Closing the Gap: A Comprehensive Analysis of Software Engineering Curriculum and Industry Requirements

[1]Jamilurahman Faizi., [2]Zargay Habibi., [1]Mohmmad Sarosh Umar

[1]Department of Computer Engineering, Aligarh Muslim University, Aligarh, 202001, India

[2]Faculty of Computer Science, Nangarhar University, Jalalabad, 2601, Afghanistan

DOI: https://doi.org/10.51583/IJLTEMAS.2024.130805

Received: 03 August 2024; Accepted: 22 August 2024; Published: 29 August 2024

Abstract: This paper investigates the gap between software engineering education and the industry needs by suggesting solutions to close that gap. The implication is that classrooms agendas are to be more flexible in nature to meet the dynamism in technology needs by the tech sector, focusing on the inclusion of current data science technologies in education programs. This study is backed by data-driven inferences which help to identify the linkage between academia and industry, and with the use of the predictive model of the regression one can estimate the graduation success. The results have proved the importance of practical skills such as research abilities, critical thinking, and problem-solving skills over traditional metrics like GPA. Thus it should be the need of the hour to develop the industry-relevant training in order to provide vocational education to students. The coordination between academia and industry fields by merging student-centric projects that have modern technologies would aid improving adapting software engineering education to the variable industrial sector. The research results emphasize the significance of an active learning process and practical application of the learned concepts that should be employed to get students ready for the challenges awaiting them at the workplace. Finally, a paper that proposes permanent developing of the engineering curriculum and close collaboration between industry and academic institutions, so that the students receive key competences to be prosperous in software engineering. Python has been implemented to analyse the skills of the software engineering curriculum for achieving the requirements of the industry. Data visualisation, data pre-processing, and predictive models have been implemented to gather data based on industrial requirements.

Keywords: Curricula, Industrial Requirements, Industry Academia Collaboration, Software Engineering Education.

I. Introduction

The current world of technology is to a great extent rules by software engineering foundations, which are the reason of inventive solutions on various spheres and implementation of technologies in everyday activities around the globe. With the software industry demands that keep shooting up, it as a matter of fact becomes imperative to have a skilled workforce who is capable of meeting emerging industry needs [3]. Yet, apart from this, the biggest problem is in the fact that education posed in institutes of education should incorporate the very actual needs of the software engineering job market. This exploratory effort mainly considers the gap between current software engineering curriculum requirements and industry demands and seeks to address it. The focus of this research is to help determine the gap between the current state of software engineering education and the industry's needs, which it does by first looking into the problems in the current standard and then suggesting strategies for fixing the problems [4]. The rise in new technologies and approaches to the software engineering has brought about this different style people design, deploy and support the software. Agile frameworks, DevOps practices, cloud computing, artificial intelligence, and cybersecurity are the major components of current software engineering ecosystem which implies, among others, the need for re-examination of the very notion of software engineering education as well as the ways to cope with constantly changing requirements imposed on the IT professionals [5]. A gulf remains; classrooms on one side, some distance away from, the corporate world on the other. Universities incur difficulty when the curricula cannot swiftly meet the fast-growing innovative technologies or changed preferences of the employers. Consequently, the newly graduated students do not have the relevant hard and soft skills required by employers. Also, in order to attain this knowledge, the committee will survey software engineering professionals as well as relevant industry stakeholders to find out the main properties and skills that are most valued in the job market. This study points out the commonalities and variations between education and industry and thus formulate the basis for the implementation of changes in the curriculum, teachers' practice, policy-makers and industries.

II. Literature Review

A. Alignment of Software Engineering Curriculum with Industry Demands

The wide and exhaustive research have proven that software engineering curriculum should have its accompaniment ready constantly with very specifics of the fast-changing requirements. Kaur and Kumar enormously underlined the impact of connecting the knowledge and skill that the academic programs offer with industry application to bridge the gap between the theory and practice. They underline systematization of curriculum that comprises up-to-date technologies, methodologies and projects related to work-life to meet the demand of students who are able to achieve job success in volatile software engineering environment. At the same time, [6] pointed out a relevant feature of a competency-based curriculum design, becoming crucial for

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue VIII, August 2024

the industry stakeholders to act as core determiners for learning outcomes and program content [7]. Their research showed what a great result can be achieved by creating curricula in cooperation with the industry. It is not only the contribution to the development of students' innovative thinking and entrepreneurship, but it also facilitates the coming out of innovations and initiatives among academia.

