Modern Integrated Software Development Environments

Svitlana Sotnik[1], Vyacheslav Lyashenko[2], Tetyana Schakurova[2]

[1]Department of Computer-Integrated Technologies, Automation and Mechatronics, Kharkiv National University of RadioElectronics, Ukraine

[2]Department of Media Systems and Technology, Kharkiv National University of Radio Electronics, Ukraine
e-mail: lyashenko.vyacheslav@gmail.com

Abstract: The article provides overview of three most popular modern integrated development environments, during which features of each environment, advantages and disadvantages are highlighted. An overview of today's most popular IDEs became prerequisite for identifying key factors to consider when choosing not only reviewed IDEs, but also similar environments.

Keywords—peculiarities; IDE; Visual Studio; Eclipse: NetBeans.

1. INTRODUCTION

An important warehouse in process of developing software security is vibe of virtual IDE (Integrated Development Environment), so it's not deprived of only platforms, but also meaning of special preparation [1]-[3].

IDE is multifunctional program that can be used for various aspects of software development.

Integrated Development Environments simplify process of composing programs, because writing of code, compiling and running programs are performed in one program – IDE.

One of key differences between IDEs and text editors is ability to see result of their work.

It's possible to use IDE for development of software (SW).

Integration of environments development in order to maximize productivity of programmer, programmer has tightly connected components with simple brownish interfaces.

This allows developer to take fewer steps to switch between different modes, as opposed to discrete development programs.

IDE is complex software package.

The release of new IDEs has become trend in world of programming. Each of them has its own characteristics.

There are number of factors to consider when choosing development environment, so topic will always be relevant.

2. RELATED WORK

Each year, authors disclose ratings of modern IDEs.

Due to growth of new applications running on different platforms, large number of works is devoted to integrated development environments.

So, for example, work [4] is devoted to development environment "IDE" for Arduino, namely, general points: how to install it and prepare it for development; applications using Arduino modules.

In [5], comparative analysis of integrated development environment for creating mobile applications. Considered: Visual Studio, Android Studio, Eclipse IDE.

In [6] deals with Microsoft Visual Studio IDE and focuses on studying KaVE dataset; discusses build failures, uses code debugger, and as a result provides metric for measuring developer throughput.

In [7] provides complete guide to using Visual Studio.

The use of Eclipse and Theia IDE to develop graphical models is described in [8].

In [9] peculiarities of using Eclipse IDE for developing mobile applications.

In [10], [11] NetBeans environment and peculiarities of application development are investigated.

The analysis of modern development environments for mobile applications is considered in [12]-[14].

The specifics of extending IDE with plugins for detecting vulnerabilities during code entry are discussed in [15].

3. FEATURES OF MODERN INTEGRATED SOFTWARE DEVELOPMENT ENVIRONMENTS

The term "integrated development environment" means that everything necessary is provided to convert code into functioning programs.

IDE is only "program" in which all development is carried out. Such "program" usually contains many functions for:

- creation,

- corrections,

- compilation;

- deployment;

- software debugging.

Modern IDEs can significantly expand their functionality through various plug-ins. It is worth noting that some text editors (eg Vim, Visual Studio Code, Emacs), thanks to available plug-ins, also expand their functionality, bringing it to almost IDE capabilities.

An IDE is not just text editor. While text code editors, such as Sublime, offer many handy features such as syntax highlighting, custom interface, and advanced navigation tools, they only allow you to write code. At least compiler and debugger are required to create working applications.

The purpose of integrated environment is to combine different utilities in one module, which will abstract from performance of auxiliary tasks, thus allowing programmer to focus on solving actual algorithmic problem and avoid wasting time when performing typical technical actions (eg, compiler call). This increases productivity of developer.

It is also believed that close integration of development tasks can further increase productivity through possibility of introducing additional functions at intermediate stages of work.

For example, IDE allows you to analyze code and thus provide instant feedback and report syntax errors.

Most modern IIDEs are graphic. But first IDEs were used before graphical operating systems were widely used – they were based on text interface using function and hotkeys to call various functions (for example, Turbo Pascal, created by Borland).

The development environment includes:

- text editor;

- translator (compiler and / or interpreter);

- means of automation of assembly;

- debugger.

The IDE includes these components, as well as number of others.

Let's analyze most common environments for software development.

First place among IDEs in Visual Studio, second in Eclipse for 2021 [14].

Microsoft Visual Studio (VS) is most common environment used by many software development companies.

The common standard for Windows programming is free home versions. The Microsoft Visual Studio interface is shown in fig. 1 [16].

