Towards Blockchain Developer Experience (BcDEx): Exploring Dimensions of Developer Experience in Blockchain-oriented Software Engineering

Pamella Soares
Graduate Program in Computer Science
State University of Ceará
Fortaleza, CE, Brazil
pamella.soares@aluno.uece.br

Allysson Allex Araújo
Science and Technology Center
Federal University of Cariri
Juazeiro do Norte, Brazil
allysson.araujo@ufca.edu.br

Raphael Saraiva
Jerffeson Souza
Graduate Program in Computer Science
State University of Ceará
Fortaleza, CE, Brazil
raphael.saraiva@aluno.uece.br
jerffeson.souza@uece.br

ABSTRACT

The Developer Experience (DEx) in the context of blockchain and decentralized applications has to deal with new or adapted software engineering practices due to the particular characteristics of these technologies. This study proposes the idea of Blockchain Developer Experience (BcDEx) influenced by different practices of Blockchain-Oriented Software Engineering (BOSE). To achieve this purpose, we categorize and discuss critical BcDEx factors based on a well-established DEx conceptual framework covering cognitive, affective, and conation dimensions. Furthermore, we reveal new directions that can benefit both research and practice in BcDEx, encompassing practical aspects such as using technical resources and organizational considerations.

KEYWORDS

Blockchain Developer Experience, Blockchain-oriented Software Engineering

1 INTRODUCTION

Blockchain is a distributed data storage structure safeguarded by cryptography and governed by a consensus mechanism, providing benefits such as auditability, integrity, security, and transparency [4]. The evolution of blockchain technology has driven innovative businesses beyond the financial context with cryptocurrencies. This evolution was possible through Smart Contracts (SCs), which are scripts that execute the business logic of the network when certain conditions are met [3]. The use of SCs has leveraged the deployment of Decentralized Applications (dapps), a blockchain-based software usually built with no single server or entity controlling them [34].

In particular, the engineering of blockchain-based software can become complex and prone to critical errors due to the need to address challenging factors, such as distributed network infrastructure, security, etc. [26, 41]. Hence, the blockchain domain demands new software practices that deal with immutable, decentralized, and distributed databases and peer-to-peer networks, which differs from general software development [22]. Based on these particularities and the demand for specialized Software Engineering (SE) practices, Porru et al. [33] proposed the concept of "Blockchain-Oriented Software Engineering" (BOSE), defining it as "all software working with an implementation of a blockchain". In this context, Destefanis et al. [10] emphasize that BOSE is fundamental for shaping new directions in effective blockchain software development, serving as a bridge between traditional SE and this new paradigm.

Recently, SE scholars and practitioners have been arguing about the importance of paying attention to Developer Experience (DEx) due to its impact on productivity, code quality, and overall project success [29]. In summary, DEx encompasses the experiences related to all types of artifacts and activities that a developer encounters in the software development process [13]. However, there is a lack of studies examining DEx in the context of blockchain. A manual search on Google Scholar retrieved two studies that contextually mentioned DEx, although their primary focus was different. van Tonder [48] proposed a developer-oriented tool to enhance the visualization of SC source code. In turn, Chaurasia and Kamber [8] only briefly mentioned the necessity of DEx evaluation but did not concentrate on DEx in their comparative analysis of blockchain ecosystems and developer tools. While other BOSE studies have conducted experiments with developers [6, 39, 53] or investigated social aspects based on software evolution [9, 36], these did not explicitly frame their research within the DEx paradigm.

Without an understanding of DEx within BOSE, it becomes challenging to identify strategies for empowering blockchain developers to work more efficiently, effectively, and securely. Bridging this knowledge gap is critical to optimizing BOSE processes, fostering the growth of blockchain technology, and improving the overall well-being of blockchain developers. Drawing inspiration from the well-known DEx framework proposed by Fagerholm and Münch [13], this position paper aims to introduce the novel concept of Blockchain Developer Experience (BcDEx) within the context of BOSE. By deepening our understanding of BcDEx, we can address blockchain developers' specific pain points and requirements, thus boosting their productivity, well-being, and the overall quality of their development processes. Through exploring opportunities in BcDEx, we also aim to uncover new avenues for research and practical applications within BOSE.

This study presents two main contributions. Firstly, we introduce the concept of BcDEx and examine the key factors that shape the dimensions of cognition, affect, and conation. Secondly, we highlight new directions and insights centered on BcDEx, emphasizing the importance of this viewpoint in relation to its implications for both SE academia and practice.

2 METHODOLOGICAL PROCEDURES

To accomplish our research objective, we followed three main methodological steps:

(1) **Gap Identification:** Initially, we conducted an ad-hoc literature search using the search string "`developer experience framework`" `AND blockchain` in both white and grey literature. However, the search did not return any framework focusing on blockchain development, suggesting a gap in the literature concerning DEx framework proposals in the context of blockchain technology.
(2) **DEx Framework Orientation:** We approached the well-established DEx generic framework proposed by Fagerholm and Münch [13] to guide our proposal, which presents three main dimensions of DEx: development infrastructure (Cognition); feelings towards work (Affect); and value of one's own contribution (Conation).
(3) **BcDEx Design:** The following sub-steps were carried out for each dimension towards the definition of our framework:
   (a) *Factors Identification and Analysis:* This sub-step encompassed another ad-hoc literature search (in white and grey sources). The first author identified and analyzed factors based on the definitions proposed by the DEx framework (as summarized in Figure 1). For example, according to Fagerholm and Münch [13], "Platforms" is considered one of the Cognition factors. Based on that, we analyzed the blockchain platforms in the literature. This process was performed for each factor of each dimension. In addition, other new factors (e.g., Web3 hackathons and Technical job roles) emerged from our research.
   (b) *Factors Validation:* Factors and their implications were validated by the other three co-authors: one with more than four years of experience as a blockchain developer and two others with eight years of experience researching blockchain and cryptoeconomics. Disagreements were discussed among the authors, and factors were re-evaluated as necessary.

