Data Monetization Service Development Using Iterative Lifecycle Framework, Quality Assurance, and Open Web Application Security Project: A Case Study of a Utility Company in Indonesia

Wahyu Haris Kusuma Atmaja[1*], Harco Leslie Hendric Spits Warnars[2], Ford Lumban Gaol[3], and Benfano Soewito[4]
[1−3]Computer Science Department, BINUS Graduate Program – Doctor of Computer Science, Bina Nusantara University
Jakarta, Indonesia 11480
[4]Computer Science Department, BINUS Graduate Program – Master of Computer Science, Bina Nusantara University
Jakarta, Indonesia 11480
Email: [1]wahyu.atmaja@binus.ac.id, [2]shendric@binus.edu, [3]fgaol@binus.edu, [4]bsoewito@binus.edu

***Abstract*—The research aims to provide Data Monetization (DM) services for an Indonesian utility company as a pilot to generate additional revenue beyond the primary operation. The service is built using an iterative development lifecycle framework and evaluated based on five Quality Goals (QGs), including application and security testing activities. The framework includes methods for processing and modeling electricity usage data, testing application quality, checking infrastructure quality, and ensuring access security for front-end and back-end applications using the Open Web Application Security Project (OWASP). For data modeling, Support Vector Regression (SVR) is used, and it outperforms Polynomial Regression (PR) and Multi-Layer Perception (MLP) Neural Networks. Furthermore, QG shows strong performance with an Relative Root Mean Squared Error (RRMSE) value $< 10\%$, high forecasting ability with Mean Average Probability Error (MAPE) $< 10\%$, and a near-zero average error rate (Mean Squared Error (MSE)) square using minimal data from four months. The services go through functional and integration test to ensure product quality and application performance, which results in a minimum of 95% service response in throughput, 0.128 seconds for processing 2,000 requests, stability at 300–500 in one second per hour, and 7–21 seconds during peak hours. Additionally, the service passes nine penetration tests and ten vulnerability assessments using the OWASP top 10:2021 category. Based on the comprehensive testing and evaluation results, both the application and the service are considered ready and secured for deployment.**

***Index Terms*—Data Monetization, Service Development, Iterative Lifecycle Framework, Quality Assurance, Open Web Application Security Project (OWASP), Utility Company**

I. Introduction

INNOVATION is a long-term strategy adopted by utility companies in Indonesia, particularly those focusing on the electricity business, to generate additional revenue beyond the core operations. An example of this initiative is monetizing transaction data that are automatically collected and processed to provide customers' profiles, power usage reports, and energy transaction payments. The data can be used in many fields, such as finance, marketing, government, and others [1]. Potential products or services offered through Data Monetization (DM) include credit rating and scoring, market analysis, targeted campaigns, spending analysis, demographic surveys, and many other initiatives related to consumer or retail [1]. These deliverables can be developed because the Customer Relation Management (CRM) system records historical transaction data from the first application of the customer to the present. Typically, these services or products show processed or aggregated data rather than raw transaction data, as permitted by the utility company [2].

Received: July 22, 2023; received in revised form: March 12, 2024; accepted: March 13, 2024; available online: Sep. 19, 2024.
*Corresponding Author

DM is defined as "the use of statistical information or data-processed discoveries that bring quantifiable economic or scientific benefits to a corporation or research center" [2]. Additionally, two main challenges in privacy-protecting DM systems are identified. The first challenge is providing statistical data without compromising the privacy of an individual [3]. The second challenge is determining how to market data collected from customer interactions by committing not to use the product or giving the customer control over the data [2].

According to the research conducted by Parvinen, the company has two options for monetization: improving existing processes and products for internal use and generating new forms of offerings for external customers [4]. It is consistent with previous research findings, supporting the idea of commercializing the electrical company's data.

A company can monetize four types of data: a production system that automatically collects digital data usage, customer data from digital interactions, digital sales, and delivery channels [4]. In this context, examples of DM include daily, weekly, or monthly power usage of electricity customers, along with payment transactions [4]. In the research context, data are collected from the CRM system and converted into Online Analytical Processing (OLAP) or data mart format. The data are then processed, aggregated, classified, clustered, or used for predictions using various methods such as fuzzy logic algorithms, machine learning methods, or other formulas. The processed data are then loaded into the presentation or integration layer of Application Programming Interface (API) data management. Customers can access the data through a platform or via secure data communication to use an application or API management. In some cases, customers may incorporate recommendations or justifications from the processed results into a personal system [5].

The research shows the innovative process of monetizing published data, focusing on the importance of protecting individual privacy, ensuring high accuracy and precision in data content, and safeguarding data, connections, as well as containers from vulnerabilities. Therefore, the research aims to propose an incorporated method to DM that starts with selecting a method based on scientific data. The next phase includes implementing and testing service quality and deploying in a dependable and secure security layer infrastructure. Some of these selections evaluated in previous studies [4, 6–8] describe how the methods are obtained using machine learning.

II. Research Method

DM methods are divided into four major procedures: data processing, quality targets, quality assurance activity plan, and security test method. The development of DM services is repetitive because requirements are synthesized and collected from clients before the team moves on to analysis and design. After development, quality assurance, training, and implementation begin, the output is presented and authorized by the consumers.

A. Data Processing Method and Formulation

During the iterative phase, the customer and project team have engaged in three primary actions. These actions included use case discussions, data gathering, and investigation, as well as data analytics and modeling. As part of the research, five models are collected and re-recorded for presentation to the user. These models include 1) predicting energy usage in households with a complete dataset [9], 2) forecasting energy usage with a missing or incomplete dataset [10], 3) estimating energy usage from residential household usage patterns [11], 4) predicting energy usage of commercial or industrial buildings [12], and 5) estimating energy usage of government or institutional buildings [13], and in the electricity grid [14].

The data model is built using customer transaction data from utilities around the nation in 2020-2021 for residential, industrial, business, government, and social client sectors, both postpaid and prepaid. Sanitization is conducted to exclude transactions that do not occur in three months or a consistent manner. Following this process, the cleaned data obtained from these criteria are subsequently assembled into a dataset. Transactions that are not continuous are further filtered to ensure a minimum of nine months of consistent activities. As a result, the clean dataset is then analyzed for monthly and annual usage by applying a normal distribution in the Interquartile Range (IQR). The data with usage outside IQR are excluded to ensure accurate analysis.

The research proposes two models in analysis and design modeling, including the general electrical imputation method and machine learning. In this instance, machine learning is accepted by users because the historical data collected range from 7 to more than 24 months. Consequently, the dataset is processed using a machine learning model, as supported by several explorations that use this method to analyze energy consumption. Then, the candidates of machine learning methods are SVR [6, 7, 15], Polynomial Regression (PR) [8, 16], Kernel Ridge Regression (KRR) [17, 18], and Multi-Layer Perception (MLP)

Neural Network [19, 20]. These methods are also available to users who approve the models.

