

# Requirement Engineering and Software Product Line Scoping Paradigm

Ahmed Mateen, Zhu Qingsheng, Faisal Shahzad

**Abstract**—Requirement Engineering (RE) is a part being created for programming structure during the software development lifecycle. Software product line development is a new topic area within the domain of software engineering. It also plays important role in decision making and it is ultimately helpful in rising business environment for productive programming headway. Decisions are central to engineering processes and they hold them together. It is argued that better decisions will lead to better engineering. To achieve better decisions requires that they are understood in detail. In order to address the issues, companies are moving towards Software Product Line Engineering (SPLE) which helps in providing large varieties of products with minimum development effort and cost. This paper proposed a new framework for software product line and compared with other models. The results can help to understand the needs in SPL testing, by identifying points that still require additional investigation. In our future scenario, we will combine this model in a controlled environment with industrial SPL projects which will be the new horizon for SPL process management testing strategies.

**Keywords**—Requirements engineering, software product lines, scoping, process structure, domain specific language.

## I. INTRODUCTION

THE RE is a systemic and composed strategy of rousing, explaining, orchestrating, sorting out, showing, tolerating, and managing the essentials of a structure. The process of developing a set of products that share a majority of features is called Software Product Line development practice. A typical product line practice has two important activities: The Core Asset Development and the Product Development [1]. When SPLE is combined with component-based development approach, it increases the scope of reusability which enhanced the business process intelligence in terms of time and cost without compromising on the quality of the product for better decision making. However, in some cases, the configuration process may be made public to end users, e.g. when the participation in choosing concrete options serves a business value for the owner of the SPL. SPL is about producing software for a well-defined market, from a base software architecture, with a predefined set of options called variation points [2]. The review unveils that there is a wide degree of developing new strategies and frameworks in need planning to decide issues seen in various SE procedures. The focus of SPLS practice is to address and develop a common architecture with component based approach which will embed the component based functionality in the developed

framework to overcome the problems in software reusability approach [3]. It requires careful design to include appropriate levels of parameterization, generalization and specialization, and extension which lead the design process to accomplish the project's goals effectively [4].

They are used to describe variable and common properties of products in a product line, and to derive and validate configurations of software systems. An exactly written work overview driven by the researcher on RE issues revealed a couple of factors adding to these arrangements of RE issues [5]. Necessities examination, furthermore called essentials planning, is the path toward choosing customer wishes for another or balanced thing. These parts, called necessities, must be quantifiable, relevant, and point by point. In programming outlining, such necessities are as often as possible called helpful conclusions. Requirements examination is a basic piece of undertaking organization. This investigation will elucidate the issues of essentials building process and their sensible plans by working up a model. Also, it will exhibit necessities working in a spry programming headway environment [6].

Common problems encounter in software RE development methods: For programming change, various methods exist and it is not achievable to fuse each one of them here. In this manner, we have picked largely used techniques for studies. These methods consolidate Traditional programming progression, Object masterminded programming change, skillful programming headway, Component-based programming headway (CBSD) and Service arranged programming progression (SOSD). We first separate ordinary issues and troubles which are found in all item headway strategies [7].

### A. Acknowledgment of Safety at RE Level

As the late systems end up being more unavoidable, flexible and operational by various customers, the essential techniques and data have been the target of security ambushes. The attempts have been made to perceive, appear, and guarantee threats and vulnerabilities. Thus, managing RE is responsive and focuses on low-level security essentials. The work on unusual state security approaches on techniques for arranging, exhibiting, and believing is done. In any case, the behavioral detail of threats and ambushes should be streamlined at RE and arrangement level. So, to comprehend the security and insurance degree at RE level is a perceptible test related to Requirement Specification.

### B. Reconciliation of RE Models

The showing customs, approaches, and systems enhance the

Ahmed Mateen and Zhu Qingsheng are with the College of Computer Science, Chongqing University China (e-mail: ahmedmatin@hotmail.com, ahmedbuttar@uaf.edu.pk).

Faisal Shahzad is with the University of Agriculture Faisalabad, Pakistan.

