American Academic Scientific Research Journal for Engineering, Technology, and Sciences
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402
http://asrjetsjournal.org/

Agile Testing for Blockchain Development QA

Ganesh Gatla[a*], Kanchan Gatla[b], Balaji Vishwanath Gatla[c]

[a]*480 Meadowhill Drive, Alpharetta, GA, USA*
[b]*25004 striped maple drive, Aldie, VA, USA*
[c]*1215 Yarrow Street, Stallings, NC, USA*
[a]*Email: ganeshgatla73@gmail.com,* [b]*Email: Kanachangatla12@gmail.com*
[c]*Email: Balajiv.gatla@gmail.com*

Abstract

Agile testing has evolved into a commonly employed practice in most development disciplines. It has been around as long as the agile manifesto and has developed all the hallmarks of a mature set of practices, i.e., tools, metrics, techniques, etc. But its overlap with blockchain is something that has yet to reach the maturity of either – agile testing or blockchain development. The QA for blockchain development hasn't been standardized in the same manner as the QA for web development and other areas of software development, even newer ones like cloud-native development. Agile testing leans heavily towards automation, Artificial Intelligence (AI), and Machine Learning (ML) and can benefit from collective or separate advances in the three technologies. But these technologies, regardless of their influence on blockchain development and its QA, cannot become the bridge connecting the two. Blockchain development QA suffers from a significant lack of standardization and a unified set of good practices, and this hinders its ability to adapt agile testing practices into the existing paradigm. However, as blockchain development is adopted by agile teams and its QA becomes more standardized, we may see more overlap between agile testing and blockchain development QA.

Keywords: Agile testing; blockchain development; blockchain QA.

1. Introduction

The studies around Agile software development and testing practices go as far back as the agile manifesto itself (2001). The term "Agile testing" was used in the manifesto as an explorative avenue, specifically for Brian Marick, a software testing consultant and one of the authors of the manifesto [1]. Studies targeted specifically at Agile testing started emerging in 2002, and the earliest studies focused on organizational acceptance of agile testing and mistakes made by agile testing teams [2], testing checklists for specific development domains (database) [3], and good practices. Later research (until 2011) explored different tools [4], metrics [5], techniques [6], and disruption to conventional testing practices. Most of the literature produced in the last decade concerning agile testing is broader in scope.

---

* Corresponding author.

Agile testing has become a mature concept now, and a significant amount of research material exists, covering both broad spectrum and niche avenues like evolving testing practices, framework [7], its interaction with distributed software development and testing, and Machine Learning (ML). Agile software development and other approaches/methodologies are constantly evolving, including some radical changes like cloud-native development, AI-based development practices, and the mainstream adoption of blockchain.

The term blockchain predates Bitcoin, the first cryptocurrency and, arguably, the starting point for blockchain and cryptocurrency in modern history, by 17 years. The term and its description were first provided by Stuart Haber and W Scott Stornetta in 1991. But it wasn't until the whitepaper was published [8] under the pseudonym Satoshi Nakamoto in 2008 that the idea of blockchain really took root in the mainstream tech community. Even though the first real-world application of blockchain was cryptocurrency, research around its other applications started emerging in the early years. Blockchain development, however, remained a muted topic up until 2015 and 2016. Most of the early research around blockchain development focused on its application in the finance industry (due to its cryptocurrency applications) [9], but its application in the healthcare industry [10] and smart contracts were explored as well. Research surrounding blockchain development has gained more traction in the last few years, covering multiple aspects of this development avenue. Research regarding fiduciary and compliance practices (financial sector/taxation [11]) regarding blockchain development is relatively extensive due to blockchain's financial lean (cryptocurrency). A decent amount of research has been done on blockchain development for healthcare applications/healthcare use cases due to stringent regulatory compliance requirements in this arena.

This paper aims to explore the Quality Assurance/QA or testing phase of blockchain and, to an extent, blockchain application development and whether or not agile testing practices can be applied to blockchain development QA. We also aim to identify whether the QA testing of blockchain apps goes a layer deeper into the blockchain app as well and whether agile testing can result in better blockchain applications if applied to the blockchain development stage. This will be conducted by exploring the research material addressing the direct overlap of blockchain development and agile testing, then zooming out from significant to tangential overlaps of agile testing and blockchain development.

