Chapter 11
Software Engineering Analytics—The Need of Post COVID-19 Business: An Academic Review

Somayya Madakam and Rajeev K. Revulagadda

Abstract "Man Proposes, God Disposes". COVID-19; the unexpected pandemic decease is a good testimony of the above proverb by exemplifying global business operations shutting down. Pre COVID-19, the business across the globe more or less are planned one; running smoothly; trying to reach their goals and objectives; satisfying the customers, finally leading to leapfrog the companies' profits. This was the scenario, before the coronavirus; the alias of COVID-19 pandemic, which was germinated in one of the most scientific cities in china; Wuhan. Entire world business collapsed; another global economic recession after 2008; touching all the continents, countries, business, tribes, religions, lifestyles, as well as professional lives. No medicine was unable to save the lakhs of human life from contamination. Even different country leaders practised various preventive methods in order to cure the decease to save human life. In this light, this book chapter will explore the state of the art of COVID-19 across the globe with respect to business. The entire data analysis is based on secondary on-line data and thematically narrated. The book chapter furtherly discussed in detail connotes, process and state of art of software engineering and software analytics in business for sustainability. The analysis reveals that software analytics technology is the only industry lightly affected, should grow rapidly and it is the only solution provider to improve the process, and sustain business via better-automated software. In long term incurs less cost, time management, less manual intervention, integration of enterprise departments, virtual meetings, and e-commerce are the beauty of software analytics.

Keywords Software engineering · Software analytics · Analytics · Software engineering analytics · Web of science · COVID-19 · Pandemic

S. Madakam (✉)
Information Technology, FORE School of Management, New Delhi, India
e-mail: somayya@fsm.ac.in

R. K. Revulagadda
Finance & Accounting, National Institute of Industrial Engineering (NITIE), Mumbai, India

11.1   Introduction

The five-letter word 'COVID-19' given by medical scientists to a deadly virus made the whole world chaos. It does not have any barrier either developed or developing, rich or poor. This twenty-first century never knows that it will face massive destruction, which cost lakhs of human lives. This did not stop just by affecting the health system of the nations but also economic, business and political stability. The coronavirus rippled in the entire business ecosystem; effected all the industries in the world. Organizations with the best performance in the pre-COVID era trembled down day by day and worsened the condition of its affiliates as well. The ripple thrashed one industry to another i.e. from transportation, service, manufacturing, tourism etc. Before knowing about how this virus affected industry to industry, one has to know about non-negligible information about this virus.

According to the findings of scientific study on COVID-19 by Chan et al. [1], the virus was first identified in Wuhan , China. This novel virus belongs to the family of coronavirus. Till date, a sea of studies were published continuously in the top journals including medical journals like lancet. However, the symptoms vary from person to person based on age and geographical locations i.e. from country to country, but common symptoms included pneumonia, high fever, dry cough etc. The rapid spread of the virus in population led to serious concern by governments. This initiated the imposing of lockdowns and shutting down the businesses slowly. The need for health care workers has risen rapidly and highlighted the lack of research in the medical field. Slowly, many countries have started to sense the joblessness of its citizens, which has affected the manufacturing, export etc. This eventually led to slow down the income growth and in worst conditions; some countries experienced a high rate of unemployment. As the ripple never gets back, it started continuously affecting with a negative effect on the financial statements of many companies. This phenomenon and different unknown circumstances led to the question of recession. Rather diving directly to the interpretation at a macro level, it will be beneficial to understand the impact of the pandemic on the industry after industry. Economic indicators of a country depend on a pool of sectors. For example, mostly developing countries like India are dependent on agriculture outcome. Though industries like service and banking are performing better, a high segment of the population is dependent on agriculture. Hence, it is evident that the impact on agriculture will show an effect on commodities export and imports. Lack of credit availability for purchasing the raw material, and increase in expenditure for seeds, fertilizers and unavailability of labor has increased the cost of production in the agriculture sector. On one side the countries experienced a fall of growth in income and wages and on the other side, prices are very odd. Essential commodities are available but the imposition of transport barriers made to purchase them at a high price. These issues are even worse for the people under the poverty line across nations. The other burdens like lack of medical staff and infrastructure further worsen the situation of the people. Hence, the negative impact on a high proportion of the population led to the downfall of sales in other industries. When compared with the agriculture sector, many sectors

are most affected by this pandemic. There are many sectors, which are badly affected by COVID-19 in detail in the following section.

11.2   COVID19-Effects in Business

11.2.1   Tourism and Hospitality

The impact of the pandemic on tourism is highly negative and the industry is in complete stress. When the lockdowns are imposed and travel restrictions are initiated in many countries, passenger commute was totally shut which led many aviation services to suspend. To reduce the outbreak caused by the pandemic, many tourists have cancelled their touring plans. Lockdown imposition has led to close down the hotels, restaurants, malls, museums, open-air theatres, archaeological sites etc. in many countries. UNWTO (UN-World Tourism Organization) raised the concern to all the countries that, governments are responsible to prioritize the health of the public and they should be responsible for protecting jobs.

In all the regions of the world as shown in Fig. 11.1, the tourism industry was worst hit by this pandemic. The revenues for Asia on tourism is decreasing at an increasing pace. This travel restriction has caused many jobs at risk. The demand for tourism and hospitality has to rise in due course, as this sector is the worst hit.
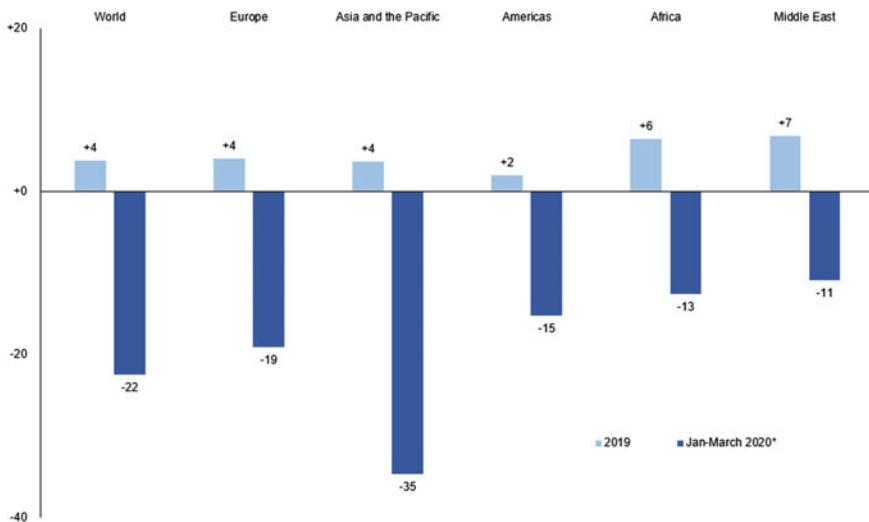