RE techniques. Showing theory which joins RE showing parts is delineated. Most research endeavors focus on a singular RE issue, for instance, elicitation and there has been little work on the interconnection of essential models and joining RE stages. All around portrayed procedures are required to interrelate RE targets, circumstances, data, limits, state-based direct and confinements. This issue impacts all RE stages, and the ensuing effect will be on building and diagram of the structure [8].

#### *C. Elicitation Technique Collection*

There are a lot of elicitation procedures accessible in the business for finishing elicitation errands. Every one of the methods is utilized as a part of the half-breed way to assemble prerequisites from clients. In any case, there is no institutionalized procedure committed for the separate worldview.

#### *D. Requirements Reuse*

The reusability approach is very effective in terms of costs reduction, increasing quality and productivity in software development. For the successful implementation of reusability approach, all the technical and non-technical attributes must be clearly understandable, and goals should be focused on how the organization business strategies meet the system requirements [9].

#### *E. Improvement in Requirement Quality*

The necessities evoked from accomplices may be ambiguous, deficient, clashing, mistaken and out-of-date. A couple of essentials are shown with simply concentrated dialects rather than business space wording. The Quality Assurance (QA) undertaking should be begun from the RE stages itself and it continues towards resulting stages. The both parties like development and Quality Assurance have to work jointly in a way that communication door is always open in every stage of SDLC process which reduced the defect if captured in final stage and increased the cost and time and delayed the whole product. So, before adopting the reusability approach, all the implementation goal and achievements should be precise and clear in terms of software quality with respect to integration of components in the whole software development process [10].

#### *F. Missing Requirements*

The customer does not think about giving 100% of the requirements for the structure to be created. The medium size and tremendous system winds up with a large number of requirements and is divided into various subsystems. It is hard to perceive some missing essentials, and their non-attendance is as often as possible missed until the system is joined, attempted or sent. Main focusing factor during this process is to have good relationship with staff and management team so every missing aspect can easily be traced and understood with institution business flow and working [11]. Grown-up procedures and frameworks can be used to deal with the trial of missing essentials.

#### *G. Self-Loader Process for Generic Template Creation*

Presently, there are various organizations available for get-together necessities in an instinctive way. This approach guides the team by learning documentation and reporting which guide about modules compatibility and is re-reusable in whole development. This overall has a positive effect in product development in terms of cost and time saving by mapping and sorting the particular process according to the requirement of the module which will lead to a movement toward the strategy of the self-loader essential planning.

#### *H. Excessive Requirements Volatility*

The use of iterative and incremental model of programming headway is prodded in light of conforming steady developing essential. Regardless, if essentials change in an uncontrolled way, then it may have a critical effect on existing building and plan. To reduce the volatility effect, team focused on the main factors which have great impact during software development phases like change type and dependency measures which are correlated with additional work and increase the cost of whole project [4]. Requirement Specification has great effect on business change and development change, so formulation of these measures should be incorporated in the analysis phase and approved by the committee which will save effort of re-designing and re-development of the product.

#### *I. Inadequate Requirements Management*

Numerous tasks store and deal with their necessities in paper reports and spreadsheets with divergent organizations overseen by various profile groups [8]. The decentralized and individual administration of scattered prerequisites makes it hard to approve colleagues for performing operations on the necessities. In the event of worldwide programming improvement, dynamic changes happen at all the destinations, and administration of circulated necessities is difficult to execute. The metadata of these prerequisites must be ranked with their status, must be elaborated with reasoning for adopting the method, and it must highlight the changes for the approval by the management.

#### *J. Exactness and Performance Measurement of Requirements*

The operational proficiency of the item checks the general execution of the thing made. The business requires new regard creation and that is the reason of prominent need of productivity and tuning in programming. In any case, when requirements needs are found, planned and prioritized according to new structuring approached in a systematic way then it will be easily collaborated in the main system at final stage [7]. However main parameters during Re-Engineering process are the precision and execution estimation, and these should be prioritized before developing the programing interface.