According to the use case discussion, consumers require a system that predicts energy usage for the coming month as well as the invoice amount based on energy consumption. This prediction is determined by processing historical data on energy usage over a specific time. As a result of the five initial models proposed, users agree to anticipate energy consumption in commercial, industrial, and institutional facilities. The method is divided into two steps: training and validation. Following this discussion, customers provide data as input for training activities. User ID and kWh history usage are then analyzed using multiple models, including SVR, PR, KRR, and MLP. After this process, the Mean Average Probability Error (MAPE) score is used to compare the four models, followed by the outcome, which depends on customer ID and presents the predicted kWh consumption. Then, testing is conducted similarly to the training process but with a 30% training dataset. The results are then compared to the kWh usage output associated with each customer ID.

B. Machine Learning Methodology

SVR, a model regression of the Support Vector Model (SVM) developed by Cortes and Vapnik in 1995, is the best model for predicting energy consumption because it is based on statistical learning theory and the concept of structural risk reduction [21]. SVR is suitable for a general energy consumption model and has the highest accuracy in predicting energy usage for renewable energies such as wind power [22]. SVR is a supervised training method that builds an input-output mapping from the data training set and produces a function estimator as shown in Eq. (1) [6]. It shows $f(x)$ as the decision function for predicting the class. Then, $w \epsilon \mathbb{R}^n$ represents a weighted feature vector with n features, and b is the intercept (bias). Moreover, $\phi(x)$ represents the non-linear mapping function, and x is the input factor $\mathbb{R}^n$, which includes both outside and indoor environment factors.

$$f(x) = w^T(x) + b. \tag{1}$$

PR is defined as "a regression model that assesses the nth degree of the connection between dependent and independent variables" [23]. This model is defined in Eq. (2). It has y as the dependent variable, x as the independent variable, $\beta_0 \cdots \beta_h$ as the coefficients or regression terms, h as the degree of the polynomial, and ε as the error or disturbance or noise term.

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_h x^h + \varepsilon. \tag{2}$$

PR is a common method for estimating power usage, with one of its applications being the prediction of the power consumption model of data center servers. In this use case, the polynomial model shows an error of less than 4% [24]. Another application of the model includes approximating vector position in the neighborhood, a_t point a_0 using a low-order polynomial (say, q), that is fitted using only points in the same neighborhood of a_0 [8]. The formula for this calculation is shown in Eq. (3). Then, Weighted Least Square estimates Parameter $\hat{a}_j$ by minimizing Eq. (4). It shows that $K\delta(\alpha)$ weights Kernel function, $(a_t - a_0)$ represents the difference between the vector position at a_t and the reference point a_0, and $(a_t - \hat{a}_t)^2$ represents squared residuals. Then, it has δ as the bandwidth, a_0 as the size of the neighborhood around and locally start approximation, and a_t as vector position in the neighborhood.

$$\hat{a}_t = \sum_{(j=1)}^{q} \hat{a}_j (a_t - a_0)^j, \tag{3}$$

$$\sum_{(t=1)}^{N} (a_t - \hat{a}_t)^2 K\delta(\alpha)(a_t - a_0). \tag{4}$$

MLP is a type of Artificial Neural Network (ANN) known as feed-forward neural network including neurons that are completely attached with weighted connected properties. In addition, a single or ensemble MLP model is trained using a set of input-output pairs to learn the correlation or dependency model between the input and output [9]. MLP is used in combination with Softmax Ensemble Network (SENet), a bagging ensemble of MLP subnetworks, to predict energy usage. It is compared to various imputation methods such as Linear Interpolation, Historical Average, and Optimally Weighted Average (OWA). In most situations, MLP with SENet produces the lowest MAPE and Root Mean Squared Error (RMSE) values of any imputation method [10]. However, there is one instance where OWA produces lower MAPE and RMSE [10]. The previous research compares the model to other machine learning algorithms, such as Linear Regression, Random Forest, Adaptive Boosting, Gradient Boosting Machine, XGBoost, Single Vector Regression, Single MLP (MLP-S), MLP ensemble with subnetwork average (MLP-AVG), Convolution Neural Network, and Recurrent Neural Network. Following the comparison, the outcomes are all MLP-based, with MLP-SENet, MLP-S, and MLP-AVG having the lowest MAPE and RMSE values [10]. As a result, MLP is a strong choice to be used in this experiment.

C. Modelling Scenario

In the experiment, each CUSTOMERID is assigned a scenario including past customers' invoices from periods of 12, 9, and 6 six months, and the amount of data is 1,349 records. Each scenario is separated into two circumstances, namely training and testing, which are compared as follows. Then, each CUSTOMERID is processed using the most effective model of the four machine-learning methods previously mentioned.

1) 12 months record = 11:1
2) 9 months record = 8:1
3) 6 months record = 5:1

D. Evaluating the Machine Learning Results

Each machine learning output is then evaluated using many performance indicators. First, MAPE is a performance metric in regression [25]. Previous research recommends tasks with more sensitivity to relative fluctuations than absolute variations [26]. However, in another previous research, MAPE is insufficient for forecasting models with significant errors because it is biased toward low projections or struggles with small or zero denominators [27]. As a result, the dataset is normalized before it is used in the training or testing phases. The ideal value is close to zero, while the worst is between $+$ and ∞ [25]. The formula of MAPE (M) is in Eq. (5). It consists of M as MAPE, n as the number of times that the summation occurs, A_t as the actual value, and F_t as the forecast value.

$$M = \frac{1}{n}\sum_{t=1}^{n}|\frac{A_t - F_t}{A_t}|. \tag{5}$$

Second, according to previous research, Mean Average Error (MAE) is derived from a measure of average error and is used to evaluate the multivariate regression model [28]. In addition, MAE is used when the corrupted regions of the dataset are in the outlier [25]. The exploration also shows that MAE does not penalize training outliers, leading to unsatisfactory model performance when the data contain a large number of outliers [25]. Relating to this discussion, the best MAE values are near 0, while the worst are $+$ to ∞ [25]. MAE (M) formula assesses the amount of the absolute differences between n anticipated vectors of x and the actual vector y [28], as shown in Eq. (6). It has M as MAE, n as the number of times that the computation is replicated, y_i as the actual value, and x_i as the forecast value.

