e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:10/October-2023 Impact Factor- 7.868 www.irjmets.com

COMPARATIVE ANALYSIS OF SMART CONTRACT DEVELOPMENT ENVIRONMENTS: REMIX IDE VS. VS CODE

Swapnil Manish Jyotsna Pareek*1

*1Department Of Information Technology, B. K. Birla College(Autonomous), Maharashtra, India.

ABSTRACT

Blockchain technology has offered several helpful changes in the business application industry with the help of smart contracts. The Development environment of these smart contracts plays an important role in the development lifecycle. This study makes a comparison between two similar environments: Remix IDE and Visual Studio Code (VS Code). This study evaluated these areas based on various key metrics, including ease of compilation, file management, usability, community support, etc. By evaluating these metrics, this study aims to provide developers with useful information to make decisions and choose best practices when choosing development tools.

Keywords: Smart Contract Development, Remix IDE, Visual Studio Code, Comparative Analysis, Usability.

I. INTRODUCTION

Evolution of blockchain and smart contracts:

Launched by the mysterious Satoshi Nakamoto, blockchain technology was initially designed to facilitate the trust of peer-to-peer transactions through Bitcoin. This innovation created trust and transparency in a culture that is not controlled by one organization and is decentralized in nature. Bitcoin is known as the most popular cryptocurrency nowadays. People have gained more attention towards this decentralization structure of the finance industry and have explored another way through which transactions can be performed with more security and efficiency. Blockchain technology provides sustainability and scalability to various industries due to its decentralized nature. With these aspects of blockchain, new advancements and innovations were carried out later on bitcoin. Launched in 2015, Ethereum further expands the potential of blockchain by offering smart contracts. These programmable systems are self-executing digital contracts that decide the application's main logic, the software's architecture, and how it can be integrated with blockchain.

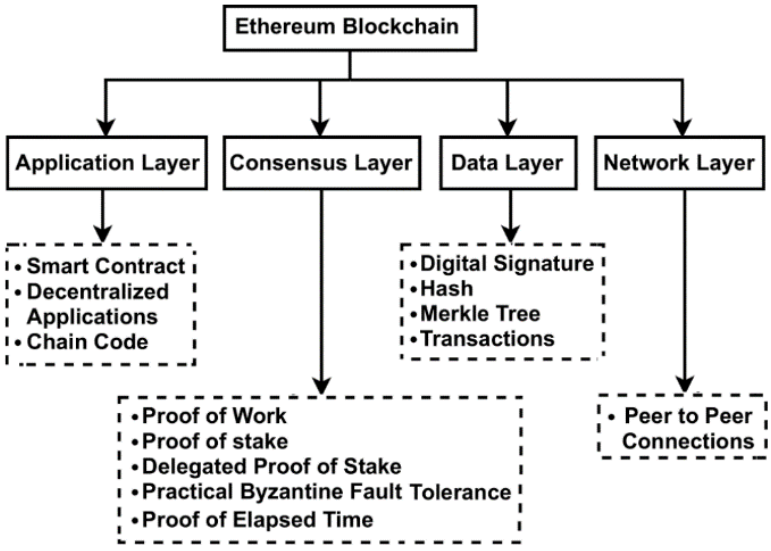

Figure 1: Layered structure of Ethereum blockchain

Significance of smart contract development:

Smart contracts offer many benefits within blockchain technology, including trust and transparency on a public blockchain, these contracts ensure the rules and guidelines for the service and the task that needs to be done, as cost efficiency by eliminating intermediaries as there are just developers who make these contracts and deploy it to the blockchain, great security through cryptographic algorithms, additionally, it uses addressing techniques to fetch the transactions and user, and global accessibility that facilitates borderless transactions

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:10/October-2023 Impact Factor- 7.868 www.irjmets.com

and international collaboration. These features underline the importance of smart contract development in revolutionizing digital contracts and transactions across various industries.

These contracts help the system to be more focused on the automation of the process. As it is made on and for decentralized platforms, users can be made trustless transactions and smart contracts provide the functionality to check and verify the transactions on blockchain as blockchain is immutable. As it offers high security, users need not to worry about the privacy. These smart contracts are generally written in solidity language, although there are several options such as Rholang, Go, Vyper, etc., but solidity has a huge community in terms of smart contract development and has a rich market with awesome features. Solidity offers a time-to-time update in terms of its versions. By using solidity, secure smart contracts can be developed using its functionality and user-friendliness.

