International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 5, Issue 3, May-June 2016 **ISSN 2278-6856**

Optimal Release Planning and Software Reliability Modeling for Multi-Release Software

Tanuja Mulla[1], Prof.Amol Kadam[2], Dr. S. D. Joshi[3]

[1]M.tech Research Scholar Dept. of Computer Engineering
Bharati Vidyapeeth Deemed University College of Engineering, Pune.

[2]Phd. Research Scholar Dept. Computer Engineering
Bharati Vidyapeeth Deemed University College of Engineering, Pune.

[3] Dr. Research Scholar Dept. Computer Engineering
Bharati Vidyapeeth Deemed University College of Engineering, Pune.

Abstract

Reliability of Software has been major Issue in Software industry. Failure Free Code Development is to be achieved with in time of Software version upgrades. Although this give rise to numerous faults. Analyzing and resolving these Defects to achieve Software reliability is objective of Research. To overcome these defects Software domain has come up with software Reliability and Growth Models (SRGM's). This Model predict faults that would occur and present count of defects in module. While Resolving this defects SRGM's fail to achieve optimal efficiency in issues removal. Fault removal metric help product developer to evaluate performance and affectivity of software and overall work load computation. As such optimal planning and Reliability Modeling for Software system is active research area. Proposed System Analysis and model defects from previous Software Versions eliminating them. Modeled on this Concept a mathematical Analyzing and Modeling framework for multiple release of Software product is Designed and Developed. Research work achieves reliability with 2-dimension work flow applying testing for deployed product and Cobb- Douglas factor. Testing cannot be done continuously and research seeks to minimize testing, value minimization reliability maximization are outcomes of research. Optimal design is complicated non-linear equation to solve. Proposed system Deploys GA (genetic algorithm) to achieve Reliability as of survey outcomes presents GA to be best for Search and optimization, which is implemented interactively on real binary solution of population, with 4 genetic operations: crossover, mutation and reproduction with S shape Model at core. Functional dataset consisting test week, test time, resources and identified Faults. Optimal model solves and determines when to stop testing and predict with time newer version would be released, with removal of dead test cases and test coverage evaluation as added workings.

Keywords:-SRGM, Software reliability, Software Engineering, Optimal Release.

1. INTRODUCTION

A Product based Software is Enriched with Functionality over Time and Becomes Massive and Complex over version of Development[3].In decision making for Further Product Development "Reliability" is Major factor in quantifying determining as off when testing is to be halted and release Software System .To Achieve this Goals[5] Industry has come up with Software reliability Growth models i.e. SRGM's. This model compute metrics on software like: No of Remaining Faults, failure rates and reliability level of software. NHPP framework in SRGM's Class identifies complete and incomplete Debugging works [3].future Added factors like mean time and until next failure have also been identified.

Software fault debugging is very complex process, but when test maker find them they usually propose alternative modification deviating from requirement giving a new set of work to developer. This updated code has to go through all Test cases again and is Time and cost consuming process.

Developing Software's with less dependency helps to achieve software with economical and effective test case plan.

2. APPROACHES

Approaches Taken in SRGM development are basically two fold.

1. Mathematical dependableness prediction.
2. Mathematical correlations of actual defect detection

First Approaches is based on better system development with effective plan for test cases and harness services to a certain level achieving depend less development.

Whereas second approach accounts in software code characteristics like line of code, nested loops, references, input-output call. Principle working second approach is based on test case results for identification of errors.

A alternative classification model called as predictive model one of which is NHPP .these model identify error seeds, failure rates, non homogenized Poisson method .the NHPP is based on S form of mean calculation

2.1 Different SRGM models

1) Jelinski-Moranda Model 2) Goel-Okumoto Model.3) Generalized Goel NHPP prototype 4) Inflected S-Shaped Model 5) Logistic Growth Curve Model 6) Musa okumaoto Model

The SGRGM models can be elaborated as detailed as:

[1]Jelinski-Morandaprototype:

prototype was first presented as reliability growth prototype 1972 by jelinski moranda.It is a continuous time-independently dispersed inter breakdown times and self-governing and identical error conduct prototype. The software failure speed of hazard method at any time is directly calculative to current fault content of the source code. The distribution of statistics is Exponential division.

[2]Goel-Okumoto prototype:

Prototype most accepted NHPP model in area of software reliability modeling. also termed exponential NHPP model. jelinski assumptions are Incorporated in this model .