3 EXPLORING THE DIMENSIONS OF BLOCKCHAIN DEVELOPER EXPERIENCE

This section discusses the factors identified in the blockchain ecosystem based on dimensions proposed by Fagerholm and Münch [13].

3.1 [Cognition] DEx factors on blockchain systems development infrastructure

> *"The **cognitive dimension** consists of factors that affect how the developers perceive their development infrastructure on an intellectual level. This includes concrete interactions with development tools and execution of a software process." [13]*

3.1.1 Platforms and Development Enviroments. Currently, we are faced with different blockchain ecosystems [8], which represent interconnected networks of participants, technologies, and applications built around a specific ***blockchain*** protocol. Irimia et al. [19] identified at least a thousand blockchain systems ready to be used by 2022. In this regard, the developer or role in charge of the technical decisions must possess the expertise to align the business model with the appropriate technologies to be used, especially the choice of a blockchain network and the evaluation of the technical feasibility [1]. The literature has presented frameworks to guide these decisions, considering different properties such as transparency, privacy, integrity, storage, immutability, scalability, efficiency, latency, and interoperability [5, 50].

However, defining the infrastructure is just one of the initial challenges in the experience of developing blockchain-based software, typically undertaken by senior software engineers. In this process, the professional can also handle two main barriers: complex implementation and usage due to the distributed nature of blockchain and its particularities and a scarcity of experienced blockchain developers within the team [24]. To mitigate these problems and improve the learning curve, platforms such as Bitcoin, Ethereum, Polygon, Polkadot, and Hyperledger Fabric have provided exclusive ecosystems for developers, each with its characteristics and a wide range of tools that empower them to develop innovative solutions securely and transparently [8].

One of the main features each blockchain ecosystem provides is the supported or compatible ***programming languages***. Implementing specific languages for SCs can impact the SE practices and the DEx due to the immutable nature of the blockchain. In this regard, Solidity is still one of the most widely used for SC development based on the Ethereum Virtual Machine (EVM) [40], a JavaScript-like language with readable and easily understandable code. Vyper is also a language designed for blockchains that aims to make it challenging to write deceptive or malicious code and protect developers from unintentional vulnerabilities in contracts [20]. In turn, the Rust language has emerged prominently in this context, originating in 2006 and being extensively employed in blockchain development due to its emphasis on performance and security. However, Rust's syntax is significantly more complex, which can potentially prevent the onboarding of developers.

Although these programming languages have been tested in the industry, the literature could expand investigations into how their characteristics impact DEx. For example, Parizi et al. [32] analyzed usability, using implementation time per developers when working with Solidity, Pact, and Liquidity. Voloder and di Angelo [49] examined metrics such as lines of code, required experience, and number of issues, although the use cases were implemented only by the first author. These studies highlight the ongoing need for further research on DEx in practical applications of SC languages. Evaluating the use of established and emerging languages, such as updated versions of Solidity, Vyper, Rust, Move, and Cairo, among others, could yield practical benefits for enhancing DEx, including the advancement of SC Development Environments [33, 37].

Web3 development environments and native libraries have been established to facilitate developers' entry into the Web3 ecosystem and simplify the creation of dapps, enhancing development efficiency and ensuring an optimized cost-effectiveness [52]. For the development environment, tools such as Truffle, Hardhat, and Brownie are frameworks for testing, compiling, debugging, and deploying SCs through local, private, or testnet nodes. The choice of such tools depends on the blockchain and development stack preferred by the team, including Web2 languages such as JavaScript, Python, or Java. In turn, native libraries are also fundamental in

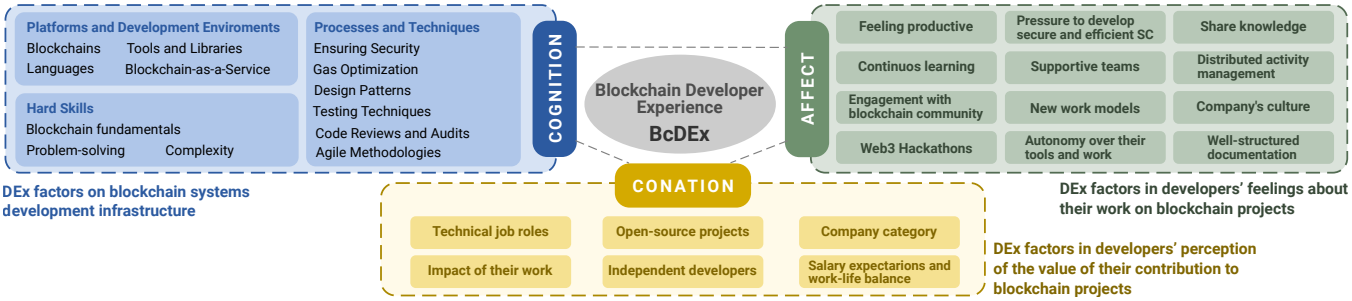

Figure 1: Analysis of DEx dimensions in Blockchain-oriented Software Engineering.

helping developers integrate and communicate with blockchain. Currently, the most popular library for interacting with Ethereum-based networks is Web3.js, with other alternatives such as Ethers.js and Web3.py, as well as libraries for other types of blockchains [51].