#### *K. Interactive RE Tool Maintenance*

Various associations are using assurance record, essential spreadsheets or RDBMS tables to store and manage their essential needs. The requirements, their related models and

layouts are stored in a manner that can be easily traceable at every stage in the whole process. Without adequate, impeccable and consolidated instrument support, essentials get the chance to end in enormous flaws and errors, untraceable and obsolete [3]. The ideal, adaptable, successful and straightforward contraption should be used to catch requirements, their charts, related substance and metadata. The mechanical assembly should have the highlights like profitable elicitation interface, traceability wizard, exchange plan and joining of a central store.

#### L. Communication Gap

The correspondence hole between customer and RE gathering is a critical issue. This gap will show its bad effect in the setup and execution phases. Instead of these issues, synchronization problem of the whole system is also affected. For a better understanding of RE process, proper document traceability and methodology should be adopted within the module which will be beneficial to understand the work flow activities. This task does not depend on team programming capabilities but team competency is important factor so precise interview questions should be performed to find out the hidden information of the organization.

#### M. Conflicting and Confusing Requirements

The unmistakable accomplices' suppositions, targets and needs may have different suggestions and may conflict with darkening words. While motivating the necessities, the wording, watchwords and space learning should be fittingly prompted. The methodologies should be depicted all around to decide the conflicts of essentials. To oust the obscurity and conflicts, the assembled necessities can be secured in configuration which demonstrates the right techniques [2]. The customers can check this model and oust any conflicting need in a step-wise refinement show.

#### N. Elimination of Extraneous Requirements

The game-plan of prerequisites asked from assistants may join two or three focuses which are not fundamental under any condition. Imperative flaws can be experienced regarding dataset and unessential necessities. The course toward disposing of inconsequential six fundamentals is dull. So, the development and sharp refinement resolve the issues faced by clients.

#### O. Prioritization of Necessities

The accomplice's floundering attitude changes the need for the requirements. Rousing necessities from accomplices by their position in the affiliation catches the system of need errand. The ID system is required to give a rating on need essentials [12].

## II. MATERIALS & METHODS

Programming Product Lines Software Product Lines However, before we present the case, we will take a little bypass into the premise of SPLE. The fundamental contrast from "ordinary", unique programming improvement, is a coherent detachment between the advancement of center, reusable programming resources (the stage) and genuine applications. Amid application improvement, stage programming is chosen and arranged to meet the needs of the application. The Product Line's shared traits and inconsistencies are depicted in the Problem Space. This mirrors the craved scope of utilization ("item variations") in the Product Line (the "area") and only calls for software artifacts to be created when reuse is predicted in one or more products in a well-defined product line. Thus, while creating an item variation, the application engineer utilizes the issue space definition to depict the sought mix of issue fluctuations and to actualize the item variation.

