NOVA IMS Information Management School

MGI

Master Degree Program in
Information Management

Understanding the determinants of Developers' satisfaction and adoption of Copilots

Margarida Belo Carrilho Maurício da Costa

Master Thesis

presented as partial requirement for obtaining the Master Degree in Information Management

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

Understanding the determinants of Developers' satisfaction and adoption of Copilots

by

Margarida Belo Carrilho Maurício da Costa

Master Thesis presented as partial requirement for obtaining the Master's degree in Information Management, with a specialization in Information Systems and Technologies Management

Supervised by

Professor Mijail Naranjo, Ph.D., NOVA Information Management School

May, 2024

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

[Lisbon, 30th May 2024]

ABSTRACT

Leveraged by recent advances in generative models, Artificial Intelligence assistants, such as GitHub Copilot, promise to change the way developers do their work, and in consequence increase human's productivity. This thesis proposes a theoretical model based on the unified theory of acceptance and use of technology (UTAUT) to identify and investigate the determinants influencing user satisfaction and actual use of Copilots. To perform the quantitative research, a survey destined to programming students and developers was developed, collecting a total of 144 valid answers. The partial least squares-structural equation modelling (PLS-SEM) was applied to explore the data and assess the research model.

The results suggest that 52% of user satisfaction can be explained by both perceived learning and perceived productivity. Additionally, the proposed model highlights that 53% of use and adoption of copilots is explained by facilitating conditions and user satisfaction. The results provide significant insights in the adoption of Copilots in the software engineering field as programmers use the tool mainly for productivity reasons.

KEYWORDS

Copilot; Generative AI; AI Adoption; Satisfaction; UTAUT; Structural equation modeling

Sustainable Development Goals (SDG):

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

A	Age
AI	Artificial Intelligence
AU	Actual Use
AVE	Average Variance Extracted
CQ	Code Quality
DIT	Diffusion of Innovative Theory
EE	Effort Expectancy
FC	Facilitating Conditions
GEN-AI	Generative Artificial Intelligence
GPT	Generative Pre-trained Transformer
GP	Genetic Programming
HTMT	Heterotrait-Monotrait
IS	Information Systems
IT	Information Technology
LLM	Large Language Models
ML	Machine Learning
NLP	Natural Language Processing
PL	Perceived Learning
PLS-SEM	Partial Least Squares-Structural Equation Modelling
PP	Perceived Productivity
SI	Social Influence
TAM	Technology Acceptance Model
US	User Satisfaction
UTAUT	Unified Theory of Acceptance and Use of Technology
VIF	Variance Inflation Factor

1. INTRODUCTION

1.1. PROBLEM STATEMENT

Artificial Intelligence (AI) is emerging, and in the last years has become an important subject between individuals and organizations. At the same time, code-generating models are starting to change the way developers do their job (Barke et al., 2023), and in consequence, promise to increase human's productivity. In 2023, based on a study performed by GitHub, the Artificial Intelligence Index Report by Stanford University published that developers who used Copilot completed a task 56% more rapidly than the developers who did not use Copilot (Maslej et al., 2023).

Copilots, such as GitHub released in 2021, are a recent tool powered by a generative AI model, Codex (M. Chen et al., 2021) a descendant of GPT-3 (T. B. Brown et al., 2020), and Machine Learning (ML) models. This AI tool was developed as an extension for the Visual Studio Code to help developers programming, by generating code suggestions based on the context of what they are working on.

With rising pressure on software developers to create code quickly, the development and appearance of this kind of tools will probably have a huge influence on programmers' daily work performance and productivity. If for developers it is normal to work in pair-programming (a common technique where two programmers work together on a task, one of them writes the code (driver) and the other reviews it and provides feedback (navigator)), now, Copilots could be a replacement for pair-programming.

In this dissertation, a research about software developers' experience with GitHub Copilot will be made, with the objective to analyze the impact of the tool in their work, as well as understand what are the principal factors that influence the user satisfaction with the tool and potential measures that could be implemented to turn the tool more interesting.

1.2. RESEARCH GAP AND OBJECTIVES

As the majority of the existing studies on this subject rely on measuring the correctness of Copilot's code solutions and emphasize its problems, as well as comparing the solution generated from Copilot with human generated solutions. This work aims to do a not so technical study by building an understanding of developers' experience and attitude towards Copilot, namely, learning the usefulness of this tool in their daily work, what encourage them to use it and how they see the influence of Copilots for programming on their productivity.

For that, a Research Question was raised: "What are the elements that influence programmers' satisfaction and when using Copilots?"

With the aim to respond to the research question and to address the main objective of creating a conceptual model, it is necessary to achieve the subsequent objectives:

1. Understand which factors influence developers to use Copilots for programming
2. Propose a theoretical model to explain the elements that influence programmers' satisfaction and use of Copilots
3. Test and validate the model empirically

2. LITERATURE REVIEW

Copilots are a recently new solution released a couple of years ago, and likewise there isn't much research conducted on this topic. However, over the years, some studies in the artificial intelligence field, concerning generative AI, have been conducted. Also, studies with the aim of improving developers' productivity with artificial intelligence have been developed.

2.1. DEFINITION OF COPILOT

Since producing code is getting more complex and demanding despite frequently necessary, programmers have been thinking of an AI assistant that could work alongside with them and help them pair-programming. Due to the breakthroughs in large language models (LLM), and more specifically in Generative AI, this assistant finally arrived to transform the developers' experience.

As mentioned before, GitHub Copilot is powered by generative AI models developed by GitHub, OpenAI, and Microsoft. Copilot is accessible as an extension for, among others, Visual Studio Code, Visual Studio, JetBrains IDEs, and Neovim (Friedman, 2021; Rosenkilde, 2023). Generative AI has been around for several years, and it is defined as a subset of AI that is able to generat text, images, videos and music. (Totlani, 2023). Generative AI, namely GEN-AI tools such as GitHub Copilot and ChatGPT, have the potential to considerably improve the software development field by automating monotonous tasks, enhancing code quality, and improving accuracy and efficiency.

GitHub Copilot can be considered as an AI pair programmer that helps programmers by offering autocomplete ideas as they code or write comments. The suggestions appear as developers start writing code, or by writing a comment explaining what the code is supposed to do. This tool aims to help programmers to quickly learn new alternatives to solve problems by reducing developers' work and increasing their productivity.

Moreover, it is important to state that GitHub continues to evolve and works hard to bring new features that contribute to a more personalized developer experience. In December 2023, GitHub Copilot adopted OpenAI's GPT-4 (OpenAI, 2023) and introduced chat and voice (Dohmke, 2023). The Copilot is becoming a ChatGPT-like experience directed to developers.

2.2. RELATED WORKS

Some studies were already conducted to analyze how programmers interact and benefit from Copilots and to explore its different functions and capabilities. Most of them studied this topic by assessing if the copilot suggested code is correct, analyzing the code quality and compare it to human code, and understand how programmers behave while using the tool (how they resolve proposed tasks and how it improved productivity).

Barke et al. (2023) analyzed how developers cooperate with code-generating models. Finding suggested that the interactions with Copilot are bimodal, with programmers using Copilot in acceleration mode when they know what they have to do, and in exploration mode when they are uncertain how they should proceed.

Moradi Dakhel et al. (2023) presented a study about the performance and functionality of Copilot's suggestions and compared them with human solutions. The findings express that Copilot is capable of deliver solutions for most of the problems, but some of them are buggy. Moreover, Copilot struggles in merging different methods to generate a solution. At the end, the authors concluded that humans' solutions are often more correct than Copilots'. Finally, the study shows that the programming experience is an important aspect once that Copilot can turn an asset when used by expert developers, and a liability when used by beginner developers who may have difficulties evaluating the code correctness.