Fig. 11.1 International tourist arrivals, 2019 and Q1 2020 (% Change). *Source* UNWTO-World Tourism Organization (https://www.unwto.org/news/covid-19-international-tourist-numbers-could-fall-60-80-in-2020)
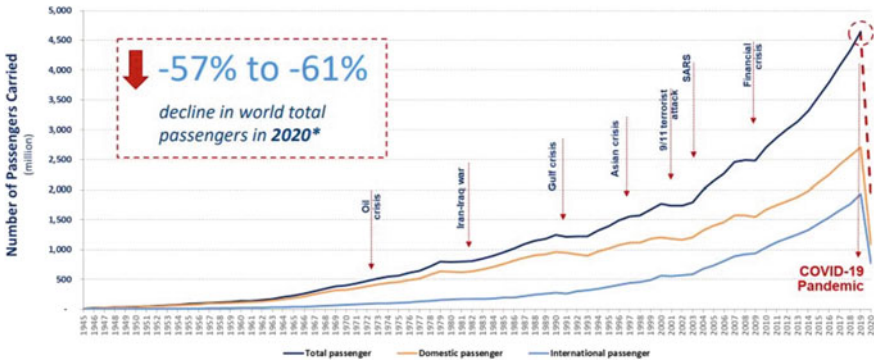
Fig. 11.2 World passenger traffic, 1945–2020*. *Source* ICAO, The International Civil Aviation Organization (https://www.icao.int/sustainability/Documents/COVID19/ICAO%20COVID%202020%2009%2023%20Economic%20Impact.pdf)

11.2.2 Airlines and Aviation

COVID-19 also affected the aviation sector with generating heavy losses and closing down of services. As the transmission rate is high from person to person through cough or sneeze, it is very reluctant for the aviation industry to continue the services. The potential loss of gross passenger operating revenues is approximately USD 375 to 395 billion. ICAO reveals that a reduction from 49 to 51% of seats offered by airlines and passenger overall reduction of 635 million passengers. The study reveals that this pandemic made the world passenger traffic with unprecedented decline ever. With the summer season missed, global travel got heavily impacted. The data in Fig. 11.2 from 1945 to 2020 shows that COVID pandemic crises made all other major crises of the world less impact on the world passenger traffic. The ripple effect caused by the demand of tourism has also affected the growth and survival of the aviation industry. If this severity level grows furthermore for international services, this will eventually affect the growth of the aviation sector. Thus due to COVID-19, this industry is heavily affected.

11.2.3 Oil and Gas

Oil and gas industry was also one of the hardest hit by the pandemic COVID-19. The imposition of lockdown and unavailability of the workforce, the industry has to stop either the operations or irregular in operating the physical operations. This has affected oil and gas production with a bad flow of continuity in operations. The industry has many problems in which unstable prices of oil barrels, doubt of employee safety and security in the pandemic environment and other crises that come from a different set. Those include supply chain risk and other unidentified risks. The

depressed price continued for a longer time due to the crisis. The main cause for the reduction of barrels prices cannot be said unless identified all the possible impacts. The ongoing trade war between the countries has made sensitive prices to change continuously i.e. unstable. The companies, which are public sector, has an edge when compared to the companies that service the other industrial purposes. This COVID-19 has made the employees, and the management to think about the security and continual flow of business. The pandemic related issues but also sensitive towards the government's decisions not only affect this. Hence, with the ongoing challenges, this industry was hardly hit.

11.2.4 Automotive Manufacturing

The automotive manufacturing sector is a significant sector but it has the ripple effect caused by COVID-19; the fall in demand for products. The plant closures due to employee safety made furthermore worst on the performance of the automotive sector. The credit from banks and the burden of repayments to the credit lenders made some companies vulnerable. However, the demand has increased slowly, but it takes time to settle the ongoing issues of the automotive sector. The sales of the vehicle have rapidly fallen in all the countries. When compared with 2019, the industry is facing a turbulent environment Sales data of China, Europe and the US has sales down when compared with 2019 (Fig. 11.3). This shows the industries crucial issue in the pandemic season and considered as the worst hit.
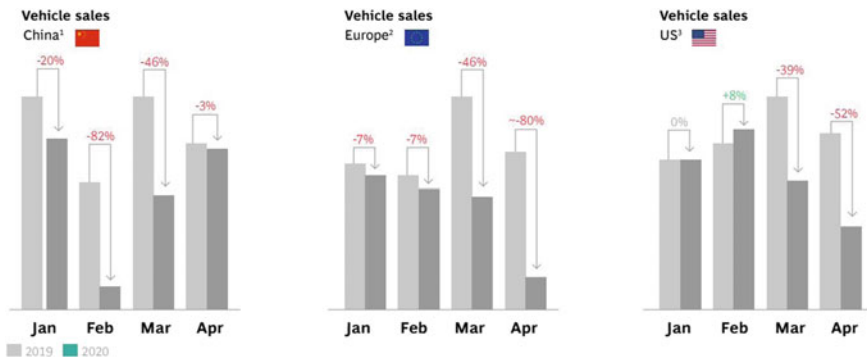

Fig. 11.3 Drop in auto sales. *Source* BCG, Boston Consulting Group (https://image-src.bcg.com/Images/Auto%20post%20COVID-19_052920_tcm9-249607.pdf)

11.2.5 Consumer Products

Consumer products are somehow good, but the revenues that rise from the consumption by the public will be high. The supply chain connectivity is also becoming a big problem, in the pandemic season. The consumption of alcohol and beverages will be hit a hard time in demand. However, daily essentials demand will be increased rapidly. The change in consumption will be seen in different types of different products. The things that changes in operations are very crucial in handling consumer products. Agility, portfolio evaluation, shopper insights, flexible production are some of the key elements that have to be considered for transformation in the industry. This not only helps the industry to survive from these pandemic environments but also helps to mitigate the risks from the unknown and uncertain barriers. Hence, this sector is not that worst hit but product segmentation plays a key role in overcoming the barriers. Along with these sectors, the pandemic environment also hit electronics but, after the movement and lifting up the travel restrictions and other key barriers, this helped to survive better than many worst hit business sectors. Hence, the COVID-19, most hits to better-performing sectors were explained. Coping up with the pandemic challenges and barriers by getting the customer insights, need and type of the product will help the industries to survive in these chaotic and unprecedented conditions. To overcome the ripples caused by the other business, the industries have to be agile and active.