Reviewing the literature, we found some studies that compare these tools in technical aspects [17, 51] or use them for the development and testing of specific use cases [16]. However, there is a substantial gap related to more investigation into the impacts of these technologies on the developer's and team's overall experience. Examining whether these tools truly enhance productivity and reduce barriers and integration difficulties using blockchain from the developer's perspective is relevant for improving these tools and making them more accessible and practical for developers.

Despite the availability of these tools for dapp development, challenges in deployment, high operational costs, and a shortage of experienced developers pose barriers to blockchain adoption in the industry [26]. In this sense, ***Blockchain-as-a-Service*** (BaaS) platforms have also emerged to aid the use of blockchain, allowing developers to focus on coding business rules while cloud services deploy, manage and monitor the network infrastructure [31]. Regardless of the potential for improving the efficiency of these platforms, there is a lack of studies evaluating their usage by developers and their impact on DEx.

3.1.2 Techniques and Processes. Regarding the techniques available in SE, we highlight the important role of employing the best design patterns, security, and optimization practices for dapps [33]. Zou et al. [53] conducted an experiment with developers to identify the main challenges in Ethereum development, finding that ***ensuring security*** was one of the most significant challenges. This concern stems from the immutability of the blockchain and the sensitivity of data [33]. Moreover, popular applications of SCs such as tokenization and decentralized exchanges enable peer-to-peer trading of cryptocurrencies, handling substantial financial transactions [47]. Thus, these dapps are targets for malicious users, as evidenced by approximately 8.21 billion in hacks recorded on Defillama[1]. In this context, ensuring reliability in blockchain application development requires the application of SE methodologies [33].

Currently, there are studies proposing and evaluating tools and frameworks aimed at test generation, identification of contract defects, and approaches to verify and validate aspects of blockchain [47]. In the context of this work, some recent studies have investigated developers' perspectives regarding security in the development of dapps and the use of security practices and tools [6, 39], revealing developers' preferences for open-source tools, semi-automated tools to mitigate vulnerabilities, and the time spent by auditors on security audits. The authors also explore the tools and strategies to ensure security and analyze the distinctions between junior and experienced developers. Furthermore, we noticed an opportunity for research to explore sociotechnical issues surrounding developers' trust in the security risks in SCs and how these perceptions impact their development practices.

On the other hand, Zou et al. [53] highlighted the importance of developers paying particular attention to optimization since SCs run based on gas consumption (the measure of computational work required for executing a transaction directly affects transaction costs), unlike traditional software. Therefore, developers must ensure their decisions keep gas consumption levels within acceptable limits [47]. The application of ***gas optimization*** is an essential skill for developers and SC engineers, as even simple changes in the order of state variable declarations can impact gas consumption [27]. These practices demand developers to know new design patterns and new techniques such as i) management of memory slots and types of storage in SCs, ii) blockchain-specific opcodes, and iii) low-level language since developers can embed inline assembly in a language close to that of the EVM called YUL[2]. BcDEx faces a significant challenge in this regard, as some developers do not widely understand these concepts and can introduce security threats and affect optimizations [7].

As previously mentioned, blockchain ***design patterns*** encompass aspects beyond security and gas optimization. Apart from the well-known software patterns from SE, developers must grasp new patterns tailored for dapps. Six et al. [40] present a taxonomy of design patterns, subdividing them into categories: On/off-chain interaction pattern, Smart-contract pattern, Data management pattern, Domain-based pattern, and On-chain pattern—each one of these categories with its subcategories. In addition to the mentioned standards, the blockchain ecosystem, especially Ethereum, deals with Ethereum Improvement Proposals (EIPs)[3], which consist of formal proposals defining new features or potential processes for the Ethereum platform. A new Ethereum Request For Comment (ERC) can be originated through EIP submissions. Subsequently,

[1]https://defillama.com/hacks

[2]https://docs.soliditylang.org/en/latest/yul.html

[3]https://eips.ethereum.org/erc

the community evaluates the documents, and developers create the new ERC once it receives community approval.

These standards are necessary for developing secure, reliable, and easily maintainable dapps. Developers may invest considerable time in absorbing such resources and are likely to specialize according to the demands of their projects. However, the ability to continuously adapt to changes in the community may be a differentiating factor in the BcDEx. Furthermore, engaging in proposing and discussing new standards and ideas can provide developers with new experiences, making them feel part of the evolution of this ecosystem.

