbot also provides a list of suggested questions that are retrieved using a custom function using the ast library. By traversing each node of the code's abstract syntax tree, we can determine which concepts are being implemented and map it to a natural language question (fig. 4).

IV. Evaluation

This exploratory research aimed at identifying which features a chatbot designed to promote the learning of programming should have, as well as evaluating how the users interact with it. Thus, the following research question: *How do introductory programming students respond to having a chatbot as an assistant?*.

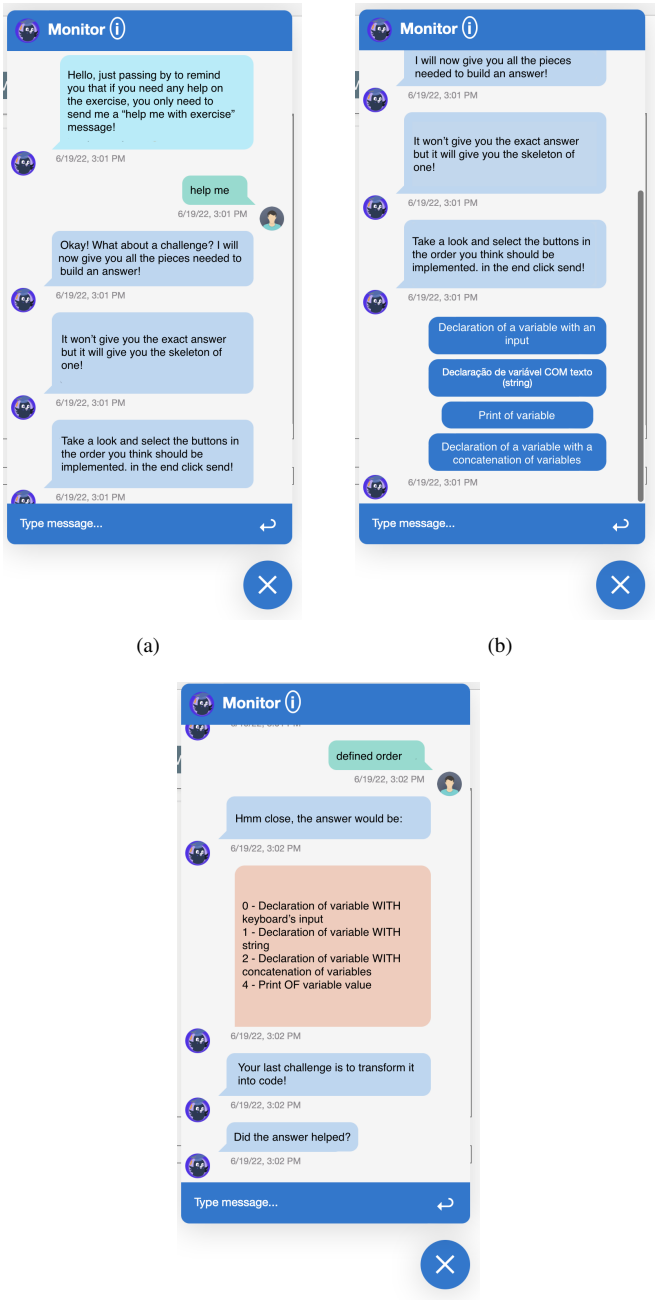

Fig. 1. Conversation where a student asks for help with the exercise [3]

A. Pedagogical context

A mid-sized public university in an urban area of Brazil offered an online programming course for beginners. The professor was the second author of this study, and the two-month, 40-hour course was aimed at people with minimal programming experience. The syllabus was based on the ACM Curricula for fundamental programming concepts and

[2]https://docs.python.org/3/library/ast.html

[3]Images have been translated for the purpose of this paper. Original conversations are held in Portuguese.

