Bulletin of Business and Economics, 13(1), 885-891
https://bbejournal.com
https://doi.org/10.61506/01.00286

Identification of the Requirement Change Management and Maturity Model in the Internet of Things (IoT)

Merry Sardar[1], Munawar Hussain[2], Mubbashar Hussain[3], Shahbaz Hussain[4], Muhammad Waseem Iqbal[5], Khalid Hamid[6], Saleem Zubair Ahmad[7]

Abstract
In planning a new-fangled system of the Internet of Things (IoT) an organization needs to capture the requirements especially once it comes near collecting confidentiality and security requirements just before winning over the community. The study is focused on creating an identification of the need maturity model (IRMM). The suggested methodology would help software development companies enhance and change their requirement engineering procedures in relationships with IoT safekeeping and confidentiality. The IRMM model determination remains found in the IoT works that are now available on safekeeping and confidentiality, industrial experiential study, and comprehension of the difficulties that can adversely affect the putting into practice of safekeeping and confidentiality in IoT. We will take the ideas of other software engineering fields' existing maturity models into consideration IRMM levels.
Keywords Supplies analysis, Supplies engineering process controlling, ISO/IEC/IEEE 12207 & ISO/IEC 21500

1. Introduction
The creation and development of the Internet of Things (IoT) during the previous few years are examples of technical progress (IoT). The phrase was initially used in 1999 and it combines two ideas: the Internet and Thing. Our drive to research and contribute to the IoT paradigm is thus backed by its significance, as well as the necessity for a holistic approach and a multidisciplinary perspective in the creation of innovative software solutions. This is reflected in the necessity for technical abilities and skills held by various practitioners to create such software systems and the lack of dedicated software engineering processes to enable IoT. Some obstacles are centered on interaction concerns, whether between humans or objects, which are required for the paradigm to be fully established. As a first start towards resolving some of these challenges, we provide a complex conceptual framework in our proposal (Chan, 2015). The Internet and Things are physically addressable things with features like identification, sensing, actuation, and processing. The Internet is the global network of networked computers. To accomplish a shared objective, these objects can also talk to one another and work together. The Internet of Things (IoT) adds to a new technological revolution that has an impact on society. The Internet of Things (IoT) is a paradigm that allows the building of platforms from uniquely addressable entities (things) equipped with recognizing, sensing, or reacting behaviors along with processing skills that may collaborate to achieve a goal. The Internet of Things will include all areas of fascination, from tiny devices with basic programming techniques to large-scale, outstanding-performance software applications that create and analyze massive amounts of data. Because of its huge potential, IoT may use any current technology and will drive the development of fresh software platforms to address previously unanticipated difficulties (Iqbal et al., 2023). The phrase "Internet of Objects system" frequently mentions a collection of IoT strategies plus the software that controls their make contact and interaction since we require a software infrastructure to enable things to serve purposes (Zambonelli, 2017). This technology is used in numerous applications, including smart farms, smart buildings, homes, smart mobility (Hamid, Iqbal, Muhammad, Basit, et al., 2022) (Ibrahim et al., 2023), and smart cities. The Internet of Things (IoT) may offer some benefits, including raising the quality of life granting autonomy, and automating tasks and actions. IoT systems take on new problems and traits when physical devices are included. These traits are blended into a variety of aspects for instance corporeal make contacts, software, and human-interactive classifications which makes these systems more complex—system and software development benefits greatly from requirements engineering. "The elicitation, specification, and management of requirements are covered". The process of gathering and defining requirements serves as the foundation for building any product. Inconsistencies, ambiguities, and omissions in the requirements specification may impact the operator's happiness through the finished product (Rasool et al., 2023b). These factors also typically raise development costs and effort. Requirements engineering faces new difficulties in eliciting, defining, and managing requirements for IoT systems. For these systems as opposed to traditional ones, the situation is difficult to elicit, agree on, and bring about needs (Reggio, 2018). The best way to be able to define the interactions between hardware & and software pieces as well as how changes in cutting-edge context variables or allowances affect their conduct are all connected concerns that have yet to receive a conclusive resolution. IoT application development does not now use any systematic methodologies. IoT-based requirements engineering has received some proposals but they are only partial answers that do not cover all the necessary tasks. The specification represents the first step in developing a tool to assist in the activities related to requirements engineering for (IoT) systems.

