INTERNATIONAL JOURNAL OF LATEST TECHNOLOGY IN ENGINEERING, MANAGEMENT & APPLIED SCIENCE (IJLTEMAS)
ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

A Quantitative Analysis of Information Verification Habits and Cross-Checking Behaviors Toward AI Hallucinations Among College Students

John Wilmer T. Bagayan, Harold R. Lucero, Dianne R. Mananghaya, Christina Ysabel T. Patulilic,Arjay D. Garabiag , Vince Joseph G. Vargas

College of Computer Studies, Quezon City University

DOI: https://doi.org/10.51583/IJLTEMAS.2026.150500084

Received: 08 May 2026; Accepted: 13 May 2026; Published: 02 June 2026

ABSTRACT

This study investigated the information verification habits and cross-checking behaviors of Computer Studies students when encountering Generative Artificial Intelligence (AI) hallucinations. The primary objective was to determine how students' demographic and AI usage profiles influence their active verification methods, error detection success rates, and perceived reliability of AI tools. A quantitative descriptive-correlational design was adopted, utilizing a validated, scenario-based digital survey deployed via Google Forms to 200 Bachelor of Science in Information Technology (BSIT) and Bachelor of Science in Computer Science (BSCS) students at Quezon City University. The findings revealed that 84.5% of the respondents were highly active daily or weekly users of generative AI, primarily utilizing general web chatbots (58.5%) and dedicated Integrated Development Environment (IDE) assistants (41.5%) for debugging (45.5%) and code generation (33.5%). Students demonstrated a structured baseline of verification rigor (Composite Mean = 3.13, Agree), heavily prioritizing external validation such as manually confirming the existence of new libraries (Weighted Mean = 3.23) and cross-referencing official documentation (Weighted Mean = 3.20) rather than trusting the AI to self-verify its own outputs (Weighted Mean = 2.94). However, a significant cognitive vulnerability to automation bias was established, as students explicitly agreed that convenience and speed outweigh the risks of incorrect syntax (Weighted Mean = 3.11) and falsely perceived models as possessing deep contextual understanding (Weighted Mean = 3.20). Furthermore, a statistically significant negative correlation was found between the primary tool utilized and verification habits ($r = -0.177$, $p = 0.012$), proving that seamless inline code generation within dedicated IDE assistants suppresses manual auditing. Conversely, verification rigor remained entirely uniform across all academic year levels ($F(3, 196) = 1.35$, $p = 0.261$), overall usage frequencies ($r = 0.053$, $p = 0.453$), and task complexities ($r = -0.097$, $p = 0.170$). Ultimately, the study concludes that cognitive over-reliance is driven primarily by interface delivery friction rather than student seniority, designating an urgent institutional need to transition away from restrictive policies and proactively embed formal "AI Auditing" instruction, mandatory external citation protocols, and foundational mechanics of Large Language Models directly into the core programming curriculum.

Keywords: AI Hallucinations, Automation Bias, Computer Studies, Generative AI, Information Verification

INTRODUCTION

The emergence of Generative Artificial Intelligence (AI) has shifted the paradigm of modern problem-solving, particularly within the field of Information Technology. Tools that provide instant coding, logical explanations and solutions are essential for College students. However, the perceived intelligence of these models is often shadowed by a phenomenon known as AI Hallucination. This occurs when a model provides a response that seems extremely certain and decisive but is fundamentally incorrect, outdated, or lacking in logical coherence (Ji et al., 2023).

In a programming context, these hallucinations are particularly deceptive. Since AI models produce text based on pattern recognition but without understanding of the real world, they can create new syntax or recommend

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

outdated libraries. For the College students, the danger is twofold: It leads to broken programs and, more importantly, to over-reliance that bypasses the learning process for the skills required. This over-reliance is often characterized as Automation Bias, the tendency to favor suggestions from automated systems even when they lack empirical proof (Kahn et al., 2024).

While much of current literature focuses on the utility of AI, there is a significant gap in understanding the skepticism of its users. Recent studies suggest that while students recognize these errors, their actual "active verification strategies" vary wildly in effectiveness (Shoufan & Esmaeil, 2026). Ultimately, the value of AI in education isn't decided by the suggestion it makes, but by the user's ability to source these suggestions. If the students treat AI as a definitive source rather than a suggestive tool, the fundamentals skill of debugging and fact-checking is lost.

This research aims to investigate the information verification habits of college students in technical courses. By analyzing how students navigate the "confident-but-unreliable" nature of AI, this study seeks to determine if they possess the systematic habits required to cross-check hallucinations against verified documentation and compilers (Soares et al., 2025), or if the convenience of the tool is beginning to outweigh the necessity of accuracy.

Statement of the Problem

This study aims to evaluate the information verification habits of Computer Studies students when encountering AI-generated "hallucinations," a well-documented phenomenon where models generate plausible but factually incorrect information. Specifically, it seeks to determine how the students' AI usage profiles influence their verification methods, error detection success rates, and perceived reliability of Artificial Intelligence tools.

Specifically, this study seeks to answer the following questions:

1. What is the demographic profile of the respondents in terms of:
    - Age;
    - Sex at birth;
    - Year Level in the Information Technology program;
    - Enrollment Status;
    - Frequency of Generative AI usage for technical problem-solving?

2. What is the AI usage profile of the respondents in terms of:
    - Type of AI Tool utilized ; and
    - Nature of the Task performed?

3. What is the level of the students' information verification habits when encountering AI hallucinations in terms of:
    - Frequency of verification methods used ;
    - Success rate of error detection;
    - Perceived reliability of AI?

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

4. Is there a significant difference in the information verification habits of the respondents when they are grouped according to their demographic profile?

5. Is there a significant relationship between the students' AI usage profile and their information verification habits?

6. What "AI Auditing" guidelines or curriculum updates can be proposed based on the findings of the study?

RELATED LITERATURE AND STUDIES

The emergence of Generative Artificial Intelligence (AI) has shifted the paradigm of modern problem-solving, particularly within the field of Information Technology. Tools that provide instant coding, logical explanations, and immediate technical solutions have become essential for college students. However, the perceived intelligence of these models is often shadowed by a well-documented phenomenon known as AI Hallucination. This occurs when a natural language generation model provides a response that appears extremely certain and decisive but is fundamentally incorrect, outdated, or lacking in logical coherence (Ji et al., 2023).

In a programming context, these hallucinations are particularly deceptive. Since generative AI models produce text based on complex pattern recognition rather than an empirical understanding of the real world, they frequently fabricate new syntax or recommend outdated libraries (Ji et al., 2023). For college students, the danger is twofold: relying on unverified outputs leads directly to broken programs and, more importantly, fosters an over-reliance that bypasses the fundamental learning processes required to master technical skills.

This cognitive over-reliance is often characterized as Automation Bias, defined as the tendency of human decision-makers to favor suggestions from automated systems even when those suggestions lack empirical proof or present flawed logic (Horowitz & Kahn, 2024). Recent investigations into AI-supported technical environments emphasize the real-world consequences of this bias, quantifying substantial shortfalls in students' programming practices when they rely heavily on generative tools without proper auditing (Mehra et al., 2025).

