Using Software Quality Evaluation Standard Model for Managing Software Development Projects in Solar Sector

Farhad Khosrojerdi*, Stéphane Gagnon*, Raul Valverde**

* Université du Québec en Outaouais (UQO), Gatineau (Québec), Canada,

** Concordia University, Montreal (Québec), Canada,

Abstract **- This paper proposes a framework for managing Project Quality Management (PQM) processes of software development projects related to photovoltaic (PV) system design. The International Organization for Standardization (ISO) quality evaluation model, Software Product Quality Requirements and Evaluation (SQuaRE) standard, is used to determine quality characteristics and quality metrics of the software. This work presents the following contributions: I) defining quality characteristics associated with a PV design software using the SQuaRE standard model, II) adding the proposed framework as a tool and technique which is used by practitioners following the global standard book for project managers, A Guide to the Project Management Body of Knowledge (PMBOK), and III) Identifying quality measures and sub-characteristics of a PV design software. The presented model can be employed for simulation-based and/or model-based software products in various technical fields and engineering.**

Keywords— software quality evaluation; quality model; software development projects; renewable energy projects; PMBOK; SQuaRE standard;

I. Introduction

Globally, the contribution of PV systems to energy generation was approximately 14,000 MW in 2010 and is expected to be 70,000 MW in 2020 [1]. In Canada, the domestic market has been growing on average at about 26% per year since 1993 and about 48% since 2000 [2]. Due to growing usage of solar energy and PV system installations, there are several online software products that provide designing, simulation, analysis, and calculation of the power produced. Residential and commercial applications of PV systems have motivated information technology (IT) and utility companies to create PV design software packages. These customized software products or applications estimate the energy production and the cost for the PV system installation. The goal for software engineers and project managers in this field is to develop high quality PV design software products.

PV design software users can rely on output reports of a PV design program that is produced based on state-of-the-art technical approaches related to PV model and simulation. Some of these applications, such as PV-LIB, are open source and allow for an open assessment of software quality [3-6]. This helps prevent excessive cost to overcome poor software quality during development [7].

Usually, software development projects are initiated to modify one or different characteristic(s) of a software and improve its quality. For a project management team, it is essential to be able to describe various characteristics of a project deliverable(s). Quality characteristics and quality metrics of a software help the project team to plan, execute, and monitor quality of performing a software development project. Only in this matter, a product that is delivered at the end of a project meets quality requirements defined by stakeholders. The project management global standard's guidebook (PMBOK) [8] presents guidelines for managing projects regardless of their context. PMBOK defines processes in each of eight knowledge areas in project management. These processes include inputs, outputs, and tools and techniques that enable a project team to manage activities correlated to each phase of the project. In this study, we focus on the activities that identify and guarantee the quality of a software development project and the project deliverable, a PV design and simulation software product.

A quality tool or model can assure best quality evaluation outcomes [9] and provides a framework to validate quality of the software's outputs. Use of quality models enables project managers to outline quality characteristics of a product and manage activities required for the software quality assurance. Quality model is defined as "a set of characteristics, and of relationships between them, which provides a framework for specifying quality requirements and evaluating quality" [10]. Software quality models have been developed to be used for evaluating a software. Numerous quality models have been developed over the last 40 years. Ambiguities in the evaluation methods [11], uncertainty and vagueness about selected criteria [12], and difficulties in defining inputs, outputs, and metrics [13] cause complications applying these models. Therefore, managing software development projects, herein PV designing, would be problematic even more.

To overcome these problems, we present a quality model defining quality characteristics of a PV design software. Two important technical characteristics of a PV system model are identified as quality characteristics of a PV design related software. In addition, the quality sub-characteristics and the quality metrics based on SQuaRE standard are defined. Obtaining quality measures assist a project management team to utilize them for: I) the process of identifying quality requirements in planning phase, II) the process of implementing quality management plan in the execution phase, and III) the process of controlling and monitoring the quality in the executing phase of the project life cycle. In addition to specifying quality measures, we propose the use of an Adaptive Neuro-Fuzzy Inference System (ANFIS) methodology to analyze linguistic reviews and expert's judgement of the quality evaluations. The ANFIS technique constitutes a fast and appropriate systematic solution to the PV system even with inaccurate quality measurement data.

The rest of this paper is structured as follows: section II provides a literature review describing quality models. PQM processes and PV technical factors used in this study are described in section III. In section IV, quality measures for a PV design and simulation software is defined. In section V, the ANFIS method is applied on random quantified data to demonstrate how real-world data can be used for quality measurement of a specific characteristic of the software. Finally, a conclusion is made in section VI.

