I.J. Intelligent Systems and Applications, **2017, 5, 43-54**
Published Online May 2017 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2017.05.06

An Empirical Perspective of Roundtrip Engineering for the Development of Secure Web Application Using UML 2.0

Nitish Pathak
Research Scholar and Corresponding Author, Uttrakhand Technical University (UTU), Dehradun, India
E-mail: nitishforyou@gmail.com

B. M. Singh and Girish Sharma
Department of CS&E, College of Engineering Roorkee, UTU, Dehradun India
Departments of CS&E, BPIBS, GGSIP University, Government of NCT of Delhi, India
E-mail: bmsingh1981@gmail.com, gkps123@gmail.com

***Abstract*—**This research paper propose experimental support to secure Round Trip Engineering and use of security performance flexibility trusted operating systems for the designing of secure web applications. In this research paper, for security concern, we suggest use of trusted operating systems as a platform to run these web applications. In this regard, a number of trusted operating systems like Argus, Trusted Solaris, and Virtual Vault have been developed by various companies to handle the increasing need of security.

For improving the performance of same web applications, we observe that all security checks in a Trusted Operating System are not necessary. As per our suggestion, various unnecessary security checks can be skipped by administrator, so that system performance of these web applications can improve. These unnecessary security checks, system calls and operations can be easily identified at the time of requirement elicitation and Requirement Engineering. For example, as we know, the popular web servers deal with public information. In this web application, the need for security checks during reads from disk seems like a waste of CPU cycles. On the other hand the real security need for servers seems to be of the write accesses.

This research paper aims to support the efficiency of object-oriented class-based programming and object oriented modeling in secure software development.

***Index Terms*—**Component-based development, secure round trip engineering, Model driven Class identification and Modeling, Trusted Operating Systems, Re-engineering.

I. INTRODUCTION

The Unified Modeling Language is an extensible language for software design such as web applications, database applications, business modeling, data warehouses, legacy systems and so on [1].In the perspective of web application development, various object oriented model-driven software development processes have come into view [2]. Object oriented design uses objects in the programming language which helps in modeling the patterns of any system. It manages the software complexity and also enhances usability and maintainability of software. By the help of the suitable models, organizations are able to understand the structure and overall functionality of any system, by identifying flaws in the system and applying required modifications at an early stage of its development [3].

In Object Oriented analysis and designing, we know, Objects and classes are central part for object models. UML is gradually more used for capturing conceptual object oriented models of software, as it supports conceptual modeling of real life domains. In this research paper, we codify a set of invariant blueprints formalized here for confining a well-to-do class of constraints on class diagrams [4][5]. In this paper, we are concerned with the capture of abstract entities (or classes), the associations and relationship existing between them and adjacent ones, as represented in one or more object oriented diagrams [6].

With the help of forward engineering, we can identify the degree of coupling and the degree of cohesion from the source code [7]. Classes in object-oriented systems for any web applications, written in diverse object oriented programming languages, have identifiers, member functions, friend functions, friend classes, nested classes and nested functions, which replicate concepts from the domain of the web application [8].

This research article presents an object oriented designing of a complex and large Web application, where huge amounts of data and information are accessed by no. of users. In model-driven Web engineering methodologies, the Web application can be particular through a set of conceptual models dealing with the special aspects of the web application and the ultimate C++ source code can be generated from such conceptual class models [9][10]. In this paper, we are focusing on a case study of Store Stock Control system. In this process, we are using security performance flexibility trusted

operating systems for maintaining the security of Store Stock Control system and UML 2.0 for the development and design of same complex web application [11].

Far above the ground, quality of software system design is necessary for the extended tenure success of software [12]. This research work not only enrich UML profiles but also represents a proposed system design based on object oriented methodology of system design, which is also well adopted and industry leading paradigm for designing and development of large, complex software[13].

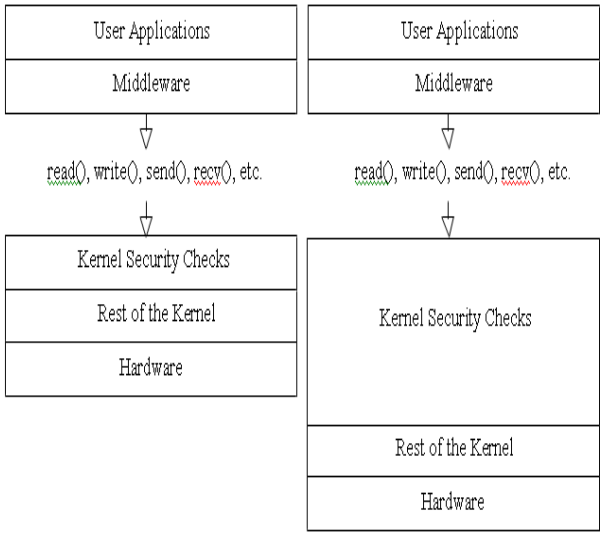
We know the significance of security in web applications; we advise trusted operating systems as platform for these web applications. In reality, trusted operating systems are improved choice for web applications to maintain the security concern for the same, but this security will come at a cost. For maintaining the more security for web applications, we have to implement so much security checks in trusted operating systems. With this approach, our web application will be more and more secure, but due to more security checks, the performance of the same system will disgrace in all respect. [14] [15].