$$M = \frac{1}{n}\sum_{i=1}^{n}|y_i - x_i|. \tag{6}$$

Third, the formula for Mean Squared Error (MSE) is the quadratic value of the average size of n predicted vectors of y and n actual vectors of y [28]. The model error increases since the formula has squared values [25]. Another previous mentions that MSE is used to find outliers. In the research context, the best value of MSE (M) is the same as MAE [25]. In Eq. (7), M represents MSE, n is the number of times that the summing occurs, y represents the actual value, and $\hat{y}$ is the forecast value.

$$M = \frac{1}{n}\sum_{i=1}^{n}(y - \hat{y})^2. \tag{7}$$

The fitness value (f) of the data is calculated from MSE by applying the real MSE value [29] to an ensemble using the suitable method. Following this process, the lowest MSE achieves the best fitness value. Equation (8) shows f as the fitness value, MSE as the MSE value, and ε as a small number to avoid the division to zero.

$$f = \frac{1}{MSE + \varepsilon}. \tag{8}$$

Fourth, RMSE is related to MSE value, and the best value of RMSE is the same as MSE [25]. It shows R as RMSE, n as the number of times that the summation is repeated, y as actual value, and $\hat{y}$ as forecast value (Eq. (9)).

$$R = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y - \hat{y})^2}. \tag{9}$$

Fifth, Relative Root Mean Squared Error (RRMSE) is based on the MSE formula, but RRMSE aims to identify each residual (relative) value that is scaled against its true value [30]. Therefore, in regression situations, the model separates the generalized error into various components or as a weighted function to maximize ensemble integration [30]. Relating to this discussion, the formula RRMSE is in Eq. (10). It consists of R as RRMSE, n as the number of times that the calculation occurs, y as actual value, and $\hat{y}$ as forecast value.

$$R = \sqrt{\frac{\frac{1}{n}\sum_{i=1}^{n}(y - \hat{y})^2}{\sum_{i=1}^{n}(\hat{y})^2}}. \tag{10}$$

Sixth, In 1921, the Coefficient of Determination (R^2) was proposed by Wright [31]. This model measures the effects of the independent variable on the dependent variable using proportions of variance [25]. For centuries, R^2 has been a valuable tool in research that offers detailed definitions and features, refines interpretations, treats specific cases, and provides ad hoc alterations [25]. The best value of R^2 is +1 as well and the worst is $-\infty$. The formula for this model is in Eq. (11). It has r as the correlation coefficient,

R as r^2, n as the number of times that the addition is repeated, x as the actual value, and y as the forecast value.

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}}. \quad (11)$$

E. Quality Goals

After determining the DM method, the research establishes quality objectives with the user as part of the requirement process. These aims are initially defined in qualitative terms and converted to a quantitative measurement to be used as objective criteria during system development and testing. Moreover, the method helps to create useful and user-friendly applications from the start by identifying and resolving issues early [32]. The practical and easy-to-use application simplifies the operation and maintenance phases.

Based on the findings, quality targets are set to show the high quality and secure aspects of DM. The result is obtained from a Request for Proposal (RFP) or software requirement specification and removed from the performance, time, cost, efficiency, or service level attribute, which all team members use as a benchmark. Additionally, the Quality Goal (QG) is approved by the user as a critical document. This method follows best practices in requirement engineering, which includes understanding detailed necessities (content dimension), reaching agreements about needs among relevant stakeholders (agreement dimension), and documenting requirements according to specific formats and rules (documentation dimension). In the context of the exploration, these dimensions are described in previous research [33].

F. Activity Plan

After determining the quality plan, the project activity plan is implemented. This plan includes schedule reviews, software tests, acceptance examinations, and configuration management [34]. Previous research describes a test environment workbench lifecycle to improve the effectiveness of software testing, providing a detailed understanding of the testing method [35].

The developer provides the acceptance examination, which is conducted with the end user. According to software-defining documents, the user may be internal or external. The final step is configuration management, which includes assessing the project's tools or methods. Subsequently, secure access and data transmission to consumers are completed using the security test method.

G. Security Test Method

The security testing method comprises two parts: vulnerability assessment and penetration testing [36]. Both tests are performed on each product service during the research, with the front-end system and API system acting as the back-end. Moreover, the method performs Black Box testing using specific featured scenarios on the top ten methods of Open Web Application Security Project (OWASP) and grey box testing to identify security flaws in the application or domain. These examinations are conducted using both passive and active penetration testing methods.

Then, recommendations for solutions based on existing findings are provided, followed by a process to fix the identified vulnerabilities or other security issues. After the repairs, the results are retested to ensure that the previously discovered vulnerabilities or security gaps are successfully resolved and do not reappear.

According to previous research, ten possible attacks are aimed at a web application [37]. These attacks are referenced by OWASP in its risk rating. The ten identified attacks are injection, broken authentication and session management, cross-site scripting, insecure direct object references, security misconfiguration, sensitive data exposure, missing function level access control, cross-site request, component used with known vulnerabilities, and invalidated redirects and forwards [38].

III. Results and Discussion

The findings show product quality and security assurance activities that are planned to aid in the creation and execution of DM processes. These results are divided into four primary parts: evaluation of machine learning methods, QG in terms of references, software test plan, and implementation in the quality and software test plan, which comprises functionality, integration, performance, and security tests.

A. Evaluation of Machine Learning Method

As mentioned in the previous section, all methods are trained with modeling scenarios, and 12 sets of CUSTOMERID records are ready to conduct the evaluation procedure as functional testing. In addition, six factors are analyzed, but only four sets meet the quality standards. These results are shown in Table I.

Table I shows that all methods follow four primary parameters and one supporting parameter. Among other methods, SVR achieves the five best criteria around the threshold, depicted by asterisk (*). As a result, the SVR method meets the product quality objectives reviewed in the preceding section and is used to process energy consumption and invoice prediction.

TABLE I
EVALUATION RESULT OF MACHINE LEARNING METHOD.

Method	RMSE*	MAE	MSE*	MAPE*	R-Squared*	RRMSE*
MLP	20.57	58.85	423.19	2.52	1.00	0.90
Polynomial	2.77	3,805.20	7.68	5.42	1.00	4.15
SVR	1.23	3,905.01	1.52	1.42	1.00	9.23
Criteria	Near to 0		Near to 0	< 10	Near to 1	< 10

Note: Support Vector Regression (SVR), Multi-Layer Perception (MLP), Root Mean Squared Error (RMSE), Mean Average Error (MAE), Mean Squared Error (MSE), Mean Average Probability Error (MAPE), and Relative Root Mean Squared Error (RRMSE).

TABLE II
QUALITY LISTS OF DATA MONETIZATION (DM) FEATURES.