Many versions of this IDE are capable of creating all types of applications, from web applications to mobile applications, video games. This line of software includes many tools for compatibility testing.

VS – one of oldest software products for creating both console applications and have graphical interface.

VS has standard editor and debugger, and includes compilers, code execution tools, graphical constructors, and many other features to simplify software development process.
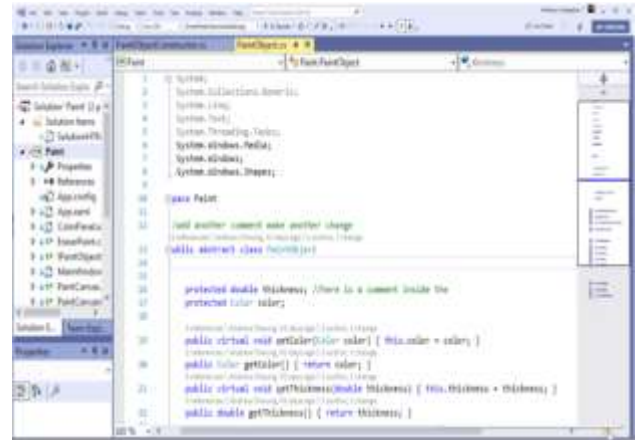

Figure 1. Microsoft Visual Studio 2019 interface

Features of VS are given in table. 1.

Table 1: Features VS

Feature	Description
Officially supported by platforms	Android, iOS and Windows
Supported languages	Ajax, ASP.NET, DHTML, JavaScript, JScript, Visual Basic, Visual C#, Visual C++, Visual F#, XAML and other.
Development	All types of programs ranging from web applications to mobile applications to video games.
Possibilities	Visual Studio 2015, 2017, and 2019 can be installed side-by-side. Visual Studio includes compilers.

Popular productivity tools Visual Studio:

1. Wavy lines indicate errors or potential code problems while typing. These visual cues help you fix problems immediately without waiting for build or runtime errors.

2. Quick Steps make it easy to create, modify, and refactor code in a single step. Quick Steps are available for C #, C ++.

3. Code cleanup, which is currently only available for C # code, helps you troubleshoot problems in your code before moving on to validate it.

4. Refactoring includes operations such as intelligently renaming variables, extracting one or more lines of code into new method, and changing order of method parameters.

5. IntelliSense – it is set of features that display information about your code directly in editor and, in some cases, automatically generate small snippets of code.

6. Live Share allows you to edit and debug code with other users in real time, regardless of programming languages you use and types you create.

7. The library is majestic as it grows steadily.

Adding third-party plugins can significantly expand functionality of environment, including to cross-platform state.

Advantages of VS:

- high speed of development;

- program is regularly updated;

- 36 programming languages;

- debugging tools are best way to track "mysterious" errors and diagnose "strange" program behavior;

- compactness (does not require a lot of memory to load);

- presence of native components in full;

- debugger support and code editor.

Disadvantages of VS:

- this IDE can be called one significant drawback – price;

- Visual Studio a foldable IDE, it is heavyweight to display and launch additional resource requirements;

- on some devices, when working with large files or when making simple changes, it can take a long time.

Another IDE will be Eclipse (fig. 2), is second most popular and used by about 24 % of programmers [14]. The C programming environment (which supports many other programming languages) is open source.

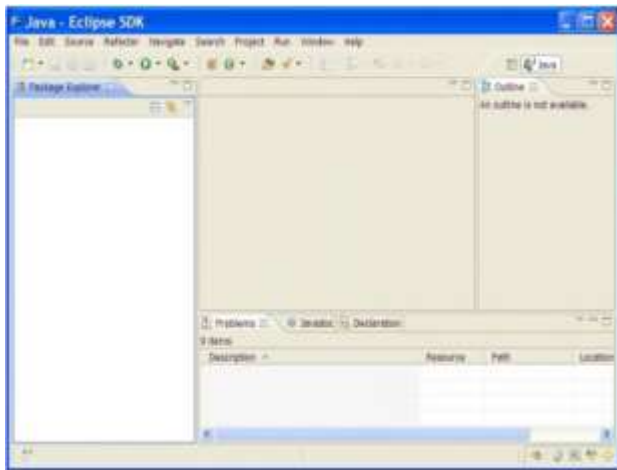The Eclipse interface is shown in fig. 2 [17].

Figure 2. Interface Eclipse

Eclipse is one of richest open source IDEs out there. It was originally primarily used for Java development, but now supports wider variety of languages.