11.2.6 Education

The education sector did not leave untouched by this pandemic. A long halt and uncertain timing have made the education sector unstable. Pandemic showed the impact on the nursery, primary, high school to professional educations. University management has made many efforts to limit the loss by pandemic lock-down by offering on-line teaching services. Though the internet has removed the boundaries between countries and helped many to seek the knowledge of top universities, there are still rigid barriers for many students. Mostly these issues are for students in developing countries. Students in rural and semi-urban explain the lack of infrastructure and monetary support to purchase the infrastructure. India is one of the examples that the education sector affected very badly. Still, the schools and colleges are closed due to the increasing number of corona cases. Nevertheless, software companies have tied relationships with academic institutes to offer the best online educational experience. Due to pandemic situations and physical classroom access barriers, the software giants are come up with video conferencing teaching tools or virtual classroom systems. Microsoft's teams; Cisco's Webex Meet; Zoom rooms; GoToWebinar, and Google talk etc. Depending on sub-scribed licenses, the number of students can get connection for online learning.

11.2.7 Software Industry

The impact on the software industry is also crucial for comprehensive development. This industry has an option to face the crisis by offering work from home options to the employees. This helped to cope up with the contracts and credit flow to the companies. The software sector helped to resist the unemployment to some extent. It helped the pharma companies and governments to track the real-time data of COVID-19 patients and tracking the victims on time. Understanding its characteristics and elements will further help the decision-makers to overcome these situations. The software industry got good opportunity to do the business from the home. However, the industry lost most of the business opportunities due to interdependency among all business. Software industry/engineering incorporates all aspects of software production from business strategy to designing and coding, testing, quality and management of large-scale complex software systems in a bigger way during post COVID-19; also to conduct business mostly on-line. Software industry pave the path for e-business and e-commerce round the clock to get competitive advantage as well as business sustain. Software engineering is about multi-person development of multi-version software that is large and complex based on the Windows, Linux and Android operating systems relating to office software, flight control systems, automation, education, and manufacturing applications. Software engineering is an old field emerging from the Computer Science and IT management; however now develop all kinds of business applications to run smoothly during post COCID-2019 or all kinds of situations.

11.3 Research Methodology

As the software analytics or software engineering analytics field is novel and much academic scribbling is not available, it is the need of the hour to do more rigorous research work and develop new applications and innovative findings for better software engineering process during post or in these pandemic days. Since, the concept is contemporary and not much empirical evidence is not available, the present manuscript graphed based on secondary data (articles). The google scholar and google databases are bases for searching the articles on this phenomenon. The writings carried out during 1-1-2019 to 30-6-2020. Since, almost one and half year duration, the authors got enough time to browse, came across and conceptualize the phenomenon in a better way, in which the scribblings are bringing the crystal clear ideas on the software engineering analytics connotes as well as on COVID-19 and its relationships. Besides the web of science articles are extracted for the composing this article. The author discovered 35 articles on software analytics. Besides, some of the articles and blogs considered with reference to COVID-19 phenomenon. The general awareness of academic and research database of the web of science is much more advanced in an Information Technology global world who are into the research

field. The study by Bakkalbasi et al. [2] says that for year's researchers looking for this type of information had only one resource to consult: the Web of Science from Thomson Scientific (Clarivate Analytics). Therefore, the web of science is vital for any kind of research across the globe.

11.4 Software Development Life Cycle (SDLC)

The software is inevitable in our daily life. The entertainment devices like computers, mobiles, tablets, laptops, televisions, etc. run by different embedded software. Journeys are based on Geographical Information Systems (GIS), Geo-graphical Position System (GPS) and other remote sensing software etc. Besides, Management Information Systems (MIS), Enterprise Resource Planning (ERP), Computer Aided Design (CAD), Computer Aided Manufacturing (CAM) software ate backing organizations, manufacturing companies, and corporate business and so on. Government organizations, Academic Institutions, Research & Development (R&D) offices are handicapped without Microsoft office suite, International Business Machines (IBM)'s analytical Statistical Package for Social Sciences (SPSS), Atlas-ti, and N-Vivo many more application software suites. That means human life is completely dependent on the multiple software for more comfortable business life. Scripts, applications, programs and a set of instructions, all terms often used to describe software. In a nonprofessional point of view, the software is a collection of programs, the programs are nothing but a collection of logically, and sequential arranged instructions to do particular tasks or to resolve one issue. Different languages are using to develop the software including JavaScript, Python, R, Go, and many more. However, the base languages in decades back are FORTRAN (Formula Translation), COBOL (Common Business Oriented Languages), and, Pascal. Moreover, the C, C++, C#, PHP (Hypertext Preprocessor), HTML 5, Macromedia, Swift, Ruby are some of the best dynamic high-level languages using to develop many application software and utilities in these days. Currently the software development using high-level languages is becoming very easy in IDEs. By putting minimum efforts, anybody can write codes for application software development with the help of many available built-in libraries. Therefore, developing a software is very easy these days irrespective of application. Software are available in the market freely, so-called open source as well as other licensed software in which the user has to pay the amount for the software usage. Any software generally developed by Software Development Life Cycle (SDLC) process. Moreover, different companies practicing different development approaches including spiral, incremental, agile, RAD, DevOps to name a few.