II. Quality Models

A. McCall model

This model defines software product qualities as a hierarchy of factors, criteria, and metrics [14]. Eleven product quality features into three perspectives are presented as: I) product review (maintenance, flexibility, and testing), II) product operation (correct, reliable, efficient, integrity, and usability), and III) product transition (portability, reusability, and interoperability). The problems in the model involve values associate with combining distinctive metrics, as they are based on "Yes" or "No" responses [15, 16].

B. Boehm model

Boehm's model proposes different identification and classification of quality characteristics and presents metric automated evaluation [7]. Although it introduces a metric providing quantitative measurement of the degree to which a program has the associated characteristics, it is performed for specific program. Besides, the process of definitions of quality attributes are arbitrary [15].

C. Dromey model

This model establishes a measurement for including a software property in a model [15]. It argues that hierarchical models are vague in their definitions of lower levels. The quality evaluation is performed unlikely for each product in a dynamic fashion. However, there is no discussion of how this can be implemented [16].

D. FURPS model

In this model the quality characteristics are divided into functional requirements (RF), defined by the inputs/outputs, and non-functional requirements (NF) including usability (U), reliability (R), performance (P), and product support (S) [17]. The drawback of the model is that major quality characteristics are not identified herein [16].

E. ISO/IEC Standard series

In 1991, the ISO published its first international consensus on the terminology for the quality characteristics for software product evaluation (ISO 9126:1991). From 2001 to 2004, the ISO published an expanded version, containing both the ISO quality models and inventories of proposed measures for these models.

The current version of the ISO 9126 series of standards now consists of four documents: quality models - ISO 9126-1, external metrics 1 - ISO TR 9126-2, internal metrics - ISO TR 9126-3, and quality in use metrics - ISO TR 9126-4. SQuaRE - ISO 25000. This series of standards will replace the current ISO 9126 and ISO 14598 series of standards. The SQuaRE series will consist of five divisions: quality management division (ISO 2500n), quality model division (ISO 2501n), quality measurement division (ISO 2502n), quality requirements division (ISO 2503n), and quality evaluation division (ISO 2504n). It defines evaluation methods for the quality of software products and provide common standards, called the SQuaRE series [18] including ISO/IEC 25022:2016 [19] and ISO/IEC 25023:2016 [20].

One of the main objectives of (and differences between) the SQuaRE series of standards and the current ISO 9126 series of standards is the coordination and harmonization of its contents with ISO 15939 [21]. The set of goals or quality characteristics can be the same or similar to the one defined in ISO/IEC 9126. Such a definition helps in evaluating the quality of software, but gives no guidance on how to construct a high quality software product.

Table 1 depicts quality measures for a PV design and simulation software. The presented quality metrics may be used for software quality evaluation of a PV software from an end-user's point of view. The quality characteristics represent various quality criteria that a user expects from a PV design application. From technical perspectives, only Functional Suitability demonstrates technical quality dimensions associated with a PV performance model.

Table 1 Quality measures for a PV design and simulation software

Quality characteristic	**Quality sub-characteristic**	**Quality measure name**
Functional Suitability	**Functional completeness**	Functional coverage
	Functional Correctness	Functional correctness
	Functional Appropriateness	Functional appropriateness of usage objective
		Functional appropriateness of system
Compatibility	**Co-existence**	Co-existence with other products
	Interoperability	Data format exchangeability
		External interface adequacy

Security	**Confidentiality**	Access controllability
	Integrity	Data integrity
	Accountability	User audit trail completeness
	Authenticity	Authentication mechanisms sufficiency
Usability	**Appropriateness recognisability**	Description completeness
	Learnability	User guidance completeness
	Operability	Operational consistency
		Functional customizability
		User interface customizability
		Undo capability
		Understandable categorization of information
	User error protection	Avoidance of user operation error
		User entry error correction
	User interface aesthetics	Appearance aesthetics of user interface
Reliability	**Maturity**	Fault correction
		Test coverage
	Availability	System availability
	Fault tolerance	Failure avoidance
Maintainability	**Modularity**	Coupling of components
	Reusability	Reusability of assets
		Coding rules conformity
	Analysability	System log completeness
	Modifiability	Modification efficiency
		Modification correctness
		Modification capability
	Testability	Test function completeness
Performance efficiency	**Time Behaviour**	Mean response time
	Resource utilization	Mean processor utilization