[3] Generalized Goel NHPP prototype:

In sort to describe circumstances that software breakdown intensity increase somewhat at start and then begin to reduce, Goel proposed a easy overview of Goel-Okumoto model with an extra factor c.

[4]Inflected S-Shaped prototype:

Resolve technical limitations of okumoto prototype and ohaba introduced it .Observations of model suggest that faults when evaluated graphically take S-shape in curve .Analyzing this curve would help reduce errors.

Model Name	Mean Value Function	Intensity Function
1. Jelinski-Moranda model	$m(t) = n_0[1 - \exp(-\phi t)]$	$\lambda i = (N-k)\mu$
2. Goel-Okumoto Model	$m(t) = a(1 - exp[-bt]), a > 0, b > 0$	$\lambda(t) = ab * exp[-bt], a > 0, b > 0$
3. Generalized Goel NHPP Model	$m(t) = a(1 - exp[-bt^c]),$ $a > 0, b > 0, c > 0$	$\lambda(t) = abct^{c-1}\, exp[-bt^c],$ $a > 0, b > 0, c > 0$
4. Inflected S-Shaped Model	$m(t) = a * \frac{1 - exp[-bt]}{1 + \psi(r) * exp[-bt]},$ $\psi(r) = \frac{1-r}{r}, \quad a > 0, b > 0, r > 0$	$\lambda(t) = \frac{abexp[-bt](1+\beta t)}{(1+\beta * exp[-bt])^2},$ $a > 0, b > 0, \beta > 0$
5. Logistic Growth Curve Model	$m(t) = \frac{a}{1 + k * exp[-bt]},$ $a > 0, b > 0, k > 0$	$\lambda(t) = \frac{abexp[-bt]}{(1+k * exp[-bt])^2},$ $a > 0, b > 0, k > 0$
6.Musa-Okumoto Model	$m(t) = a * ln(1 + bt), \quad a > 0, b > 0$	$\lambda(t) = \frac{ab}{(1+bt)}, \quad a > 0, b > 0$

Fig 1: SRGM Models [17]

3.TABULATED LITERATURE SURVEY

Year	Author	Abstract	Methodology	Merits Demerits	Scope
1997	Wood	SRGM's are differentiated with Assumptions made at testing fault removal. This work evaluates these models with Tandem's software development. Defects are increased with new code lines and many times defect identification efficiency is assumed. Research work presents suggestions on to quantify model inaccuracy and find accuracy Vs. complexity factors.	An evaluator research work. Tandem software development and test environment are been presented. . **[Assumption Reliability and effect of model]** are been presented.	Research outcomes like simple models are good .realistic assumptions are better are work merits. Only Tandem environment is considered demerit	Compute degradation in accuracy due to SRGM assumption violations

2000	Gokhale1	NHPP model present either monotonic decreasing or increasing of failure rates but fail to consider process underneath of dataset. Work presents a log logistic SRGM that identifies increased or decreased occurrence of failure per faults. Analysis on two dataset are presented .Outcomes present equation on Finite failure NHPP model with log logistic model.	Finite Failure NHPP Model 1.finite failure 2. infinite failure Models: 1. GoelOkumoto (GO) model. 2. Generalized Goel Okumoto (GO) model. 3. S shaped model. 4. Log-logistic reliability growth model.	Failure occurrence rate per fault increase or decreases. Log based is bias model. The Model is been tested for only for two dataset. Methodology recognizes increasingdecreasing nature of failure occurrence rate per fault.	Determining appropriate finite or infinite model is challenging and combined or dynamic model can be better approach.
2002	Huang	Logistic testing work in SRGM is been presented. Rayleigh curve is been drawn for efforts in software testing. presented work shows logistic testing-effort method could be uttered as software-development test-work curve and which gives a good analytical Values based on real failure-data. Comparative analysis show model is better.Cost effective parameter is outcome of research.	Logistic based NHPP model is been deployed. Dataset of Rome Air Development Center (RADC) is been used.	Equation set used is complex. LSE minimizes deviation and found better for medium sample data.	The system can be simplified with Simple equations.
2003	Zhang	Research presents Methodology to incorporate fault exclusion efficiency into software reliability Evaluation.Imperfect debugging is measured in sense that new burdens could be introduced into software at debugging and detected errors might be detached completely. Model provides better decision.	Akaike's information criteria. Maximum-likelihood estimate (MLE). nonhomogeneous poison Process (NHPP).Imperfect debugging is measured in sense that not very fault could will be removed completely and new faults could be presented while removing present ones	Software reliability models and the Mean value functions is been presented. Fault TBF and Cumulative TBF are been calculated.	More data collection in better way can facility better Development.
2005	Zhao	Work presents SRGM on NHPP. Only certain research work consider	fault detection rate FDR isfrequently used to amount	TOESRGM is assessed using only a set of software	Work is tested on three factors: Fault

