Current Journal of Applied Science and Technology

39(14): 125-134, 2020; Article no.CJAST.57799
ISSN: 2457-1024
(Past name: British Journal of Applied Science & Technology, Past ISSN: 2231-0843, NLM ID: 101664541)

Modeling Distributed Agile Software Development for Big Data Projects: Evolution in Process

Ali Sever[1*]

[1]*Department of Computer Information Systems, Pfeiffer University, Charlotte, NC, 28109, USA.*

Author's contribution

The sole author designed, analysed, interpreted and prepared the manuscript.

Article Information

DOI:10.9734/CJAST/2020/v39i1430710
Editor(s):
(1) Prof. Samir Kumar Bandyopadhyay, University of Calcutta, India.
Reviewers:
(1) Suman Madan, Jagan Institute of Management Studies, India.
(2) Luz Angela Martinez Martinez, Universidad Autónoma de Occidente, Colombia.
Complete Peer review History: http://www.sdiarticle4.com/review-history/57799

Original Research Article

Received 28 March 2020
Accepted 03 June 2020
Published 16 June 2020

ABSTRACT

Although big data has been around for ages, finding the right development method for the specific domain of big data has always been challenging. Many companies are taking maximum benefits from the large amount of data that is available to them. However, in order to make use of this large amount of data, developing and maintaining a dependable and robust software system is a major problem today. We proposed an enriched nonlinear distributed agile development model in big data applications. The model makes it possible to overcome the difficulties of traditional software process models by pairing up evolving technology of big data and distributed agile methodology. In the paper we, first, present arguments behind the multi-agent model. Next, it is shown how it may help improve the interaction between big data and software development project life cycle. Finally, we suggest how the proposed model can be tested experimentally to show how devising a multi-agent computational system may offer an efficient way of monitoring, managing, and deploying software products in big data applications.

Keywords: Big data analytics; distributed agile software development; agile methodologies; collaborative cloud computing; multi-agent systems.

**Corresponding author: E-mail: delafener@gmail.com;*

1. INTRODUCTION

Many recent studies found that the failure rate of big data projects is as high as 85% and it will not get better soon: “Through 2022, only 20% of analytic insights will deliver business outcomes” [1]. Big Data technology is the focus of attention for almost every major technology company today due to the immense potential it promises of generating insights and results that were previously deemed too difficult to generate with traditional linear processes [2,3,4]. The emergence of the internet provide the users an easy way to perform any analysis on software applications, which are used in everyday life.

There is an intrinsic tie between Big Data applications and distributed agile software development project life cycle (SDPLC). Big data is a revolutionary innovation in the digital era in the field of Information and communication technology [5,6,7,8]. Several researchers are focusing on big data implementation in extracting maximum and critical data from the data available that would help in making key business decisions in gaining competitive advantage and by adding value in many sectors. Despite many advances in big data applications, there has been ongoing research in enhancing the application for stability and scalability. In a holistic approach, how distributed teams can be incorporated in distributed agile software system development was proposed in [9].

Some of the questions have not get enough attention such as how big data is being used today, how distributed agile SDLC can help development of big data projects, and how these two together makes solution much more timely and effective. Mainly, how these two pair up. The Internet changed everything by generating information in volume that we have never seen before. The changing face of data, spreadsheets to dashboards and advance analytic tools to Hadoop, requires a change in the SDPLC.

Big Data has become one of the most important concepts that is being studied in Software Engineering. The data produced in recent years has increased rapidly and exponentially that most of the researchers and major corporations are mainly concentrating on extracting maximum value from the data produced through big data applications. This paper focuses on the study of existing research in exploring distributed agile SDPLC. Although big data has been around for ages, finding the right development method for the specific domain of big data has always been challenging [10].

Big Data has become the trending technology and it is possible to extract much data from the information all around the world. However, until date, it has not been possible to develop a stable big data application. Researchers have been aiming to build a good big data application by refining the Software Development Life Cycle. Many major big companies have started using big data as their storage and their success stories have urged other organizations to start developing a big data application of their own [11,12]. Enterprise resource applications have been making good use of big data to enhance customer service.