Furthermore, empirical evaluations of AI-generated code confirm that users must maintain a strict, systematic approach to actively cross-check automated logic against verified documentation, secure environments, and actual compilers (Soares et al., 2025). If students treat AI as a definitive source rather than a suggestive tool, the fundamental, critical skills of independent debugging and systematic fact-checking are lost.

While much of the current literature focuses on the utility of AI, there is a significant gap in understanding the skepticism and active verification habits of its technical users. Recent studies suggest that while students generally recognize that generative models make errors, their actual active verification strategies vary wildly in execution and effectiveness (Shoufan & Esmaeil, 2026).

Specifically, undergraduate students enrolled in Bachelor of Science in Information Technology (BSIT) and Bachelor of Science in Computer Science (BSCS) programs are the primary users of AI tools in programming-related tasks, making them the most relevant population for examining AI hallucinations and verification behaviors. Limiting research samples to specific IT-oriented student groups is highly supported by recent literature investigating these exact dynamics, such as the UiTM Generation Z study (2025) and targeted explorations of online learning habits and academic productivity among BSIT students (“An Exploration of Online Learning Habits”, 2025).

Because BSIT and BSCS students frequently interact with digital and cloud-based resources, navigating these platforms safely requires robust information literacy, aligning with broader studies on how undergraduates search for, vet, and verify critical information (Göksel & Akgül, 2021). Effectively navigating the modern infodemic requires a deliberate, conscious effort from students to vet misinformation and cross-reference unverified claims against authoritative sources (“Navigating the Infodemic”, 2024).

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

In technical domains, the direct relationship between AI usage frequency and verification rigor is a focal point of ongoing research. Recent descriptive-correlational studies focusing on STEM students indicate a distinct "Hallucination Effect," where the frequency of generative AI usage directly intersects with, and sometimes correlates negatively against, an individual's source verification habits (“The Hallucination Effect”, 2024). When the convenience of an immediate, automated solution begins to outweigh the perceived necessity of accuracy, students risk bypassing systematic cross-checking behaviors entirely.

The reviewed literature is highly relevant to the current investigation as it establishes the foundational tension between generative AI's utility and its inherent risks, directly informing the study's core objectives and variables. While existing literature clearly demonstrates that generative AI tools offer immense convenience and accelerate technical productivity, it simultaneously underscores that these tools introduce confident but unreliable errors that actively exploit human automation bias.

Prior studies provide a vital conceptual framework by documenting AI hallucinations from a general student perspective (Shoufan & Esmaeil, 2026) and exploring online academic productivity among IT-oriented groups (“An Exploration of Online Learning Habits”, 2025). More importantly, these works highlight a distinct empirical gap: the lack of documented, standardized verification habits specifically employed by Bachelor of Science in Information Technology (BSIT) and Bachelor of Science in Computer Science (BSCS) students to audit automated outputs.

Consequently, the literature directly justifies the necessity and direction of this research. By establishing that the educational and practical value of AI in computer studies depends entirely on a user's systematic capacity to source, cross-check, and verify outputs against validated documentation, prior research provides the exact rationale for this study. It directly frames the current investigation into how BSIT and BSCS students navigate AI unpredictability, specifically guiding the examination of how their AI usage profiles influence their active verification methods, error detection success rates, and perceived reliability of artificial intelligence tools.

METHODOLOGY

A. Research Design

This study adopts a quantitative descriptive-correlational research design to describe the information verification habits and cross-checking behaviors of Computer Studies students and to examine relationships between these habits and selected variables. According to recent studies on AI-related student behaviors, this design is appropriate when the goal is to describe current practices and to examine associations without manipulating conditions (Göksel & Akgül, 2021; “The Hallucination Effect”, 2024; Mehra et al., 2025).

B. Respondents of the Study

The respondents of the study are Computer Studies students enrolled in the BSIT and BSCS programs in Academic Year 2025-2026 at Quezon City University. The target population consists of all undergraduate students currently taking programming and core Computer Studies courses across all year levels (1st to 4th year). The study focuses only on these respondents, as they are the primary users of AI tools in programming-related tasks and are therefore most relevant to the research problem on AI hallucinations and verification behavior, similar to recent studies that limit their sample to specific IT-oriented student groups (UiTM Generation Z study, 2025; “An Exploration of Online Learning Habits and Academic Productivity among BSIT Students”, 2025).

C. Sampling Technique

The study will use stratified random sampling as the sampling technique. According to Göksel and Akgül (2021), stratified sampling is appropriate when comparing student groups across levels of experience, as it helps control for differences in exposure to programming courses and technology use. Computer Studies students will be grouped by year level (1st, 2nd, 3rd, and 4th year). Using proportional allocation, the exact number of respondents drawn from each year level will be determined based on their respective total population sizes,

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

yielding a total sample of 200 respondents. This sample size is consistent with recent descriptive-correlational studies on AI-related behaviors among college students (e.g., "The Hallucination Effect", 2024; Mehra et al., 2025).

D. Data Gathering

This section delineates the systematic approach executed by the researchers to acquire the necessary empirical data for the study (Calderon & Gonzales, 1993). To ensure the reliability of the findings and the ethical treatment of the respondents, a structured protocol was strictly observed throughout the data collection phase (Creswell & Creswell, 2018). The following procedures outline the sequential process undertaken by the researchers, encompassing the acquisition of administrative clearances, the digital deployment of the validated survey to the Computer Studies students, and the secure retrieval of their responses for statistical analysis (Flick, 2018).

E. Instrument Used

The primary tool for data collection in this study is a structured survey questionnaire deployed digitally via Google Forms. Utilizing a cloud-based platform allows for the efficient distribution of the survey and real-time, organized data collection from the target respondents. A standardized, self-administered digital survey is an optimal approach for quantitative research, as it ensures all participants are presented with the exact same stimuli and formatting, thereby increasing the reliability of the collected data (Fowler, 2013).

Construction of the Instrument

The questionnaire is divided into three distinct sections designed to comprehensively evaluate the students. The first section gathers demographic data and establishes the students' baseline AI usage profiles. The second section utilizes a 4-point Likert scale (Never, Rarely, Often, Always) to measure the frequency of their verification methods; utilizing an even-numbered scale eliminates the neutral midpoint, which forces respondents to make a definitive choice and yields more precise data regarding their actual behavior (Joshi et al., 2015). The third section employs scenario-based questions where students identify their responses to domain-specific AI "hallucinations," such as encountering fake library functions or deprecated syntax. Framing questions within specific scenarios effectively translates abstract theoretical concepts into practical, relatable situations, improving the accuracy of the responses (Rowley, 2014).

Validation of the Instrument

To ensure the accuracy and reliability of the gathered data, the constructed Google Form will undergo a rigorous validation process prior to deployment. The draft will be presented to a validation panel composed of a statistician, the researchers' advising professor, and IT experts. This expert panel will evaluate the instrument to establish face and content validity, ensuring that the programming scenarios and AI terminology are technically sound, and that the data points align perfectly with the required statistical treatments. Incorporating expert judgment is a critical step in the validation process to confirm that the survey consistently and accurately measures what it intends to measure without ambiguity (Taherdoost, 2016).