2. Agile Testing

Agile testing commonly refers to software/application testing following agile methodology. However, the actual interpretation may vary based on a developer's/development team's or an organization's approach to agile. Different attempts have been made to standardize the process, including adding it to a software testing syllabus [12] and developing a model for teaching and testing in an agile environment [13]. Some interpretations of agile testing are connected to technical debt [14]. Earlier in this century, when agile was still a budding concept in the software development community, and the practices weren't nearly as commonplace as they are now, many professional testers had certain qualms about the potential efficiency of agile testing, but practical examples, especially on large-scale projects answered some of the major concerns they had [15]. Testing is a crucial part of agile development, and continuous testing is one of the defining characteristics of an agile environment and follows the same principles. This includes iterative and incremental testing [16]. Iterative testing is the process

of continuously user-testing an application, a software package, or a feature, making small changes each time (after each iteration) to finalize the perfect product. NIST defines incremental testing as testing a system/device after a small change to see whether or not it has impacted the overall functionality or security of the system. It's also referred to as regression testing in the agile development context. A natural benefit of this approach is that testing can be more localized, i.e., focused on specific developmental elements, functionalities, and security vulnerabilities (when working with DevSecOps). A far-reaching benefit of early and continuous agile testing is that it also helps align development goals with business goals, like providing more certainty regarding the marketability of the tested system and, by extension, the product it's being developed for [17]. Another trend in agile testing that's gaining traction is its overlap with Machine Learning (ML), which can augment automated agile testing, facilitating easy test and development integration for a variety of agile environments.

One example is using machine learning to automate which tests need to be prioritized in which development phases, i.e., tests that are more likely to yield errors than others or tests that may yield more errors than others are run prior to others. A machine-learning algorithm decides this hierarchy [18]. Another example of machine learning used to improve agile testing via automating the Root Cause Analysis (RCA), which aims to identify the cause of an error system failure. This is a natural fit as the continuous nature of agile testing generates a lot of raw data which can be used to train an ML algorithm. And when trained, the ML can partially or fully replace the manual RCA [19].

Limitations of agile testing (and agile methodologies at large) should also be taken into account. One of which is the systems where hardware and software development is to keep pace. When researched in the context of mechatronics, it was identified that the difference in software and hardware prototyping timeline is a major limitation [20]. If the software is developed and tested ahead of the hardware, it may have to be retested for the hardware, revealing different issues. This repetition goes against the core benefit agile development and testing offers.

3. Blockchain Development QA

A good interpretation of blockchain and the problem it solves is: establishing trust in a distributed system [21]. This is what's behind the groundbreaking impact of cryptocurrency, as it directly challenges the established financial system where banking institutions broker trust when it comes to financial transfers. However, the immutability of blockchain, i.e., once a change is made, it cannot be reverted or deleted, is not considered an infallible attribute by all the experts, and some have raised serious concerns about it in the past, and the possibility of its adoption and absorption (albeit not in its purest form) was hypothesized in the past, and it's now happening [22] (to an extent).

Bitcoin and Ethereum are two of the most popular blockchains, developed in 2009 and 2015, and currently, there are at least a thousand different blockchains. Virtually anyone can develop a blockchain, but in the presence of public blockchains like Bitcoin and Ethereum and newer blockchains like Solana, more developers focus on blockchain app development rather than new blockchain development. Still, there is adequate research available on blockchain testing, which can help us identify different practices, tools, and trends associated with

blockchain testing and QA.

A 2017 paper [23] on the overview of blockchain technology divided blockchain testing into two phases:

- The standardization phase focuses on testing a blockchain for its standard/agreed-upon/claimed features.
- The testing phase, in which the blockchain is tested for a specific or wide range of potential use cases.

The cost was identified as a challenge in blockchain/blockchain application testing because running a blockchain on thousands of hardware nodes (to test in the real-world environment) would be a costly endeavor. Virtualization is a potential answer [24] to this cost problem, which can be augmented by containerization in a cloud environment. The testing and QA complexity are different with the type. Private blockchains are more challenging to test compared to public ones [25].

Then there are testing frameworks for specific blockchain features/applications, like penetration testing for smart contracts [26]. Due to their decentralization and non-standardized control practices, blockchains are vulnerable to unique attack vectors not present in traditional comparable apps/systems. Thus, it requires a unique approach to testing as well. Penetration testing to identify code vulnerability in a blockchain identified problems like Timejacking for the Bitcoin blockchain or Integer Overflow vulnerability for blockchains in general.