Sobania et al. (2022) compared the program synthesis performance of GitHub Copilot with genetic programming (GP) on standard program synthesis problems. It was concluded that despite GP approaches taking more time to generate solutions and are more expensive, both approaches perform similar.

Vaithilingam et al. (2022) studied the usability of Copilot by observing how the 24 programmers performed the three given programming tasks with different stages of difficulty (1. Edit CVS; 2. Web Scrapping; 3. Graph Plotting). The majority of the participants preferred using Copilot, however it did not lower the task completion time neither increased the success ratio of solving programming tasks.

Wermelinger (2023) chose to study how does the Copilot performs, compared with Davinci, and see the implications and limitations when the users are programming students tasked with writing, explaining and fixing code. The findings of the study were that Copilot does less well compared to Davinci, once the suggestions given by it are often wrong.

Some papers also analyzed GitHub Copilot as an AI-powered pair programming tool (Bird et al., 2023; Imai, 2022). Bird et al. (2023) conducted three different studies to understand how developers use Copilot (first, an analysis of GitHub forum discussions; then, a case study where 5 expert Python programmers used Copilot for the first time; lastly, it was performed an analysis of 2047 survey answers from programmers). Through these studies, the authors concluded that despite developers spending more time reviewing code than actually writing,

users perceive themselves to be more productive when accepting suggestions from Copilot. Imai (2022), on the other hand, compared the effectiveness of pair programming with GitHub Copilot to human pair programming in terms of code productivity and code quality. The results show that pair-programming with Copilot is totally different of human pair-programming. Moreover, on average, code produced by Copilot has inferior quality when compared to the code produced by human pair-programming.

2.3. GENERATIVE AI

With the development of Gen-AI assistants, many students and educational professionals have shown increased interest in using these tools for educational proposes. In so, most researchers focused on the importance of this field in education.

Almufarreh (2024) presented insights about which factors affect students' general satisfaction with AI tools. Wang et al. (2023) studied how AI can improve learning efficiency and provide educational support. The study also identified possible risks and limitations as privacy and ethical concerns. (Yilmaz & Yilmaz, 2023) stated that by using ChatGPT for programming learning, students can deliver faster and mostly corrected answer, however, it also triggered laziness and anxiety among students. L. Chen et al. (2020), affirmed that instructors have been more effectively and efficiently in their job since they use AI tools. Also, as AI enables a more personalized content the learning process of students have been improved. Chatterjee & Bhattacharjee (2020) chose to study, based on the UTAUT model, the factors that influence the adoption of AI in higher education in India.

However, though these studies deliver appreciated considerations on how users perceive Gen-AI tools, these insights are about students or education professionals. For this master thesis, the main focus is to study how programmers use and adopt Gen-AI tools, more specifically Copilot.

2.4. ADVANTAGES AND CONCERNS

Analyzing the studies presented above, there is not a unanimous conclusion of the advantages and benefits that Copilot offers, also due to the difficulty to measure productivity and code quality (Murphy-Hill et al., 2021).

Based on 2000 responses of developers about their experience using Copilot, GitHub identified that 88% of surveyed respondents felt more productive, 74% were able to focus on more satisfying work, and 88% claimed to complete tasks more quickly (Kalliamvakou, 2022; Maslej et al., 2023). Developers who use Copilot reported to be 55% more productive at coding than the developers who did not use GitHub Copilot, and up to 75% more fulfilled with their jobs.

On the other hand, one of the major problems and concerns is regarding the security of this tool. Since the Copilot is trained on public code repositories of GitHub, there is a high probability of problems and vulnerabilities (Asare et al., 2023), for example, there is a possibility that the model was learned on buggy and biased code.

Pearce et al. (2022) chose to investigate if Copilot generates insecure code. For that, there were created diverse scenarios on high-risk cybersecurity problems. The conclusions show that Copilot produces insecure code about 40% of the time, and for that reason, developers should remain "awake" when using this tool. Furthermore, according to the 2023 Open Source Security and Risk Analysis report by Synopsys, 84% of the 1703 codebases examined contained at least one vulnerability, and 48% contained high-risk vulnerabilities (*Open Source Security and Risk Analysis 2023*, 2023).

Liang et al. (2024) analyzed 410 developers' responses to understand the usability challenges related to the use of AI programming assistants. The conclusions of this study express that despite the participants appreciating the autocompletion capabilities, as well as the fact that help them to finish tasks faster, they generally have concerns about privacy and security problems. AI programming assistants can produce code that may infringe on intellectual property and this AI tools can have access to users' code.

To summarize, there are different conclusions about the impacts of Copilots in software engineering. If, on one side, there are possible advantages, such as improving productivity and the output quality of code, that can revolutionize developers' experience. On the other side, there exist some ethical aspects, as privacy and security concerns, that leverage the consideration of users to implement this tool.

3. THEORETICAL MODEL BUILDING

Over the years, many models and frameworks have been developed with the aim to study information technology (IT) acceptance. Some examples of these models are the technology acceptance model (TAM), the diffusion of innovative theory (DIT), and the unified theory of acceptance and use of technology (UTAUT) (Taherdoost, 2019).

From the existing frameworks, UTAUT is one of the most significant adoption models used in information systems, and for that it will be used as a guiding framework to help provide a better understanding of developer's adoption and use behaviors towards Copilot. UTAUT resulted from the combination of eight different frameworks, and proposed four main factors (performance expectancy, effort expectancy, social influence, and facilitating conditions) (Venkatesh et al., 2003). Years later, three other constructs were incorporated to this model (hedonic motivation, price value, and habit), extending UTAUT into UTAUT 2 (Venkatesh et al., 2012).

For this work, despite the standard constructs, four more were added (user satisfaction, code quality, perceived learning and age) with the aim of completing the model. Being UTAUT itself developed based on a review of existing technology acceptance models, it offers a flexibility to incorporate other factors that may help to deliver a more comprehensive understanding of the model (Venkatesh et al., 2016).

Additionally, thirteen hypotheses were settled to further investigate how they influence the developer's intention to use Copilot. The proposed model for this study was design, as illustrated in Figure (1).

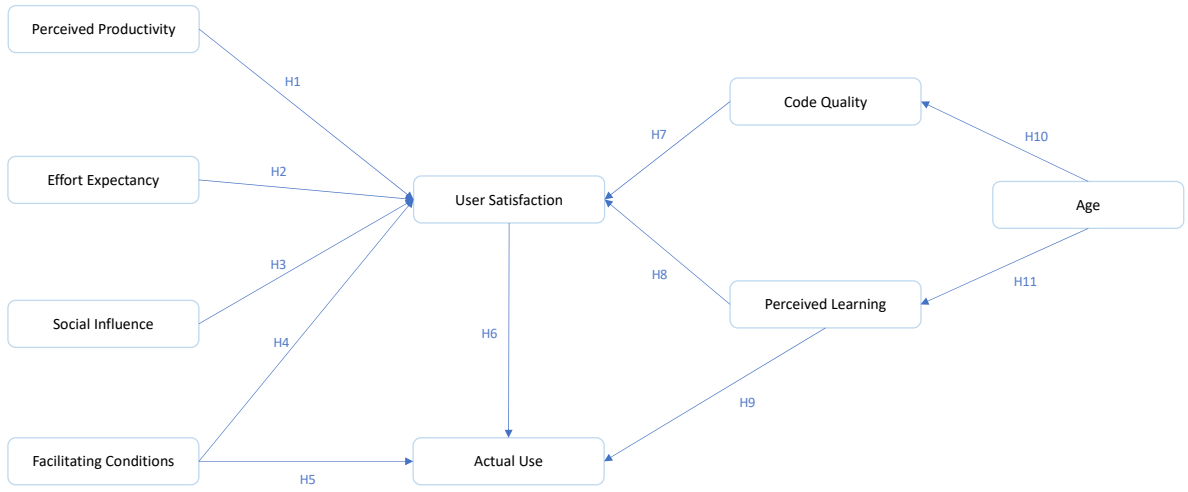

Figure 1 - Research Model

3.1. PERCEIVED PRODUCTIVITY

Perceived Productivity (PP), or Performance Expectancy as it is called in the original model of UTAUT, is designated as the level to which a person considers that using a system will help to achieve additional benefits in job performance (Venkatesh et al., 2003). This factor relates to the advantages that developers would achieve from using a new tool, in this case, Copilot, in their daily work. By being able to complete more tasks, or tasks more quickly, productivity will be affected.