Alongside the development cycle, we also need to apply effective ***testing techniques*** to identify vulnerabilities and optimize performance, especially in complex environments such as blockchain [12]. Despite its importance, this phase is often neglected due to development delays, typically conducted only towards the end of the coding phase and before delivery to the client [38]. In addition to existing approaches in SE, different testing approaches have been created or adapted in the literature to meet blockchain applications' needs. Elakaş et al. [12] list Search-Based Testing, Fuzz Testing, Mutation Testing, Model-Based Testing, for example. Some of these approaches have been instantiated in the industry through tools such as Mythril, Oyente, Slither, Securify, and SmartChecks [11].

Fortunately, unlike other factors previously discussed, some studies have dedicated efforts to understand the use of these approaches from the perspective of blockchain developers and testers [39, 53]. As pointed out by Zou et al. [53], over 75% of developers agree that SC has much higher security requirements than traditional software. Security is critical for SCs, and to reinforce security verification practices in SCs, ***code reviews*** and ***audits*** are also essential to ensure code quality. Sharma et al. [39] conducted an experiment involving code review tasks with 29 developers and found that the detection rates of security vulnerabilities were alarmingly low, often below 50%, regardless of the type of vulnerability or the participants' experience in SC development. This result deserves attention, considering the criticality of the SC context. In this sense, it is essential to investigate subjective factors and their impacts on developers' experience when performing code reviews.

Finally, to follow a development process and employ solid SE practices, recent studies have explored using ***agile methodologies*** to manage blockchain-based projects. Marchesi et al. [28] propose 'Agile Block Chain Dapp Engineering' (ABCDE), an agile process for developing dapps covering all phases of the software lifecycle. As stated by Ibba [18], dapps are innovations that require agile development for fast industry release, which ensure frequent deliveries through incremental iterations. Although Marchesi et al. [28] have evaluated their method with 14 blockchain developers, much of the research focuses on applying the method to use cases [18, 28]. In addition, Khalid and Brown [23] have investigated developers' perceptions of the importance of applying SE practices in blockchain development, with agile methods being the most popular (57%). As DEx also encompasses processes, there is still much to be discussed about BcDEx in the different phases of the software development cycle in real-world environments to complement these findings.

3.1.3 Hard Skills. Designing and developing dapps pose complex challenges that require a ***solid understanding of the fundamentals*** underpinning blockchain technology. Blockchain developers face significant challenges and should have experience in several areas of Computer Science, such as Cryptography, Networks and Distributed Systems, Mathematics, Consensus Algorithms, Cloud Infrastructure, Security and Privacy, and Game Theory, which are essential pillars that enable these professionals to explore opportunities fully [1]. According to Gartner [14], 23% of surveyed CIOs stated that blockchain requires the most extensive set of new skills for implementation compared to other technology areas. At the same time, 18% indicated that blockchain skills are the most challenging to find. In addition, ***problem-solving skills*** should also encompass the business domains of dapps by addressing challenges in sectors such as healthcare, supply chain, and finance, with the latter being particularly demanding in terms of ***complexity***.

Moreover, a relevant discussion in the community revolves around the requirements for developers to enter the era of Web3 [35]. Although it is possible to learn Web3 tools directly, developers should be familiar with and have a solid foundation in Web2 technologies. Kassab et al. [22] revealed that employers value professionals with a broad understanding of the software development process. One of the results indicated that Java leads with 34.5% of the ads, while Solidity, the primary language used for SC development, ranks 12th, with 5.4%. This trend indicates the importance of a seamless transition to Web3 paradigms, where an adequate developer's progression in Web3 depends on a medium to a high level of proficiency in Web2 technologies. Thus, there is an opportunity to explore the impacts on aspects of DEx and the careers of those who have chosen to transition from Web2 to Web3.

3.2 [Affect] DEx factors in developers' feelings about their work on blockchain projects

> *"The **affective dimension** consists of factors that influence how developers feel about their work. Respect and belonging are social factors that work to create a feeling of security. Attachment to persons, teams, or even habits of work also belong to this dimension. Positive feelings in general can be an important factor in good DEx." [13]*

Kassab et al. [22] demonstrated that approximately 31.3% of job postings regarding the blockchain ecosystem requested at least one interpersonal skill. Among them, achievement, creativity, enterprise skills, life-long learning, and outcome-oriented skills were categorized as "workplace productivity skills," accounting for 0.2% over ten categories. Indeed, ***feeling productive*** is one of the relevant aspects to explore related to the feelings about the work of blockchain developers as it can directly impact the developers' satisfaction. Navigating the complexities of this field also requires a commitment to increasing efficiency and productivity [30]. The feeling of unproductivity can cause frustration for developers, impacting aspects of their development experience.

Joining in activities for knowledge and collaboration can be an advantage for this professional. The developers need to have a ***continuous learning*** as blockchain technology is growing fast. The learning curve is particularly challenging for entry-level developers, who may find obstacles when solving specific problems, affecting project efficiency and deadlines. ***Engagement with the blockchain community*** is also relevant for their development and experience. Today, blockchain companies have used forums,

social media groups, and Discord servers (or alternatives) to share knowledge and promote specific ecosystems within Web3. In addition, ***Web3 hackathons*** has also been widely disseminated by blockchain platforms[4][5][6] to promote immersive learning and new experiences for developers and organizations.

