LUND UNIVERSITY

SCHOOL OF ECONOMICS AND MANAGEMENT

Institutionen för informatik

SYSG16: Examensarbete för kandidatexamen
VT 2023

How do students perceive usability and usefulness of the Eclipse Integrated Development Environment? A survey study on students' perspectives at Lund University.

A quantitative study of students' perceptions of usability and usefulness.

Supervisor: Nam Aghaee
Writers: Måns Jönsson & Max Andersson

Publisher: Institutionen av informatik, Ekonomihögskolan, Lunds Universitet

Examinator: Blerim Emruli

Published: May 2023

Type of Document: Bachelor Thesis

Number of Pages: 87

Keywords: Perception, Usability, Usefulness, The Eclipse IDE, External Factors, TAM, Modes of Interaction

Abstract:
This study examines the factors regarding the perceived usability and usefulness of the Eclipse IDE and how this influences students' perceptions of the Eclipse IDE in the context of education. The theoretical framework of the study is the Technology Acceptance Model (TAM). The theoretical framework Modes of Interaction is introduced to further explain external factors. The external factors including training, instructor and peer support, and the accessibility of online resources, were found to significantly influence students' perceptions of the software. A survey was conducted to gather students' opinions on the perceived usability, usefulness, and external factors related to the Eclipse IDE. Integrated Development Environments are often complex applications, not friendly for novice programmers, with a learning process of several weeks and with usability and satisfaction of use not always as good as expected. The findings indicate that perceived usability, such as learnability and navigation, varies among students and may impact their technology acceptance. Moreover, the survey reveals mixed results of the Eclipse IDE's usefulness, with some students believing it enhances their ability to develop applications, while others do not. The result of this study, based on earlier studies, addresses that external factors such as peer and instructional support and documentation can lead to improved technology adoption rates and better learning outcomes. By considering external factors, educators can better support students in their programming education and effectively evaluate the suitability of the Eclipse IDE for their courses.

Figures

Tables

Definitions of terms and acronyms

Term	**Definition**
Usability	Usability refers to the ease of use and the effectiveness of a product or system. It considers how efficiently, effectively, and satisfactorily users can interact with a product to achieve their intended objectives.
Usefulness	Usefulness refers to the degree to which a product or system provides value to its users by enabling them to conduct tasks more effectively, efficiently, or with greater satisfaction. It assesses the practical benefits or advantages that users perceive from using a particular product.
External Factors	External Factors refers to the degree to which factors not directly associated with the product or system provides value to its users by enabling them to conduct tasks more effectively, efficiently, or with greater satisfaction.
IDE	Integrated Development Environment (IDE) is a software application that combines various tools and features, such as code editors, debuggers, and compilers, to facilitate the development of software programs. It enhances the software development process by providing a comprehensive and integrated environment for coding, testing, and debugging.
Perception	Perception is the process through which individuals interpret, organize, and make sense of information gathered from their environment. In the scope of this research, it refers to students' understanding and interpretation of the usability and usefulness of the Eclipse IDE based on their experiences and interactions with the software.

1. Introduction

The purpose of the following introduction is to give a brief overview of the topic of the thesis. It starts by providing some background information, followed by a discussion of the problem area and the research question that is based on the previously presented context. Finally, the section concludes by defining the scope and objectives of the study.

1.1 Background

The Eclipse Integrated Development Environment (IDE) stands as a seminal tool in the landscape of software development, with its significance being markedly felt in the realm of academia (Eclipse, 2022). Software development tools, such as IDE's, play an important role in the multifaceted landscape of software development. IDE's are specialized software tools designed to facilitate the entire software development process, providing programmers with an all-in-one toolkit that integrates a wide array of functionalities (Parnin, Görg & Rugaber, 2010).

These functionalities typically encompass a source code editor, build automation tools, and a debugger, often integrated through a single graphical user interface. This integration allows developers to write, compile, test, and debug code in a single environment, eliminating the need to switch between different tools (Parnin, Görg & Rugaber, 2010; Murphy, Kersten & Findlater, 2006). IDE's can significantly boost productivity, streamline the software development process, and help team collaboration.

IDEs, while integral to the intricate processes involved in software development, are notorious for their complexity. This poses an intimidating obstacle for beginners in the field, suggesting a steep learning curve that may span several weeks, often leading to user experiences that fall short of expectations (Morales, Botella, Rusu & Quiñones, 2019). In the era of digital technology, where the role of Technology Enhanced Learning (TEL) is increasingly important (Kirkwood & Price, 2014), comprehending how students perceive the usability and usefulness of such tools becomes a matter of paramount importance.

In the broader context of higher education, digital technology and tools like IDE's are reshaping the teaching and learning landscape (Bower, 2019). As institutions continue to embrace digital transformation, the importance of tools like the Eclipse IDE in creating an engaging and productive learning environment cannot be understated (Bower, 2019; Selwyn, 2016). They do have the potential to revolutionize not only how students learn programming but also how educators design and deliver their education. However, the successful integration of these tools into the educational setting hinges on their perceived usability and usefulness.

While there is a wealth of research delving into the usability and usefulness of various IDE's,

studies centering specifically on student perceptions of the Eclipse IDE remain relatively sparse. Existing research has identified certain usability challenges of the Eclipse IDE, such as its complicated interface and scarce documentation (Bower, 2019). Furthermore, Aljafer & Cantrell (2020) mentions that the suite of advanced features offered by the Eclipse IDE, though potent, may overwhelm students who are simultaneously struggling with the complexities of learning programming.

This overarching narrative underscores the necessity for a comprehensive, in-depth exploration of how students perceive the Eclipse IDE. This understanding will not only enrich the body of knowledge in the field of Technology Enhanced Learning (Kirkwood & Price, 2014) but also provide valuable insights that could shape the future of digital education and software development training (Bower, 2019; Selwyn, 2016).

1.2 Problem Area

A shortage of research focusing on students' perceptions of the Eclipse IDE's suitability for classroom use presents a significant challenge to understanding its role in an introductory programming context (Bower, 2019). The Eclipse IDE is the primary tool for students enrolled in the Information System bachelor program. This study aims to delve into students' experiences with the Eclipse IDE, with an emphasis on its perceived usability and usefulness.

The aim of this research is to pinpoint and explore the challenges students face when using the Eclipse IDE and to assess its suitability for academic use. By investigating students' perceived usability and usefulness of the Eclipse IDE, we hope to shed light on how to better serve educational activities within the programming field.

1.3 Aim & Research Question

While existing research on IDE's offers useful insights on improving user experience, there is a noticeable gap regarding student perceptions of its usability and usefulness (Bower, 2019). This thesis intends to bridge this gap by capturing the student perspective on using the Eclipse IDE in programming courses. This objective led to the formulation of the following research question: *What are students' perceptions of the usability and usefulness of the Eclipse Integrated Development Environment (IDE) in programming courses?*

1.4 Purpose of Study

The purpose of this study is to identify and evaluate the factors, affecting students' overall experience of using the Eclipse IDE, in programming courses. This thesis aims to gain a better picture of how students currently enrolled in the Information Systems bachelor program at Lund University perceive the Eclipse IDE, in terms of usability and usefulness. By providing insights

into the Eclipse IDE's usability and usefulness in supporting students' educational activities, this study contributes to the enhancement of learning outcomes in software development for students. By understanding the factors that influence students' perceptions of the Eclipse IDE's usability and usefulness, educators can tailor the software to better align with students' needs, ultimately fostering a more conducive learning environment for achieving success in software development education.

1.5 Limitations

Our research is limited to the Eclipse IDE and how students studying the Information System program at Lund University that come in contact with the Eclipse IDE perceive its usability and usefulness. This research will not compare this with any other IDE's or target audiences. The paper does not aim to provide technical or analytical solutions to the challenges that may arise from the study. Interviews were not conducted for one major reason, since the population consists of students currently enrolled in three different years of the Information Systems program, they will likely have very different Information Systems literacy. A student enrolled in the first year may not have the same tools to discuss a topic in depth that is highly complex, whereas a student enrolled in the third year has vast experience of the Eclipse IDE. Therefore, it was concluded that conducting interviews might have resulted in a skewed outcome.

2. Literature review

It is relevant to look at the scientific literature to understand what has been studied within the field of Technology Enhanced Learning.

2.1 Integrated Development Environment (IDE)

In short, an IDE refers to a type of computer software that usually includes a source code editor, a compiler or interpreter (or both), build-automation tools, and a debugger. Additionally, many IDE's may also provide a version control system or tools for creating a graphical user interface (Kline & Seffah, 2005). Carrington (2004) means that an IDE should meet a set of specified goals. They should 1) reduce the cognitive load on the developer, 2) free the developer to concentrate on the creative aspects of the process, 3) reduce any administrative load associated with applying a programming method manually, and 4) make the development process more systematic.

2.1.1 About the Eclipse IDE

The Eclipse Foundation is a non-profit association that provides a collaborative environment for open source software development, with over 400 open source projects, including the Eclipse IDE that is used in computer programming (Eclipse, n.d.). Eclipse IDE was originally created by IBM in 2001 and is used by millions of software developers all over the world. Eclipse IDE is built on mainly Java and is mostly used for Java applications, although plug-ins provide the possibility to build projects in a variety of programming languages (Eclipse, n.d.).

2.1.2 Usability of the Eclipse IDE

Aljafer & Cantrell (2020) discusses the advantages and potential challenges of using IDE's to teach introductory Java programming courses. The Eclipse IDE offers features like keyword highlighting, error highlighting, code suggestion, and the ability to compile and run code without using the command line. They also simplify file management for multi-file applications. However, some students may struggle to learn how to use the Eclipse IDE itself, which could take away from the time they should be spending learning programming concepts. Students may also get distracted by exploring the various features of the Eclipse IDE and struggle with things like showing or hiding windows (Aljafer & Cantrell, 2020). If a student receives a suggestion or example of code, they may assume that it is the correct code to use for a particular task. However, the suggested code may not actually perform the same task as the one the student is trying to accomplish, even though it looks similar. In other words, code suggestions can be misleading if they do not specifically match the requirements of the task or if they are not adapted to the particular context in which the code will be used. This can be confusing and frustrating for students who are trying to learn how to write code. While keyword highlighting

and error checking are helpful features, Aljafer & Cantrell (2020) means that relying on them exclusively may cause students to only focus on syntax errors and not logical errors since they may feel like the Eclipse IDE is a safety net that catches their mistakes.

2.1.3 Usefulness of the Eclipse IDE

Chen & Marx (2005) discussed advantages and disadvantages of using the Eclipse IDE in programming courses for new programmers. They decided to try the Eclipse IDE since it is built upon open-source technologies and provides a plug-in architecture suitable for teaching. Furthermore, it provides students with a popular professional environment. The authors however found that students with no programming experience might be overwhelmed by the advanced features of the Eclipse IDE. These students were already struggling with program-specific syntax and problem-solving strategies and the introduction of the Eclipse IDE might hurt their learning abilities. Chen & Marx (2005) also found that while the usefulness of the Eclipse IDE for students is well due to the professional environment, it can only be reached after a steep learning curve and huge student efforts. This is since the Eclipse IDE is characterized by how advanced and heavy it is, both in terms of learning and as an actual IDE. The authors concluded that the Eclipse IDE is only useful if the instructor continuously demonstrated the complexity of the Eclipse IDE and asked the students to follow suit. Even if this made the Eclipse IDE manageable for the students, they found that the students became too reliant on the built-in features and thus lost out on the logical understanding of how their code performed. Some students also decided to use simpler IDEs when they were given a choice.

2.2 Usefulness

Since the thesis aims to measure usefulness, this section will discuss what usefulness is and the challenges and opportunities that arise with it.

2.2.1 Usefulness as a term

Usefulness is a term that refers to the degree to which a product, service, or system can fulfill its intended purpose effectively and efficiently (Nielsen, 1993). In the context of information systems and technology, usefulness is often associated with the extent to which a tool or software can help users achieve their goals, enhance their performance, or complete tasks more efficiently (Davis, 1989). This concept is closely related to the perceived usefulness construct in the Technology Acceptance Model (TAM), which emphasizes the importance of users' perceptions of a technology's ability to improve their performance (Venkatesh & Davis, 2000).

2.2.2 Critical Review of Usefulness

While the concept of usefulness is essential in evaluating technology adoption, it is important to consider its limitations and potential implications. One limitation is that perceived usefulness might not always align with actual usefulness, as users may overestimate or underestimate the value of a technology, based on their experiences or biases (Venkatesh & Davis, 2000). Additionally, usefulness may be influenced by factors beyond the technology itself, such as user preferences, task complexity, or contextual factors, which may not be captured by the perceived usefulness construct alone (Marangunić & Granić, 2015). Despite these limitations, usefulness, and particularly perceived usefulness, remains a critical factor in understanding user acceptance and adoption of technology. As such, it is important to consider both perceived and actual usefulness when evaluating the effectiveness of a technology or system, as well as identifying ways to enhance its usefulness to better meet user needs and expectations.

2.2.3 Usefulness of IDE's

An IDE's perceived usefulness, which refers to the degree to which a user believes the IDE will enhance their learning and performance, plays a significant role in their willingness to adopt and use the IDE (Venkatesh & Davis, 2000). Understanding users' perceptions of IDE's can have significant implications for learning and technology adoption in educational settings. By addressing usefulness concerns, educators and developers can design and implement IDE's that better support students' learning needs and preferences, leading to higher levels of engagement, satisfaction, and ultimately, better learning outcomes (Naps et al., 2002). Furthermore, identifying and addressing the factors that influence students' perceptions can also improve technology adoption rates, as students are more likely to embrace and effectively use IDE's that meet their needs and expectations (Venkatesh et al., 2003).

2.3 Usability

Since the thesis aims to measure usability this section will discuss what usability is and the challenges and opportunities that arise with it.

2.3.1 Usability as a term

Usability is a term that describes the degree to which a product, system, or service can be effectively and efficiently used by its intended users (Nielsen, 1993). In the context of information systems and technology, usability focuses on how easily and effectively users can interact with a tool or software to achieve their goals. Usability encompasses multiple aspects, including learnability, efficiency, memorability, error prevention, and user satisfaction (Nielsen, 1993). The International Organization for Standardization (ISO) further defines usability as the extent to which a product can be used by specified users to achieve specific goals with

effectiveness, efficiency, and satisfaction in a specified context of use (ISO, 2018). Understanding and addressing the factors that contribute to the usability of a tool or system, such as the Eclipse IDE, can lead to improved user experiences, increased adoption rates, and better learning outcomes (Hartson & Pyla, 2012).

2.3.2 Critical Review of Usability

While usability is a crucial aspect of technology adoption and user satisfaction, it is important to consider its potential limitations and challenges. One limitation is that the evaluation of usability often relies on subjective user feedback, which might not capture the complete picture of a system's usability (Hornbæk, 2006). Users' self-reported experiences may be influenced by factors such as their personality, cognitive abilities, or emotional states, which may not directly relate to the system's design or functionality.

Another challenge in usability is the difficulty in identifying a universally applicable set of usability metrics or criteria, as the importance of different usability aspects may vary depending on the specific technology, user population, or context (Hornbæk, 2006; Albert & Tullis, 2008). Furthermore, usability is often intertwined with other factors, such as system performance, security, or aesthetics, which may also influence user acceptance and satisfaction (Hassenzahl, 2004). Focusing solely on usability might overlook the interplay between these factors and their impact on user experience as a whole. Despite these limitations and challenges, usability remains a critical factor in understanding user acceptance and adoption of technology. When evaluating a technology or system, it is important to consider multiple dimensions of usability, as well as the specific context in which the technology is used, to better understand and address users' needs and expectations.

2.3.3 Usability of IDE's

There is evidence suggesting that the goals specified for the usability of IDE's are not often met. Seffah & Rilling (2001) suggests that poorly designed IDE's are the culprit. The authors also suggest that this limits the usability of an IDE, since a poorly designed IDE makes it difficult for the developer to learn and use. They also argue that the documentation available for IDE's is difficult to understand and poses a major issue for users. Kline & Seffah (2005) mentions that there are several factors that make it difficult to design an IDE that is usable. An instance of this is when it could be advantageous to create the fundamental user interface for a source code editor in a similar style to that of a typical word processor, as this would make use of the developer's familiarity with this type of everyday office productivity tool for manipulating text. However, other components of an IDE must have a higher integration of representation of program elements, such as classes and packages, which is typically presented in an unfamiliar way (Kline & Seffah, 2005). Usability is a critical factor in shaping students' perceptions of IDE's. A study

by Grover et al. (2013) found that students prefer IDE's that are easy to use, have clear and concise documentation, and provide useful features to support their learning process.

2.4 Technology Acceptance Model (TAM)

This episode describes the main research model used for this study and is also a prerequisite for the quantitative survey.

2.4.1 The Structure of the TAM model

One of the most influential research models in the field of information systems is the Technology Acceptance Model (TAM). Davis and Bagozzi developed this model in 1989 to predict the adoption and acceptance of information systems and technology by individual users. The key factors in the model influencing the intention to use a technology are primarily the user's perceived usefulness (PU) and perceived ease-of-use (PEU).

TAM is grounded in several theories, notably the "Theory of Planned Behavior," which is a model illustrating human behavior and behavioral control. Another theory underpinning TAM is the "Theory of Reasoned Action," which is a model that explains social behaviors of an individual in a specific situation. Davis (1989) contends that TAM can be applied to measure technology acceptance among individuals for all types of information technology. Figure 2.1 demonstrates the structure of the model.

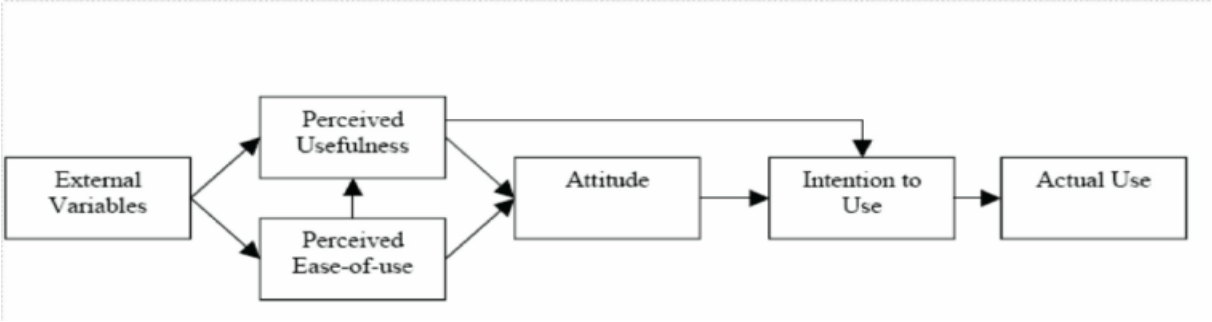


Figure 2.1 Technology Acceptance Model (Davis, 1989)

External Variables
External variables in the TAM model refer to factors that are external to the user but have an impact on their perceptions of usefulness and ease of use. These variables can include factors such as system design, training, social influence, organizational support, and individual characteristics (Venkatesh & Bala, 2008). By considering these external variables, researchers can better understand the context in which the technology is used and how it affects users' perceptions.

Perceived Usefulness & Perceived Ease of Use (Usability)
The difference between "Perceived Usefulness" and "Perceived Ease of Use" is that PU considers the extent to which a person believes a technology will improve their work performance, while PEU pertains to the degree to which a person believes the technology is easy to use. The Technology Acceptance Model consists of 12 statements, with six measuring PU and six measuring PEU (Lewis, 2019).

Attitude
Attitude, in the context of TAM, refers to an individual's positive or negative feelings towards using a technology. It is influenced by the user's perceived usefulness and ease of use of the technology (Davis, 1989). A positive attitude towards a technology indicates that the user is more likely to accept and adopt it, while a negative attitude may result in resistance or rejection of the technology.

Intention to Use
Intention to use is a user's self-reported likelihood to use a technology in the future. It is an important predictor of actual technology usage and is influenced by the user's attitude, perceived usefulness, and perceived ease of use (Davis, 1989). A strong intention to use a technology implies a higher likelihood of its adoption and usage by the individual.

Actual Use
Actual use is the real-world use of a technology by users. In the TAM model, it is the ultimate outcome of the user's intention to use the technology, influenced by their attitude, perceived usefulness, and perceived ease of use (Davis, 1989). Measuring actual use can provide valuable insights into the effectiveness of a technology in practice and help researchers and designers identify areas for improvement.

2.4.2 Critical Review of TAM

In the context of the research question, it is crucial to assess the implications and relevance of the Technology Acceptance Model as a suitable framework for this research. The strength of TAM lies in its ability to predict the adoption and acceptance of a specific technology based on users' perceived usefulness and ease of use (Marangunić & Granić, 2015). For this study, TAM provides a solid foundation to understand how students perceive the usability and usefulness of the Eclipse IDE, which can subsequently influence their willingness to adopt the software. By identifying the factors contributing to students' perceptions, the study can recommend improvements to the IDE's design and enhance its overall user experience.

However, there are certain limitations to consider when applying TAM to this research context.