The literature showed that PP has been significantly impacting the individual's satisfaction and continued use in IS field (Joo et al., 2017; Li & Fang, 2019; Niu & Mvondo, 2024). In addition, it is expected that Copilot can provide advantages and positive outcomes related to perceived productivity. In so, this hypothesis is proposed:

Hypothesis 1: *Perceived Productivity has a positive effect on user satisfaction.*

3.2. EFFORT EXPECTANCY

Effort Expectancy (EE) states the level of ease related with using a system (Venkatesh et al., 2003). It is concerned about the ease of using Copilot or, with other words, the effort employed by developers when using the tool to resolve programming tasks. It is likely that effort expectancy will have an impact on the satisfaction of users with a new technology once it may require some extra effort to learn how to operate the tool. Previous studies assumed contradictory findings. Hong et al. (2011) identified a connection between effort expectancy and users' satisfaction on Agile Information Systems. In the context of m-commerce, Chong (2013) found EE impact to be insignificant on consumer satisfaction, and S. Chen et al. (2009) also determined that satisfaction is positively influenced by ease of use. Thus, the following hypothesis was settled:

Hypothesis 2: Effort Expectancy has a positive effect on user satisfaction.

3.3. SOCIAL INFLUENCE

This factor reflects the importance that a person gives to what others consider he/she should use the new tool (Venkatesh et al., 2003). In the context of this thesis, social influence addresses the extent to which the society, friends and most important work colleagues, accept and incentive using Copilot. Thus, as social influence can explain certain choices and attitudes, this can be considered one key variable when exploring developer's aim to use Copilot. Hsiao et al. (2016) researched and confirmed that social ties have a meaningful influence on satisfaction. Consequently, the following hypothesis was developed:

Hypothesis 3: *Social Influence has a positive effect on user satisfaction.*

3.4. FACILITATING CONDITIONS

Facilitating conditions (FC) refer to the level a person considers that exists an organizational and technical foundation to assist the use of Copilot (Venkatesh et al., 2003). This measure addresses the developers' aptitude to use Copilot, as well as the conditions related to infrastructure, such as internet and computer. Studies using the UTAUT framework, support the facilitating conditions may positively affect intention to use and actual use (Chauhan & Jaiswal, 2016; Palau-Saumell et al., 2019). The facilitating conditions also reflect the compatibility with work tools, lifestyle and methods, in so, as it also represents user evaluations of the usage environment it is important to test the connection between FC and user satisfaction (Chan et al., 2011).

Hypothesis 4: *Facilitating Conditions has a positive effect on user satisfaction.*

Hypothesis 5: *Facilitating Conditions has a positive effect on actual use.*

3.5. USER SATISFACTION

This factor refers to users' overall experience of using a system (Hong et al., 2011), and it plays an essential role in determining users' attitudes toward and continued use of a technology. The meeting of expectations is an essential requirement to achieve user satisfaction, then, when users accept the tool, there is a high probability they integrate it into their routine. User satisfaction plays an important role in motivating technology continuous use (Delone & McLean, 2003; Hong et al., 2011; D. Jaiswal et al., 2022; Urbach & Müller, 2012). Thus, the following hypothesis was settled:

Hypothesis 6: *User satisfaction has a positive effect on actual use.*

3.6. CODE QUALITY

Code quality contributes to users' intention to use Copilot. When the tool consistently generates high-quality and accurate code suggestions, users are more likely to rely on it for assistance in their coding tasks. For instance, this dimension addresses if the output generated by the tool is as reliable as the average human code.

We can see code quality, as the quality of the output generated by Copilot. TAM 3 posits that output quality is related to the level of a person considers that the system performs job tasks well (Venkatesh & Bala, 2008; Venkatesh & Davis, 2000). Additionally, the information systems success model posits that user satisfaction is explained by the quality of information as an output of an IS (Laumer et al., 2017). When the retrieved information is not pertinent to a task or is hard to comprehend, individuals might follow substitute techniques to obtain the

information needed, otherwise, if Copilot performs human like tasks in an accurate, reliable and understandable manner, this will probably increase the user satisfaction with the tool. Therefore, this aims to examine the following hypothesis.

Hypothesis 7: *Code quality has a positive effect on user satisfaction.*

3.7. PERCEIVED LEARNING

Assessing whether developers recognize Copilot as a tool for learning and skill enhancement is important. Jaiswal et al. (2021) operationalized upskilling as a process of learning new skills, this concept is similar to the proposed of perceived learning. If users see Copilot as a mean to improve their coding proficiency and learn new concepts and new ways of writing code, it can positively influence users' satisfaction and intention to use the tool. This construct pretends to measure the programmers' perception of Copilot as a learning tool. Copilot can generate code and provide explanations about it, which allow users to receive constant feedback. It also opens a door to obtain more knowledge and discover different ways to achieve the same output. Consequently, the succeeding hypotheses were established:

Hypothesis 8: *Perceived learning has a positive effect on user satisfaction.*

Hypothesis 9: *Perceived learning has a positive effect on actual use.*

3.8. AGE

Even though technology is more present than ever in everything we do, older users are less likely to enjoy and adopt new technologies. This happens because of the difficulty that is to start learning something new from scratch. Additionally, older users might limit the information they use in their decision-making and mostly rely on their personal opinions and knowledge (Chattaraman et al., 2019). Younger users are more technologically enthusiastic compared to older users (Venkatesh et al., 2003). In so, as older users might not be disposed to use Copilot and learn with it, the study aims to investigate if the age impacts the code quality opinion and the perceived learning. The association among age and technology is typically negative, denotation that as the age of users' increases, their negative approaches concerning technology also rise. Therefore, the following hypotheses were settled:

Hypothesis 10: *Age has a negative effect on code quality.*

Hypothesis 11: *Age has a negative effect on perceived learning.*

4. METHODOLOGY

4.1. PROCEDURES

In order to collect the necessary data to evaluate the research model, an online questionnaire was developed within the Qualtrics platform. As seen in Appendix A, before distribution, the questionnaire was submitted to the NOVA IMS Ethics Committee, and therefore approved. It was assured and informed to all participants that the collected data was exclusively treated for academic purposes, in a totally anonymous approach. The questionnaire was available in two versions: English, and Portuguese.

The constructs presented in the survey were mostly adapted from the literature and measured using a seven-point numerical scale, ranging from "1 – Strongly disagree" to "7 – Strongly agree", according to the participants' level of understanding about each statement, except for the User Satisfaction construct that was measured from "1 – Extremely Negative" to "7 – Extremely Positive" and the Actual Use construct, whose scale was ranged from "1 – Never" to "7 – Always".

For the purpose of this study, a total of 237 answers from developers with different programming experience levels were examined. The survey consisted mainly of questions associated with the general attitude of developers to the use of Copilot, their perceived productivity and perceived learning about this tool and their satisfaction by using it.

As mentioned before, some of the measurement items were slightly modified from the literature considering the goals and context of this study, while others were self-developed. The constructs with the respective sources are exhibited in the table below.