Administration and Retrieval of the Instrument

The administration of the instrument will commence once formal written approval is secured from the relevant authorities. The validated Google Forms link will be distributed to the selected respondents, featuring a digital informed consent section on the landing page that details the study's purpose, the voluntary nature of their participation, and the strict confidentiality of their responses. Establishing these clear ethical parameters and securing informed consent prior to data retrieval is a fundamental requirement in research to protect the rights and privacy of human participants (Creswell & Creswell, 2018).

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

F. Statistical Treatment of Data

To ensure systematic analysis and interpretation of the gathered data regarding the information verification habits of Computer Studies Students, the following statistical tools will be employed (Babbie, 2020). The data will be processed using descriptive and inferential statistics to provide a comprehensive answer to the research objectives (Gravetter & Wallnau, 2017).

Frequency and Percentage. This tool will be utilized to describe the demographic profile of the respondents and to show the distribution of responses for each item in the survey (Calderon & Gonzales, 1993).

Formula:

$$P(\%) = \frac{F}{N} \quad x\ 100$$

Where:

P (%) = Percentage

F = Frequency

N = Total Number of Respondents

100 = Constant

Weighted Mean. According to Bluman (2018), the weighted mean is used to calculate the central tendency of a data set where values are assigned specific weights, such as in survey scales. In this study, it will be calculated to determine the central tendency of the students' responses. This allows the researchers to identify the average level of frequency or agreement concerning how students cross-check AI-generated content.

Formula:

$$\mathrm{x} = \frac{\sum \mathrm{Fx}}{\mathrm{N}}$$

Where:

$\boldsymbol{x}$ **=** Weighted Mean

$\sum$ **FX =** Total of Frequency and Response

N = Total Number of Respondent

Likert Scale Description. To interpret the calculated weighted means, a 4-point Likert scale is utilized (Likert, 1932). This forced-choice format requires respondents to indicate a definitive habit. The following scale, adapted from Vagias (2006), will serve as the basis for the verbal interpretation of the results:

Table 1 Likert Scale used in assessing the Quantitative Analysis of Information Verification Habits and Cross-Checking Behaviors Toward AI Hallucinations Among College Students

Scale	Weighted Mean	Description
4	3.26-4.00	Strongly Agree (SA)

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

3	2.51-3.25	Agree (A)
2	1.76-2.50	Disagree(D)
1	1.00-1.75	Strongly Disagree (SD)

Inferential Statistics (One-Way ANOVA). To determine if there are significant differences in verification habits when respondents are grouped by year level, a One-Way Analysis of Variance (ANOVA) will be used. This test determines if the variation in habits is statistically significant across different academic stages (Field, 2018). A p-value of less than 0.05 will be the threshold for rejecting the null hypothesis (Pallant, 2020).

RESULT AND DISCUSSION

This chapter deals with the presentation and interpretation of data. Which are presented in tables, analyzed, and interpreted using the descriptive rating data.

A. Demographic Profile of the Respondents

This section presents the baseline characteristics of the surveyed Computer Studies respondents (n = 200) enrolled in the Bachelor of Science in Information Technology (BSIT) and Bachelor of Science in Computer Science (BSCS) programs at Quezon City University for the Academic Year 2025–2026.

To establish a clear context for the sample population, the respondents are characterized across five primary demographic variables: **age, sex at birth, academic year level, enrollment status,** and their **frequency of Generative AI usage for technical problem-solving**. Analyzing these demographic variables is essential, as they provide the underlying context for interpreting the students' subsequent technical behaviors, susceptibility to automation bias, and active cross-checking habits when encountering AI hallucinations.

Age

Table 2 reveals that exactly half of the surveyed respondents (50.0%, n = 100) fall within the 17–19 age bracket, making it the most prominent demographic group. This is closely followed by students aged 20–22, who account for 43.5% (n = 87) of the sample.

Combined, these two groups constitute the vast majority (93.5%) of the population, indicating that the study primarily captures the perspectives of traditional, college-aged youth. Only a small fraction of the respondents are aged 23–25 (5.5%, n = 11) or 26 and above (1.0%, n = 2).

Table 2 Demographic profile of the respondents in terms of Age

Age	Frequency	Percentage
17-19	100	50.0%
20-22	87	43.5%
23-25	11	5.5%
26 and above	2	1.0%
Total	200	100%

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

Sex at Birth

Table 3 Demographic profile as to Sex

Sex at Birth	Frequency	Percentage
Male	136	68.0%
Female	64	32.0%
Total	200	100%

Table 3 presents the distribution of respondents according to their sex at birth. The data shows a distinct majority of male respondents, comprising 68.0% (n = 136) of the total sample, compared to female respondents who make up the remaining 32.0% (n = 64). This distribution is characteristic of typical enrollment patterns observed within technical and computing degree programs, where male students historically represent a larger proportion of the student body.

Year Level in the Information Technology Program Academic Year 2025-2026

Table 4 Demographic profile as to Year Level on Academic Year 2025-2026

Year Level	Frequency	Percentage
1st Year	36	18.0%
2nd Year	26	13.0%
3rd Year	120	60.0%
4th Year	18	9.0%
Total	200	100%

Table 4 illustrates the distribution of respondents across their respective academic year levels, derived via proportional allocation to match the department's enrollment density. Third-year students represent the largest segment of the sample at 60.0% (n = 120), followed by first-year students at 18.0% (n = 36) and second-year students at 13.0% (n = 26).

Fourth-year students comprise the smallest portion of the sample at 9.0% (n = 18). This heavy concentration of third-year students provides a highly relevant sample, as upperclassmen typically engage in more advanced, programming-intensive coursework where generative AI tools and potential hallucinations are frequently encountered.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

Enrollment Status

Table 5 Demographic profile as to Enrollment Status

Status	Frequency	Percentage
Full-time Student (Not working)	173	86.5%
Working Student (Part-time)	22	11.0%
Working Student (Full-time)	5	2.5%
Total	200	100%

Table 5 details the enrollment status of the respondents. An overwhelming majority of the students (86.5%, n = 173) are classified as full-time students who are not currently working. Only a small subset balances their academic load with employment: 11.0% (n = 22) are part-time working students, while just 2.5% (n = 5) maintain full-time employment. This indicates that the vast majority of the respondents have the capacity to dedicate their primary focus to their academic and coding requirements without the competing time constraints of formal employment.

Frequency of Generative AI Usage for Technical Problem-Solving

Table 6 Frequency of Generative AI Usage for Technical Problem-Solving

Frequency of Use	Frequency	Percentage
Always (Daily / Almost every task)	101	50.5%
Often (3-4 times a week)	68	34.0%
Sometimes (1-2 times a week)	27	13.5%
Rarely (Once a month or less)	4	2.0%
Total	200	100%

Table 6 establishes the frequency with which respondents integrate Generative AI tools into their workflow. A slight majority of the surveyed students (50.5%, n = 101) report using these tools "Always," indicating daily engagement or reliance on AI for almost every technical task. An additional 34.0% (n = 68) utilize them "Often" (3–4 times a week). Combined, 84.5% of the population represents highly active users, demonstrating that generative assistance is deeply embedded in the modern academic routine. Only a small minority use AI "Sometimes" (13.5%, n = 27) or "Rarely" (2.0%, n = 4).