First, TAM may not capture all the nuances and contextual factors that could influence students' perceptions of the Eclipse IDE, such as prior programming experience, personal preferences, or the learning environment (Marangunić & Granić, 2015). Additionally, the model primarily focuses on the individual level and may not consider the broader social or organizational factors that could impact the students' experiences. One such factor could be the classroom setting, if this has any effect on the student's perception of the IDE is difficult to estimate.

Despite these limitations, TAM remains a valuable model for this study, as it provides a robust framework to explore the relationship between students' perceived usability and usefulness of the Eclipse IDE (Marangunić & Granić, 2015). By leveraging TAM, the study can generate insights into the factors influencing students' experiences and thereby better support educational activities in programming.

2.4.3 The Relevance of TAM in the Context of Usability and Usefulness

The Technology Acceptance Model offers a well-established framework for investigating the factors influencing students' perceptions of the Eclipse IDE's usability and usefulness. By focusing on perceived usefulness and perceived ease of use, TAM can help provide insights into the factors that drive students to adopt and utilize the Eclipse IDE in the context of their programming education.

TAM has been widely applied to various types of information technology and in diverse contexts, including educational settings (Marangunić & Granić, 2015). Several studies have demonstrated the effectiveness of TAM in predicting technology adoption in educational environments, such as e-learning platforms, online courses, and learning management systems (Teo, 2011; Šumak, Heričko & Pušnik, 2011). As such, applying TAM to the context of the Eclipse IDE can help in understanding students' experiences.

Furthermore, TAM can be adapted and extended to include additional variables relevant to the research context (Venkatesh & Bala, 2008). In this case there are several options to consider when incorporating variables such as prior programming experience, learning environment, or instructional support in order to capture a more comprehensive picture of the factors affecting students' perceptions of the Eclipse IDE. This adaptability of TAM allows for more tailored research that addresses the specific needs and characteristics of the student population in question.

Moreover, TAM's focus on the individual user aligns with the research question's emphasis on students' perceptions, ensuring that the study captures the subjective experiences of the target audience. By exploring the relationships between perceived usefulness and perceived ease of use of the Eclipse IDE, the study can identify the aspects of the software that most significantly

impact students' experiences.

In summary, the Technology Acceptance Model is a highly relevant and appropriate framework for addressing the research question. Its emphasis on perceived usefulness and perceived ease of use, adaptability to incorporate additional context-specific variables, and focus on individual users make it a valuable tool for investigating students' perceptions of the usability and usefulness of the Eclipse IDE.

2.5 Students Perception of IDE's in Educational Settings

This section aims to focus on the existing research and findings related to how students perceive IDE's, like the Eclipse IDE, in educational contexts. This section will explore factors that influence students' preferences, learning outcomes, and overall experiences when using IDE's for programming and software development.

2.5.1 Factors Influencing Students' Preferences

Several factors influence students' preferences for IDE's in educational settings. These factors include the ease of use, available features, and the programming languages supported by the IDE (Altadmri & Brown, 2015). Additionally, the level of support and documentation, as well as the compatibility of the IDE with various platforms and devices, may affect students' preferences (Jonnalagadda & Petkovic, 2014). Understanding these factors can provide insights into designing IDE's that cater to students' needs and preferences in educational contexts.

2.5.2 Learning Outcomes and Experiences

The choice of IDE can have a significant impact on students' learning outcomes and experiences. A well-designed IDE can improve students' understanding of programming concepts, enhance their problem-solving abilities, and foster collaboration and communication among peers (Brusilovsky et al., 2010). Additionally, an IDE with a user-friendly interface and helpful features can reduce the cognitive load on students, allowing them to focus on learning and mastering programming skills (Kelleher & Pausch, 2005).

In summary, students' perceptions of IDEs in educational settings play a crucial role in shaping their preferences, learning outcomes, and technology adoption. By understanding and addressing the factors that influence these perceptions, including usability and usefulness, educators and developers can create more effective and engaging learning environments that support students in their programming and software development education.

2.6 Modes of Interaction

In the realm of distance education, modes of interaction play a crucial role in providing a meaningful and engaging learning experience for students. Garrison and Anderson's work on interaction in distance education highlights the importance of various forms of interaction that are essential for effective learning outcomes. This section discusses the modes of interaction in distance education as proposed by Garrison and Anderson (1998) and further elaborated on by Anderson (2003).

2.6.1 Garrison and Anderson's Six Modes of Interaction

Modes of Interaction propose six distinct modes of interaction in distance education. These modes include: 1) Student-teacher interaction: Refers to the communication between students and instructors, such as through online discussions, email, or video conferencing. This interaction is crucial for clarifying doubts, providing feedback, and encouraging student engagement. 2) Student-student interaction: Involves collaboration and communication among students, which can occur in online discussion forums, group projects, or peer review activities. This interaction fosters a sense of community and enables students to learn from one another's perspectives. 3) Student-content interaction: Refers to the engagement of students with course materials, such as readings, videos, and quizzes. This interaction is critical for knowledge acquisition and the development of cognitive skills. 4) Teacher-teacher interaction: Involves the collaboration and communication between instructors, often in the context of course design and professional development. This interaction allows teachers to share expertise and improve their teaching practices. 5) Teacher-content interaction: Refers to the engagement of instructors with course materials, which may include the creation, adaptation, or evaluation of content. This interaction ensures the quality and relevance of course materials. 6) Content-content interaction: Involves the dynamic interplay between various content resources, enabled by digital technologies and the semantic web. This interaction allows for the creation of new knowledge and the integration of diverse perspectives (Garrison and Anderson, 1998; Anderson, 2003).

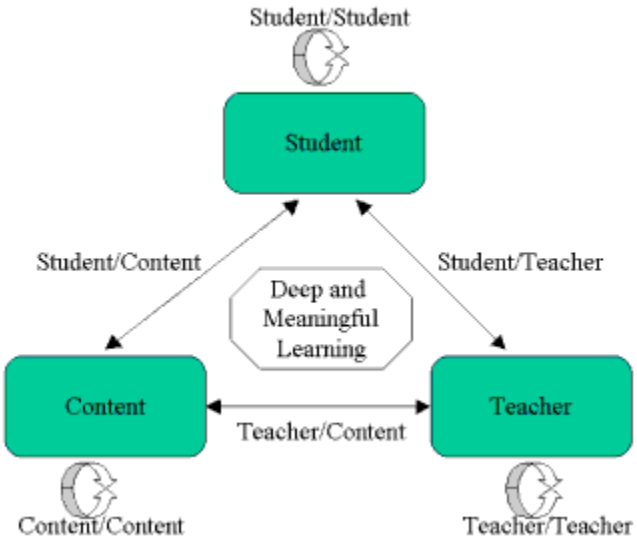


Figure 2.2 Modes of interaction (Garrison and Andersson, 1998)

2.6.2 The Importance of Balancing Modes of Interaction

Garrison and Anderson argue that a successful distance education experience requires a balance among the different modes of interaction. Designers and instructors must consider the unique needs and preferences of students, as well as the context in which distance education is being delivered. By creating learning environments that offer a variety of interaction opportunities, distance education programs can cater to diverse learning styles and facilitate deep and meaningful learning experiences.

2.6.3 Implications for Distance Education Practice

Understanding the modes of interaction proposed by Garrison and Anderson is essential for designing effective distance education courses. By incorporating a mix of student-teacher, student-student, and student-content interactions, educators can foster an engaging and supportive learning environment. Furthermore, considering teacher-teacher, teacher-content, and content-content interactions can help improve course design and facilitate the sharing of knowledge among educators. In conclusion, Garrison and Anderson's modes of interaction provide a valuable framework for designing and implementing successful distance education programs.

2.7 Five Aspects of Interface

Akindunjoye (2018) mentions that there are some key fundamental aspects of interface design. Language refers to how designers can work with words both as a tool for naming of features,

tabs and other interface-related items as well as a tool for attitudes and/or tones. Color can be used to set the mood for interfaces, as well as a concept of navigation. Imagery can be used as content, mood-setting and navigation for an interface. Typography can be used as content, interface and branding. Icons can be used as both branding and interface.

2.8 Categorization of Literature

Below (table 2.1) is a categorization of the above presented literature, with a summary of central concepts.

Categorization of literature

Concept	Keywords	Literature
Integrated Development Environment	History Usefulness Usability Objectives Programming Interface	Akindunjoye, 2018 Aljafer & Cantrell, 2020 Carrington, 2004 Chen & Marx, 2005 Eclipse, n.d. Kline & Seffah, 2005
Usefulness	Fulfillment of purpose Objective achievement Performance enhancement Biases External factors User needs/expectations Interface	Akindunjoye, 2018 Marangunić & Granić, 2015 Naps et al., 2002 Nielsen, 1993 Venkatesh & Davis, 2000 Venkatesh et al., 2003
Usability	Effectivity Efficiency Objective achievement User satisfaction Subjective External factors Lack of metrics IDE design Familiarity	Akindunjoye, 2018 Albert & Tullis, 2008 Grover et al, 2013 Hassenzahl, 2004 Hornbæk, 2006 Kline & Seffah, 2005 Nielsen, 1993 Seffah & Rilling, 2001 ISO, 2018 Hartson & Pyla, 2012
Technology Acceptance Model	Perceived usefulness	Davis, 1989

	Perceived ease-of-use Acceptance Adoption Attitudes Intentions Non-hollistic Insights Individuality	Lewis, 2019 Marangunić & Granić, 2015 Šumak, Heričko & Pušnik, 2011 Teo, 2011 Venkatesh & Bala, 2008
External variables (factors)	Interface Students Preferences Support IDE design Adoption Acceptance	Akindunjoye, 2018 Altadmri & Brown, 2015 Brusilovsky et al., 2010 Jonnalagadda & Petkovic, 2014 Kelleher & Pausch, 2005
Six modes of interaction	External factors Student-teacher Student-student Student-content Teacher-teacher Content-content Interaction Perspective	Garrison & Andersson, 1998 Andersson, 2003

Table 2.1: Categorization of literature

3. Methodology

3.1 Quantitative Method

The methodology used in this study was a quantitative research approach, characterized by systematic empirical research. The reason for employing a quantitative research method in this study was its strengths, particularly its ability to collect numerical data from a sample size, providing a comprehensive picture of the phenomenon under investigation. Quantitative research provides an efficient means to gather data, thus enabling the generalization of findings across groups of people to explain trends (Creswell, 2009).

In the context of this study, a survey, specifically, was used as the primary data collection instrument. Surveys, as a quantitative method, have the advantages of being fast to administer and capable of providing quantifiable responses. They also offer adaptability to the respondent's needs, as they can be completed at the respondent's own convenience (Bryman, 2008).

The survey questions were derived from the Technology Acceptance Model (TAM), a widely recognized and used model in quantitative studies in the field of information technology, originally proposed by Davis (1989). TAM is a useful tool for predicting how users will accept and use a new technology, with its questions adapted and modified for this study on the Eclipse IDE. The use of TAM in this context is well-founded, as it has been previously used in numerous studies examining the acceptance of software tools (Venkatesh & Davis, 2000).

The survey was conducted using Google Forms, a digital tool that allowed for efficient and fast distribution and collection of responses. This data collection method aligns well with the research question, as it facilitated the use of multiple-choice questions, often sufficient to capture users' opinions (Lewis, 2019). In retrospect, the chosen methodology and method proved to be effective in capturing the students' perceptions of the usability and usefulness of the Eclipse IDE in programming courses.

3.2 Sample Selection Method

Sample selection is an important step in the research process, providing a subset of the population that is representative and capable of providing meaningful insights relevant to the research question. The method used for this study was purposive sampling, a non-probability sampling technique where the researcher selects participants based on their characteristics and specific knowledge (Palinkas et al., 2015).

The sample for this study was selected based on certain criteria. Participants had to be currently enrolled in the Information Systems program at Lund University. This focus on Information

Systems students was due to their extensive interaction with the Eclipse IDE as part of their academic curriculum, providing them with a nuanced understanding and perspectives on its usability and usefulness. Furthermore, the sample was diversified to include male, female, and non-binary students of various age groups, enhancing the width of perspectives gathered.

The total population from which the sample was drawn consisted of around 300 students. The chosen sample size and the criteria for selection were deemed appropriate to answer the research question effectively and comprehensively. The goal was not just to identify common themes or issues faced by students in the Information Systems program when using the Eclipse IDE, but also to gather diverse insights that could potentially inform pedagogical strategies.

3.2.1 Pilot Study

A pilot study plays a crucial role in determining the sample size and assessing all aspects of the main study, while reducing the burden on researchers and participants, as well as conserving research resources. To fulfill its purpose, the factors outlined in the text must be well-defined and thoroughly completed before conducting the pilot study. Moreover, the pilot study offers valuable insights not only for the primary study but also for other comparable studies (Junyong, 2017).

The pilot study was conducted using three participants, all regarded as part of the research population. The participants were all currently enrolled in the Information Systems program at Lund University and spread across the three years (year one, two and three) of the program. The survey was considered complete when the pilot study was conducted, thus ensuring us that all major theories had been presented and defined. Finalized definitions of concepts were also presented in the pilot study. The participants were asked to complete the pilot study at their own pace, and the instructions were to analyze if there were any uncertainties in both the questions or the concepts. The pilot study was a success, in terms that we were given positive feedback about the layout of the survey, as well as how the questions were formulated. Therefore, the decision was made to not include any changes in the finalized survey.

3.2.2 Finalized Study

All three sample sets had groups on social media (Facebook). The surveys were distributed in these groups, giving the students in all three years the possibility to answer the survey. A link to a google form was posted, as well as information about the researchers and what the purpose of the study was. The participants were given the link to the google form. If they decided to participate, they were met with an information sheet where the purpose of the study as well as ethical aspects were clearly stated. If they agreed to participating further, the respondents were asked to give demographic and background information. They were then presented with three

sections where the actual questions about the Eclipse IDE were located. After responding to all questions, the survey was submitted.

3.3 Data Collection Method

In this study, a survey was chosen as the data collection instrument due to its suitability for gathering quantitative data from a large and diverse sample group (Creswell & Creswell, 2017). Surveys offer considerable flexibility, allowing respondents to participate at their convenience. In an academic setting, where students often have rigorous schedules, this flexibility is particularly valuable. Moreover, administering the survey online via Google Forms enabled access, reaching a larger population and enhancing the potential for generalization of the findings.

Anonymity is another important advantage of surveys. By ensuring responses were anonymous, it was anticipated that students would feel more comfortable sharing their true experiences and perceptions, leading to more honest data (Bryman, 2008). From a practical perspective, surveys are time and cost-effective, especially when collecting data on a large scale. The efficiency of surveys was particularly important in this study given the relatively large population from which the sample was drawn.

The survey conducted for this study consisted of five sections with a total of 27 questions aimed at achieving different metrics. The sections were Ethics, Demographic & Background, Perceived Usability, Perceived Usefulness, and External Factors. The survey incorporated both closed-ended and open-ended questions, as suggested by Desai and Reimers (2019). Open-ended questions allowed respondents to provide detailed insights into complex situations, while closed-ended questions eased the response process and reduced social pressure. However, the potential drawbacks of open-ended questions, such as discouraging responses and a potentially higher dropout rate, were carefully considered (Desai & Reimers, 2019). Furthermore, closed-ended questions, while easier to answer, could inadvertently provide cues to respondents about the researcher's expectations. Balancing these considerations, a mix of both question types was deemed appropriate for this study, resulting in a diverse and rich data set.

For the closed-ended questions, a Likert scale ranging from 1 to 7 is employed, with 4 being the neutral point. This choice is supported by several reasons. First, a 7-point Likert scale allows for a wider range of responses compared to smaller scales, such as the 5-point scale (Preston & Colman, 2000). This increased granularity enables respondents to more accurately represent their opinions, resulting in better data quality. Secondly, having an odd number of response options creates a clear midpoint, which in this case is 4. This neutral option serves as an anchor for respondents who do not feel strongly towards either end of the scale and prevents them from feeling forced to choose a side (Krosnick & Presser, 2010). Furthermore, research has shown that a 7-point Likert scale provides an optimal balance between reliability and discriminatory power while reducing the potential for response bias and satisficing (Simms, Zelazny, Williams, &

Bernstein, 2019). Thus, the use of a 7-point Likert scale with 4 as the neutral point in this survey is well-founded, as it promotes accurate and reliable data collection while minimizing potential biases and response errors.

3.3.1 Ethics

When designing a survey, it is important to consider the ethical implications of collecting data from respondents. The Ethics section of this survey is a crucial component that ensures that respondents understand their rights and the expectations of their participation. The Ethics section consists of information that a respondent must agree to before participating, including anonymity, usage requirements, and informed consent as well as the purpose of the study. Header 3.5 further explains the reasoning regarding ethics in this research.

3.3.2 Developing the Demographic & Background Questions

The Demographic & Background section of the survey served as a crucial component to gather important information about the respondents. This section was designed to validate that the participants were indeed currently enrolled in the Information System bachelor program at Lund University, a necessary criterion for the subsequent data analysis and conclusions drawn (Creswell & Creswell, 2017).

Furthermore, demographic information such as academic year and gender might have provided insights into potential differences in perceptions and experiences across different groups. For example, research has suggested that gender and academic year can influence technology acceptance and usage (Venkatesh & Morris, 2000). In summary, the Demographic & Background section was not only useful for confirming the eligibility of the respondents but also enabled a deeper understanding of the data collected, contributing to the robustness of the findings.

3.3.3 Developing the Main Survey Questions

Perceived Usability Questions

The Perceived Usability section of the survey was constructed to evaluate users' perceptions of the usability and satisfaction with the Eclipse IDE. The development of the questions in this section was grounded in one key resource: the Technology Acceptance Model (TAM) by Davis (1989).

The TAM (Davis, 1989), which focuses on perceived usefulness and perceived ease of use (usability), served as the theoretical framework for the questions. These questions were designed to gauge the users' experiences with the Eclipse IDE, encompassing their initial use, navigation, and the ability to locate specific features. Furthermore, feedback was solicited on aspects of the interface, aligning with the TAM’s emphasis on user perceptions of specific technology characteristics. This combination of theoretical and practical guidance led to the creation of a comprehensive Perceived Usability section. It facilitated the identification of areas where

students were satisfied or dissatisfied with their user experience, providing critical insights for the study.

Perceived Usefulness Questions

The Perceived Usefulness section of the survey was carefully crafted to assess users' perceptions of the Eclipse IDE's usefulness. The forming of the questions in this section was significantly influenced by the Technology Acceptance Model (TAM) proposed by Davis (1989). The TAM posits that perceived usefulness - the degree to which a person believes that using a particular system would enhance their job performance - is an important factor influencing the acceptance of a technology. Drawing from this theoretical construct, the questions were designed to appraise users' beliefs about the overall usefulness of the Eclipse IDE, as well as their experiences with specific features, including syntax coloring, code refactoring, and debugging. The objective was not just to gather data on what users found useful, but also to understand how these perceptions of usefulness could potentially impact their acceptance and usage of the Eclipse IDE in their educational journey.

Moreover, the formulation of the questions was informed by empirical studies and literature on IDE's. These references provided insights into the aspects of IDE's that users often find beneficial and thus helped in tailoring the questions to capture users' experiences accurately and comprehensively with the Eclipse IDE. Ultimately, the Perceived Usefulness section was designed to provide vital information to understand how the Eclipse IDE could better support students in their education, echoing the central pillars of the TAM model and empirical studies on IDE usage.

External Factors Questions

The External Factors section of the survey was created with the aim of understanding the influence of external factors on users' ability to use the Eclipse IDE effectively. Inspired by Davis's (1989) Technology Acceptance Model (TAM) and Garrison & Andersson's (1998) six modes of interaction, this section incorporated questions that dug into users' experiences with training and instruction, support from instructors and peers, and the availability of online resources, such as tutorials, forums, and documentation.

The questions were not initially developed explicitly based on these models; however, the framing and orientation of the questions aligned well with the key principles mentioned in both TAM and the six modes of interaction model. For example, questions investigating the availability and effectiveness of support from instructors and peers echo the collaborative interaction mode highlighted by Garrison & Andersson (1998), while the TAM model emphasizes the role of external variables such as training and support resources in shaping users' perceptions of a technology's usefulness and ease of use (Davis, 1989).

Thus, the External Factors section allowed for the identification of factors that could either help or hinder users' ability to navigate the Eclipse IDE effectively. The responses gathered from this section provided insights into the external factors influencing users' interaction with the Eclipse IDE, serving as a critical resource for identifying areas where improvements could better support users in their engagement with the Eclipse IDE.

3.4 Data Analysis Method

The data collected from the survey underwent a systematic and comprehensive process of analysis. First, each question was individually analyzed in order to gain insight into the respondents' views and experiences as captured in distinct sections of the survey. This process, a form of univariate analysis, helped to illustrate the general trends and patterns shown in the responses to each question.

For the open-ended questions, a thematic analysis was undertaken, which allowed for an in-depth exploration of the responses. This involved careful reading and re-reading of the responses, followed by the identification of emerging themes, as recommended by Braun and Clarke (2006). Each answer was individually analyzed, and this detailed scrutiny allowed for the identification of clear themes, contributing to a nuanced understanding of the responses.