This paper is structured as follows: Introduction, beginning of software system analysis, design and development, Security performance flexibility model and its importance is discussed in section 1. Section 2 reports on the forward engineering and SPF based secure Web applications. Sections 3 focus on Problem Description and Object Oriented Solution methodology. Section 4 discuss about the details of Implementation of Object Oriented Forward Engineering methodology. Section 5 comprises the transition process of forward engineering for secure web application. Further, section 6 focus on the Transition Process of Forward and Reverse Engineering that is also known as round trip engineering. The last section 7 comprises the conclusions, future research directions and references.

II. Forward Engineering and SPF Based Secure Web Applications

Web application's safety and security is a significant difficulty in today's Internet world. Because of increasing software's ability to trace and collect a large amounts of data and performing mining operations to obtain useful knowledge or information, which is helpful in determining future business policies and useful patterns[16].

As we know, for maintaining the more security, the trusted operating systems are better choice for the implementation of web applications [17]. For safety and security, we focus on trusted operating systems as operating system for secure web applications. These trusted operating systems may be more preferable choice for maintaining the high security in web projects. The blueprint of Trusted Operating Systems for web application is verified in the Figure 1. In Trusted operating systems for web applications, the better quantity of security is placed into the operating system kernel. Figure 1.b, shows this security and safety with a significantly thicker layer of kernel security checks [18][19]. During system programming, the source code structure of kernel security check layer of a Trusted Operating System depends on the requirement elicitation, design & development of a particular web application.



(a)Ordinary Operating Systems (b) Structure of proposed trusted Operating Systems for web applications

Fig.1. Structure of Trusted Operating Systems and Ordinary Operating Systems

In figure 1.b, the kernel safety checks are much more in trusted operating systems than standard operating systems. Due to more security checks, these operating systems are much more secure. This thick layer of security and safety causes Trusted Operating Systems to bear performance issue in all web applications [20]. All system calls to the Operating system kernel, must go throughout this bulky layer of security checks before they can perform any valuable and important work [21][22]. If we implement the trusted operating systems to web application than these web applications may suffer from performance issue due to much more security checks in trusted operating systems.

This system performance will depend upon the attributes that we are selecting during development from mining in various web applications [23][24].

III. Problem Description and Object Oriented Solution Methodology

This paper explains an object oriented Model Driven approach, secure execution environment and structure to Web application designing. As we know, the security of web application is also a major concern for web applications, this paper also focus on Security Performance Flexibility (SPF) based security for web applications [25]. Figure 1.b demonstrates the additional safety and various security checks in the kernel will cause trusted operating Systems to be slower than standard operating systems. If we relate the same trusted operating system for web applications, then we get very

lower performance and lower efficiency in all respect, but more and more security features for web applications, the main difficulty is to balance security parameters and performance of this operating system for desired web applications.

As for as computing workload is concern, we suggest that only a little amount of parts of the operating system security are actually necessary. According to figure 2, SPF allows computer administrators to skip some unnecessary parts of the security for web applications. During the development of secure operating systems, Designer and Developer can use the SPF structure to balance the security and performance needs in their particular web application. Whenever we prevent some useless and unnecessary parts of the Trusted Operating System security, performance, quality and efficiency of the web application can potentially be increased in all respect.

For such designing and development, first of all, we have to recognize which part can be disabled to get the highest performance in specific web application. We can achieve this through requirement elicitation and engineering. We should recognize a variety of operations in Store Stock Control web application to disable the unwanted system calls and operations to improve the performance. As we know all security constraints are not significant in all web applications. Selection of these security checks will differ from one web application to another web application.

The architectural consideration behind the SPF configuration is demonstrated in Figure 2. After object oriented modeling, we can develop the blueprint for secure web server. After this, we can implement and run on system SPF based trusted operating systems.