Another significant factor is the ***pressure to develop secure and efficient SCs***, as the developers may work on code for high-profile clients or decentralized finance protocols that manage substantial sums of money. These circumstances raise considerations about subjective aspects intrinsic to the developer's experience in developing dapps, which have not been explored in the literature to date. However, their findings could provide valuable insights for blockchain companies, enabling them to offer essential strategies to improve the experience of their blockchain developers and achieve excellent and secure outcomes in their projects. In addition, other factors may influence developers' behavior, such as technical activities and the social and environmental context, including team dynamics and available resources. In this sense, ***supportive teams*** and practical guidance from senior developers and leaders positively impact the experience of novice blockchain developers.

Additionally, companies have adopted remote or hybrid work models, highlighting the need for effective technological solutions to facilitate communication and collaboration among distributed teams. In our context, blockchain characteristics reshape team structures, operations, and global collaboration through decentralization and peer-to-peer interactions [15]. The distributed employees of companies such as Binance illustrate the dynamism of blockchain, allowing them to capture local trends and provide rapid responses to global customers [21]. Decentralized Autonomous Organizations (DAOs) enable distributed teams to govern, coordinate, and incentivize collaboration without hierarchies or traditional intermediaries [15]. These blockchain companies or organizations influence ***new work models*** that can impact the DEx directly or indirectly.

Thus, blockchain developers gain ***autonomy over their tools and work*** and can interact and ***share knowledge*** with global and diverse talents. Since DEx is not limited to technical aspects but also encompasses social aspects, these professionals also have the opportunity to create informal moments, such as virtual coffee breaks or games [15]. On the other hand, challenges related to ***distributed activity management*** still require efficient solutions to allow development teams to overcome the communication and collaboration barriers inherent to remote teams in general.

In addition to an adequate adaptation to the ***company's culture***, it is essential that the resources and materials produced by the team are accessible and easily understandable to enrich the DEx. ***Well-structured and informative documentation*** simplify the work progress for team members and the developers' community that will use the service or product. Web3 initiatives allocate considerable resources to bolster certain job functions typically undervalued in SE, notably Developer Relations (DevRel) and Technical Writing. Given the frequent releases of protocols, blockchains, tokens, and cryptocurrencies, these endeavors actively organize events and craft essential documentation to support and integrate blockchain developers into their technologies.

[4]https://chain.link/hackathon
[5]https://www.celocamp.com/
[6]https://solana.com/news/tag/hackathon

3.3 [Conation] DEx factors in developers' perception of the value of their contribution to blockchain projects

> *"The **conation dimension** consists of factors that affect how developers see the value of their contribution. Intentional, planned activity with personal goals that are properly aligned with the goals of others is likely to increase the sense of purpose, motivation, and commitment, and thus positively affect DEx." [13]*

Blockchain developers can create value primarily through their contributions by applying their knowledge, skills, responsibilities, and autonomy. According to recent studies [1, 22], the development of dapps has generated different professional profiles alongside those already existing in traditional software companies. These new ***technical job roles*** influence the effective organization of the team and the value contributed by each member. For example, the SC engineer develops SCs using specific languages and security techniques, while the blockchain architect designs solutions, data security, and cloud infrastructure. In turn, blockchain developers and engineers play more comprehensive roles, including Web2 backend and frontend programming and the aforementioned skills. Roles such as blockchain protocol engineer and researcher usually focus on low-level topics, such as new cryptography, protocol privacy solutions, and scalability, essential for technological advancement. Blockchain developer advocates and technical writers also disseminate knowledge to the developer community.

These responsibilities impact in *motivation* and *commitment* of the professionals involved in creating blockchain-based software, ensuring both quality and functionality and integrating their distinct layers effectively. In this regard, there is a space to assess the value of these contributions in the development of dapps, investigating developers' perceptions at both individual and collaborative levels within the software development cycle and evaluating how they recognize the ***impact of their work*** on the project, the company, or the blockchain ecosystem as a whole.

Factors such as the characteristics of the projects or the environment can also influence the developer's contribution. According to Raval [34], dapps should ideally be full ***open-source*** and function autonomously, without a central authority monopolizing the network, while enabling third-party verification. These applications predominantly leverage public blockchains, implement open-source software protocols, and can define formal procedures for submitting and managing collaborative software enhancement proposals. In this context, ***independent developers*** contributing to open-source projects should adhere to new principles and guidelines, including governance. As stated by Song [42], public blockchain networks and dapps governance involves the protocol rules governing participants' interactions, software development and updates, and managing rights and relationships within the stakeholder community. This developer can contribute more effectively to the evolution of blockchain protocols by adhering to formal guidelines for submitting and managing collaborative software improvement proposals.

Another aspect to consider is the ***company category***, ranging from startups to large corporations. Due to the field's growth, the number of blockchain startups created in recent years has increased significantly. Amadeo [2] identified 1,410 blockchain startups in

the Crunchbase database, 518 of which were established in 2021 alone. However, it is essential to note that startups may not offer as reliable employment structures as established companies, especially in the information technology sector, where the success rate of startups is below 50% [46]. According to data from Startuptalky [43], the failure rate of blockchain startups is higher than the average, reaching 95%, with an average lifespan of only one year. These companies typically face challenges such as intense competition, limited regulation, insufficient financial resources, incomplete business plans, and teams lacking in knowledge, qualifications, or experience. These uncertainties for professionals impact their DEx.