The analyzed data was then visually represented using charts and graphs, which were created using data Google Forms tools. This helped to present the findings in a digestible and engaging manner, making it easier to show patterns and trends. The data analysis process was conducted with a commitment to transparency, and accuracy. The combination of these analysis methods provided a multi-faceted understanding of the data, and the insights drawn from this process are presented in the following section.

3.5 Ethics

Svartal (2001) mentions that there are ethical questions that need to be answered when conducting a survey. These ethical aspects should be the cornerstone when the survey is created. The ethical aspects considered in our survey are the demand for informed consent, anonymity and usage requirements. These are relevant and have been taken into consideration in this survey.

Informed Consent

Informed consent is a crucial ethical principle that underlies all research involving human subjects (Svartdal, 2001). It involves obtaining the voluntary and informed agreement of a participant to participate in a study after they have been provided with all relevant information about the study. The purpose of informed consent is to ensure that participants have sufficient information to make an informed decision about whether or not to participate in a study. This

information typically includes the study's purpose, procedures, risks, benefits, and the right to withdraw from the study at any time without penalty.

Obtaining informed consent is not only an ethical requirement but a legal one as well. In many countries, it is a legal requirement for researchers to obtain written consent from participants. Moreover, informed consent is an ongoing process, and researchers must ensure that participants continue to provide their consent throughout the study (Svartdal, 2001; Etikprövningsmyndigheten, n.d.). Researchers must be aware of potential power imbalances and ensure that participants are not coerced into participating or continue to participate if they no longer wish to do so (Etikprövningsmyndigheten, n.d.). Overall, informed consent is a fundamental principle that protects the welfare and autonomy of research participants and ensures that research is conducted ethically and responsibly.

Anonymity

Anonymity is an essential aspect of research ethics (Svartdal, 2001). It refers to the principle of protecting the identities of research participants. Anonymity ensures that participants can share their experiences, opinions, and personal information without fear of retribution or negative consequences. In research, anonymity is typically achieved by collecting data without linking it to participants' identities (Vetenskapsrådet, 2017). For example, researchers may assign a unique identifier to each participant to keep their identities anonymous. Protecting participants' identities can help to build trust and increase the likelihood of honest and accurate responses (Svartdal, 2001).

Usage Requirements

Usage requirements are an important aspect of research ethics that pertain to how data can be collected and used for research purposes (Svartdal, 2001). In order to protect the rights and privacy of participants, researchers must clearly define how data will be collected, stored, analyzed, and shared. This includes receiving informed consent from participants, ensuring their anonymity and confidentiality, and following strict ethical guidelines throughout the research process. Researchers should also consider the potential risks and benefits of data collection and ensure that participants are fully aware of their rights and responsibilities before consenting to participate in the study.

3.6 Limitations of the Survey

Andrade (2020) is of the opinion that surveys suffer from two methodological limitations. Firstly, the author mentions that a sample cannot be described, meaning that it is not possible to determine that the survey has reached the intended respondents. Secondly, it is suggested that biased respondents may assert themselves into the sample. By distributing the surveys in closed online groups where the members are supposed to be currently enrolled in the Information Systems program at Lund University, it is somewhat certain that the survey reached its intended

target audience. Sampling bias is harder to control. It may be so that the respondents are either very pleased or displeased with the Eclipse IDE and therefore answered the survey for their own gain.

3.7 Validity & Reliability

Validity and reliability are fundamental properties of a measuring instrument in research (Sürücü & Maslakçi, 2020). Although the two concepts are interrelated, they refer to distinct aspects of the measuring instrument. While a measuring instrument can be reliable without being valid, a valid instrument is more likely to be reliable. Nonetheless, reliability alone is insufficient to guarantee validity, since a reliable instrument may not necessarily capture the intended behavior or quality being measured. Consequently, Sürücü & Maslakçi (2020) mentions that researchers must evaluate both the validity and reliability of their measuring instrument to ensure that it meets these two criteria. Failure to meet these criteria may render the research findings unreliable and invalid.

Hence, it is crucial for researchers to recognize that the validity of the measuring instrument is essential for the success of any research project. A valid instrument measures what it is intended to measure accurately. In contrast, reliability refers to the consistency and stability of measurements obtained from the instrument over time (Sürücü & Maslakçi, 2020). While a reliable instrument ensures that the measurements obtained are consistent, it does not necessarily ensure that the measurements are valid. Therefore, a valid instrument that measures what it is intended to measure accurately is the first step towards producing reliable and credible results. Therefore, it is necessary for researchers to assess both the validity and reliability of their measuring instrument before interpreting research findings to ensure the accuracy and credibility of their research results. Below, a more detailed description of the two concepts are presented and how they are implemented in this study.

3.7.1 Validity

To ascertain the validity of this study, a quantitative approach was employed, utilizing a survey grounded in the Technology Acceptance Model (TAM) which was adapted for this research of the Eclipse IDE in order to assess students' perceptions of its usability and usefulness. The application of a well-established model like TAM enhanced the probability of accurately measuring the intended constructs.

The survey questions were devised to be unambiguous and specific, minimizing the risk of misinterpretation and ensuring the validity of the responses (Dillman, Smyth, & Christian, 2014). Furthermore, the sample group was carefully chosen, concentrating on students enrolled in the Information Systems program at Lund University with experience using the Eclipse IDE. This selection criterion, which focused on the relevant characteristics of participants (i.e., enrollment

in the Information Systems program and experience with the Eclipse IDE), ensured that the participants could provide pertinent and valid insights into the usability and usefulness of the Eclipse IDE, in accordance with the best practices for sample selection discussed by Bryman (2016).

3.7.2 Reliability

To guarantee the reliability of the study, several strategies were implemented to ensure consistency and accuracy in data collection and analysis. Firstly, a standardized survey instrument based on TAM was utilized, encompassing a set of predetermined questions posed to all participants (Creswell & Creswell, 2017). This helped maintain consistency in data collection across participants, reducing potential biases that may have emerged due to variations in the questions or the manner in which they were posed. Secondly, measures were taken to ensure consistent survey administration. Google Forms was used to disseminate the survey, allowing control over the format and presentation of the questions. Additionally, the use of an online platform ensured that all participants had equal opportunities to complete the survey at their own time, further enhancing the reliability of the data. In summary, by meticulously considering and addressing both validity and reliability throughout the research process, the study aimed to deliver high-quality results that offer valuable insights into students' perceptions of the usability and usefulness of the Eclipse IDE, supported by the literature on research design and data analysis (Bryman, 2016; Creswell & Creswell, 2017).

4. Empirical Findings

4.1 Quantitative Empirical Findings

This section will mention and highlight all relevant quantitative related information. Two rounds of distribution were done, with the first round gathering 44 respondents. 44 respondents were not considered enough, and therefore the second round of distribution was made. This gathered another 30 respondents, totalling 74 respondents. The link was open for respondents for one week. After one week data saturation was achieved, with 74 responses believed to be enough samples to draw conclusions from the population. There was also a clear indication that no more responses were being made, after a full day of no answers.

4.1.1 Demographic and Background Questions

The respondents were split quite evenly between the genders they identify with. 45,9% of the respondents identify as women, and 54,1% identify as men. This means that the research represents the sample group well, and that no measures need to be taken to exclude part of the population from the research findings. All respondents are also currently enrolled in the Information Systems bachelor program, which was a prerequisite for the findings to be valid. By establishing the user's specific academic context, the inquiry aims to provide valuable background information that can be used to better understand their responses and experiences regarding the Eclipse IDE.

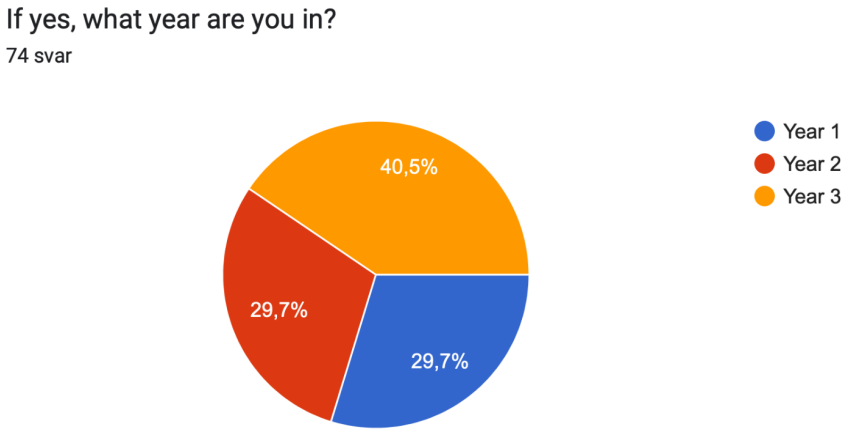


Figure 4.1.1: Current year of students

This question seeks to ascertain the user's current academic year within their educational program. The results are evenly distributed between year 1 and 2 (29,7%) while year 3 has 40,5%. With an even result spread across the three years, the sample group provides insights into how the population may perceive the Eclipse IDE. The respondents were also asked if they had

any prior experience with the Eclipse IDE before enrolling. The inquiry aims to establish the user's familiarity with the software, which may influence their learning curve, adaptability, and overall success in the program. By evaluating the user's previous exposure to the Eclipse IDE, the analysis can provide insights into the potential impact of pre-existing familiarity on their engagement with the software and other aspects of the program. 91,9% of the students had never used the Eclipse IDE before. Since most respondents did not have any prior experience with the Eclipse IDE before enrolling in the program, it can be assumed that the answers will not differ due to the previous level of familiarity.

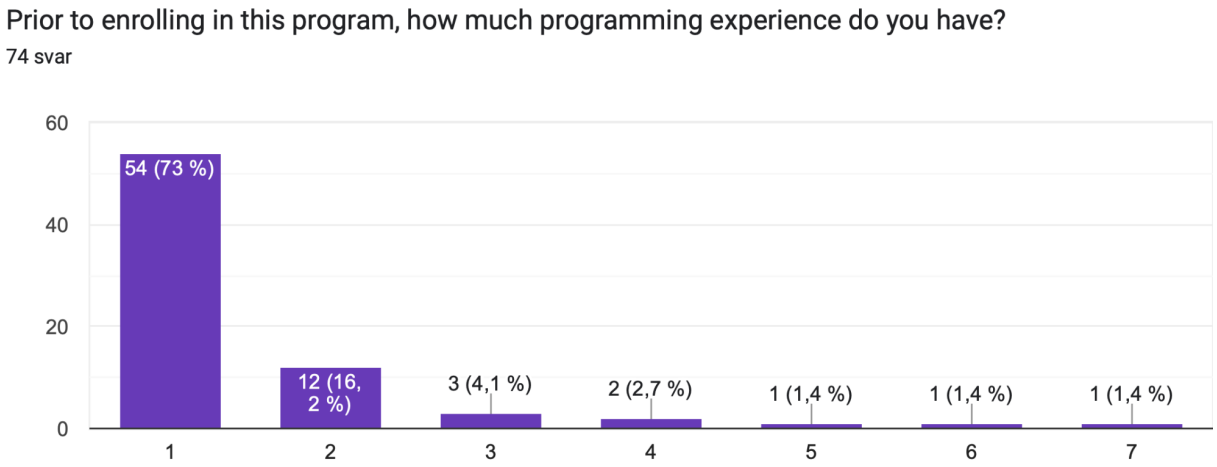


Figure 4.1.2: Programming experience

This question seeks to ascertain the user's prior programming experience before enrolling in the given program. The inquiry aims to understand the user's pre-existing knowledge, which may influence their learning experience, ability to adapt to new software, and overall success in the program. By evaluating the user's previous programming background, the analysis can provide insights into the potential impact of prior experience on their engagement with the Eclipse IDE and other aspects of the program. 73% of the students had no prior experience in programming before enrolling, and only 3 total respondents had much experience, it may be concluded that this background question will not have an effect on the findings.

4.1.2 Perceived Usability

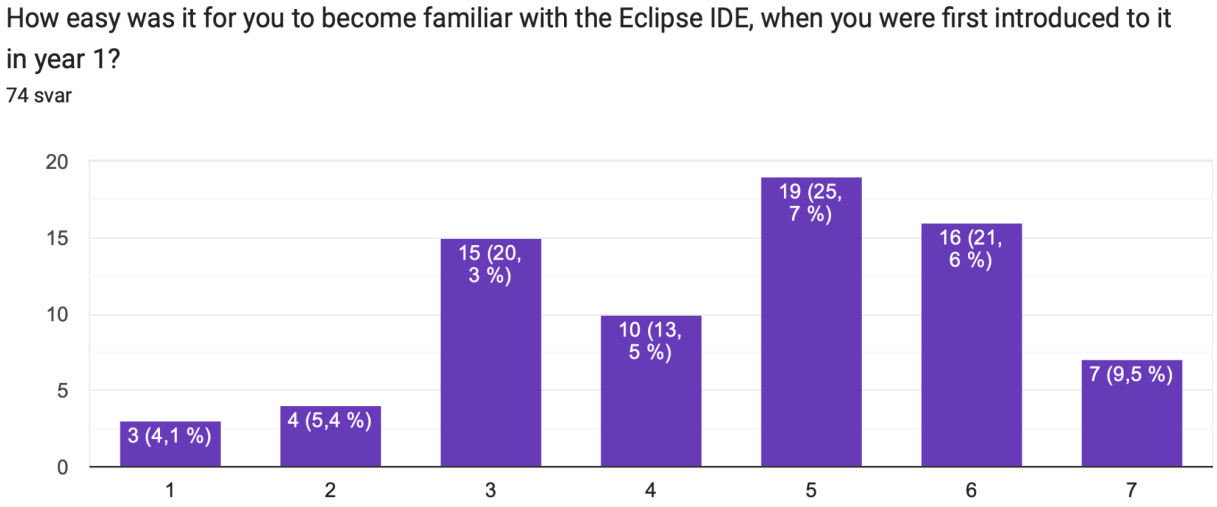


Figure 4.1.3: Familiarity with the Eclipse IDE

This question aims to assess the initial ease of acclimatization with the Eclipse IDE experienced by first-year students upon their introduction to the software. The primary focus of this inquiry is to gauge the usability and accessibility of the Eclipse IDE, as well as to identify any potential barriers or challenges that may have been encountered during the initial stages of engagement with the platform. 56,8% found it difficult to become familiar with the Eclipse IDE while 29,8% found it easy. This means that the respondents found it difficult to become familiar with the software, and that the learning curve might be steep.

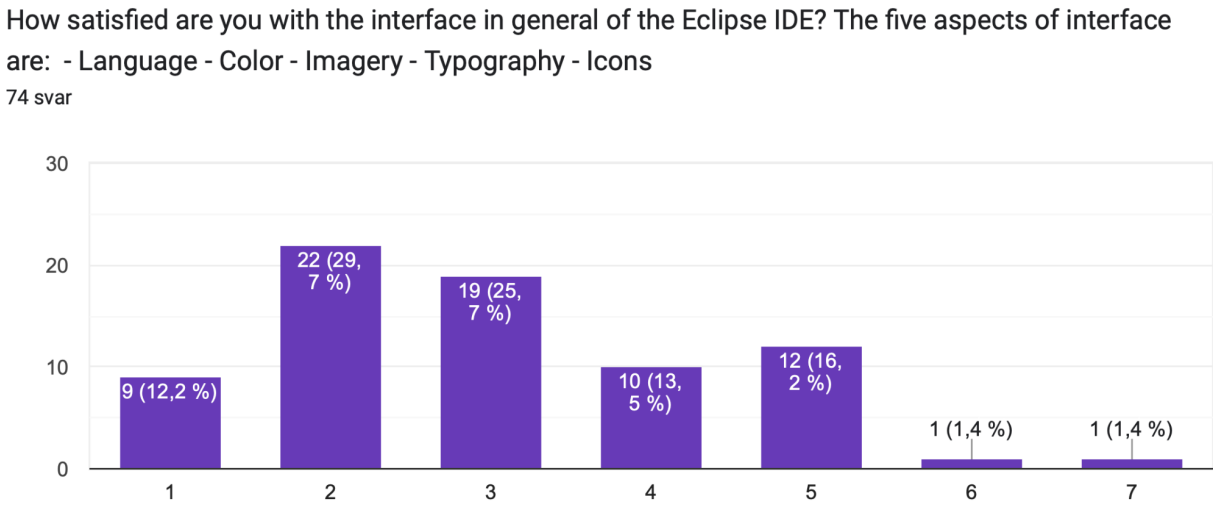


Figure 4.1.4: Satisfaction with the Eclipse interface

This question seeks to evaluate the overall satisfaction with the interface design of the Eclipse IDE by considering five distinct aspects: language, color, imagery, typography, and icons. The objective of this inquiry is to gain insights into the user experience and ascertain the effectiveness of the interface in facilitating efficient interaction with the software. By examining these specific elements, the analysis aims to identify areas for potential improvement and establish a comprehensive understanding of user satisfaction with the Eclipse IDE's interface. The results point out that most respondents are unsatisfied with the interface, where 67,6% showcasing this.

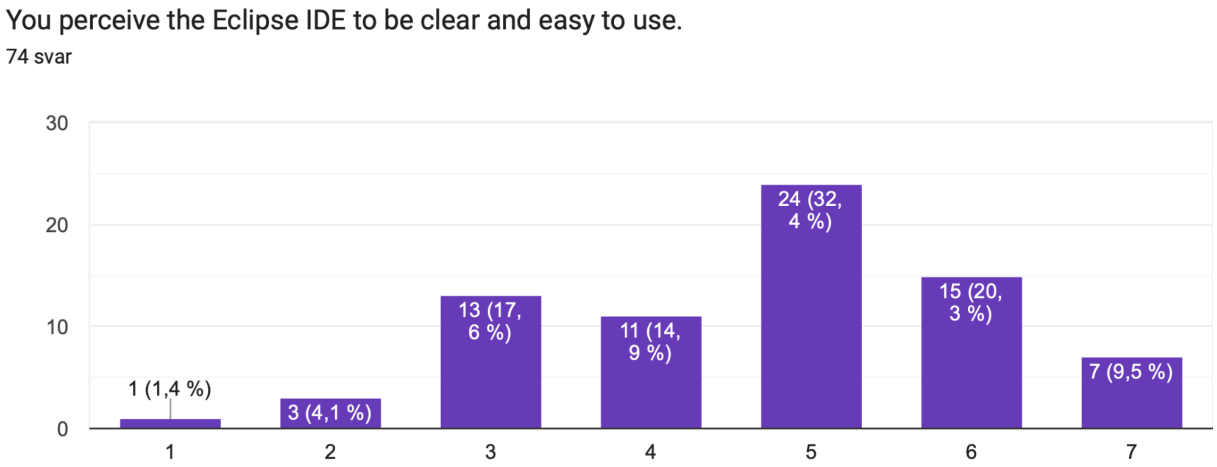


Figure 4.1.5: Perception of Eclipse as being clear and easy to use

This statement suggests that the individual perceives the Eclipse IDE as a usable and straightforward tool. The clarity and ease of use are indicative of an intuitive interface design that facilitates efficient interaction with the software, minimizing barriers and challenges that may hinder the user's productivity and learning experience. 62,2% disagrees with the statement that Eclipse IDE is clear and easy to use, of which 32,4% somewhat disagrees (6). 5,5% totally agree (1) or agree (2) while 17,6% somewhat agree (3). The findings show that most respondents perceive the Eclipse IDE as unclear and not easy to use. This is indicative of the Eclipse IDE's usability, with students finding it not to be very usable.

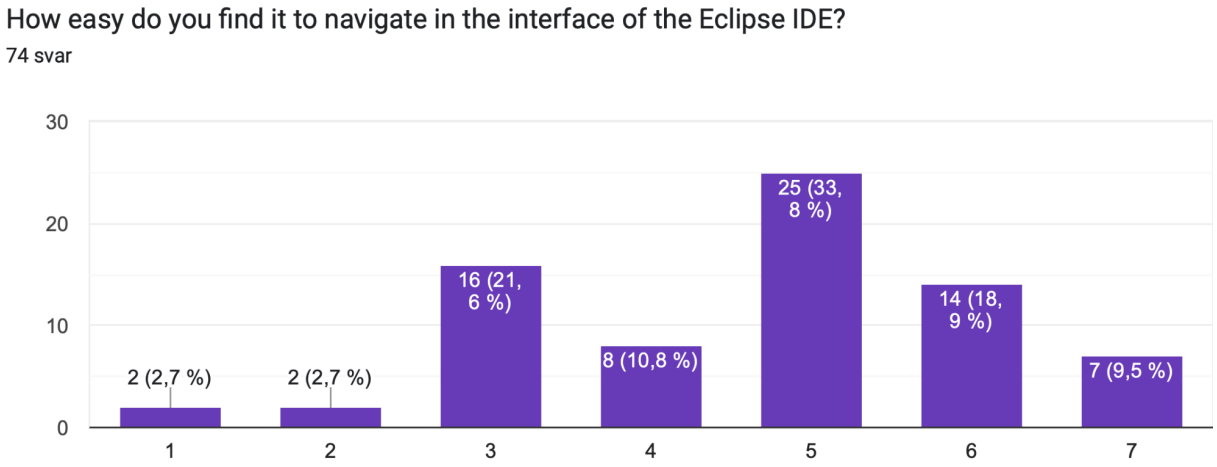


Figure 4.1.6: Navigation in the interface of Eclipse

This question aims to assess the user's perceived ease of navigation within the interface of the Eclipse IDE. By focusing on navigation, the question seeks to understand how intuitively the interface is organized and whether the layout and arrangement of elements contribute to a seamless user experience, allowing for efficient and effective interaction with the software. A majority finds it hard to navigate in the Eclipse IDE 62,2%, while 5,4% finds it very easy (1) or easy (2). The results show that the navigation of the Eclipse IDE's interface is somewhat difficult. A poor navigation can be perceived as low usability.

