IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 9, June 2025

Impact Factor: 7.67

GitHub Feature Use and Programming Skill Development among BSICT Students

Dr. Jessica Rose E. Fernandez[1], Christian George Santillan[2], Lordwaren Lariba Puntos[3]

Faculty, College of Engineering and Information Technology, Surigao City, Philippines [1]

BSIT Students, College of Engineering and Information Technology, Surigao City, Philippines[2,3]

Abstract: *This study examined the relationship between GitHub feature utilization and the development skills of third- and fourth-year Bachelor of Science in Information and Communication Technology (BSICT) students at SNSU. The researchers focused on repository management skills, collaboration features, and version control practices, analyzing their impact on project organization, code management confidence, and collaboration readiness. A descriptive-correlational quantitative design was employed, involving 15 participants (14 males, 1 female) who completed a structured survey using a four-point Likert scale. Data were analyzed with Jamovi, employing descriptive statistics, Pearson correlations, and regression analyses. Results revealed moderate proficiency across all GitHub feature domains, with mean scores of 3.10 for repository management skills, 3.15 for collaboration features, and 3.07 for version control practices (SD range: 0.497–0.555). Participants demonstrated the highest confidence in collaborative tasks, such as coordinating group projects (M = 3.20), but showed lower proficiency in error recovery and task monitoring (M = 2.93–3.00). The study recommended structured GitHub training, hands-on pedagogical approaches, and peer learning networks to address challenges like uneven engagement. This research provided a foundation for optimizing GitHub's educational impact in ICT curricula, aligning with industry demands.*

Keywords: ICT Students, GitHub Utilization, Programming Skills, Quantitative Study

I. INTRODUCTION

The integration of GitHub into educational settings has gained significant attention due to its potential to enhance software development skills among students. As a widely used platform for version control, collaboration, and project management, GitHub offers a range of features that mirror real-world software development practices. This study aims to examine how third-year and fourth-year Bachelor of Science in Information and Communication Technology (BSICT) students at SNSU utilize GitHub features and how this utilization correlates with their development skills. By focusing on repository management, collaboration tools, and version control practices, the research seeks to uncover insights that can inform educational strategies to better prepare students for industry demands.

Prior research highlights the transformative role of GitHub in education. For instance, research [1] demonstrated that integrating GitHub repositories into programming courses fosters collaborative learning and self-directed skill development, with students reporting improvements in teamwork, problem-solving, and familiarity with industry-standard tools like pull requests and automated testing. Similarly, GitHub's 2020 Classroom Report revealed that 40% of early-career developers use browser-based integrated development environments (IDEs) to streamline workflows, while educators increasingly rely on GitHub Classroom to automate grading and track student contributions in group projects [3]. These findings are reinforced by studies showing that GitHub's collaborative features, such as issue tracking and code reviews, significantly enhance technical proficiency and teamwork in software engineering courses [4].

Despite these advancements, gaps remain in understanding the specific relationship between GitHub feature utilization and the development skills of ICT students. While existing studies have explored general benefits, there is limited research on how different aspects of GitHub usage—such as repository management or version control habits—impact project organization, code management confidence, and collaboration readiness. For example, an author [1] identified

DOI: 10.48175/IJARSCT-28254

challenges such as uneven student engagement and reliance on lecturer-provided codebases, which may dilute the platform's potential for fostering independent problem-solving. Furthermore, most studies focus on general populations or advanced developers rather than undergraduate ICT students in their formative years [3] [4]. This study seeks to address these gaps by providing a detailed analysis of how third-year and fourth-year BSICT students engage with GitHub features and how this engagement influences their development capabilities.

To bridge this knowledge gap, this study proposed a quantitative approach to analyze the relationship between GitHub feature utilization and development skills among third-year and fourth-year BSICT students at SNSU. By examining specific metrics such as commit habits, pull request activity, and branch management proficiency, the research aimed to provide actionable insights for educators and curriculum designers. Drawing on recommendations from an author [1], the study emphasized early integration of GitHub into curricula to reduce learning curves and encouraged structured pair programming to strengthen collaboration. The findings contributed to improving pedagogical practices by aligning them more closely with industry standards, ultimately enhancing students' readiness for professional software development roles through technical expertise and collaborative competencies.

This study aimed to examine the extent to which third- and fourth-year BSICT students at SNSU utilized key GitHub features (repository management, collaboration tools, and version control practices); analyze the relationship between GitHub feature utilization and core development skills (project organization, code management confidence, and collaboration readiness); determine which specific GitHub features contributed most significantly to enhancing students' development skills; identify challenges faced by students in using GitHub effectively for educational purposes; and propose recommendations for integrating GitHub into ICT curricula to better prepare students for industry demands and best practices.

This study was guided by a conceptual framework that examined the relationship between GitHub feature utilization and the development skills of third-year and fourth-year BSICT students at SNSU. This framework posited that effective use of GitHub features significantly influenced students' technical and collaborative competencies, as illustrated in Figure 1. By analysing this relationship, the study aimed to provide actionable insights for improving educational practices and aligning them with best practices or industry standards.