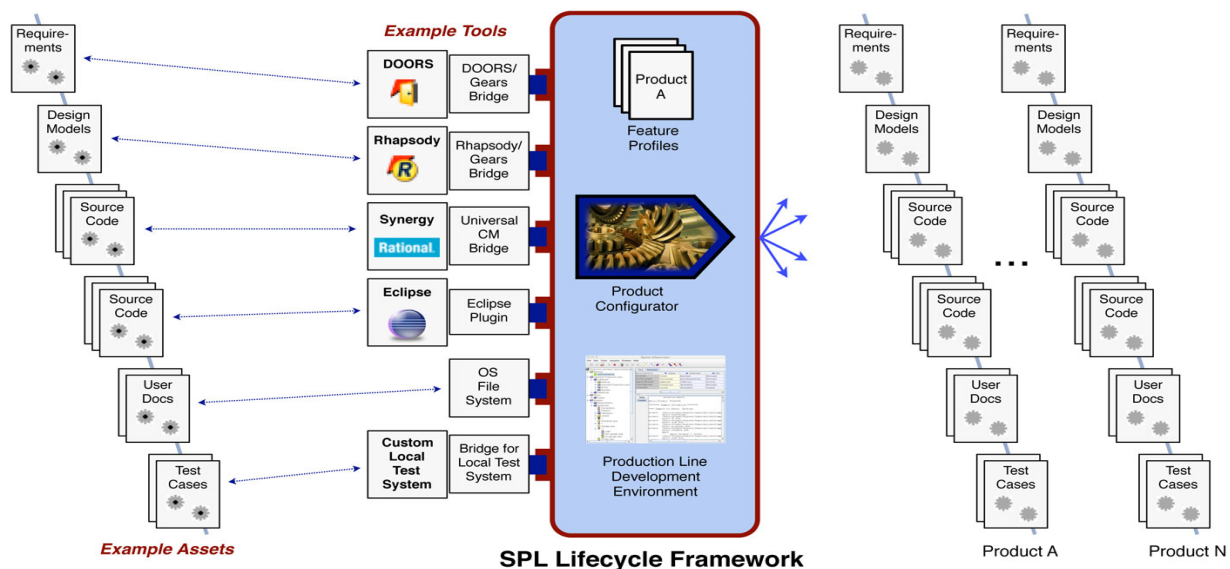


Fig. 1 SPLE with the SPL Lifecycle Framework

A related Solution Space depicts the constituent resources of the Product Line (the "stage") and its connection to the issue space, i.e. rules for how components of the stage are chosen when certain qualities in the issue space are chosen as a major aspect of an item variation. The four-section division coming about because of the mix of the issue space and arrangement space with area and application designing appears in requirements and technology volatility [13].

The SPL SC Study began with the pre-checking stage, trailed by space perusing, item checking [16] and resources perusing. Amid the pre-checking, a few gatherings were booked with various venture partners to assess the SPL accessibility, advantages, disadvantages, and the area in which it would be executed. Next, the space perusing was held expecting to recognize the areas and sub-spaces with more potential to make the items out of the product offering. The product scoping was performed to distinguish survey highlights, recognize items and build and approve the item outline. At last, resource perusing made measurements, connected them and organized the components on the item delineate. The SPL SC study was performed by nine individuals from the SPL group and three programming engineers from the organization which were nominated for following designations: one checking master, one product offering director, seven designers, one modeler, one supervisor, one market investigator and one business expert.

One representative satisfied the parts of venture director, space master, and engineer. Amid the execution, we recognized 3644 elements. Toward the end, a pool of 840

components, four items, and 102 sub-areas was merged and approved amid gatherings with the SPL colleagues. The wiki framework and the items from the organization were utilized to gather information about the items, spaces, and element. Each of them was merged in an SPL vision report and put away in a store. The SPL RE study began with the model degree, trailed by characterizing necessities, and utilizing cases in a steps wise manner. This contextual analysis concentrated on two emphases in the RE teach. The main emphasis was led by the prerequisites examiners and investigated by an alternate group. Every stage (display scope, characterizing prerequisites, and utilizing cases) in the necessities building discipline brought about the ancient rarities – components, necessities, and utilize cases [17]. After the necessities, investigators played out every one of the stages and the review procedure was begun to guarantee the nature of the antiques.

The Requirement Analyst is the main role to perform this process. Table II presents a summary of the process, with activities, tasks, inputs, outputs and roles. The second cycle in SPL RE was performed with contrasts. For instance, the assessment was completed after every stage, that is, after the elements were characterized by the model degree step, the overseers led gatherings to investigate the predetermined elements before beginning the following stride (characterizing necessities). We gathered information from the wiki framework documentation, delivered checking antiquities, area specialists, and the legacy items.

TABLE I  
A COMPARATIVE ANALYSIS

Issues and challenges	DRA MA Kim J et al. [18]	Tanhaei, M. et al.[19]	Mellado, D et al.[20]	Guendouz, A et al. [21]	Alam, M et al. [22]
Product line management					√
Requirements	√	√	√	√	√
Variability management	√	√	√	√	√
Design and architecting	√	√	√	√	√
Implementation	√	√	√	√	√
Cross-cutting concerns			√		√
Testing & validation		√	√		√
Maintenance & Evaluation			√		√