		test environment and operational environment. novel NHPP model presented take in features experimented from real failure data as method of testing time	effectiveness of fault recognition bytest Methodologies and test cases. A bell-outlined FDR function is presented which add both environmental features and Inherent FDR per fault. TOE-SRGM is built to calculate environment factors.	failure data which bring limitations to work.	content, FDR and environmental factor this can be added with more software metrics
2006	Huang	Article presents cost for optimal release time based on testing and efficiency of modeling reliability. Major two issues reliability molding & software reliability economics: test efforts efficiency is been considered. Software cost model is outcome. The policy outcomes would help manager to decide when and where to stop test.	Methodology first takes in ***logistic* testing-effort to find Testing patterns.** 1. Basic SRGM is grounded on assumptions. 2. A New Generalized Logistic TEF. 3. Comparison Criteria.	Optimal Software release policy is outcome help definitely enhances research.	Analyses would significantly aid software staffs in selecting best software economic rule based on cost testing effort, and test efficiency. **Optimal Policy is best and scope**
2006	Teng	A novel effort to predict reliability of software in random filed environment. System is been based on random environmental effects **(**RFE) software reliability Which covers both testing phase and operating phase in SDLC	1. *Generalized NHPP Software Reliability Model.* 2. Generalized random field environmen t. Alpha-RFE model, and Beta-RFE model	What application the system is been tested needs to examined as its results may vary in accordance to application.	cost prototypes and optimum software release Strategies under RFE.
2007	Stiebe r	Paper presents complete hierarchy of SRGM's which are parameter dependent. Recognizing failure rate and overall defects	A mathematical process of SRGM is been presented by article.	Optimistic to pessimistic model are been presented .the feedback control mechanism is found to be novel in work.	Plot of evaluation can be generated in better way with Extension to N0 value. **Feedback control** can be scope.

2008	Alhazmi	Loop holes have been found in windows and Linux systems .In such systems a Vulnerability model is been required. This would help in patch development and fault removal efficiency. Six models have been tested and found to be determine certain	1.Alhazmi-Malaiya Logistic Model (AML) 2. Rescorla Quadratic Model (RQ) 3.Rescorla Exponential Model (RE):	evaluation found AML model is usually best for extended term, performing well for schemes that as Windows 95 Linux as Because it identifies "s"-shape pattern data	Vulnerability model are required and have scope in large organization and .longer use products
2016	Shrivastava	Testing work does not give faults to be removed .as such better reliability can be achieved with combined approach. A two dimensional framework is presented with hazard rate method for evolving multi release models	Two Dimensional Modeling Gamma Erlang method. Mean square error	Mean value function for six SRGM approaches is presented which is major merit. And numerical examples justify the research	phenomenon of change-fact and incomplete debugging in proposed model would make it more general

3.1 Problem statement

Design and **D**evelop an optimal software release and reliability modeling system for multi-release software. The system is built on KIS principle Keep it simple and an iterative model is been used for developing it.

4. PROPOSED SYSTEM

a. Design of system:

The System consists of Three phase model and Object oriented architecture where every module is attached on input from previous module.

1. Dataset Module: specify dataset and release dataset is main function of module.

2. Two Dimensional Framework: A basic 2 Dimensional mathematical model (SRGM) based on NHPP to solve optimal release problem

3. Analysis Module: A Module that evaluates system performance on dataset and finds testing condition no of present errors and what plan of testing is to be selected A decision support module.

4.2 Design of Algorithm:

A generic algorithm is been proposed as to optimize the Process of Optimal Software release and Work to AchieveSoftware reliability. The System is been applied to 4 release Of Software system. The System suffers from downside as Testing and Fault Identification need to be done at in faster Dynamic way.

Why GA?

A**genetic algorithm** (**GA**) is a search Method that mimics the process of Selection in natural way.it optimizes solution and reduces search time. It a class of Evolutionary Algorithm (EA) with mutation inheritance and selection as three methods.

Genetic algorithms are simple to implement, but their behavior is difficult to understand. In particular it is difficult to understand why these algorithms frequently succeed at generating solutions of high fitness when applied to practical problems.