2. Background
Although different security and privacy protection methods have been created, privacy and security have consistently been the key issues to be addressed in various genres of literature. The internet remains being expanded with smooth devices now a worldwide and scattered environment (Hamid, Iqbal, Aqeel, Rana, et al., 2023) (Hamid, Iqbal, Aqeel, Liu, et al., 2023) as a result of the IoT's explosive growth, which has created some major issues like privacy and security. "An international standard for systems and software

[1] Department of Software Engineering, Superior University, Lahore, Pakistan, msse-f21-004@superior.edu.pk
[2] Department of Computer Science, Superior University, Lahore, Pakistan, su92-mscsw-f22-012@superior.edu.pk
[3] Department of Computer Science, University of Gujrat, Pakistan, mubbashar.hussain@uog.edu.pk
[4] Department of Economics, IBA Karachi, Pakistan, su92-mscsw-f22-011@superior.edu.pk
[5] Professor, Department of Software Engineering, Superior University, Lahore, Pakistan, waseem.iqbal@superior.edu.pk
[6] Corresponding Author, Assistant Professor, Department of Computer Science, NCBA & E University, East Canal Campus, Lahore, Pakistan, khalid6140@gmail.com
[7] Department of Software Engineering, Superior University, Lahore, Pakistan, saleem.zubair@superior.edu.pk

engineering, ISO/IEC/IEEE 15288:2015" (14:00-17:00, n.d.) specifies the steps involved in the system life cycle and some processes of the software development life cycle.

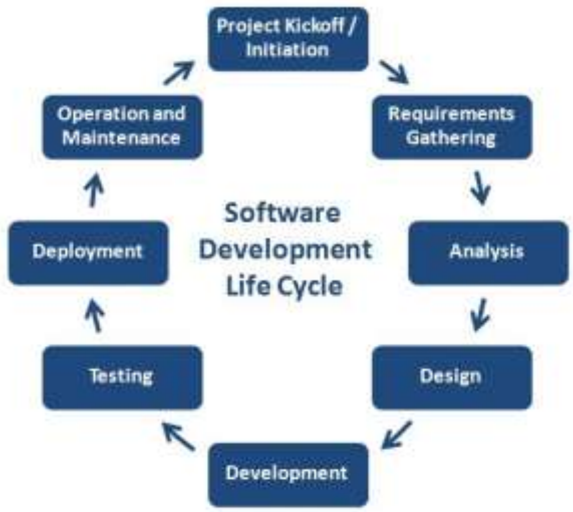

Figure 1: Software Development Life Cycle (Radoš, 2021) (Salahat et al., 2023)

"The software life cycle process is described in ISO/IEC/IEEE 12207:2017 another ISO systems" (Silva et al., 2019). Both standalone software systems and those integrated into larger more complicated systems of (IoT) are covered. This standard offers a reference model that is explained in standings of the process's goals and results that follow from the implementation of its activities. This standard though cannot meet the requirements of every system because it is meant to be broad. The goal of putting the Tailoring process into practice is to redefine existing processes or change existing ones to accomplish particular goals and results. "The life cycle processes of systems and software" are seen after a unified standpoint by ISO 15288 and ISO 12207. The System Requirements process of ISO 12207 has been renamed from the System Requirements; otherwise, the processes in these two standards are identical. And the ISO 12207 processes associated with agreements. Only in the direction from ISO 12207 to RIMM is the strength of the interaction between process areas and process demonstrated. The specified RIMM practices are fulfilled by the ISO 12207 process activities, to put it another way. The Assessment Management process (ISO 2008) the area from RIMM and a process from ISO 12207 are links between PAM and both outer project management requirements (PRM), the internal Dimension Framework of ISO 12207 is used to create requirements intended for conformance now standard. These models are meant to assist in the execution of an evaluation. Models may differ significantly in terms of their structure but they can all be linked to the Process Assessment Model (PAM) offering a structure designed for the alignment of various assessment methods. "The International Standardization Organization ISO" is tackling this problem in the contract procedures sort of ISO 12207:2008 and the "Software Engineering Institute (SEI)" has and "Capability Maturity Model & Integration (CMMI)" recently established CMMI-ACQ (Pino et al., 2010) concerning the process reference model for software procurement. For representing best practices within its area of application, each model has a unique structure, set of processes, and set of process elements.