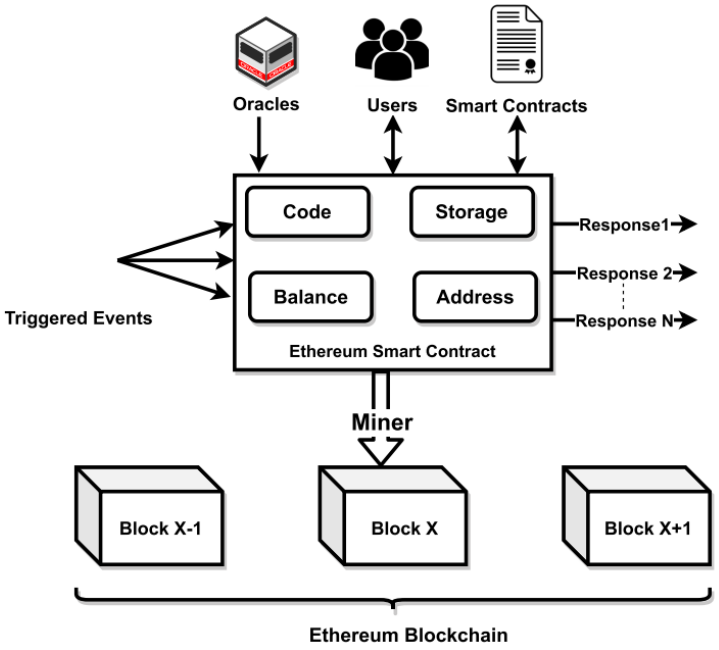

Figure 2: Core Working of an Ethereum Smart contract

As these smart contracts are deployed on blockchain and are immutable, the accuracy and reliability of these contracts are a big concern. Therefore it is important to test these smart contracts before deploying them. By testing the contracts we can find the bugs, errors, and many more mistakes that can cause harm to the organization or the system. In terms of testing smart contracts, there are various tools and frameworks such as truffle hardhat, waffle, slither, etc. Hardhat is one of the most used globally and it is reliable.

What is Hardhat?

The flexible Ethereum smart contract tool Hardhat offers a strong framework to streamline the development process. It distinguishes out for its excellent code authoring, thorough testing tools, expansive and diverse ecosystem of plugins, strong interaction with the Ethereum network, and scripting capability. Hardhat is a valuable tool for blockchain engineers because of these essential components, which enhance performance and promote blockchain technology. With these aspects, smart contracts have become an essential and a must thing to look upon. When it comes to developing these smart contracts, one should consider the best practices and best development environment in making these contracts to ensure that the final contracts are reliable and secure they can cause big harm to organizations, and if not made carefully they can lead to attacks, common examples include the DAO(Decentralized autonomous organizations) attack, reentrancy attacks, and much more. So it is a good practice to carefully look up the parameters from the start while choosing the development environment and learning their capabilities and limitations.

Research Objectives:

The primary objective of this research study is to conduct a thorough comparative analysis of two leading smart contract development environments, Remix IDE and Visual Studio Code (VS Code). The driving force behind this research stems from the pressing need to equip developers, especially those venturing into the

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:10/October-2023 Impact Factor- 7.868 www.irjmets.com

complex domain of blockchain and Ethereum development, with essential insights into the unique qualities and limitations of this development environment. Our research seeks to cover a comprehensive assessment of metrics, functionality, and user experiences, seeking to provide a robust framework that enables developers to make informed choices when faced with the critical decision of choosing the best tool to meet their specific smart contract development requirements. Ultimately, this research aims to contribute to the enhancement of the smart contract development process facilitating more efficient, secure, and innovative applications in the evolving landscape of blockchain technology.

Problem Statement:

As blockchain technology continues to evolve, the creation of smart contracts has become essential. However, developers often find it challenging to choose the best tools to create and implement smart contracts. This research compares two popular and intelligent design environments: Remix IDE and Visual Studio Code (VS Code). By exploring their features and functions, this study aims to provide beginner developers with information that will help them choose the optimal platform for their smart contract projects.

II. LITERATURE REVIEW

While exploring the functionalities and applications of blockchain technology, as demonstrated by David Nadler Prata et al[3]. and the co-authors, it becomes evident that smart contracts, a core focus of their work, play an important role in this digital landscape. In a recent study, Shafaq Naheed Khan and co-authors[1] explored the profound impact of blockchain technology and cryptocurrencies on the financial industry, fostering the emergence of a new crypto-economy and next-generation decentralized applications through smart contracts. They provided a comprehensive survey of blockchain-enabled smart contracts, discussed technical aspects, and identified challenges and future trends in this dynamic field. Furthermore, In a systematic exploration of smart contract development and blockchain applications, Anna Vacca[2] conducted a comprehensive literature review analyzing 96 articles from 2016 to 2020. The review primarily focuses on software engineering challenges and solutions within the realm of smart contracts and blockchain application development. The study encompasses six specific topics, including smart contract testing, code analysis, metrics, security,

Furthermore, In their study, S. S. Kushwaha[4], and co-authors focused on the Ethereum blockchain's security vulnerabilities within smart contracts, offering insights into detection tools, real-life attacks, and prevention measures. Their work highlights the need for ongoing research to address the challenges and vulnerabilities in this evolving domain.