TABLE II  
SUMMARY OF THE FPLE-RE PROCESS

Activities	Tasks	Inputs	Outputs	Roles
Model scope	Elicit	Existing assets, product map, glossary	Raw DRS, glossary	Requirements analyst, domain expert, decision-markets, domain analyst
	Model features	Product map, raw DRS	Feature model, trace links	Requirements analyst
	Verify	Feature model	Verification, report, change request	A domain expert, domain analyst, requirement analyst, SQA
Define requirements	Elicit	Existing assets, glossary, feature model	Glossary, raw DRS	Domain expert, end user, requirements analyst
	Describe responsibilities	Feature model, raw DRS	Domain requirement trace links	Requirement analyst
	Verify	Domain requirement	Verification report, change request	Domain expert, end users, requirement analyst, SQA
Define use cases	Elicit	Existing assets, features model, domain requirements, glossary	Glossary, raw DRS	Domain expert, requirement analyst, end user
	Describe use	Domain requirements, raw DRS	Domain use cases, trace links	Requirements analyst
	verify	Domain use cases	Verification report, change request	Domain expert, requirement analyst, end users, SQA

### III. RESULTS

This work showed essential methodology for programming items which offers an effective way to guide about necessities, their detail examination, and exchange them after checking. This system relied upon a wide systematic review whose results can be used to a genuine course of actions, techniques and activities for worked in SPL. Besides, a test case was directed to survey the feasibility, capability and convenience of the FPLE-RE preparation. The subjective examination of the strategy was better than average prospective as demonstrated by customers' satisfied. On another side, a couple of inconveniences were represented by the matter, demonstrating changes at the same time. Concerning quantitative examination, the trial showed that the technique has a not too bad feasibility. In any case, its adequacy can be upgraded with the usage of an appropriate supporting

mechanical assembly. Along these lines, this composition can be seen as a critical duty to the zone of programming reuse and requirements building. Fig. 3 [15] exhibits the histogram with the assignment thickness of the found inconveniences.

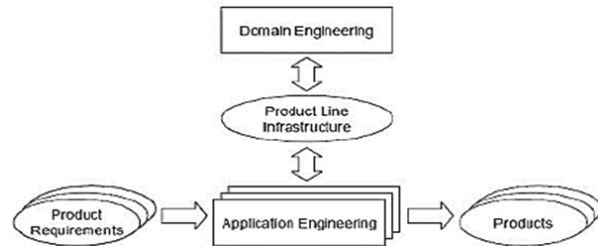


Fig. 2 Overview of SPLE activities

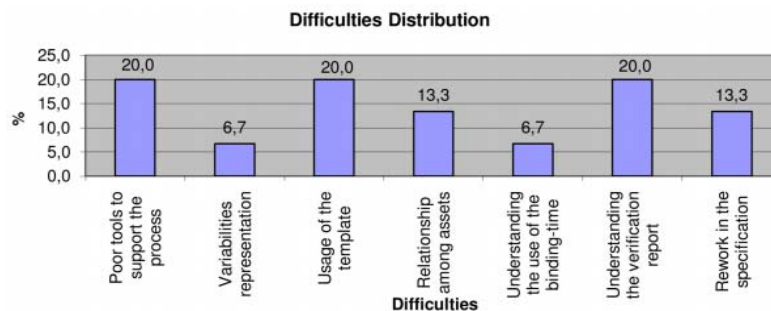


Fig. 3 Difficulties histogram

TABLE III  
RESULT OF DEFECT INDEX

Measure	Defect Index
Mean	3.15
Maximum	4.42
Minimum	3.69
Standard Deviation	0.78
Defect Index	4.11
Null Hypothesis	>5

As demonstrated in Fig. 3 [18], one subject (ID 7) experienced issues in addressing variabilities, exceptionally toward the beginning, when the gathering was a slightly bit mixed up for the components, partly caused by nonattending to the association of both SPL thoughts and the rental space. For another matter (ID 3), the method must upgrade the depiction 60 about the usage of the coupling time. Three matters (ID 3, 6, 8) detailed [6] inconvenience in the usage of configuration in XML, which can be sorted out by labeling [6], [14], [19]. A few matters (ID 1, 2, 6) experienced issues in the utilization of the system in view of nonattendance of a gadget to assist the full strategy. Three matters (ID 3, 5, 7) experienced issues of fathoming the objective of checking matters, so there were issues in its use. Two matters (ID 6, 7) detailed that the relationship among assets (segments, association and utilize cases) must be made progressive. Around the end, two matters (ID 5, 8) showed patch up in

assurance. Table III shows [18] that our results were correct as we achieved SDV less value.