B. Data-driven Insights Integration as a Part of the Software Engineering Curriculum.

The literature, particularly in the context of education, usually focuses on a potent integration of some data-driven analysis methods with software engineering education. When we look at the technology landscape we can see more and more data-oriented technological advancements and educators are not part of this. They, along with others, are exploring ways to incorporate machine learning, data analytics, etc. into the school curriculum. [8] put forward a framework for learning the concepts and data science skills required for software engineering sessions that involve lots of practice and real-world applications. Their investigation gave insights concerning the benefits of data-cantered techniques for sharping the students' problem solving skills and decision-making capacity when dealing with soft development projects. The experts namely found that there is a big divide between the different approaches to teaching data science in software engineering programs [9]. Through experimentation on the effectiveness of project-based learning and experiential strategies they realized that those methods are much highly effective than the traditional methods in terms of engagement and learning retention. These findings give pragmatic reason for studying integrated data-driven analysis to be held in software engineering curricula as a way to beef up students for modern data-driven environment.

III. Methods and Materials

Data Collection

Intermediate data from Kaggle, a well-known relay for varied datasets across diverse domains, like software engineering, shall be employed. The dataset will include data on how to predict the future role of software developers and the skills they possess, the principal resource for this study.

Data Pre-processing

Upon collection, the data-set will be subject to the pre-processing-step to confirm its quality and compatibility. This will imply imputing missing values, spotting the wrong spreads, and then converting a useful set of data [10]. Python, besides libraries like Pandas and NumPy will be among the tools used in data manipulation tasks for data management efficiency. Categorized data will be initially converted so that further statistical analyses can be performed.

Data Analysis

The analysis of the software engineering curriculum and its alignment with industry requirements will encompass several key steps:

- **Exploratory Data Analysis (EDA)**

In order to get a better understanding of the nature of the underlying dataset, exploratory data analysis will be undertaken to identify the distribution, relationships and various patterns among its constituents [11]. Visualization techniques using histograms, count plots, and correlation matrices will be applied to identify the first essential features of the data. The figures will help point out patterns and see connections among the variables which will be assisted to yes these data for deeper analysis.

- **Regression Analysis**

Regression analysis is going to be used to create a model describing the relationship between the variables in the dataset, having paid most attention to those variables that are relevant to the software engineering's curriculum including the ability to predict the outcomes [12]. Two types of regression models will be utilized:

- Linear Regression: It is model that will be used to prophesy a Student's GPA using their skills or any other supposed independent variable. Linear regression model performance will be evaluated using metrics such as MSE and R-squared to measure the degree of dispersion and goodness-of-fit of the model.
- Ridge Regression: Besides linear regression, ridge regression model will be added to this study to increase the forecasting accuracy with the GPA. The weight frequency will be 1, and the model will be evaluated in connection with linear regression to confirm its accuracy.

Evaluation of Results

The data thus retrieved will be analysed and assessed. The conclusions that will be drawn will focus on the correlation between the software engineering curriculum and industry's needs [13]. Models, visual stat and among others will be used to evaluate the usefulness of the curriculum by way of providing students with the necessary skills needed to tackle the ranging demands of the industry.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue VIII, August 2024

Discussion

The outcomes will be examined within the frame of context of their import for software education, engineering, and industry practices. Among the main issues addressed we will see the issue regarding the importance of specific skills, the effectiveness of the teaching, fastening and the places for progress will be discussed [14]. Furthermore, the limits of the study will be discussed also and some possibilities for future research directions will be given to bring a general view of the research.

Table 1 Task and Descriptions

Task	Description
Missing Values	Handled using appropriate imputation techniques
Outlier Detection	Identified and treated using statistical methods
Data Transformation	Converted categorical variables for analysis purposes

IV. Results and Discussion

Result

Software engineering curriculum plays a pivotal role in improving the requirements of the industry and effective skill can easily determine the industrial trends. The carrier path prediction dataset has been collected from effective data collection sources such as Kaggle for analysing the comprehensive analysis of the software engineering curriculum activities in the industrial requirements. In the dataset, several kinds of skills of software development have been analysed to meet the requirements of the industry. This study helps in determining the requirements of software engineering for improving the functional activities of the industrial requirements. Software engineering programme helps in evaluating the requirements of the industry based on effective skills and knowledge. Software engineering curriculum helps in managing the design process, control systems, maintenance, as well as the testing process that improves the technological operation in the industry [1]. In this case, project-based learning, online learning, as well as lecture-based learning are used to improve the software engineering curriculum that allows for meeting the requirements of the industry. Hence, Python programming language has been used to analyse the software engineering curriculum to gather data regarding technological trends.