III. PQM Processes for a Software Development Project

In order to meet stakeholders' objectives, the PQM includes the processes for incorporating the required quality regarding planning, managing, and controlling [8]. As defined in PMBOK: "Plan Quality Management is the process of identifying requirements and/or standards for the project and its deliverables and documenting how the project will demonstrate compliance with quality requirements and/or standards. Manage Quality is the process of translating the quality management plan into executable quality activities that incorporate the organization's quality policies into the project. Control Quality is the process of monitoring and recording the results of executing the quality management activity to assess performance and ensure the project outputs are complete, correct, and meet customer expectations."

Inputs, outputs, tools and techniques that can be used to manage the processes of quality in a project are shown in PMBOK [8].

Considering the technical characteristics of a PV design software and the PQM processes provided in PMBOK, the inputs, the outputs, the tools and the techniques are defined for the PV software development project (Table 2). As indicated, the SQuaRE-based quality model is added to the techniques that can be applied in plan quality management process and control quality process.

Table 2 Project Quality Management processes overview

PQM processes	Inputs	Tools and Techniques	Outputs
Plan Quality Management	1.Project charter 2.Project management plan 4.Organizational process assets	1.Expert judgement 2.Data gathering 3.Data Analysis 4.Test and inspection planning **5.SQuaRE-based Quality model**	1.Quality management plan 2.Quality metrics 3.Project management plan updates 4.Project documents updates
Manage Quality	1.Project management plan 2.Project documents 3.Organizational process assets	1.Data gathering 2.Data analysis 3.Decision making 4.Audits 5.Problem solving	1.Quality reports 2.Test and evaluation documents 3.Change request 4.Project management plan updates 5.Project documents updates
Control Quality	1.Project management plan 2.Project documents 3.Approved change requests 4.Deliverables 5.Work performance data 6.Organizational process assets	1.Data gathering 2.Data analysis 3.Inspections 4.Testing/product evaluations **5.SQuaRE-based Quality model**	1.Quality control measurements 2.Verified deliverables 3.Work performance information 4.Change requests 5.Project management plan updates 6.Project documents updates

This quality measurement framework may be used by software and systems engineers to ensure that PV design applications fulfill all requirements. Whether functional or non-functional, PV design requirements imply close attention to PV mathematical modeling [22, 23], along with various designs of PV integration within smart renewable energy projects [24-26]. These design parameters must also be submitted to optimization techniques, to ensure accurate information output from PV design software, especially for power forecasting tasks and load balancing within grid-connected systems [27-31].

IV. **Quality measures for a PV design and simulation software**

A. Technical Characteristics of a PV System

A PV cell model, representing its physical behaviors under all operating conditions, is needed to be able to simulate the PV cell [32]. An electrical circuit model could predict variations of *I-V* and *P-V* curves to the ambient conditions and environmental factors. The calculation of the model parameters including the values of components and variables of the model describe the physical reactions of the PV cell to the external factors. Figure 1 depicts the overall topology of a PV system and its power conversion concept.

In power conversion in a PV system, the most important part is operating at maximum power point in different environmental and ambient conditions. Partial shading conditions and performing under various temperature require a control system for the PV system, known as maximum power point tracking (MPPT) controller [33, 34]. There are various MPPT approaches and algorithms used for a MPPT controller to generate maximum output power [35].

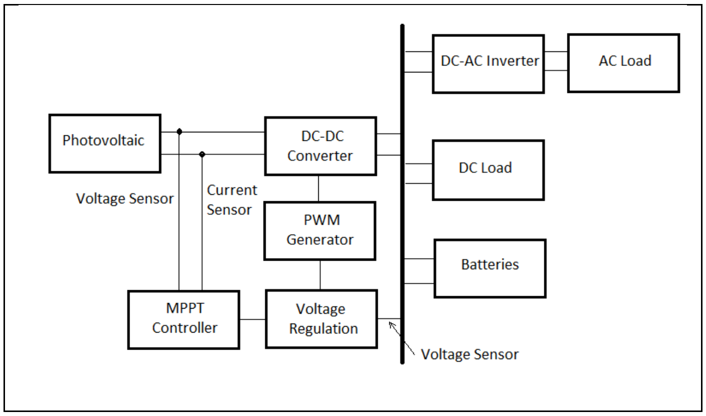

Fig. 1 The overall topology of a PV system

Hence, technical characteristics of a PV simulation rely on considering the two important technical factors, I) PV model used and II) MPPT technique.