Table 1 - Measurement items and sources

Construct	Statement	Reference
Perceived Productivity	PP1. Using Copilot is useful in my work PP2. Using Copilot improves my job performance PP3. Using Copilot improves my productivity PP4. Using Copilot enables me to accomplish tasks more quickly	(Martins et al., 2014; Venkatesh et al., 2003)
Effort Expectancy	EE1. I find Copilot easy to use EE2. My interaction with Copilot is clear and understandable EE3. It would be easy for me to become skillful at using Copilot EE4. Learning to operate the system is easy for me	(Venkatesh et al., 2003)

	EE5. I understand the suggestions Copilot gives me	
Social Influence	SI1. My colleagues think I should use Copilot SI2. I use Copilot because my coworkers use it SI3. My peers use Copilot frequently SI4. In general, the organization I work for has supported the use of Copilot	(Venkatesh et al., 2003)
Facilitating Conditions	FC1. I have the resources necessary to use Copilot FC2. I have the knowledge necessary to use Copilot FC3. Copilot is compatible with other systems/tools I use FC4. Using Copilot fits into my work style	(Venkatesh et al., 2003)
Code Quality	CQ1. Copilot code quality is better than the average human code CQ2. Copilot code is consistent CQ3. Copilot code is reliable	Personal development
User Satisfaction	US1. My overall satisfaction of Copilot is: Extremely Negative to Extremely Positive	(Chan et al., 2011; Hong et al., 2011)
Perceived Learning	PL1. I learn programming when using Copilot PL2. I learn more programming by using Copilot rather than with peers PL3. My programming skills have improved since I use Copilot	Personal development
Actual Use	AU1. How frequently do you use Copilot? Never to Always	(S. A. Brown et al., 2010)

4.2. DATA COLLECTION

To ensure the success and viability of the survey, it was shared across different social media platforms, namely in Facebook groups and LinkedIn. In total, 237 individuals participated in the survey, that was opened between April and the end of June 2024. Around 39% of the responses were deleted due to incompleteness, remaining 144 (about 61%) valid responses.

Approximately 61% of the participants were men, 23% aged between 22 and 25 years old. Furthermore, 7% of the respondents admitted that they never use the tool. More detailed characteristics about the respondents are shown in Table 2.

The data evaluation and analysis were made throughout the use of partial least squares-structure equation modeling (PLS-SEM), which is a common approach in the information

systems field. PLS-SEM is a causal modeling method aimed to test and explain the association between dependent and independent variables (Hair et al., 2011).

Table 2 - Sample Characteristics

Sample Characteristics	Percentage (%)	Frequency (n=144)
Gender		
Male	61,1	88
Female	37,5	54
Prefer not to say	1,4	2
Age		
18 - 21	15,3	22
22 - 25	29,9	43
26 - 29	10,4	15
30 - 33	11,8	17
34 - 37	8,3	12
38 - 41	4,9	7
42 or more	19,4	28
Programming Experience (Years)		
0 - 1	19,4	28
1 - 2	17,4	25
2 - 3	22,2	32
3 - 4	5,6	8
5 or more	35,4	51

The participants were questioned about which programming languages they normally use. They could choose more than one option and write additional languages not listed. By analyzing Figure 2, it is observed that Python and SQL are the most used languages by programmers, followed by JavaScript and Java.

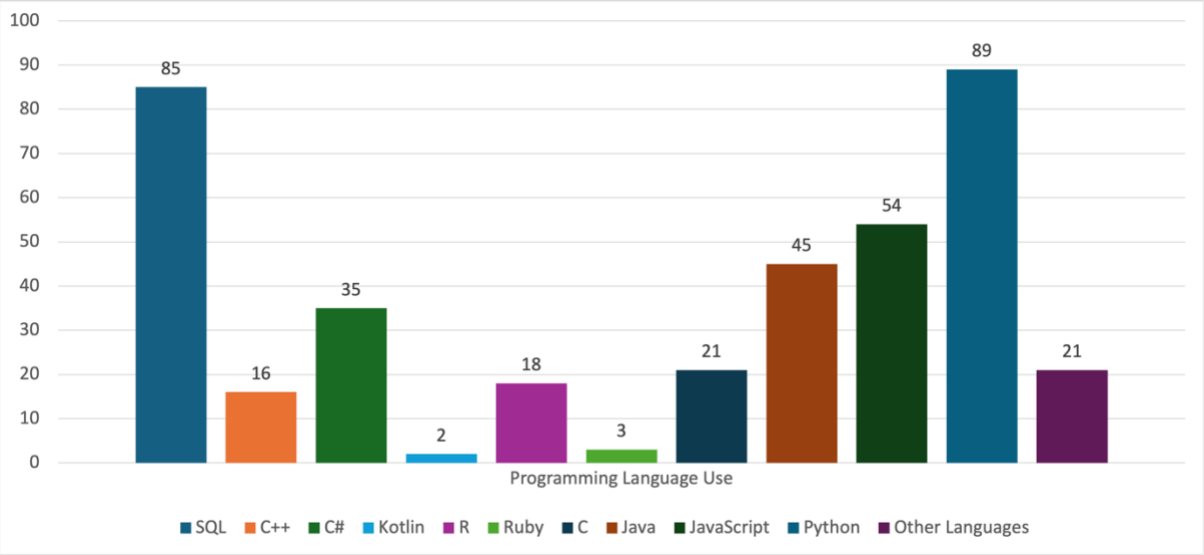

Figure 2 - Programming Languages

4.3. MEASUREMENT MODEL RESULTS

The SEM-PLS method was used to examine the model proposed in this thesis. The measurement model was performed and tested to evaluate the reliability and construct validity. In the succeeding paragraphs, the results of this analysis will be presented in detail.

Table 4 presents an overview of the reliability measures (Cronbach's alpha and composite reliability) and validity measures (average variance extracted) to each item. Cronbach's alpha was used to measure the internal consistency reliability of the constructs. According to the literature, the values of cronbach's alpha must be bigger than 0.70, that criteria was accomplished for all the variables. Additionally, the composite reliability coefficients, rho_a and rho_c, also fulfill the recommended threshold of 0.70, therefore, the indicator reliability is satisfied (Hair Jr. et al., 2021). The average variance extracted (AVE) was analyzed to measure the convergent validity of the constructs, once the AVE value is greater than 0.50 for all measures, which is in accordance with the literature, it indicates convergent validity of the measurement model.

Table 3 – Construct Reliability and Validity

	Cronbach's alpha	Composite reliability (rho_a)	Composite reliability (rho_c)	Average variance extracted (AVE)
Code Quality	0.810	0.813	0.887	0.724
Effort Expectancy	0.824	0.853	0.876	0.587
Facilitating Conditions	0.877	0.889	0.915	0.729
Perceived Learning	0.858	0.861	0.913	0.779
Perceived Productivity	0.930	0.934	0.950	0.825
Social Influence	0.758	0.830	0.836	0.569

To access and evaluate the discriminant validity (if a construct is different from the other constructs in the model), three measures were used: Fornell-Larcker criterion, indicator's loadings and heterotrait-monotrait (HTMT) ratios. The Fornell-Larcker principle establishes discriminant validity if the square root of the AVE of each item (the diagonal value) exceeds its maximum association with any other construct (Fornell & Larcker, 1981). Table 4 confirms this criterion. An analysis of the cross-loading table ensures that each item demonstrates stronger loadings on the related construct, compared with all of its cross loadings (Hair et al., 2011). As shown in Appendix B, all of the indicators primarily measure its intended construct, which demonstrates a strong discriminant validity. Lastly, the HTMT ratio was conducted, and as seen in Table 5 all values are beneath 0.90, which means that the constructs differ from one another. To summarize, as the three criteria are satisfied, it is possible to confirm the discriminant validity of the constructs.