No.	Data Monetization Qualitative Requirements	Related Quantitative Quality Goals
1	DM should be user-friendly.	The DM operator operates the feature in a day and comprehends nominal predicted energy usage on a weekly and monthly basis for a maximum of one hour.
2	DM should be reliable.	DM feature has an average of 95% availability.
3	DM should operate continuously.	DM feature recovery does not exceed 30 minutes and is accessed 24×7 days.
4	DM should be highly efficient.	DM feature is comprehended in less than 30 minutes.
5	DM should provide high-quality service to the applying customer.	DM feature achieves excellent points with $< 10\%$ of RRMSE value, good forecasting ability with $< 10\%$ of MAPE value, and a near-zero average of the square of MSE with minimal four months data.

B. Quality Goals

As previously shown, QG for DM is reviewed in the Request for Proposal (RFP) or Terms of References (TOR) feature. RFP requires the feature to show a time series of monthly energy usage over a year and a daily time series of energy consumption for the month. At the end of the month, for postpaid power customers, the system provides a prediction of energy usage for the next month and an invoice. Following this discussion, Table II shows the QG for projected consumption next month. It shows the quality plan, consisting of five needs that reflect the requirement for quality in DM feature specified in terms of references. Additionally, the software test plan is implemented after the quality objectives are acknowledged.

C. Software Test Plan

The plan consists of functional, system integration, and user acceptability tests that are performed before deployment in an operational environment. Before deployment, examinations are conducted to ensure the functionality and requirements of the user are met. After deployment, stress and security tests are performed. During this deployment, the tests focus on performance and security assurance. The results of these examinations are divided into four categories based on severity. First, critical risk is the most serious issue in the core application system and is entirely resolved before proceeding to the next phase.

High risk follows the first category, which is a bug that is relevant to the core application but does not need to be solved immediately. This risk is addressed as the first patch following the initial release deployment. The third type is considered medium risk, including a defect with medium severity. User is able to complete tasks with this severity, but the results are compromised. The final category is low risk, which includes issues with minor severity and impact on the user. Relating to this risk, the user is allowed to complete the activity with good results.

The test results are calculated based on the percentage of scenarios that pass out of the total number of examination cases. The percentage considered as passing is 100% with 0% critical and high-risk severity or 90% with only 10% unpassed due to medium or low-risk severity. Following this examination, the final criteria for passing requires a minor improvement sign.

When conducting functional tests, which comprise two test scenarios and integration tests with three tests, positive and negative test scenarios are used. All test results pass 100%. It fully satisfies the requirements. In accordance with each explanation, the specifics will be covered in next section.

D. Implementation of Quality and Software Test Plan

This section describes the implementation of standards set previously. First, in the functional test, the requirements of the customer and QG lead to the development of an Energy Management System (EMS). This model includes predicted energy usage and invoices, which are presented in a monthly and annual preview.

Cite this article as: W. H. K. Atmaja, H. L. H. S. Warnars, F. L. Gaol, and B. Soewito, "Data Monetization Service Development Using Iterative Lifecycle Framework, Quality Assurance, and Open Web Application Security Project: A Case Study of a Utility Company in Indonesia", CommIT Journal 18(2), 197–209, 2024.

TABLE III
MONTHLY BASIS ENERGY USAGE PREDICTION TEST RESULTS.

Test Code SMQA EMS – 01	Test Description Monthly energy prediction test
Test Purpose This test ensures that: 1. The feature of monthly energy prediction performs well and meets user requirements. 2. Quality Goal (QG 1) is fulfilled.	Test Prerequisite Using CUSTOMERID (Sample): 5**********8 Using EMS Credential: Username: **********; Password: ********
User	QA SQD4.3 and SQD4.4
Test Result: Graphical result of energy usage of current and next month prediction is successfully shown as expected.	Status o PASS o FAILED o PASS with NOTE

TABLE IV
DAILY BASIS ENERGY USAGE PREDICTION TEST RESULTS.

Test Code SMQA EMS – 03	Test Description Prediction of daily energy prediction test
Test Purpose This test ensures that: 1. The feature of daily energy prediction performs well and meets user requirements. 2. Quality Goal (QG 1) is fulfilled.	Test Prerequisite Using CUSTOMERID (Sample): 5**********4 Using EMS Credential: Username: **********; Password: ********
User	QA SQD4.3 and SQD4.4
Test Result : The graphical result of energy usage and the invoice of daily use prediction is successfully shown as expected.	Status o PASS o FAILED o PASS with NOTE

The main result of the test scenario executed in four activities shows the expected energy consumption in EMS using CUSTOMERID prepared, as shown in Table III. Based on the input processing of historical electrical energy usage data for the previous month, Table III presents a scenario to show the outcomes of computing estimates of electrical energy demand for the upcoming month. In addition, the scenario shows graphs on one graph representing past and projected electrical energy usage. The functional test, which employs a sample of CustomerID and password credential, yields the appropriate and satisfactory results by demonstrating the system's ability to display energy consumption graphs and calculation results.

Then, the major result of the test scenario, which is executed in four activities, is the predicted daily energy

TABLE V
APPLICATION PROGRAMMING INTERFACE (API) PREDICTION OF MONTHLY BASIS ENERGY USAGE PREDICTION TEST RESULTS.

Test Code SMQA EMS – 02	Test Description API prediction of monthly energy prediction test
Test Purpose This test will ensure that: 1. The monthly energy prediction feature performs well and meets user requirements. 2. API prediction used in point 1 performs excellently. 3. Quality Goal (QG 1) is fulfilled.	Test Prerequisite Using CUSTOMER ID (Sample): 5**********1 Using EMS Credential: Username: **********; Password: ******** Using the Postman Application to test the API
User	QA SQD4.3
Test Result : From the test result, the energy usage shown in the web graphic complies with the response of API management prepared to predict the result.	Status o PASS o FAILED o PASS with NOTE

usage in EMS based on the provided CUSTOMERID. The results are shown in Table IV. Based on the successful results of functional tests for daily and monthly checks, this service forecasts energy consumption for the current and the next month. Table IV presents graphs and functional testing for daily electrical energy usage projections based on historical daily usage. Utilizing the provided Customer ID and user password credentials, the test scenario is executed. The results suitably and correctly present the daily prediction results. The test scenario is deemed successful and fulfilled QG as a result of this test's execution.

Second, the integration test verifies the result and requirements of combining the EMS application with prediction API. The outcome is divided into two scenario tests, which estimate monthly and daily usage. In the integration test of API prediction of monthly basis energy usage, the test scenario is executed in two activities containing the main result that shows the outcome of API. Table V displays the outcomes. The interface using the API is shown to calculate energy usage and make predictions for the upcoming month. These results are utilized by other systems that have secure connections and data protection to make predictions. After determining login and password credentials using API returns that are compatible and appropriately displayed on other systems, the test scenario is executed using the customer's scenario.