What actions and activities described in ISO/IEC 12207 support particular CMMIACQ approaches can be learned through comparison at this level of abstraction. Additionally, a study at this level of abstraction can point up ways in which an organization's existing model (ISO 12207) can partially satisfy the criteria for establishing a new model (CMMI-ACQ). When a new model is compared to one that is currently in use inside the company it may be easier to deploy and less expensive. A benefit of a model's implementation is its lower implementation costs. A planned regulated and controlled endeavor called software process improvement seeks to improve an organization's software growth processes. It is important for many methods of software process improvement (Rout et al. 2007). There are various models involved the "Process Reference Model (PRM) and Process Assessment Model (PAM)" used to optimize the process. The goal of procedure reference models is to offer information on the procedures and the objects that may be used throughout the "procurement, supply, development, operation, and maintenance of software". SEI has created the CMMI-ACQ in a PRM for software acquisition then the ISO is addressing the covenant procedures group of ISO **12207**. Every model needs a unique structure set of procedures and a set of procedure entities for articulating best practices within its purview. The ISO 21500 direction on PM standard defines ideas and procedures that exist thought to constitute moral project management practices. An organization's maturity is related to its ability to implement and improve the procedures it employs to do its effort. Different frameworks of the container a considered by a process-oriented organization then again it can implement production procedures related to the activities carried out in its occupational area. However specific procedures are associated with the company's undertaking "SPICE for Project Management" (PMSPICE). A PRM & PAM will be described in the PMSPICE Framework. The PMSPICE must be built on ISO 21500 project management procedures. PMSPICE (PAM) should be employed toward the behavior of ISO 15504 capability by ISO 15504-2 requirements. "The international scientific community dedicated to process assessment and improvement has focused their research on developing Process Reference Models (PRMs) and Process Assessment Models (PAMs) for specific areas" which has resulted in the presence of innovative process models now various areas. The existing processes that must be technologically advanced to assess compared to the ISO 15504 normal were used to develop PRM and PAM for the PMSPICE structure. The authors of (Mesquida et al. 2015) describe the creation of the core content management system standards of a project. A method for assisting in PRM construction and validation by a collection of criteria the information gained after this knowledge spirit be used to develop our unique PMSPICE Framework for project management (Hamid et al., 2024).

2.1. PMSPICE (Project Management SPICE) Structure

"ISO 21500 identifies the recommended project management processes to be used during a project for individual phases or both". ISO 21500 process and primary Input and output are mutually exclusive. Even though they are presented in the standard as distinct fundamentals, they intersect and interact in unspecified ways. Furthermore, it does not specify the cause of all primary responses and the destination of primary outputs.

ISO 21500 can be observed in two ways: as "project management process groups and subject groups for collecting processes" by subject. ISO 21500 classifies processes into five categories: starting, planning, executing, controlling, and ending are the four stages of the process. The ten variables are "integration, stakeholder, scope, resource, time, expense, risk, quality, procurement, and communication" subject groups in ISO 21500. When developing the PMSPICE Framework described the ISO 21500 standards was one of the key inputs. This framework will be built in two stages:

2.2. Process Reference Models (PRM)

A model that includes procedure defined in terms of procedure intent then results, an architecture defining the connections between the systems. Serve the aim of describing and processing entities that will be evaluated. In the offer, a standardized nomenclature and definition of the process evaluation span Process Reference Models are in genuine sense standards (Rwelamila & Purushottam, 2012).