Table 4 - Fornell-Larcker Results

	AU	A	CQ	EE	FC	PL	PP	SI	US
Actual Use (AU)	**1**								
Age (A)	-0.103	**1**							
Code Quality (QC)	0.302	-0.171	**0.851**						
Effort Expectancy (EE)	0.496	-0.044	0.230	**0.766**					
Facilitating Conditions (FC)	0.675	-0.100	0.298	0.656	**0.854**				
Perceived Learning (PL)	0.377	-0.178	0.476	0.292	0.284	**0.883**			
Perceived Productivity (PP)	0.592	0.053	0.273	0.641	0.691	0.215	**0.909**		
Social Influence (SI)	0.490	-0.073	0.303	0.455	0.548	0.381	0.426	**0.754**	
User Satisfaction (US)	0.633	0.032	0.425	0.506	0.572	0.475	0.596	0.418	**1**

Table 5 - HTMT Ratios

	AU	A	CQ	EE	FC	PL	PP	SI	US
Actual Use (AU)									
Age (A)	0.103								
Code Quality (QC)	0.334	0.184							
Effort Expectancy (EE)	0.537	0.146	0.277						
Facilitating Conditions (FC)	0.710	0.107	0.350	0.759					
Perceived Learning (PL)	0.407	0.190	0.560	0.339	0.317				
Perceived Productivity (PP)	0.614	0.062	0.311	0.726	0.752	0.238			
Social Influence (SI)	0.531	0.129	0.422	0.515	0.629	0.487	0.439		
User Satisfaction (US)	0.633	0.032	0.472	0.541	0.602	0.512	0.615	0.420	

Finally, the variance inflation factor (VIF) measures the collinearity of predictor constructs. To avoid collinearity issues the threshold should be bellow 5 for all variables. As observed in Table 6, as all values are bellow 3. In so, the predictor variables are not correlated, which suggests that collinearity is not a problematic issue.

Table 6 - Variance Inflation Factor (VIF)

	VIF
Age -> Code Quality	1.000
Age -> Perceived Learning	1.000
Code Quality -> User Satisfaction	1.364
Effort Expectancy -> User Satisfaction	2.062
Facilitating Conditions -> Actual Use	1.488
Facilitating Conditions -> User Satisfaction	2.521
Perceived Learning -> Actual Use	1.292
Perceived Learning -> User Satisfaction	1.431
Perceived Productivity -> User Satisfaction	2.197
Social Influence -> User Satisfaction	1.581
User Satisfaction -> Actual Use	1.766

To conclude, the analysis evidence positive outcomes through all measures, which proves that the model is valid, reliable as well as robust.

4.4. STRUCTURAL MODEL

After validating the conceptual model, it is necessary to analyze and validate the structural model. This second phase of the PLS-SEM method is calculated using bootstrapping technique to compute the p-values with a subsample of 5000. Bootstrapping is a resampling technique that generates numerous subsamples with replacement from the original sample (Hair et al., 2011). Both table 7 and Figure 3 provide a summary of the outcomes of the model estimation, namely the path coefficients and the R^2 values.

Table 7 - Research Hypothesis Results

Hypothesis		Original sample (O)	Sample mean (M)	Standard deviation (STDEV)	T statistics (|O/STDEV|)	P values	Significance
H1	PP -> US	0.335	0.335	0.084	3.983	0.000	***
H2	EE -> US	0.061	0.067	0.084	0.720	0.471	NS
H3	SI -> US	0.005	0.008	0.074	0.069	0.945	NS
H4	FC -> US	0.182	0.180	0.114	1.603	0.109	NS
H5	FC -> AU	0.464	0.465	0.064	7.304	0.000	***
H6	US -> AU	0.324	0.323	0.078	4.150	0.000	***
H7	CQ -> US	0.136	0.138	0.071	1.921	0.055	NS
H8	PL -> US	0.267	0.261	0.058	4.604	0.000	***
H9	PL -> AU	0.092	0.091	0.061	1.491	0.136	NS
H10	A -> CQ	-0.171	-0.174	0.086	2.004	0.045	*
H11	A -> PL	-0.178	-0.176	0.086	2.069	0.039	*

Note: NS = not significant; * significant at p < 0.05; ** significant at p < 0.01; * significant at p < 0.001 (Cohen, 1992)**

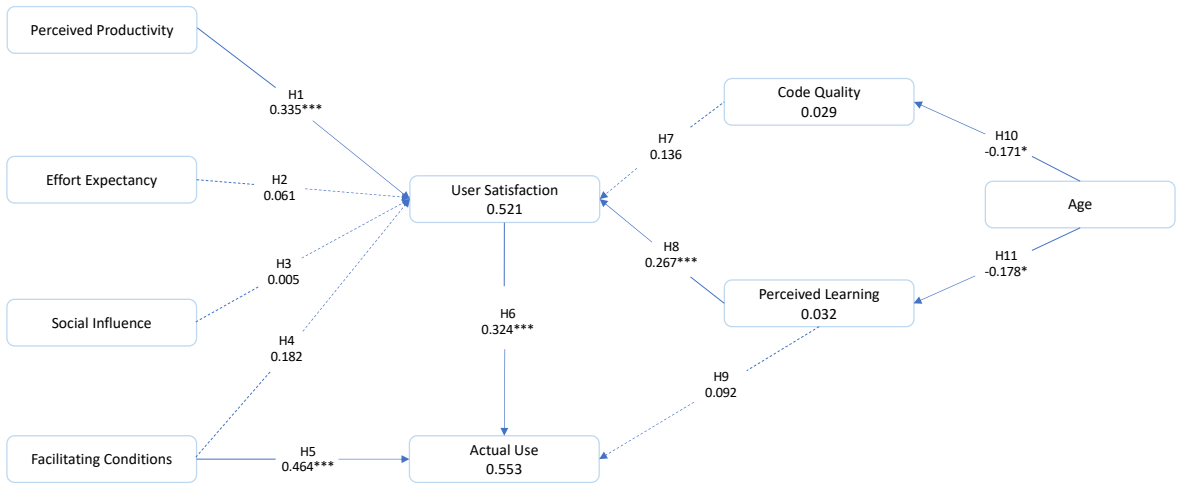

Figure 3 - Structural Model Results

The structural model presented in Figure 3 provides the tested hypothesis results. The model explains 52% of the variation in user satisfaction. Perceived productivity (β = 0.335, p < 0.001) and perceived learning (β = 0.267, p < 0.001), are both statistically significant to explain user satisfaction with Copilot, providing support to H1 and H8. Notably, perceived productivity has a stronger impact, which insinuates that user satisfaction is better explained by this construct than by perceived learning. Moreover, the model explains 55% of the variance in actual use, being it impacted by facilitating conditions (β = 0.464, p < 0.001) and user satisfaction (β =

0.324, $p < 0.001$). Thus, the hypothesis H5 and H6 are confirmed. The hypothesis H10 and H11, despite having a negative coefficient, does not mean that they do not have a significant impact. In fact, it only suggests that an increase in age is related with a decline in the code quality/ perceived learning variable. In so, code quality and perceived learning are negatively impacted by age ($\beta = -0.171$, $p < 0.05$) and ($\beta = -0.178$, $p < 0.05$), respectively.

It is equally important to recognize the non-significant findings that add to the overall understanding of the model. As observed in the model, H2, H3, H4, H7 and H9 did not have a significant impact on their independent variables. Therefore, effort expectancy ($\beta = 0.061$, $p > 0.05$), social influence ($\beta = 0.005$, $p > 0.05$), facilitating conditions ($\beta = 0.182$, $p > 0.05$) and code quality ($\beta = 0.136$, $p > 0.05$) do not significantly impact user satisfaction. It is important to mention that despite code quality did not have a significant impact, it was very closed to have (p-value = 0.055). Also, perceived learning ($\beta = 0.092$, $p > 0.05$) does not impact the actual use of Copilots.

5. DISCUSSION

The proposed research model uses UTAUT framework as a reference model to study the classical variables that moderate the impact of user satisfaction on the acceptance and use of Copilots for programming. Other particular variables, which have rarely been used in prior literature, were added to the model in order to analyze how they could affect and influence it. From the eleven hypotheses tested, six showed significant results, while the five non-significant findings offered insights into the limits of certain variables.