This IDE comes with great graphical user interface and drag-and-drop functionality.

Features Eclipse hovering in table. 2.

Table 2: Features Eclipse

Feature	Description
Officially supported by platforms	Android (ARM), Windows, Linux and MacOS
Supported languages	C, C++, C#, Java, JavaScript, Perl, PHP, Python, Ruby, COBOL and other.
Development	Modular cross-platform applications
Possibilities	Integration with JUnit. compiler. Requires pre-installed JDK or SDK compiler
Access	Free

Eclipse is based on Rich Client Platform (RCP).

Popular productivity tools Eclipse:

- automatic code analysis;

- static code analysis;

- integration of GIT, CVS;

- sometimes it is necessary to disable spell checking, which will reduce load on eclipse.

Eclipse has wide range of capabilities due to its large number of plugins and extensions, that is, there are number of both free and commercial modules for environment.

Advantages Eclipse:

- serves primarily as platform for developing extensions, which it gained popularity, that is, developer can extend Eclipse with his modules;

Free and high quality has become prerequisite for Eclipse as corporate standard for application development in many organizations;

- easy to install;

- open source and free (unlike competitors such as IntelliJ IDEA);

- step-by-step non-stop compilation and instant error checking;

- heavy.

Disadvantages Eclipse:

- many of parameters of this development environment can be confusing for beginners;

- Eclipse does not have all features of IntelliJ IDEA, but it is open source IDE;

- work in Eclipse with large projects can be difficult, because IDE is slow, sometimes stops and does not respond, which may depend on OS, version of Eclipse;

- debugging function is not as perfect as that of other paid IDE environments;

- work is related to servers, and you have to run clean and build frequently.

The third framework will be NetBeans, one of best IDEs for C and C ++ programming. It has user-friendly interface as well as several useful project templates.

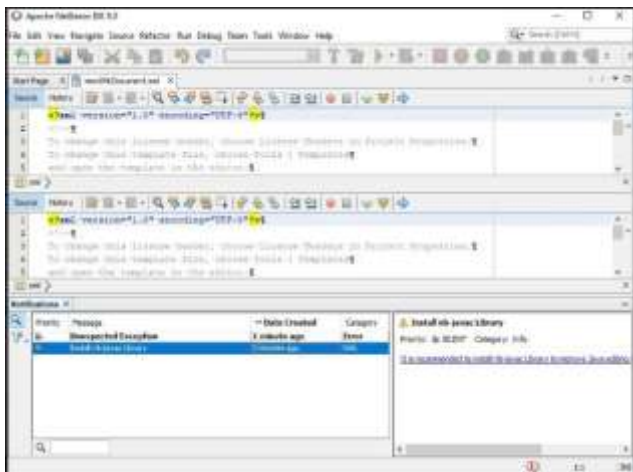Interface Eclipse is shown in fig. 3. Free open source development environment.

Figure 3. Interface NetBeans

NetBeans allows you to develop applications from set of modular software components called modules.

NetBeans is robust tool for writing low-level programming languages, particularly C.

The supplied functionality can be extended with variety of useful plugins.

Support for wide range of compilers, including CLang / LLVM, Cygwin, GNU, MinGW and Oracle Solaris Studio.

The IDE's workspace is fully customizable – you can customize panel actions, assign hotkeys, and more.

Popular productivity tools NetBeans:

- drag-and-drop functionality;

- integration of GNU-editor's decoder sessions with accompanying code;

- latest versions of NetBeans IDE support refactoring;

- highlighting syntactic constructions in color;

- auto-completion of typed constructions on fly and many predefined code templates;

- static analysis.

Features NetBeans hovering in table. 3.

Table 3: Features NetBeans

Feature	Description
Officially supported by platforms	Windows, Mac OS X, Linux and Solaris
Supported languages	C, C++, Java, JavaScript, Python, HTML, HTML 5, RHTML, JSP, PHP, CSS, XML, Groovy and Ruby
Development	Creation of professional desktop applications, corporate, mobile and web applications
Possibilities	Requires pre-installed Sun JDK or J2EE SDK of correct version. For C or C ++ in NetBeans, compilers must be installed separately.
Access	Free

Advantages NetBeans:

- free IDE;

- simple and effective tools for project management;

- ability to monitor development of project remotely;

- modularity;

- has sufficient set of tools for working with version control systems, which allows you not to use command line or third-party applications for this.