QA and testing of blockchain-based applications is a far more extensive avenue to explore because it adds another layer of code over the code running the blockchain, managing nodes, working on consensus mechanisms, etc. Integration testing, performance testing, specific use case testing, and security testing (independent of the underlying blockchain) make the blockchain development QA in the context of individual applications more complicated. Efforts to standardize blockchain-based application and software testing have been made and have identified certain elements like Smart Contract Testing, Blockchain Transaction Testing, and vulnerabilities in the modeling languages (which may be different for other mainstream applications) that should be part of the typical testing process [27].

Blockchain development QA and testing are still evolving disciplines, but a number of challenges associated with blockchain application testing have already been identified. Starting with lack of standardization, architectural unfamiliarity, lack of specialized tools, and challenges associated with the nature of blockchain, like immutability, which might make testing errors costly, and its distributed nature which makes it challenging to use automated tools like the ones used for dependability assessment. These challenges remain, but there is progress as well, especially in the form of specialized tools for testing and benchmarking tools for performance evaluation [28].

4. Agile Testing For Blockchain Development QA

The research focused on Agile blockchain, and blockchain-based application development can also offer significant insights into agile testing for blockchain development and QA since testing is an integral part of

blockchain development. One aspect of agile that can easily be applied to blockchain QA is breaking down the process into small components. This methodology can be applied to individual block testing in a blockchain. Smart contract testing, which is a core element of blockchain QA and testing, is ideally suited for agile user stories, though the parameters have to be defined differently for blockchain QA. Different business users may have different potential use cases for a blockchain's smart contracts, and taking each user story into account can help a blockchain development team (or blockchain application development team) to streamline the addition and modification of functionalities within a smart contract and how it's executed for different end users.

The lack of standardization in blockchain testing paradigms and QA makes it challenging to identify areas where agile testing methodologies can be applied. One way to meet this challenge is to explore different types of blockchain testing processes. Functional testing, i.e., testing different functional areas of a blockchain, like smart contracts and native tokens, to assess if they are working as intended. Functional testing directly on public blockchains, where nodes can join and leave ad-hoc, consistently changing the size of the blockchain [29] may not be practical. So agile testing can be conducted on virtual machines or dummy blockchains for each functionality or each addition to functionality before it's deployed. The same approach can be applied to blockchain application development QA, i.e., testing each function as it's developed, including its integration to the underlying blockchain, instead of pushing the testing right of the development process (Waterfall testing).

Simulations and analytical modeling are typically employed for a blockchain's performance evaluation [30]. Simulations are a useful tool in an agile testing paradigm, creating an overlap with blockchain performance QA. The analytical modeling-based performance evaluation can be made agile by modifying and reapplying the analytical model for every new test case/user story.

Security testing is a crucial part of blockchain development QA. Penetration testing can be considered a part of security testing, though it's usually cited as a separate blockchain testing category. An agile approach to security testing would be to test the blockchain or blockchain app after every addition or change in code or in the environment. Agile security testing, in this perspective, can go a layer deeper with multiple test iterations *during* the development of blockchain, an app, or a feature to test for vulnerabilities. Static testing can be conducted if dynamic testing or hybrid testing isn't feasible.

A few insights generated from assessing the overlap of agile testing with blockchain development QA are:

- The rarity and complexity of blockchain development, which requires an in-depth understanding of consensus mechanisms, blockchain architecture, hardware nodes, etc., and the fact that it's still lacking tools and standardized practices (compared to web/app development), reduces the parallels between each blockchain development. All development processes/approaches might be too unique to be streamlined via a broad set of good practices, which includes agile testing.
- The resource cost of iterative testing on blockchains, where each block might be a precious resource, especially if the total number is capped, can be too high.
- Different public blockchains have different testing tools that can be applied to the QA of blockchains and apps leveraging the underlying public blockchain. But these tools may not be interchangeable with

other blockchains with different consensus mechanisms and architecture. But if tools and agile testing frameworks are designed for a commonly used blockchain like Ethereum, it may trigger the development community behind other public blockchains to do the same, but the frameworks and agile testing practices may still be disparate due to the inherent differences between the underlying blockchains.