The results of this study reveal that perceived productivity has a significant impact on user satisfaction, which specifies that one of the reasons for programmers being satisfied with the tool is because they feel more productive when using it (Lee & Kim, 2022; Maillet et al., 2015; Tam et al., 2020).

On the other hand, the remain standard variables of UTAUT, effort expectancy, social influence and facilitating conditions, in contradiction with most of the results observed across different studies, are not statistically significant to the model. This means developers do not consider these three factors important for their level of satisfaction with Gen-AI tools, more specifically, Copilot. As effort expectancy is related to the ease of using Copilot, it may have a less significance for programming tools, once that programmers typically have advanced technical skills as well as a higher tolerance and willing to learn new tools. Despite contradicting the findings of Venkatesh et al. (2012), there are other papers that align with the findings of this study, namely Kosiba's (2022). Aligned with Magsamen-Conrad et al. (2015) and Russo (2024), social influence also presented to not be significant. A possible reason for this might be because currently, technology is adopted massively and rapidly by everyone at the same time, other reason can rely on the individual-centric that programming is. Facilitating Conditions also do not impact user satisfaction, which is aligned with Lee & Kim (2022) previous study. However, in conformance with several studies such the Venkatesh et al (2003) study, facilitating conditions exhibit significance on actual use, which informs that FC only impact the use of Copilot, not the general satisfaction with the tool.

Furthermore, the notable impact of the variable age on code quality and perceived learning implies that younger users perceive to learn more code by using Copilot, and that younger users, in contrary to older ones, understand that the quality of the code is good. Code Quality has a non-significant impact on user satisfaction, while perceived learning has a positive influence. Nevertheless, it is important to refer that that perceived productivity proves a higher impact on user satisfaction compared to perceived learning, settling that the main reason users value Copilot is due to the fact it enhances their productivity.

Finally, user satisfaction impacts positively on actual use, once satisfied users are more probable to continue using Copilot. Several authors studied how satisfaction impacts continuous using of a tool (Tam et al., 2020).

6. CONCLUSIONS AND FUTURE WORK

Based on the UTAUT framework, this thesis aimed to investigate the factors motivating the satisfaction of developers with Copilot, as well as explore which factors influence the adoption of the tool and understand the relationship between user satisfaction and use of Copilots. To achieve these objectives, a primary analysis was made to identify the determinants influencing user satisfaction and actual use. Consequently, it was developed a questionnaire and a conceptual model that was tested using PLS-SEM.

The results of the study reveal significant theoretical implications. Perceived productivity posits a strong positive impact on user satisfaction, which promotes the potential of the tool in completing more tasks in a period of time or helping to perform tasks more quickly. The study's findings suggest that in the context of Gen-AI tools for programming, and more specifically Copilot, perceived learning expresses a significant impact, influencing user satisfaction rather than actual system use. This notable impact of perceived learning on user satisfaction suggests possible outcomes in the education field. It also suggests that users may not adopt the tool because learning matters but appreciate the fact they could learn post-adoption. The more developers think they learn from the tool, the more satisfied they are. Also, the impact of user satisfaction on use and adoption of Copilot suggests that it is important to adjust the UTAUT model to better access the predictors of technology acceptance. The greater the satisfaction of users, the higher their likelihood of using Copilot.

Regarding the practical implications, as the results specify that perceived productivity has a significant impact on satisfaction, providers should put effort on advertising and sharing the usefulness and potential of the Gen-AI tool, while organizations and individuals should incorporate the tool in their work basis.

This study, like any other, faced several limitations, however, these limitations provide opportunities for further research. The sample size was narrow and may not be representative of the IT population, consisting in 144 valid answers, most of them given by male participants, which may limit the generalizability of the findings. Furthermore, the conducted study only focused on the context of software professionals' use of Copilot. Future research should consider a larger sample size and study the validity of the results in broader or distinctive contexts. Additionally, the moderating variables presented in UTAUT framework were not contemplated, neither was the connection between behavioral intention and actual use.

The study only reflects a limited temporal scope, which can narrow the results of the study. This can occur because Gen-AI is always evolving, and perceptions of users can be different. Future studies are needed to verify the generalization of the model. Lastly, the data collection of this research was made through a questionnaire, future research could explore alternative approaches, such as qualitative methods.

REFERENCES

Almufarreh, A. (2024). Determinants of Students' Satisfaction with AI Tools in Education: A PLS-SEM-ANN Approach. Sustainability, 16(13), Artigo 13. https://doi.org/10.3390/su16135354

Asare, O., Nagappan, M., & Asokan, N. (2023). Is GitHub's Copilot as bad as humans at introducing vulnerabilities in code? Empirical Software Engineering, 28(6), 129. https://doi.org/10.1007/s10664-023-10380-1

Barke, S., James, M. B., & Polikarpova, N. (2023). Grounded Copilot: How Programmers Interact with Code-Generating Models. Proceedings of the ACM on Programming Languages, 7(OOPSLA1), 78:85-78:111. https://doi.org/10.1145/3586030

Bird, C., Ford, D., Zimmermann, T., Forsgren, N., Kalliamvakou, E., Lowdermilk, T., & Gazit, I. (2023). Taking Flight with Copilot. Communications of the ACM, 66(6), 56–62. https://doi.org/10.1145/3589996

Brown, S. A., Dennis, A. R., & Venkatesh, V. (2010). Predicting Collaboration Technology Use: Integrating Technology Adoption and Collaboration Research. Journal of Management Information Systems, 27(2), 9–54. https://doi.org/10.2753/MIS0742-1222270201

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners (arXiv:2005.14165). arXiv. https://doi.org/10.48550/arXiv.2005.14165

Chan, F. K. Y., Thong, J. Y. L., Venkatesh, V., Brown, S. A., Hu, P. J. H., & Tam, K. Y. (2011). Modeling Citizen Satisfaction with Mandatory Adoption of an E-Government

Technology (SSRN Scholarly Paper 1976951). https://papers.ssrn.com/abstract=1976951

Chattaraman, V., Kwon, W.-S., Gilbert, J. E., & Ross, K. (2019). Should AI-Based, conversational digital assistants employ social- or task-oriented interaction style? A task-competency and reciprocity perspective for older adults. Computers in Human Behavior, 90, 315–330. https://doi.org/10.1016/j.chb.2018.08.048

Chatterjee, S., & Bhattacharjee, K. K. (2020). Adoption of artificial intelligence in higher education: A quantitative analysis using structural equation modelling. Education and Information Technologies, 25(5), 3443–3463. https://doi.org/10.1007/s10639-020-10159-7

Chauhan, S., & Jaiswal, M. (2016). Determinants of acceptance of ERP software training in business schools: Empirical investigation using UTAUT model. The International Journal of Management Education, 14(3), 248–262. https://doi.org/10.1016/j.ijme.2016.05.005

Chen, L., Chen, P., & Lin, Z. (2020). Artificial Intelligence in Education: A Review. IEEE Access, 8, 75264–75278. IEEE Access. https://doi.org/10.1109/ACCESS.2020.2988510

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. de O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., Ray, A., Puri, R., Krueger, G., Petrov, M., Khlaaf, H., Sastry, G., Mishkin, P., Chan, B., Gray, S., … Zaremba, W. (2021). Evaluating Large Language Models Trained on Code (arXiv:2107.03374). arXiv. https://doi.org/10.48550/arXiv.2107.03374

Chen, S., Chen, H., & Chen, M. (2009). Determinants of satisfaction and continuance intention towards self-service technologies. Industrial Management & Data Systems, 109(9), 1248–1263. https://doi.org/10.1108/02635570911002306

Chong, A. Y.-L. (2013). Understanding Mobile Commerce Continuance Intentions: An Empirical Analysis of Chinese Consumers. Journal of Computer Information Systems, 53(4), 22–30. https://doi.org/10.1080/08874417.2013.11645647

Cohen, J. (1992). Statistical Power Analysis. Current Directions in Psychological Science, 1(3), 98–101.