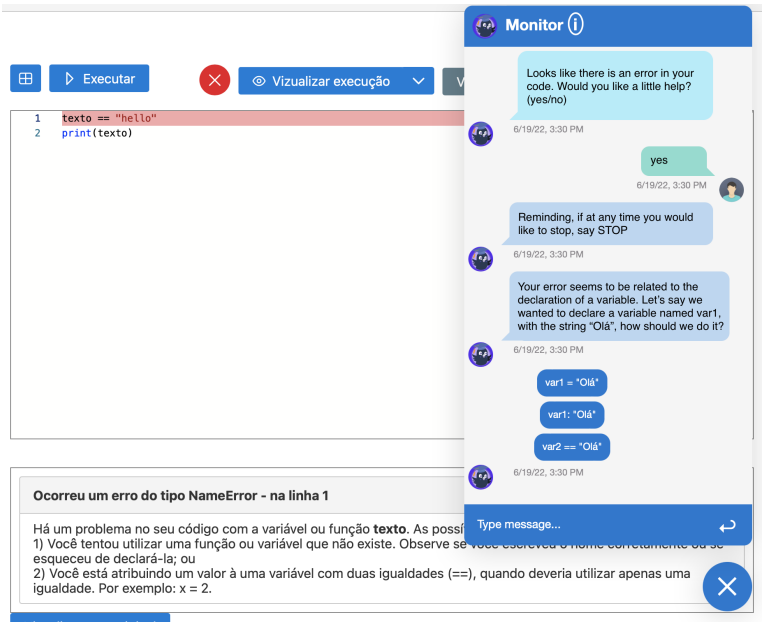

Fig. 2. Conversation where a student asks for help with an error.

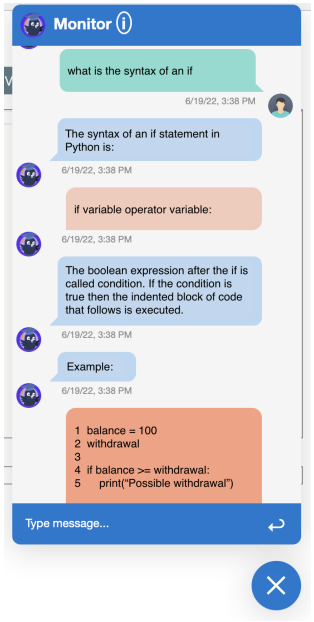

Fig. 3. Conversation in which the student asks help with the syntax of an if statement.

included variables, input/output, conditionals, loop structures, and functions[4], using the Python language.

Students had access to recorded video materials about the programming content at any time and location. The programming practice was conducted on a self-developed programming platform that incorporated Pyo. This platform includes instructional materials, programming exercises, and a code editor. The editor resembles conventional editors such

[4]https://www.acm.org/education/curricula-recommendations

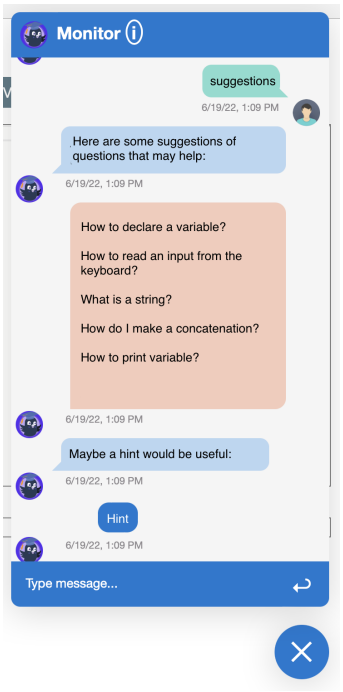

Fig. 4. Conversation where a student asks for suggestions of questions related to the exercise they are in.

as Eclipse and Netbeans, in which the student can write code and receive feedback from the Python interpreter regarding any errors or program outputs.

A total of 70 exercises (13 from variables, 24 from conditionals, 21 from loops, and 12 from functions) were provided. To promote reproducibility, the exercises have been translated into English and made available in a repository for Open Science.