III. METHODOLOGY

Study setup

This study creates a controlled and structured study setup to compare Remix IDE and Visual Studio Code (VS Code) as a smart contract development environment. The comparison was performed on a machine with the following specifications:

-Hardware: The machine is equipped with an AMD Ryzen 5 5600H processor, 8 GB RAM, and SSD storage and provides a good testing experience.

- Software: For comparison Node.js was installed, which is the main dependency on smart contracts, into the system. Latest versions of Remix IDE and VS Code to ensure that both environments were updated and optimized for experiments and installed the latest version of Hardhat to test and compile the smart contracts.

Components

Smart contracts: Written in solidity for testing on both environments.

Remix IDE: This integrated development environment provides key components such as the integrated and real-time Solidity compiler, a built-in Ethereum Virtual Machine (EVM) for local testing, and a user interface. Remix IDE provides a convenient environment for writing, compiling, and implementing smart contracts.

Hardhat: The latest version 9.6.7 of hardhat for compiling smart contracts.

Visual Studio Code (VS Code): When it comes to VS Code, extensions to blockchain and smart contracts are especially important. These extensions include Solidity syntax highlighting, debugging tools, and integration with multiple blockchain networks. Additionally, smart contracts created in VS Code are tested using the

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:10/October-2023 Impact Factor- 7.868 www.irjmets.com

Hardhat testing framework, which is an important part of ensuring code quality and reliability. As VS Code has a rich extension marketplace it would be a genuine comparison between both.

Metrics:

- Ease of compilation and testing: This study evaluates the ease of compilation and testing smart contracts with VS Code(using HardHat) in Remix IDE, focusing on the performance of development.
- File Management: This comparison examined file management capabilities in both environments, including file organization, management and ease of use, and smart contracts.
- User interface and development environment: This metric includes user interface (UI) design, overall user experience, and development usefulness of Remix IDE and VS Code.
- Compatibility and updates: This study analyzed both environments for compatibility with development tools and tracked their update history to ensure they remain compatible, updated, and forever adapting to change.
- Community and support: This study measured the size and activity of the Remix IDE and VS Code user and developer communities.
- Features and plugins: This study examined the plug-ins and additional features provided by Remix IDE and VS Code and evaluated their contribution to program performance.

For the 1st metric, a set of 8 smart contracts has been tested and compiled on both of the development environments. In the subsequent sections, we will dive into the experimental procedures and present our findings based on these metrics additionally we will analyze some other attributes for both of the development environments.

IV. COMPARATIVE ANALYSIS

Metric 1: Ease of Compilation and testing

Remix IDE: Remix IDE offers instant compilation, providing immediate error feedback in a user-friendly GUI. This makes it a favorable choice for developers seeking ease of use, showing performance and user-friendliness. However, it may not require additional tools for testing.

Visual Studio Code with Hardhat: Visual Studio Code, combined with Hardhat, compiles in a couple of seconds (hardly 1 or 2 seconds) but may require additional commands and external tools for thorough testing. It comes at the cost of requiring more manual testing steps.

Metric 2: File Management

Remix IDE: Remix simplifies file management, providing pre-existing files and artifacts for smart contract development. Users only need to write, compile, and deploy their contracts for testing.

Visual Studio Code with Hardhat: Integration is required by installing Hardhat, which automatically generates JSON and JavaScript configuration files and a contracts folder in the project directory. Users need to manage these files and structures as part of the development process.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:10/October-2023 Impact Factor- 7.868 www.irjmets.com

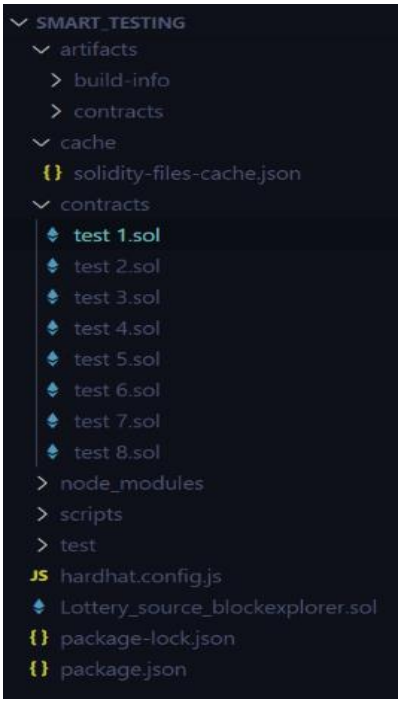

Figure 3: File management in VS Code after hardhat installation.