11.4.1 Software Requirements Analysis

The first phase of software engineering or SDLC is Software Requirements Analysis (SRS). Here the actual product or service specifications will be taken from the end user or client. This business can be a kind of Business to Business (B2B), Business to Customer (B2C), Business to Government Business-to-Business (B2B), Consumer-to-Business (C2B), Business-to-Administration (B2A) and Consumer-to-Administration (C2A). All the software specifications are noted down very clearly including product details like media design, user interface, color, compatibility, language, operating system, and the number of workforce involvement, technical staff selection, project cost, and time of completion, payment options, and maintenance. The criticality of the Software Requirements Specifications phase of the software life cycle for the success of the whole software project is widely recognized and the attention played on it by software developers is more and more significant [3]. Early inspections of software requirements specifications are known to be an effective and cost-efficient quality assurance technique. However, inspections are often applied with the underlying assumption that they work equally well to assess all kinds of quality attributes [4]. Software Requirement Specifications are a key result of the requirement engineering process and an important basis for every large industrial software development project [5]. SRS document is the first deliverable product/milestone in the software development process and acts as a basis for the formal contract between the user and the developer of the software of an information system [6]. Moreover, the requirement specification is gaining increasing attention as a critical phase of software development [7].

11.4.2 Software Design

The concept of information hiding modularity is a cornerstone of modern software design thought, but its formulation remains casual and its emphasis on changeability imperfectly related to the goal of creating benefit in a given context [8]. The software design is the most critical part of the entire software engineering process. This is the abstract view or graphical form of any software product before transforming into reality. All the software requirement specifications logically converted into software design or model. The graphical representation of this phase is the medium from product idea to a physical/logical form of the software. The term "software architecture" typically refers to the bigger structures of a software system, whereas "software design" typically refers to the smaller structures [9]. However, software architecture and software design are two aspects of the same topic. Both are about how software structured in order to perform their tasks. Despite a diversity of software architectures supporting information visualization, it is often difficult to identify, evaluate, and re-apply the design solutions implemented within such frameworks

[10]. Even for software design, there are different new software products are available in the market including CAD, CAM, Microsoft project management and many more developed by Information Technology firms. Moreover, the software design-cognitive aspects cover a variety of areas including software analysis, design, coding and maintenance [11]. Apart from these, PLEASE is an executable specification language that supports software development by incremental refinement. PLEASE is part of the ENCOMPASS environment that provides automated support for all aspects of the development process [12]. Therefore, briefly, the software design is the blueprint for entire software product including time, cost, workforce, and logical relationships among submodules. That means the software design is the birds eye view of the software project. The tools and techniques using in software design are Flowchart, Data Flow Diagram (DFD), Data Dictionary, Structured English, Decision Table and Decision tree. The nest section will discuss in detail about the software implementation phase that means how to do the coding.

11.4.3 Software Implementation

This phase is a vital portion of all software engineering projects. Software implementation generally done at company level. It involves all the technical people including website designers, programmers, network administrators, systems analysts, functionalists, chief information officers, and database managers. Depending on the project type, the professionals will be involved in developing the software. The languages will be used based on the application either frontend-or-backend level. However, currently, the development of the app is becoming very easy due to the availability of built-in libraries, syntax prepared by regular English language and keywords well versed. This is the phase in which, the right programmer to identifying and assigning from the technical bench from the corporate. The functionalist always monitors the completion of the task, results checking, and modified inputs from the customer to developers based on the software model using. We knew that in today's software development environment, requirements often change during the product development life cycle to meet shifting business demands, creating endless headaches for development teams [13].

Moreover, properly managed, architecture-centric methods can be a cost-effective addition to the software development process and will increase system and product quality [9]. However, a model of software development maturity describes managerial processes that can be used to attack software development difficulties from the managerial control perspective at five maturity levels [14]. An optimal software development process regarded as being dependent on the situational characteristics of individual software development settings including the nature of the application(s) under development, team size, requirements volatility and personal experience [15]. The developing team should meet all the criteria as per Software Requirement Specifications. The next phase of software implementation is software testing; discussing in the next section.

11.4.4 Software Testing

The Software testing is also an important part of software engineering. Testing generally done whether the software requirement specifications are met or not. If not, the deviations can be incorporated to meet the exact specifications, otherwise convincing the client with the developed specifications. Therefore testing is the measurement of the exactly predefined list at the time of customer project specifications. In software engineering or development process, the product divided into sub-modules in which the developers can easily code and finish it as early as possible in terms of big projects. In this case, the submodules are not only divided but also tested individually and later by integrated as one. Software testing is a critical stage in entire software development used to ensure that a program meets required specifications and does not contain errors in programming code [16]. Exhaustive testing of computer software is intractable, but empirical studies of software failures suggest that in some cases testing can be effectively exhaustive [17]. Recent years have witnessed a surge of interest in symbolic execution for software testing, due to its ability to generate high-coverage test suites and find deep errors in complex software applications [18]. Moreover, in commercial software development organizations, increased the complexity of products, shortened development cycles, and higher customer expectations of quality have placed a major responsibility on the areas of software debugging, testing, and verification [19].

11.4.5 Software Deployment

At the customer's site, the technical team will install necessary application software along with necessary hardware and networking components based on the clients' requirement. Worldwide one of the most important parts of the software development life cycle (SDLC) is user acceptance. Software deployment is a post-production activity that performed for or by the customer of a piece of software [20]. Even though the product is not 100% as per the specifications, convincing the user to accept the software is a vital part. This completely depends on the situation, level of customer acceptance and skill of the functionalist to convince the customer/client. This happens at the time of deployment of the software in the customer premises. The functional analysts or coordinator of software development will assist. Software deployment process consists of all the preparing a software application to run and operate in a specific environment and encompasses right from installation, configuration, testing and making changes to optimize the performance. The process already discussed in the seminal paper by saying that the software deployment is an evolving collection of interrelated processes such as release, install, adapt, reconfigure, update, activate, deactivate, remove, and retire [21]. Moreover, software applications are no longer stand-alone systems. They are increasingly the result of integrating heterogeneous collections of components, both executable and data, possibly dispersed

over a computer network [22]. Hence, software development should be taken care of by both parties to make the software project successfully. The deployment process includes installation of software and hardware in the premises of client along with documentation and user training. Moreover, the software conversion process done in one of the four approaches either by (1) Parallel Strategy (2) Direct Strategy (3) Pilot Study Strategy (4) Phased Approach Strategy. This process make sure that (1) How to execute the packages? (2) How to enter the data? (3) How to process the data (processing details)? (4) How to take out the reports?