2.3. Process Assessment Models (PAM)

The representations are meant to assist in the execution of an evaluation. Models may differ significantly in terms of their, however, organization and substance they canister all remain linked to the Process Assessment Model (PRM) offering the framework for alignment of various assessment methods.

The evaluation of the CMMI-specific AQC's process areas for acquisition and the ISO 12207 processes associated with agreements. Only in the path from ISO 12207 to CMMI does the power exist of the interaction between process areas and the process demonstrated. The specified CMMI-ACQ practices are fulfilled by the ISO 12207 process activities, to put it another way. The CMMI-ACQ Agreement Management method (Bhatti et al., 2023) (Rasool et al., 2023a) region and a process from ISO/IEC 12207 are links between PAM and both external PRM. The internal Dimension Framework of ISO/IEC 12207 is used to create the supplies for PAM conformance in the Standard.

3. Methodology

In the background of the Internet of Things, systems describe a few proposals that deal with requirements specifications. Explaining a method of requirements for eliciting and specifying needs for IoT devices for (SEI 2007) domain modeling, requirements gathering, and IoT requirement specification, the technique makes use of Unified modeling language (UML). This approach allows for the inclusion of nonfunctional requirements as well. Pay attention to the significance of non-functional requirements. Because IoT systems depend on these writers think that IoT non-functional needs analysis is important that's why it focuses on proceeding with tangible software, network standards, and component interaction. Analysis of non-functional requirements is crucial for choosing system workings, and the achievement of an IoT structure depending on the examination of these supplies (ISO 2008).

- Pay attention to conflicts between non-functional IoT requirements. Provides a summary of recent IoT research on the creation of (IoT) structures aimed at IoT-focused software development engineering (GSEM-IoT). The first step in determining the requirements is to determine the system actors, stakeholders, and consumers because it is from them that the requirements should be gleaned. The identification of functional and nonfunctional requirements is necessary now that the main actors have been identified. The specifications also need to specify "the desirable state of things to strive for in the context of the system's operational scenario". Applying the use case concept, a method for defining requirements for smart environments. To meet the needs typical use case structure and description of smart areas and other Internet of Things (IoT) uses are provided. updated using five phases that should be followed (Hamid, Muhammad, et al., 2022) (Hamid, Muhammad, Iqbal, Nazir, et al., 2023) (Hamid, Iqbal, Muhammad, Fuzail, et al., 2022):
- Identify potential actions that might occur with action being represented through a use case.
- Designate the functionality provided by apiece use case.
- Determine who will carry out the identified actions.
- Create use case representations.
- Describe a procedure for gathering requirements for Internet of Things tools to assist emergency room actions. This method entails modeling the system with a requirements modeling language (IoT-RML) for developing system supply models.

3.1. Six IoT software development methods (SDM)

- Ignite IoT Procedure
- Internet of Things Methodology (IoT-Meth).
- Internet of Things Application Development (IoT-AD).
- Analysis of extreme limiting dilution (ELDA-IoT).
- Agent Software Product Line Process (SPLP-IoT).
- Methodology for IoT General Software Engineering (GSEM-IoT).

3.2. Ignite activities are divided into two categories

- IoT Strategy Execution
- IoT Solution Delivery
- IoT Strategy Execution

IoT Strategy Execution remains an occupational perspective that entails identifying a company's possibility, creating a business model, and deciding in what way to achieve these changes.