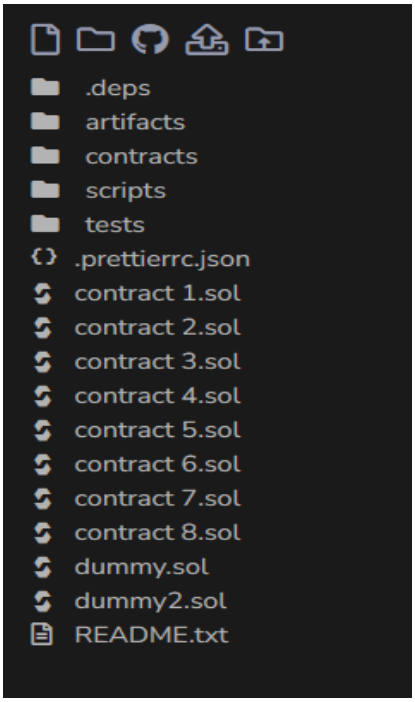

Figure 4: Default file Management of Remix IDE with smart contracts added.

Metric 3: User Interface and Development Environment

Remix IDE: Remix offers a user-friendly, web-based interface suitable for both beginners and experts. It reduces the learning curve for beginners while offering rapid implementation and effective testing for experienced developers.

Visual Studio Code with Hardhat: The combination of Visual Studio Code offers a versatile environment. Beginners may find it difficult at first, but Hardhat's extensive ecosystem, supported by documentation and community resources, eases the learning curve and allows for significant customization. Professionals appreciate this flexibility and advanced features, making it ideal for complex projects and experienced developers. Also, it offers tons of themes for the working environment.

Metric 4: Compatibility and updates

Remix IDE: Remix IDE has a strong connection with the Ethereum network and occasional updates, providing solid support for the latest Solidity compiler versions and Ethereum protocol changes. Currently, it is on v1.3.6. It is under the team lead of remix Yann Levreau[5].

Visual Studio Code with Hardhat: Visual Studio Code (VS Code) integrates with the Hardhat framework, offering flexibility in network selection. Developers gain the flexibility and confidence to adapt to specific network requirements and stay current in the dynamic Ethereum ecosystem. As VS Code is offered by Microsoft, it offers monthly updates and it can be seen on their official website.

Metric 5: Community and Support

Remix IDE: Provides a strong community, thanks to its popularity and direct Ethereum integration, offering accessibility to developers of all levels. As this field is eventually growing there are chances that it will be featured in other sectors too with its unique features.

Visual Studio Code with Hardhat: We observed that this environment is slowly tapping into a growing Ethereum community, complemented by a rich ecosystem of extensions and strong Hardhat support. Although it is much bigger now with numerous framework support such as hardhat, truffle, ganache, etc.

Metric 6: Features and plugins

Remix IDE: Remix IDE supports core plugins such as Sourcify, Etherscan, Tenderly, Flattener, and Gas Profiler, enhancing its functionality for Ethereum smart contract development.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:10/October-2023 Impact Factor- 7.868 www.irjmets.com

VS Code: It Offers rich development support with integration options for testing tools like Slither and Echnida. Its extension marketplace includes essential extensions such as "Hardhat," "Solidity," "Etherscan," "Truffle," "Live Share," and "Debugger for Solidity." These plugins enhance usability and functionality, making it a powerful choice for Ethereum developers.