5. Result

Agile testing can be applied to blockchain development QA, but there is a low probability that it will ever be standardized, partly because blockchain development QA has yet to be standardized and streamlined itself. But since agile testing, like agile development methodology, is a flexible set of good practices and approaches rather than a rigid framework, different blockchain development teams can interpret it differently and still benefit from the advantages it offers. It might be easier to adopt for Blockchain app development, especially if it's high level enough to be relatively architecture-agnostic since the agile-testing practices in web development can be adapted to this end (with adjustments). It can also be easier for private blockchains, where developers and testers will have more control over the nodes, their spread, and the computing power they lend to the chain, compared to public blockchains. A decent selection of testing and evaluation tools and frameworks for a public blockchain like Ethereum can help with blockchain application testing and QA, following agile methodologies.

6. Discussion

Further research is needed to explore the overlap of agile testing and blockchain development QA in more depth and from a different perspective, including whether or not blockchain developers can adopt an agile methodology for development and testing independent of each other. It's also important to discuss and evaluate the role compliance plays in blockchain development QA. If the compliance requirements are enforced at the blockchain level by regulatory bodies for applications based on the blockchain to be acceptable in domains like financial services and healthcare, it may radically change how blockchains are developed, improved, and tested for QA.

There are still boundaries between web and app development and blockchain development (including blockchain applications), but they are being eroded rapidly. It's important to identify whether developers who typically follow agile methodologies and testing will be able to make agile testing a standard part of blockchain development QA or if they will face difficulties because of blockchain QA's inherent clashes/contrast with agile testing. Agile testing predates the first generation of blockchains. It has evolved faster than blockchains, and studies into whether blockchain development has adopted agile testing practices at the same pace as other development circles did, and if not, then the reasons behind the reluctance and lack of adoption, can be instrumental in defining agile testing's future in blockchain development QA. Ideally, these studies should also explore the differences in testing practices of first, second, and third-generation blockchains and how they could have been improved by agile testing. This would give us more insights about the currently being developed fourth-generation blockchains (and whether they can benefit from agile testing) and their potential for fifth and later-generation blockchains.

7. Conclusion

Blockchain development, testing, and QA have yet to mature and evolve further, and future generations of blockchain may lean more toward agile testing and development. If more DevOps teams, well-versed in agile methodology that has already implemented agile testing on several projects, start working on blockchain development, they may contribute towards streamlining and redefining agile testing for blockchain development QA. But as it stands now, blockchain development hasn't integrated agile testing as well as other facets of software development, and it has yet to become a standard practice for blockchain QA.

References

[1] K. Beck, M. Beedle, A. v. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland and D. Thomas, Manifesto for Agile Software Development, Agilemenfesto.org, 2001.

[2] K. Johansen and A. Perkins, "Establishing an Agile Testing Team: Our Four Favorite "Mistakes," *Extreme Programming and Agile Methods — XP/Agile Universe 2002,* vol. 2418, pp. 52-59, 2002.

[3] R. Nelson, "A Testing Checklist for Database Programs: Managing Risk in an Agile Environment," *Extreme Programming and Agile Methods - XP/Agile Universe 2003,* vol. 2753, pp. 91-95, 2003.

[4] R. A. Razak and F. R. Fahrurazi, "Agile testing with Selenium," in *IEEE - 2011 Malaysian Conference in Software Engineering*, Johor Bahru, Malaysia, 2011.

[5] I. Kayes, "Agile Testing: Introducing PRAT as a Metric of Testing Quality in Scrum," *ACM SIGSOFT Software Engineering Notes,* vol. 36, no. 2, pp. 1-5, 2011.

[6] B. K. Madhu, M. Jigalur and V. Lokesha, "A study on Agile Software Testing: Emergence and techniques," *African Journal of Mathematics and Computer Science Research,* vol. 3, pp. 288-289, 2010.

[7] J. Yu, S. Zhu, J. Zhang, Y. Chen, N. Wu, Y. Mei, D. Zhang, and L. Zhu, "An Agile Testing Framework of Four Quadrants," *Journal of Physics: Conference Series,* 2021.

[8] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," Bitcoin.org, 2008.

[9] D. Shrier, D. Sharma, and A. Pentland, "Blockchain & financial services: The fifth horizon of networked innovation," *MIT Connection Science,* 2016.

[10] R. Krawiec, M. Filipova, F. Quarre, D. Barr, A. Nesbitt, K. Fedosova, J. Killmeyer, A. Israel and L. Tsai, "Blockchain: Opportunities for Health Care," Deloitte Consulting, 2016.