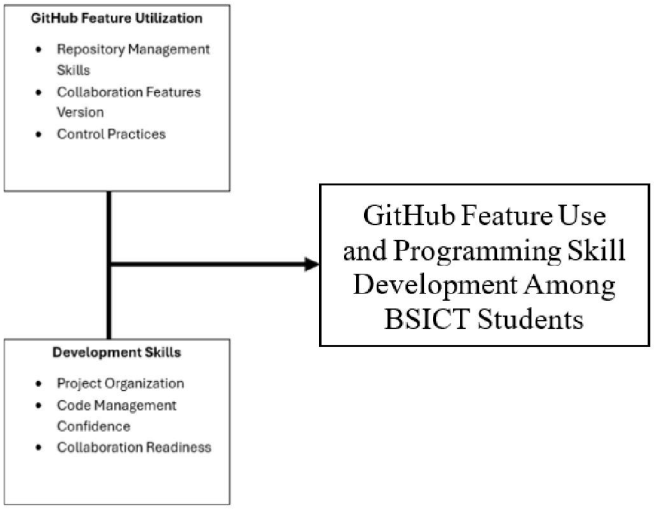

GitHub Feature Utilization
Repository Management Skills
Collaboration Features Version
Control Practices
GitHub Feature Use and Programming Skill Development Among BSICT Students
Development Skills
Project Organization
Code Management Confidence
Collaboration Readiness

Fig. 1 Aconceptual framework of the study

Fig. 1 illustrated the key elements of this framework. The independent variable, GitHub feature utilization, encompassed repository management skills, collaboration features, and version control practices. The dependent variable, development skills, included project organization, code management confidence, and collaboration readiness.

The quantitative studywas grounded in social, cognitive, and constructivist learning theories. The study drew upon Social Learning Theory [28], which posited that learning occurs through observation, modeling, and social interaction, to investigate how GitHub's collaborative features (pull requests, code reviews) fostered peer learning and enhanced collaboration readiness (e.g., a mean of 3.20 for coordinating group tasks), developing skills crucial for IT project management and teamwork. Self-Efficacy Theory [20] informed the examination of how GitHub's version control features (commits, branching) impacted students' code management confidence (mean of 3.13 for managing code changes), addressing areas of lower confidence such as error recovery (mean of 3.00). Finally, Constructivist Learning Theory [29], emphasizing active learning and hands-on experience, provided a framework for exploring how GitHub's practical application in repository management and version control supported the development of project organization skills (mean of 3.10 for repository management skills) through experiential learning. The study thus investigated the interplay of these theoretical perspectives to understand GitHub's impact on BSICT students' development skills within an ICT context.

II. RESEARCH METHOD

This chapter detailed the research methodology employed to investigate the relationship between GitHub feature utilization and development skills among BSICT students at SNSU, including the study design, participants, data collection, and analysis.

Research Design

This study employed a descriptive-correlational quantitative research design to investigate the relationship between GitHub feature utilization (independent variable) and development skills (dependent variable) among third-year and fourth-year BSICT students. This design was selected to identify patterns and measure association strength between variables without experimental manipulation, aligning with the study's objective of analyzing how GitHub usage correlates with skill development in educational contexts.

Participants and Sampling Method

The study involved 15 participants comprising third-year (n=9) and fourth-year (n=6) BSICT students at SNSU. Participants included 14 males and 1 female, with ages distributed as follows: 21-22 years (n=9), 23-24 years (n=3), and 25+ years (n=2).A purposive sampling method was employed to intentionally select students from these year levels, ensuring adequate exposure to GitHub features through academic coursework. This approach excluded first- and second-year students who may lack sufficient experience, focusing on participants capable of providing meaningful insights into GitHub's impact on development skills.

Data Collection Methods

Data were collected using a structured survey questionnaire assessing students' self-reported proficiency across three GitHub feature domains and corresponding development skills:Repository Management Skills (9 items), Collaboration Features (9 items), Version Control Practices (9 items), Dependent Variables (Development Skills), Project Organization, Code Management Confidence, Collaboration Readiness. Each item was rated on a four-point Likert scale (1 = Strongly Disagree, 2 = Disagree, 3 = Agree, 4 = Strongly Agree). The survey required approximately 15-20 minutes to complete. Prior to participation, students were informed of the study's purpose and provided voluntary consent, with all responses maintained anonymously and confidentially in accordance with ethical guidelines.

Data Analysis Techniques

Statistical analyses were conducted using Jamovi (version 2.3) with α = 0.05. Analytical procedures included:Frequency distributions, measures of central tendency and variability, and Cronbach's alpha reliability coefficients for scale dimensions.Pearson correlation coefficients examining relationships between GitHub utilization domains and development skills outcomes.Independent samples t-tests comparing year-level differences, with Cohen's d effect sizes calculated for practical significance.Multiple linear regression identifying GitHub features most predictive of development skills, with assumption verification using Jamovi's diagnostic tools.All statistical analyses were conducted using Micrsoft Excel and Jamovi Statistical Open-Source Software. Results were interpreted within the study's limitations and the specialized BSICT student context.