- **IoT Solution Delivery**

IoT Solution Delivery statements the entire development of an IoT system's life cycle, IoT-Meth tackles practicality, including requirements engineering, requirement engineering, analysis, design, and placement in part but ensures not go into detail on in what way to carry these out. IoT-AD, ELDA Meth, SPLP-IoT, and GSEM-IoT all have varying degrees of detail in terms of requirements engineering, analysis, design, and development. Following the assessment of SDM, it was determined that no one of these six methods covers all of the phases required for developing IoT systems that must be validated in real-world projects.

An IoT is a scenario-based method for supporting in the creation of (IoT) software systems, and scenario specifications are used. This method is based on Interaction Arrangements, which represent recurring interactions between abstract (IoT) domain elements.

3.3. IoT System Requirements Engineering

A requirement engineering method for IoT devices is a necessary step customized to the standardized form of the ISO 12207 procedures for business or task analysis, stakeholder needs and requirements definition, and system/software requirements definition. The Process of Requirements Engineering for IoT Structures has three sub-processes to achieve its goal:

- Project Scope Development
- Definition of an IoT Organization
- (IoT) System Supplies.

Following ISO 12207 each sub-process Process results (fast the anticipated results after the effective implementation of the sub-process), activities (groups of coherent process responsibilities), and tasks are used to define the process. As a result, each sub-process has its own goal and results, which when merged satisfy the IoT Requirements Engineering Process purpose and outcomes.

3.4. Project Scope Description of Sub-process

The problematic or opportunity, as well as the needs of stakeholders following outcomes, are produced because of the effective completion of this sub-process:

- The issue or chance is found analyzed, and described in detail.
- Identification of stakeholders and definition of stakeholders' needs
- The needs of stakeholders are translated into stakeholder requirements/goals.
- Validation of stakeholder requirements/goals

i. Define the issue or opportunity

The problem or opportunity which entails analyzing to comprehend its scope it’s defined by the setting and constraints.

ii. Make a list of potential stakeholders

Determine potential stakeholders or people with an interest in (IoT) systems.

iii. Needs of Stakeholder

Stakeholder requirements include identifying the context in which the (IoT) system will be used eliciting stakeholder needs, prioritizing the elicited needs, and developing a finishing number of stakeholder requirements.

iv. Analyze the problematic or opportunity & and stakeholder requirements

Analyze the problematic or opportunity & and stakeholder requirements to determine whether nearby enough thoughtful to determine whether the building of an IoT system is possible and useful, more information is required (da Silva et al., 2021).

v. Issue and opportunity

Describe the issue or opportunity in detail if the building of the (IoT) system this action has been accepted as an initiative and is carried out to create a more detailed definition of a problem or opportunity.

vi. Stakeholder Transform need & requirements

Convert stakeholder needs into stakeholder standards by identifying objectives, roles, constraints, and quality traits.

vii. Analyze supplies of stakeholders

Analyze stakeholder supplies to determine, among other things, whether they are essential, full, unambiguous, feasible, verifiable, and cheap.

viii. Obtain unambiguous agreement on supplies

Obtain unambiguous agreement on supplies after stakeholders indicate a common thoughtful of (the IoT) system has been accomplished.

3.5. Sub-process IoT System Definition

It is defined as an IoT scenario of sub-processing the documentation of IoT modules and selecting the preferred one based on alternative solutions. The following outcomes are produced below (De Souza et al. 2019):

- IoT scenarios
- The components and activities of the IoT system are recognized
- Another arrangement is identified
- Preferred arrangement is chosen

i. Develop an IoT solution

Developing an Internet of Things solution created scheduled stakeholder needs and then a comprehensive grasp of the issues before the opportunity.

ii. Create IoT scenarios

IoT-based scenarios must be described in narrative procedure to convey system behavior in such a manner that parties are satisfied with varying areas of expertise and then degrees of information can comprehend and add to the conversation.

iii. Identify the components of the IoT system

The scenario description (Sensors, actuators, tag scanners, software systems, cellphones, and wearables, among others, are all examples of electronic devices).

iv. Determine actions of the IoT system

Along with identifying the workings the activities taken by these workings must be recognized and defined to convey the system's performance and interdependence.

v. Identify alternative arrangement compositions

Interaction Activities of IoT represent the movement of interaction between procedures and other workings. Recurrent contact patterns must be represented; nine IoT Interaction Arrangements were defined. One of these configurations, or a mix of them, could be used in an IoT device. This exercise results in the identification of one or more alternative configurations.

vi. Assess alternative arrangement compositions

The goal of this activity is to assess the different arrangement compositions found in the previous activity to choose the best one for the particular IoT system.