As big data is becoming more and more prominent in the information technology industry, many organizations making use of it to extract the most from the vast amounts of data to benefit them. As this involves tremendous quantities of data to be handled, developing and maintaining the big data applications for the continuously increasing data becomes tedious and challenging. The main objective of this paper is to consider the existing research on phases of Software Development Project Life Cycle (SDPLC) and utilize it to create better big data applications with modified SDPLC. The results help in analyzing the gaps and make the distributed agile SDPLC fit well for development of big data applications (Fig. 1).

Big data is a hot topic in current technology research, articles, and fields of work. Industries are using various technologies that gather large amounts of data about everyone around the world. Yet, there is a shared common issue in regards to big data; how can people apply software engineering phases to such projects of work to receive the greatest value from the data. It seems that a vast majority of the research from Information Technology applications, not providing much information in regards to other fields. Fields such as banking, healthcare, and transportation provide many data that can be mined to establish foundational software engineering applications for big data [13,14]. Moreover, looking at the software design project lifecycle as a whole and applying it to big data from outside the technology realm would provide a better view on how agile software engineering phases to big data projects [15].

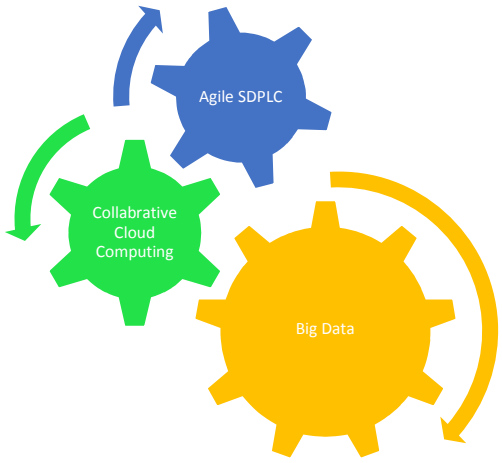

Fig. 1. Distributed Agile SDPLC for Big Data Applications

In terms of flexibility and architecture compatibility, architecture with one specific programming paradigm is not flexible since it may not be compatible with any changes on existing legacy services or new processing services. A rule based modular architecture is an ideal solution due to the control of execution on multiple platforms and its capability of executing wide range of processing services is detailed in [16].

Big data is being extensively and successfully used in social networking sites like Twitter, Linked In, and Facebook in monitoring the network traffic. The successful implementation of technology independent reference architecture for big data systems based on the published implemented architecture of big data use cases has been detailed clearly in [17,18,19]. The architecture was designed with inductive reasoning as per the empirical data from the big data use cases published. Functional components, data stores and connecting data stores, are designed in the reference architecture in construction of big data systems.

The goal of this study is, first, to show how the existing software engineering processes worksin the context of big data applications, then, explore the intrinsic tie between Big Data applications and distributed agile software development project life cycle (SDPLC) as a new model.

The remaining sections in this paper are structured as follows: Section 2 addresses the issues related to big data challenges, why existing process model is failing. Section 3 provides the proposed model and its interworking. We explain how it could help to overcome the challenges presented in the previous section. Section 4 depicts the results and discussions under the lens of proposed model. Last section, Section 5, provides what has been achieved and future enhancement. It concludes with the need of it.

2. BIG DATA CHALLENGES

Building software systems capable of operating successfully in big-data environments presents many challenges. Therefore, SDPLC must constantly be updated due to the changing needs of people, business models, and technological advances. The design of applications need to accustom the changing needs and alter their architectural process. In the era of big data, decades old technologies and architectures being used are not able to handle these streams and introduced the need for new processes and tools. The entire procedure of creating software as described in Fig. 2 when applied to big data projects are loaded with issues. There have been a few examples of overcoming adversity of Big Data being utilized by innovative companies to rule their rivals in regions, for example, web based life, web crawlers, and web based business and video gushing administrations. The achievement of these use cases have triggered the enthusiasm within various organizations to renew interest in big data applications. There is a big gap between the software engineering phases and their applicability to big data applications. First, it is important to understand the limitations of current