The building block hypothesis (BBH) consists of:

[1.] A description of a heuristic that performs adaptation by identifying and recombining "building blocks", i.e. low order, low defining-length schemata with above average fitness.

[2.] A hypothesis that a genetic algorithm performs adaptation by implicitly and efficiently implementing this heuristic.

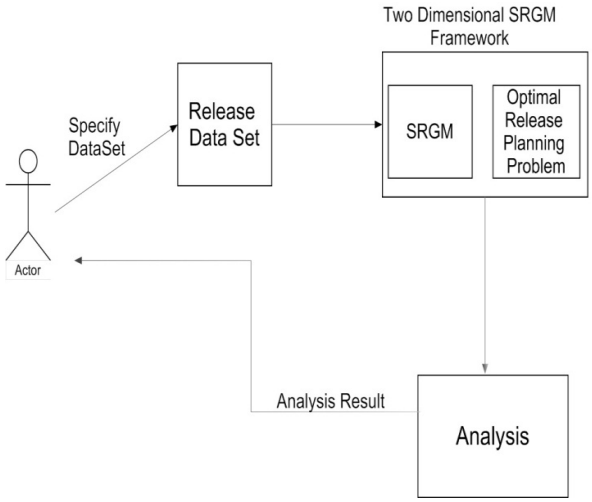

Fig2: Proposed System

❖ **Genetic Algorithm**

A classical genetic algorithm require:

[1].A genetic presentation of solution domain.

[2] .A fitness method to evaluate solution domain.

❖ **Steps in GA Algorithm**

[1] Initialization
[2] Selection
[3]Termination condition

Research work: Proposed Algorithm and Mathematical Model

Parameters are
N or a =====> Initial No of Faults before testing specific release
b ====> No of faults removed after testing from specific release(Fault detection rate per remaining fault)
α =====> Resource elasticity to testing time
β ====> Logistic learning factor
Estimated parameters a, b, α, β will be same with minor differences because we are using same dataset.

Mathematical Model:

S-Shaped Model:

Cobb- Douglas production equation:

$\tau = s^{\alpha}u^{1-\alpha}$

Where,

S= testing time
α= effect of testing time/resource elasticity to testing time.
τ= Two-dimensional variable which represents s and u.

Release 1:

$$m_1(\tau) = \frac{N_1(1-exp(-b_1 s^{\alpha}u^{1-\alpha}))}{1+\beta\, exp\,(-b_1 s^{\alpha}u^{1-\alpha})}$$

Where,

= Expected number of faults removed in time in
b = fault detection rate per remaining faults
β = Logistic learning factor
N = initial number if faults for i^{th} release.
$\tau = s^{\alpha}u^{1-\alpha}$ for first release.

Release 2:

$$m_2(\tau) = (N_2 + N_1(1 - F_1(\tau_1)) * F_2(\tau - \tau_1)\tau_1 \le \tau \le \tau_2$$

Release 3 and Subsequent:

$$m_n(\tau) = (N_n + N_{n-1}(\tau_n - 1)) * F_n(\tau - \tau_{n-1})\tau_{n-1} \le \tau \le \tau_2$$

Testing resources fault detection no of test cases and test coverage

Table 1: Release 1

Test Week	Testing resource	Faults Detected	No.test cases	Test Coverage
1	**519**	**16**	**1038**	**53.62**
10	**5823**	**75**	**11646**	**89.05**
20	**10000**	**100**	**20000**	**100**

Table 2: GA performance

Value	No.of Generations	Cross probability	Mutation probability
100	**25**	**0.9**	**0.1**