A. Quality metrics for PV design software evaluation

The quality of a PV design software is the degree to which it satisfies the implied needs of users. These stated needs are presented in the SQuaRE standard by quality models [10]. Based on the standard, the measureable quality-related technical features of a PV design software are called properties to quantify and can be associated with quality measures. According to ISO/IEC 25023, "A measurement method is a logical sequence of operations used to quantify properties with respect to a special scale." Applying a measurement method results in obtaining quality measure elements. A measurement function, which is capable of combining quality measure elements, can quantify quality characteristics and sub-characteristics. The outcome of applying a measurement function is called quality measure [10].

Table 3 demonstrates quality measures for a PV design and simulation software. The quality characteristics represent various quality criteria that a user expects from a PV design application. From technical perspective, the quality characteristic of Functional Suitability demonstrates accurate functionality of the power calculated by the PV design software. It means that the estimated output power for a system in certain time and location can be reliable.

These quality measures are identified through the following steps:

1) The information and data are gathered from the manual, the technical reference, and the website.
2) The PV system performance models are validated based on expert's judgement.

3) The thresholds for the quality metrics are defined.

4) The quality of the metrics is measured based on the thresholds.

Based on ISO/IEC 25023, quality measures defined as: functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, and portability. For our purposes concerning quality of the software in the term of its accuracy in designing PV systems and calculating the power produced, three quality characteristics are chosen: I) functional suitability, II) compatibility, and III) usability.

Table 3 Quality measures used for a PV design and simulation software/application

Quality characteristic	Quality sub-characteristic	Quality measure name	Description
Functional Suitability	Functional completeness	Functional coverage	What proportion of the specified functions has been implemented?
	Functional Correctness	Functional correctness	What proportion of functions provides the correct results?
	Functional Appropriateness	Functional appropriateness of usage objective	What proportion of the functions required by the user provides appropriate outcome to achieve a specific usage objective?
		Functional appropriateness of system	What proportion of the functions required by the users to achieve their objectives provides appropriate outcome?

Applying quantitative analysis for the three quality sub-characteristics ultimately will evaluate quality of the functional suitability factor of the PV design application. The four quality metrics: I) functional coverage, II) functional correctness, III) functional appropriateness, and VI) functional appropriateness of system can be used as measures to quantify the functional suitability characteristics of the software. For simplicity, the functional appropriateness sub-characteristic is not measured in the proposed methodology.

Thus, the following questions describing the two quality measures, functional coverage and functional correctness, should be clarified and then be answered:

1. *What proportion of the specified functions has been implemented?*
2. *What proportion of functions provides the correct results?*

The type of PV performance model used and application of MPPT approaches validate the quality of the simulation used for the PV design software. In fact, these properties indicate the level of accuracy and functionality expected from a PV design software. The proportion and the degree of which the software deliver its functionality can be measured by these two metrics. Table 4 presents the final quality measures used for the ANFIS methodology. Ultimately, the following questions describing functional suitability of the software can be answered by the software designers and used by the project team:

1. *What is the type of PV model used in the simulation?*
2. *Is a MPPT method applied?*

Table 4 Quality measures used for the methodology

Quality characteristic	Quality sub-characteristic	Quality measure name	Description
Functional Suitability	Functional completeness	Functional coverage	What is the type of PV model used in the simulation?
	Functional Correctness	Functional correctness	Is a MPPT method applied?

V. Application of the ANFIS methodology

An ANN-based controller relies on the trained data instead of requiring prior knowledge about the PV parameters [36, 37]. The roles of FL and ANN methods in modeling, simulation, forecasting and control of PV systems are significant [38-40]. The ANFIS is a hybrid method combining neural networks and fuzzy logic. Employing the ANFIS method requires to define three important terms known as fuzzification, fuzzy rule base and defuzzification. Unlike the FL, the ANFIS can benefit from being trained from previous operational data. The Sugeno method has been employed in the proposed model in MATLAB simulation to ensure a better compatibility with the conventional ANFIS approach [38, 39, 41].

A. *The fuzzification operation*

The process of fuzzification and training the ANFIS model could be performed either by using experimental data or by producing an artificial set of data in MATLAB [41-45]. The process includes converting crispy PV performance and simulation values into fuzzy variables reflecting quality sub-characteristics of the PV software in several quality measurements. These operational conditions identify the quality characteristic and sub-characteristics of the PV system and the optimal quality evaluation points. Using experimental data [41, 45] validates the effectiveness of the ANFIS-based quality system and offers a reliable and effective training process for the evaluator.