B. AI Usage Profile of the Respondents

This section characterizes the specific patterns and operational profiles of Generative AI adoption among the surveyed Computer Studies respondents **(n = 200)**. To fully understand how students navigate automated assistance during technical problem-solving, their usage typology is examined through two primary dimensions: the specific types of AI tools utilized and the nature of the tasks performed Establishing this baseline typology

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

is essential, as it contextualizes the respondents' technical ecosystem and provides the necessary framework for evaluating how different tool preferences and task complexities influence students' subsequent verification habits and susceptibility to AI hallucinations.

Type of AI Tool Utilized

Table 7 Primary Types of Generative AI Tools Utilized by the Respondents

Primary Tool Type	Frequency	Percentage
General Web Chatbots (e.g., ChatGPT, Claude)	117	58.5%
Dedicated AI IDE Assistants (e.g., GitHub Copilot)	83	41.5%
Total	200	100%

Table 7 details the specific categories of Generative AI tools preferred by the respondents. General web-based conversational chatbots, such as ChatGPT and Claude, are the primary choice for 58.5% (n = 117) of the students. Conversely, 41.5% (n = 83) primarily rely on dedicated AI Integrated Development Environment (IDE) assistants, such as GitHub Copilot. This distribution shows a strong dual preference: while the majority favor flexible, dialogue-driven platforms for broad explanations and logic generation, a highly significant portion prefers specialized, context-aware assistants embedded directly within their coding environments.

Nature of the Task Performed

Table 8 Primary Nature of the Programming Tasks Performed Using Generative AI

Primary Task Nature	Frequency	Percentage
Debugging or finding specific errors	91	45.5%
Generating entire blocks of new code/logic	67	33.5%
Explaining existing code or documentation	42	21.0%
Total	200	100%

Table 8 characterizes the operational intent behind the respondents' AI usage. The most prevalent application is debugging or finding specific errors, accounting for 45.5% (n = 91) of the tasks. This indicates that students frequently treat AI as an interactive diagnostic tool to troubleshoot broken logic. The second most common application is generating entire blocks of new code or logic (33.5%, n = 67), where AI acts as a primary development accelerator. Finally, 21.0% (n = 42) use the tools primarily for explaining existing code or documentation, leveraging the models as personalized instructional aids to comprehend complex syntax.

Level of the Students' Information Verification Habits

This section evaluates the core dependent variables of the study, characterizing the active information verification habits and cross-checking behaviors of the respondents (n = 200) when encountering potential

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

Generative AI hallucinations. To comprehensively assess how students navigate the confident but highly unreliable nature of automated outputs (Ji et al., 2023), their habits are analyzed across three specific dimensions aligned with the research objectives: the frequency of verification methods used (measured via weighted means), the success rate of error detection across domain-specific programming scenarios, and the overall perceived reliability of Artificial Intelligence tools. Examining these dimensions is critical to determining whether Computer Studies students possess the systematic auditing rigor required to independently validate automated logic against authoritative documentation (Soares et al., 2025), or if the convenience of instant generation is fostering automation bias and eroding foundational debugging skills (Horowitz & Kahn, 2024; Mehra et al., 2025).

Frequency of Verification Methods Used

Table 9 Frequency of Verification Methods Used by Respondents When Encountering AI Outputs

Indicator	**Weighted Mean**	**Verbal Interpretation**	**Rank**
1. I manually cross-check AI-suggested syntax against official documentation.	3.20	Agree	2
2. I search human-verified forums (e.g., Stack Overflow) to confirm solutions.	3.10	Agree	6
3. I independently verify the existence of new packages or libraries.	3.23	Agree	1
4. I rely solely on my IDE error tracking or syntax linter.	3.06	Agree	7
5. I review complex code blocks line-by-line to ensure I understand logic.	3.18	Agree	3
6. I intentionally test AI-generated code with edge cases or boundary values.	3.18	Agree	3
7. I assume new function/variable names are valid without checking.	2.88	Agree	8
8. I explicitly request and manually verify citations or reference URLs.	3.18	Agree	3
Composite Mean	**3.13**	**Agree**	

Table 9 indicates that the respondents actively employ structured verification methods when encountering Generative AI outputs, as evidenced by an overall composite mean of 3.13 (Agree). The highest-ranked verification habit among the students is independently verifying the existence of new packages or libraries (Weighted Mean = 3.23, Agree), demonstrating that respondents prioritize checking external dependencies to avoid critical compilation failures caused by AI package hallucinations (Ji et al., 2023). Closely following is the practice of manually cross-checking AI-suggested syntax against official documentation (Weighted Mean = 3.20, Agree), reinforcing a strong reliance on primary, authoritative sources.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

Furthermore, respondents demonstrate robust engagement through three tied practices ranked third (Weighted Mean = 3.18, Agree): reviewing complex code blocks line-by-line to ensure logical comprehension, intentionally testing generated code against edge cases or boundary values, and explicitly requesting and manually verifying citations or reference URLs. Secondary validation layers include consulting human-verified forums such as Stack Overflow (Weighted Mean = 3.10, Agree) and relying on IDE error tracking or syntax linters (Weighted Mean = 3.06, Agree). Conversely, assuming new function or variable names are valid without checking received the lowest weighted mean (Weighted Mean = 2.88, Agree). While this comparative reluctance indicates an aversion to blindly trusting fabricated identifiers, its placement within the affirmative range highlights a persistent vulnerability to automation bias (Horowitz & Kahn, 2024), where the convenience of instant generation entices students to bypass granular manual vetting.

Success Rate of Error Detection

Table 10 Success Rate of Error Detection in Domain-Specific AI Hallucination Scenarios

Indicator	Weighted Mean	Verbal Interpretation	Rank
1. I manually search the web to confirm a library exists before compiling.	3.22	Agree	1
2. I ask the AI to verify if the library is real and trust its follow-up.	2.94	Agree	4
3. I rely entirely on the IDE's error highlighting for outdated syntax.	3.13	Agree	3
4. I cross-reference suggested syntax with official language documentation.	3.18	Agree	2
Composite Mean	**3.12**	**Agree**	

Table 10 details the specific cross-checking behaviors employed by respondents when encountering potential AI hallucinations in domain-specific programming scenarios, yielding an overall composite mean of 3.12 (Agree). An analysis of the ranked indicators reveals a clear preference for independent, external validation over internal AI confirmation. The highest-ranked habit among respondents is manually searching the web to confirm whether a suggested library actually exists prior to compilation (Weighted Mean = 3.22, Agree), demonstrating an acute awareness of the common AI pitfall of fabricating non-existent libraries (Ji et al., 2023).