	GPA	Extracurricular_Activities	Internships	Projects	Leadership_Positions	Field_Specific_Courses
count	101.000000	101.000000	101.000000	101.000000	101.000000	101.000000
mean	3.729678	4.544554	0.950495	1.940594	0.495050	4.633663
std	0.708606	2.783971	0.829171	1.420013	0.502469	2.663542
min	2.536451	0.000000	0.000000	0.000000	0.000000	0.000000
25%	3.044955	2.000000	0.000000	1.000000	0.000000	2.000000
50%	3.739830	5.000000	1.000000	2.000000	0.000000	5.000000
75%	4.350802	7.000000	2.000000	3.000000	1.000000	7.000000
max	4.926268	9.000000	2.000000	4.000000	1.000000	9.000000

Fig. 1. Explaining data description

Fig. 3. Explaining data description

The above data description has been determined for gathering data based on the collected dataset and the statistical parameters have been used to determine the trends, as well as patterns of the dataset.

```
curriculum.isnull().sum()

Field                          0
Career                         0
GPA                            0
Extracurricular_Activities     0
Internships                    0
Projects                       0
Leadership_Positions           0
Field_Specific_Courses         0
Research_Experience            0
Coding_Skills                  0
Communication_Skills           0
Problem_Solving_Skills         0
Teamwork_Skills                0
Analytical_Skills              0
Presentation_Skills            0
Networking_Skills              0
Industry_Certifications        0
dtype: int64
```

Fig. 2. Evaluating the null values

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue VIII, August 2024

In the curriculum dataset, there are no null values for analysing the software engineering curriculum in the industry requirements.

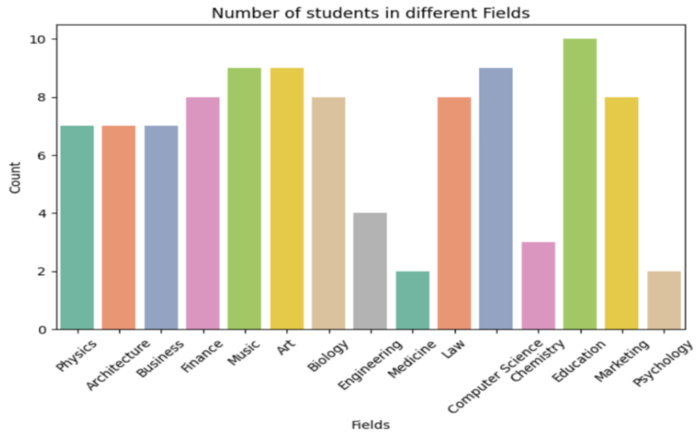

Fig. 3. Analyzing the several fields in the graphical format

The above count plot has been used for determining the field of the software engineer and it has been seen that the students of different fields are interested in the software engineering curriculum [15]. This curriculum helps in enhancing the functionality and meeting the industrial requirements. The figure shows that diversity of student's interest and background in various fields contributes to the software engineering curriculum that helps in meeting the requirements of the industry. Based on the industry demand, varied interest and background of the students in software engineering curriculum can close the gap between academia and industry.

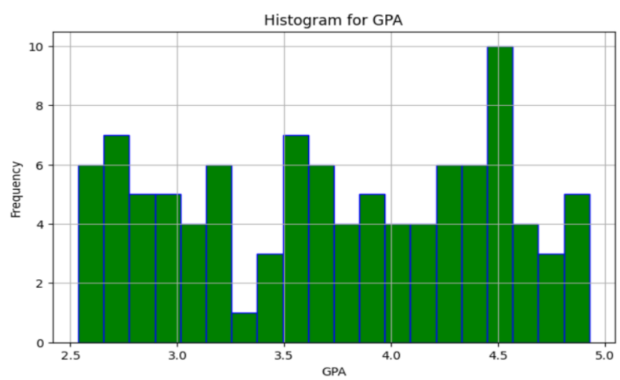

Fig. 4. Histogram Plot

The above histogram plot has been used for determining the data distribution of GPA and most of the students have more than 4 GPA. Hence, it can be said that the students are interested in developing software curriculum activities for achieving the requirements of the industry.