As indicated in Table 4, the two main quality sub-characteristics, namely the Functional Completeness and the Functional Correctness, are determined as input variables of the control system reflecting Functional Suitability that affect values in the related quality measures. The output of the ANFIS are determined as the Quality Evaluation indicating the functionality of the PV system Fig. 2.

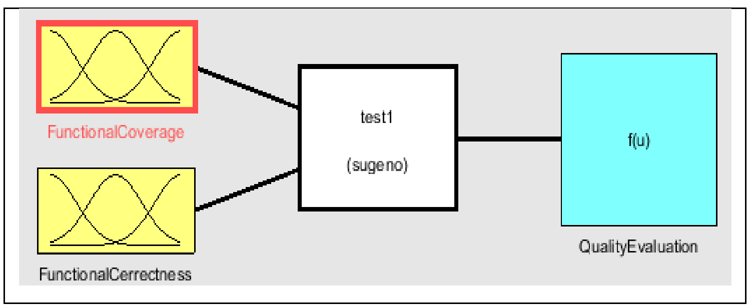

Fig. 2 Input membership functions and the ANFIS designer

B. *The fuzzy rule base and the defuzzification*

The fuzzy rule base describes the relationships and mapping between the output and input membership functions that are generated by the ANFIS. It reflects the previous knowledge obtained from the training data set or from the operational experiences and experts' judgements to define the relationships between the membership functions. Fig 3 shows the rules identified for the ANFIS, based on the considerations stated based on PV performance systems controlled by appropriate MPPT algorithms. In the last step, the defuzzification, converts the fuzzy sets into the crispy sets.

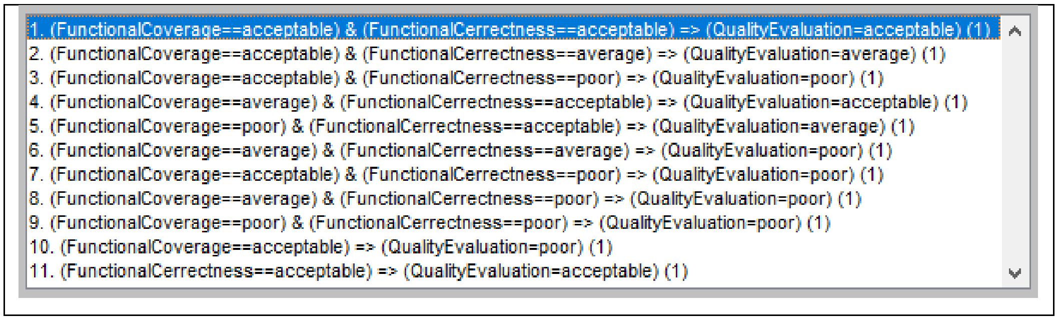

Fig. 3 The rules based on expert's judgement and the PV performance model

As it is shown (Fig 4), quality of the software can be evaluated according to the two-quality metrics, Functional Coverage and Functional Correctness. The linguistic rules defining the relationships of the input quality measures to the output heavily depend on the expert's judgement. By moving the red lines (Fig 4) which are indicating the quality values associated with the quality characteristics, the correlated quality evaluation number is shown in the rule viewer.

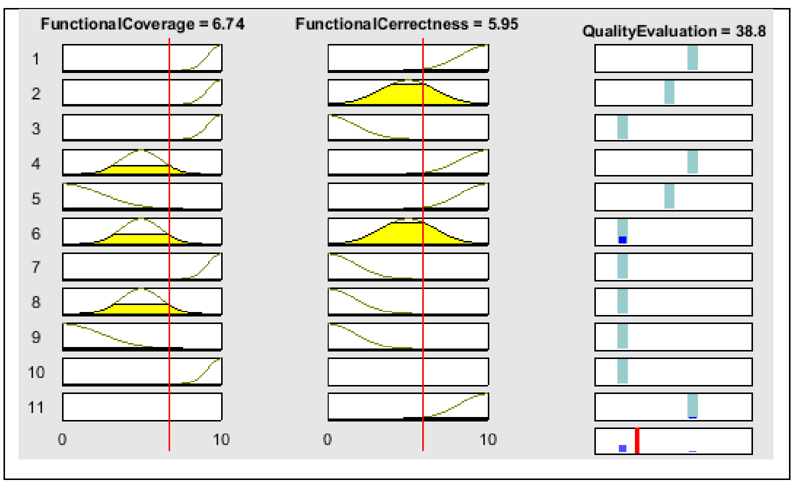

Fig. 4 The rule viewer and relationship of the inputs and the output

MATLAB Toolbox additionally provides a surface viewer illustrating the relationship between the two quality sub-characteristics and the result of quality evaluation (Fig 5); in fact, presents uncertainty around high quality metrics of Functional Correctness. It means that it is more difficult to evaluate a PV design software when the actual system and its related devices are not selected.