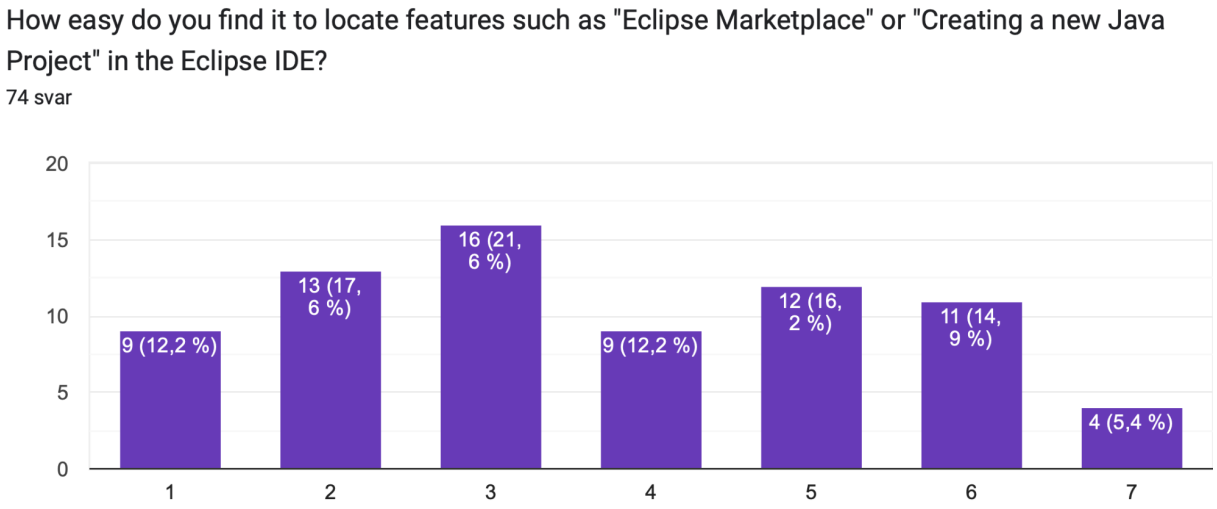


Figure 4.1.7: Locating features in Eclipse

This question seeks to evaluate the ease with which users can locate specific features, such as the "Eclipse Marketplace" or "Creating a new Java Project," within the Eclipse IDE. The focus of this inquiry is on the discoverability of essential functionalities, aiming to determine if the interface design effectively guides users to these features without unnecessary confusion or delay, thereby contributing to an efficient and productive user experience. 51,4% of the asked find it easy while 36,5% find it difficult to locate features. The findings show that the respondents find it easy to locate essential features within the Eclipse IDE. These features are among the most common features to use in an IDE, and shows that the usability of these features are high.

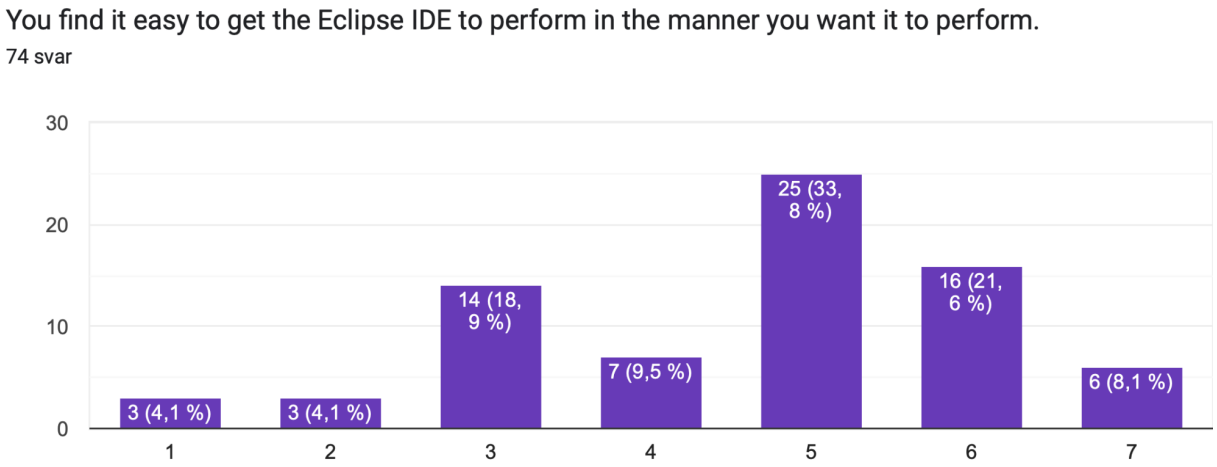


Figure 4.1.8: Ease of use in Eclipse

This statement indicates that the individual experiences a sense of ease in utilizing the Eclipse IDE to achieve their desired outcomes. A majority finds it easy to get Eclipse to perform in the expected manner, 63,5%. The user's ability to effectively manipulate the software suggests a well-designed and intuitive interface that aligns with their expectations and needs, ultimately contributing to a positive and efficient user experience.

4.1.3 Perceived Usefulness

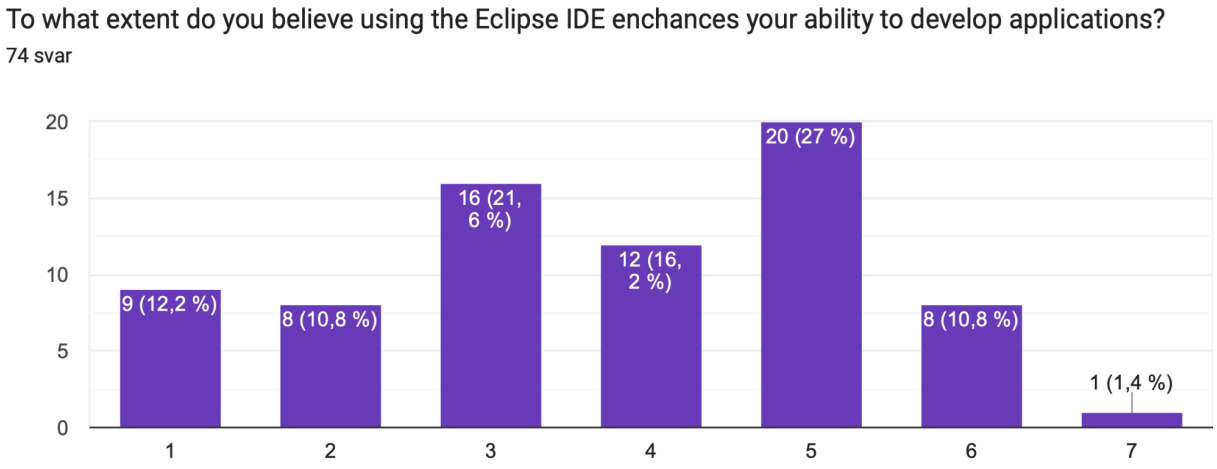


Figure 4.1.9: Eclipse influence in the ability to develop

This question aims to explore the degree to which using the Eclipse IDE enhances the user's ability to develop applications. The inquiry seeks to ascertain the software's overall effectiveness in facilitating the development process, as well as its potential to streamline workflows, improve productivity, and augment the user's programming capabilities. Ultimately, this examination will provide insights into the value and impact of the Eclipse IDE on the application development experience. The results imply that 39,2% believes the Eclipse IDE enhances the ability to develop applications while 44,6% believes it does not. The results provide an insight into how the respondents perceive that the Eclipse IDE actually helps them as developers. The answers are quite evenly distributed towards both ends of the spectrum, but leaning towards that the Eclipse IDE does not enhance their ability which should be considered as one of the primary objectives of an IDE.

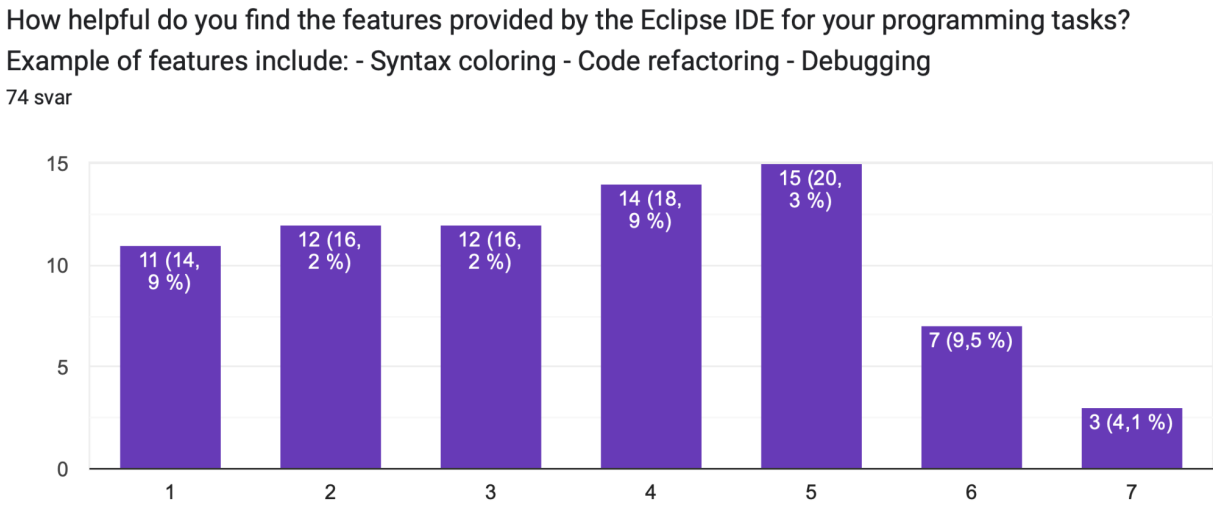


Figure 4.1.10: The features helpfulness in Eclipse

This question seeks to assess the perceived utility of the various features provided by the Eclipse IDE in supporting users' programming tasks. By specifically examining features such as syntax coloring, code refactoring, and debugging, the inquiry aims to gauge the extent to which these tools contribute to a more efficient, effective, and streamlined programming experience. The analysis will offer insights into the IDE's ability to enhance users' productivity and facilitate the application development process through its feature set. The results are quite evenly distributed except for helpful (6) with 9,5% and very helpful (7) with 4,1% although summed together 47,3% have a negative attitude towards the helpfulness of the Eclipse IDE's features.

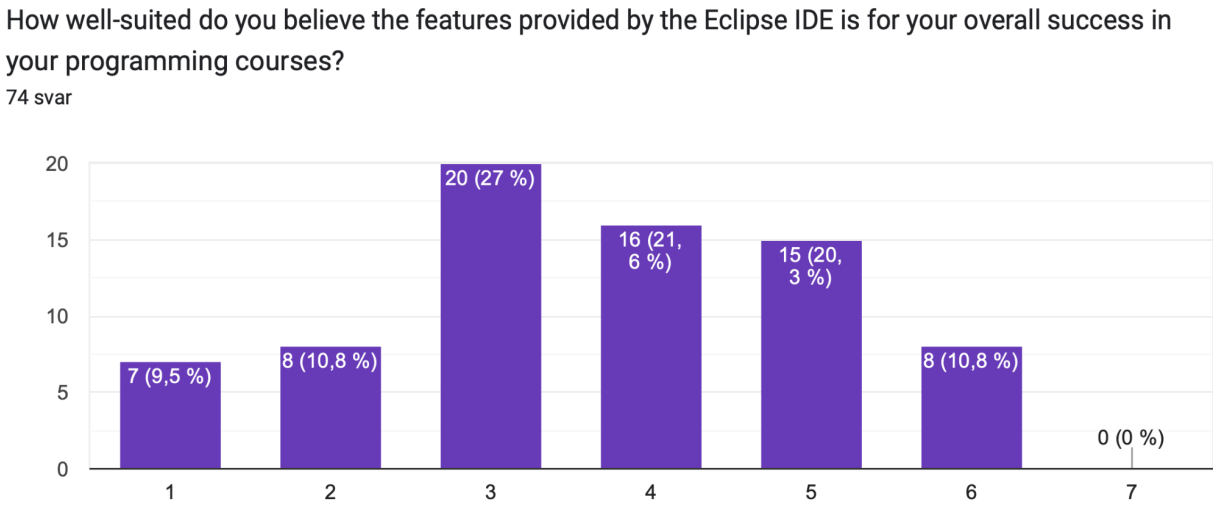


Figure 4.1.11: Well-suitedness of the features provided by Eclipse

This question aims to evaluate the perceived suitability of the features provided by the Eclipse IDE in contributing to the user's overall success in their programming courses. The inquiry seeks to determine the extent to which the IDE's tools and functionalities align with the user's educational needs and effectively support their learning experience. The results are concentrated towards the middle, but what stands out is that 0% believes the features of Eclipse IDE are very well suited for the success in programming courses, although 31,1% believe they are important (6) or somewhat important (5). 47,3% do not think the features are important. This shows that most of the respondents do not perceive the features of the Eclipse IDE as well-suited for their success. While a third of the respondents believe that the features are important or somewhat important, almost half do not think they are essential. This indicates that the IDE's features might not be entirely aligned with students' educational needs, which could impact their perceived usefulness and willingness to use the software.

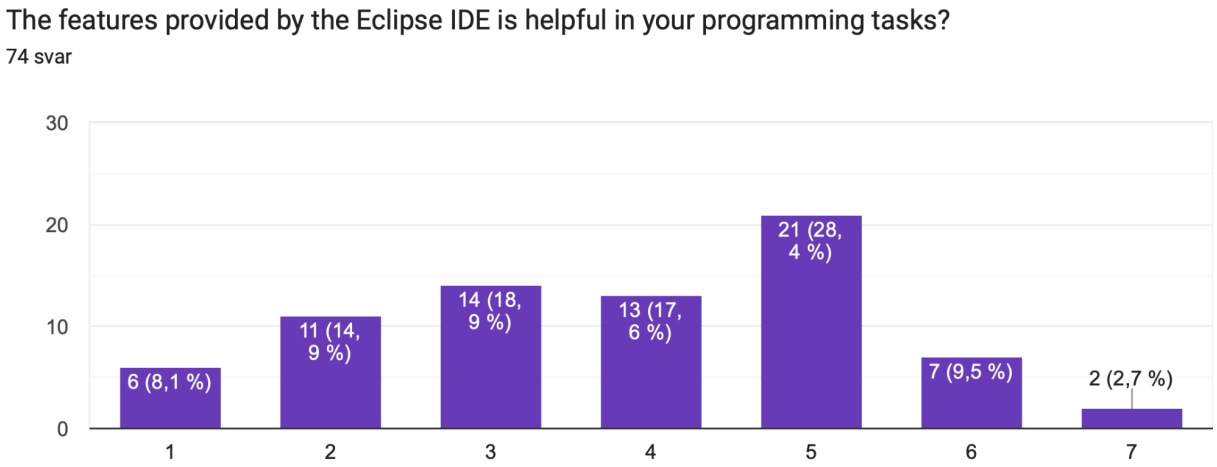


Figure 4.1.12: Helpfulness in features for programming tasks in Eclipse

This statement suggests that the individual perceives the features offered by the Eclipse IDE as helpful in facilitating their programming tasks. This implies that the IDE's tools and functionalities effectively support the user in streamlining their workflow, enhancing productivity, and enabling a more efficient application development process. The results show that 40,6% believe the features are beneficial and 41,9% believe they are not. This further highlights the variability in students' perceptions of the Eclipse IDE's usefulness and suggests that the features are not optimal in the support of students' programming tasks.

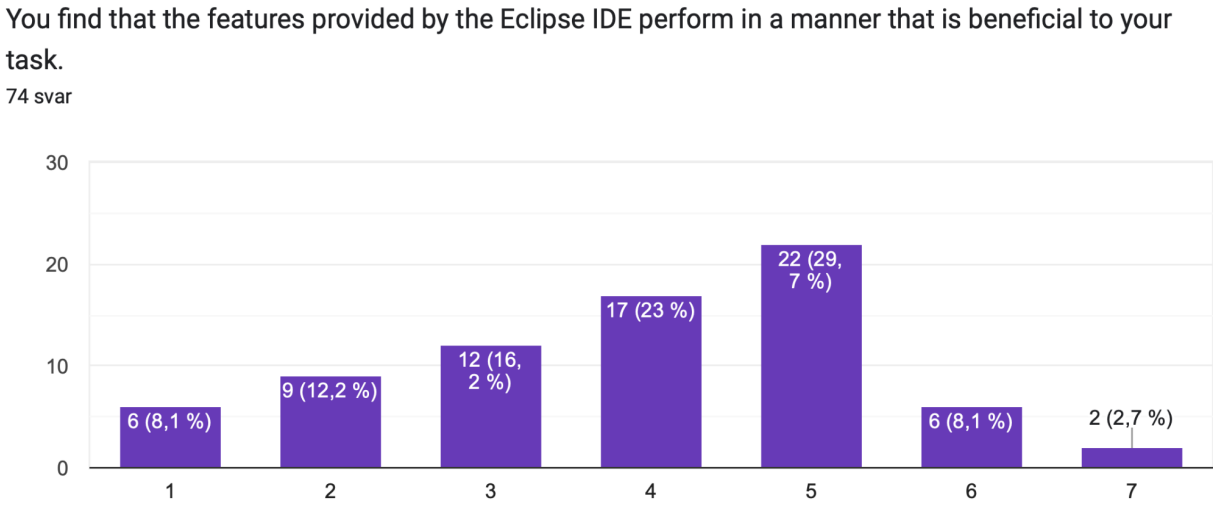


Figure 4.1.13: Features ease of use

This statement indicates that the individual experiences the features offered by the Eclipse IDE as performing in a manner that is advantageous for their tasks. This suggests that the tools and functionalities of the IDE effectively contribute to a more efficient and productive programming workflow, thereby enhancing the user's overall application development experience. The results are gathered towards the middle, almost 30% somewhat agree with the statement.

4.1.4 External Factors

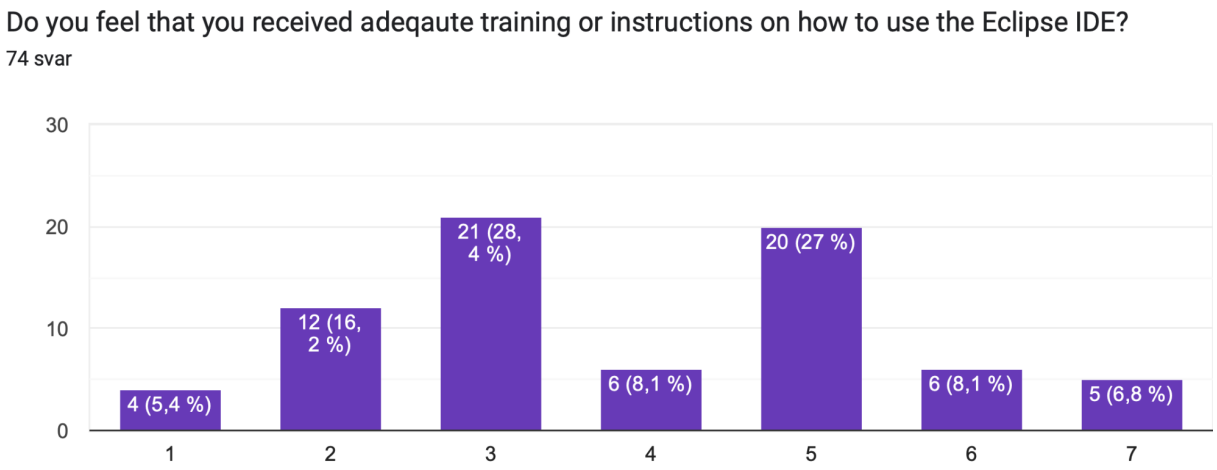


Figure 4.1.14: Training in Eclipse

This question seeks to determine whether the user feels they received adequate training or instructions on how to utilize the Eclipse IDE effectively. By focusing on the sufficiency of the provided training or guidance, the inquiry aims to assess the user's level of preparedness and confidence in using the software, as well as to identify any potential gaps in knowledge that may impact their ability to harness the full potential of the Eclipse IDE. The results are almost evenly split but a small majority, 50% (not counting the 8,1% who remained neutral) believes that they didn't receive adequate training on how to use Eclipse IDE. A more comprehensive instruction on how to use the Eclipse IDE could have enhanced the perceived usability and usefulness of the software.