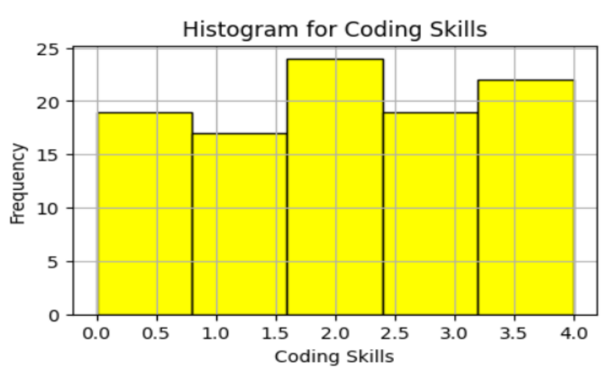

Fig. 5. Analyzing the coding skills of the students

In meeting the requirements of the industry, skills play an effective role and the software engineering curriculum can fulfil the needs. Most of the students have good coding skills that allow for improving the performance and quality of industrial functionalities [17]. Coding skill of the student acts as a bridge that helps in closing the gap between academia and industry. The

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue VIII, August 2024

plot shows that the software engineering curriculum design can develop the coding skills for achieving the industrial requirements. Hence, the curriculum activities prepare the students for closing the gap between industry expectations and educational training.

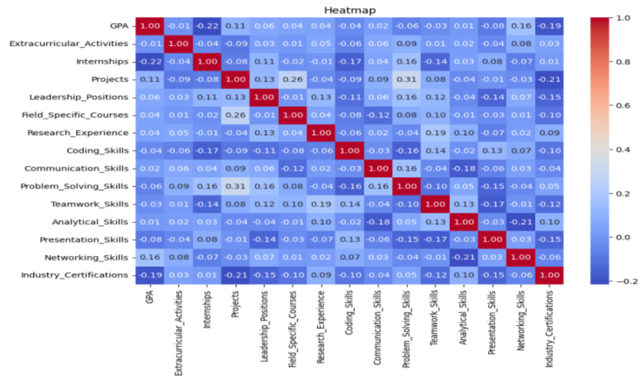

Fig. 6. Correlation analysis

The above figure has been used to find the relationship between the data variables. A positive correlation value denotes a strong relationship between the variables, as well as a negative correlation value indicates a weak relationship between the variables [18]. Hence, the Python programming language can easily evaluate the relationship among the data variables of the carrier obtained dataset.

Data pre-processing

```
[12] curriculum['Field'] = curriculum['Field'].astype("category").cat.codes
```

Fig. 7. Data pre-processing

The Data pre-processing method is used to remove the irrelevant values in the dataset to enhance the accuracy, as well as efficiency [2]. In this case, the field column has been used for changing the data types and the categorical data type allows for collecting the outcomes.

Test-train split

```
x = curriculum.drop(columns=['GPA', 'Career'])
y = curriculum['GPA']
```

```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.25, random_state=42)
```

Fig. 8. Performing test-train

GPA column has been selected as the target column and various skills are determined in the performance of the students. 25% of test data has been collected as the test dataset that allowed for collecting the predicted GPA based on the software engineering curriculum [16]. In this case, python library functions have been implemented to split the test and train the dataset to analyse the gap between industry expectations and curriculum.

Linear Regression

```
from sklearn.metrics import classification_report,mean_squared_error, r2_score
from sklearn import datasets, linear_model
```

```
[21] linear_reg = linear_model.LinearRegression()
     linear_reg.fit(x_train, y_train)

     ▾ LinearRegression
     LinearRegression()
```

```
[22] lr_y_prediction = linear_reg.predict(x_test)
     print("Coefficients: \n", linear_reg.coef_)

     Coefficients:
      [ 0.02264411 -0.02614093 -0.1338119  -0.00143429 -0.11139708 -0.00488205
      -0.02921132 -0.10486536  0.01986804 -0.05148992 -0.01694826 -0.04509467
      -0.114166    0.05512381 -0.24576422]
```

```
[23] print("Value of mear square error is: %.2f" % mean_squared_error(y_test, lr_y_prediction))
     print("Coefficient of determination value is: %.2f" % r2_score(y_test, lr_y_prediction))

     Value of mear square error is: 0.82
     Coefficient of determination value is: -0.39
```

Fig. 9. Gathering data based on linear regression analysis

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue VIII, August 2024

The above figure demonstrates the predictive outcomes of GPA based on linear regression models. The mean square error of the linear regression model is 0.82 and the coefficient of determination is negative.