Closely following this is the practice of cross-referencing suggested syntax directly with official language documentation (Weighted Mean = 3.18, Agree), reflecting a solid foundation in information literacy where official documentation is treated as the primary authoritative source (Göksel & Akgül, 2021). Additionally, respondents frequently rely on their Integrated Development Environment's (IDE) error highlighting for outdated syntax (Weighted Mean = 3.13, Agree), leveraging their automated environments as an immediate, real-time safety net. Conversely, asking the AI itself to verify if the library is real and trusting its follow-up received the lowest weighted mean among the indicators (Weighted Mean = 2.94, Agree). Although students still generally agree with this practice, the noticeably lower score proves a documented, healthy skepticism (Shoufan & Esmaeil, 2026); learners recognize the risks of recursive errors when asking a fallible model to verify its own output, ultimately preferring external web searches and authoritative documentation over trusting the AI's internal self-correction.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

Perceived Reliability of AI

Table 11 Perceived Reliability of Generative Artificial Intelligence Tools

Indicator	Weighted Mean	Verbal Interpretation	Rank
1. I trust AI assistants provide code that is secure and free of vulnerabilities.	2.98	Agree	6
2. I assume eloquent explanations from AI are factually correct.	3.04	Agree	3
3. AI models are highly reliable for complex math or logical algorithms.	3.00	Agree	4
4. Convenience and speed outweigh the risk of receiving incorrect syntax.	3.11	Agree	2
5. I feel anxious or less confident if forced to code without AI tools.	2.96	Agree	7
6. I believe AI tools possess a deep, contextual understanding of concepts.	3.20	Agree	1
7. I am more likely to trust the AI model over a textbook or professor.	2.92	Agree	8
8. AI reduces the necessity for developers to master foundational skills.	3.00	Agree	4
Composite Mean	**3.03**	**Agree**	

Table 11 assesses the respondents' overarching perceptions regarding the trustworthiness and reliability of Generative AI tools, resulting in an overall composite mean of 3.03 (Agree). This indicates that Computer Studies students generally attribute a high degree of authority to automated assistants, reflecting strong underlying trust that heavily intersects with automation bias (Horowitz & Kahn, 2024). The highest-ranked belief among respondents is that AI tools possess a deep, contextual understanding of concepts (Weighted Mean = 3.20, Agree), demonstrating a tendency to overestimate the actual cognitive comprehension of pattern-matching generative models (Ji et al., 2023).

Closely tied to this reliance is the second-ranked indicator, where students agree that convenience and speed outweigh the risk of receiving incorrect syntax (Weighted Mean = 3.11, Agree). This specific finding provides direct empirical evidence of a critical trade-off, confirming that the immediate efficiency of automated generation frequently entices students to accept higher risks of encountering AI hallucinations.

Furthermore, respondents are highly susceptible to the persuasive presentation of natural language models, ranking the assumption that eloquent explanations from AI are factually correct third (Weighted Mean = 3.04, Agree). Tied at the fourth rank (Weighted Mean = 3.00, Agree) are the belief that AI models are highly reliable for complex math or logical algorithms, and the view that AI reduces the necessity for developers to master foundational skills.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

Other established perceptions include trusting that AI assistants provide code secure and free of vulnerabilities (Weighted Mean = 2.98, Agree) and experiencing anxiety or reduced confidence when forced to code entirely without AI tools (Weighted Mean = 2.96, Agree). Finally, claiming to be more likely to trust the AI model over a textbook or professor received the lowest weighted mean among the indicators (Weighted Mean = 2.92, Agree). While placing last indicates that traditional academic authorities retain a comparative edge in credibility, the fact that this indicator still falls within the affirmative "Agree" range highlights a profound shift in technical education, where automated tools are increasingly viewed as highly authoritative sources alongside established academic expertise.

Differences in Information Verification Habits Grouped by Demographic Profile

To determine if there are significant differences in information verification habits when respondents are grouped according to their demographic profile—specifically across their academic year levels—a One-Way Analysis of Variance (ANOVA) was utilized. This inferential test evaluates whether the variation in active cross-checking habits is statistically significant across different academic stages (Field, 2018). In accordance with the established statistical treatment of data, a p-value of less than 0.05 serves as the definitive threshold for rejecting the null hypothesis (Pallant, 2020).

The analysis indicates no statistically significant difference in information verification habits across the academic year levels ($F(3, 196) = 1.35$, $p = 0.261$). Since the computed p-value of 0.261 exceeds the significance threshold of 0.05, the null hypothesis is accepted. The descriptive data shows relatively uniform composite verification means across the year levels (1st Year = 3.16; 2nd Year = 3.08; 3rd Year = 3.09; 4th Year = 3.35). This suggests that the propensity to cross-check automated outputs or succumb to automation bias is a uniform behavioral trait among Computer Studies students, operating independently of their accumulated academic experience or seniority within the program.

Table 12 One-Way ANOVA Comparing Information Verification Habits Across Academic Year Levels

Source of Variation	Sum of Squares (SS)	Degrees of Freedom (df)	Mean Square (MS)	Computed F-Ratio	p-value	Interpretation
Between Groups	1.12	3	0.37	1.35	0.261	Not Significant
Within Groups	54.5	196	0.28			
Total	55.63	199				

Relationship Between AI Usage Profile and Information Verification Habits

This section examines whether a statistically significant relationship exists between the students' baseline AI usage profiles and their active information verification habits. To interpret these findings, a correlational matrix maps specific usage variables, namely the frequency of AI tool utilization, the primary type of tool preferred, and the nature of programming tasks performed, against the respondents' composite verification scores.

The correlational analysis reveals distinct insights into how different dimensions of artificial intelligence adoption influence students' active cross-checking behaviors. A statistically significant negative correlation was found between the primary type of tool utilized and the respondents' information verification habits ($r = -0.177$, $p = 0.012$).

Specifically, students who primarily rely on dedicated AI Integrated Development Environment (IDE) assistants, such as GitHub Copilot or Cursor AI, exhibit significantly lower composite verification scores (Mean = 2.94) compared to those who utilize general web-based conversational chatbots (Mean = 3.18). This provides strong empirical proof of automation bias within seamless inline environments; because dedicated IDE assistants provide automated code completions directly inside the text editor, the reduced friction of generation makes students significantly less likely to manually consult official documentation, verify package existence, or trace logic line-by-line. Conversely, the analysis reveals no statistically significant relationship between the overall frequency of AI usage and active verification habits ($r = 0.053$, $p = 0.453$), indicating that whether students use generative AI daily or only occasionally, their baseline auditing rigor remains relatively uniform. Similarly, no statistically significant correlation was observed between task complexity—such as generating entire code blocks from scratch versus debugging existing code—and verification scores ($r = -0.097$, $p = 0.170$). This confirms that students apply a consistent level of skepticism and cross-checking across different operational tasks, demonstrating that the immediate interface and delivery method of the AI tool, rather than the frequency or nature of the task itself, is the primary driver of cognitive over-reliance.