In contrast, conventional companies have established reputations and extensive networks of business relationships with clients and partners. However, breaking their successful business models to innovate can take much work, risking cannibalizing revenues from existing products and services [25]. Thus, establishing blockchain teams within these organizations and fostering interaction and collaboration with other technical teams is challenging. In both categories, these peculiarities can also influence the developer's *motivation*, their *plans*, and *alignment* with the company.

Additionally, other company characteristics, such as ***salary expectation*** and ***work-life balance***, can influence aspects like developers' plans, commitment, and alignment. Due to the growing blockchain industry and the shortage of qualified talent, salaries tend to be high. In addition, a peculiar aspect of token projects is incentive strategies in their token distribution models, which can motivate developers or network validators while ensuring alignment with the project's long-term goals [45]. Work-life balance in the blockchain industry can be challenging due to its fast-paced and evolving nature. It involves intensive project sprints, critical network launches, tight deadlines, and security incidents, often resulting in extended working hours [44].

4 NEW DIRECTIONS AND CONCLUSION

Based on the previous discussion, this section aims to present new research directions and opportunities in the context of BcDEx, as summarized in Figure 2.

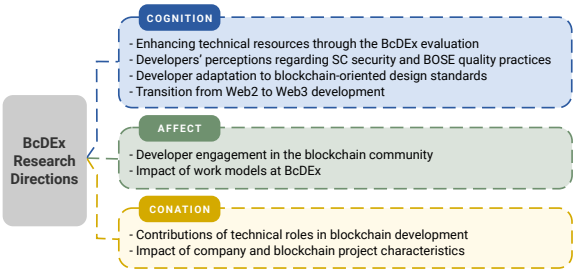

Figure 2: Research directions for BcDEx.

- **Enhancing technical resources through the BcDEx evaluation:** Researches on the impacts of using the latest SC languages, Web3 development environments, native libraries, and platforms BaaS provides analysis of aspects such as learning curve, as well as compare efficiency, productivity, and ease of integration with blockchain networks from a developer's perspective. These studies would contribute to optimizing blockchain development and integration.
- **Developers' perceptions regarding SC security and BOSE quality practices:** The analysis of subjective factors related to developers' perception of security risks in SC development and working in critical systems to provide insights into how these aspects influence their development practices and the adoption of quality standards.
- **Developer adaptation to blockchain-oriented design standards:** Evaluating developers' understanding and implementation of new blockchain design standards and collaboration in EIPs and ERCs to explore how the new standards shape their development experience and foster a sense of community within the blockchain ecosystem.
- **Transition from Web2 to Web3 development:** Exploring the experiences of software engineers transitioning from Web2 to Web3 to identify the skills and knowledge required to effectively adapt to the demands of Web3 technologies.
- **Developer engagement in the blockchain ecosystem and community:** We may investigate how participation in forums, social media groups, Discord servers, and Web3 hackathons impacts developers' learning curve and progression. This opportunity focuses on how these collaborative resources help overcome initial coding challenges and strengthen developers' integration and experience.
- **Impact of work models at BcDEx:** Investigate the impact of work models (remote, hybrid, or in-person) and the challenges related to managing distributed activities, focusing on how decentralization and peer-to-peer interactions are redefining blockchain teamwork and global collaboration.
- **Contributions of technical roles in blockchain development:** Assessing the contributions of technical roles in blockchain development by investigating their impact on team organization, productivity, and project outcomes, and exploring developers' perceptions of their individual and collaborative contributions in the blockchain development.
- **Impact of company and blockchain project characteristics:** Investigating how the company category (startups, small, and large corporations) and project types (private and open-source) influence DEx, considering factors such as job stability, innovation, collaboration opportunities, salary expectations aligned with plans, and developer commitment.

In conclusion, we identify the complexities and challenges intrinsic to BOSE, emphasizing their impact on DEx dimensions related to cognition, affect, and conation. For this purpose, we categorized the factors that shape BcDEx into a solid conceptual framework, enabling a comprehensive understanding of critical aspects. Additionally, we propose new research directions to provide an initial foundation regarding DEx to support research and industry related to BOSE, addressing both technical and social aspects. While factors related to Conation and Affect are more generic, they were specifically discussed in the context of the blockchain domain. In addition, we noticed that the factors related to Cognition are discussed more in the literature. For future work, we plan to propose a conceptual catalog considering DEx aspects to guide blockchain developers in developing decentralized applications.

REFERENCES

[1] ACQUIN. 2023. Blueprint of a VET qualification on Blockchain and Distributed Ledger Technologies.

[2] Tommaso Amadeo. 2021. Blockchain and distributed ledger technologies in business innovation: an analysis of the international blockchain startup ecosystem. (2021).

[3] Imran Bashir. 2017. *Mastering blockchain*. Packt Publishing Ltd, Birmingham, United Kingdom. 540 pages.

[4] R. Beck, M. Avital, M. Rossi, and J. B. Thatcher. 2017. Blockchain technology in business and information systems research. *Business & Information Systems Engineering* (nov. 2017), 381–384. https://doi.org/10.1007/s12599-017-0505-1

[5] Marianna Belotti, Nikola Božić, Guy Pujolle, and Stefano Secci. 2019. A vademecum on blockchain technologies: When, which, and how. *IEEE Communications Surveys & Tutorials* 21, 4 (2019), 3796–3838.