11.4.6   Software Maintenance

Maintenance is the most important post activity of any business sales. After delivering the software product to the end customer or user, most of the products and services are needs technical support from the supplier of the product. Many times the user may not be aware of the small repairing issues also, which may occur at the time of using the software product. Even though the entire software usage and bug deductions are given in the manual at the time of software delivery, installation and, users are scared of how to use and debugging the errors. Since most of the software developed with the help of computing programming languages, it is better to provide maintenance support annually. Nowadays companies are coming with different maintenance packages like 1-year free service otherwise, if the user goes for extended maintenance—the charges will be very less cost sort of. However, software maintenance and evolution characterized by their huge cost and slow speed of implementation [23]. Management has turned to software engineering tools designed to support software maintenance as a potential solution to maintaining productivity and quality problems [24]. Software maintenance is the dominant factor contributing to the high cost of software [25]. Therefore, the bugs need to be analysed before shutting down the systems. In this scenario, if we develop automatic prior intimation systems or automatic maintenance systems or would be grateful to system people. Moreover, software engineering comprehends several disciplines devoted to prevent and remedy malfunctions and to warrant adequate behaviour [26]. Annual Maintenance Contract (AMC) stands for annual maintenance contract where the company charges some lumsum amount from their customer for specified product for a fixed period and fixed services. This AMC includes (1) Knowing the full capabilities of the system (2) Knowing the required changes or the additional requirements (3) Studying the performance.

All industries today require computer scientists with advanced skill sets and an ability to develop, test, and maintain software engineering applications to meet ever-evolving modern industry and commercial requirements. A systems analyst produces a software requirement specification that precisely describes the attributes of the software to be produced [10]. As we are aware that the general software development process involves software requirements analysis, design, implementation, testing, integration, deployment, and maintenance. However, in the business landscape, there are different software development models, i.e. water-fall, spiral, the

process of prototype, incremental, phases of iterative development and the principles of agile, and dev-ops methods. Local and multinational companies like Mphasis, Hindustan Computers Limited (HCL) Technologies, Mind-tree, IBM, Rolta India Ltd, Oracle Financial Services Software, Wipro, and Tata Consultancy Services (TCS) are working so hard to develop software in different domains including healthcare, pharmaceutical, manufacturing, aviation, marine, entertainment since decades. The IT/ITes companies offering multiple services relating to custom business solutions, web branding, internet marketing, collaborative content management, database migration, customization, application development including outsourcing, ERP solutions, iPhone apps development, collaborative commerce, programming and quality assurance, and testing services, multimedia offering and consulting. We are all familiar with traditional software engineering research and practices including requirements engineering, software project management, design, development, testing, and implementation of software applications. This software engineering discipline developed as a practice which has been well established over the past 50 years. Software development is an art; it is a skill and complex group task too. Software development is a professional activity. It is completely programming by different software developers including computer engineers, functional analysts, and business tycoons.

11.5 Software Analytics

11.5.1 Literature Analysis

The following are the total articles published as on dated 30/06/2019

Total Articles Published: 35
Sum of the Times Cited: 316
Average Citations per Item: 9.03
H-index per these articles: 10

Out of the 35-research manuscript, the software analytics/software engineering analytics connotes clearly defined including the physiognomies and importance of field. From the above data Table 11.1, it was understood that up to now the research is on software analytics in practice including understanding users' behavior with software operation data mining and recommendations for mass spectrometry data quality metrics for open access data. Besides, there are different practices including visual software analytics for the build optimization of large-scale software systems. Some of the academic studies are based on qualitative including a retrospective study of software analytics projects: in-depth interviews with practitioners. Besides, the papers are on continuously assessing and improving software quality with software analytic tools: a case study. Like that, many faces of software analytics were scribbled. Using Analytics to Guide Improvement during an Agile-DevOps Transformation another study which intensifying both Agile-DevOps models. However,

Table 11.1 Software analytics in practice (Articles Titles: 35)

Title of the articles
Software Analytics: So What?
On the automatic classification of app reviews
Tuning for software analytics: Is it really necessary?
Human–computer interaction in evolutionary visual software analytics
Understanding users' behavior with software operation data mining
Recommendations for mass spectrometry data quality metrics for open access data (corollary to the Amsterdam principles)
Behavioral Response to a Just-in-Time Adaptive Intervention (JITAI) to Reduce Sedentary Behavior in Obese Adults: Implications for JITAI Optimization
Student and Faculty Member Perspectives on Lecture Capture in Pharmacy Education
A Retrospective Study of Software Analytics Projects: In-Depth Interviews with Practitioners
Recommendations for Mass Spectrometry Data Quality Metrics for Open Access Data
Rapid Releases and Patch Blackouts A Software Analytics Approach
Are Software Analytics Efforts Worthwhile for Small Companies? The Case of Amisoft
Recommendations for Mass Spectrometry Data Quality Metrics for Open Access Data
Roundtable: What's Next in Software Analytics
Searching under the Streetlight for Useful Software Analytics
How Evolutionary Visual Software Analytics Supports Knowledge Discovery
Recommendations for mass spectrometry data quality metrics for open access data (corollary to the Amsterdam principles)
Visual software analytics for the build optimization of large-scale software systems
Knowledge discovery in software teams by means of evolutionary visual software analytics
Investigating and Projecting Population Structures in Open Source Software Projects: A Case Study of Projects in GitHub
The Unreasonable Effectiveness of Software Analytics
Implementation of a consumer-focused eHealth intervention for people with moderate-to-high cardiovascular disease risk: protocol for a mixed-methods process evaluation
IRISH: A Hidden Markov Model to detect coded information islands in free text
Addressing problems with replicability and validity of repository mining studies through a smart data platform
The Many Faces of Software Analytics Introduction
Using Analytics to Guide Improvement during an Agile-DevOps Transformation
Roundtable: Research Opportunities and Challenges for Emerging Software Systems
Understanding software artifact provenance
Towards base rates in software analytics Early results and challenges from studying Ohloh
Bad smells in software analytics papers
Software Engineering Data Analytics: A Framework Based on a Multi-Layered Abstraction Mechanism

<div align="right">(continued)</div>

Table 11.1 (continued)

Title of the articles
Continuously Assessing and Improving Software Quality with Software Analytics Tools: A Case Study
Software Analytics: What's Next?
Experience report on applying software analytics in incident management of online service

some research papers intensify about bad smells in software analytics papers. Moreover, what is Next? In software, the Analytics buzz word in the future. From the articles it can be understood that software analytics is interdisciplinary and can bring applications in all fields.