Table 13 Correlation Matrix Between AI Usage Profiles and Information Verification Habits

AI Usage Variable	Pearson's r	p-value	Statistical Significance	Direction of Relationship
Frequency of AI Usage	0.053	0.453	Not Significant	Positive
Primary Tool Type Utilized (Dedicated IDE Assistants vs. Web Chatbots)	-0.177	0.012	Significant	Negative
Nature of Tasks / Complexity (Generating Logic from Scratch vs. Others)	-0.097	0.17	Not Significant	Negative

F. Proposed "AI Auditing" Guidelines and Curriculum Updates

Table 14 Proposed "AI Auditing" Guidelines and Curricular Frameworks

Focus Area	Empirical Justification	Proposed Curriculum Update & Policy Action
Active Code Auditing	• 84.5% highly active generative AI users. • Significant negative correlation between dedicated IDE assistant usage and verification rigor ($r=-0.177$).	• Transition away from outdated restrictive or banning policies. • Embed formal "AI Auditing" modules directly into core programming courses.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

	• Vulnerability to seamless inline automation bias.	• Instruct students to systematically trace automated logic, inspect linting errors, and manually verify external dependencies.
External Validation Protocols	• Underlying instinct to prioritize external validation exists (manual web searches rank 1st at Weighted Mean = 3.22). • Internal AI self-verification ranks lowest (Weighted Mean = 2.94), demonstrating healthy skepticism.	• Mandate strict citation and validation protocols in course syllabi for all programming assignments. • Require inclusion of official documentation URLs or comment blocks detailing human-verified cross-checking sources. • Break recursive confirmation loops where models self-verify outputs.
Countering Presentation Bias	• False perception that AI possesses deep contextual understanding ranks highest (Weighted Mean = 3.20). • Explicit agreement that speed/convenience outweigh incorrect syntax risks (Weighted Mean = 3.11).	• Integrate foundational instruction on LLM architectural mechanics in introductory computing courses. • Explicitly teach that models operate on probabilistic pattern-matching rather than factual comprehension. • Dismantle perceptions of infallible intelligence to foster software engineering self-reliance.

Based on the empirical findings established in the preceding tables, actionable curricular frameworks and concrete institutional guidelines for vetting artificial intelligence hallucinations are synthesized below. Addressing the documented gaps in student verification methods is critical to mitigating the risks of automation bias within technical degree programs. To achieve this, institutions must first integrate formal "AI Auditing" modules directly into core programming curricula.

The empirical data establishes that 84.5% of the respondents are highly active users of generative AI, yet their verification habits remain highly vulnerable to inline automation bias, as proven by the statistically significant negative correlation between dedicated IDE assistant usage and verification rigor ($r = 0.177$). Rather than enforcing outdated restrictive policies, academic departments must transition to proactive, structured instruction by embedding modules that teach students how to systematically trace automated logic, inspect linting errors, and manually verify external dependencies instead of passively relying on generated outputs.

Furthermore, this instructional shift must be coupled with the establishment of mandatory external validation protocols. Findings from the domain-specific scenarios in Table 10 indicate that students already possess an

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

underlying instinct to prioritize external validation, ranking manual web searches to confirm library existence as their primary strategy (Weighted Mean = 3.22) while placing internal AI self-verification at the bottom (Weighted Mean = 2.94).

Academic departments should reinforce this documented skepticism by requiring course syllabi to mandate strict citation and validation protocols for all programming coursework. Specifically, instructors should direct students to include official documentation URLs or detailed comment blocks documenting the human-verified sources used to cross-check automated logic, thereby actively breaking recursive confirmation loops where fallible models are used to verify their own outputs.

Finally, technical curricula must place a strong emphasis on the underlying mechanics of Large Language Models (LLMs) to actively counter presentation bias. Results regarding perceived reliability in Table 11 demonstrate a significant cognitive vulnerability among students, as respondents highly rank the erroneous belief that AI possesses a deep, contextual understanding of concepts (Weighted Mean = 3.20) and explicitly agree that convenience and speed outweigh the risk of receiving incorrect syntax (Weighted Mean = 3.11).

To mitigate this manifestation of automation bias, introductory computing courses must incorporate foundational lectures detailing the architectural reality of generative models. Educating developing developers that these tools operate on probabilistic pattern matching rather than factual or conceptual comprehension directly dismantles the false perception of infallible intelligence, ultimately fostering a more rigorous, self-reliant, and independent approach to software engineering (Ji et al., 2023).

Synthesis of Baseline Technical Behaviors and AI Adoption

Table 15 Synthesis of Baseline AI Adoption and Operational Profiles

Operational Parameter	Empirical Metric Distribution	Core Finding & Analytical Implication
Highly Active AI Usage	84.50%	Combined total of students using AI "Always" (50.5%) and "Often" (34.0%); confirms generative assistance is deeply embedded in the modern academic routine.
Primary Tool Preference	58.50%	Utilizes general web-based conversational chatbots (e.g., ChatGPT, Claude) for flexible dialogue, broad explanations, and logic generation.
Secondary Tool Preference	41.50%	Utilizes dedicated AI Integrated Development Environment (IDE) assistants (e.g., GitHub Copilot) embedded directly within coding environments.
Primary Task Nature	45.50%	Applies AI primarily as an interactive diagnostic aid for debugging or finding specific syntax errors.

The empirical outcomes characterize a student population deeply engaged with automated assistance. Surveyed Bachelor of Science in Information Technology (BSIT) and Bachelor of Science in Computer Science (BSCS) students at Quezon City University for the Academic Year 2025–2026 operate as highly active users of generative artificial intelligence. Specifically, 84.5% of the respondents integrate these tools into their technical problem-solving workflows either daily or multiple times per week. This high frequency confirms that generative

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

assistance is firmly embedded in the modern academic routine. The respondents primarily utilize these platforms as interactive diagnostic aids, with debugging or finding specific errors representing the most prevalent programming application at 45.5%. Furthermore, the student ecosystem demonstrates a strong dual preference between flexible platforms and specialized coding environments: 58.5% primarily rely on general web-based conversational chatbots like ChatGPT or Claude, while 41.5% favor dedicated AI Integrated Development Environment (IDE) assistants such as GitHub Copilot.

Verification Rigor versus Inline Automation Bias

Table 16 Summary of Verification Habits, Scenario Success Rates, & Reliability Perceptions

Key Indicator	Statistical Value	Standard Interpretation & Narrative Context
Overall composite verification mean	Composite Mean = 3.13	Agree. Demonstrates structured auditing habits when encountering potential AI hallucinations.
Independent verification of new packages or libraries	Weighted Mean = 3.23	Agree. Ranked as the primary habit; students heavily prioritize checking external dependencies to prevent critical compilation failures.
Cross-checking suggested syntax directly against official documentation	Weighted Mean = 3.20	Demonstrates that students maintain a strong foundational reliance on authoritative primary sources.
Assuming new function or variable names are valid without checking	Weighted Mean = 2.88	Agree. Received the lowest verification mean, demonstrating that the convenience of instant code generation occasionally entices students to bypass granular manual vetting.
Manually searching the web to confirm library existence prior to compilation	Weighted Mean = 3.22	Ranks highest among domain-specific error detection scenarios, aligning directly with baseline dependency verification habits.
Asking the AI model to verify its own output	Weighted Mean = 2.94	Received the lowest score among error detection indicators. Proves a documented, healthy skepticism where learners actively recognize recursive error risks and avoid confirmation loops.
Correlation between primary type of tool utilized and active information verification habits	$r=-0.177 p=0.012$	Statistically Significant Negative Correlation. Provides strong empirical proof of automation bias driven directly by the specific interface utilized.

General web chatbots composite verification score	Mean = 3.18	Serves as the higher comparative baseline for active verification auditing.
Dedicated AI IDE assistants composite verification score	Mean = 2.94	Significantly lower composite verification scores compared to web chatbots. Generating completions directly within the text editor reduces friction, which directly suppresses manual auditing and makes students significantly less likely to trace logic line-by-line, verify package existence, or consult official documentation.