Table VI shows the main result of the five-activity test scenario, which is the prediction of API of daily

TABLE VI
APPLICATION PROGRAMMING INTERFACE (API) PREDICTION OF DAILY BASIS ENERGY USAGE PREDICTION TEST RESULTS.

Test Code SMQA EMS – 04	Test Description API prediction of daily energy prediction test
Test Purpose This test ensures that: 1. The feature of daily energy prediction performs well and meets user requirements. 2. API prediction used in point 1 performs excellently. 3. Quality Goal (QG 1) is fulfilled.	Test Prerequisite Using CUSTOMER ID (Sample): 5**********4 Using EMS Credential: Username: **********; Password: ******** Using the Postman Application to test the API
User	QA SQD4.3
Test Result : The response to the prediction of daily energy usage by API is consistent with the EMS results recorded in the database.	Status o PASS o FAILED o PASS with NOTE

TABLE VII
NEGATIVE TEST OF APPLICATION PROGRAMMING INTERFACE (API) PREDICTION OF DAILY BASIS ENERGY USAGE PREDICTION RESULTS.

Test Code SMQA EMS – 05	Test Description Negative test of daily energy prediction usage with unsynchronized data
Test Purpose This test ensured that: 1. The feature of daily energy prediction performs well and meets user requirements. 2. API prediction used in point 1 performs excellently. 3. Quality Goal (QG 1) is fulfilled.	Test Prerequisite Using CUSTOMER ID (Sample): 5**********1
User	QA SQD4.3
Test Result : No graphical data or result due to failed synchronization as expected.	Status o PASS o FAILED o PASS with NOTE

energy consumption in EMS using CUSTOMERID. Based on historical daily power consumption, Table VI shows the expected daily energy use results for the following day. It is achieved by the display of API testing results on other systems. It correctly presents the same computation results as the intended test case on other systems.

Next, Table VII shows the major outcome of the test scenario executed in two activities that API prediction of daily energy usage in EMS is not shown because the data used are not synced. According to the negative test scenario, there is interference in the connection, which is why the target system in Table VII does not display the results of the API calculation. In this case, the test is passed and the findings meet standards. A negative functional examination is performed as part of a comprehensive set of test scenarios to meet QG 2, which is reliability of the system and its interface. During the examination, the system responds to certain events, such as a network outage or a paused process, demonstrating that the system can handle the obstacle that occurs and can provide results that are appropriate for the situation and convey obstacle alerts to the system.

The positive and negative functional and incorporation test results show that the DM process can be used in both front-end and back-end settings and securely transmitted via API links. Moreover, the product quality material has been confirmed and incorporated into a web application, which can be accessed through activity scenarios. In addition to this discussion, the infrastructure environment meets quality standards for running the application.

Third, the performance test completes QG 2 and 3, achieving an average of 95% availability and the ability to be assessed 24/7. All parameters are sourced from the EMS server and service to assess the feature using the available average. As a result, the parameter includes response reviews to the server, response time and throughput, and total transaction per second. The system parameters and server endurance to process the service are used to achieve 24×7 operational capability and recovery features. Therefore, active threads over time, the response time distribution of the server, and response time versus request are measured.

As previously stated, QG 2 measures service response to the server, response time and throughput, and total transactions per second. In the research, the measurements gathered by Apache JMeter are explained as follows. Out of 2,000 sample services performed in 2 minutes, 94.6% of the samples meet the criteria while 5.4% fail due to response issues. Given that the user tolerance threshold is 95% with a 1% margin, the service request meets the user quality requirement at 94.05%.

During a two-minute measurement of response time and throughput, the system has an average response time of 22.4 seconds, with a maximum of 91 seconds, and a minimum of 0.128 seconds for processing 2,000 requests. This condition is supported by a received bandwidth of 7.5 kbps and a sending bandwidth of 39.98 kbps. Moreover, the system meets the quality criteria based on reaction time and throughput. Even

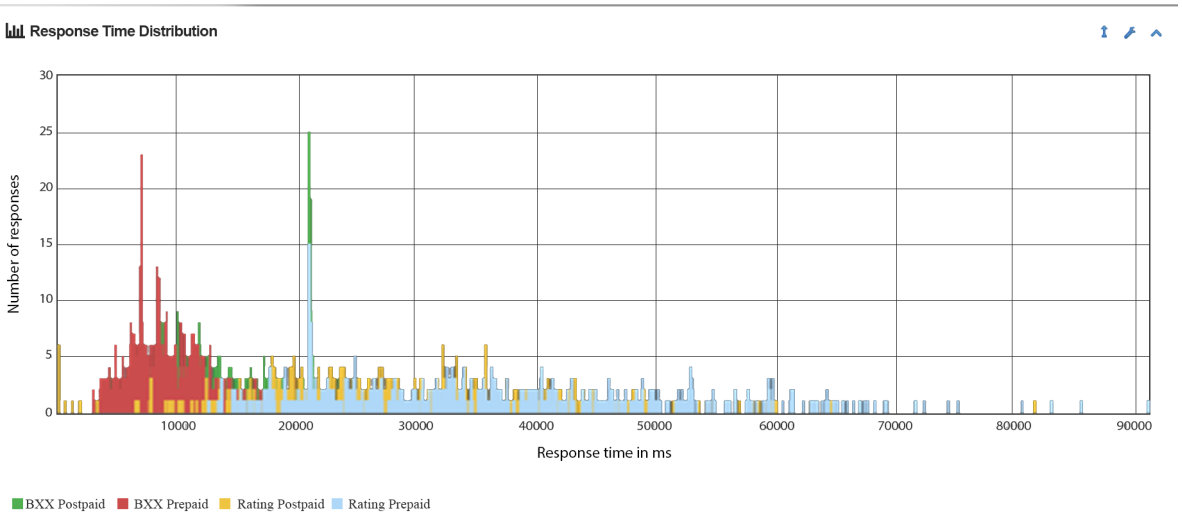

Fig. 1. Response time distribution result.

when the system is simulated with a maximum reaction time of 91 seconds while processing 2,000 samples, it still achieves an excellent user experience by providing 21 samples per second. This performance meets the user’s requirement of showing at least 20 samples per view in one second.

After measuring the response time and throughput of the system, the number of Transactions Processed per Second (TPS) is also measured. The system initially processes 5 TPS and increases to a maximum of 14.2 TPS before stabilizing at 12.13 TPS. Based on these measurements, the system engineer adjusts the service to achieve the same page view result of 21 TPS.