[6] Stefanos Chaliasos, Marcos Antonios Charalambous, Liyi Zhou, Rafaila Galanopoulou, Arthur Gervais, Dimitris Mitropoulos, and Benjamin Livshits. 2024. Smart Contract and DeFi Security Tools: Do They Meet the Needs of Practitioners?. In *Proceedings of the 46th IEEE/ACM International Conference on Software Engineering*. 1–13.

[7] Stefanos Chaliasos, Arthur Gervais, and Benjamin Livshits. 2022. A study of inline assembly in solidity smart contracts. *Proceedings of the ACM on Programming Languages* 6, OOPSLA2 (2022), 1123–1149.

[8] Vivek Chaurasia and Manoj Kamber. 2023. Unleashing blockchain magic: a comparative journey through developer ecosystems and tools in Ethereum, Polygon, and Polkadot. (2023).

[9] Saori Costa, Matheus Paixao, Igor Steinmacher, Pamella Soares, Allysson Allex Araújo, and Jerffeson Sousa. 2024. Sociotechnical Dynamics in Open Source Smart Contract Repositories: An Exploratory Data Analysis of Curated High Market Value Projects. In *Proceedings of the 20th International Conference on Predictive Models and Data Analytics in Software Engineering*. 22–31.

[10] Giuseppe Destefanis, Michele Marchesi, Marco Ortu, Roberto Tonelli, Andrea Bracciali, and Robert Hierons. 2018. Smart contracts vulnerabilities: a call for blockchain software engineering?. In *2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE)*. IEEE, 19–25.

[11] Monika Di Angelo and Gernot Salzer. 2019. A survey of tools for analyzing ethereum smart contracts. In *2019 IEEE international conference on decentralized applications and infrastructures (DAPPCON)*. IEEE, 69–78.

[12] Anıl Elakaş, Hasan Sözer, Ilgın Şafak, and Kübra Kalkan. 2024. A systematic mapping on software testing for blockchains. *Cluster Computing* (2024), 1–16.

[13] Fabian Fagerholm and Jürgen Münch. 2012. Developer experience: Concept and definition. In *2012 international conference on software and system process (ICSSP)*. IEEE, 73–77.

[14] Gartner. 2024. Gartner survey reveals the scarcity of current blockchain deployments. https://www.gartner.com/en/newsroom/press-releases/2018-05-03-gartner-survey-reveals-the-scarcity-of-current-blockchain-developments.

[15] Prince Harrison. 2024. Remote Work in Web3: How Distributed Teams Are Driving Innovation. https://medium.com/@chibyke.prince2017/remote-work-in-web3-how-distributed-teams-are-driving-innovation-b2e31bb6dcbd.

[16] Atharva Hire, Harsh Lanjewar, Prathamesh Haridas, Makarand Jadhav, and Milind Rane. 2023. Decentralized Lottery Using Blockchain. In *2023 3rd International Conference on Pervasive Computing and Social Networking*. IEEE, 1035–1041.

[17] Samu Honkamäki. 2022. *Evaluating Ethereum development environments*. Master's thesis. S. Honkamäki.

[18] Simona Ibba. 2019. Agile methodologies and blockchain development. (2019).

[19] Cosmin Irimia, Luciana Bejan, and Adrian Iftene. 2023. BlockchainPedia: A Comprehensive Framework for Blockchain Network Comparison. *Procedia Computer Science* 225 (2023), 2302–2311.

[20] Mudabbir Kaleem, Anastasia Mavridou, and Aron Laszka. 2020. Vyper: A security comparison with solidity based on common vulnerabilities. In *2nd conference on blockchain research & applications for innovative networks and services*. IEEE, 107–111.

[21] Natalia Karayaneva. 2019. The Growing Trend in Workspaces: Distributed Remote Teams.

[22] Mohamad Kassab, Giuseppe Destefanis, Joanna DeFranco, and Prince Pranav. 2021. Blockchain-engineers wanted: an empirical analysis on required skills, education and experience. In *2021 IEEE/ACM 4th International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB)*. IEEE, 49–55.

[23] Shawal Khalid and Chris Brown. 2023. Software engineering approaches adopted by blockchain developers. In *2023 Tenth International Conference on Software Defined Systems (SDS)*. IEEE, 1–6.

[24] Raj Kumar Reddy Kotha and Micheal Sony. 2023. Complexity and ambiguity for blockchain adoption in supply chain management. In *Blockchain in a Volatile-Uncertain-Complex-Ambiguous World*. Elsevier, 29–41.

[25] Mary C Lacity. 2018. Addressing key challenges to making enterprise blockchain applications a reality. *MIS Quarterly Executive* 17, 3 (2018), 201–222.

[26] Xiaoyun Li, Zibin Zheng, and Hong-Ning Dai. 2021. When services computing meets blockchain: Challenges and opportunities. *J. Parallel and Distrib. Comput.* 150 (2021), 1–14.

[27] Lodovica Marchesi, Michele Marchesi, Giuseppe Destefanis, Giulio Barabino, and Danilo Tigano. 2020. Design patterns for gas optimization in ethereum. In *2020 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE)*. IEEE, 9–15.

[28] Lodovica Marchesi, Michele Marchesi, and Roberto Tonelli. 2020. ABCDE—Agile block chain DApp engineering. *Blockchain: Research and Applications* 1, 1-2 (2020), 100002.