When encountering potential AI hallucinations (Ji et al., 2023), respondents display structured auditing habits, yielding an overall composite verification mean of 3.13, interpreted verbally as "Agree". Students heavily prioritize checking external dependencies to prevent critical compilation failures, ranking the independent verification of new packages or libraries as their primary habit with a weighted mean of 3.23. This aligns directly with their error detection success rates in domain-specific scenarios, where manually searching the web to confirm library existence prior to compilation ranks highest with a weighted mean of 3.22. Furthermore, students maintain a strong foundational reliance on authoritative primary sources, frequently cross-checking suggested syntax directly against official documentation. Conversely, asking the AI model to verify its own output received the lowest score among error detection indicators at 2.94. This proves a documented, healthy skepticism; learners actively recognize the risks of recursive errors and avoid confirmation loops where fallible models self-verify fabricated claims.

Despite these active auditing strategies, the results reveal persistent vulnerabilities to cognitive over-reliance (Horowitz & Kahn, 2024).Assuming new function or variable names are valid without checking received the lowest verification mean at 2.88, yet still falls within the affirmative "Agree" range. This demonstrates that the convenience of instant code generation occasionally entices students to bypass granular manual vetting. Most critically, the correlational analysis provides strong empirical proof of automation bias driven by the specific interface utilized. A statistically significant negative correlation exists between the primary type of tool utilized and active information verification habits ($r = -0.177$, $p = 0.012$). Respondents who rely primarily on dedicated AI IDE assistants exhibit significantly lower composite verification scores (Mean = 2.94) compared to those utilizing general web chatbots (Mean = 3.18). Because embedded IDE assistants generate completions directly within the text editor, the reduced friction directly suppresses manual auditing, making students significantly less likely to trace logic line-by-line, verify package existence, or consult official documentation (Mehra et al., 2025; Soares et al., 2025).

I. Perceptions of Reliability and the Trade-off Between Speed and Accuracy

Table 17 Perceptions of Reliability and Speed vs. Accuracy Trade-Offs

Indicator	Weighted Mean	Verbal Interpretation	Rank
Belief that AI possesses deep, contextual understanding of concepts	3.2	Agree	1st
Convenience and speed outweigh the risk of receiving incorrect syntax	3.11	Agree	2nd

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

Assuming eloquent explanations from AI models are factually correct	3.04	Agree	3rd
Trusting the AI model over a textbook or professor	2.92	Agree	8th
Overall Composite Reliability Mean	**3.03**	**Agree**	

The findings regarding perceived reliability yield an overall composite mean of 3.03, indicating that students attribute a high degree of authority to automated assistants. Respondents heavily overestimate the actual cognitive capabilities of pattern-matching models, ranking the erroneous belief that AI possesses a deep, contextual understanding of concepts as their highest perception with a weighted mean of 3.20 (Ji et al., 2023). Furthermore, direct empirical evidence establishes a critical trade-off: students explicitly agree that convenience and speed outweigh the risk of receiving incorrect syntax, ranking this indicator second with a weighted mean of 3.11. The immediate efficiency of automated generation frequently drives users to accept higher risks of encountering AI hallucinations. This reliance is compounded by presentation bias, as respondents assume eloquent explanations from AI models are factually correct (Weighted Mean = 3.04). Although trusting the AI model over a textbook or professor ranked last among the indicators (Weighted Mean = 2.92), its placement within the affirmative "Agree" range underscores a profound shift in technical education, where automated tools are increasingly viewed as highly authoritative sources alongside established academic expertise.

Uniformity Across Academic Stages and Operational Parameters

Table 18 Uniformity Across Academic Stages and Operational Parameters

Demographic Variable	Statistical Applied Test	Statistical Result Output	Significance Interpretation	Analytical Deduction & Impact
Academic Year Level	One-Way ANOVA	$F(3,196)=1.35$ $p=0.261$	Not Significant	Verification habits operate independently of accumulated academic experience or seniority; composite means remain highly uniform from the 1st to 4th year.
Frequency of AI Tool Utilization	Pearson's r	$r=0.053$ $p=0.453$	Not Significant	Baseline auditing rigor remains consistent regardless of how often a student engages with generative tools.
Nature of Task Performed	Pearson's r	$r=-0.097$ $p=0.170$	Not Significant	Skepticism and cross-checking practices remain uniform regardless of programming task complexity.
Interface Delivery Method (Seamless Inline vs. Web Chatbot)	Correlational Isolation	Mapped via Interface Bias ($r=-0.177$)	Primary Operational Driver	Isolates seamless inline code generation within the text editor as the central operational driver suppressing manual auditing.

Inferential analysis confirms that verification habits operate independently of a student's accumulated academic experience or seniority within the computing program. A One-Way Analysis of Variance (ANOVA) comparing cross-checking habits across academic year levels revealed no statistically significant difference ($F(3, 196) =$

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

1.35, $p = 0.261$). The composite verification means remain relatively uniform from the first year to the fourth year. This establishes that the propensity to actively audit outputs or succumb to automation bias is a uniform behavioral trait across all academic stages. Moreover, baseline auditing rigor remains consistent regardless of how often a student uses AI or the complexity of their programming task. Neither the frequency of AI tool utilization ($r = 0.053$, $p = 0.453$) nor the nature of the task performed ($r = -0.097$, $p = 0.170$) exhibits a statistically significant relationship with verification scores. These outcomes isolate the interface delivery method—specifically seamless inline generation—as the primary operational driver of cognitive over-reliance among technical students.

Implications for Proactive Curriculum Integration

Table 19 Actionable Frameworks for Proactive Curriculum Integration

Curriculum Focus Area	Documented Empirical Vulnerability	Proposed Institutional Framework & Policy Action
Institutional Policy Shift	84.5% of the student population actively relies on generative AI while remaining highly susceptible to inline automation bias.	Transition away from outdated restrictive or banning policies toward proactive, structured classroom integration.
Core Course Instruction	Suppressed manual auditing caused by inline IDE generation friction drops.	Embed formal "AI Auditing" instruction directly within core programming courses to systematically teach logic tracing, linting inspection, and manual dependency verification.
Validation Protocols	Students exhibit an underlying instinct to prioritize external validation over internal AI confirmation.	Mandate strict citation protocols in course syllabi, directing developers to include official documentation URLs or comment blocks to actively break recursive self-verification loops.
Mechanics Education	Susceptibility to presentation bias and assuming models possess deep contextual understanding.	Incorporate foundational instruction on the probabilistic pattern-matching mechanics of Large Language Models (LLMs) in introductory courses to dismantle false perceptions of infallible intelligence.

The compiled outcomes designate an urgent need for institutional frameworks that address these documented vulnerabilities. Because 84.5% of the student population actively relies on generative AI while remaining highly susceptible to inline automation bias, academic departments must transition away from outdated restrictive policies. Instead, curricula must proactively integrate formal "AI Auditing" instruction embedded directly within core programming courses. These instructional updates should teach students to systematically trace automated logic, inspect linting errors, and manually verify external dependencies (Soares et al., 2025). Institutions can capitalize on the students' documented underlying instinct to prioritize external validation over internal AI confirmation by mandating strict citation protocols in course syllabi. Directing developers to include official documentation URLs or detailed comment blocks breaks recursive confirmation loops where models verify their own outputs. Finally, introductory courses must explicitly educate students on the underlying probabilistic pattern-matching mechanics of Large Language Models (LLMs) to dismantle false perceptions of infallible intelligence, fostering a rigorous and self-reliant approach to software engineering (Ji et al., 2023).