III. RESULTS AND DISCUSSION

This chapter presented the findings from a survey of 15 third- and fourth-year BSICT students at SNSU, analyzing the relationship between GitHub feature utilization and development skills. Results were organized into respondent profiles, descriptive statistics, correlational, comparative, and regression analyses, with interpretations contextualized within the study's objectives.

Fig. 2 presented a bar graph illustrating the gender distribution of the 15 participants in the study. The graph clearly showed a significant gender imbalance. A substantial majority (14 out of 15) of the participants were male, while only one participant was female. This heavily skewed sample raised concerns about the generalizability of the study's findings to a broader population of BSICT students, which likely exhibited a more balanced gender distribution. The limited female representation may have introduced bias and limited the ability to draw conclusions about gender differences in GitHub utilization or development skills. Future research should strive for a more representative sample to ensure greater validity and generalizability of the results.

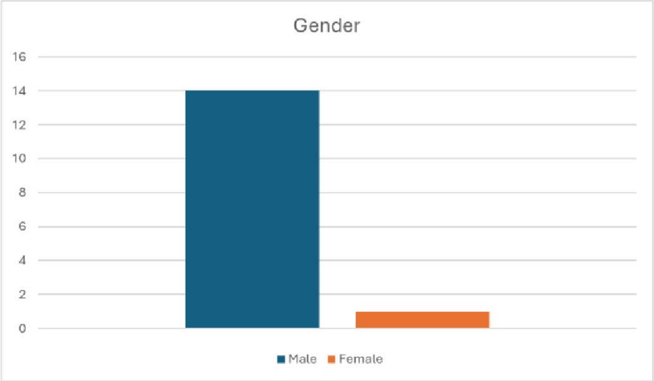


Fig.2A gender distribution of respondents

Fig. 3 displayed the age distribution of the 15 study participants. The majority of participants (10 out of 15) fell within the 21-22 age range.

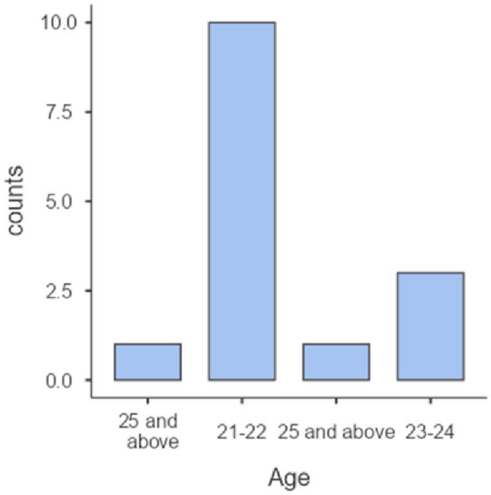


Fig.3Age Distribution of the Respondents

DOI: 10.48175/IJARSCT-28254

A smaller number of participants (3) were aged 23-24, and only a small number (2) were 25 or older. This age distribution suggested a relatively homogenous sample, primarily composed of younger students. The concentration of participants in a narrow age range may have limited the generalizability of findings to older BSICT students or those outside this age cohort. While this homogeneity might have reduced the influence of age as a confounding variable in the analysis, it also restricted the scope of the study's conclusions. Future studies could benefit from including a wider age range to enhance the generalizability of their results and explore potential age-related differences in GitHub usage and skill development.

Figure 4 showed the year level distribution of the 15 participants using a pie chart. The chart indicated that a larger proportion (60%) of the participants were fourth-year BSICT students, while 40% were in their third year. This imbalance in year levels might have influenced the results, as fourth-year students may have had more experience with software development tools and collaborative projects, potentially affecting their GitHub usage and skill levels. The unequal representation of year levels should have been considered when interpreting the findings, as it could have introduced bias and limited the generalizability of the results to a broader population of BSICT students with a more even distribution across year levels. Future research could benefit from a more balanced representation of year levels to provide a more comprehensive understanding of GitHub utilization across different stages of the BSICT program.