V. RESULTS AND ANALYSIS

Table 1. Comparison of all the metrics of Remix vs Visual Studio Code

Metrics	Remix IDE	VS Code with hardhat
Ease of Compilation and testing functions	Real-time compilation with instant feedback on errors and warnings. Provides a GUI interface to interact with and test contract functions easily.	Compilation success notification after a couple of seconds. Requires additional commands and potentially external tools for testing contract functions.
File management	It simplified the project setup with minimal manual file management. Contracts can be created directly within Remix.	Requires manual creation of a project structure, including directories and configuration files. Generates dummy contracts, artifacts, JSON files, and additional configuration files but allows for customization and integration with a wide range of tools and extensions.
UI and Dev environment	This study has found that Remix IDE gets an edge by providing a user-friendly web-based interface that caters to both beginners and professionals. For beginners, Remix's intuitive and straightforward UI reduces the learning curve for smart contract development.	Visual Studio Code with Hardhat is versatile, accommodating both beginners and professionals. Beginners may face an initial learning curve, but it offers an extensive ecosystem for gradual adaptation. Professionals value its adaptability and advanced features for complex projects.
Compatibility and updates	Despite less frequent updates, it quickly adapts to Ethereum network changes, ensuring strong compatibility and a dependable environment. The latest web application is 0.36.2 while the desktop version is v1.3.6.	Provides monthly updates, but Remix excels with faster updates and superior compatibility, making it the choice for staying current in the Ethereum ecosystem. The latest version is v1.83 as of Sept 2023.
Community and support	Remix has a dedicated community of users and contributors, especially among Solidity developers.	Visual Studio Code is a widely adopted code editor and the Hardhat, as a popular Ethereum development framework, has an active community of users and contributors.
Features and Plugins	Remix IDE provides real-time Solidity code compilation, an integrated editor, and a user-friendly GUI for contract deployment and testing. It relies primarily on built-in functionality	Offers advanced debugging and testing features, with a rich extension marketplace. Key extensions include "Hardhat" for Ethereum development, "Solidity" for language support, "Etherscan"

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:10/October-2023 Impact Factor- 7.868 www.irjmets.com

	and plugins, offering features like a debugger, static analysis tools, and a plugin manager.	for blockchain integration, "Truffle", "Live Share" for collaboration, and "Debugger for Solidity" for debugging.

Here are some additional attribute comparisons for both development environments:

Table 2: Comparison chart for additional metrics

COMPARISON CHART

METRICS	REMIX IDE	VS CODE WITH HARDHAT
REAL TIME COMPILATION	✓	✗
GUI FOR DEPLOYMENT AND TESTING	✓	✗
RICH EXTENSION MARKETPLACE	✗	✓
MULTIPLE OS SUPPORT	✓	✓
ADVANCED DEBUGGING CAPABILITIES	✗	✓
DEVELOPMENT AND COMMUNITY SUPPORT	✓	✓

VI. CONCLUSION

To compare Remix IDE and Visual Studio Code (VS Code) with Hardhat, we looked at several important characteristics, including simplicity of use. Remix IDE is great for new users since it provides quick feedback and an easy-to-use interface. On the other hand, VS Code with Hardhat offers simplicity and flexibility, and it is designed for seasoned developers who can make use of its unique features and capabilities. The project requirements and the developer's skill level will determine which option is best. Future research will focus on improving the user interface, creating custom plugins, and adapting to new blockchain trends.

Future Opportunities: Future research on smart contract design environments may explore the continued evolution of the human-computer interface and user experience in the Remix IDE and VS code. Special plug-ins and tools that meet specific needs represent another area of research. Additionally, as blockchain technology evolves, these areas need to be constantly evaluated to adapt to new trends and technologies in the field.

ACKNOWLEDGEMENTS

I would like to thank my friends and professors whose support, guidance, and understanding were important in the success of this study. I would also like to thank my institute for providing the resources and environment for good research work.

VII. REFERENCES

[1] Khan, S.N., Loukil, F., Ghedira-Guegan, C. et al. Blockchain smart contracts: Applications, challenges, and future trends. Peer-to-Peer Netw. Appl. 14, 2901–2925 (2021). https://doi.org/10.1007/s12083-021-01127-0

[2] Anna Vacca, Andrea Di Sorbo, Corrado A. Visaggio & Gerardo Canfora, A systematic literature review of blockchain and smart contract development: Techniques, tools, and open challenges, Journals of system and software, Elsevier, Volume 174, April 2021, 110891. https://doi.org/10.1016/j.jss.2020.110891

[3] Prata, David & Araújo, Humberto & Santos, Cleorbete & Patel, Pratham. (2021). A Literature Review about Smart Contracts Technology. SSRN Electronic Journal. 8. 1-4. 10.22161/ijaers.82.1.

[4] S. S. Kushwaha, S. Joshi, D. Singh, M. Kaur, and H. -N. Lee, "Systematic Review of Security Vulnerabilities in Ethereum Blockchain Smart Contract," in IEEE Access, vol. 10, pp. 6605-6621, 2022, doi: 10.1109/ACCESS.2021.3140091.

[5] Yann Levreau, Team Lead at Remix, yann300 · GitHub.