[29] Anders Nylund et al. 2020. A multivocal literature review on developer experience. (2020).

[30] Jeffery Oke-Samuel. 2024. How to Boost Efficiency and Productivity as a Blockchain Engineer. https://medium.com/@jefferyokesamuel1/how-to-boost-efficiency-and-productivity-as-a-blockchain-engineer-6421f6f719ee.

[31] Md Mehedi Hassan Onik and Mahdi H Miraz. 2019. Performance analytical comparison of blockchain-as-a-service (baas) platforms. In *International Conference for Emerging Technologies in Computing*. Springer, 3–18.

[32] Reza M Parizi, Amritraj, and Ali Dehghantanha. 2018. Smart contract programming languages on blockchains: An empirical evaluation of usability and security. In *Blockchain–ICBC 2018: First International Conference, Held as Part of the Services Conference Federation*. Springer, 75–91.

[33] Simone Porru, Andrea Pinna, Michele Marchesi, and Roberto Tonelli. 2017. Blockchain-oriented software engineering: challenges and new directions. In *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*. IEEE, 169–171.

[34] Siraj Raval. 2016. *Decentralized applications: harnessing Bitcoin's blockchain technology*. " O'Reilly Media, Inc.".

[35] Ali Raza. 2024. Why You Should Learn Web 2.0 Before Getting into Web 3.0. https://javascript.plainenglish.io/why-you-should-learn-web-2-0-before-getting-into-web-3-0-d251625990db.

[36] Alan Rodrigues, Allysson Allex Araújo, Matheus Paixao, and Pamella Soares. 2022. Aspectos Colaborativos no Desenvolvimento de Contratos Inteligentes da Plataforma Ethereum: Um Estudo Exploratório-descritivo Preliminar. In *VII Workshop sobre Aspectos Sociais, Humanos e Econômicos de Software*. SBC, 56–60.

[37] Vincenzo Sagristano. 2023. *Smart Contract Developer Experience Enhancement*. Ph.D. Dissertation. Politecnico di Torino.

[38] Nicolás Sánchez-Gómez, Jesus Torres-Valderrama, Julián Alberto García-García, Javier J Gutiérrez, and MJ Escalona. 2020. Model-based software design and testing in blockchain smart contracts: A systematic literature review. *IEEE Access* 8 (2020), 164556–164569.

[39] Tanusree Sharma, Zhixuan Zhou, Andrew Miller, and Yang Wang. 2023. A {Mixed-Methods} study of security practices of smart contract developers. In *32nd USENIX Security Symposium (USENIX Security 23)*. 2545–2562.

[40] Nicolas Six, Nicolas Herbaut, and Camille Salinesi. 2022. Blockchain software patterns for the design of decentralized applications: A systematic literature review. *Blockchain: Research and Applications* 3, 2 (2022), 100061.

[41] Jie Song, Pengyi Zhang, Mohammed Alkubati, Yubin Bao, and Ge Yu. 2022. Research advances on blockchain-as-a-service: Architectures, applications and challenges. *Digital Communications and Networks* 8, 4 (2022), 466–475.

[42] Mario Song. 2020. FOSS Governance and Blockchain Networks.

[43] Startuptalky. 2023. Startup Failure and Success Rates: 2023 Research. https://startuptalky.com/startup-failure-success-rates-statistics.

[44] Teal. 2024. Blockchain Developer Work-Life Balance. https://www.tealhq.com/work-life-balance/blockchain-developer.

[45] TokenMinds. 2024. Token distribution ultimate guide - Maximizing your crypto project's success in 2024.

[46] Upsilon. 2024. Startups' success and failure rate in 2024. https://www.upsilonit.com/blog/startup-success-and-failure-rate.

[47] Anna Vacca, Andrea Di Sorbo, Corrado A Visaggio, and Gerardo Canfora. 2021. A systematic literature review of blockchain and smart contract development: Techniques, tools, and open challenges. *Journal of Systems and Software* 174 (2021), 110891.

[48] Rijnard van Tonder. 2024. Verifying and Displaying Move Smart Contract Source Code for the Sui Blockchain. In *Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings*. 26–29.

[49] Ammar Voloder and Monika di Angelo. 2023. Comparison of Smart Contract Platforms from the Perspective of Developers. In *International Conference on Blockchain*. Springer, 104–118.

[50] Karl Wüst and Arthur Gervais. 2018. Do you need a blockchain?. In *2018 crypto valley conference on blockchain technology (CVCBT)*. IEEE, 45–54.

[51] Jaka Zavratnik. 2022. *Analysis of web3 solution development principles*. Master's thesis. Universitat Politècnica de Catalunya.

[52] Zibin Zheng, Jianzhong Su, Jiachi Chen, David Lo, Zhijie Zhong, and Mingxi Ye. 2024. Dappscan: building large-scale datasets for smart contract weaknesses in dapp projects. *IEEE Transactions on Software Engineering* (2024).

[53] Weiqin Zou, David Lo, Pavneet Singh Kochhar, Xuan-Bach Dinh Le, Xin Xia, Yang Feng, Zhenyu Chen, and Baowen Xu. 2019. Smart contract development: Challenges and opportunities. *IEEE transactions on software engineering* 47, 10 (2019), 2084–2106.