The students are left to learn at their own pace with a new subject of the course being made available approximately every 2 weeks. Worthy to note that students are a part of a WhatsApp group allowing them to interact with their peers and the professor. Students were also informed about the data collection procedure and notified that the data would be used for investigative purposes only and not shared with their peers.

The conversation agent comes in as an assistant to the professor, offering similar help an educator would provide. Help with the exercise and errors, as well as definitions of key concept, are its main characteristics. There are a number of benefits associated with the agent, including its 24-hour availability, which enables students to receive instant assistance whenever they need it, their ability to assist multiple students simultaneously, and their ability to support shy students who would not otherwise ask their questions.

The interactions between the agent and its users are conducted in Portuguese as it is the language of the platform and its students. An important aspect, to be able to closely analyse the interactions with the chatbot, is the store of the conversations on a database, including informations sent from the application to Pyo, such as error messages and the

solutions of the exercises.

B. Data collection and analysis

Throughout the duration of the course, the chatbot's interactions were analyzed on a daily basis in order to promptly correct any errors or make modifications to its features based on their usage. Additionally, a questionnaire was created to collect the students' feedback and opinions regarding the chatbot and its features.

Ten of the 65 enrolled students completed more than 50% of the course, with just one student completing more than 90%, similarly with previous RESPE's previous courses. Only nine students responded to the survey that was prompted three weeks before the course's conclusion.

Students were made aware of the chatbot and its experimental nature. Additionally they were warned of the storage and analysis of their conversations.

It is important to note that Pyo was present in every question, regardless of whether it required the development of an algorithm, but exercise assistance and error guidance were only available for programming questions.

V. Results and Discussion

As mentioned weekly analysis was undertaken to deterr which percentage of the active students was engaging with (figure 5). Due to an error in conversation storage, week 1 been removed from further analysis.

The progression of the proportion of active students engaged with Pyo at least once is depicted in fig 5. The sec week of the course had the highest percentage of stud interacting with the agent, with 67% of students interac with the agent.

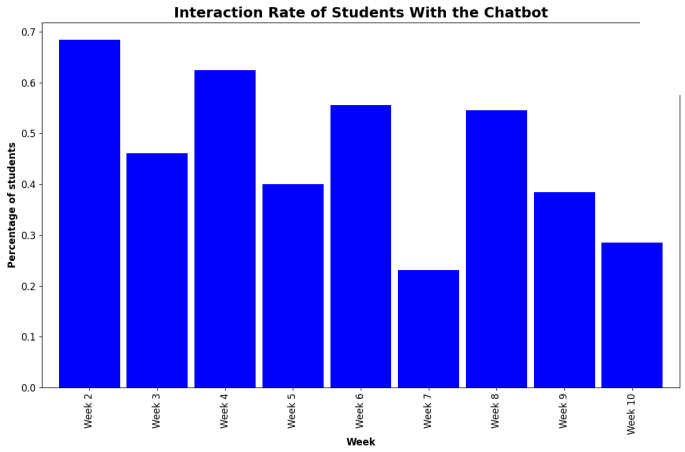

Fig. 5. Analysis of the percentage of active students that engaged with the chatbot

Despite the availability of Pyo, on average, only 46% of the students made at least one interaction. Analyzing students' submissions, we observed that in some situations the student had difficulties but did not use the provided support.

Two possibles explanations on why the percentage is not higher were raised. The students have varying levels of programming knowledge, and those with previous understanding from other programming languages might not needed the chatbot's support. One other reason is the WhatsApp group, as it enables the students to exchange difficulties with their peers. A proportion of the students resort to the group where, most of the times, they receive the direct solution from their peers. This was also concluded from the questionnaire made to the top 3 students that had more interactions with Pyo, as 2 completed agreed and 1 was indifferent to the statement I rather interact with my pears/professor than with the chatbot. Even though this is known, the agent is not intended to provide the answer, but rather to guide them towards it, however this aspect was what lead to the development of the hints feature. Our data limits further conclusions about the students' lack of interaction, an aspect that will be investigated in the future.