3.6. Requirements for an IoT System Defined Sub-process

Define validation of IoT components following results arise after the effective execution:

- The gadget components have been defined and tested.
- The programmer components have been defined and tested.
- Use cases are defined and validated.
- • Stakeholder needs can be traced back to IoT system specs.

i. Specify device components

The purpose of this task is to develop a complete specification of workings that comprise the IoT Structure.

ii. Check the specifications of the device's components

The goal of this exercise is to figure out whether the specs of IoT device components meet the system's criteria. As part of the testing process, all nonconformities must be addressed.

iii. Specify software components

The goal of this action is to create a requirement for each software factor, including functional and nonfunctional, interaction supplies.

iv. Check the specifications of software components

Verify the software component design to ensure that it is accurate, understandable, consistent, and viable to execute and test. This action involves resolving all non-conformities that have been identified.

v. Explain use cases

Describe use scenarios. The goal of this exercise is to create a comprehensive account of each situation, including the interaction flow among components, as well as substitute and exception processes. A summary of the business principles is included in the use case description.

vi. Validate Use Cases

Examine the Use Case explanations to make sure that they are accurate, comprehensible, and consistent. This action involves resolving all found non-conformities.

vii. Create and sustain traceability

Establish and keep transparency between stakeholders and system software requirements. You can use traceability to validate any or all IoT system criteria to satisfy the demands of stakeholders.

4. Results and Discussion

4.1. Project Management

- **Scope Management**

The most difficult and critical duty is controlling the project scope due to specifications frequently shifting and at times those changes impact the project's opportunity. Once a project has started, a written agreement is created between the customer and the software business, requiring fulfillment and value one extra. Every project drive experience changes from start to end, and scrum provides a simple method to deal with them after the team lead's directions, the developer executes these requirements following the needs the team analyses and decides their extent and relevance. This is the action sprint, if some adjustments are rapidly implemented through obtaining "Change Control Board (CCB)" permission ((Noaman et al., 2023).

4.2. Time and Cost Management

An endeavor is one of the main challenges later establishing scope and the majority of IT developments have a poor track of the cost, making it especially difficult to handle both at the same time. Projects may have also away over prices or behind plan. Since the effort is divided into small chunks and sprints are created, developers can construct their chunks on time and within budget by utilizing methodologies, that elegantly tackle this problem. As a result, organizational development and efficiency are extremely important from top to bottom and cost-effective.

4.3. Quality Management

When an endeavor or product has been completed with all of its structures, the excellence of the project is obvious as the prerequisite that development be completed on time, within the financial plan, and with a full covering of its planned purview SPM. Because of methods place a high value on producing fine, high-quality goods, they provide best practices that are constantly monitored to produce better results. This approach provides the best methods for saving time and ensuring supply compliance if quality is a priority. In terms of quality, all techniques are successful. A collection of quality requirements for all software industries that have to guarantee that some undeveloped projects or goods are easy to use, released on schedule, and conform to the project's scope.

4.4. Risk Management

In most companies, individuals serve as project assistants. As a result of this techniques put a tough emphasis on team management and team growth, provisional on the people's degree of expertise and capability now the topic. However, a big team ensures that a good product and pleased clients are not guaranteed. Two conflicting theories exist. While some organizations do not place a high value on teamwork, the majority say that the technique enables them to employ only people who are driven by objectives and can work well in groups. Furthermore, emphasizes cooperation and member participation. An organization's HR management has a significant influence on (Hamza et al. 2020) project management when using their project management (PM) methodology because tasks are spread among the team and methods require that team members help one another after completing distinct responsibilities.