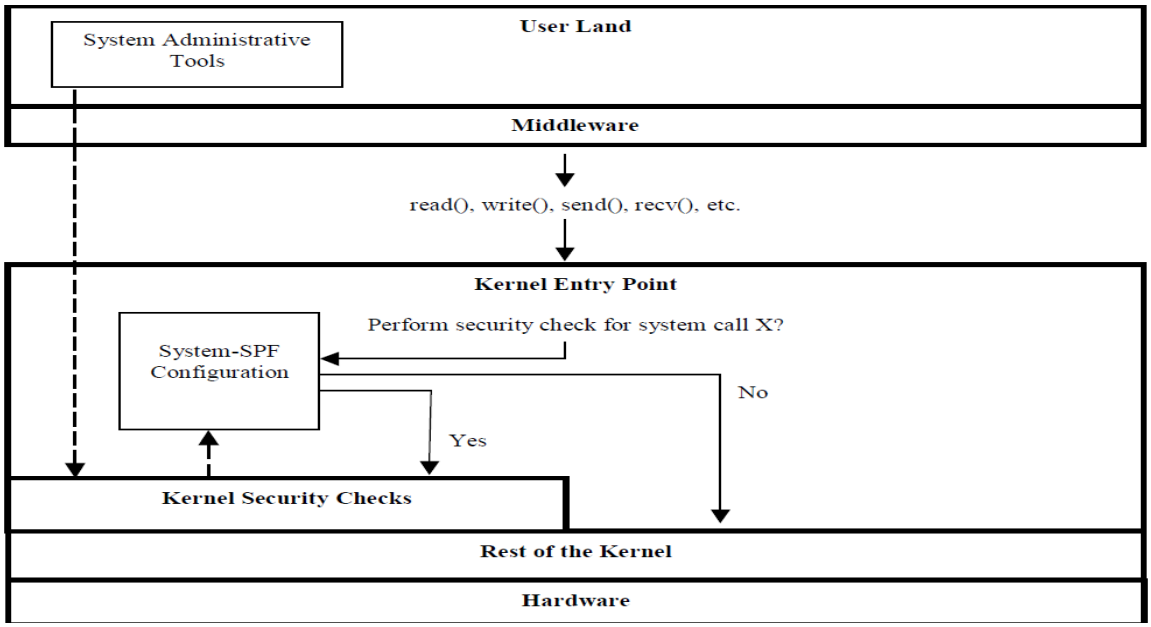


Fig.2. System-SPF structural design for Stock Control web application

This System SPF, we can also implement at process level for any web application. Process-SPF of Trusted Operating Systems offers the capability to disable and stop system calls and security checks in particular web applications or web processes. For instance, a computer system administrator may stop or block the read security checks for a MPEG video player. So, by skipping these read security checks on MPEG frame in MPEG video application, we may be able to recover the quality and clarity of the MPEG video.

In this paper, we consider Process based SPF. Process based SPF only disables and stops the read security checks on the MPEG video player. This selection of operations will vary from one web application to another web application. Figure 3. Shows the entire implementation of Process SPF for Stock Control web applications. This is also confirmed in Figure 3. The choice of System-SPF or Process-SPF and both will depend upon the requirement and complexity of web application. In this article, we are using the experimental software designing and development for Store Stock Control. UML 2.0 is for the same purpose. No doubt, that software design and development process and object oriented empirical modeling is very important for the development of e-commerce applications and IT applications.

Consider the subsequent requirements for Store Stock Control system: Store Stock Control system requires a new point of sale and stock control system for their many stores throughout the United Kingdom to replace their ageing mini store based systems. A sales assistant will be capable to process an order by entering product numbers (This product No. will be unique in stock control system) and required quantities into the system. The Store Stock Control system will display a description, price and available stock respectively. In-stock products will normally be collected immediately by the consumer from the Store Stock Control system but may be selected for delivery to the consumer's home address for which there will be a charge. If stock is not available the sales

assistant will be capable to create a backorder for the product from a regional warehouse. The products will then either be delivered direct from the regional warehouse to the consumer's home address or to the store for collection by the consumer.