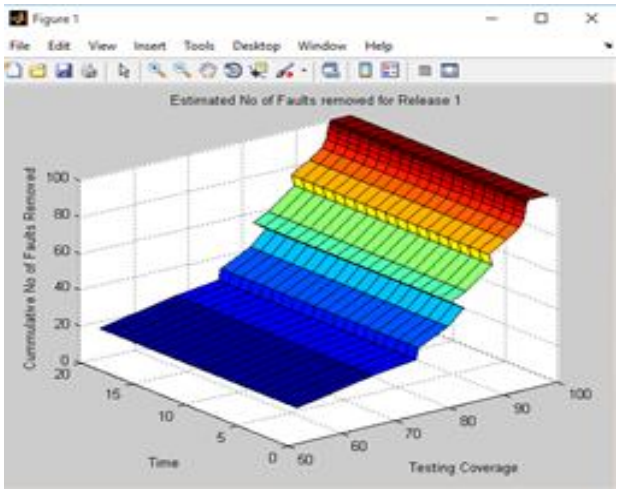

Fig2: Performance of release1

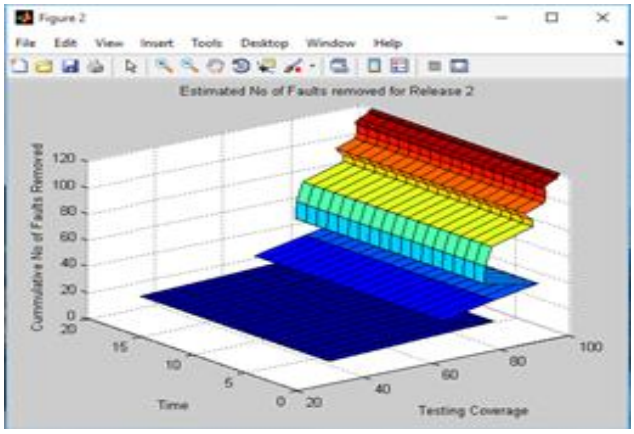

Fig3: Performance of release 2

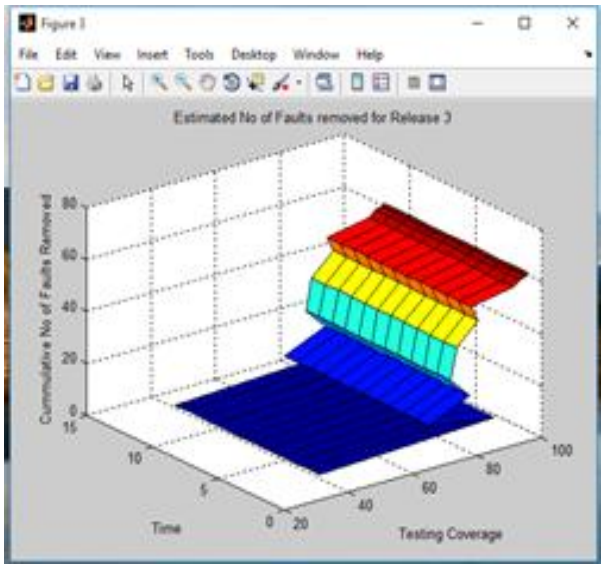

Fig4: Performance of release 3

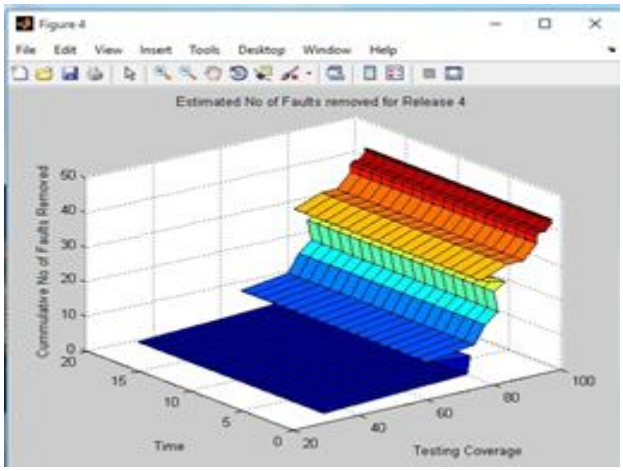

Fig5: Performance of release 4

Observing above 2 Tables the GA is found to be best and all four release set are been evaluated graphically.

6. CONCLUSION AND FUTURE WORK

Proposed methodology assist to detect effect of errors generated in software using multi-release software reliability modeling framework. Model exclusively takes into justification errors of that release that are under stage of testing and faults absent in preceding release ((operational phase) System assiststo achieve scope by spreading the statement like time and funds simultaneously. The expressed planning problem assists in determining both best release period and optimal resource intake simultaneously. Numerical evaluation justify proposed work.

References

[1] Alan Wood "Software Reliability Growth Models: Assumption vs. Reality"1071-9458/97 $10.00 0 1997 IEE.R.

[2] Swapna S. Gokhale1and Kishor S. Trivedi "Log-Logistic Software Reliability Growth Model "Center for Advanced Computing and Communication, by the National Science Foundation grant number EEC-9714965 and Charles Stark Draper Laboratory Grant # DL-H-505333.