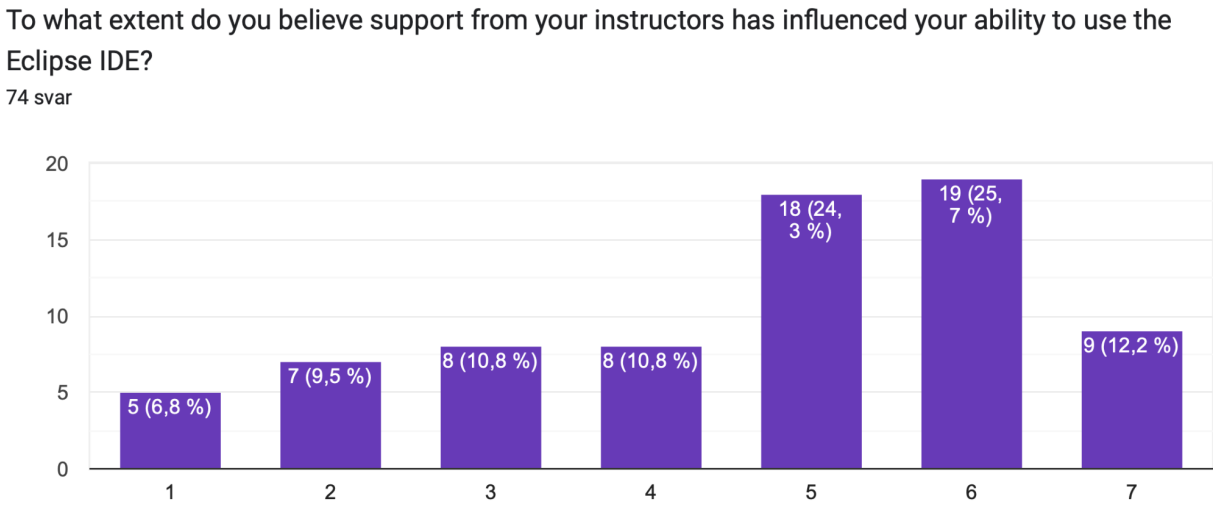


Figure 4.1.15: Support from instructors in Eclipse

This question aims to explore the degree to which support from instructors has impacted the user's ability to effectively use the Eclipse IDE. The inquiry seeks to understand the role of instructional guidance in facilitating the user's learning experience and enhancing their proficiency with the software. A majority, 62%, believes that instructors influence the ability to use Eclipse IDE.

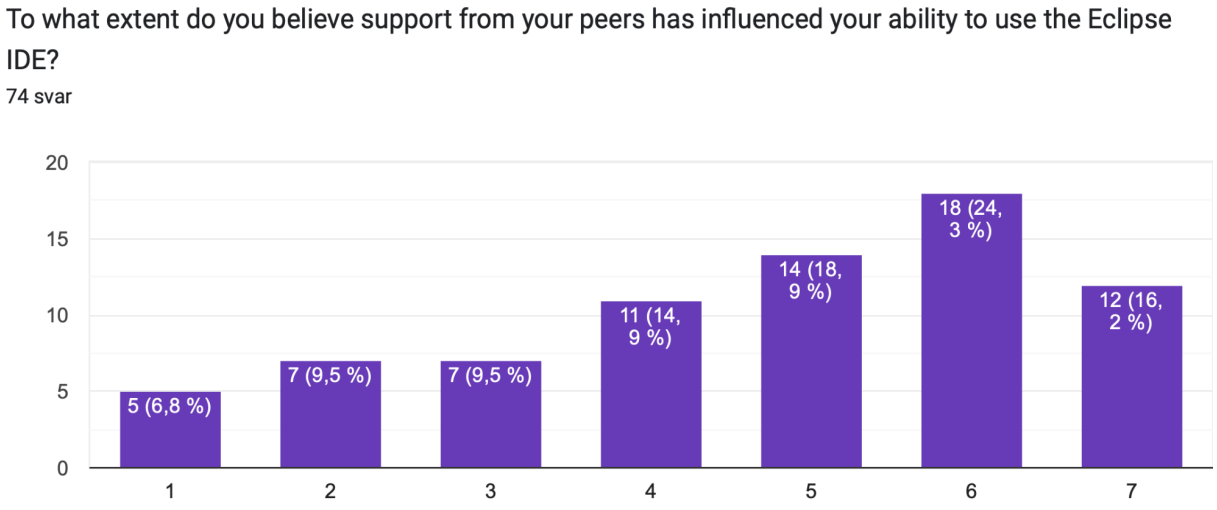


Figure 4.1.16: Support from peers in Eclipse

This question aims to investigate the degree to which support from peers has impacted the user's ability to effectively use the Eclipse IDE. The inquiry seeks to understand the role of peer collaboration and assistance in fostering the user's learning experience and enhancing their proficiency with the software. 60% of the results are pointing towards that peers influence the ability to use Eclipse IDE in some sense or a lot.

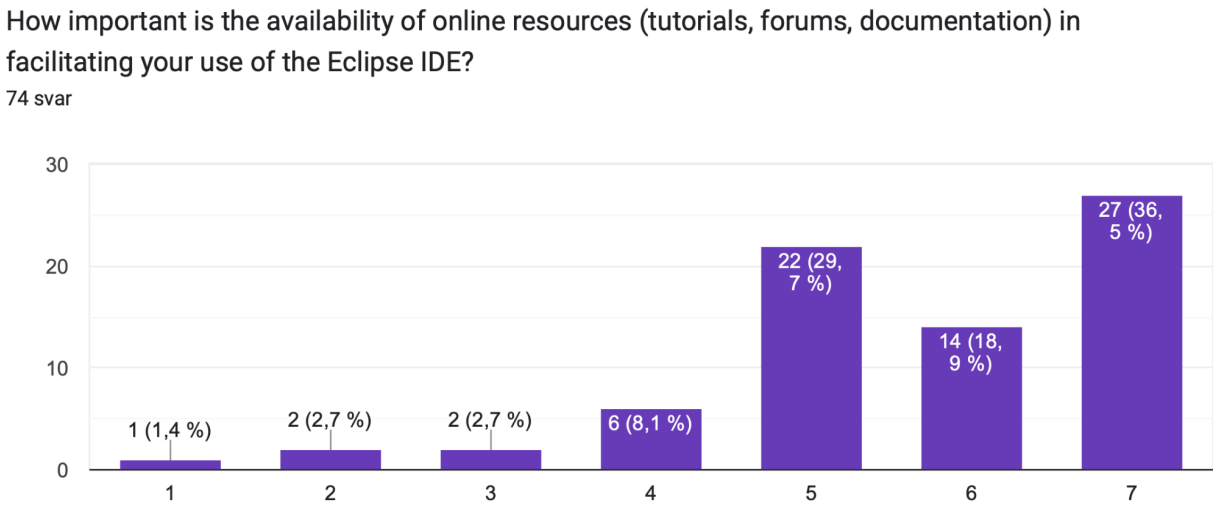


Figure 4.1.17: The importance of online resources

This question seeks to evaluate the importance of online resources, such as tutorials, forums, and documentation, in facilitating the user's effective use of the Eclipse IDE. The inquiry aims to determine the extent to which these resources contribute to the user's learning experience, as well as their ability to overcome challenges and develop proficiency with the software. By examining the role of online resources in supporting the user's interaction with the Eclipse IDE, the analysis will provide insights into the significance of external resources for successful software adoption and utilization. 85% of the users asked think online resources are important or very important in facilitating the use of the Eclipse IDE.

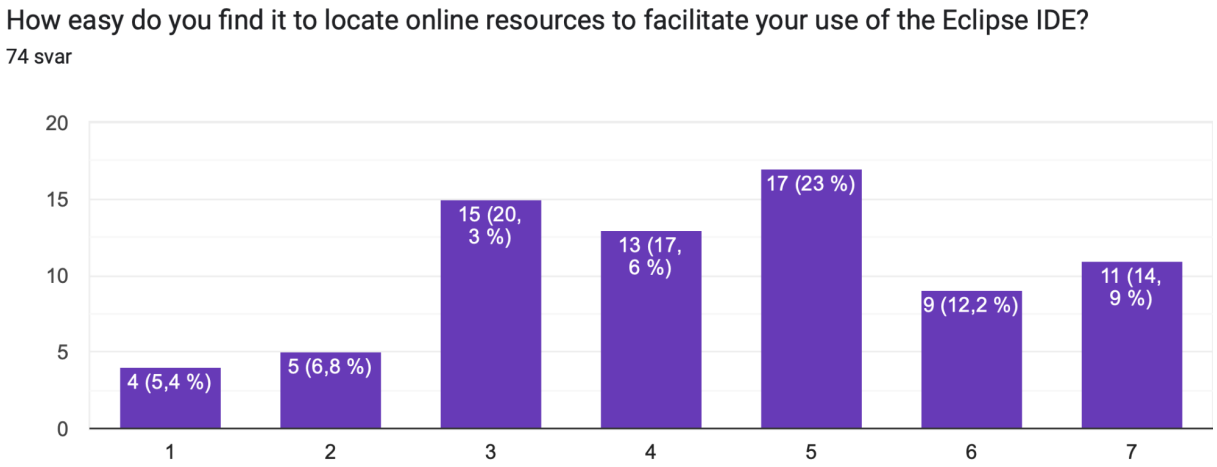


Figure 4.1.18: The importance in locating online resources

This question aims to assess the user's perceived ease of locating online resources that facilitate their effective use of the Eclipse IDE. The inquiry seeks to understand the accessibility and discoverability of external resources, such as tutorials, forums, and documentation, which contribute to the user's learning experience and problem-solving capabilities. By examining the user's ability to find relevant resources, the analysis will provide insights into the availability and organization of online support materials for the Eclipse IDE. The answers are scattered but it's leaning towards difficult, where 50,1% find it difficult and 17,6% remain neutral .

4.2 Qualitative Empirical Finding

This section will mention and highlight all relevant qualitative related information. The use of open-ended questions gave the respondents opportunity to elaborate on their answers on certain questions. The theme of these answers will be presented in this section. In appendix 2 you will find a complete list of all answers on open-ended questions.

4.2.1 Perceived Usability

The participants were asked to elaborate on how easy it was for them to become familiar with the Eclipse IDE, when they were first introduced to it. Some respondents found it hard at first, mainly due to a lack of programming knowledge or an insufficient introduction to the program. However, others found it manageable with good instructions or by following a guide for the set-up. The level of difficulty also depended on the scope of the programming tasks, with some finding it easy for basic programming and hard for more advanced tasks. Some respondents found the program not very intuitive or easy to learn, while others found it outdated compared to

modern IDE's. Overall, the difficulty of learning Eclipse seems to vary depending on the individual's level of experience and the quality of instruction or guidance they receive.

The respondents were also asked to give more detailed answers on how they believe the interface can be improved. The responses indicated that most respondents are not satisfied with the interface's overall design. They describe it as outdated, cluttered, messy, and unintuitive. Some respondents suggest specific changes, such as using dark mode by default or simplifying the icons and expanding menus. Others suggest a complete overhaul to make it more modern and easier to use.

4.2.2 Perceived Usefulness

The participants were asked to give a detailed answer to if they believe there are any features missing in the current version of the Eclipse IDE. The answers show that some respondents think that the Eclipse IDE lacks certain features that they have seen in other IDE's. These features include Github copilot, code autocomplete, code suggestions, and optimizations. However, some respondents cannot name specific features that are missing or are not aware of any.

4.2.3 External Factors

The respondents were asked to elaborate on how well they feel they received adequate training or instructions on how to use the Eclipse IDE. The responses indicate that some respondents think that they did not receive adequate training or instructions to use the Eclipse IDE. They describe the guides as difficult to understand, resulting in errors, and not providing clear instructions. However, some respondents feel that the teacher's extensive video tutorials helped them understand the Eclipse IDE better. Others suggest that a more detailed guide would be more efficient in learning how to use the tool. Some respondents feel that they only learned the basic functions of the Eclipse IDE, such as creating new projects/classes and using the debugger. One respondent suggests that the training was sufficient, but the poor user experience of the tool made it challenging to learn all its features.

Participants were also asked if they felt that the survey was not covering areas they wanted to elaborate on. The responses suggest that Eclipse IDE has limitations compared to other available IDE's. Some respondents have mentioned that the interface looks outdated and the tool is not very intuitive. However, the Eclipse IDE serves its purpose as a tool for students to learn how to code, and it has ample support for secondary Java-technologies taught in the program. Some respondents have also mentioned the need for better training or instructions to use the tool effectively. Overall, the responses suggest that there is room for improvement in the Eclipse IDE.

5 Discussion

The survey results offer valuable insights into the trends and patterns related to the theories explored in the literature review. This section presents an analysis of these trends in the context of the Technology Acceptance Model, usefulness, usability and external factors, examining their implications for the Eclipse IDE. Attached below is a table that summarizes the findings:

Summary of Findings

Demographic & Background	The respondents were all students currently enrolled in the Information Systems bachelor program at Lund University. They were evenly split between gender and current year of study. Most respondents had no prior experience with the Eclipse IDE or programming.
Usability	The majority of respondents found it hard to become familiar with the Eclipse IDE and were unsatisfied with the interface. Furthermore, the respondents perceived the Eclipse IDE as unclear and not easy to use, as well as the navigation as hard. The findings also show that the majority of the respondents found it somewhat easy to find common features as well as getting the Eclipse IDE to perform in a suitable manner.
Usefulness	The majority of respondents feel that the Eclipse IDE may not enhance their ability to develop applications, and do not experience that the features are helpful or well-suited for their success in programming tasks.
External factors	The majority of respondents believe that they did not receive adequate training of the Eclipse IDE, but believe that support from instructors and peers influence their ability to use the software. Furthermore, the majority of respondents experienced that online resources should be easy to find and will enhance their ability to use the Eclipse IDE.

Table 5.1: Summary of findings

5.1 Usability

In this section, the study explores the relationship between usability and technology acceptance using the Technology Acceptance Model in the context of the Eclipse IDE.

5.1.1 Familiarity and Perceived Usability

According to Figure 4.1.3, the participants faced challenges in becoming familiar with the Eclipse IDE, while only a smaller portion found it usable. This finding implies that the perceived usability, a critical determinant of technology acceptance in TAM, may not be optimal for the Eclipse IDE. The literature suggests that usability significantly influences users' acceptance and adoption of technology (Davis, 1989), and this initial difficulty in familiarization may negatively affect students' attitudes and intentions to utilize the software.

A difficult-to-learn interface may deter users from fully exploring and utilizing the software's features, resulting in reduced satisfaction and productivity (Venkatesh & Davis, 2000). This observation is consistent with Nielsen's (1993) usability principles, which emphasize learnability as a vital aspect of usability. Figure 4.1.6 shows that a majority of the respondents find it hard to navigate in the interface of the Eclipse IDE while Figure 4.1.7 shows that more than a third of the sample find it difficult to locate two of the most commonly used features. In the context of the Eclipse IDE, this suggests that educators could value a more learnable interface which may have positive effects on the user experience and promote software acceptance.

5.1.2 Perceived Usability Satisfaction and Navigation

Figure 4.1.6 shows that a large majority of the respondents experienced difficulties navigating the Eclipse IDE interface, suggesting that the software may not meet the usability criteria of efficiency and memorability (Nielsen, 1993). These findings, in the context of TAM, indicate that there is room for improvement in the perceived usability as users struggle with interface navigation and feature location. Previous research demonstrates that the significance of various usability aspects may depend on the specific technology, user population, or context (Hornbæk, 2006; Albert & Tullis, 2008; Hassenzahl, 2004). Thus, it is essential to consider these factors when evaluating the usability of the Eclipse IDE.

5.1.3 Connecting Perceived Usability to TAM

The Technology Acceptance Model (TAM) posits that perceived usability plays a significant role in determining users' intention to adopt and utilize a technology (Davis, 1989). In the case of the Eclipse IDE, the varying perceptions of usability among students could affect their inclination to adopt and use the software in their programming education. As literature suggests, improving the perceived usability of a technology can result in higher adoption rates and better learning outcomes (Naps et al., 2002; Marangunić & Granić, 2015).

The survey results suggest that students' perceptions of the Eclipse IDE's usability might influence their decision to adopt and use the software. While some students find the Eclipse IDE easy to use and effective in supporting their programming tasks, others struggle with aspects such as learnability, navigation, and interface design, which aligns with previous studies (Seffah & Rilling, 2001; Kline & Seffah, 2005). Educators can use this information to understand how students' perceptions of the Eclipse IDE's usability might impact technology adoption rates and learning outcomes. By acknowledging the relationship between usability and technology adoption, educators can more effectively evaluate the suitability of the Eclipse IDE for their programming courses and make informed decisions about its implementation and instructional support.

Furthermore, recognizing the challenges faced by students, as highlighted by Seffah & Rilling (2001) and Kline & Seffah (2005), educators can identify areas for improvement in the Eclipse IDE, such as enhancing documentation clarity, providing more intuitive navigation, and offering features that support the learning process (Grover et al., 2013). By addressing these usability issues, educators can contribute to a more positive user experience and potentially increase the likelihood of students adopting and using the Eclipse IDE for their programming tasks.

5.2 Usefulness

This section aims to explore the role of usefulness in the context of the Eclipse IDE and its influence on students' technology acceptance, based on the findings of the survey and the Technology Acceptance Model.

5.2.1 Perceived Usefulness and Actual Use

Despite challenges in learnability and navigation, Figure 4.1.8 shows that the majority of the respondents reported easily achieving the expected performance with the Eclipse IDE. This finding suggests a relatively high perceived usefulness for the Eclipse IDE, another key determinant of technology acceptance in TAM. Davis (1989) defines perceived usefulness as the extent to which an individual believes a technology will improve their work performance. In this case, the Eclipse IDE appears to satisfy this criterion for many users.

Nonetheless, it is crucial to consider the broader context in which the Eclipse IDE is employed, such as the learning environment, instructional support, and prior programming experience. These factors may also influence users' perceptions and experiences (Marangunić & Granić, 2015). It is important to consider that all students are exclusively provided with support for the Eclipse IDE. This could mean that students perceive the usefulness of the Eclipse IDE as higher, since they do not have any other tools where they get the same support. To clarify, the students may perceive the Eclipse IDE as useful, since there are no other viable options. Moreover, an assessment of the actual use of the Eclipse IDE could have provided valuable insights into the software's effectiveness in practice, as discussed by Venkatesh et. al (2003).

5.2.3 Perceived Usefulness and the Eclipse IDE

The results indicate a mixed perception of the Eclipse IDE's usefulness among students. Figure 4.1.9 shows that while a larger portion of the respondents believe that the Eclipse IDE enhances their ability to develop applications, more participants do not share this view. This suggests that the perceived usefulness of the Eclipse IDE varies significantly among students, which might be influenced by factors such as prior programming experience, personal preferences, or learning environment, as discussed in the literature review by Marangunić & Granić (2015) and Chen & Marx (2005).

Moreover, the results reveal an overall uncertainty regarding the helpfulness of the Eclipse IDE's features (Figure 4.1.13). Although a small percentage find the features helpful and very helpful, the majority of respondents have neutral or negative opinions on this aspect. This indicates that there might be implications in the design and implementation of the Eclipse IDE's features to better support students' programming tasks and enhance their perceived usefulness. Altadmiri & Brown (2015) discuss that availability of features enhances perceived usefulness for students. With the findings showing that the students have negative opinions of the features, this may be due to a lack of availability. The results may also mean that students feel that the Eclipse IDE's features increase their cognitive load, as discussed by Kelleher and Pausch (2005) as well as Carrington (2004), and that they therefore feel negatively about them.

5.2.4 Connecting the Findings to TAM

According to the TAM, perceived usefulness is a crucial factor in influencing users' intention to use a technology (Davis, 1989). In the context of the Eclipse IDE, the mixed perceptions of usefulness among students could impact their willingness to adopt and use the software for their programming education. As the literature review indicates, enhancing the perceived usefulness of a technology can lead to higher adoption rates and better learning outcomes (Naps et al., 2002).

The survey results suggest that there might be opportunities for improvement in the design and implementation of the Eclipse IDE's features to enhance their perceived usefulness and better support students' programming tasks. By addressing these concerns and tailoring the IDE's features to meet students' needs and expectations, educators can potentially increase technology adoption rates and improve the overall educational experience.

The findings from the survey highlight the importance of considering perceived usefulness when evaluating the effectiveness of a technology or system, such as the Eclipse IDE, in the context of programming education. As the literature review and TAM framework suggest, addressing the factors that influence students' perceptions of usefulness can lead to improved technology adoption rates and better learning outcomes.

The mixed perceptions of usefulness found in the survey results imply that there are areas in the Eclipse IDE that are sub-optimal to meet the needs and expectations of students. By identifying and addressing these areas, educators can work towards enhancing the overall usefulness of the Eclipse IDE and, consequently, support students' learning and performance in programming courses.

5.3 External Factors

The survey results provide valuable insights into the external factors that may influence students' perceptions of the Eclipse IDE in educational settings. These factors include training in Eclipse, support from instructors and peers, and the importance and accessibility of online resources. In the following sections, the trends observed in the survey are discussed and connected to the Technology Acceptance Model and the literature review.

5.3.1 Training in the Eclipse IDE

The survey results reveal a near-even split among respondents regarding the adequacy of training in the Eclipse IDE, with a slight majority (50%) indicating that they did not receive sufficient training. This suggests that there may be gaps in the training provided to students, which could potentially impact their perceived usability and perceived usefulness of the Eclipse IDE, as highlighted by the TAM framework (Davis, 1989). A lack of adequate training may contribute to students experiencing difficulties in using the software, thus negatively affecting their willingness to adopt and continue using the Eclipse IDE for their programming education. As indicated in the literature review, providing effective training and instructional support is essential in enhancing students' perceptions of technology (Venkatesh & Bala, 2008). Additionally, incorporating modes of interaction (Garrison and Anderson, 1998; Anderson, 2003), such as student-teacher and student-content interactions, can help improve the effectiveness of the training experience.

5.3.2 Support from Instructors and Peers

The survey results demonstrate that the majority of respondents (62%) believe instructor support has a significant influence on their ability to use the Eclipse IDE. Additionally, two thirds of the respondents indicate that peer support plays a role in their ability to use the software. These findings align with the notion of external variables in the TAM model, which suggests that factors such as instructional support and social influence can impact users' perceptions of usefulness and usability (Venkatesh & Bala, 2008). By fostering a supportive learning environment with guidance from instructors and collaboration among peers, educators can enhance students' perceptions of the Eclipse IDE, which may in turn improve technology adoption rates and overall learning outcomes (Jonnalagadda & Petkovic, 2014, Brusilovsky et. al., 2010). Moreover, applying Garrison and Anderson (1998) and Andersson (2003) concepts of student-teacher and student-student interactions can further strengthen the support system for students using the Eclipse IDE.

5.3.3 The Importance and Accessibility of Online Resources

In Figure 4.1.17 A substantial majority (85%) of respondents consider online resources to be important or very important in facilitating the use of the Eclipse IDE. This highlights the significance of external resources, such as tutorials, forums, and documentation, in supporting students' learning experiences and problem-solving abilities when using the software. However, Figure 4.1.18 also reveals that half of respondents find locating online resources to be difficult, while a small portion remain neutral on the issue. This suggests that the accessibility and organization of these resources may be sub-optimal, potentially hindering students' ability to effectively utilize the Eclipse IDE and negatively impacting their perceptions of the software's usefulness and usability.

Connecting these findings to Garrison and Anderson's (1998) modes of interaction, it is essential to consider the student-content interaction when designing and organizing online resources. By making these resources easily accessible and well-structured, students can engage more effectively with the content and enhance their learning experience with the Eclipse IDE. By extension, it is evident that addressing external factors such as training, instructional and peer support, and the availability and accessibility of online resources can significantly influence students' perceptions of the Eclipse IDE. Enhancing these external factors can improve students' perceived usefulness and usability, leading to higher adoption rates and better learning outcomes (Naps et al., 2002).

6. Conclusion

The survey results and literature review provide insights for educators to make informed decisions about the positives and drawbacks of the Eclipse IDE in educational settings. External variables influence the perceptions of usability and usefulness of students, in regards to the Eclipse IDE. The research questions this study aimed to answer was:

What are students' perceptions of the usability and usefulness of the Eclipse Integrated Development Environment (IDE) in programming courses?

To answer that question you have to take into consideration the external factors that influence the perceived usefulness and usability of the Eclipse IDE. Students enrolled in the Information System bachelor program do not perceive the Eclipse IDE as either very useful or usable. They found the software hard to become familiar with and navigate through, albeit easy to get to perform in the manner they wanted. The respondents further answer that the Eclipse IDE may not enhance their programming abilities. Furthermore, the student perceives that the support around them when encountering the Eclipse IDE is not adequate, which may influence their perceptions.

The following insights may help educators make more informed decisions when evaluating students perceived usability and usefulness of the Eclipse IDE: 1) Be aware of the importance of learnability and consider the value of offering onboarding experiences, contextual help, and clear documentation to support students in their initial encounters with the Eclipse IDE. 2) Take into account the interface design, as well as the navigation and discoverability of essential features. Understanding these aspects can help educators address potential challenges faced by students and support them in their learning process. 3) Consider the influence of context-specific variables, such as prior programming experience, learning environment, and instructional support, when evaluating the Eclipse IDE. 4) Foster a positive attitude towards the Eclipse IDE by emphasizing its benefits and value in the context of programming education. Providing ongoing support and resources to help students overcome any challenges or barriers they may encounter can contribute to a more favorable attitude towards the Eclipse IDE. Encouragement and support from instructors and peers can further enhance students' experience with the software.

In conclusion, understanding and addressing the external factors revealed by the survey results is crucial in shaping students' perceptions of the Eclipse IDE and, by extension, their willingness to adopt and use the software for programming education. The connection between these external factors and the TAM framework highlights the importance of considering perceived usefulness, as well as usability when evaluating the Eclipse IDE for educational purposes. By considering these factors, educators can make more informed decisions about the choice of implementing

the Eclipse IDE and provide appropriate support to help students overcome potential obstacles and achieve better learning outcomes.

6.1 Critical Review and Suggestions for Future Research

This study has provided valuable insights into students' perceptions of the Eclipse IDE in educational settings, examining its usability and usefulness. However, there are some limitations and areas for future research that warrant consideration.

First, the survey's sample size and its focus on a single educational institution may limit the generalizability of the findings. Future research could expand the sample to include students from various institutions and diverse backgrounds to provide a more comprehensive understanding of the factors affecting the adoption and use of the Eclipse IDE.

Second, the study primarily relies on self-reported data from students, which may be subject to recall bias or social desirability bias. Future research could incorporate objective measures, such as tracking actual usage patterns or analyzing students' work products, to provide a more accurate assessment of the Eclipse IDE's effectiveness in supporting programming tasks.

Third, the study does not explore the potential impact of alternative IDEs on students' perceptions and experiences. Comparing the Eclipse IDE with other popular IDEs, such as Visual Studio Code or IntelliJ, could offer additional insights into the specific factors that influence students' preferences and decision-making processes when selecting a programming environment.

Future research could also investigate the role of instructional support and teacher engagement in shaping students' experiences with the Eclipse IDE. For example, examining the strategies and resources that instructors use to facilitate learning and support students' engagement with the Eclipse IDE could provide valuable information on best practices and potential areas for improvement.

Additionally, longitudinal studies examining students' perceptions and experiences with the Eclipse IDE over time could offer insights into the changing dynamics of technology adoption and use in educational settings. Such research could help identify potential patterns and trends in students' preferences, as well as the factors that contribute to the persistence or abandonment of the Eclipse IDE.

Finally, future research could explore the relationship between students' perceptions of the Eclipse IDE and their learning outcomes, such as academic performance or skill development. Understanding the impact of the IDE on students' learning and achievement could provide

essential information for educators and developers seeking to optimize the design and implementation of programming tools and resources.

In conclusion, this study offers a valuable starting point for understanding the factors that influence students' perceptions and experiences with the Eclipse IDE in educational settings. However, there are several areas for future research that could further deepen the understanding of the complex interplay between technology, user experience, and learning outcomes in programming education. By exploring these topics, researchers and educators can work together to develop more effective strategies for supporting students' engagement with programming tools and resources, ultimately enhancing their learning experiences and promoting their success in the field.

References

Anderson, T. (2003). Getting the Mix Right Again: An updated and theoretical rationale for interaction. Athabasca University – Canada's Open University. Available online: https://www.irrodl.org/index.php/irrodl/article/view/149/230 [Accessed May 4th 2023]

Anderson, T., and Garrison, D.R. (1998). Learning in a networked world: New roles and responsibilities. In C. Gibson (Ed.), *Distance Learners in Higher Education*. Madison, WI. Atwood Publishing. Available online: https://www.researchgate.net/publication/259874364_Learning_in_a_networked_world_New_roles_and_responsibilities [Accessed May 4th 2023]

Akindunjoye, O. (2018). 5 fundamental elements of interface design. Available online: https://uxdesign.cc/5-fundamental-elements-of-interface-design-7d8b8be47044 [Accessed April 24th 2023]

Albert, B., & Tullis, T. (2008). Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics. Morgan Kaufmann.

Aljafer, H., & Cantrell G. (2020). Learning and Teaching Undergraduate Introductory Programming Courses in Java – The Use of an IDE VS Command Line. *2020 International Conference on Computational Science and Computational Intelligence (CSCI),* Available online: https://american-cse.org/sites/csci2020proc/pdfs/CSCI2020-6SccvdzjqC7bKupZxFmCoA/762400a992/762400a992.pdf [Accessed March 14th 2023]

Altadmri, A., & Brown, N. (2015). 37 million compilations: Investigating novice programming mistakes in large-scale student data. *Proceedings of the 46th ACM Technical Symposium on Computer Science Education - SIGCSE '15*, 522-527.

Andrade, C. (2020). The Limitations of Online Surveys. *National Library of Medicine*. Available online: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7735245/ [Accessed April 24th 2023]

Bower, M. (2019). Technology-mediated learning theory. In The Encyclopedia of Education Research. Available online: https://www.researchgate.net/publication/331866840_Technology-mediated_learning_theory [Accessed April 26th 2023]

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology.* Available online: https://www.researchgate.net/publication/235356393_Using_thematic_analysis_in_psychology [Accessed May 12th 2023]

Brusilovsky, P., Sosnovsky, S., Yudelson, M., Kumar, A., & Hsiao, I. (2010). Learning SQL programming with interactive tools: From integration to personalization. *ACM Transactions on Computing Education*, 9(4), 1-15.

Bryman, A. (2008). Samhällsvetenskapliga metoder. (Upplaga 2:8). Stockholm: Liber AB. ISBN 978-91-47-09068-6.

Bryman, A. (2016). Social research methods. Oxford University Press. Available online: https://handoutset.com/wp-content/uploads/2022/05/Social-Research-Methods-4th-Edition-Alan-Bryman.pdf [Accessed April 20th 2023]

Carrington, D. (2004). Software Engineering Tools and Methods. *Department of Computer Science and Electrical Engineering,* The University of Queensland, Available online: http://www.cs.joensuu.fi/tSoft/dokumentit/Trial_Version1_00_SWEBOK_Guide.pdf#page=172 [Accessed March 21st 2023]

Chen, Z., & Marx, D. (2005) EXPERIENCES WITH ECLIPSE IDE IN PROGRAMMING COURSES. *CCSC: Eastern Conference*, Available online: https://dl.acm.org/doi/pdf/10.5555/1089053.1089068 [Accessed March 21st 2023]

Creswell, J. W. (2009). Research Design: Qualitative, Quantitative, and Mixed Methods Approaches. SAGE Publications. Available online: https://www.ucg.ac.me/skladiste/blog_609332/objava_105202/fajlovi/Creswell.pdf [Accessed May 12th 2023]

Creswell, J. W., & Creswell, J. D. (2017). Research design: Qualitative, quantitative, and mixed methods approaches. Sage Publications. Available online: https://fe.unj.ac.id/wp-content/uploads/2019/08/Research-Design_Qualitative-Quantitative-and-Mixed-Methods-Approaches.pdf [Accessed April 20th 2023]

Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use and User Acceptance of Information Technology. Michigan University. DOI: 10.2307/249008

Desai, S.C., & Reimers, S. (2019). Comparing the use of open and closed questions for Web-based measures of the continued-influence effect. *Behavior Research Methods*. Available online: https://link.springer.com/article/10.3758/s13428-018-1066-z [Accessed April 24th 2023]

Eclipse (2022). Getting to know Eclipse.org, Available online: https://www.eclipse.org/home/welcome/ [Accessed March 21st 2023]

Eclipse (n.d.). About the Eclipse Foundation, Available online: https://www.eclipse.org/org/ [Accessed March 21st 2023]

Etikprövningsmyndigheten, (n.d.). Forskningsperson. Available online: https://etikprovningsmyndigheten.se/for-forskningsperson/ [Accessed March 28th 2023]

Grover, S., Pea, R., & Cooper, S. (2013). Designing for deeper learning in a blended computer science course for middle school students. *Computer Science Education*, 23(2), 324-348. Available online: https://www.academia.edu/32467427/Designing_for_Deeper_Learning_in_a_Blended_Computer_Science_Course_for_Middle_School_Students [Accessed March 23th 2023]

Hartson, R., & Pyla, P. (2012). The UX Book: Process and Guidelines for Ensuring a Quality User Experience. Elsevier. Available online: https://www.sciencedirect.com/book/9780123852410/the-ux-book [Accessed April 26th 2023]

Hassenzahl, M. (2004). The interplay of beauty, goodness, and usability in interactive products. *Human-Computer Interaction*. Available online: https://www.researchgate.net/publication/233864402_The_Interplay_of_Beauty_Goodness_and_Usability_in_Interactive_Products [Accessed March 23th 2023]

Hornbæk, K. (2006). Current practice in measuring usability: Challenges to usability studies and research. *International Journal of Human-Computer Studies*. Available online: https://www.sciencedirect.com/science/article/pii/S1071581905001138 [Accessed March 23th 2023]

ISO 9241-11. (2018). Ergonomics of human-system interaction - Part 11: Usability: Definitions and concepts. International Organization for Standardization. Available Online: https://www.iso.org/standard/63500.html [Accessed April 26th 2023]

Jonnalagadda, S., & Petkovic, D. (2014). An empirical investigation of factors influencing adoption of an e-learning system in the workplace. *Journal of Computer Information Systems*, 54(4), 67-76. Available online: https://www.academia.edu/1837378/An_empirical_investigation_into_factors_influencing_the_adoption_of_an_e_learning_system [Accessed March 23th 2023]

Junyong, I. (2017). Introduction of a pilot study. *National Library of Medicine.* Available online: ncbi.nlm.nih.gov/pmc/articles/PMC5716817/ [Accessed April 24th 2023]

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys*.. Available online: https://dl.acm.org/doi/10.1145/1089733.1089734 [Accessed March 21st 2023]

Kirkwood, A., & Price, L. (2014). Technology-enhanced learning and teaching in higher education: What is 'enhanced' and how do we know? A critical literature review. Learning,

Media and Technology, 39(1). Available online: https://www.researchgate.net/publication/259357668_Technology-enhanced_Learning_and_Teaching_in_Higher_Education_What_is_%27enhanced%27_and_how_do_we_know_A_Critical_Literature_Review [Accessed April 26th 2023]

Kline, R. B., & Seffah, A. (2005) Evaluation of integrated software development environments: Challenges and results from three empirical studies. *International Journal of Human-Computer*. Available online: https://www.sciencedirect.com/science/article/pii/S1071581905001102 [Accessed March 21st 2023]

Krosnick, J. A., & Presser, S. (2010). Question and questionnaire design. Handbook of survey research. Emerald Group Publishing Limited. Available online: https://web.stanford.edu/dept/communication/faculty/krosnick/docs/2009/2009_handbook_krosnick.pdf [Accessed April 25th 2023]

Lewis, J, R. (2019). Comparison of Four TAM Item Formats: Effect of Response Option Labels and Order. *Journal of Usability Studies.* Available online: https://uxpajournal.org/tam-formats-effect-response-labels-order/ [Accessed March 21st 2023]

Marangunić, N., & Granić, A. (2015). Technology acceptance model: a literature review from 1986 to 2013. *Universal Access in the Information Society*. Available online: https://link.springer.com/article/10.1007/s10209-014-0348-1 [Accessed March 21st 2023]

Maslakçi, A., & Sürücü, L. (2020). Validity and Reliability in Quantitative Research. *Business & Management Studies: An International Journal.* Available online: https://www.researchgate.net/publication/344379869_Validity_and_Reliability_in_Quantitative_Research [Accessed March 28th 2023]

Morales, J., Botella, F., Rusu, C., & Quiñones, D. (2019). How "Friendly" Integrated Development Environments Are? In A. Marcus & W. Wang (Eds.), Social Computing and Social Media. Design, Human Behavior and Analytics. *Lecture Notes in Computer Science*. Springer, Available online: https://link.springer.com/chapter/10.1007/978-3-030-21902-4_7 [Accessed March 14th 2023]

Murphy, G. C., Kersten, M., & Findlater, L. (2006). How are Java software developers using the Eclipse IDE? IEEE Software, 23(4). Available online: https://www.researchgate.net/publication/220093228_How_Are_Java_Software_Developers_Using_the_Eclipse_IDE [Accessed April 26th 2023]

Naps, T. L., Rößling, G., Almstrum, V., Dann, W., Fleischer, R., Hundhausen, C., ... & Rodger, S. H. (2002). Exploring the role of visualization and engagement in computer science education.

ACM SIGCSE, Bulletin. Available online: https://dl.acm.org/doi/10.1145/782941.782998 [Accessed March 14th 2023]

Nielsen, J. (1993). Usability Engineering. Morgan Kaufmann. Available online: https://www.sciencedirect.com/book/9780125184069/usability-engineering [Accessed 21st of March 2023]

Palinkas, L. A., Horwitz, S. M., Green, C. A., Wisdom, J. P., Duan, N., & Hoagwood, K. (2015). Purposeful Sampling for Qualitative Data Collection and Analysis in Mixed Method Implementation Research. *Administration and Policy in Mental Health and Mental Health Services Research*. Available online: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4012002/ [Accessed May 12th 2023]

Parnin, C., Görg, C., & Rugaber, S. (2010). Codepad: Interactive spaces for maintaining concentration in programming environments. SoftVis '10: Proceedings of the 5th international symposium on Software visualization. Available online: https://dl.acm.org/doi/10.1145/1879211.1879217 [Accessed April 26th]

Preston, C. C., & Colman, A. M. (2000). Optimal number of response categories in rating scales: reliability, validity, discriminating power, and respondent preferences. *Acta Psychologica.* Available online: https://www.sciencedirect.com/science/article/pii/S0001691899000505 [Accessed April 25th 2023]

Seffah, A., & Rilling, J. (2001). Investigating the relationship between usability and conceptual gaps for human-centric CASE tools. *IEEE Computer Society,* Available online: https://ieeexplore.ieee.org/abstract/document/995263 [Accessed 21st of March 2023]

Selwyn, N. (2016). Is technology good for education?. Polity. Available online: https://research.monash.edu/en/publications/is-technology-good-for-education [Accessed April 26th 2023]

Simms, L. J., Zelazny, K., Williams, T. F., & Bernstein, L. (2019). Does the number of response options matter? Psychometric perspectives using personality questionnaire data. *Psychological Assessment.* Available online: https://www.researchgate.net/publication/331765650_Does_the_Number_of_Response_Options_Matter_Psychometric_Perspectives_Using_Personality_Questionnaire_Data [Accessed April 25th 2023]

Šumak, B., Heričko, M., & Pušnik, M. (2011). A meta-analysis of e-learning technology acceptance: The role of user types and e-learning technology types. *Computers in Human Behavior*. Available online:

https://www.sciencedirect.com/science/article/pii/S0747563211001609 [Accessed March 28th 2023]

Svartdal, F. (2001). Psykologins forskningsmetoder: en introduktion. (1. uppl.). Liber.

Teo, T. (2011). Factors influencing teachers' intention to use technology: Model development and test. *Computers & Education*. Available online: https://www.sciencedirect.com/science/article/pii/S0360131511001370 [Accessed March 28th 2023]

Venkatesh, V., & Bala, H. (2008). Technology acceptance model 3 and a research agenda on interventions. *Decision Sciences*. Available online: https://onlinelibrary.wiley.com/doi/full/10.1111/j.1540-5915.2008.00192.x [Accessed March 28th 2023]

Venkatesh, V., & Davis, F. D. (2000). A theoretical extension of the technology acceptance model: Four longitudinal field studies. *Management Science*. Available online: https://www.researchgate.net/publication/227447282_A_Theoretical_Extension_of_the_Technology_Acceptance_Model_Four_Longitudinal_Field_Studies [Accessed March 28th 2023]

Venkatesh, V., & Morris, M. G. (2000). Why don't men ever stop to ask for directions? Gender, social influence, and their role in technology acceptance and usage behavior. *MIS quarterly*. Available Online: https://www.researchgate.net/publication/220260030_Why_Don%27t_Men_Ever_Stop_to_Ask_for_Directions_Gender_Social_Influence_and_Their_Role_in_Technology_Acceptance_and_Usage_Behavior [Accessed May 12th 2023]

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. *MIS Quarterly*. Available online: https://www.jstor.org/stable/pdf/30036540.pdf [Accessed March 28th 2023]

Vetenskapsrådet (2017). God Forskningsed. Available online: https://www.vr.se/download/18.2412c5311624176023d25b05/1555332112063/God-forskningssed_VR_2017.pdf [Accessed March 28th 2023]

Appendix 1: Survey without responses

Thank you for your interest in participating in our survey. The completion time is less than 5 minutes. *

The primary objective of this study is to identify and evaluate specific usability and usefulness factors affecting students' overall experience with the Eclipse IDE, and to explore potential solutions for addressing these concerns. Our study is conducted as part of Bachelor Thesis at Lund University School of Economics and Management (Department of Informatics).

No personal data will be stored and therefore your answers will be completely anonymous. Your answers will only be used for analytical purposes within the scope of the thesis. Upon publication of this study, all data will be destroyed.

By clicking on "I agree" you confirm that:
You have been informed about the purpose of this study.You have been informed that your answers are anonymous. You have been informed that your answers will be deleted upon publication and until then be kept confidential.You voluntarily agree to participate in this survey.

◯ I agree

Next Clear form

Demographic & Background Questions

What gender do you identify with? *

- ◯ Man
- ◯ Woman
- ◯ Non-binary
- ◯ Other

Do you currently study at Lund University and are enrolled in the Information Systems bachelor programme (Systemvetenskap)? *

- ◯ Yes
- ◯ No

If yes, what year are you in? *

- ◯ Year 1
- ◯ Year 2
- ◯ Year 3

Prior to enrolling in this program, have you used the Eclipse IDE? *

- ◯ Yes
- ◯ No

Prior to enrolling in this program, how much programming experience do you have? *

	1	2	3	4	5	6	7	
No experience at all	◯	◯	◯	◯	◯	◯	◯	A lot of experience

Back Next Clear form

Perceived Usability

Usability refers to the ease of use and the effectiveness of a product or system. It considers how efficiently, effectively, and satisfactorily users can interact with a product to achieve their intended goals.

This section aims to measure the usability of the Eclipse IDE in an educational setting.

How easy was it for you to become familiar with the Eclipse IDE, when you were first introduced to it in year 1? *

	1	2	3	4	5	6	7	
Very Easy	○	○	○	○	○	○	○	Very Difficult

(Non mandatory) Please elaborate on the previous question:

Your answer

How satisfied are you with the interface in general of the Eclipse IDE? The five aspects of interface are: *
- Language
- Color
- Imagery
- Typography
- Icons

	1	2	3	4	5	6	7	
Not satisfied at all	○	○	○	○	○	○	○	Extremely satisfied

Do you believe any interface aspects could be improved upon, in that case which and how?

Your answer

You perceive the Eclipse IDE to be clear and easy to use. *

	1	2	3	4	5	6	7	
Completely Agree	○	○	○	○	○	○	○	Completely Disagree

How easy do you find it to navigate in the interface of the Eclipse IDE? *

	1	2	3	4	5	6	7	
Very Easy	○	○	○	○	○	○	○	Very Difficult

How easy do you find it to locate features such as "Eclipse Marketplace" or "Creating a new Java Project" in the Eclipse IDE? *

	1	2	3	4	5	6	7	
Very Easy	○	○	○	○	○	○	○	Very Difficult

You find it easy to get the Eclipse IDE to perform in the manner you want it to perform. *

	1	2	3	4	5	6	7	
Completely Agree	○	○	○	○	○	○	○	Completely Disagree

Perceived usefulness

Usefulness refers to the degree to which a product or system provides value to its users by enabling them to perform tasks more effectively, efficiently, or with greater satisfaction. It assesses the practical benefits or advantages that users perceive from using a particular product.

This sections aims to measure the usefulness of the Eclipse IDE in an educational setting.

To what extent do you believe using the Eclipse IDE enchances your ability to develop applications? *

	1	2	3	4	5	6	7	
Not at all	○	○	○	○	○	○	○	Significantly enhances

How helpful do you find the features provided by the Eclipse IDE for your programming tasks? Example of features include: *
- Syntax coloring
- Code refactoring
- Debugging

	1	2	3	4	5	6	7	
Not helpful at all	○	○	○	○	○	○	○	Extremely helpful

How well-suited do you believe the features provided by the Eclipse IDE is for your overall success in your programming courses? *

	1	2	3	4	5	6	7	
Not important at all	○	○	○	○	○	○	○	Extremely important

The features provided by the Eclipse IDE is helpful in your programming tasks? *

	1	2	3	4	5	6	7	
Not helpful at all	○	○	○	○	○	○	○	Extremely helpful

You find that the features provided by the Eclipse IDE perform in a manner that is beneficial to your task. *

	1	2	3	4	5	6	7	
Completely disagree	○	○	○	○	○	○	○	Completely agree

Do you believe there are any features missing from the Eclipse IDE?

Your answer

External factors

External factors refers to factors that are external to the user, but have an impact on their perceptions of usefulness and ease of use.

This sections aims to measure the external variables of the Eclipse IDE in an educational setting.

Do you feel that you received adeqaute training or instructions on how to use the Eclipse IDE? *

	1	2	3	4	5	6	7	
Not at all	○	○	○	○	○	○	○	Very well

(Non mandatory) Please elaborate on your previous answer:

Your answer

To what extent do you believe support from your instructors has influenced your ability to use the Eclipse IDE? *

	1	2	3	4	5	6	7	
No influence at all	○	○	○	○	○	○	○	Extremely influential

To what extent do you believe support from your peers has influenced your ability to use the Eclipse IDE? *

	1	2	3	4	5	6	7	
No influence at all	○	○	○	○	○	○	○	Extremely influential

How important is the availability of online resources (tutorials, forums, documentation) in facilitating your use of the Eclipse IDE? *

	1	2	3	4	5	6	7	
Not important at all	○	○	○	○	○	○	○	Extremely important

How easy do you find it to locate online resources to facilitate your use of the Eclipse IDE? *

	1	2	3	4	5	6	7	
Very Easy	○	○	○	○	○	○	○	Very Difficult

Back Next Clear form

(Non mandatory) Do you have any further comments about the Eclipse IDE that you feel is relevant, please elaborate on that here:

Your answer

(Non mandatory) We are always thankful for feedback! If you feel that we in any way could have conducted this study in a different manner, please elaborate on that here:

Your answer

Back Submit Clear form

Appendix 2: Survey with responses

Thank you for your interest in participating in our survey. The completion time is less than 5 minutes.

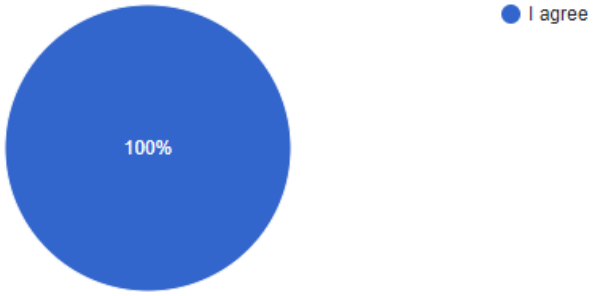
Copy

The primary objective of this study is to identify and evaluate specific usability and usefulness factors affecting students' overall experience with the Eclipse IDE, and to explore potential solutions for addressing these concerns. Our study is conducted as part of Bachelor Thesis at Lund University School of Economics and Management (Department of Informatics).

No personal data will be stored and therefore your answers will be completely anonymous. Your answers will only be used for analytical purposes within the scope of the thesis. Upon publication of this study, all data will be destroyed.

By clicking on "I agree" you confirm that:
You have been informed about the purpose of this study.You have been informed that your answers are anonymous. You have been informed that your answers will be deleted upon publication and until then be kept confidential.You voluntarily agree to participate in this survey.

75 responses

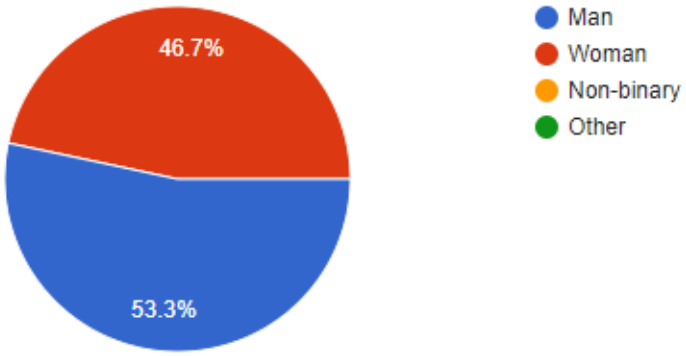


Demographic & Background Questions

What gender do you identify with?

Copy

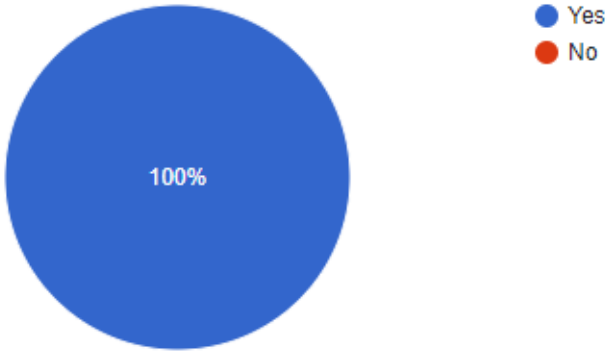
75 responses



Do you currently study at Lund University and are enrolled in the Information Systems bachelor programme (Systemvetenskap)?

Copy

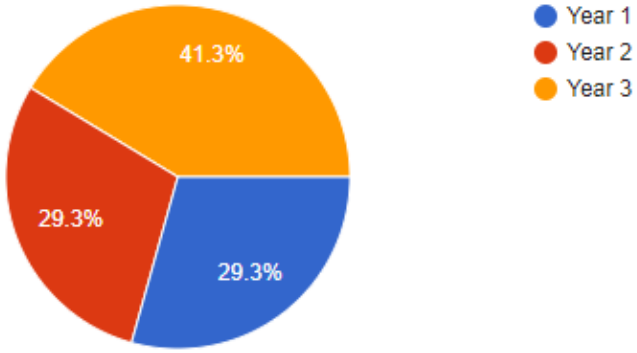
75 responses



If yes, what year are you in?

Copy

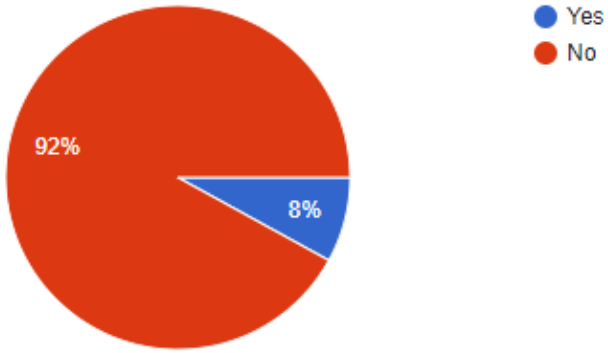
75 responses



Prior to enrolling in this program, have you used the Eclipse IDE?

Copy

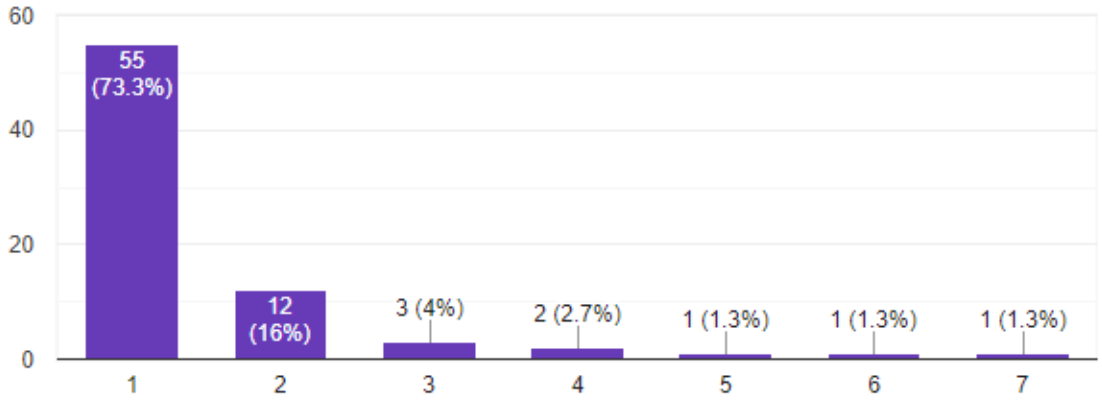
75 responses



Prior to enrolling in this program, how much programming experience do you have?

Copy

75 responses

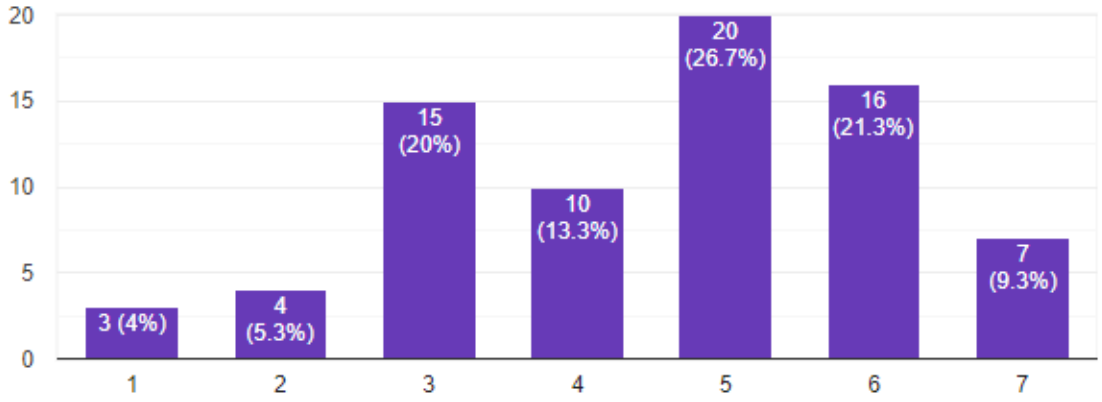


Perceived Usability

How easy was it for you to become familiar with the Eclipse IDE, when you were first introduced to it in year 1?

Copy

75 responses



(Non mandatory) Please elaborate on the previous question:

It was hard at first because I lacked programming knowledge.

Hard at first but with good instructions it wasn't that hard

Vi fick ingen introduktion i hur Eclipse fungerar vilket gjorde det svårt att använda programmet. Det förutsattes väldigt ofta att man redan visste hur man skulle göra. I och med det grafiska gränssnittet som inte är på samma nivå som nyutvecklade program försvårades introduktionen till Eclipse

Doing basic programming like we did in the first year was easy to do in Eclipse but beyond that I would have a hard time navigating the interface.

I just followed a guide for the set up

The use case in year 1 was very narrow and specific, and there were step-by-step guides for everything. However, the program itself is not very intuitive or easy to learn.

not a modern IDE

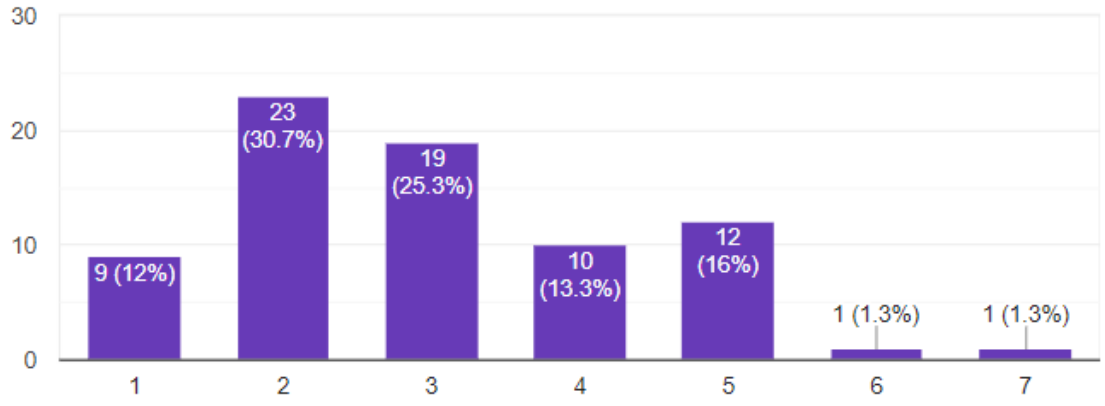
I found it hard to understand alot of the features

Many different funtions and hard to understand the GUI

How satisfied are you with the interface in general of the Eclipse IDE? The five aspects of interface are:
- Language
- Color
- Imagery
- Typography
- Icons

Copy

75 responses



Do you believe any interface aspects could be improved upon, in that case which and how?

Icons are messy and unclear

In general it looks old and outdated

Har använt eclipse väldigt lite och det var länge sen så minns inte specifika saker

Simplicity. The interface is too cluttered.

It should be in darkmode by default in my opinion. Also certain logos make it hard to see what type of project or such some things are. Ejb projects are quite similair to regular java projects for example

GUI is ugly and looks like it was designed in the 90s (which, to be fair, it probably was.)

The GUI is unintuitive and appears "messy" to the uninitiated. Larger and fewer icons, in expandable lists and menus would probably make it easier to understand.

The whole interface needs an update to current UX standards in my opinion. It feels very ”windows xp” in the design.

I don’t have any references but it felt old and not intuitive

I have used other IDEs before Eclipse, just feel they are better

-

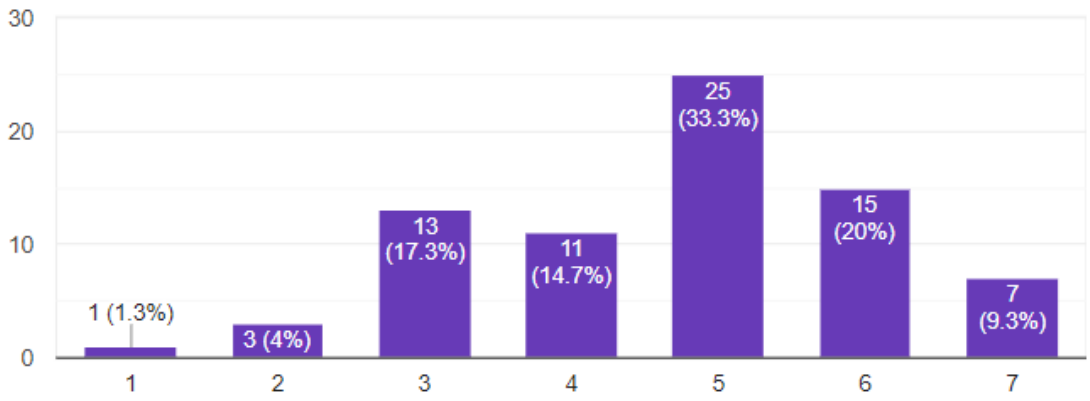
Havent used in a while, dont remember

Modify everything. Make it easier. Make it more modern.

You perceive the Eclipse IDE to be clear and easy to use.

Copy

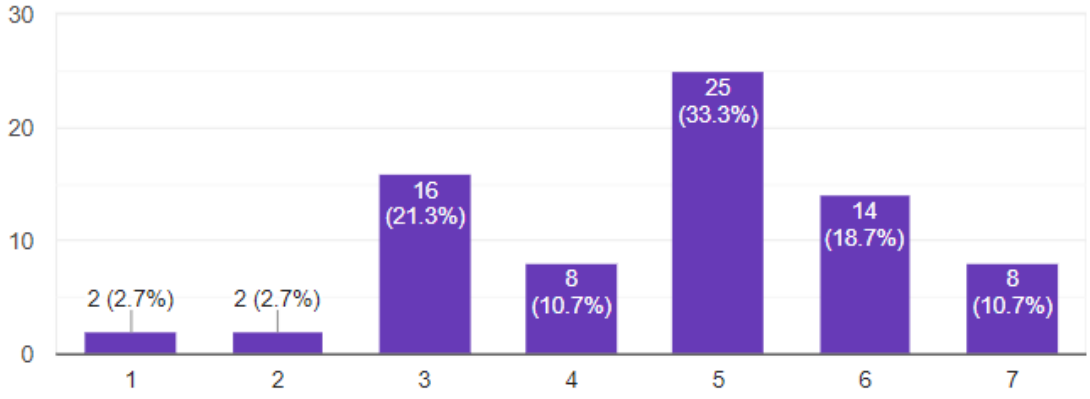
75 responses



How easy do you find it to navigate in the interface of the Eclipse IDE?

Copy

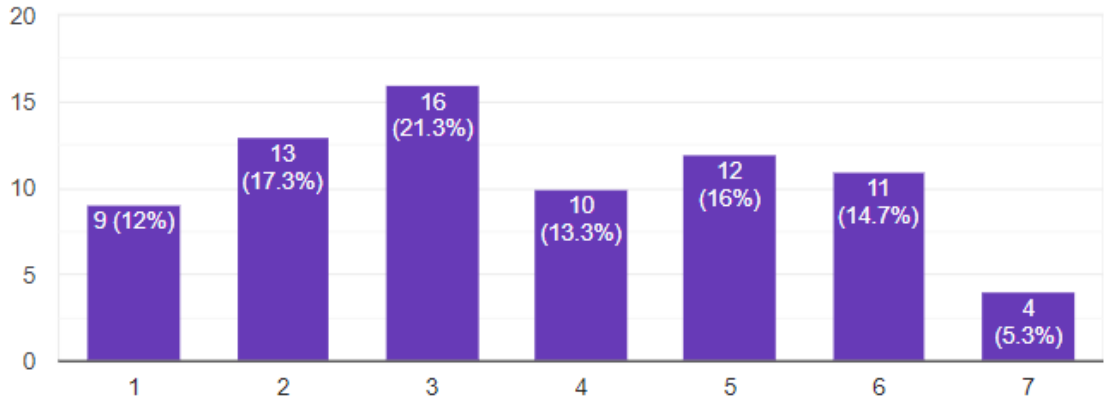
75 responses



How easy do you find it to locate features such as "Eclipse Marketplace" or "Creating a new Java Project" in the Eclipse IDE?