The percentages follow the increases and decreases in the number of active students throughout the weeks (figure 6), except for week 4 that saw a decrease in number of active students but and increase in the percentage of students interacting with Pyo.

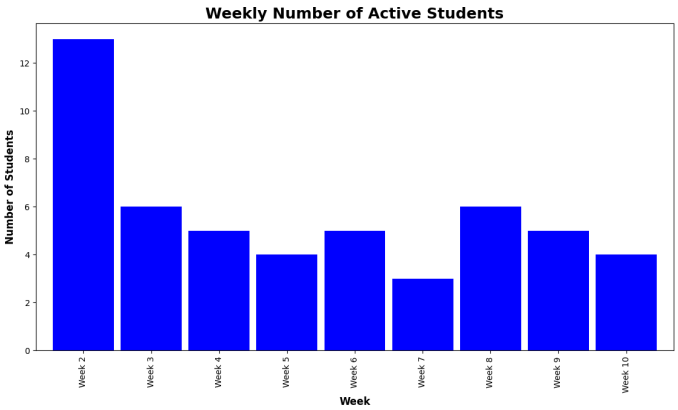

Fig. 6. Weekly evaluation of the number of active students.

In addition to general engagement, a weekly usage analysis of the functionalities was conducted, but no conclusions could be drawn because the data did not appear to follow any interesting pattern. Therefore, table I depicts the absolute values of the number of times a student utilized a particular feature throughout the whole course. The most frequently utilized feature was the assistance on the exercise, followed by the explanation of concepts, and finally, error guidance.

TABLE I
Functionalities utilization frequency.

Functionality	Number of Times Used
Concept Definition	53
Error Guidance	33
Exercise Assistance	61

The questionnaire allowed us to determine how students felt about various aspects of Pyo. On the one hand, the chatbot was deemed user-friendly, realistic, possessing an engaging personality, perceptive regarding its features, adept at handling misunderstandings, and helpful, thereby achieving the primary objective of "assisting introductory programming students".

The students, on the other hand, viewed the chatbot as neither accurate nor inaccurate and favored seeking assistance from their peers and professor. Possible causes include the presence of critical errors early on and the absence of individualized assistance based on the student's knowledge level.

Nevertheless, the analysis demonstrated Pyo's usefulness and worth, as the majority of questionnaire responses were positive. It was also of great assistance in identifying areas requiring improvement for future work, as described in the final chapter.

It is intended to continue the current procedure for future endeavors. Future research would focus on training a Question Answering (QA) model specifically for the programming domain in order to automate the agent's answer-providing capabilities with precision.

In addition, the students' participation in the course should be analyzed to determine ways to personalize Pyo's support.

VI. Limitations

A significant limitation of the evaluation was the quantity of questionnaire responses. Only twelve of the 65 students enrolled in the course reached the final week, and only nine completed the questionnaire, preventing us from drawing conclusions that are representative of the entire population, in this case the students. Even though this data limit impacts our capability to generalize our findings, it indicate avenues of research that can be carried out to gather more details.

VII. Conclusion

A chatbot can be a useful tool to work with teachers in assisting students. They offer benefits such as immediate response and feedback along with the 24 hour availability. It can be used to proportionate the individualized support a professor may not be able to give to every student. Pyo was implemented to not only answering the queries of students but also to be proactive by offering help on the exercise and on mistakes in their algorithms. Even though the evaluation results were not optimal, as more engagement was expected, it was very beneficial in terms of improvements in the implementation. On the other hand, the questionnaire allowed us to see that some students do, in fact, see the chatbot as useful on their programming learning journey.

References

[1] Computer and information research scientists : Occupational outlook handbook. https://www.bls.gov/ooh/computer-and-information-technology/computer-and-information-research-scientists.htm, Apr 2022.

[2] Record numbers have applied for UK Computer Science degrees this year. https://www.bcs.org/articles-opinion-and-research/record-numbers-have-applied-for-uk-computer-science-degrees-this-year/, Feb 2022.