Next, QG 3 ensures an accessibility system of 24×7 in 365 days in the research context. Three measurements are related to the active thread, response time distribution, and mapping of response time to request. All of these measurements are collected in a specific time range. It shows that the original thread is stable at 500 and then drops to 300 before ending. From the perspective of the user, the number of active threads leads to 20–21 records examined per second. The system performs well, maintaining 60% of threads or 12-13 records per view, which are still acceptable.

After measuring active threads across time, the procedure proceeds to determine the response time distribution, as shown in Fig. 1. This response shows that there are four times as many peak processes. Three of the responses react faster than the average of 22 seconds. The biggest peak before 22 seconds is 7–7.1 seconds for 23 service processes. The transaction has the largest distribution, lasting 7–21 seconds following the process. The requirements for quality number two and performance number three are still met.

The scatter plot in Fig. 2 shows 15 classifications above average and 18 classifications below average. The response time distribution shows that 55% of responses are below average. According to the graphical results, 62 services achieve fast responses in 8.9 ms, while 15 categories have an average of 21 seconds. The criterion remains consistent with quality objectives.

The performance test findings show that the infrastructure meets most of the quality requirements. Additionally, the TPS parameter threshold of 21 is achieved by using a load balancer. It leads to the implementation of the service in infrastructure.

Fourth, a security test is performed as the final step to confirm that the security quality is met before the system is deployed to the operational environment. This security test has ten agendas based on OWASP Top 10: 2021, according to [37]. This process is conducted by an impartial security consultant and application security team. According to the test results in Table VIII, three vulnerabilities are classified as A01:2021 and A09:2021, patched by the system developer, and retested with a confirmed close status.

Based on the recommendations of the security team, the developer rectifies the vulnerabilities and develops application fixes that will be tested and distributed. The following processes are shown in Table IX–XI, consisting of the list of patched vulnerabilities. The vulnerability is severe, leading to patch-fixing requirements, as shown in Table IX. Based on the advice, the system validates user authentication and authorization for input requests. The user then conducts a 200-number brute-force penetration test on API. Following this process, Table X shows the improvement of brute

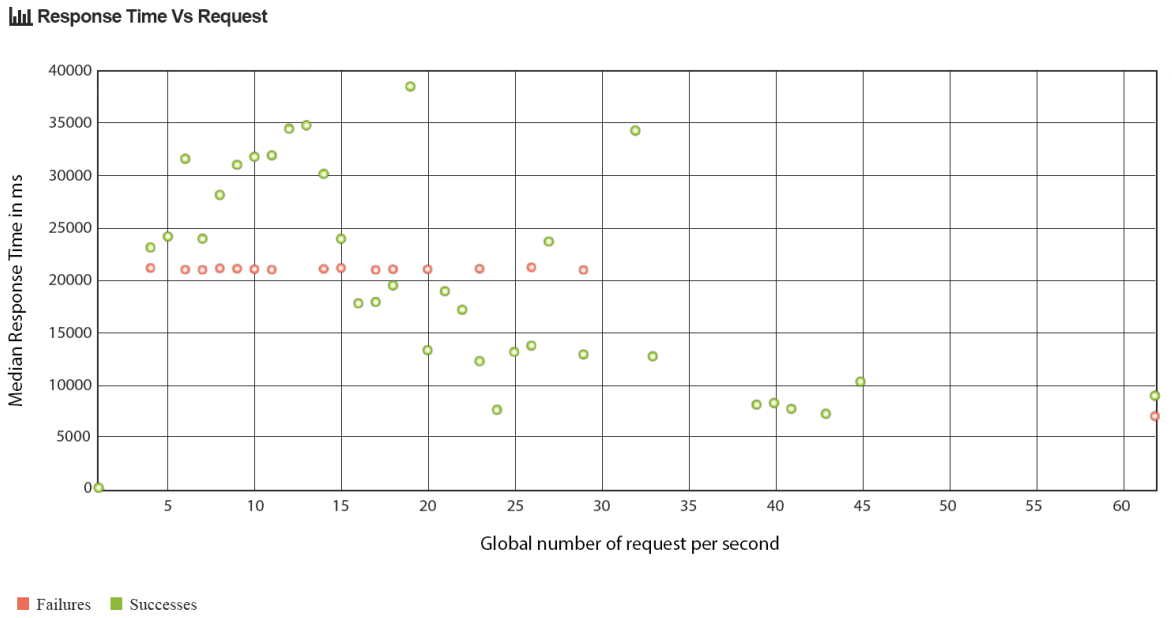

Fig. 2. Response time vs request result.

TABLE VIII
RESULTS OF VULNERABILITY ASSESSMENT.

No	OWASP Top 10:2021 Category	Category No.	Finding
1	Broken Access Control	A01:2021	Yes (CLOSED)
2	Cryptographic Failures	A02:2021	No
3	Injection	A03:2021	No
4	Insecure Design	A04:2021	No
5	Security Misconfiguration	A05:2021	No
6	Vulnerable and Outdated Components	A06:2021	No
7	Identification and Authentication Failures	A07:2021	No
8	Software and Data Integrity Failures	A08:2021	No
9	Security Logging and Monitoring Failures	A09:2021	Yes (CLOSED)
10	Server-Side Request Forgery	A10:2021	No

TABLE IX
EVIDENCE OF IMPROVEMENT FROM BROKEN ACCESS CONTROL.

OWASP Code A01:2021	Vulnerabilities Unauthorized Access - User Information	Severity HIGH
OWASP Methodology Broken Access Control	Category Web Application	Reference [39, 40]
Impact 1. Non-formatted input can disrupt system flow and lead to recurrent attacks. 2. An attacker can access and get data from the program.	Recommendation The system verifies user authentication and authorization from any input request to prevent the control flow of the application.	Final Status Completed

force in the Login Menu.

According to Table X, the medium vulnerabilities attempt to obtain access to the system by exploiting the weak password. As a result, people with weak or odd passwords are prevented from logging in. After the user generates a weak or uncommon password, the system refuses the request and proposes a strong password based on good criteria. Moreover, when the user attempts to log in using a recursive password list, the system declines the second attempt. Based on these efforts, the system implements the proposal. The vulnerability of password plaintext is shown in Table XI.

The password plaintext vulnerability is rated medium severity in the research context. The attacker uses a man-in-the-middle assault. Therefore, 3DES encryption is implanted to improve the security of the system against this type of threat. In this exploration, the user types in the username and password, which the system encrypts and provides recommendations based on the test results.

Aside from the vulnerability test, the penetration test (Pentest) is conducted in parallel. This test includes both active and passive penetration tests, with the results shown in Table XII. The test results confirm that the system is not exploitable and meets all security standards. Table XII shows the activities that are examined.