The standard deviation is small thus, the defect index and error distribution results validate the process potential to aid in generating domain requirements specification in compliance with its standards.

### IV. CONCLUSION

This examination displayed the idea, orchestrating performance, examination and explanation, and completed a trial research to survey the FPLE-RE handle. The review inspected the methodology convenience, sufficiency and capability of the set of programming. The subjective examination exhibited that the methodology has a more unmistakable potential according to customers' satisfaction. Of course, a couple of inconveniences were represented by the subjects, demonstrating changes all the while.

As to quantitative examination, the examination displayed that the methodology has a not too bad sufficiency. Regardless of this fact, its capability can be upgraded with the use of a fitting supporting instrument. In fact, even with the decreased number of subjects participating in the examination, and the perceived authenticity threats, the results prescribe that the proposed RE prepared for SPL may be achievable after a couple of arrangements. Results express that the trial was working in a particular association. To examine conceivable advantages or exchange offs of the offered procedure, more

analyses ought to be led later on. The suggestions to diminish advancement time and enhance item quality make the product reuse approach extremely appealing for programming affiliations. Nowadays, a couple of affiliations are getting programming item offerings approaches as a way to deal with fulfilling these favorable circumstances.

The crucial responsibilities of this work can be part of the following perspectives: i. A substantial considerable review on essentials working for programming item offerings; ii. The definition and methods of essentials methodology for programming item: iii. The headway of a gadget to reinforce the proposed discovered show; iv. The execution of an exploratory review which surveyed the proposed system. These responsibilities are further depicted next. Additionally, the 77 requirements and utilized cases are shown fundamentally, engaging their instantiation for SPL and thing.

#### V.FUTURE WORK

In our future scenario, we will combine this model in a controlled environment with industrial SPL projects which will be the new horizon for SPL process management testing. The strategies and standards in the SPL system require rectification and making a technique which could be delivered at lower cost.