4.5. Human Resource Management (HRM)

The procedures for organizing and managing the project crew are included in Human Resource Management. It all comes down to making the best use of individuals, including donors, customers, and other collaborators to separate participants. HR planning as well as project team formation, growth, and administration are altogether aspects of Human Resources (HR) administration. The project manager is in charge of creating the project team and bringing it together to finish the project (Mesquida et al., 2015).

4.6. There are two types of (HRM)

1. **Administrative**
2. **Behavioral**

- Administrative responsibilities include employee interactions, pay, and assessment, as well as compliance with government laws and review. Much of the manager's administrative work remains guided by groups and non-project organizations.
- Behavioral elements concern their assignment team personnel interactions as a group as well as their interactions with people outside the scope of the undertaking. These include collaboration, motivating, team development, and dispute resolution.

4.7. Communication Management

The methods for ensuring that project information is produced collected, disseminated, saved, recovered, and disposed of in a timely and appropriate way.

4.8. Procurement Management

Procedures for obtaining products, services, or results from sources other than the project team to finish the job.

5. Conclusion

A Procedure for Specifying Requirements for IoT Structures The entire procedure remains a customized form of the ISO IEC/IEEE 12207:2017 intended to suit the requirements of IoT system engineers. Our work has made major advances to IoT System Classification and IoT System Needs Definition. A structure that depicts the difficulties in developing IoT devices it is made up of seven components, all of which are linked to IoT software structures (connectivity, effects, performance, smartness, issue area, interactivity, and environs). The Requirements Engineering technique described in this piece is for institutions, and the procedure necessitates an assessment of real-world developments to identify and resolve flaws. The project management will begin once the process's assessment and security have been established.

References

14:00-17:00. (n.d.). *ISO/IEC 15288:2008*. ISO.

Bhatti, S., Hamid, K., Bashir, A., zafar, zishan, raza, ahmad, & Iqbal, M. waseem. (2023). *Solutions, Countermeasures, And Mitigation Methods For The Rise Of Automotive Hacking*. *56*, 77–99.

Chan, H. C. Y. (2015). Internet of Things Business Models. *Journal of Service Science and Management*, *08*(04), 552–568.

da Silva, D., Pedraça de Souza, B., Gonçalves, T., & Travassos, G. (2021). *A Requirements Engineering Technology for the IoT Software Systems*.

Hamid, K., Ibrar, M., Delshadi, A. M., Hussain, M., Iqbal, M. W., Hameed, A., & Noor, M. (2024). ML-based Meta-Model Usability Evaluation of Mobile Medical Apps. *International Journal of Advanced Computer Science and Applications*, *15*(1).

Hamid, K., Iqbal, M. waseem, Aqeel, M., Liu, X., & Arif, M. (2023). *Analysis of Techniques for Detection and Removal of Zero-Day Attacks (ZDA)* (pp. 248–262).

Hamid, K., Iqbal, M. waseem, Aqeel, M., Rana, T., & Arif, M. (2023). *Cyber Security: Analysis for Detection and Removal of Zero-Day Attacks (ZDA)* (pp. 172–196).

Hamid, K., Iqbal, M. waseem, Muhammad, H., Basit, M., Fuzail, Z., † Z., & Ahmad, S. (2022). *Usability Evaluation of Mobile Banking Applications in Digital Business as Emerging Economy*. 250.

Hamid, K., Iqbal, M. waseem, Muhammad, H., Fuzail, Z., & Nazir, Z. (2022). Anova Based Usability Evaluation Of Kid's Mobile Apps Empowered Learning Process. *Qingdao Daxue Xuebao(Gongcheng Jishuban)/Journal of Qingdao University (Engineering and Technology Edition)*, *41*, 142–169.