The OWASP Top 10: 2021, which consists of ten checking items, includes two recommendations for

TABLE X
EVIDENCE OF IMPROVEMENT FROM BRUTE FORCE IN LOGIN MENU.

OWASP Code A09:2021	Vulnerabilities Brute Force Menu Login	Severity Medium
OWASP Methodology Security Logging and Monitoring Failures	Category Web Application	Reference [41]
Impact 1. An attacker can assess the credentials of a user and log into the system. 2. An attacker can find a weak password by attempting every possible combination of letters, numbers, and symbols until an appropriate one is found.	Recommendation The system blocks the user account that attempts to enter the system with anomaly passwords.	Final Status Completed

TABLE XI
EVIDENCE OF IMPROVEMENT FROM PASSWORD PLAINTEXT.

OWASP Code A01:2021	Vulnerabilities Password Plaintext	Severity Medium
OWASP Methodology Broken Access Control	Category Web Application	Reference [42]
Impact Allow attacker to access user credentials in plaintext via a Main-in-the-Middle attack.	Recommendation Implement encryption at the application layer for passwords. In general, the use of 3DES or the use of double hashes (MD5 and SHA-1) simultaneously makes it more difficult for an attacker to see the original password of the customer who is logging in.	Final Status Completed

improvement for A01:2021 and A09:2021. It has input of user authentication and authorization to prevent the application's control flow, attempts to enter the system with anomaly passwords, and implementation of 3DES and double hashes (MD5 and SHA-1) at the application layer for passwords. The three recommendations have been properly completed, so they are considered complete. Meanwhile, nine pentest operations produces results that cannot be exploited. Based on the security test results, which include ten agendas linked to OWASP Top 10: 2021 and nine penetration test activities, it is concluded that the application and API service are secure. Additionally, the supporting or integration environment is also secured and not vulnerable to intruder.

IV. CONCLUSION

In conclusion, the research aims to provide DM services to an Indonesian utility company as part of a long-term strategic plan to generate additional revenue beyond its core operations. The service is

TABLE XII
PENETRATION TEST ACTIVITIES AND RESULTS.

No.	Penetration Test Activity	Status
1	DDOS Slowloris	Failed to Exploit
2	Scan Nmap Server	Failed to Exploit
3	SQL Injection	Failed to Exploit
4	Cross-Site Scripting (XSS)	Failed to Exploit
5	Upload Backdoor on Menu Upload	Failed to Exploit
6	Server Side Request Forgery (SSRF)	Failed to Exploit
7	Scan Technology Framework	Failed to Exploit
8	Account Take Over	Failed to Exploit
9	Session Fixation	Failed to Exploit

built using strong and safe methods in an iterative development lifecycle framework to achieve the plan. This framework includes processing electricity usage data, testing application quality, checking infrastructure quality, and ensuring access security for both front-end as well as back-end applications via OWASP. The data modeling method uses SVR with five evaluation criteria, including 1.42 MAPE, 1.52 MSE, 1.23 RMSE, 9.23 RRMSE, and 1.0 Coefficient of Determination. It outperforms PR and MLP Neural Network as the final option to ensure product quality and security.

The application and service features are analyzed and tested, leading to a 95% SLA per month, 24×7 access, and a processing time of less than 30 minutes per batch. The feature meets the required threshold, including 95% service response in throughput, 0.128 seconds for processing 2,000 requests, stability at 300-500 in one second per hour, and 7-21 seconds during peak hour. In addition, the feature is secured by passing the nine penn tests item and ten vulnerability assessments using OWASP top 10:2021 category. According to the method and test results, the application, as well as the service, are ready for deployment.

The research aims at exploration and application service providers who use data utilities to generate revenue from publicly available data. The research effectively solves data privacy, exacted data generation, and protection against threats to data, containers, and connectivity. In the context of this exploration, initial areas requiring an incorporated strategy include processing, data modeling, application development, service testing, infrastructure verification, and service security. Implementing strong security services will help businesses to maximize potential revenue.

The limitation of the research is its focus on processing and showing comprehensive data from all locations only every two-year interval. Based on the restrictions, a future exploration is proposed to implement DM initiatives through energy usage and customers' invoice amount prediction. It can apply an incomplete and scattered data set of electricity consumption in the residential, business, industry, government, and social

segments over a limited period.

Author Contribution

Writing—original draft, W. H. K. A.; Methodology, W. H. K. A. and F. L. G.; Formal analysis, W. H. K. A., H. L. H. S. W., and B. S.; Analysis result review, W. H. K. A., H. L. H. S. W., and F. L. G. All authors have read and agreed to the published version of the manuscript.

References

[1] A. Bleier, A. Goldfarb, and C. Tucker, "Consumer privacy and the future of data-based innovation and marketing," *International Journal of Research in Marketing*, vol. 37, no. 3, pp. 466–480, 2020.

[2] J. Baecker, M. Engert, M. Pfaff, and H. Krcmar, "Business strategies for data monetization: Deriving insights from practice," in *Wirtschaftsinformatik (Zentrale Tracks)*, 2020, pp. 972–987.

[3] A. S. George and A. S. H. George, "Data sharing made easy by technology trends: New data sharing and privacy preserving technologies that bring in a new era of data monetization," *Partners Universal International Research Journal*, vol. 1, no. 3, pp. 13–19, 2022.

[4] P. Parvinen, E. Pöyry, R. Gustafsson, M. Laitila, and M. Rossi, "Advancing data monetization and the creation of data-based business models," *Communications of the Association for Information Systems*, vol. 47, pp. 25–49, 2020.

[5] S. Mehta, M. Dawande, G. Janakiraman, and V. Mookerjee, "How to sell a data set? Pricing policies for data monetization," *Information Systems Research*, vol. 32, no. 4, pp. 1281–1297, 2021.

[6] H. Zhong, J. Wang, H. Jia, Y. Mu, and S. Lv, "Vector field-based support vector regression for building energy consumption prediction," *Applied Energy*, vol. 242, pp. 403–414, 2019.

[7] F. Kaytez, "A hybrid approach based on autoregressive integrated moving average and least-square support vector machine for long-term forecasting of net electricity consumption," *Energy*, vol. 197, 2020.

[8] I. Shah, H. Iftikhar, S. Ali, and D. Wang, "Short-term electricity demand forecasting using components estimation technique," *Energies*, vol. 12, no. 13, pp. 1–17, 2019.

[9] M. Bilal, H. Kim, M. Fayaz, and P. Pawar, "Comparative analysis of time series forecasting approaches for household electricity consumption prediction," 2022. [Online]. Available: https://arxiv.org/abs/2207.01019

[10] S. Jung, J. Moon, S. Park, S. Rho, S. W. Baik, and E. Hwang, "Bagging ensemble of multilayer perceptrons for missing electricity consumption data imputation," *Sensors*, vol. 20, no. 6, pp. 1–16, 2020.