11.5.2   Journal Publications with Frequency

The above Fig. 11.4 shows that the ascending order of Journal of publications. IEEE Software (11), Science of Computer Programming (4), Computers in Hu-man Behavior (2), IEICE Transactions on Information and Systems (2), Information and Software Technology (2), American Journal of Pharmaceutical Edu-cation (1), Automated Software Engineering (1), BMJ Open (1), Computational Statistics (1), Empirical Software Engineering (1), Health Psychology (1), IEEE Access (1), Journal of Computer Science and Technology (1),, Journal of Proteome Research (1), Molecular & Cellular Proteomics(1), Proteomics (1), Proteomics Clinical Applications (1), and Requirements Engineering (1).
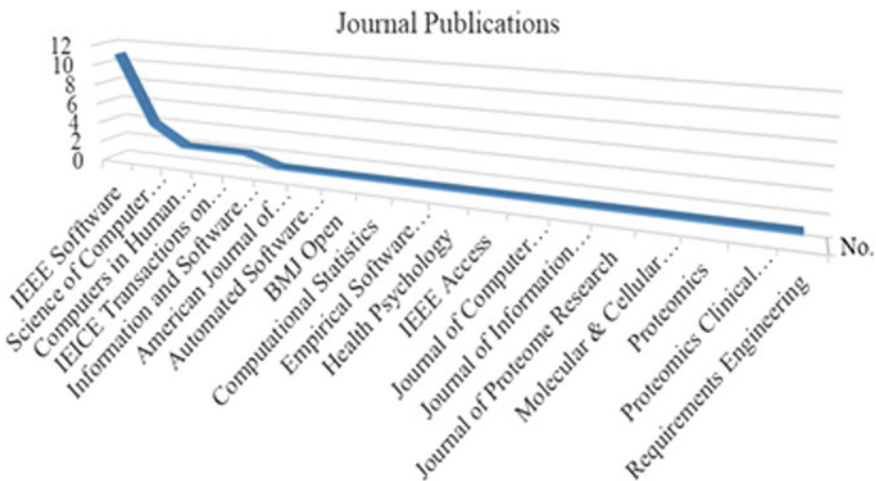

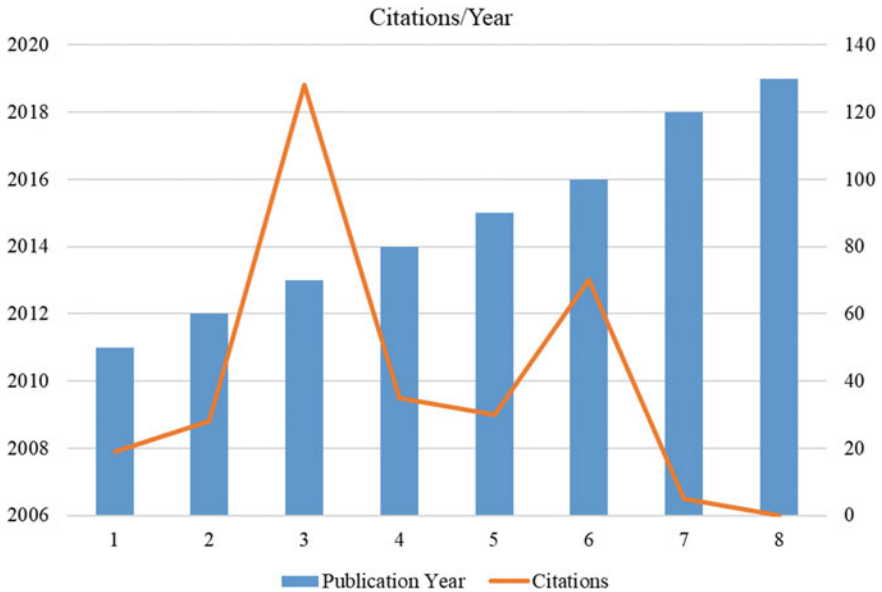**Fig. 11.4** The Journal wise publications

Fig. 11.5 Citations/year

11.5.3 Citations/Year

Less had been talked about the phenomenon of software analytics and software engineering analytics. We are getting this evidence even from Fig. 11.5. The graphs explore that the number of academic and research publication citations over a period is very less. That means the field is new and contemporary. Besides the phenomenon is well represented in the Graph 5. by year wise publication citations including 2011 (19), 2012 (28), 2013 (128), 2014 (35), 2015 (30), 2016 (70), 2018 (5), and up to now June 2019 it was only zero citations as per web of science database. However, there was ongoing research during 2011, 2012 and 2013. Later there was a sudden decrease in software analytics and software engineering analytics research, publications and its citations. Probably, the analytics field was expanding its applications in other fields like business analytics, big data analytics, marketing analytics, HR analytics, and social media analytics and so on. Hence, the researchers started working and exploring the above field, instead of working, publishing, and citing on software analytics phenomenon.

11.5.4 Software Engineering Analytics: Connotes

The software development is a data-rich activity with many sophisticated metrics. Software engineering data are in many forms such as user stories, use cases,

requirements specifications, issue and bug reports, source code, test cases, execution logs, app reviews, user and develop mailing lists, discussion threads, and so on. For example, the well-known Web browser, Mozilla Firefox project, had over 1 million bug reports by 2014. Hidden in those big data are valuable insights about the quality of the development process, and the experience that software users receive. Despite large volumes of data and many types of metrics, software projects continue to be difficult to predict and risky to conduct [27]. We are well versed with 4–5 decades onwards on software development; it is a data-rich activity with many sophisticated metrics. Yet engineers often lack the tools and techniques necessary to leverage these potentially powerful information resources to-ward decision-making [28]. Hence, analytics is the need of hour across the globe in current scenario. The word "Analytics" is becoming a buzz in every domain all over the world. Analytics is now eye-catching from the public. We often come across the phrases with Analytics as suffix like 'Data Analytics'; 'Business Analytics'; 'Big Data Analytics'; 'Descriptive Analytics'; 'Diagnostic Analytics'; 'Prescriptive Analytics'; 'Predictive Analytics'; 'HR Analytics'; 'People Analytics'; 'Supply Chan Analytics', 'Marketing Analytics'; 'Consumer Analytics'; 'Fraud Analytics'; 'Financial Analytics'; Software Analytics'; and many more.