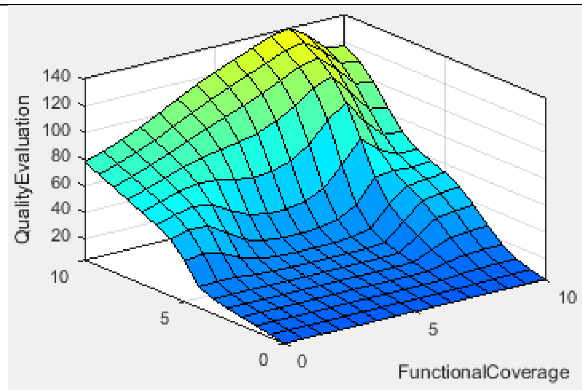

Fig. 5 Surface viewer illustrating the relationship between the two quality measures and the final result of evaluation

VI. Conclusion

To the best of our knowledge, the quality characteristics and sub-characteristics of a PV design software has not been investigated using the ISO/IEC SQuaRE standard. In addition, PQM terminology from PMBOK, can be utilized to help better manage the quality evaluation of PV design software. The ontology-based and fuzzy-reasoning approaches have also been suggested for dealing with a variety of technical software products, which shall be the focus of our future work. Finally, a common vocabulary and definitions introducing different PV design project inputs, tools and techniques, and outputs enable practitioners to employ appropriate applications. To summarize, this paper presents the followings:

I) creating a tool and technique for PQM processes which is used by practitioners following the PMBOK, the global standard book for project managers.

II) The SQuaRE is used to evaluate quality of an industry-specific software.

III) The framework can be employed for simulation-based and/or model-based software products in different fields of engineering.

The proposed research focus can provide significant benefit to overall PV design software quality. It is argued that uncertainty and vagueness about selected criteria, and difficulties in defining inputs, outputs, and quality metrics as well as ambiguities in various evaluation methods, all make it difficult to quantify simulation and model-based software products in various industries such as PV.

References

[1] M. Seyedmahmoudian *et al.*, "State of the art artificial intelligence-based MPPT techniques for mitigating partial shading effects on PV systems – A review," *Renewable and Sustainable Energy Reviews,* vol. 64, pp. 435-455, 2016.

[2] Y. Poissant and A. C. Vikis, "OVERVIEW OF PHOTOVOLTAIC SOLAR CELL R&D CAPABILITY IN CANADA ED. 4," in "CanmetENERGY, Varennes Research Centre," Natural Resources Canada2013.

[3] W. F. Holmgren, R. W. Andrews, A. T. Lorenzo, and J. S. Stein, "Pvlib python 2015," in *Photovoltaic Specialist Conference (PVSC), 2015 IEEE 42nd*, 2015, pp. 1-5: IEEE.

[4] K. A. Klise and J. S. Stein, "Automated performance monitoring for PV systems using pecos," in *Photovoltaic Specialists Conference (PVSC), 2016 IEEE 43rd*, 2016, pp. 3431-3435: IEEE.

[5] W. F. Holmgren and D. G. Groenendyk, "An open source solar power forecasting tool using PVLIB-Python," in *Photovoltaic Specialists Conference (PVSC), 2016 IEEE 43rd*, 2016, pp. 0972-0975: IEEE.

[6] J. S. Stein, W. F. Holmgren, J. Forbess, and C. W. Hansen, "PVLIB: Open source photovoltaic performance modeling functions for Matlab and Python," in *Photovoltaic Specialists Conference (PVSC), 2016 IEEE 43rd*, 2016, pp. 3425-3430: IEEE.

[7] B. W. Boehm, J. R. Brown, and M. Lipow, "QUANTITATIVE EVALUATION OF SOFTWARE QUALITY," in *ICSE '76 Proceedings of the 2nd international conference on Software engineering*, San Francisco, California, USA, 1976, pp. 592-605 IEEE Computer Society Press Los Alamitos, CA, USA.