[3] Piraye Bayman and Richard E. Mayer. Using conceptual models to teach basic computer programming. *Journal of Educational Psychology*, 80:291–298, 1988.

[4] Anders Berglund and Raymond Lister. Introductory programming and the didactic triangle. In *Proceedings of the Twelfth Australasian Conference on Computing Education - Volume 103*, ACE '10, page 35–44, AUS, 2010. Australian Computer Society, Inc.

[5] Hsiu-Ling Chen, Gracia Vicki Widarso, and Hendri Sutrisno. A chatbot for learning chinese: Learning achievement and technology acceptance. *Journal of Educational Computing Research*, 58(6):1161–1189, 2020.

[6] Okonkwo Chinedu and Abejide Ade-Ibijola. Python-bot: A chatbot for teaching python programming. *Engineering Letters*, 29:25–34, 02 2021.

[7] Robert Dale. The return of the chatbots. *Natural Language Engineering*, 22(5):811–817, 2016.

[8] Karl Flinders. Computer science undergraduates most likely to drop out. https://www.computerweekly.com/news/252467745/Computer-science-undergraduates-most-likely-to-drop-out, Aug 2019.

[9] Sebastian Hobert. Say hello to 'coding tutor'! design and evaluation of a chatbot-based learning system supporting students to learn to program. In *ICIS 2019 Proceedings. 9*, 2019.

[10] Gwo-Jen Hwang and Ching-Yi Chang. A review of opportunities and challenges of chatbots in education. *Interactive Learning Environments*, 07 2021.

[11] Mohammed Ismail and Abejide Ade-Ibijola. Lecturer's apprentice: A chatbot for assisting novice programmers. In *2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, pages 1–8, 2019.

[12] W. Lewis Johnson. Pedagogical agent research at carte. *AI Mag.*, 22(4):85–94, oct 2001.

[13] Einari Kurvinen, Niko Hellgren, Erkki Kaila, Mikko-Jussi Laakso, and Tapio Salakoski. Programming misconceptions in an introductory level programming course exam. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '16, page 308–313, New York, NY, USA, 2016. Association for Computing Machinery.

[14] Dastyni Loksa and Amy J. Ko. The role of self-regulation in programming problem solving process and success. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*, ICER '16, page 83–91, New York, NY, USA, 2016. Association for Computing Machinery.

[15] Linxiao Ma, John Ferguson, Marc Roper, and Murray Wood. Investigating the viability of mental models held by novice programmers. In *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '07, page 499–503, New York, NY, USA, 2007. Association for Computing Machinery.

[16] Santiago Melián-González, Desiderio Taño, and Jacques Bulchand-Gidumal. Predicting the intentions to use chatbots for travel and tourism. *Current Issues in Tourism*, 24:1–19, 12 2019.

[17] Andrés Pedreño Muñoz. Lola, el chatbot inteligente que triunfa entre los estudiantes. https://elpais.com/retina/2018/11/30/innovacion/1543580663_865121.html, 2018.

[18] José Quiroga Pérez, Thanasis Daradoumis, and Joan Manuel Marquès Puig. Rediscovering the use of chatbots in education: A systematic literature review. *Computer Applications in Engineering Education*, 28(6):1549–1565, 2020.

[19] Yizhou Qian and James Lehman. Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Trans. Comput. Educ.*, 18(1), oct 2017.

[20] Shefaly Shorey, Emily Neo Kim Ang, John Yin Gwee Yap, Esperanza Debby Ng, Siew Tiang Lau, and CheeKong K. Chui. A virtual counseling application using artificial intelligence for communication skills training in nursing education: Development study. *Journal of Medical Internet Research*, 21, 2019.

[21] Rachel R Van Der Stuyf. Scaffolding as a teaching strategy. *Adolescent learning and development*, 52(3):5–18, 2002.

[22] Matthew Verleger and James Pembridge. A pilot study integrating an ai-driven chatbot in an introductory programming course. In *2018 IEEE Frontiers in Education Conference (FIE)*, pages 1–4, 2018.