The word Analytics is not new and derived long back from the Greek word "Analytic', which means 'of or proceeding by analysis' or 'skilled in the careful study of something'. That means the Analysis is nothing but the methods to follow one particular task systematically. These analytics using for analysing the huge amount of data generated from different sources and trying to find out the patterns using the software. These patterns or insights are using for different purposes in all sorts of business, depending on the nature of business. Therefore, whatever, the sort of analytics, the objective of analytics is to understand the hidden patterns/insights in the organizational data? As technology continues to advance, we constantly generate an ever-increasing amount of data. Data analytics is the process of examining large data sets to uncover hidden patterns, unknown correlations, market trends, customer preferences, and other useful business information. In data analytics, the data is huge in volume in generation, processing and storage levels. This data is generating in the giga / tera / zetta / peta / yottabytes per second from different electronic computing and communication devices. Besides, presently even the data is germinating from the physical objects or things too because of their connectivity to the internet. We knew that social media sites like Skype, Facebook, WhatsApp, LinkedIn, Twitter, Instagram, Myspace, Telegram, Viber, etc. for every second producing official, personal, and social information in the form of text, pictures, audios, and videos. All this data is sometimes structured / unstructured and many times mixtures of these two. So analysing this complex data with Analytics tools for useful decisions of netizens is essential in this world.

In this line, the software analytics also entering as an emerging concept, in which all the traditional Software Engineering research and its practices include requirements engineering, software project management, design, development, testing and implementation of software applications merged with this new technology "Analytics". While coming to the software analytics is to enable software practitioners

to perform data exploration and analysis to obtain insightful and actionable information for data-driven tasks around software and services [29]. Software analytics is the analytics specific to the domain of software systems taking into the account source code, static and dynamic characteristics as well as related processes of their development and evolution. Besides, software analytics is to enable software practitioners to perform data exploration and analysis to obtain insightful and actionable information for data-driven tasks around software and services. Nowadays, a huge amount of data continuously generated at a rapid rate during the engineering of software systems. In the last decade, modern data analytics technologies have enabled the creation of software analytics tools offering real-time visualization of various aspects related to software development and usage. These tools seem to be particularly attractive for companies doing agile software development [30]. With software analytics, software practitioners explore and analyze data to obtain insightful, actionable information for tasks regarding software development, systems, and users [31]. Abdellatif et al. [32] said that is concerned with the analysis of all software artifacts, not only source code, defines software analytics. Its importance comes from the need to extract support insights and facts from the available software artifacts to facilitate decision-making. Artefacts are available from all software development life cycle steps, beginning with the proposal and project initiation phases and ending with the project closure and customer satisfaction surveys. Yet engineers often lack the tools and techniques necessary to leverage these potentially powerful information resources toward decision-making. Despite large volumes of data and many types of metrics, software projects continue to be difficult to predict and risky to conduct [28]. Software engineering analytics (Shah [33]) is the process of development, and testing activities—uncovering quintessential insights and recommendations. It is a very complex and data-heavy activity, involving millions of lines of code, a huge bug database, and complex testing frameworks.

However, understanding and analyzing user needs plus using of core programming languages like C, C++, Java, etc. and testing, with the help of tools like maven, ant, gradle etc., configuration tools like chef, puppet, etc. play an important role in Software Engineering. Software analytics and the use of computational methods on "big data" in software engineering is transforming the ways software is developed, used, improved and deployed [34]. Developers sometimes develop their own ideas about good and bad software, based on just a few past projects. Using software analytics, we can correct those misconceptions. Software analytics lets software engineers learn about AI techniques [35] Off course, the sweeter fruit of software analytics is spoiling: data-mining techniques are oblivious to the software domain [36]. Besides, this literature, now the researchers and academicians, and software engineers set out scribbling on software analytics.

11.6 Conclusion

The Software Development Life Cycle (SDLC) is a systematic process for building software that ensures the quality and correctness of the software built. SDLC process aims to produce high-quality software, which meets customer expectations. Software engineering is never a dying product. As long as the human being exists on the earth, the software and their applications are inevitable in their day to life. Consequently, both local and multinational companies need to plan, design, develop and deploy the novel software time-to-time-based on user requirements. That is why the novel software analytics is very essential to predict better software solutions without any manual interventions. Therefore, the Software Engineering Analytics/ Software Analytics is the enriching and upcoming field across the global. Many multinational Information Technology /Information Technology enabled Services companies are trying to develop the new analytical software in which it can automatically identify the errors, auto-mate the software process and make sure integration of submodules. Later this will cause user-level acceptance of the software at the specified time within the budget. Hence, the dawn of software analytics or software engineering is emerging; the academicians, researchers, and corporates have to come together to reap the fruits across the media, banking, finance, entertainment, heavy industries, manufacturing and other IT related software analytics product and services. Hoping that the day is very soon, where we will have software analytics and will take care of our software products in the future for reducing the cost and time. Besides, reducing the complexities in implementing software and error-free. Therefore, it is the last minute all the corporate people should work seriously for the better software analytics. Therefore, in this post-COVID-19 scenario, all the global business sectors need to adopt the software deployment in full-pledged in all departments including human resource, training & development, production, manufacturing, projects, operations, systems, marketing, sales, accounting and finance to name a few for operational efficiency. Moreover, all levels in all industries including transportation, aviation, mining, media, entertainment, healthcare, governance, security, education and special applications. The software deployment process will smoothen the process even during COVID-19 (pandemic) situations like round the clock in all business disciplines. The developments of software analytics or software engineering analytics will definitely help business organizations to surpass any kind of pandemics including COVID-19 to sustain by proper deployment.