System Life Cycle Processes listed in Fig. 2, then, how appropriate software processes would make it possible to build desired software. Therefore, it is important to study present implementations of big data to understand where software companies currently stand. Currently, the most popular areas for big data projects are in information technology, healthcare, and banking [20]. These data-rich areas are prime examples of how software projects can benefit from big data, but they represent a small percentage of the possible scope of big data projects. Software engineers still face problems with validity, confidentiality, scalability, and analytical mechanisms in big data projects [21,22]. Therefore, the goal is to explore the untapped potential that exists for software engineering projects in the realm of big data analytics in regards to the modified software development project life cycle.

Nowadays all the big organizations are taking advantage of big data to analyze existing data to make user experience better and beat their competitors. These organizations are distributed across domains like social media, e-commerce etc. Due to this growing popularity, many organizations seek to build big data applications to benefit from it. Big data is now being used in many sectors like trading, tourism, healthcare etc. Movie recommendations, product recommendations and other such suggestions are all a result of big data.

A relation between Big Data and Software Engineering must be uncovered, next. Big data has four Vs, which include the following:Volume that is the increasing data and collected data over the last decade, Velocity is the requirement to process real time data without failure. Variety is related to handling the varied types of data format that exists, Veracity implies preprocessing the data to make it useful for the application. Variety implies data is restricted to what you want or completely not in shape; you want it and having lots of ambiguity. For example, the data may be text or some video. Velocity implies the data needs immediate attention and needs to be taken care of in situations of failure. Veracity implies that data has been modified or pre-processed to make efficient use of it. Volume is dedicated to the amount of data being generated every single day from the numerous sources available in the world today. Every day, terabytes of data are being generated from different sources from the world.

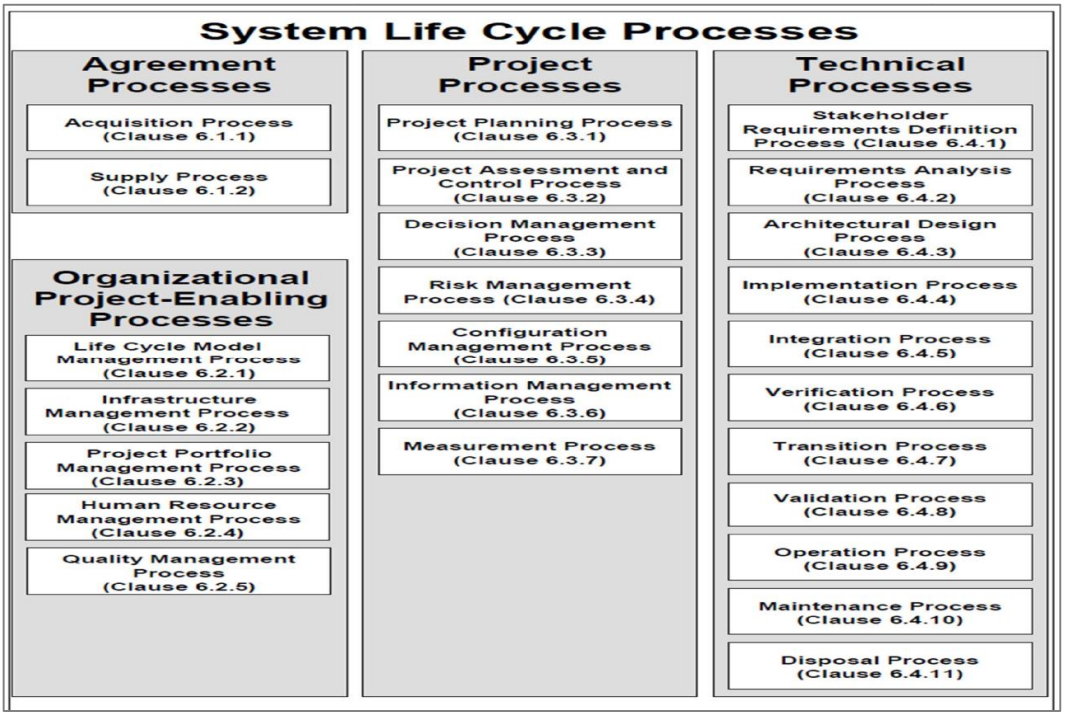

Fig. 2. ISO/IEC standard 15288:2008—Systems and Software Engineering—System Life Cycle Processes [2]