Ridge Regression

```
[24] from sklearn.linear_model import Ridge
     ridge_reg = Ridge(alpha=1.0)
     ridge_reg.fit(x_train, y_train)
     ridge_y_prediction = ridge_reg.predict(x_test)

[25] print("Coefficients values are: \n", ridge_reg.coef_)

     Coefficients values are:
      [ 2.23766130e-02 -2.62275762e-02 -1.31474054e-01 -1.18939274e-04
      -1.02459709e-01 -4.75757190e-03 -2.89013204e-02 -1.02732392e-01
       1.98900379e-02 -5.18221314e-02 -1.60924093e-02 -4.53166948e-02
      -1.11836451e-01  5.51544988e-02 -2.29392938e-01]

[26] print("Mean squared error value: %.2f" % mean_squared_error(y_test, ridge_y_prediction))
     print("Coefficient of determination value: %.2f" % r2_score(y_test, ridge_y_prediction))

     Mean squared error value: 0.82
     Coefficient of determination value: -0.38
```

Fig. 10. Collecting outcomes based on the ridge regression model

In the ridge regression, the alpha value has been selected as 1 and the linear model function in “Python” has been implemented to predict the GPA value based on the software engineering curriculum dataset. Similarly, the MSE value of ridge regression is 0.82 and the value predicts the effective outcomes of the software engineering. The proactive approach helps in gathering data based on the academia outcomes and real world demand. Hence, the advanced statistical technique helps in forecasting the outcomes regarding the software engineering curriculum. It has been noticed that GPA scores cannot decide the skills of the students.

Discussion

The result section of the analysis allows for gathering data regarding the software engineering curriculum in managing the entire industrial requirements. Additionally, the software developer is the career path that helps in shaping the requirements of the industry. An effective data source has been used for collecting information for analysing the insights into knowledge and skills requirements in the industry. The data visualisation has been performed to analyse the details of the skills requirement in improving the operational activities of the industry that close the gap between software engineering curriculum and industry demands. Hence, the coding and analytical skills of the student help in prosperous the future of the software developer [19]. The factors allow for enhancing the efficiency of the industry and “Python programming language” evaluates the required skills of the software engineer. Correlation analysis has been used to determine the relationship between the effective skills and GPA of the software developer. Analytical skills are necessary for identifying issues of the industry and the data interpretation method is used to describe the patterns of the industry requirements.

Predictive models have been implemented to gather information regarding the GPA of the students. Hence, regression models are implemented in collecting data for describing the relationship between skills and GPA. In such circumstances, the linear regression model can easily evaluate the predictive data based on the test dataset. The mean square value of the linear regression model is 0.82 and the value is close to 1. However, the R-squared value of the predictive model is negative and the model cannot describe the effective relationship between the variables. Moreover, the ridge regression model has a positive value in MSE and the negative value indicates that the predictive model did not predict appropriate GPA regarding the software engineering curriculum. Therefore, it can be said that GPA cannot analyse the skills of students while various skills in software engineering curricula improve the industrial requirements.

V. Conclusion

The main idea of this paper was to indicate how the programming curriculum is integrated with the ever-changing needs of the industry and the application of data science techniques into school curriculum. The targeted analysis inherently brings out the vital role of a dynamic and evolving curriculum for learning programming skills which is a crucial essential in the workplace environment and is full of challenges as well as opportunities in the software engineering trade. By means of a close analysis of the software engineering curriculum versus the industry needs, we realized that there is a demand for a joint effort that will help to fill in this gap between theoretical knowledge and practical experience involving universities and industry relevant partners. Research on competency-based course development and industry-academic collaboration puts stress on the necessity to apply real-world projects, advanced technologies and university-business agreements when doing coursework program. Also, data driven analysis techniques have taken quite an advantageous role in the implementation of programing education standards and the students are therefore learning how to apply the same to solve problems and meet the industry’s data-oriented demands. Although research literature had focused on the validity of hands-on learning and project-based approaches, it had nevertheless shown that for learning to be deeper and more effective, learners must engage actively and not rely on instructors as providers of information only. Besides, present situation is an evidence that curriculum development, pedagogies, and collaboration with industry must play a significant role in ensuring that the software engineering education keeps pace with the modern trends and technologies. Through welcoming (incorporating?) continuously advancing technologies, adapting to emerging technologies and

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue VIII, August 2024

building strong partnerships with the private sector, educational institutions can equip students with the essential skills, knowledge and competencies that the field of software engineering calls for in the never-ending changes.