References

1. Chan JFW, Yuan S, Kok KH, To KKW, Chu H, Yang J, et al (2020) 333 RWS Poon, HW Tsoi, SKF Lo, KH Chan, VKM Poon, WM Chan, JD Ip, JP Cai, 334 VCC Cheng, H. Chen, CKM Hui, KY Yuen, A familial cluster of pneumonia associated 335 with the 2019 novel coronavirus indicating person-to-person transmission: a study of a 336 family cluster. Lancet 395:514–523
2. Bakkalbasi N, Bauer K, Glover J, Wang L (2006) Three options for citation tracking: Google Scholar, Scopus and Web of Science. Biomed Digit Libr 3(1):7
3. Fabbrini F, Fusani M, Gnesi S, Lami G (2000) Quality evaluation of software requirement specifications. In: Proceedings of the software and internet quality week 2000 conference, pp 1–18
4. Salger F, Engels G, Hofmann A (2009) Inspection effectiveness for different quality attributes of software requirement specifications: an industrial case study. In: 2009 ICSE workshop on software quality. IEEE, pp 15–21
5. Pekar V, Felderer M, Breu R (2014) Improvement methods for software requirement specifications: a mapping study. In: 2014 9th international conference on the quality of information and communications technology. IEEE, pp 242–245
6. Singh Y, Sabharwal S, Sood M (2004) A systematic approach to measure the problem complexity of software requirement specifications of an information system. Int J Inf Manage Sci 15(1):69–90
7. Hassine J, Dssouli R, Rilling J (2004) Applying reduction techniques to software functional requirement specifications. In: International workshop on system analysis and modeling. Springer, Berlin, Heidelberg, pp 138–153
8. Sullivan KJ, Griswold WG, Cai Y, Hallen B (2001) The structure and value of modularity in software design. ACM SIGSOFT Softw Eng Notes 26(5):99–108
9. Shaw M, Garlan D (1996) Software architecture, vol 101. Prentice Hall, Englewood Cliffs
10. Heer J, Agrawala M (2006) Software design patterns for information visualization. IEEE Trans Visual Comput Gr 12(5):853–860
11. Détienne F (2001) Software design–cognitive aspect. Springer Science & Business Media
12. Terwilliger RB, Campbell RH (1989) Please: executable specifications for incremental software development. J Syst Softw 10(2):97–112
13. Rising L, Janoff NS (2000) The Scrum software development process for small teams. IEEE Softw 17(4):26–32
14. Jiang JJ, Klein G, Hwang HG, Huang J, Hung SY (2004) An exploration of the relationship between software development process maturity and project performance. Inf Manag 41(3):279–288
15. Clarke P, O'Connor RV (2012) The situational factors that affect the software development process: towards a comprehensive reference framework. Inf Softw Technol 54(5):433–447
16. Kaner C, Bach J, Pettichord B (2008) Lessons learned in software testing. Wiley
17. Kuhn DR, Wallace DR, Gallo AM (2004) Software fault interactions and implications for software testing. IEEE Trans Software Eng 30(6):418–421
18. Cadar C, Sen K (2013) Symbolic execution for software testing: three decades later. Communications 56(2):82–90
19. Hailpern B, Santhanam P (2002) Software debugging, testing, and verification. IBM Syst J 41(1):4–12
20. Dearle A (2007) Software deployment, past, present and future. In: Future of software engineering (FOSE'07). IEEE, pp 269–284
21. Hall RS, Heimbigner D, Wolf AL (1999) A cooperative approach to support software deployment using the software dock. In: Proceedings of the 1999 international conference on software engineering (IEEE Cat. No. 99CB37002). IEEE, pp 174–183
22. Carzaniga A, Fuggetta A, Hall RS, Heimbigner D, Van Der Hoek A, Wolf AL (1998) A characterization framework for software deployment technologies. Colorado State University Fort Collins Department of Computer Science

23. Bennett KH, Rajlich VT (2000) Software maintenance and evolution: a roadmap. In: Proceedings of the conference on the future of software engineering. ACM, pp 73–87
24. Dishaw MT, Strong DM (1998) Supporting software maintenance with software engineering tools: a computed task–technology fit analysis. J Syst Softw 44(2):107–120
25. Yau SS, Collofello JS (1980) Some stability measures for software maintenance. IEEE Trans Software Eng 6:545–552
26. Bertolino A (2007) Software testing research: achievements, challenges, dreams. In: 2007 future of software engineering. IEEE Computer Society, pp 85–103
27. Buse RP, Zimmermann T (2010) Analytics for software development. In: Proceedings of the FSE/SDP workshop on future of software engineering research. ACM, pp 77–80
28. Buse RP, Zimmermann T (2012) Information needs for software development analytics. In: Proceedings of the 34th international conference on software engineering. IEEE Press, pp 987–996
29. Zhang D, Dang Y, Lou JG, Han S, Zhang H, Xie T (2011) Software analytics as a learning case in practice: Approaches and experiences. In: Proceedings of the international workshop on machine learning technologies in software engineering. ACM, pp 55–58
30. Martínez-Fernández, S., Vollmer, A. M., Jedlitschka A, Franch X, López L, Ram P, et al (2019) Continuously assessing and improving software quality with software analytics tools: a case study. IEEE access
31. Zhang D, Han S, Dang Y, Lou JG, Zhang H, Xie T (2013) Software analytics in practice. IEEE Softw 30(5):30–37.2
32. Abdellatif TM, Capretz LF, Ho D (2015) Software analytics to software practice: a systematic literature review. In: Proceedings of the first international workshop on BIG data software engineering. IEEE Press, pp 30–36
33. Shah A—Senior Management Trainee ERS Practice (2017) Engineering analytics–what is next in software engineering? Accessed dated on 20/6/2019. https://www.hcltech.com/blogs/engineering-analytics-what-next-software-engineering
34. Menzies T, Williams L, Zimmermann T (2016) Perspectives on data science for software engineering. Morgan Kaufmann
35. Menzies T, Zimmermann T (2018) Software analytics: what's next? IEEE Softw 35(5):64–70
36. Hassan AE, Hindle A, Runeson P, Shepperd M, Devanbu P, Kim S (2013) Roundtable: what's next in software analytics? IEEE Softw 30(4):53–56