Software development already has many challenges like problems between stakeholders, constantly changing requirements, environmental issues and so on. Along with these challenges, fulfilling the 4Vs for development of big data application becomes even more challenging. Even though these problems exist the phases of SDPLC can be used to maximum advantage in building big data applications [23,24,25,26]. Although there have been many reviews on big data there is none that reviews existing software engineering so as to utilize it for development of big data applications.

Many Technological companies are now using Big Data to survive and win in this competitive world like Facebook, Google, Netflix, etc., This has created curiosity and interest in developing Big Data applications with traditional software development life cycle so that they can acquire maximum benefit from it. Only a few big and important successful leaders in the big data field include Facebook, Twitter, Google, Amazon and Netflix. Big data enhances the customer experiences and would bring profits to stakeholders. Only a few services for customer experience enhancement using big data include Netflix's "Movie Recommendations", Facebook's and LinkedIn's "People you may know", and "Frequently bought together" by Amazon. The current software development deals with few challenges with errors due to requirement changes, communication gaps between stakeholders in an organization. Despite of the complexities involved in implementing the big data in traditional Software Development Life Cycle (SDLC), a robust and scalable big data application would overcome many of the challenges and would bring best practices of software engineering in building reliable fault tolerant systems.

Many errors come while developing a software application with traditional SDPLC due to many factors. This is true for any size application. These issues may increase more when the four V's are considered during the application development. Despite all the issues and difficulties faced, all the phases of SDPLC must be leveraged to building Big Data applications and projects in a scalable and robust manner. The traditional software development life cycle is full of errors and this is primarily due to the changing requirements every day.

The big data movement has energized the software architecture world, and it has introduced complex, interesting questions for the community [27,28,29]. Big data is an evolving field and has proven its ability in improving the competitive advantage in several industries and Government sectors. The fundamental characteristics of big data rely on Vs that include Velocity, Variety and Volume. Data that is growing and rapidly changing would come under Velocity and Variety characteristic is the data that is available in different and multiple formats, and Volume characteristic is the large/huge amount of data being generated every seconds.

Formal evaluation of the proposed integration platform, like performing several stress test cases should be performed to test the performance of the big data systems in the case of excessive high data loads. Moreover, there are several challenges in integrating multiple systems while constructing large-scale infrastructure, data integration and semantic processing within the context of a platform for Big Data analytics [30].

3. DESCRIPTION OF PROPOSED MODEL

Any successful software development process models must be adaptable to various strategies to cope with all the issues and challenges associated to big data applications. Such a model adapts to a new environment with minimal involvement of human expertise. The proposed architecture is built around the agile manifesto philosophy, which emphasizes on individuals, rather than processes, working product over its extensive documentation, customer collaboration rather than a contractual relationship, and response to change over plan-based systematic development. To pair up distributed agile methodology with big data, we make the following observations in the first phase of the proposed model: First, the 4V's have always been a major concern in software development in Big Data. There are numerous domains of big data and each one of them sometimes requires a different development method. Through this modelling, we have tried to find the most comprehensive development phases that define a framework for big data software development to provide a potential solution to four Vs and figure out the domains to implement these solutions in. Next, we tried to explore many phases of big data, specifically, targeted design, architecture and testing phases. These phases are the most common ones and are essential for the smooth functioning of big data applications. Last, the traditional development and testing strategies are obsolete and cannot cope up with the massive requirements of big data