[8] PMI, *A Guide to the Project Management Body of Knowledge, PMBOK GUIDE*. Project Management Institute (PMI), 2017, p. 977.

[9] A. V. Alberto VILLAMOR ORDOZGOITI, P. D. HITO, E. M. G. COMELLAS, C. M. F. SANCHEZ, M. Milagros GARCIA, and H. T. L. CANUT, "Software Quality Evaluation Models Applicable in Health Information and Communications Technologies. A Review of the Literature," *IOS Press,* p. 5, 2016.

[10] *Systems and software engineering-Systems and software Quality Requirements and Evaluation (SQuaRE)-Measurement of system and software product quality*, 2016.

[11] J. Heidrich and J. Münch, "Goal-oriented Data Visualization with Software Project Control Centers," in *DASMA Software Metric Congress (MetriKon 2005): Magdeburger Schriften zum Empirischen Software Engineering*, Kaiserslautern, Germany, 2005, pp. 65-75.

[12] A. Mansoor, D. Detlef Streitferdt, and F. F¨ußl, "Fuzzy Based Evaluation of Software Quality Using Quality Models and Goal Models," *(IJACSA) International Journal of Advanced Computer Science and Applications,,* vol. 6, no. 9, p. 9, 2015.

[13] H. Nakai, N. Tsuda, K. Honda, H. Washizaki, and Y. Fukazawa, "A SQuaRE-based Software Quality Evaluation Framework and its Case Study," *IEEE,* p. 4, 2016.

[14] J.A. McCall, P.K. Richards, and G.F. Walters, "Factors in Software Quality," presented at the Nat'l Tech. Information Service, Springfield, Va., 1977.

[15] B. Kitchenham and S. L. Pfleeger, "Software Quality: The Elusive Target," *IEEE SOFTWARE,* pp. 12-21, 1996.

[16] J. P. Miguel, D. Mauricio, and G. Rodríguez, "A Review of Software Quality Models for the Evaluation of Software Products," *International Journal of Software Engineering & Applications (IJSEA),* vol. 5, p. 23, 2014.

[17] R. B. Grady, *Practical Software Metrics for Project Management and Process Improvement*. Prentice Hall, Englewood Cliffs, NJ, USA, 1992.

[18] *Systems and software Quality Requirements and Evaluation (SQuaRE)- Guide to SQuaRE.* , 2014.

[19] *Measurement of quality in use-Systems and software Quality Requirements and Evaluation (SQuaRE)*, 2016.

[20] *ISO/IEC 25023:2016-Measurement of system and software product quality*, 2015.

[21] *Software Engineering – Software Product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE*, 2004.

[22] B.-K. Kang, S.-T. Kim, S.-H. Bae, and J.-W. Park, "Diagnosis of Output Power Lowering in a PV Array by Using the Kalman-Filter Algorithm," *Energy Conversion, IEEE Transactions on,* vol. 27, no. 4, pp. 885-894, 2012.

[23] J. Bai, S. Liu, Y. Hao, Z. Zhang, M. Jiang, and Y. Zhang, "Development of a new compound method to extract the five parameters of PV modules," *Energy Conversion and Management,* vol. 79, pp. 294-303, 3// 2014.

[24] N. C. Batista, R. Melício, J. C. O. Matias, and J. P. S. Catalão, "Photovoltaic and wind energy systems monitoring and building/home energy management using ZigBee devices within a smart grid," *Energy,* vol. 49, no. 0, pp. 306-315, 1/1/ 2013.

[25] T. Broeer, J. Fuller, F. Tuffner, D. Chassin, and N. Djilali, "Modeling framework and validation of a smart grid and demand response system for wind power integration," *Applied Energy,* vol. 113, no. 0, pp. 199-207, 1// 2014.

[26] N. Phuangpornpitak and S. Tia, "Opportunities and Challenges of Integrating Renewable Energy in Smart Grid System," *Energy Procedia,* vol. 34, no. 0, pp. 282-290, // 2013.

[27] G. Wang, M. Ciobotaru, and V. G. Agelidis, "Optimal capacity design for hybrid energy storage system supporting dispatch of large-scale photovoltaic power plant," (in English), *Journal of Energy Storage,* vol. 3, pp. 25-35, 2015, Art. no. 24.