Delone, W. H., & McLean, E. R. (2003). The DeLone and McLean Model of Information Systems Success: A Ten-Year Update. Journal of Management Information Systems, 19(4), 9–30. https://doi.org/10.1080/07421222.2003.11045748

Dohmke, T. (2023, março 22). GitHub Copilot X: The AI-powered developer experience. The GitHub Blog. https://github.blog/2023-03-22-github-copilot-x-the-ai-powered-developer-experience/

Fornell, C., & Larcker, D. F. (1981). Evaluating Structural Equation Models with Unobservable Variables and Measurement Error. Journal of Marketing Research, 18(1), 39–50. https://doi.org/10.2307/3151312

Friedman, N. (2021, junho 29). Introducing GitHub Copilot: Your AI pair programmer. The GitHub Blog. https://github.blog/2021-06-29-introducing-github-copilot-ai-pair-programmer/

Hair, J. F., Ringle, C. M., & Sarstedt, M. (2011). PLS-SEM: Indeed a Silver Bullet. Journal of Marketing Theory and Practice, 19(2), 139–152. https://doi.org/10.2753/MTP1069-6679190202

Hair Jr., J. F., Hult, G. T. M., Ringle, C. M., Sarstedt, M., Danks, N. P., & Ray, S. (2021). Partial Least Squares Structural Equation Modeling (PLS-SEM) Using R: A Workbook. Springer Nature. https://doi.org/10.1007/978-3-030-80519-7

Hong, W., Thong, J. Y. L., Chasalow, L. C., & Dhillon, G. (2011). User Acceptance of Agile Information Systems: A Model and Empirical Test. Journal of Management Information Systems, 28(1), 235–272. https://doi.org/10.2753/MIS0742-1222280108

Hsiao, C.-H., Chang, J.-J., & Tang, K.-Y. (2016). Exploring the influential factors in continuance usage of mobile social Apps: Satisfaction, habit, and customer value perspectives. Telematics and Informatics, 33(2), 342–355. https://doi.org/10.1016/j.tele.2015.08.014

Imai, S. (2022). Is GitHub Copilot a Substitute for Human Pair-programming? An Empirical Study. 2022 IEEE/ACM 44th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 319–321. https://doi.org/10.1109/ICSE-Companion55297.2022.9793778

Jaiswal, A., Arun, C. J., & Varma, A. (2021). Rebooting employees: Upskilling for artificial intelligence in multinational corporations. The International Journal of Human Resource Management, 33(6), 1179–1208. https://doi.org/10.1080/09585192.2021.1891114

Jaiswal, D., Kaushal, V., Mohan, A., & Thaichon, P. (2022). Mobile wallets adoption: Pre- and post-adoption dynamics of mobile wallets usage. Marketing Intelligence & Planning, 40(5), 573–588. https://doi.org/10.1108/MIP-12-2021-0466

Joo, Y. J., Park, S., & Shin, E. K. (2017). Students' expectation, satisfaction, and continuance intention to use digital textbooks. Computers in Human Behavior, 69, 83–90. https://doi.org/10.1016/j.chb.2016.12.025

Kalliamvakou, E. (2022, setembro 7). Research: Quantifying GitHub Copilot's impact on developer productivity and happiness. The GitHub Blog. https://github.blog/2022-09-

07-research-quantifying-github-copilots-impact-on-developer-productivity-and-happiness/

Kosiba, J. P. B., Odoom, R., Boateng, H., Twum, K. K., & Abdul-Hamid, I. K. (2022). Examining students' satisfaction with online learning during the Covid-19 pandemic—An extended UTAUT2 approach. Journal of Further and Higher Education, 46(7), 988–1005. https://doi.org/10.1080/0309877X.2022.2030687

Laumer, S., Maier, C., & Weitzel, T. (2017). Information quality, user satisfaction, and the manifestation of workarounds: A qualitative and quantitative study of enterprise content management system users. European Journal of Information Systems, 26(4), 333–360. https://doi.org/10.1057/s41303-016-0029-7

Lee, U.-K., & Kim, H. (2022). UTAUT in Metaverse: An "Ifland" Case. Journal of Theoretical and Applied Electronic Commerce Research, 17(2), Artigo 2. https://doi.org/10.3390/jtaer17020032

Li, C.-Y., & Fang, Y.-H. (2019). Predicting continuance intention toward mobile branded apps through satisfaction and attachment. Telematics and Informatics, 43, 101248. https://doi.org/10.1016/j.tele.2019.101248

Liang, J. T., Yang, C., & Myers, B. A. (2024). A Large-Scale Survey on the Usability of AI Programming Assistants: Successes and Challenges. Proceedings of the 46th IEEE/ACM International Conference on Software Engineering, 1–13. https://doi.org/10.1145/3597503.3608128

Magsamen-Conrad, K., Upadhyaya, S., Joa, C. Y., & Dowd, J. (2015). Bridging the divide: Using UTAUT to predict multigenerational tablet adoption practices. Computers in Human Behavior, 50, 186–196. https://doi.org/10.1016/j.chb.2015.03.032

Maillet, É., Mathieu, L., & Sicotte, C. (2015). Modeling factors explaining the acceptance, actual use and satisfaction of nurses using an Electronic Patient Record in acute care settings: An extension of the UTAUT. International Journal of Medical Informatics, 84(1), 36–47. https://doi.org/10.1016/j.ijmedinf.2014.09.004

Martins, C., Oliveira, T., & Popovič, A. (2014). Understanding the Internet banking adoption: A unified theory of acceptance and use of technology and perceived risk application. International Journal of Information Management, 34(1), 1–13. https://doi.org/10.1016/j.ijinfomgt.2013.06.002

Maslej, N., Fattorini, L., Brynjolfsson, E., Etchemendy, J., Ligett, K., Lyons, T., Manyika, J., Ngo, H., Niebles, J. C., Parli, V., Shoham, Y., Wald, R., Clark, J., & Perrault, R. (2023). The AI Index 2023 Annual Report (arXiv:2310.03715). arXiv. https://doi.org/10.48550/arXiv.2310.03715

Moradi Dakhel, A., Majdinasab, V., Nikanjam, A., Khomh, F., Desmarais, M. C., & Jiang, Z. M. (Jack). (2023). GitHub Copilot AI pair programmer: Asset or Liability? Journal of Systems and Software, 203, 111734. https://doi.org/10.1016/j.jss.2023.111734

Murphy-Hill, E., Jaspan, C., Sadowski, C., Shepherd, D., Phillips, M., Winter, C., Knight, A., Smith, E., & Jorde, M. (2021). What Predicts Software Developers' Productivity? IEEE Transactions on Software Engineering, 47(3), 582–594. IEEE Transactions on Software Engineering. https://doi.org/10.1109/TSE.2019.2900308

Niu, B., & Mvondo, G. F. N. (2024). I Am ChatGPT, the ultimate AI Chatbot! Investigating the determinants of users' loyalty and ethical usage concerns of ChatGPT. Journal of Retailing and Consumer Services, 76, 103562. https://doi.org/10.1016/j.jretconser.2023.103562

Open Source Security and Risk Analysis 2023. (2023).