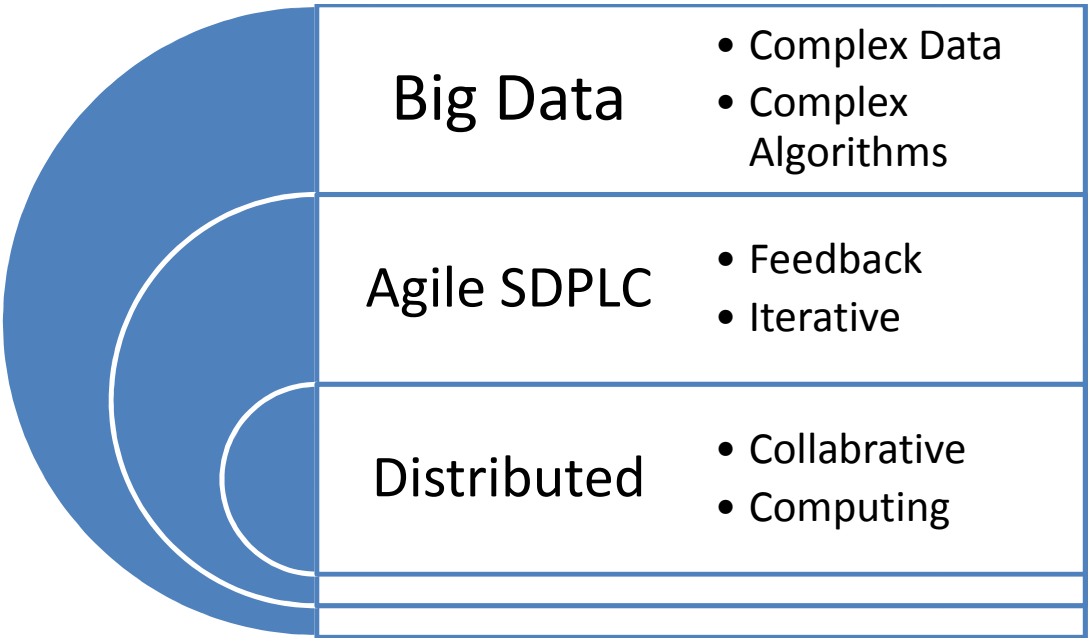

Fig. 3. Features of Big Data, Agile SDPLC, distributed development

applications. The main challenge in a big data application is the nature of the data; it can be both semi-structured and unstructured. Apart from this, another aspect of a big data application is the infrastructure of distributed parallel processing. It needs to be robust enough to provide the strength to run complex queries at a limited time. When it comes to big data testing, the tools are not yet matured enough. Not only this, the development tools are themselves at a very early stage.

The system development environment in big data projects are complex and rapidly changing. As stated in [30], "the models that monitor that develop system development process are rarely accurate and local monitoring models should be used in dynamic and uncertain environments". We suggest using an architecture composed of four interdependent cooperating multi-agent systems capable of monitoring a complex environment and adapting to its changing structures. This architecture makes it possible to divide the monitoring system into four subsystems and reliably monitor the processes with the most relevant challenges and/or issues in each subsystem.

Next phase is the implementation of the proposed model by tying the features of a big data application back to agile manifesto by adapting the proposed distributed agile SDPLC system. Fig. 3 shows the general principals of the proposed approach. It consists of four main agents. Each agent as a sub software system, adopting to changing conditions, is based on the four pillars of agile development. This is a multi-agent system each one of the agents is adaptive, cooperative, and autonomous. For effective practical application of the proposed multi-agent solution, it is necessary to tackle series of steps, each one implemented by a single agent, the results of which are given below.

i. How responding to change and interactions are associated with velocity, which refers to the speed at which vast amounts of data are, being generated, collected and analyzed. Big data technology analyzes the data while it is being created, without the use of databases. The changing requirements of the software and the communication gap between stakeholders makes the software developing task complex. In addition, if the developing application is integrated with the four V's of big data, the complexity level of the project increases. However, this may be overcome by adopting to changes, one of the main features of agile methodology, and making the project a robust and an efficient big data project.
ii. How collaborative cloud computing model as a distributed system is used to store data in multiple locations and processed together via