Hamid, K., Muhammad, H., Iqbal, M. waseem, Bukhari, S., Nazir, A., & Bhatti, S. (2022). Ml-Based Usability Evaluation Of Educational Mobile Apps For Grown-Ups And Adults. *Jilin Daxue Xuebao (Gongxueban)/Journal of Jilin University (Engineering and Technology Edition)*, *41*, 352–370.

Hamid, K., Muhammad, H., Iqbal, M. waseem, Nazir, A., shazab, & Moneeza, H. (2023). Ml-Based Meta Model Evaluation Of Mobile Apps Empowered Usability Of Disables. *Tianjin Daxue Xuebao (Ziran Kexue Yu Gongcheng Jishu Ban)/Journal of Tianjin University Science and Technology*, *56*, 50–68.

Ibrahim, A., Al-Rajab, M., Hamid, K., Aqeel, M., Muneer, S., Parveen, M., & Saleem, M. (2023). *Usability Evaluation of Kids' Learning Apps* (p. 10).

Iqbal, M. W., Khaliq, K., Al-Dmour, N. A., Aqeel, M., Ali, N., & Hamid, K. (2023). Internet of Things (IoT) in Smart Cities: A Statistical Survey. *2023 International Conference on Business Analytics for Technology and Security (ICBATS)*, 1–6.

Mesquida, A.-L., Mas, A., & Barafort, B. (2015). The Project Management SPICE (PMSPICE) Process Reference Model: Towards a Process Assessment Model. In R. V. O'Connor, M. Umay Akkaya, K. Kemaneci, M. Yilmaz, A. Poth, & R. Messnarz (Eds.), *Systems, Software and Services Process Improvement* (pp. 193–205). Springer International Publishing.

Noaman, S., Ibrahim, A., Ali, L., Iqbal, M., Ashraf, A., Haseeb, U., Muneer, S., Almajed, R., & Hamid, K. (2023). *Playing the video games during COVID-19 pandemic and its effects on player's well-being* (p. 5).

Pino, F. J., Baldassarre, M. T., Piattini, M., Visaggio, G., & Caivano, D. (2010). Mapping Software Acquisition Practices from ISO 12207 and CMMI. In L. A. Maciaszek, C. González-Pérez, & S. Jablonski (Eds.), *Evaluation of Novel Approaches to Software Engineering* (pp. 234–247). Springer.

Radoš, A. (2021). *Izazovi Upravljanja Projektima Razvoja Softvera Primjenom Scrum Okvira Za Agilno Upravljanje Projektima*, University of Zagreb. Faculty of Economics and Business. Department of Organization and Management].

Rasool, N., Khan, S., Haseeb, U., Zubair, S., Iqbal, M. W., & Hamid, K. (2023a). Scrum And The Agile Procedure's Impact On Software Project Management. *Jilin Daxue Xuebao (Gongxueban)/Journal of Jilin University (Engineering and Technology Edition)*, *42*, 380–392.

Reggio, G. (2018). A UML-based proposal for IoT system requirements specification. *Proceedings of the 10th International Workshop on Modelling in Software Engineering*, 9–16.

Rwelamila, P. D., & Purushottam, N. (2012). Project Management Trilogy Challenges in Africa—Where to from Here? *Project Management Journal*, *43*(4), 5–13.

Salahat, M., Said, R., Hamid, K., Haseeb, U., Ghani, E., Abualkishik, A., Iqbal, M. W., & Inairat, M. (2023). *Software Testing Issues Improvement in Quality Assurance* (p. 6).

Silva, D., Gonçalves, T. G., & da Rocha, A. R. C. (2019). A Requirements Engineering Process for IoT Systems. *Proceedings of the XVIII Brazilian Symposium on Software Quality*, 204–209.

Zambonelli, F. (2017). Key Abstractions for IoT-Oriented Software Engineering. *IEEE Software*, *34*, 38–45.