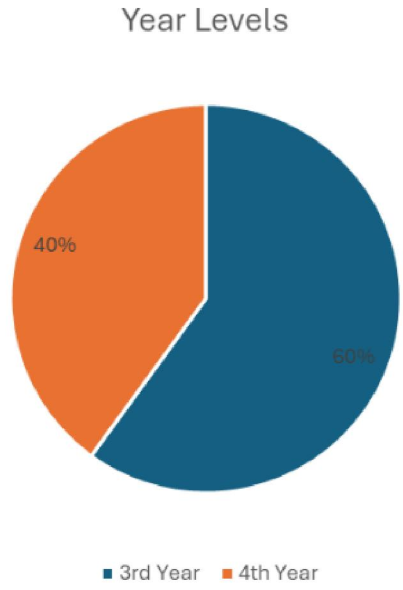
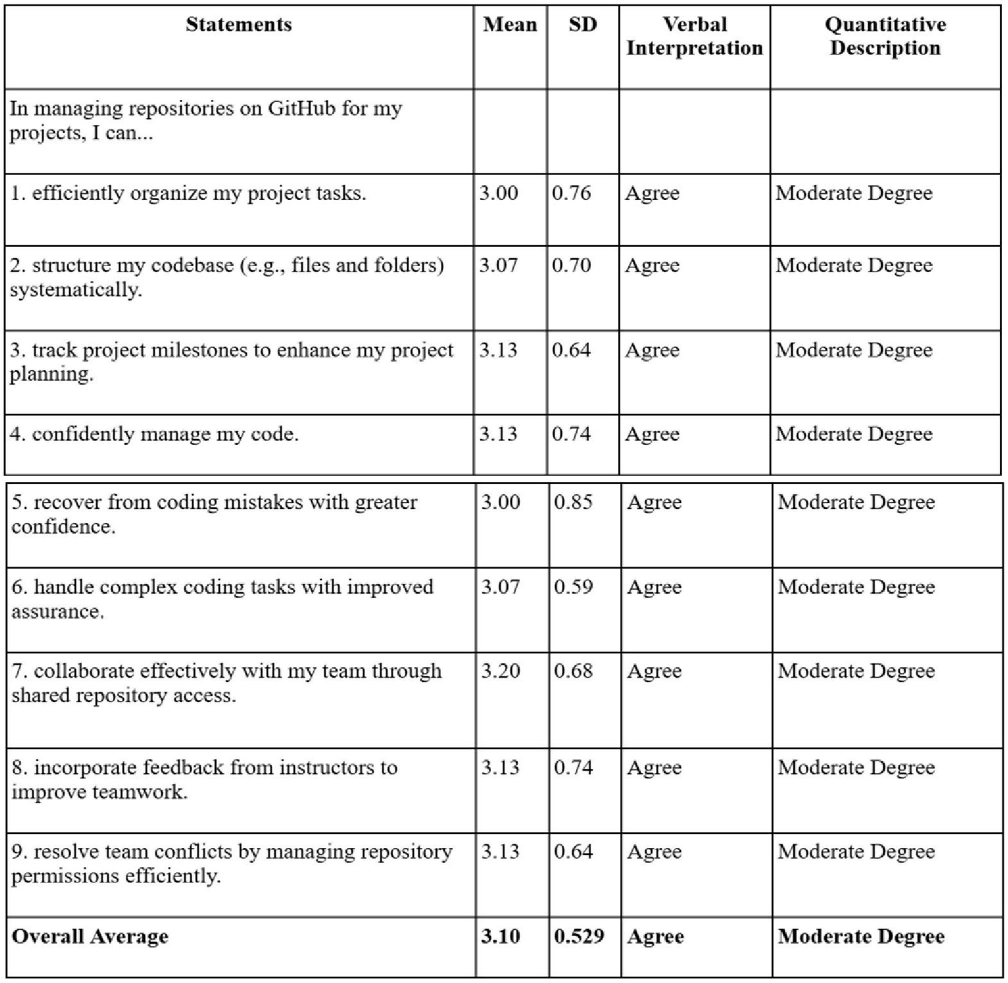


Fig.4Year level distribution

Analysis of GitHub Feature Utilization on Development Skills

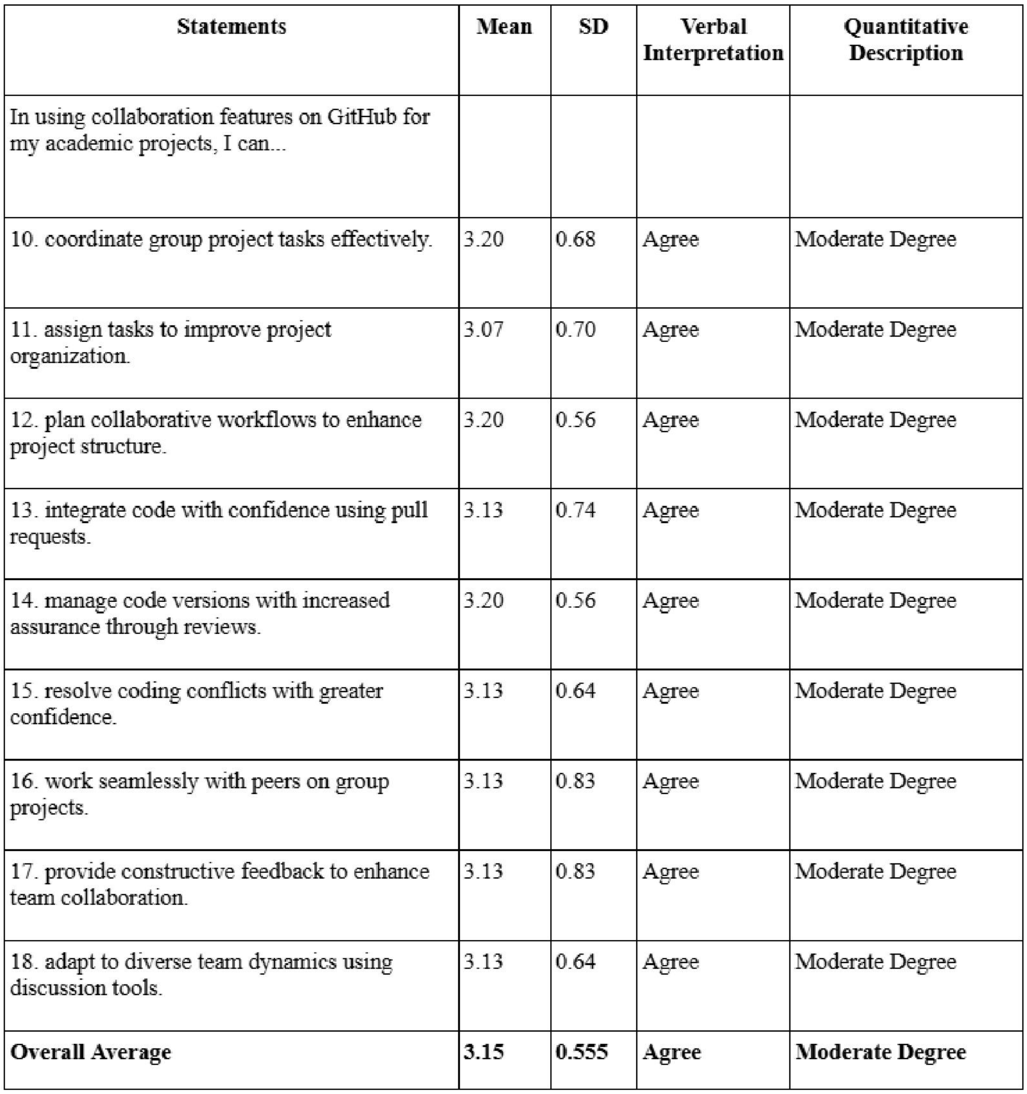
TABLE 1: The results of the survey assessing students' self-reported skills in repository management on GitHub