software may help with the v*olume* that refers to the incredible amounts of data generated each second from many different sources such as social media, cell phones, cars, credit cards, photographs, and videos.In big data, the data also comes from many different sources, mainly from the internet, sensors, and large-scale projects and includes snapshot of videos and audio in it. The leading technological organizations, for example, LinkedIn developed many programs through Big Data that includes people you may know, companies you may follow. The main aim of the companies is to extract all the information from the data, which is coming to them. Therefore, companies have invested in developing the Big Data products for them to enhance the growth and efficiency of their organization by providing an infrastructure for sharing tremendous amounts of resources over the cloud. The colossal volume of data isconsidered as the most important characteristic of big data. The high frequency with which the data is coming to the companies is known by Velocity. In present times, data is available in many forms like audio, video, images, gifs and much more. This represents the variety of data coming towards the company. In addition, the trustworthiness of data is determined by Veracity. All these features of big data require a carefully architecture distributed model.

iii. How responding to Change & Distributed systems principals of distributed agile methodology may help with the v*ariety,* which refers to different types of data that can be used. Big data technology allows both structured (name, phone number etc.) and unstructured data (audio, video, text etc.) to be stored and used simultaneously. All projects have a Big Data and advanced analytic element, which creates some technical complexity to every project. Though methodologies vary, empowered of agile teams select the methodologies that best suit their project. Larger teams generally use Scrum, while smaller teams tend towards Kanban and Lean that may be very useful with dealing variety effectively.

iv. How collaboration with customer and working software principle of agile philosophy may help with the v*eracity* refers to the trustworthiness and quality of data.A study in 2020 suggests, “Success requires a much more extensive engagement to clarify the client’s perspective and objectives and develop the insights” [3].The refinements can be adjustments to the backlog based on user feedback which may improve the ability of the agile team to deliver required features and as a result of this generates trustworthiness. This way automation may improve rapid development and delivery more frequently. There is an operational overhead to deploy and maintain distributed systems that is best avoided until it is really needed. Effort spent maintaining the capability is effort you are not spending on user facing features; you cannot demonstrate a Hadoop cluster to an end user [31]. Just-in-Time scalability of agile method needs to be balanced against the need to mitigate technical complexity. When you are confident that scale-out is needed it may be necessary to start prototyping Big Data tools to understanding the level of effort required, providing better estimates for future tasks. Important factors to be considered in facilitating realization of big data systems include architectural design of the system, underlying technologies utilization. This, in turn, may be achieved by close customer interaction and results in quality data.

4. RESULTS AND DISCUSSION

Big data applications became the preferred choice in the world of software engineering in order to handle the growing volume, variety, and velocity of growing real-time data. However, a shift to big data applications means that other processes like business processes, software development methodologies, and infrastructure need to be more streamlined. Moreover, Big data operations require powerful and advanced technologies. The proposed model depends on many open source and proprietary resources, which are available for designing and developing big data solutions in the proposed distributed agile SDLC framework. Although Hadoop is used for massive calculations and operations, there may be some problems during testing Hadoop-based software – such as long processing time, transfer and transform among many services, validating data. A scalable big data test framework is proposed to generate representative datasets, validate data transfers and transformations.

Requirement analysis, maintenance, validation, verification and quality assurance are the distributed agile SDLC phases that were used to

enable big data applications. Although due to 4V properties of big data the requirement gathering is a complex process, this is the crucial step that, in the proposed model, we suggest the use of agile methodology before building the application. Other phases such as validation, verification, and maintenance are the phases that would benefit from the proposed model. The software architecture and design phase of proposed model would provide immense help to agile SDPLC. Maintenance is another phase, which requires attention because in order to keep the application running smoothly it has to be given more importance.

Flexibility and architecture compatibility may be improved bya rule based modular architecture which is an ideal solution due to the control of execution on multiple platforms and its capability of executing wide range of processing services is detailed.

After going through the results, it is evident that each domain in the application domain and each subfield in the software engineering subfields needs a unique level of attention. If the big data projects use the correct software methodologies, we can reduce the failures and errors. Using proposed software engineering methodology in big data projects very much improves efficiency.