OpenAI. (2023, março 15). GPT-4 Technical Report. arXiv.Org. https://arxiv.org/abs/2303.08774v4

Palau-Saumell, R., Forgas-Coll, S., Sánchez-García, J., & Robres, E. (2019). User Acceptance of Mobile Apps for Restaurants: An Expanded and Extended UTAUT-2. Sustainability, 11(4), Artigo 4. https://doi.org/10.3390/su11041210

Pearce, H., Ahmad, B., Tan, B., Dolan-Gavitt, B., & Karri, R. (2022). Asleep at the Keyboard? Assessing the Security of GitHub Copilot's Code Contributions. 2022 IEEE Symposium on Security and Privacy (SP), 754–768. https://doi.org/10.1109/SP46214.2022.9833571

Rosenkilde, J. (2023, maio 17). How GitHub Copilot is getting better at understanding your code. The GitHub Blog. https://github.blog/2023-05-17-how-github-copilot-is-getting-better-at-understanding-your-code/

Russo, D. (2024). Navigating the Complexity of Generative AI Adoption in Software Engineering. ACM Transactions on Software Engineering and Methodology, 33(5), 135:1-135:50. https://doi.org/10.1145/3652154

Sobania, D., Briesch, M., & Rothlauf, F. (2022). Choose your programming copilot: A comparison of the program synthesis performance of github copilot and genetic programming. Proceedings of the Genetic and Evolutionary Computation Conference, 1019–1027. https://doi.org/10.1145/3512290.3528700

Taherdoost, H. (2019). Importance of Technology Acceptance Assessment for Successful Implementation and Development of New Technologies. Global Journal of Engineering Sciences, 1(3). https://doi.org/10.33552/GJES.2019.01.000511

Tam, C., Santos, D., & Oliveira, T. (2020). Exploring the influential factors of continuance intention to use mobile Apps: Extending the expectation confirmation model.

Information Systems Frontiers, 22(1), 243–257. https://doi.org/10.1007/s10796-018-9864-5

Totlani, K. (2023). The Evolution of Generative AI: Implications for the Media and Film Industry. International Journal for Research in Applied Science and Engineering Technology, 11(10), 973–980. https://doi.org/10.22214/ijraset.2023.56140

Urbach, N., & Müller, B. (2012). The Updated DeLone and McLean Model of Information Systems Success (Vol. 1, pp. 1–18). https://doi.org/10.1007/978-1-4419-6108-2_1

Vaithilingam, P., Zhang, T., & Glassman, E. L. (2022). Expectation vs. Experience: Evaluating the Usability of Code Generation Tools Powered by Large Language Models. Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems, 1–7. https://doi.org/10.1145/3491101.3519665

Venkatesh, V., & Bala, H. (2008). Technology Acceptance Model 3 and a Research Agenda on Interventions. Decision Sciences, 39(2), 273–315. https://doi.org/10.1111/j.1540-5915.2008.00192.x

Venkatesh, V., & Davis, F. D. (2000). A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies. Management Science, 46(2), 186–204.

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). User Acceptance of Information Technology: Toward a Unified View. MIS Quarterly, 27(3), 425–478. https://doi.org/10.2307/30036540

Venkatesh, V., Thong, J. Y. L., & Xu, X. (2012). Consumer Acceptance and Use of Information Technology: Extending the Unified Theory of Acceptance and Use of Technology. MIS Quarterly, 36(1), 157–178. https://doi.org/10.2307/41410412

Venkatesh, V., Thong, J. Y. L., & Xu, X. (2016). Unified Theory of Acceptance and Use of Technology: A Synthesis and the Road Ahead (SSRN Scholarly Paper 2800121). https://papers.ssrn.com/abstract=2800121

Wang, T., Lund, B. D., Marengo, A., Pagano, A., Mannuru, N. R., Teel, Z. A., & Pange, J. (2023). Exploring the Potential Impact of Artificial Intelligence (AI) on International Students in Higher Education: Generative AI, Chatbots, Analytics, and International Student Success. Applied Sciences, 13(11), Artigo 11. https://doi.org/10.3390/app13116716

Wermelinger, M. (2023). Using GitHub Copilot to Solve Simple Programming Problems. Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1, 172–178. https://doi.org/10.1145/3545945.3569830

Yilmaz, R., & Yilmaz, F. G. K. (2023). Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning. Computers in Human Behavior: Artificial Humans, 1(2), 100005. https://doi.org/10.1016/j.chbah.2023.100005

APPENDIX A. ETHICS COMMITTEE APPROVAL

This is to certify that

Project No.: **INFSYS2024-4-16115**

Project Title: **Copilot**

Principal Researcher: **Margarida Costa**

according to the regulations of the Ethics Committee of NOVA IMS and MagIC Research Center this project was considered to meet the requirements of the NOVA IMS Internal Review Board, being considered **APPROVED** on 4/1/2024.

It is the Principal Researcher's responsibility to ensure that all researchers and stakeholders associated with this project are aware of the conditions of approval and which documents have been approved.

The Principal Researcher is required to notify the Ethics Committee, via amendment or progress report, of

- Any significant change to the project and the reason for that change;
- Any unforeseen events or unexpected developments that merit notification;
- The inability of the Principal Researcher to continue in that role or any other change in research personnel involved in the project.

Lisbon, 4/1/2024

NOVA IMS Ethics Committee
ethicscommittee@novaims.unl.pt

APPENDIX B. CROSS LOADINGS TABLE

	Actual Use	Age	Code Quality	Effort Expectancy	Facilitating Conditions	Perceived Learning	Perceived Productivity	Social Influence	User Satisfaction
AU1	**1.000**	-0.103	0.302	0.496	0.675	0.377	0.592	0.490	0.633
Age	-0.103	**1.000**	-0.171	-0.044	-0.100	-0.178	0.053	-0.073	0.032
CQ1	0.323	-0.138	**0.832**	0.215	0.258	0.494	0.228	0.214	0.382
CQ2	0.230	-0.022	**0.843**	0.247	0.260	0.248	0.274	0.244	0.374
CQ3	0.215	-0.263	**0.878**	0.132	0.244	0.454	0.201	0.314	0.330
EE1	0.362	-0.129	0.182	**0.730**	0.514	0.182	0.531	0.299	0.336
EE2	0.276	-0.104	0.143	**0.672**	0.351	0.230	0.350	0.330	0.286
EE3	0.441	-0.038	0.242	**0.804**	0.496	0.285	0.520	0.362	0.502
EE4	0.419	-0.067	0.174	**0.887**	0.616	0.253	0.530	0.395	0.426
EE5	0.370	0.170	0.114	**0.720**	0.519	0.145	0.511	0.357	0.332
FC1	0.533	-0.145	0.294	0.476	**0.840**	0.285	0.553	0.536	0.445
FC2	0.535	-0.009	0.217	0.595	**0.860**	0.181	0.552	0.438	0.478
FC3	0.511	-0.103	0.205	0.522	**0.856**	0.163	0.531	0.400	0.429
FC4	0.693	-0.086	0.291	0.627	**0.859**	0.317	0.691	0.490	0.575
PL1	0.364	-0.139	0.382	0.276	0.322	**0.863**	0.187	0.357	0.412
PL2	0.314	-0.126	0.391	0.232	0.188	**0.872**	0.182	0.353	0.394
PL3	0.320	-0.202	0.482	0.262	0.237	**0.912**	0.199	0.302	0.449
PP1	0.554	0.041	0.192	0.535	0.588	0.142	**0.889**	0.307	0.496
PP2	0.533	0.069	0.303	0.585	0.631	0.219	**0.924**	0.402	0.588
PP3	0.524	0.098	0.158	0.547	0.576	0.171	**0.925**	0.373	0.508
PP4	0.542	-0.011	0.322	0.653	0.705	0.240	**0.895**	0.455	0.564
SI1	0.400	0.013	0.257	0.450	0.469	0.305	0.436	**0.856**	0.416
SI2	0.155	-0.144	0.308	0.096	0.196	0.256	0.059	**0.519**	0.080
SI3	0.373	-0.115	0.237	0.350	0.444	0.294	0.325	**0.862**	0.350
SI4	0.480	-0.071	0.210	0.334	0.463	0.338	0.302	**0.728**	0.270
US1	0.633	0.032	0.425	0.506	0.572	0.475	0.596	0.418	**1.000**

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa