

Usability Evaluation of Visual Studio Programming Tool

1. **Iqra Ashraf** Department of Computer Science, Bahria University, Lahore, 54000, Pakistan
2. **Abdul Hafeez Muhammad** (Corresponding Author)
Department of Computer Science, Bahria University, Lahore, 54000, Pakistan

Abstract



In computer science, programming skill is considered as an expected result of a student's education. It has been investigated by heuristic testing that how much the programming ability students have, as they complete their first one or two computer science (CS) courses. In order to explore possibilities for assessing CS students, a trial assessment is arranged to evaluate whether students can program. The main goal of this research was to initiate dialog in the CS community on how to done these type of assessments and improve the usability of such kind of programming framework. Assessment is done in university with the help of heuristic testing in order to explore the usability problems of programming tool Visual Studio with the help of qualitative evaluation that involves Nielson heuristics to improve the usability of the interface. Visual studio experts evaluated the usability of it by 10 Nielson heuristics. It shows that the major issue arises are of interface students of first semester face problems as they were novice and layman users having little bit know how of the tool. Difficulty they face were in commenting the code, small buttons and not proper labels mentioned. Proper interface is recommended by the experts that ease in programming.

Keywords: Effectiveness, Efficiency, Satisfaction, Usability, Heuristic Evaluation, Experts, Coding, Visual Studio

Introduction

The reviewed literature defines that usability is not a single aspect to evaluate interface. Different usability attributes should be considered to measure effective interface. For the acceptance of a product, **Shackel [6]** define four-dimensional attribute: which are **attitude**, **effectiveness** of product, how easily product can be **learned**, how **flexible** is the system that it has internal locus of control for users. Based on 'system acceptability model' [10] defines some other usability attributes/characteristics that are **utility and usability [7]** that are categorized as sub-part of 'usefulness'. Nielsen highlighted five major attributes related to usability that are:

1. **Easy to learn(learnability):** System should take less time to learn for first time.
2. **Efficient to use (efficiency):** How efficiently and frequently a task is performed with accuracy.
3. **Easy to remember (memorability):** after learning the system how much time user takes time memorizes it after time.
4. **Preventing errors:** system should be less error prone and aware users so the user can make fewer mistakes.
5. **Subjectively/Aesthetically pleasing:** it shows how user feels about the system [8].

Nielsen[10] said that usability provides ease of use that shows design principle. **Andreas[9]** defines the system is acceptable when user can perform set of tasks easily and effectively. Both definitions explain that it is essential to provide user such an interface that user can easily interact with. Usability core components such as: **navigation, utility of system, interaction with system, appearance of interface, efficiency and satisfaction** of user with system [17], actually explains how easily users can surf website by rating it and leave website with satisfaction level achieved by them [1].

A. IMPORTANCE OF USABILITY:

The ISO standard suggests that in order to measure usability website should have [5]:

- **Effectiveness:** How user perform the target goal effectively?
- **Efficiency:** How fast is the system, how much time it required?

- **Satisfaction:** Does it satisfies the user need by providing authentic and proper feedback?

B. VISUAL STUDIO:

Computer science students are expected to masters in programming skill and knowledge. Mostly science, mathematics, engineering, and technology (SMET) field programs consider that their students will acquire skills of programming as primary part of their education. The question that needs to keep in mind is that whether these education requirements are being followed. Are the appropriate assessment is being conducted to measure if the students have learned the core programming skills?[2]

Except the theoretical study of computer science subjects, programming helps to learn practically with the help of code [11]. First year students are given the learning environment with the help of the lab arranged for the CS courses that help the students to build, design and develop applications easily and help in their future [14].

Different programming language are there to build a program like C, C++, JAVA, COLBOL, ASP .NET and many others that help the students to interact innovatively.

To develop a program there should be developing environment present that is termed as integrated development environment (IDE), different tools are used to create program, debug error and run the program [12]. Mostly used tool for the development of the programs is Visual Studio that helps students to code easily with the help of some built-in header files and function keywords [15]. Different programming skills like object oriented programming, pointers, functions, stacks, arrays, formation of files all are created and learned with the help of Visual Studio [13].

This research basically measure usability of Visual Studio tool to improve programming practice by the help of effectiveness, efficiency and satisfaction metrics.

I. METHODS AND MATERIALS

The main purpose and core part of this study is usability evaluation of visual studio tool. Study experiment was done at Bahria University Lahore campus which is a Federally Chartered Public Sector University. The participants were the CS Department University students of Bachelors of Information Technology of first, third and fourth semester. Usability problems are identified which are discovered by users and experts by usability testing with the help of effectiveness and efficiency.

ISO three metrics to measure usability is used in this research to measure the usability of Visual Studio tool that helps to identify the severity of the errors(minor, intermediate, major) due to which interface can be design better by the help of recommendations.

A. USER TYPE:

Participants of this research are university students and lecturers that can be further categorized as: **1. Novice user** (new user that use the visual studio for the first time). First year students of IT are mostly the novice users as few of them are of different backgrounds and have less know how of computer knowledge and do not practice their programming skills initially [16]. **2. Intermediate user** (that are not an expert but use the visual studio and have knowhow of it). Intermediate students can be first year students that have prior knowledge of CS and third, fourth semester students that have knowledge of programming, they know how to code but not are expert in it. **3. Expert user** (that are frequent user and are expert of the tool visual studio). Experts users code, design and develop code in an efficient manner. Experts are mainly the students of third year and fourth year having prior programming knowledge and can understand the statement and implement it on Visual Studio.

B. TESTING TECHNIQUE:

Testing techniques used in research are:

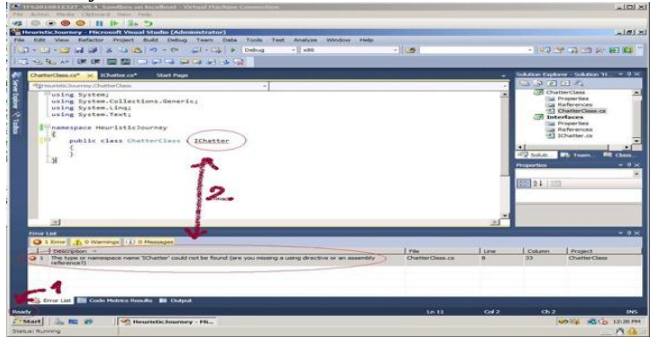
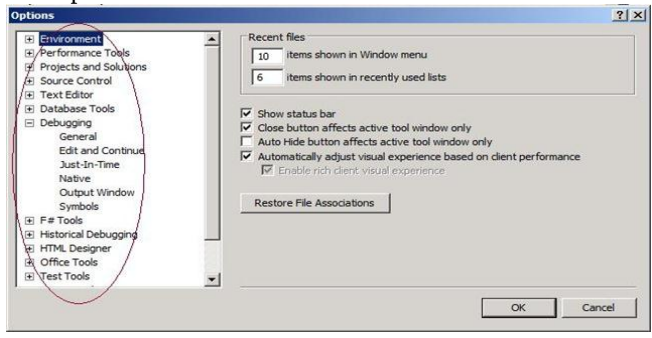
1. HEURISTIC TESTING

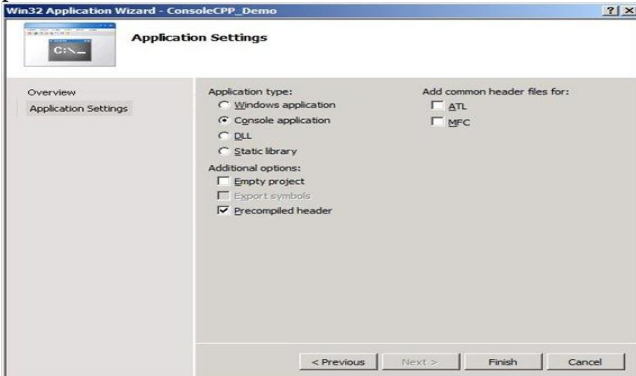
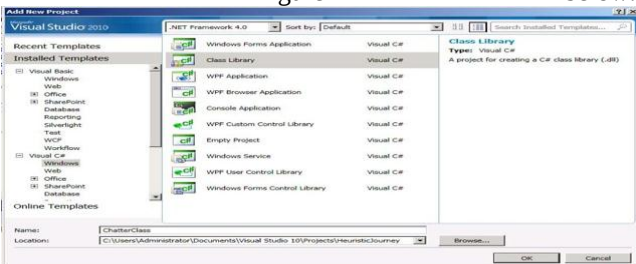
In this research, heuristic testing is performed in which expert's presence is necessary. Expert of Visual Studio are asked for which problems they face while using this tool and give their opinions about how to improve interface usability according to their experiences [6].

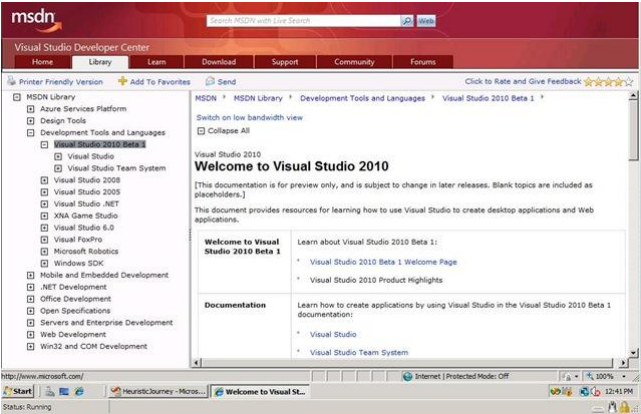
HEURISTIC TESTING EVALUATION AND RESULTS:

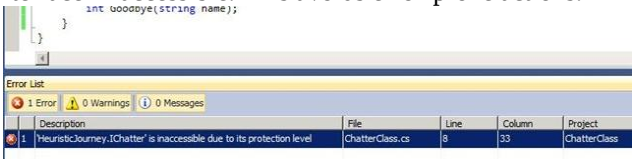
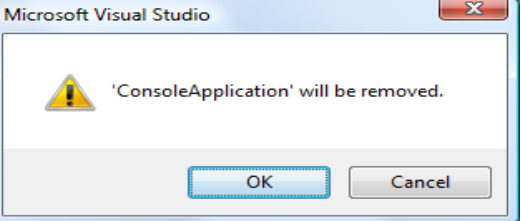
Heuristic evaluation principles are defined in the below Table. 1. Evaluation is done by the experts and errors are founded as minor, cosmetic and major errors. Further evaluation notes and suggestions are given by the experts to improve the interface.

TABLE I
Heuristic Evaluation

Heuristic evaluation Principles	Visual Studio Tool(Error found)	Evaluation Notes	Improvement Suggestions
1.Visibility (how much system status is clear)	2 minor errors, 0 cosmetic and 0 major problems that show total visible errors are 2%. Visibility issues are basically of Visual Studio icons like comment icon for commenting the program.	With the help of feedback provided by the status bar (1) Visual Studio helps the user updated, the need of use regarding to specialized windows, like the the Error List (2) and code highlighted, such as to prominent the error in the code wavy underline is used. That shows proper visibility. 	
2.Match the interface between system and real life	0 minor errors, 8 cosmetic and 0 major problems that show total errors are 8% that can be improved with time and by adding metaphors and analogies to merge interface with real world.	Visual studio uses words, phrases, concepts and descriptions that are familiar to user.  At first the glance options provided appears confusing to the user, but the ordering done logically of the options and further sub-options helps user to rapidly adapt and understand	<i>With keeping the previously searched history and changes performed helps the user to recognize easily rather than recall.</i>
3.User internal locus of control and freedom to navigate.	1 minor error, 3 cosmetic and 0 major problems that show total visible errors are 4%. Visual Studio allow users to navigate easily and switch between labels besides small changes that can be also ignored.	User can easily do undo and redo, that also includes global impact changes, for example code refactoring.  If accidentally Visual Basic program is created as the project creation is set to VB by default, instead of C#. To undo this option includes un-doing manually and deletion of files, which seems to be time consuming and unclear to novice user.	<i>It should also support the undoing of creation of program action, including deleting all the created artifacts.</i>

4.Consistency Same flow) and Standards (rules)	3 minor errors, 2 cosmetic and 0 major problems that show total consistency errors are 5% that shows flow is proper of Visual Studio little standards are being to be followed.	User interface, look and feel and feature set are consistent with the older versions of the Visual Studio. The additional features that are blend into the Visual Studio environment can be identified by the user by exploration. In other words if you are familiar with the environment of 2008 you will be good to go with 2010, as the interface is consistent and as the use of terminology, wizards , menu and actions is same/consistent with previous versions and are platform compatible.. Some settings of the project (2008 defaulted to AnyCPU, 2010 defaults to X86) have been amended that may cause problem to users.	<i>Do not enforce users to define the project start-up defaults just allow them.</i>
5.Recognition (visualization) rather than recall(remember ing the interface)	3 minor errors, 0 cosmetic and 0 major problems that show total visual errors are 3% that gives us concept that Visual Studio uses more visualization rather than text so the user can easily understand.	The tool promotes the concept of information recognizing and remembering by less burden to mind by number of ways, such as simple and properly labeled dialogs, feature to step back/forward easily in wizards, and help is provided that is context sensitive.  The new float features, also dock navigation feature, makes windows undocking and docking a breeze.	
6.Flexibility of using the interface and efficiency	1 minor errors, 1 cosmetic and 0 major problem that shows total efficiency errors are 2% that gives us concept that Visual Studio is efficient and fast to use.	The solution provides for the new, power, mouse- biased and keyboard-shortcut user, with the use of other productivity features for example code snippets as shown in figure below:  Both novice and expert users can therefore cater their integrated development environment to uniform their styles and to tailor rapid actions. Toolbar is clear and consistent as regards to other releases of Visual Studio. The question that needs to keep in mind is whether the “ Ribbon ” concept adopted in many of most familiar Windows applications, such as Office.	<i>Give user choice, as ribbons basically clutter down the working area of the Visual Studio intended users, consistency would be introduced for the users that are working Visual Studio and Office.</i>
7.Minimal design and attractive design	2 minor errors, 1 cosmetic and 0 major problem that shows total attraction design errors are 3% that	Home page that is new start up page is aesthetically pleasing. No irrelevant information is provided; information is not cluttered as its predecessors and has spontaneous images and labels for activities, such as feedback, etc.	<i>Allow the user to effortlessly customize the start-home page.</i>

	gives us concept that Visual Studio design is attractive simple and professional.	 The rest of the tool was consistent in terms of it shows only the relevant information by hiding the advanced information but access is provided to “Advanced” links.	
8.Help users to easily identify, diagnose errors and recover from errors	2 minor errors, 0 cosmetic and 0 major problem that shows total errors are 2% that gives us concept that Visual Studio can easily recover from errors.	Error messages are defined both in the form of plain message and in the form of Windows specific code, as well as giving reference to where in the program code the error was identified and debugged. The precise indication denoting the errors, as well as constructive suggestions provided are huge productivity features.	
9. Proper help and documentation (guidelines) provided.	4 minor errors, 4 cosmetic and 0 major problem that shows total errors are 8% that gives us concept that Visual Studio provided guidelines and online assistance but little difficult for new user to find help and documentation.	The context sensitive help , offline and online helpwith Visual Studio is one of the best feature, that offers the easy indexing, searching, additional samples and videos. Some of the versions only provide the online help like BETA-1, offline help will be reintroduced in the next release of BETA. 	

10.Preventing users to make less errors	7 minor errors, 2 cosmetic and 0 major problem that shows total errors are 9% that gives us concept that Visual Studio provided easy reversal of action when user mistakenly perform any action.	Visual Studio have feature that performs validity checking while the user edits code. For example when a user purposefully from the interface declaration deleted the public access modifier keyword, the system shows a warning that by performing this action would make interface in-accessible. This avoids error-prone actions.  Before deleting action is committed Visual Studio pop up the confirmation message. 	<i>The tool does not warn the user when defining a private interface, until it is referenced in the program somewhere.</i>
---	---	---	---

DISCUSSION AND RESULTS:

Usability testing held in university where university IT students were observed while performing their tasks and time in seconds was noted to perform the task. Five basic tasks were performed by the users on the best programming tool visual studio: 1) Create new project, 2) comment the program, 3) run the program, 4) save the project, 5) open project. There is a comparison made of the students of first semester and third/fourth semester students of IT. As shown in the table effectiveness, efficiency and satisfaction(usability metrics) are all satisfactory of IT 3,4 semester students as it is above 70% and IT-1 students have below the threshold value effectiveness, efficiency and satisfaction that needs to be improved. As IT-1 students are basically novice and intermediate users of visual studio and are learning to do programming as there starting point most of them are from different fields that they are not expert in it.[5] So the interface learning is bit difficult to them. Major time taken to complete the task of commenting, as many students don't know that commenting of the program can also be done by the comment button the go by line by line commenting so they face frustration. Its interface needs to be improved according to the errors catered by the interface minor, intermediate or major. Major problems should be tackled immediately.

CONCLUSION:

It is concluded that first year students don't have programming knowledge or are not expert in it so they face difficulty in performing the task most of the students face difficulty in commenting the program and saving the program. Due to little bit tricky look of the interface comment button, run button is located on the top of the tool but they are so small that it needs to be improvement by proper labelling. Third and fourth semester students have known how of the programming knowledge and they are expert in it due to attention provided to the memory for working with this tool, they are frequent users and their efficiency, effectiveness and satisfaction level is above the threshold(satisfactory level). Qualitative evaluation termed as heuristic evaluation is performed in this research where experts evaluate the Visual studio tool and define the usability interface improvements. This research provides a way how to assess the tool as programming skill is the basic knowledge of the CS student and further usability of IDE Visual Studio is evaluated that needs to be further improved for novice users as they take much time and interface is bit tricky for them.

REFERENCES

- Al-Badi AH, Michelle OO, Al Roobaea R, Mayhew P. Improving usability of social networking systems: a case study of LinkedIn. Journal of Internet Social Networking & Virtual Communities. 2013;1.
- B. Y. B. A. Myers and J. Stylos, "Improving API Usability."
- B. Stroustrup, "An Overview of the C ++ Programming Language," pp. 1–23, 1999.
- Issa T, Turk A. Applying usability and HCI principles in developing marketing websites. International Journal of Computer Information Systems and Industrial Management Applications. 2012;4:76-82.

- J. Brooke, "SUS - A quick and dirty usability scale."
- Joshi A, Sarda N, Tripathi S. Measuring effectiveness of HCI integration in software development processes. *Journal of systems and software*. 2010;83(11):2045-58.
- Lee Y, Kozar KA. Understanding of website usability: Specifying and measuring constructs and their relationships. *Decision support systems*. 2012;52(2):450-63.
- M. Mccracken *et al.*, "Report by the ITiCSE 2001 Working Group on Assessment of Programming Skills of First-year CS Students," 2001.
- Mayer, R. E. (1981). A psychology of how novices learn computer programming. *Computing Surveys*,1, pp. 121-141.
- Pranata I, Skinner G, Athauda R, editors. A survey on the usability and effectiveness of web-based trust rating systems. 2013 IEEE/ACIS 12th International Conference on Computer and Information Science (ICIS); 2013: IEEE.
- Pea, R. (1986). Language independent conceptual bugs in novice programming. *Educational Computing Research*, 2(1), pp. 25-36.
- Soloway, E., Ehrlich, K., Bonar, J., & Greenspan, J. (1982). What do novices know about programming? In A. Badre and B. Shneiderman (Eds) *Directions in Human-Computer Interactions*, Norwood, NJ: Ablex, pp. 27-54.
- Spohrer, J., & Soloway, E. (1986). Novice mistakes: Are the folk wisdoms correct? *Communications of the ACM*, 29(7), pp. 624-632.
- T. Asq, "Part 1 . After Scenario Questionnaire (ASQ)," pp. 1–8, 2011.
- Wiedenbeck, S., Ramalingam, V., Sarasamma, S. and Corritore, C.L. (1999). A comparison of the comprehension of object-oriented and procedural programs by novice programmers. *Interacting With Computers*. 11(3), March, pp. 255-282.
- Wilson, B. C., & Shrock, S. (2001). Contributing to success in an introductory computer science course: A study of twelve factors. In I. Russell (Ed.), *The Proceedings of the Thirty-second SIGCSE Technical Symposium on Computer Science Education*. In SIGCSE Bulletin inroads. 33(1). pp. 184-188.