Statements	Mean	SD	Verbal Interpretation	Quantitative Description
In managing repositories on GitHub for my projects, I can...				
1. efficiently organize my project tasks.	3.00	0.76	Agree	Moderate Degree
2. structure my codebase (e.g., files and folders) systematically.	3.07	0.70	Agree	Moderate Degree
3. track project milestones to enhance my project planning.	3.13	0.64	Agree	Moderate Degree
4. confidently manage my code.	3.13	0.74	Agree	Moderate Degree
5. recover from coding mistakes with greater confidence.	3.00	0.85	Agree	Moderate Degree
6. handle complex coding tasks with improved assurance.	3.07	0.59	Agree	Moderate Degree
7. collaborate effectively with my team through shared repository access.	3.20	0.68	Agree	Moderate Degree
8. incorporate feedback from instructors to improve teamwork.	3.13	0.74	Agree	Moderate Degree
9. resolve team conflicts by managing repository permissions efficiently.	3.13	0.64	Agree	Moderate Degree
Overall Average	**3.10**	**0.529**	**Agree**	**Moderate Degree**

Table 1 presented the results of the survey assessing students' self-reported skills in repository management on GitHub. The data revealed a generally moderate level of proficiency across all aspects of repository management, as indicated by the mean scores ranging from 3.00 to 3.20 and consistently described as "Moderate Degree". While all mean scores were above the midpoint of the Likert scale (indicating agreement), none were significantly high, suggesting areas for improvement. Specifically, students reported moderate confidence in efficiently organizing project tasks, structuring their codebase systematically, tracking project milestones, confidently managing their code, recovering from coding mistakes, and handling complex coding tasks. Higher mean scores were observed for items related to collaboration (3.20 for collaborating effectively with a team and 3.13 for incorporating instructor feedback), indicating relatively stronger self-reported skills in these areas. The overall average mean score of 3.10 further supported the conclusion of a moderate level of proficiency in repository management skills. These findings highlighted the need for further development of students' skills in all areas of repository management, particularly those related to individual code management and error recovery, to enhance their overall effectiveness using GitHub for project development.