#### REFERENCES

- [1] Y. Alsultanny, and A. M. Wohaishi "Requirements of Software Quality Assurance Model. Second International Conference on Environmental and Computer Science" 2009 pp. 23.
- [2] J. Bayer, O. Flege, P. Knauber, R. Laqua, D. Muthig, K. Schmid, T. Wide, and J. M. DeBaud, methodology to develop software product lines, in: Proceedings of the 5th ACM SIGSOFT Symposium on Software Reusability, 2011 pp. 122–131.
- [3] J. Bosch, and P. M. Bosch-Sijtsema "Introducing agile customer-centered development in a legacy software product line". Software: Practice and Experience, 2011 41(8), 871–882.
- [4] J. Erickson, and K. Siau. "Theoretical and Practical complexity of modeling methods," in ACM Communications, vol. 50, 2007 pp. 46–51.
- [5] T. Gorschek, A. Gomes., A. Pettersson, and R. Torkar, Introduction of a process maturity model for marketdriven product management and requirements engineering. Journal of Software: Evolution and Process, 2012 24(1), 83–113.
- [6] I. Inayat, L. Moraes, M. Daneva, and Salim, S. S. "A Reflection on Agile Requirements Engineering": Solutions Brought and Challenges Posed. Scientific Workshop Proceedings of the XP2015, Trondheim, 2015p. 6.
- [7] S. Lim, and A. Finkelstein Stake Rare: Using Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation. IEEE Transactions on Software Engineering, 2012 38(3), 707–735.
- [8] I García, Pacheco C, Mendoza E, Calvo-Manzano JA, Cuevas G, San Feliu T. Managing the software process with a software process improvement tool in a small enterprise. Journal of Software: Evolution and Process. 2012 24(5): 481–91.
- [9] L. Antovski and F. Imeri. "Review of Software Reuse Processes" International Journal of Computer Science Issues 2013 volume 10, 6(2), 83–88.
- [10] B. B. Morales, S. Crespo and Clifton C. "Reuse of Use Cases Diagrams: An Approach based on Ontologies and Semantic Web Technologies" International Journal of Computer Science Issues 2012 volume 9, 1(2), 24–29.
- [11] K. Wnuk, D. Callele, E.-A. Karlsson, and B. Regnell, Controlling Lost Opportunity Costs in Agile Development—The Basic Lost Opportunity Estimation Model for Requirements Scoping, in Software Business, Springer, 2012 pp. 255–260.
- [12] <http://www.biglever.com>, The Software Product Line (SPL) Lifecycle Framework: Overview of the framework', 2017. (Online). Available: [http://www.biglever.com/newsletters/SPL\\_Lifecycle\\_Part\\_2.html](http://www.biglever.com/newsletters/SPL_Lifecycle_Part_2.html). (Accessed: 13- Dec- 2017).
- [13] <http://www.methodsandtools.com>, Software Product Line Engineering with Feature Models' 2017. (Online). <http://www.methodsandtools.com/archive/archive.php?id=49>. (Accessed: 13- Dec- 2017).
- [14] Yang, J. Hu, Y. Q. Wang, G. Ruhe, and H. Wang —Value-based portfolio scoping: an industrial case study, I in Proceedings of the Third International Workshop on Product Line Approaches in Software Engineering, 2012 pp. 45–48.
- [15] K. Pohl, The three dimensions of requirements engineering. In Seminal Contributions to Information Systems Engineering 2013 pp. 63–80.
- [16] Linden, F. van der, J. Bosch, E. Kamsties, K. Kansala and H. Obbink, Software product family evaluation, in: Proceedings of the 3rd International Conference on Software Product Lines 2014 pp. 110–129.
- [17] M. Khurum, N. Uppalapati, and R. C. Veeramachaneni.,—"Software requirements triage and selection: state-of-the-art and state-of-practice", I in Software Engineering Conference (APSEC), 2012 19th Asia-Pacific, vol. 1, 2012 pp. 416–421.
- [18] F. S. N, Danuza. A requirements engineering process for software product lines, Universidade Federalde Pernambuco Centro de Informática 2009.
- [19] D. F. S. Neiva, A requirements engineering process for software product lines, MSc Dissertation, Page 51 august 2009.



**Ahmed Mateen**

#### Academic Background

2000/08 Graduated from Punjab University, Lahore, Pakistan. (Undergraduate)  
 2003/05 Graduated from The University of Lahore, Pakistan. (Master)  
 2007/04 Graduated from University of Agriculture Faisalabad, Pakistan. (M.Phil.)  
 2004/05 ~Present Lecturer in Computer Science

Department at University of Agriculture Faisalabad, Pakistan  
 2017/09 ~ Doctoral Student in College of Computer Science, Chongqing University, China

#### **Academic/ Professional awards (Govt./Institution/Society)**

- Member of Core Committee (Diploma in Agribusiness by LUMS)
- System Operating Engineer in Direct rate of Distance Learning (UAF)
- Member of Senate (University of Agriculture Faisalabad)
- Member of Departmental Board of Studies
- Member of Internet Society (USA)
- Member of Editorial Board of Journal of Computing and Applications
- Member of Turnitin Society (USA)
- Executive Member of Academic Staff Association (UAF 2012)
- Joint Sectary of Academic Staff Association 2010 UAF
- Chief Councilor ASA 2009 UAF
- Member of Computer Science Teacher Association (USA)
- Member of Social Science Research Network (United State)
- Association Computing Machinery (ACM) USA.
- Australian Computer Society Inc. (ACT)
- Member of Editorial Board of Journal of Computing and Applications. (JCA)
- Member of Editorial Board of Central Asian Journal of Information Technology (CAJIT)
- Member of Software Engineering Society UAF
- Member of the Seminar/Special problem Commatic CS Department

#### **Academic Interest**

Agriculture Modeling and Simulation, Wireless Networks, Real Time Systems, Distributed Cloud and Cluster Computing, Big Data, Software Engineering, Machine Learning.