[3] Chin-Yu Huang and Sy-Yen Kuo , "Analysis of Incorporating Logistic Testing-Effort Function Into Software Reliability Modeling", IEEE transactions on reliability, vol. 51, no. 3, september 2002

[4] Xuemei Zhang, XiaolinTeng, and Hoang Pham "Considering Fault Removal Efficiency in Software Reliability Assessment" IEEE transactions on systems, man, and cybernetics—part a: systems and humans, vol. 33, no. 1, January 2003

[5] Jing Zhao Hong-Wei Liu Gang Cui Xiao-Zong Yang, "A Software Reliability Growth Model from Testing to Operation "Proceedings of the 21st IEEE International Conference on Software Maintenance (ICSM'05) 1063-6773/05 $20.00 © 2005 IEEE.

[6] Chin-Yu Huang, *Member, IEEE,* and Michael R. Lyu ,"Optimal Release Time for Software Systems Considering Cost, Testing-Effort, and Test Efficiency" IEEE transactions on reliability, vol. 54, no. 4, december 2005

[7] XiaolinTeng and Hoang Pham, "A New Methodology for Predicting Software Reliability in the Random Field Environments"IEEE transactions on reliability, vol. 55, no. 3, september 2006.

[8] Harald A. Stieber, "A Family of Software Reliability Growth Models" 31st Annual International Computer Software and Applications Conference(COMPSAC 2007) 0-7695-2870-8/07 $25.00 © 2007.

[9] Omar H. Alhazmi and Yashwant K. Malaiya, "Application of Vulnerability Discovery Models to Major Operating Systems" IEEE transactions on reliability, vol. 57, no. 1, march 2008.

[10] W. T. Tsai, D. Zhang, Y. Chen, H. Huang, R. Paul*, N. Liao" A Software reliability growth models for web services"MSR-TR-2002-108, December 2002.

[11] P. K. Kapur, H. Pham, Fellow, IEEE, Anu G. Aggarwal, and GurjeetKaur "Two Dimensional Multi-Release Software Reliability Modeling and Optimal Release Planning"IEEE Transactions on reliability, VOL. 61, NO. 3, September 2012.

[12] N.Sasikaladevi1 Dr.L.Arockiam2 , "Relability Evaluation model for Composite Web Services"International Journal of Web Services April 2010.

[13] Sunil Kumar Khatri, Deepak Kumar, AsitDwivedi, NitikaMrinal "Software Reliability Growth Model with Testing Effort using Learning Function" IEEE Transactions on Software Engineering SE-11, pp. 1411-1423, 2005.
[14] AbhayaAsthana and Kazu Okumoto "Integrative Software Design for Reliability Beyond Models and Defect Prediction"Bell Labs Technical Journal 17(3), 37–60 (2012) © 2012 Alcatel-Lucent. • DOI: 10.1002/bltj.21557.
[15] Kaigui Wu, *2* Yong Wan "Software Reliability Prediction from Test Data"Journal of Convergence Information Technology(JCIT) Volume8, Number5,Mar 2013 doi:10.4156/jcit.vol8.issue5.149
[16] A.k.shrivastava,ruchisharma"generalized modeling for multiple release of two dimensional software reliability growth model"international journal of innovative research & development january, vol 5 issue 22016
[17] Gaurav Agarval Dr.V.K.Gupta "Software Relability Growth Model" International Journal of Advanced Research in Computer Science and Software Engineering volume 4 Issue 1 Jan 2014

AUTHORS

Scholar Tanjua is currently pursuing M.Tech (Computer) from Department of Computer Engineering, Bharati Vidyapeeth Deemed University College of engineering Pune, India. She received her B.Tech (Computer) Degree from Bharati Vidyapeeth Deemed University College of engineering Pune, India. Her area of interest include Software engineering

Prof. Amol Kadam is working as a Professor in computer engineering department at Bharati Vidyapeeth University College of engineering, Pune, Maharashtra, India. He received his ME (Computer) degree from Bharati Vidyapeeth University College of engineering, Pune. His research interests include Software Engineering.

Prof. Shashank Joshi: Innovative teacher devoted to Education and Learning for the last 25 yrs Research interests include software engineering, Software Engineering, Object Engineering, Secure Systems, Visual and Generic Modeling, Information Technology Enabled Services,Database Management Systems, Knowledge Engineering and Information Systems Management. Presently he is engaged in SDLC and secure software development methodologies.114 PG students and 22 Ph Dstudents are been imparted guidance. International Journal:102 International Conference:62 and National Conference:91 remark his research work in whole.