Big data analytics has great potential but, to fully use it, more research should be conducted to improve the development of these applications. In addition, changing the methods and practices followed for the development, testing, maintenance of these applications can help to further increase the performance and efficiency of these applications. The problems, like unpredictability and project failure for these applications can be mitigated.

5. CONCLUSIONS AND FUTURE WORK

The rational for applying modified agile SDPLC to big data projects are

i. To explore the possibilities of modified software developments models
ii. To capitalize on the emerging technology of big data.

This paper presents a new conceptual model for pairing up big data and distributed agile SDPLC by modelling it as a four-agent system adopting to changing conditions. We believe that effective agile-big data integration mitigates errors and unsatisfied client expectations. This is definitely the beginning of more general agile-big data studies. Continuous research needs to be conducted in enhancing the existing methods and practices of big data implementation in all phases of SDPLC: maintenance, development, quality assurance etc. In addition, the referenced architecture should be implemented and evaluated with a big data use case. As future enhancement, since small organizations are most interested in using big data, to evaluate the feasibility and usefulness of the proposed distributed multi-agent model, we will carry out a case study using hybrid scrum-Kanban, Scrumban, like framework for a small organization (1 to 100 employees) in the retail application domain to explore the strengths and the weaknesses of the proposed model based on the four main agents described in Section 3. The use case is that a retailer should be able to retrieve the results of the six most recently bought items for a customer.

Big data within the discipline of software engineering has vast capabilities. Along with these capabilities, the popularity is continuing to rise. Most research focused on the application of big data in information technology. This is most likely due to the push to improve the computing field directly. The healthcare industry has a great source of data, and it generates data from various sources like hospitals, clinics, medical governing bodies and many more. This data will be beneficial in developing innovative treatment and cost-effective plan for the patient. The banking and financial industry generates a massive amount of data from different sources as this data will help to know about customer expectations and improve customer service. Therefore, they can tailor their product accordingly.

Since the main goal is to have a realistic idea regarding the ongoing research to know how the broad research in the field of software engineering is allowing big data technologies and the research that is needed for the use of distributed agile SDPLC phases in building robust and scalable multi-agent big data applications as a whole, this paper presents an interesting avenue to explore and achieve tangible gains.

COMPETING INTERESTS

Author has declared that no competing interests exist.