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

CONCLUSION

The primary purpose of this study was to evaluate the information verification habits of Computer Studies students when encountering artificial intelligence hallucinations, specifically determining how their operational usage profiles influence active verification methods, error detection success rates, and perceived tool reliability. Moving from specific baseline characteristics to broader implications, the findings established that the student population consists predominantly of traditional college-aged, full-time learners who operate as highly active, regular users of generative artificial intelligence. Within their operational workflows, students demonstrate a dual preference between general web-based conversational chatbots and dedicated Integrated Development Environment (IDE) assistants, treating these platforms primarily as interactive diagnostic aids for debugging errors and accelerating new code generation.

In terms of cross-checking behaviors, the study proved that students actively employ structured verification methods and maintain a solid foundation in external information literacy. When encountering potential technical errors, respondents heavily prioritize independent validation, such as manually confirming the existence of imported libraries and cross-referencing syntax against official documentation, rather than trusting fallible models to recursively self-verify their own outputs. However, this cross-checking rigor is simultaneously compromised by a persistent vulnerability to automation bias. Students generally attribute a high degree of authority to automated tools, frequently overestimating the models' underlying capabilities by assuming they possess deep, contextual comprehension. Most critically, a fundamental behavioral trade-off was established, revealing that the immediate efficiency and convenience of instant generation frequently entice developers to accept higher risks of receiving incorrect syntax.

Inferential analysis conclusively established that active auditing habits operate entirely independent of a student's accumulated academic experience or seniority within the computing program. Because verification rigor remains uniform across all year levels, susceptibility to hallucinations represents a systemic, department-wide challenge rather than an isolated, novice-level deficiency. Furthermore, baseline skepticism remains consistent regardless of overall usage frequency or programming task complexity. Instead, a significant negative relationship was proven to exist between the primary interface utilized and active verification habits. Seamless inline completions generated directly inside dedicated IDE assistants reduce operational friction, which actively suppresses manual auditing and makes students significantly less likely to trace logic line-by-line or consult authoritative external documentation.

Ultimately, the overall contribution of this study lies in proving that cognitive over-reliance is driven primarily by interface delivery friction rather than student experience, demonstrating that outdated restrictive policies banning generative tools are fundamentally misaligned with modern developmental practices. The practical value of this work is evidenced by its direct justification for proactive curricular innovations. Specifically, it establishes an urgent institutional mandate to integrate formal artificial intelligence auditing modules directly into core programming courses, enforce strict external validation and citation protocols in course syllabi, and incorporate foundational instruction on probabilistic model mechanics to dismantle false perceptions of infallible intelligence, thereby cultivating highly rigorous and self-reliant software engineers.

REFERENCES

1. An exploration of online learning habits and academic productivity among BSIT students. (2025). International Journal of Research and Scientific Innovation (IJRSI), 12(10), 3530–3544. https://rsisinternational.org/journals/ijrsi/uploads/vol-iss-pg3530-3544-202510_pdf.pdf
2. Babbie, E. R. (2020). The practice of social research (15th ed.). Cengage Learning.
3. Bluman, A. G. (2018). Elementary statistics: A step-by-step approach (10th ed.). McGraw-Hill Education.
4. Calderon, J. F., & Gonzales, E. C. (1993). Methods of research and thesis writing. National Book Store.
5. Creswell, J. W., & Creswell, J. D. (2018). Research design: Qualitative, quantitative, and mixed methods approaches (5th ed.). SAGE Publications.
6. Field, A. (2018). Discovering statistics using IBM SPSS Statistics (5th ed.). SAGE Publications.
7. Flick, U. (2018). An introduction to qualitative research (6th ed.). SAGE Publications.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue V, May 2026

8. Fowler, F. J. (2013). Survey research methods (5th ed.). SAGE Publications.
9. Göksel, N., & Akgül, M. (2021). Social media trust and verification habits of undergraduate students. ERIC Document Reproduction Service. https://files.eric.ed.gov/fulltext/EJ1395193.pdf
10. Gravetter, F. J., & Wallnau, L. B. (2017). Statistics for the behavioral sciences (10th ed.). Cengage Learning.
11. Horowitz, M. C., & Kahn, L. (2024). Bending the automation bias curve: A study of human and AI-based decision making in national security contexts. International Studies Quarterly, 68(2), Article sqae020. https://doi.org/10.1093/isq/sqae020
12. Information literacy: How students search for & verify health information. (2025). Honors in Communication and Language Studies, Bryant University. https://digitalcommons.bryant.edu/cgi/viewcontent.cgi?article=1001&context=honors_communicationlanguagestudies
13. Information verification among undergraduates. (2025). PhilArchive. https://philarchive.org/archive/DAGPSH
14. Ji, Z., et al. (2023). Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12), 1–38. https://doi.org/10.1145/3571730
15. Joshi, A., Kale, S., Chandel, S., & Pal, D. K. (2015). Likert scale: Explored and explained. British Journal of Applied Science & Technology, 7(4), 396–403. https://doi.org/10.9734/BJAST/2015/14975
16. Likert, R. (1932). A technique for the measurement of attitudes. Archives of Psychology, 140, 1–55.
17. Mehra, V., et al. (2025). Quantifying shortfalls in students' AI-supported programming practices. International Information Systems Conference (IIS), 458–471. https://iacis.org/iis/2025/1_iis_2025_458-471.pdf
18. Navigating the infodemic: A qualitative study of university students' misinformation vetting. (2024). BMC Public Health, 24(1), 1–12. https://pmc.ncbi.nlm.nih.gov/articles/PMC10829486/
19. Pallant, J. (2020). SPSS survival manual: A step by step guide to data analysis using IBM SPSS (7th ed.). Routledge.
20. Rowley, J. (2014). Designing and using research questionnaires. Management Research Review, 37(3), 308–330. https://doi.org/10.1108/MRR-02-2013-0027
21. Shoufan, A., & Esmaeil, A. (2026). AI hallucination from students' perspective: A thematic analysis. arXiv. https://doi.org/10.48550/arXiv.2602.17671
22. Soares, F. A., Franco, M. F., Scheid, E. J., & Granville, L. Z. (2025). Trust, but verify: An empirical evaluation of AI-generated code for SDN controllers. arXiv. https://doi.org/10.48550/arXiv.2510.20703
23. Taherdoost, H. (2016). Validity and reliability of the research instrument: How to test the validation of a questionnaire/survey in a research. International Journal of Academic Research in Management, 5(3), 28–36.
24. The hallucination effect: Correlating generative AI usage frequency with source verification habits among STEM students. (2024). IMCC Journal. https://myjournal.imcc.edu.ph/publication/the-hallucination-effect-correlating-generative-ai-usage-frequency-with-source-verific
25. UiTM Generation Z study. (2025). Universiti Teknologi MARA.