[11] A. Schwanke, "Bridging the digital gap: How tax fits into cryptocurrencies and blockchain development," *International Tax Review,* 2017.

[12] I. S. Elgrably and S. R. B. Oliveira, "Construction of a syllabus adhering to the teaching of software testing using agile practices," in *2020 IEEE Frontiers in Education Conference (FIE)*, Uppsala, Sweden, 2020.

[13] I. S. Elgrably and S. R. B. Oliveira, "Model for teaching and training software testing in an agile context," in *2020 IEEE Frontiers in Education Conference (FIE)*, Uppsala, Sweden, 2020.

[14] R. Bavani, "Distributed Agile, Agile Testing, and Technical Debt," in *IEEE CloudCom 2012*, Taipei, Taiwan, 2012.

[15] D. Talby, A. Keren, O. Hazzan and Y. Dubinsky, "Agile software testing in a large-scale project," *IEEE Software*, vol. 23, no. 4, pp. 30-37, 2006.

[16] L. T. Heeager and P. A. Nielsen, "A Conceptual Model of Agile Software Development in a Safety-Critical Context: A Systematic Literature Review," *Information and Software Technology*, 2018.

[17] S. Paluch, D. Antons, M. Brettel, C. Hopp, T.-O. Salge, F. Piller, and D. Wentzel, "Stage-gate and agile development in the digital age: Promises, perils, and boundary conditions," *Journal of Business Research*, pp. 495-501, 2020.

[18] L. Butgereit, "Using Machine Learning to Prioritize Automated Testing in an Agile Environment," in *2019 Conference on Information Communications Technology and Society (ICTAS)*, Durban, South Africa, 2019.

[19] J. Kahles, J. Törrönen, T. Huuhtanen, and A. Jung, "Automating Root Cause Analysis via Machine Learning in Agile Software Testing Environments," in *2019 12th IEEE Conference on Software Testing, Validation and Verification (ICST)*, Xi'an, China, 2019.

[20] J. Heimicke, M. Niever, V. Zimmermann, M. Klippert, F. Marthaler and A. Albers, "Comparison of Existing Agile Approaches in the Context of Mechatronic System Development: Potentials and Limits in Implementation," in *International Conference On Engineering Design, Iced1*, Delft, The Netherlands, 2019.

[21] M. D. Pierro, "What Is the Blockchain?," *Computing in Science & Engineering (IEEE CS and AIP)*, 2017.

[22] D. Andolfatto, "Blockchain: What It Is, What It Does, and Why You Probably Don't Need One," *Federal Reserve Bank of St. Louis Review*, pp. 87-95, 2018.

[23] Z. Zheng, S. Xie, H. Dai, X. Chen and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," in *2017 IEEE 6th International Congress on Big Data*, Honolulu, HI, USA, 2017.

[24] C. Chen, Y. L. Zhuyun Qi and K. Lei, "Using Virtualization for Blockchain Testing," in *International Conference on Smart Computing and Communication*, 2018.

[25] B. KOTESKA, E. KARAFILOSKI, and A. MISHEV, "Blockchain Implementation Quality Challenges: A Literature Review," in *Proceedings of the SQAMIA 2017: 6th Workshop of Software Quality, Analysis, Monitoring, Improvement, and Applications*, Belgrade, Serbia, 2017.

[26] A. Bhardwaj, S. B. H. Shah, A. Shankar, M. Alazab, M. Kumar and T. R. Gadekallu, "Penetration testing framework for smart contract Blockchain," *Peer-to-Peer Networking and Applications*, vol. 14, p. 2635–2650, 2021.

[27] S. Porru, A. Pinna, M. Marchesi, and R. Tonelli, "Blockchain-oriented Software Engineering: Challenges and New Directions," in *2017 IEEE/ACM 39th IEEE International Conference on Software Engineering Companion*, Buenos Aires, Argentina, 2017.

[28] C. Lal and D. Marijan, *Blockchain Testing: Challenges, Techniques, and Research Directions*, arxiv, 2021.

[29] R. Koul, "Blockchain Oriented Software Testing - Challenges and Approaches," in *2018 3rd International Conference for Convergence in Technology (I2CT)*, Pune, India, 2018.

[30] S. Smetanin, A. Ometov, M. Komarov, P. Masek, and Y. Koucheryavy, "Blockchain Evaluation Approaches: State-of-the-Art and Future Perspective," *MDPI - Sensors*, vol. 20, no. 12, 2020.