[28] Y. M. Saint-Drenan, S. Bofinger, R. Fritz, S. Vogt, G. H. Good, and J. Dobschinski, "An empirical approach to parameterizing photovoltaic plants for power forecasting and simulation," (in English), *Solar Energy,* vol. 120, pp. 479-493, 2015.

[29] H. M. Hasanien, "An Adaptive Control Strategy for Low Voltage Ride Through Capability Enhancement of Grid-Connected Photovoltaic Power Plants," (in English), *IEEE Transactions on Power Systems,* 2015.

[30] A. Bonfiglio, F. Delfino, M. Invernizzi, R. Procopio, and P. Serra, "An approximate methodology to verify the compliance of large photovoltaic power plants to system operator steady-state requirements," (in English), *Electric Power Systems Research,* vol. 127, pp. 80-92, 2015, Art. no. 4337.

[31] D. O. Akinyele, R. K. Rayudu, and N. K. C. Nair, "Global progress in photovoltaic technologies and the scenario of development of solar panel plant and module performance estimation - Application in Nigeria," (in English), *Renewable and Sustainable Energy Reviews,* vol. 48, pp. 112-139, 2015.

[32] V. J. Chin, Z. Salam, and K. Ishaque, "Cell modelling and model parameters estimation techniques for photovoltaic simulator application: A review," *Applied Energy,* vol. 154, pp. 500-519, 2015.
[33] P. Joshi and S. Arora, "Maximum power point tracking methodologies for solar PV systems – A review," *Renewable and Sustainable Energy Reviews,* vol. 70, pp. 1154-1177, 2017.
[34] S. Saravanan and N. Ramesh Babu, "Maximum power point tracking algorithms for photovoltaic system – A review," *Renewable and Sustainable Energy Reviews,* vol. 57, pp. 192-204, 2016.
[35] K. Ishaque and Z. Salam, "A review of maximum power point tracking techniques of PV system for uniform insolation and partial shading condition," *Renewable and Sustainable Energy Reviews,* vol. 19, pp. 475-488, 2013.
[36] A. R. Reisi, M. H Moradi, and S. Jamasb, "Classification and comparison of maximum power point tracking techniques for photovoltaic system: A review," *Renewable and Sustainable Energy Reviews,* vol. 19, pp. 433-443, 2013.
[37] P. Bhatnagar and R. K. Nema, "Maximum power point tracking control techniques: state-of-the-art in photovoltaic applications," *Renewable and Sustainable Energy Reviews,* vol. 23, pp. 224-241, 2013.
[38] N. Khaehintung, P. Sirisuk, and W. Kurutach, "A novel ANFIS controller for maximum power point tracking in photovoltaic systems," p. 4, 2001.
[39] H. Afghou, F. Krim, and D. Chikouche, "Increase the photovoltaic conversion efficiency using neuro-fuzzy control applied to MPPT," *IEEE,* p. 6, 2013.
[40] A. Mellit and S. A. Kalogirou, "Artificial intelligence techniques for photovoltaic applications: A review," *Progress in Energy and Combustion Science,* vol. 34, no. 5, pp. 574-632, 2008.
[41] A. M. S. Aldobhani and R. John, "Maximum power point tracking of PV System using ANFIS prediction and fuzzy logic tracking," presented at the The International Multiconference of Engineers and Computer Scientists Hong Kong, 19-21 March, 2008, 2008.
[42] T.Boutabba, S. Drid, and M. Benbouzid, "Maximum power point tracking control for photovoltaic system using adaptive neuro- fuzzy "ANFIS"," presented at the Eighth International Conference and Exhibition on Ecological Vehicles and Renewable Energies (EVER), Monaco, Mar 2013, 2013. Available: https://hal.archives-ouvertes.fr/hal-00833734
[43] T. Shanthi and A. S. Vanmukhil, "Photovoltaic generation system with MPPT control using ANFIS," *International Electrical Engineering Journal (IEEJ),* vol. 4, no. 3, p. 11, 2013.
[44] R. K. Kharb, M. F. Ansari, and S. L. Shimi, "Design and implementation of ANFIS based MPPT scheme with open loop boost converter for solar PV module," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering,* vol. 3, no. 1, p. 8, 2014.
[45] H. Abu-Rub, A. Iqbal, S. M. Ahmed, F. Z. Peng, Y. Li, and G. Baoming, "Quasi-z-source inverter-based photovoltaic generation system with maximum power tracking control using ANFIS," *IEEE Transactions on Sustainable Energy* vol. 4, no. 1, p. 10, 2013.