Copy

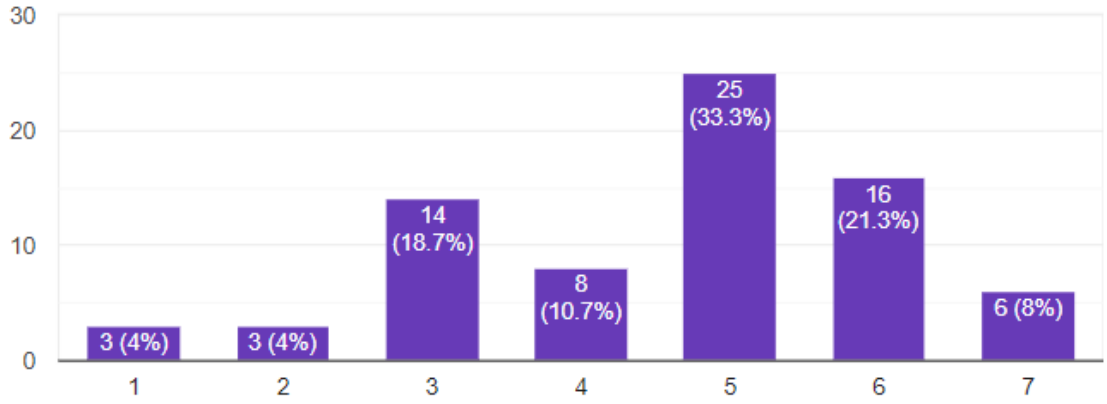
75 responses



You find it easy to get the Eclipse IDE to perform in the manner you want it to perform.

Copy

75 responses

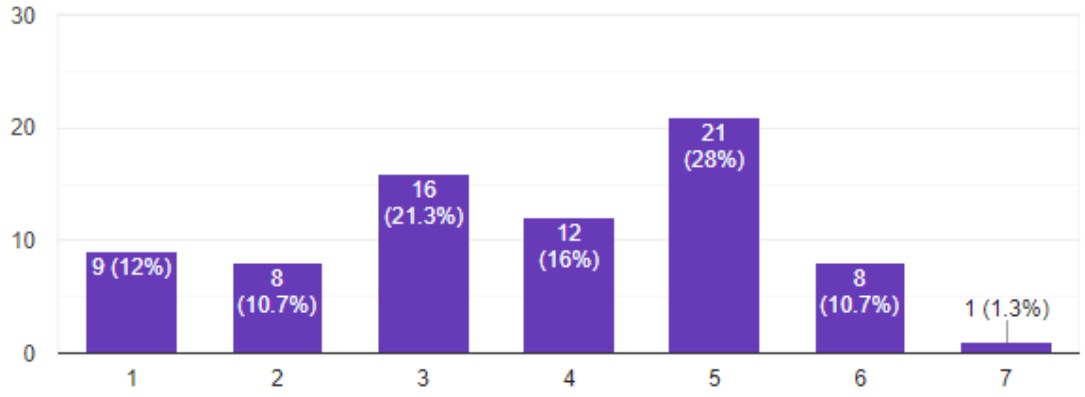


Perceived usefulness

To what extent do you believe using the Eclipse IDE enchances your ability to develop applications?

Copy

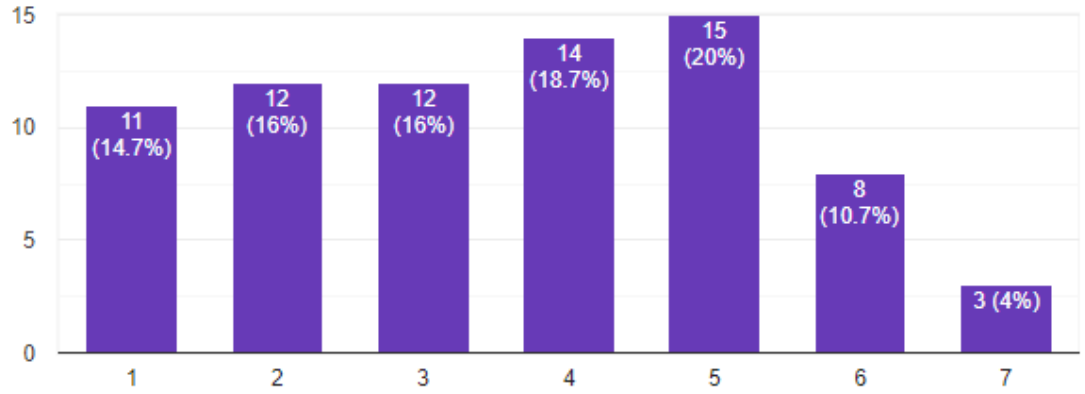
75 responses



How helpful do you find the features provided by the Eclipse IDE for your programming tasks? Example of features include:
- Syntax coloring
- Code refactoring
- Debugging

Copy

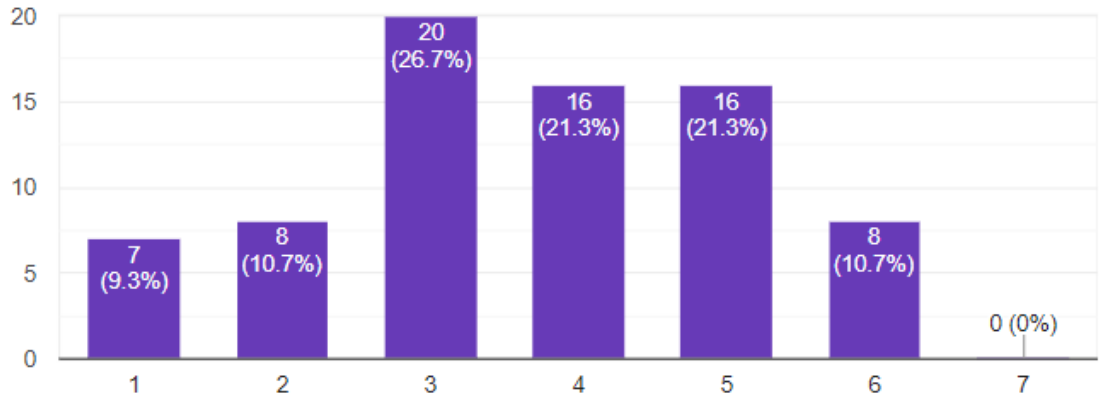
75 responses



How well-suited do you believe the features provided by the Eclipse IDE is for your overall success in your programming courses?

Copy

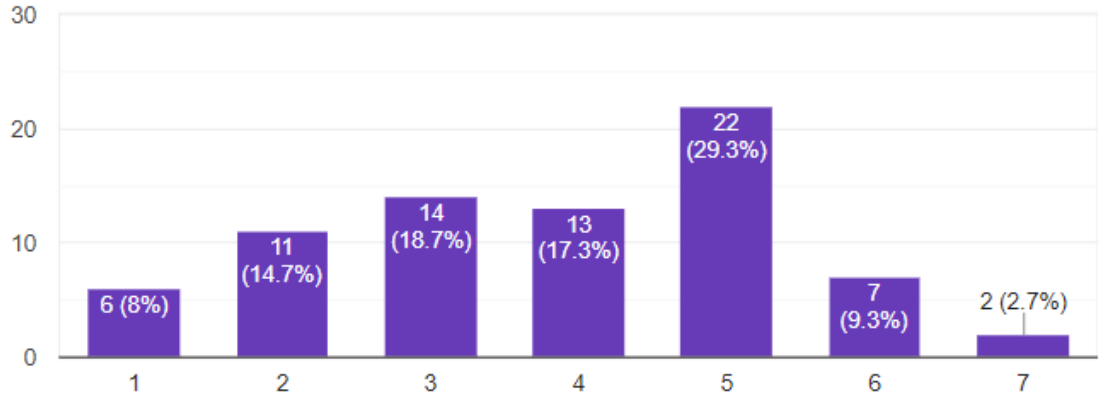
75 responses

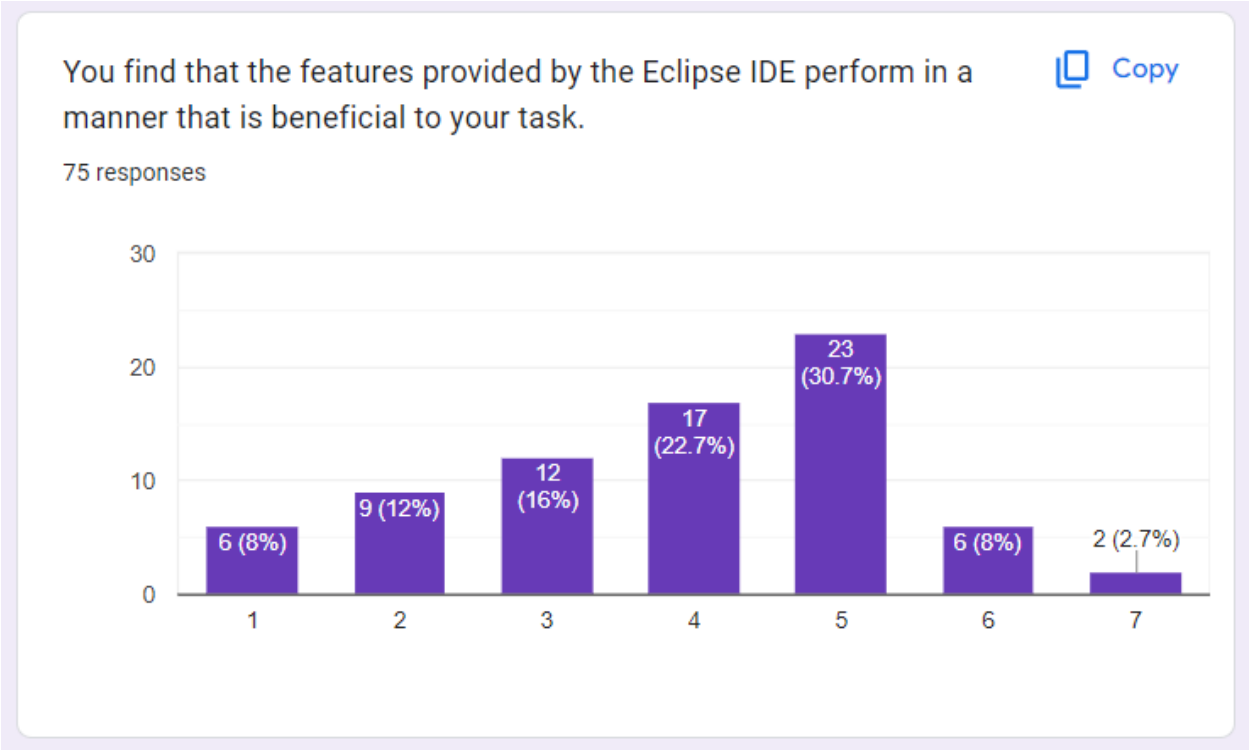


The features provided by the Eclipse IDE is helpful in your programming tasks?

Copy

75 responses





Do you believe there are any features missing from the Eclipse IDE?

Github copilot

A feature as in visual studio that anticipates what you are going to write and helps you fill in code

For sure but can’t name them right now

Många IDE´s har ju kodförslag, minns ej namnet på det men när din IDE ger förslag på optimeringar i koden eller avsluta stycken baserat på vad du använt innan

Not that I know of

code autocompletet

No

alot of features such as github copilot

-

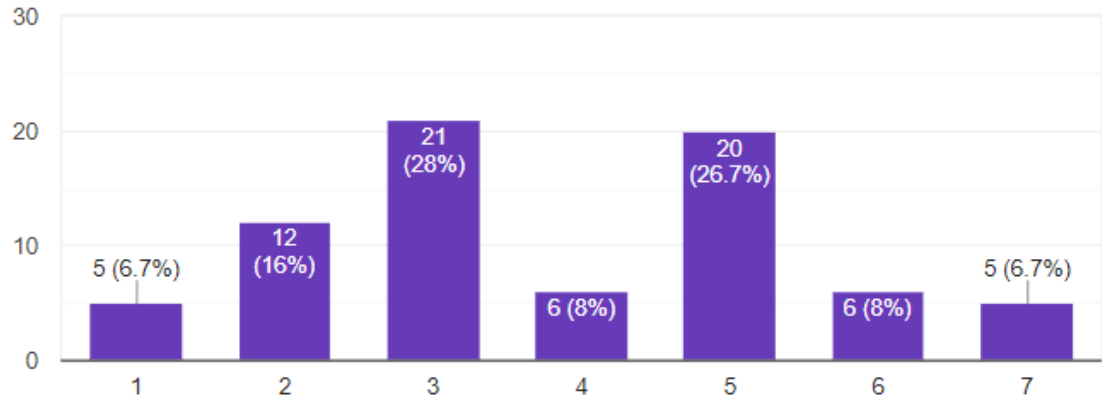
Havent ued in a while, dont remember

External factors

Do you feel that you received adeqaute training or instructions on how to use the Eclipse IDE?

Copy

75 responses



(Non mandatory) Please elaborate on your previous answer:

-

The guides during our year was to difficult to understand and resulted in errors even though you followed the guide step by step, which made it very difficult to use the program since it was the first time.

Björn have exstensive video tutorials for every question you have, so when you get stuck you can just go back and watch.

Det är väl lite eget ansvar men generellt sett nej

A more guide on how to use it would be more efficent

You only really learn how to create new projects/classes, how to use the debugger and where to find the marketplace, not much more

I think that as far as it is possible, we have received adequate training. However, because of the poor UX of the tool, the training is not quite sufficient to get all the way there.

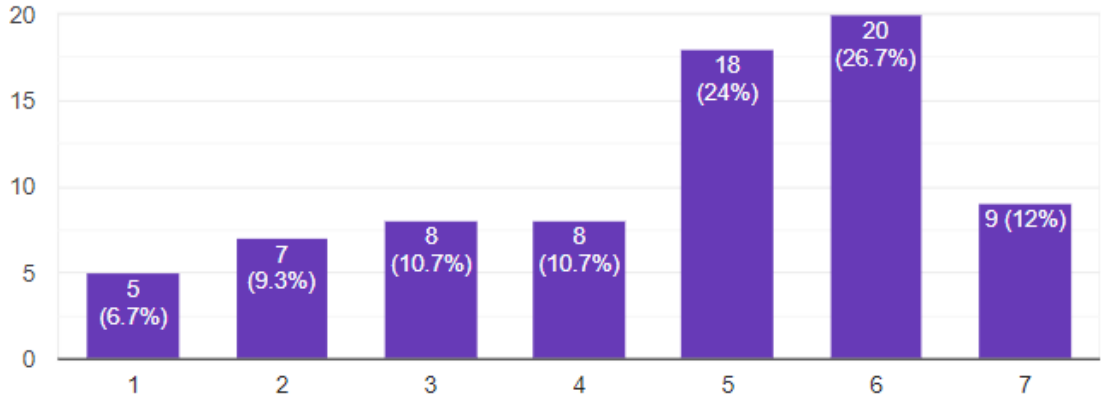
No clear instructions

Björn had a lot of videos which helped

To what extent do you believe support from your instructors has influenced your ability to use the Eclipse IDE?

Copy

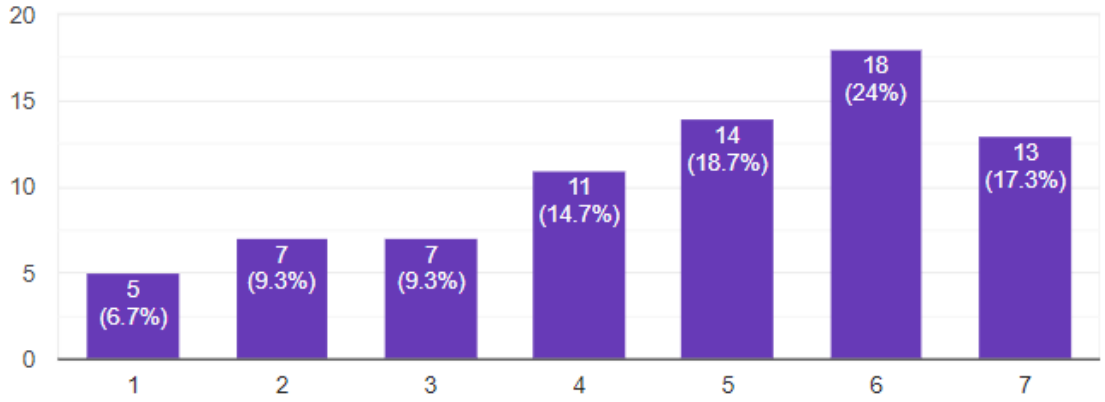
75 responses



To what extent do you believe support from your peers has influenced your ability to use the Eclipse IDE?

Copy

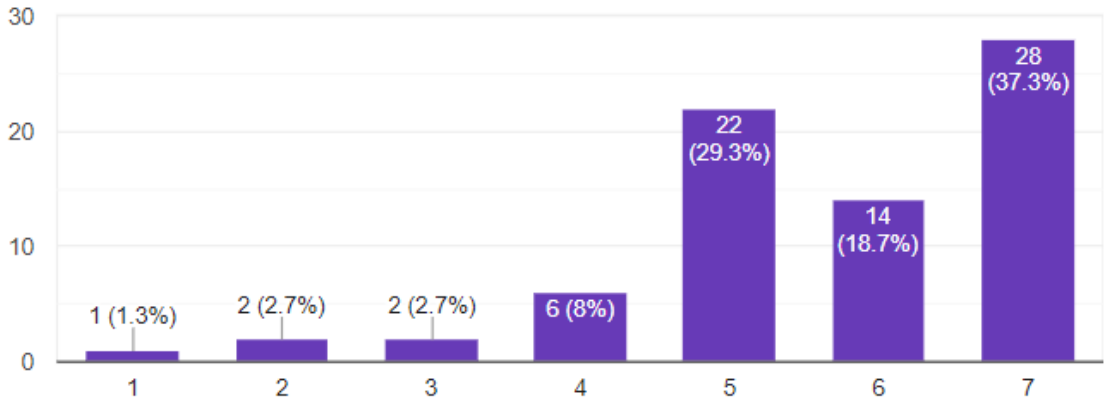
75 responses



How important is the availability of online resources (tutorials, forums, documentation) in facilitating your use of the Eclipse IDE?

Copy

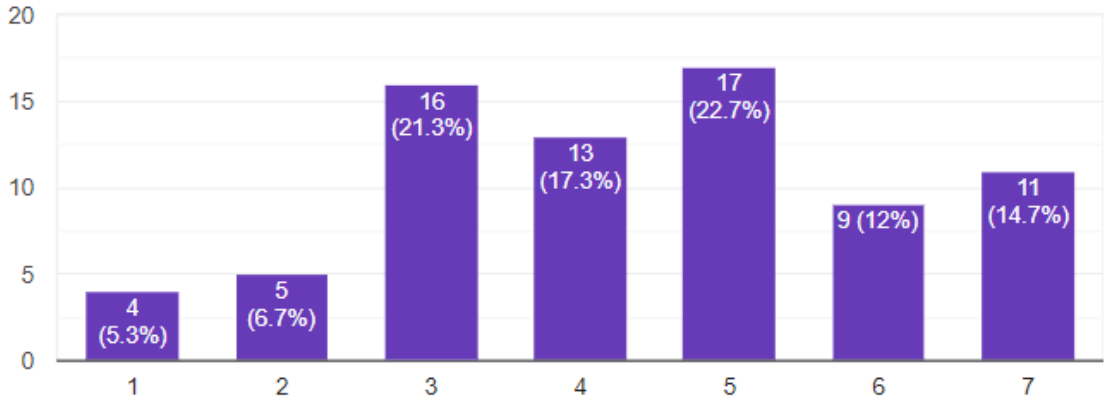
75 responses



How easy do you find it to locate online resources to facilitate your use of the Eclipse IDE?

Copy

75 responses



(Non mandatory) Do you have any further comments about the Eclipse IDE that you feel is relevant, please elaborate on that here:

It serves its purpose as a tool for students to learn how to code

VS Code seems like a better alternative

In comparison to the alternatives on the market, and for the purposes of general Java programming, Eclipse is suboptimal. There are far better options available.

However, it does have ample support for the secondary Java-technologies taught in the program, such as EJB, and as such is quite suitable. But I think that if parts of the tech stack were reviewed and possibly changed, a better IDE could be used, and I do think that this would improve our education.

Compared to other IDEs it does not feel as powerful

-

Havent used in a while, dont remember

(Non mandatory) We are always thankful for feedback! If you feel that we in any way could have conducted this study in a different manner, please elaborate on that here:

Vad är peers? Bra att ni förtydligar vad det är ni frågor om i början av varje sida. Känns som i usefulness sidan ställer ni väldigt liknande frågor. Annars är det väldigt lättläst engelska och förståligt vad ni frågar om!

fråga 1 på perceived usability, potentiellt en icke-obligatorisk fråga om varför efter? exempelvis tyckte jag att det var svårt för inget funkade ju innan man la in en specifik textsnutt på ett specifikt ställe och därför satt man typ 4 timmar med labbhandledare för att försöka lösa det vilket gjorde det very difficult, alternativt att det inte är info ni är intresserade av att samla in. Nu i efterhand så kom ju den frågan på external factors, det var ju i samband med guides dock som är varför jag ansåg det väldigt svårt men kanske att någon annan har ett annat svar? Potentiellt upprepande annars? hmm svårt, gå på magkänsla

fråga 4 & 7 på perceived usability, kanske byta sida på agree och disagree då det på de andra frågorna är lätt på vänstersidan, vilket jag tycker är med åt det positiva hållet än svårt och agree är mer positivt än disagree!

Some of the questions were pretty similar to each other.

-