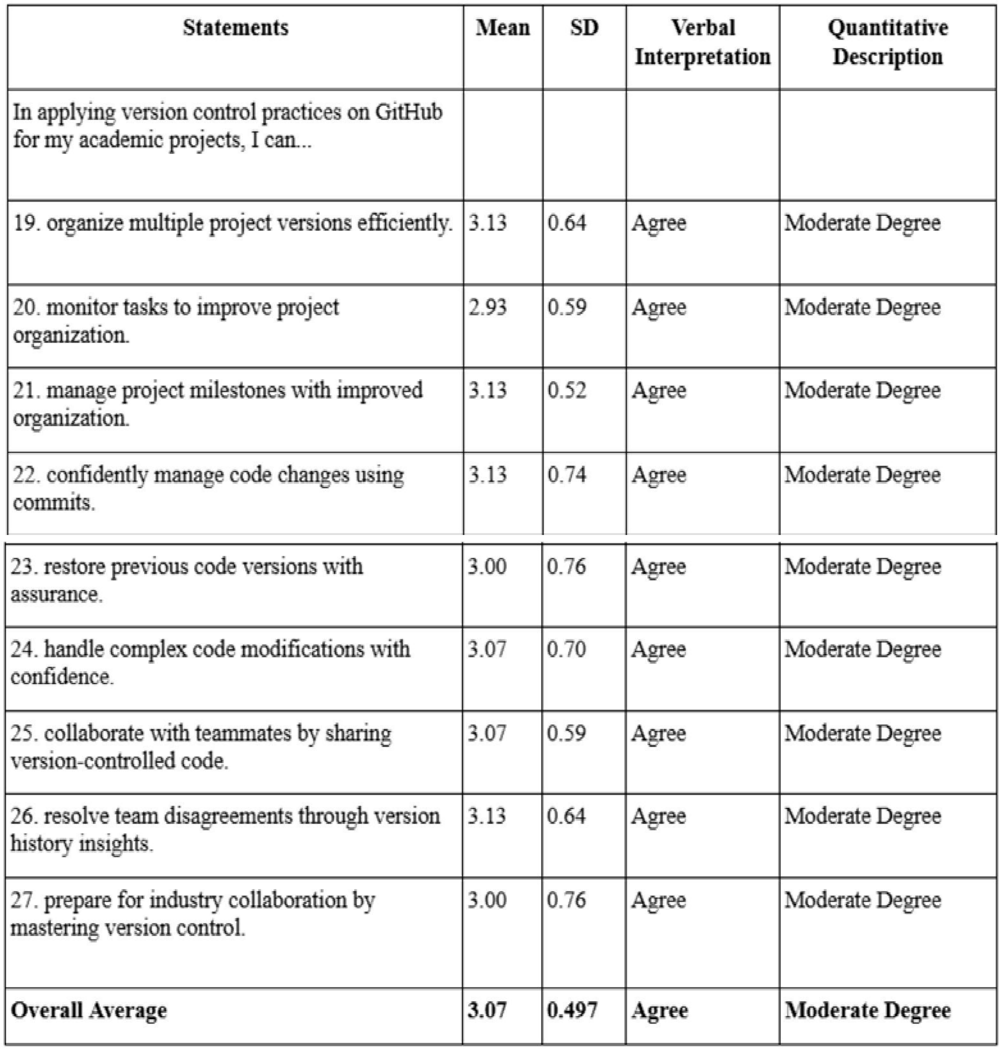
TABLE 2: Results assessing students' self-reported skills in utilizing GitHub's collaboration features



Statements	Mean	SD	Verbal Interpretation	Quantitative Description
In using collaboration features on GitHub for my academic projects, I can...				
10. coordinate group project tasks effectively.	3.20	0.68	Agree	Moderate Degree
11. assign tasks to improve project organization.	3.07	0.70	Agree	Moderate Degree
12. plan collaborative workflows to enhance project structure.	3.20	0.56	Agree	Moderate Degree
13. integrate code with confidence using pull requests.	3.13	0.74	Agree	Moderate Degree
14. manage code versions with increased assurance through reviews.	3.20	0.56	Agree	Moderate Degree
15. resolve coding conflicts with greater confidence.	3.13	0.64	Agree	Moderate Degree
16. work seamlessly with peers on group projects.	3.13	0.83	Agree	Moderate Degree
17. provide constructive feedback to enhance team collaboration.	3.13	0.83	Agree	Moderate Degree
18. adapt to diverse team dynamics using discussion tools.	3.13	0.64	Agree	Moderate Degree
Overall Average	**3.15**	**0.555**	**Agree**	**Moderate Degree**

Table 2 presented the survey results assessing students' self-reported skills in utilizing GitHub's collaboration features. The data indicated a generally moderate level of proficiency (all described as "Moderate Degree"), with mean scores ranging from 3.07 to 3.20, and all above the midpoint of the Likert scale. Students reported a moderate level of confidence and skill in coordinating group project tasks, assigning tasks to improve project organization, planning collaborative workflows, integrating code using pull requests, managing code versions through reviews, resolving coding conflicts, working seamlessly with peers, providing constructive feedback, and adapting to diverse team dynamics. The relatively consistent mean scores across all items suggested a balanced level of perceived competence in various aspects of collaborative work on GitHub. The overall average mean of 3.15 further reinforced this finding of moderate proficiency. While the results indicated a generally positive perception of their collaborative skills using GitHub, the moderate scores suggested that further development and training in these areas could enhance students' collaborative project management skills.