References

1. Yu, W., Patros, P., Young, B., Klinac, E. and Walmsley, T.G., 2022. Energy digital twin technology for industrial energy management: Classification, challenges and future. Renewable and Sustainable Energy Reviews, 161, p.112407.
2. Fan, C., Chen, M., Wang, X., Wang, J. and Huang, B., 2021. A review on data preprocessing techniques toward efficient and reliable knowledge discovery from building operational data. Frontiers in energy research, 9, p.652801.
3. Cico, O., Jaccheri, L., Nguyen-Duc, A. and Zhang, H., 2021. Exploring the intersection between software industry and Software Engineering education-A systematic mapping of Software Engineering Trends. Journal of Systems and Software, 172, p.110736.
4. Raji, I.D., Smart, A., White, R.N., Mitchell, M., Gebru, T., Hutchinson, B., Smith-Loud, J., Theron, D. and Barnes, P., 2020, January. Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In Proceedings of the 2020 conference on fairness, accountability, and transparency (pp. 33-44).
5. Shneider man, B., 2020. Bridging the gap between ethics and practice: guidelines for reliable, safe, and trustworthy human-centered AI systems. ACM Transactions on Interactive Intelligent Systems (TiiS), 10(4), pp.1-31.
6. Neumann, W.P., Winkel haus, S., Grosse, E.H. and Glock, C.H., 2021. Industry 4.0 and the human factor–A systems framework and analysis methodology for successful development. International journal of production economics, 233, p.107992.
7. Zhao, X., Cornish, K. and Vodovotz, Y., 2020. Narrowing the gap for bioplastic use in food packaging: an update. Environmental science & technology, 54(8), pp.4712-4732.
8. Wang, B., Tao, F., Fang, X., Liu, C., Liu, Y. and Freiheit, T., 2021. Smart manufacturing and intelligent manufacturing: A comparative review. Engineering, 7(6), pp.738-757.
9. Ouhbi, S. and Pombo, N., 2020, April. Software engineering education: Challenges and perspectives. In 2020 IEEE Global Engineering Education Conference (EDUCON) (pp. 202-209). IEEE.
10. Bedué, P. and Fritzsche, A., 2022. Can we trust AI? An empirical investigation of trust requirements and guide to successful AI adoption. Journal of Enterprise Information Management, 35(2), pp.530-549.
11. Geirhos, R., Narayanappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F.A. and Brendel, W., 2021. Partial success in closing the gap between human and machine vision. Advances in Neural Information Processing Systems, 34, pp.23885-23899.
12. Kasauli, R., Knauss, E., Hork off, J., Liebel, G. and de Oliveira Neto, F.G., 2021. Requirements engineering challenges and practices in large-scale agile system development. Journal of Systems and Software, 172, p.110851.
13. Lu, Y., Xu, X. and Wang, L., 2020. Smart manufacturing process and system automation–a critical review of the standards and envisioned scenarios. Journal of Manufacturing Systems, 56, pp.312-325.
14. Giray, G., 2021. A software engineering perspective on engineering machine learning systems: State of the art and challenges. Journal of Systems and Software, 180, p.111031.
15. Bodkhe, U., Tanwar, S., Parekh, K., Khanpara, P., Tyagi, S., Kumar, N. and Alazab, M., 2020. Blockchain for industry 4.0: A comprehensive review. IEEE Access, 8, pp.79764-79800.
16. Zhang, A., Wang, J.X., Farooque, M., Wang, Y. and Choi, T.M., 2021. Multi-dimensional circular supply chain management: A comparative review of the state-of-the-art practices and research. Transportation Research Part E: Logistics and Transportation Review, 155, p.102509.
17. Hutchinson, B., Smart, A., Hanna, A., Denton, E., Greer, C., Kjartansson, O., Barnes, P. and Mitchell, M., 2021, March. Towards accountability for machine learning datasets: Practices from software engineering and infrastructure. In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (pp. 560-575).
18. Mian, S.H., Salah, B., Ameen, W., Moiduddin, K. and Alkhalefah, H., 2020. Adapting universities for sustainability education in industry 4.0: Channel of challenges and opportunities. Sustainability, 12(15), p.6100.
19. Kumar, R.K., Antunes, M.J., Beaton, A., Mirabel, M., Nkomo, V.T., Okello, E., Regmi, P.R., Reményi, B., Sliwa-Hähnle, K., Zühlke, L.J. and Sable, C., 2020. Contemporary diagnosis and management of rheumatic heart disease: implications for closing the gap: a scientific statement from the American Heart Association. Circulation, 142(20), pp. e337-e357.