REFERENCES

1. Kim Miryung. SE4DA Software Engineering for Data Analytics, IEEE Software. 2020;02A:0-0. (Available online)
2. ISO/IEC15288:2008—Systems and Software Engineering—System Life Cycle Processes, standard 15288:2008, ISO/IEC; 2008; Available:www.iso.org/standard/43564.html
3. Lin, Yen-Tai, Huang, Sun-Jen. Design of a Software Engineering Lifecycle Process for Big Data Projects, in IT Professional. 2018;20(1):45-52.
4. Henrion, Max. Why Bid Data Analytics Projects Fail", in Analytics Project; 2019. DOI:https://doi.org/10.1287/orms.2019.06.08
5. Andrew White. Our top data and analytics predictions for January; 2019. Available:https://blogs.gartner.com/andrew_white/2019/01/03/our-top-data-and-analytics-predicts-for-2019/)
6. Li N, Escalona A, Guo Y, Offutt J. A scalable big data test framework, in Software Testing, Verification and Validation, Intl. Conf. on. IEEE; 2015.
7. Xu, Xiaofei, Motta Gianmario, Tu, Zhiying, Xu Hanchuan, Wang Zhongjie, Wang Xianzhi. A new paradigm of software service engineering in big data and big service era. Computing. Archives for Informatics and Numerical Computation. 2018;100(4):353-368.
8. Benjelloun FZ, Lahcen AA, Belfkih S. An overview of big data opportunities, applications and tools, in Intelligent Systems and Computer Vision; 2015.
9. Sever A. Modelling distributed agile software development utilizing cloud computing. Journal of Applied Science and Technology; 2019. DOI:https://doi.org/10.9734/cjast/2019/v35i630206
10. Madhavji NH, Miranskyy A, Kontogiannis K. Big picture of big data software engineering: With example research challenges, in Proceedings of the Intl. Workshop on BIG Data Software Engineering. IEEE Press; 2015.
11. Belo O, Cuzzocrea A, Oliveira B. Modeling and supporting ETL processes via a pattern-oriented, task-reusable framework, in Tools with Artificial Intelligence, Intl. Conf. on. IEEE; 2014.
12. Douglas CC. An open framework for dynamic bigdata-driven Application Systems (DBDDAS) development, Procedia Computer Science; 2014.
13. Bazargani S, Brinkley J, Tabrizi N. Implementing conceptual search capability in a cloud-based feed aggregator, in Innovative Computing Technology, Intl. Conf. on. IEEE; 2013.
14. Pääkkönen, Pakkala. Reference architecture and classification of technologies, product, and services for big data systems. 2015;2(4):166–168.
15. Esposito, Ficco, Palmieri, Castiglione. A knowledge-based platform for Big Data analytics based on publish/subscribe services and stream processing. Knowledge-Based Systems. 2015;79:3-17.
16. Xinhua E, Han J, Wang Y, Liu L. Big Data-as-a Service: Definition and architecture," in Communication Technology, IEEE Intl. Conf. on; 2013.
17. Shelton T, Poorthuis A, Graham M, Zook M. Mapping the data shadows of Hurricane Sandy: Uncovering the sociospatial dimensions of big data, Geoforum; 2014.
18. Demchenko Y, Turkmen F, de Laat C, Blanchet C, Loomis C. Cloud based big data infrastructure: Architectural components and automated provisioning. International Conference on High Performance Computing & Simulation (HPCS), Innsbruck. 2016;628-636.
19. Vijay Dipti Kumar, Paulo Alencar. Software engineering for big data projects: Domains, Methodologies and gaps. IEEE International Conference on Big Data (Big Data); 2016.
20. O'Sullivan P, Thompson G, Clifford A. Applying data models to big data architectures. IBM Journal of Research and Development; 2014.
21. Chen HM, Kazman R, Haziyev S. Strategic prototyping for developing big data systems. IEEE Software; 2016.
22. Cecchinel C, Jimenez M, Mosser S, Riveill M. An architecture to support the collection of big data in the Internet of Things, in Services, World Congress on. IEEE; 2014.
23. Krämer, Senner. A modular software architecture for processing of big geospatial data in the cloud. Computers & Graphics. 2015;49(C):69-81.
24. Al Zamil MG, Samarah S. The application of semantic-based classification on big data, in Information and Communication Systems, Intl. Conf. on. IEEE; 2014.

25. Agrawal R, Imran A, Seay C, Walker J. A layer-based architecture for provenance in big data, in Big Data, IEEE Intl. Conf. on; 2012.
26. Yang-Turner F, Lau L, Dimitrova V. A model-driven prototype evaluation to elicit requirements for a sense making support tool, in Software Engineering Conf., Asia-Pacific. IEEE; 2012.
27. Breuker D. Towards model-driven engineering for big data analytics–An exploratory analysis of domain-specific languages for machine learning, in System Sciences, Hawaii Intl. Conf. on. IEEE; 2014.
28. Vanauer M, Bohle C, Hellingrath B. Guiding the introduction of big data in organizations: A methodology with business-and data-driven ideation and enterprise architecture management-based implementation, in System Sciences, Hawaii Intl. Conf. on. IEEE; 2015.
29. Waltz D, Kasif S. On Reasoning from Data, ACM Computing Survey. 1995;27(3):356-359.
30. Rahman F, Slepian M, Mitra A. A novel big-data processing framework for healthcare applications: Big-data-healthcare-in-a-box. IEEE International Conference on Big Data (Big Data), Washington, DC. 2016;3548-3555.
31. Dresner Advisory Services, Big Data Analytics Market Study; 2017. Available:https://www.microstrategy.com/getmedia/cd052225-be60-49fd-ab1c-4984ebc3cde9/Dresner-Report-Big_Data_Analytic_Market_Study-WisdomofCrowdsSeries-2017.pdf

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle4.com/review-history/57799