TABLE 3: Results related to students' self-reported skills in applying version control practices on GitHub



Statements	Mean	SD	Verbal Interpretation	Quantitative Description
In applying version control practices on GitHub for my academic projects, I can...				
19. organize multiple project versions efficiently.	3.13	0.64	Agree	Moderate Degree
20. monitor tasks to improve project organization.	2.93	0.59	Agree	Moderate Degree
21. manage project milestones with improved organization.	3.13	0.52	Agree	Moderate Degree
22. confidently manage code changes using commits.	3.13	0.74	Agree	Moderate Degree
23. restore previous code versions with assurance.	3.00	0.76	Agree	Moderate Degree
24. handle complex code modifications with confidence.	3.07	0.70	Agree	Moderate Degree
25. collaborate with teammates by sharing version-controlled code.	3.07	0.59	Agree	Moderate Degree
26. resolve team disagreements through version history insights.	3.13	0.64	Agree	Moderate Degree
27. prepare for industry collaboration by mastering version control.	3.00	0.76	Agree	Moderate Degree
Overall Average	**3.07**	**0.497**	**Agree**	**Moderate Degree**

Table 3 presented the survey results related to students' self-reported skills in applying version control practices on GitHub. The findings revealed a generally moderate level of proficiency across all aspects of version control, consistently rated as "Moderate Degree". Mean scores ranged from 2.93 to 3.13, all above the midpoint of the Likert scale, indicating agreement. While students demonstrated moderate confidence in organizing multiple project versions, managing project milestones, and confidently managing code changes using commits, their self-reported skills in task monitoring and restoring previous code versions were slightly lower (means of 2.93 and 3.00, respectively). The ability to handle complex code modifications and collaborate with teammates by sharing version-controlled code also fell within the moderate range. The overall average mean score of 3.07 further supported the conclusion of moderate proficiency in version control practices. These results suggested that while students possessed a foundational understanding of version control, further training and development were needed to enhance their skills, especially in areas such as task monitoring and handling complex code modifications, to improve their overall effectiveness in using version control for collaborative software development.

IV. CONCLUSION

This study investigated the relationship between GitHub feature utilization and development skills among third-year and fourth-year BSICT students at SNSU. It specifically examined students' self-reported proficiency in repository management skills, collaboration features, and version control practices to determine how these GitHub features correlate with their development capabilities.The findings revealed that students demonstrated moderate proficiency across all three GitHub feature domains. Repository Management Skills achieved a mean of 3.10, Collaboration Features scored 3.15, and Version Control Practices recorded 3.07, all indicating "Agree" responses with moderate degrees of proficiency. Students showed highest confidence in collaborative aspects such as working with teams through shared repository access and coordinating group tasks, while areas like task monitoring and error recovery showed lower confidence levels.These results suggest that while BSICT students recognize the value of GitHub features for enhancing their development skills, there remains substantial room for improvement in their utilization proficiency. The study contributes to the limited research on GitHub's educational impact among undergraduate ICT students by providing empirical evidence of the relationship between platform usage and perceived skill development in the Philippine educational context.

ACKNOWLEDGMENT

The researchers express their sincere gratitude to the administration of Surigao del Norte State University for their invaluable support in this study. We are deeply thankful for the provision of necessary data and for granting permission to conduct this research, enabling the team to successfully complete this project.

REFERENCES

[1]. Adam, E. (2024). An Investigation Into the Perceived Effectiveness of GitHub Repositories to Teach Programming. Proceedings of the 1st International Conference on Education Research, 1(1), 1–9. https://doi.org/10.34190/icer.1.1.2774

[2]. Patani, P., Tiwari, S., & Rathore, S. (2024). The impact of GitHub on students' learning and engagement in a software engineering course. https://www.semanticscholar.org/paper/The-impact-of-GitHub-on-students'-learning-and-in-a-Patani-Tiwari/1aff2d67af78c7102ca5c40dce542813543659ac

[3]. Hollowed, E. (2020, December 30). Announcing the GitHub Education Classroom Report 2020 - the GitHub blog. The GitHub Blog. https://github.blog/news-insights/research/announcing-the-github-education-classroom-report-2020/

[4]. Bennedsen, J., Böjttjer, T., & Tola, D. (2022). USING GITHUB CLASSROOM IN TEACHING PROGRAMMING | Worldwide CDIO Inititative. Conceive Design Implement Operate. Retrieved March 3, 2025, from https://www.cdio.org/knowledge-library/documents/using-github-classroom-teaching-programming

[5]. Beckman, M. D., Çetinkaya-Rundel, M., Horton, N. J., Rundel, C. W., Sullivan, A. J., & Tackett, M. (2020). Implementing version control with Git and GitHub as a learning objective in statistics and data science courses. https://doi.org/10.48550/arXiv.2001.01988

[6]. Orvalho, P., Janota, M., & Manquinho, V. (2024). GitSEED: A Git-backed Automated Assessment Tool for Software Engineering and Programming Education. https://doi.org/10.48550/arXiv.2409.07362

[7]. Tan, S. H., Hu, C., Li, Z., Zhang, X., & Zhou, Y. (2020). GitHub-OSS Fixit: Fixing bugs at scale in a Software Engineering Course. https://doi.org/10.48550/arXiv.2011.14392

[8]. Santos, I., Felizardo, K. R., Sarma, A., Steinmacher, I., & Gerosa, M. A. (2025). OSSDoorway: A Gamified Environment to Scaffold Student Contributions to Open Source Software. https://doi.org/10.48550/arXiv.2502.07986

[9]. Buchta, J., Petrenko, M., Poshyvanyk, D., & Rajlich, V. (2022). Can pre-class GitHub contributions predict success by student teams? In Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Software Engineering Education and Training (pp. 1–11). https://doi.org/10.1145/3510456.3514144