Disadvantages NetBeans:

- free development environment consumes a lot of memory, so it can run slowly on some PCs;

- need to pre-install Java Developer Kit to run NetBeans, which accordingly affects performance of application.

In course of modern IDEs review and their features, it was determined that when choosing development environment, it is necessary to take into account number of factors:

1. Access (paid / free).

2. Cross-platform – development environment that can be installed on various devices with different operating systems.

3. Bit width of OS.

4. Programming language.

5. Difficulty when working with large files.

6. Convenient navigation.

7. Compiler (built-in / external).

8. Support and updatability (if environment does not receive regular updates, it can hardly be called modern).

That is, these are key factors in making rational choice of IDE.

4. CONCLUSION

The article provides an overview of three most popular modern IDEs (VS, Eclipse, NetBeans), during which features of each environment, advantages and disadvantages are highlighted. As a result, this review can help in choosing an IDE, which in turn is vital aspect of becoming successful developer of modern applications, be it mobile or web applications.

An overview of today's most popular IDEs became prerequisite for identifying key factors to consider when choosing not only reviewed IDEs, but also similar environments. Choosing smart IDE has direct impact on your workflow, which will further improve your productivity.

5. REFERENCES

[1] Kuzomin, O., & et al.. (2020). Mobile Expert System for Diagnostic Human State in Emergency Situations. International Journal of Advanced Trends in Computer Science and Engineering, 9(4), 6485-6489.

[2] Baker J. H., & et al.. (2021). Some Interesting Features of Semantic Model in Robotic Science. International Journal of Engineering Trends and Technology, 69(7), 38-44.

[3] Omarov, M., Tikhaya, T., & Lyashenko, V. (2019). Internet marketing metrics visualization methodology for related search queries. International Journal of Advanced Trends in Computer Science and Engineering, 8(5), 2277-2281.

[4] Fezari, M., & Al Dahoud, A. (2018). Integrated Development Environment "IDE" For Arduino. WSN applications, 1-12.

[5] Dzhabrailova, T. A., & Magomaeva. L. R. (2020). Sravnitel'nyj analiz integrirovannoj sredy razrabotki dlya sozdaniya mobil'nyh prilozhenij. Innovacii. Nauka. Obrazovanie, 21, 1102-1105.

[6] Bellman, C., Seet, A., & Baysal, O. (2018, May). Studying developer build issues and debugger usage via timeline analysis in visual studio IDE. In 2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR) (pp. 106-109). IEEE.

[7] Johnson, B. (2019). Visual Studio Code: End-to-End Editing and Debugging Tools for Web Developers. John Wiley & Sons.

[8] Saini, R., Bali, S., & Mussbacher, G. (2019, May). Towards web collaborative modelling for the user requirements notation using eclipse che and theia IDE. In 2019 IEEE/ACM 11th International Workshop on Modelling in Software Engineering (MiSE) (pp. 15-18). IEEE.

[9] Filonenko, M. V., & Kramarenko, T. A. (2018). Osobennosti ispol'zovaniya sredy ECLIPSE IDE dlya razrabotki mobil'nyh prilozhenij. Informacionnoe obshchestvo: sovremennoe sostoyanie i perspektivy razvitiya, 351-354.

[10] Kostaras, I., & et al.. (2020). What Is Apache NetBeans. In Pro Apache NetBeans (pp. 3-28). Apress, Berkeley, CA.

[11] Kostaras, I., & et al.. (2020). Apache NetBeans: New Features. In Pro Apache NetBeans (pp. 63-72). Apress, Berkeley, CA.

[12] Artemov, A. A., & et al.. (2021). Analiz sovremennyh sred razrabotki mobil'nyh prilozhenij. Innovacii. Nauka. Obrazovanie, 30, 857-868.

[13] Yamskih, M. E. (2019). Sravnitel'nyj analiz programm dlya sozdaniya mobil'nyh prilozhenij. Aktual'nye problemy aviacii i kosmonavtiki, 3, 1-22.

[14] Mills, M. (2021). Luchshie IDE v 2021 godu s funkciyami i cenami: kakuyu IDE vybrat' v 2021 godu. ITIGIC.

[15] Baset, A. Z., & Denning, T. (2017, May). IDE plugins for detecting input-validation vulnerabilities. In 2017 IEEE Security and Privacy Workshops (SPW) (pp. 143-146). IEEE.

[16] Del Sole, A. (2021). Introducing Visual Studio Code. Visual Studio Code Distilled. Apress, Berkeley, CA, 1-15.

[17] Acharya, S. (2018). Data Analytics using R. McGraw-Hill Education.