[11] J. S. Chou and D. S. Tran, "Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders," *Energy*, vol. 165, pp. 709–726, 2018.

[12] G. Chitalia, M. Pipattanasomporn, V. Garg, and S. Rahman, "Robust short-term electrical load forecasting framework for commercial buildings using deep recurrent neural networks," *Applied Energy*, vol. 278, 2020.

[13] Y. Kim, H. G. Son, and S. Kim, "Short term electricity load forecasting for institutional buildings," *Energy Reports*, vol. 5, pp. 1270–1280, 2019.

[14] Y. Hong, Y. Zhou, Q. Li, W. Xu, and X. Zheng, "A deep learning method for short-term residential load forecasting in smart grid," *IEEE Access*, vol. 8, pp. 55 785–55 797, 2020.

[15] C. Rao, Y. Zhang, J. Wen, X. Xiao, and M. Goh, "Energy demand forecasting in China: A support vector regression-compositional data second exponential smoothing model," *Energy*, vol. 263, 2023.

[16] M. Atanasovski, M. Kostov, B. Arapinoski, and M. Spirovski, "K-nearest neighbor regression for forecasting electricity demand," in *2020 55th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST)*. Serbia: IEEE, Sep. 10–12, 2020, pp. 110–113.

[17] A. Stuke, M. Todorović, M. Rupp, C. Kunkel, K. Ghosh, L. Himanen, and P. Rinke, "Chemical diversity in molecular orbital energy predictions with kernel ridge regression," *The Journal of Chemical Physics*, vol. 150, no. 20, 2019.

[18] J. B. Fiot and F. Dinuzzo, "Electricity demand forecasting by multi-task learning," *IEEE Transactions on Smart Grid*, vol. 9, no. 2, pp. 544–551, 2016.

[19] A. Rahman, V. Srikumar, and A. D. Smith, "Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks," *Applied Energy*, vol. 212, pp. 372–385, 2018.

[20] H. Hamedmoghadam, N. Joorabloo, and M. Jalili, "Australia's long-term electricity demand forecasting using deep neural networks," 2018.

[Online]. Available: https://tinyurl.com/4xes2h99

[21] A. Zendehboudi, M. A. Baseer, and R. Saidur, "Application of support vector machine models for forecasting solar and wind energy resources: A review," *Journal of Cleaner Production*, vol. 199, pp. 272–285, 2018.

[22] M. Sharifzadeh, A. Sikinioti-Lock, and N. Shah, "Machine-learning methods for integrated renewable power generation: A comparative study of artificial neural networks, support vector regression, and Gaussian process regression," *Renewable and Sustainable Energy Reviews*, vol. 108, pp. 513–538, 2019.

[23] D. Maulud and A. M. Abdulazeez, "A review on linear regression comprehensive in machine learning," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, pp. 140–147, 2020.

[24] C. Jin, X. Bai, C. Yang, W. Mao, and X. Xu, "A review of power consumption models of servers in data centers," *Applied Energy*, vol. 265, 2020.

[25] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Computer Science*, vol. 7, pp. 1–24, 2021.

[26] A. De Myttenaere, B. Golden, B. Le Grand, and F. Rossi, "Mean absolute percentage error for regression models," *Neurocomputing*, vol. 192, pp. 38–48, 2016.

[27] A. Al Mamun, M. Sohel, N. Mohammad, M. S. H. Sunny, D. R. Dipta, and E. Hossain, "A comprehensive review of the load forecasting techniques using single and hybrid predictive models," *IEEE Access*, vol. 8, pp. 134 911–134 939, 2020.

[28] J. Qi, J. Du, S. M. Siniscalchi, X. Ma, and C. H. Lee, "On mean absolute error for deep neural network based vector-to-vector regression," *IEEE Signal Processing Letters*, vol. 27, pp. 1485–1489, 2020.

[29] A. Botchkarev, "Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology," 2018. [Online]. Available: http://arxiv.org/abs/1809.03006

[30] S. Chen and N. M. Luc, "RRMSE voting regressor: A weighting function based improvement to ensemble regression," 2022. [Online]. Available: http://arxiv.org/abs/2207.04837

[31] H. E. Niles, "Correlation, causation and Wright's theory of "path coefficients"," *Genetics*, vol. 7, no. 3, pp. 258–273, 1922.

[32] C. C. Venters, R. Capilla, S. Betz, B. Penzenstadler, T. Crick, S. Crouch, E. Y. Nakagawa, C. Becker, and C. Carrillo, "Software sustainability: Research and practice from a software architecture viewpoint," *Journal of Systems and Software*, vol. 138, pp. 174–188, 2018.

[33] A. Mansoor, D. Streitferdt, E. Rozova, and Q. Abbas, "Goal based tailoring of quality models for quality requirements," in *Future Technologies Conference (FTC)*, Vancouver, Canada, Nov. 2017, pp. 681–686.

[34] D. Galin, *Software quality assurance: From theory to implementation*. Pearson Education, 2004.

[35] W. E. Perry, *Effective methods for software testing*. John Wiley & Sons, 2006.

[36] S. Elder, N. Zahan, V. Kozarev, R. Shu, T. Menzies, and L. Williams, "Structuring a comprehensive software security course around the OWASP application security verification standard," in *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. Madrid, ES: IEEE, May 25–28, 2021, pp. 95–104.

[37] O. B. Fredj, O. Cheikhrouhou, M. Krichen, H. Hamam, and A. Derhab, "An OWASP top ten driven survey on web application protection methods," in *Risks and Security of Internet and Systems: 15th International Conference, CRiSIS*. Paris, France: Springer, 2021, pp. 235–252.

[38] S. K. Lala, A. Kumar, and T. Subbulakshmi, "Secure web development using owasp guidelines," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*. Madurai, India: IEEE, May 6–8, 2021, pp. 323–332.

[39] PortSwigger, "Insecure Direct Object References (IDOR)." [Online]. Available: https://portswigger.net/web-security/access-control/idor

[40] D. Johansson, "OWASP top 10 2017." [Online]. Available: https://owasp.org/www-chapter-cambridge/presentations/prev/Cambridge_13-Mar-2018_OWASP_Top_10_2017.pdf

[41] OWASP, "A2:2017-Broken Authentication." [Online]. Available: https://owasp.org/www-project-top-ten/2017/A2_2017-Broken_Authentication

[42] ——, "A3:2017-Sensitive Data Exposure." [Online]. Available: https://owasp.org/www-project-top-ten/2017/A3_2017-Sensitive_Data_Exposure