[10]. Herlambang, A. D., Rachmadi, A., & Wijoyo, S. H. (2023). Git and GitHub Application Training Program to Support Vocational High School Students in Collaborative Computer Programming Learning. Jurnal Pendidikan dan Pemberdayaan Masyarakat, 10(1). https://doi.org/10.21831/jppm.v10i1.58550

[11]. Tushev, M., Williams, G., & Mahmoud, A. (2020). Using GitHub in large software engineering classes: An exploratory case study. Computer Science Education, 30(2), 155–186. https://doi.org/10.1080/08993408.2019.1696168

[12]. Rundel, C. W., Çetinkaya-Rundel, M., & Anders, B. (2020). Implementing Version Control With Git and GitHub as a Learning Objective in Statistics and Data Science Courses. Journal of Statistics Education, 28(2), 161–168. https://doi.org/10.1080/10691898.2020.1848485

[13]. Glassey, R., Baltatzis, A., Monga, M., Lonati, V., Barendsen, E., Sheard, J., & Paterson, J. (2024). Active Repos: Integrating Generative AI Workflows into GitHub. In Proceedings of the 2024 on Innovation and Technology in Computer Science Education V (pp. 777–778). https://doi.org/10.1145/3649405.3659517

[14]. Cui, J., Zhang, R., Li, R., Zhou, F., Song, Y., Gehringer, E., Monga, M., Lonati, V., Barendsen, E., Sheard, J., & Paterson, J. (2024). A Comparative Analysis of GitHub Contributions Before and After An OSS Based Software Engineering Class. In Proceedings of the 2024 on Innovation and Technology in Computer Science Education V (pp. 576–582). https://doi.org/10.1145/3649217.3653535

[15]. Schauer, L., Stewart, R., Maarek, M., Roychoudhury, A., Paiva, A., Abreu, R., Storey, M., Gama, K., & Siegmund, J. (2024). Integrating Canvas and GitLab to Enrich Learning Processes. In Proceedings of the 46th International Conference on Software Engineering: Software Engineering Education and Training (pp. 180–190). https://doi.org/10.1145/3639474.3640056

[16]. Zagalsky, A., Feliciano, J., Storey, M.-A., Zhao, Y., & Wang, W. (2015). The emergence of GitHub as a collaborative platform for education. In Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (pp. 1906–1917). https://doi.org/10.1145/2675133.2675284

[17]. Alves, P., & Cipriano, B. P. (2025). Automated Assessment in Mobile Programming Courses: Leveraging GitHub Classroom and Flutter for Enhanced Student Outcomes. https://doi.org/10.48550/arXiv.2504.04230

[18]. Snowberger, A. D., & Lee, C. H. (2023). A Workflow for Practical Programming Class Management Using GitHub Pages and GitHub Classroom. Journal of Practical Engineering Education, 15(2), 331–339. https://doi.org/10.14702/JPEE.2023.331

[19]. Snowberger, A. D., & You, K. (2025). Leveraging GitHub Classroom and Google Colab for Short-Term Student Machine Learning Team Projects. Journal of Practical Engineering Education, 17(1), 49–61. https://doi.org/10.14702/JPEE.2025.049

[20]. Pajares, F. (1996). Self-Efficacy Beliefs in Academic Settings. Review of Educational Research, 66(4), 543–578.https://doi.org/10.3102/00346543066004543

[21]. Hsing, C., & Gennarelli, V. (2019). Using GitHub in the Classroom Predicts Student Learning Outcomes and Classroom Experiences: Findings from a Survey of Students and Teachers. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 672–678). https://doi.org/10.1145/3287324.3287460

[22]. Fiksel, J., Jager, L. R., Hardin, J. S., & Taub, M. A. (2022). Using GitHub Classroom to Teach Statistics. Journal of Statistics and Data Science Education, 30(2), 190–204. https://doi.org/10.1080/26939169.2022.2075718

[23]. Haaranen, L., & Lehtinen, T. (2015). Teaching Git on the Side: Version Control System as a Course Platform. In Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education (pp. 87–92). https://doi.org/10.1145/2729094.2742618

[24]. Lawrance, J., Jung, S., & Wiseman, C. (2013). Git on the Cloud in the Classroom. In Proceedings of the 44th ACM Technical Symposium on Computer Science Education (pp. 639–644). https://doi.org/10.1145/2445196.2445386

DOI: 10.48175/IJARSCT-28254

[25]. Kertész, C. (2015). Using GitHub in the classroom - a collaborative learning experience. International Symposium for Design and Technology in Electronic Packaging, Brasov, Romania, 381–386. https://doi.org/10.1109/SIITME.2015.7342358

[26]. The jamovi project (2024). jamovi. (Version 2.6) [Computer Software]. Retrieved from https://www.jamovi.org.

[27]. R Core Team (2024). R: A Language and environment for statistical computing. (Version 4.4) [Computer software]. Retrieved from https://cran.r-project.org. (R packages retrieved from CRAN snapshot 2024-08-07)