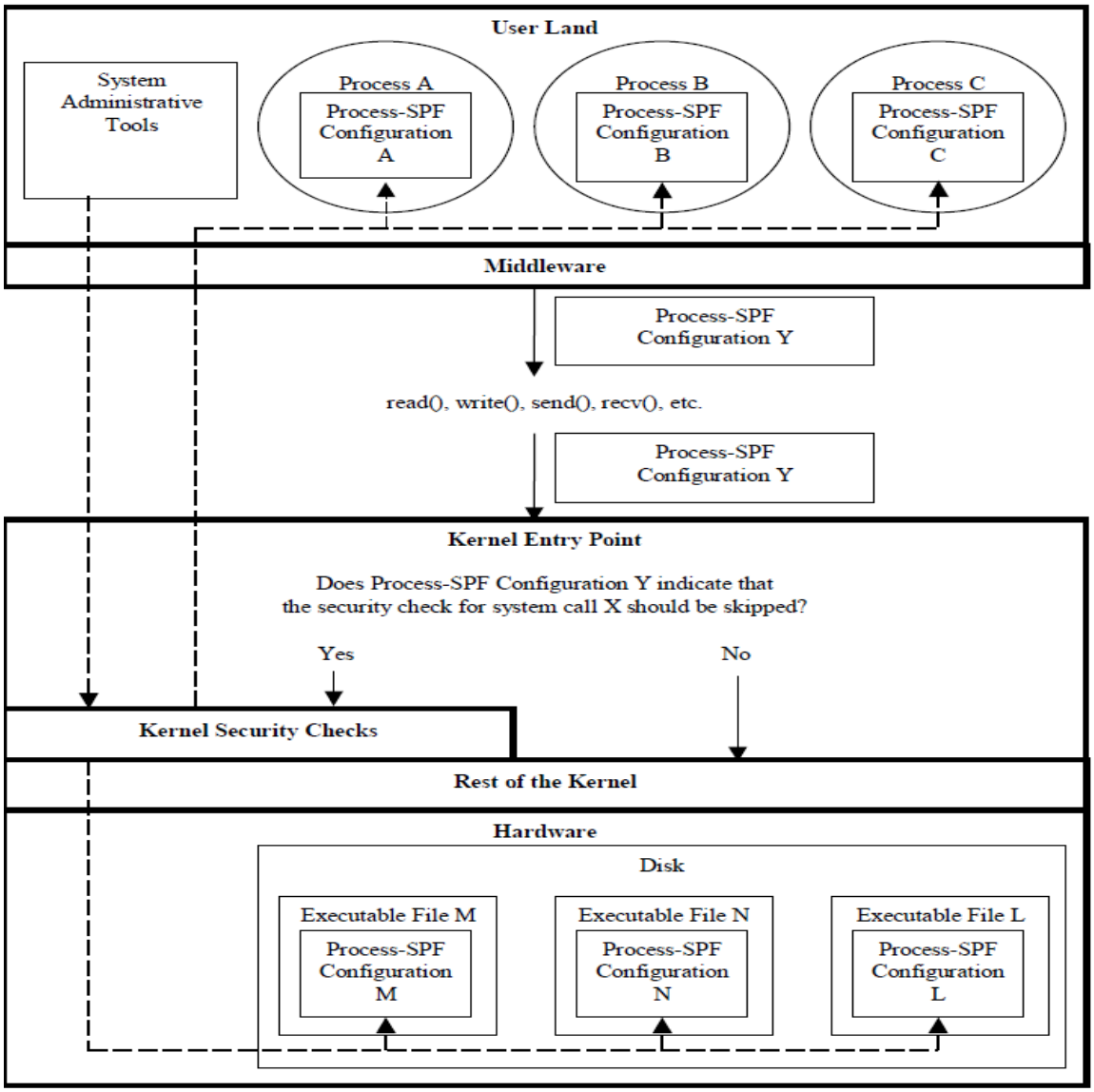


Fig.3. Process based SPF Framework of Trusted Operating Systems for Stock Control system web application

The projected approach is based on the following phases of the development process:

a) First of all, extract web application requirement and requirement discovery through requirement elicitation techniques and Requirement Engineering. This process will help in mining pattern for the system calls.
b) Design the use case diagram according to mining pattern of system calls, to represent the functional requirement of web application for the analysis phase of SDLC
c) Explain each use case in a textual manner i.e. use case description of major use cases
d) Specify process behavior using sequence diagrams
e) Designing of class diagram and implementation of Forward engineering in C++ source code using IBM Rational Rose for the same secure web application i.e. design phase
f) Concept of reverse engineering, re engineering and round trip engineering for the same secure web application.

IV. Implementation of Object Oriented Forward Engineering

The Use case diagrams have been drawn to emphasize all important scenarios and cases regarding the system use by these actors. As we know, 'Use Cases' are used to represent the functional requirements and services of the software system. So, we should design the use case diagram of above case study for round trip engineering.

In figure 4 we can see that store manager, sales assistant, customer, warehouse person and stock manager etc. are the actor. Rest of the diagram indicates the use cases (for example Display Stock Details, Payment Use Case, Order Delivery Use Case and Sales Summary Report Use Case etc.) i.e. functional requirement of Store Stock Control system and their relationships. The standard Use case diagram of Stock Control system is as follows in figure 4:–

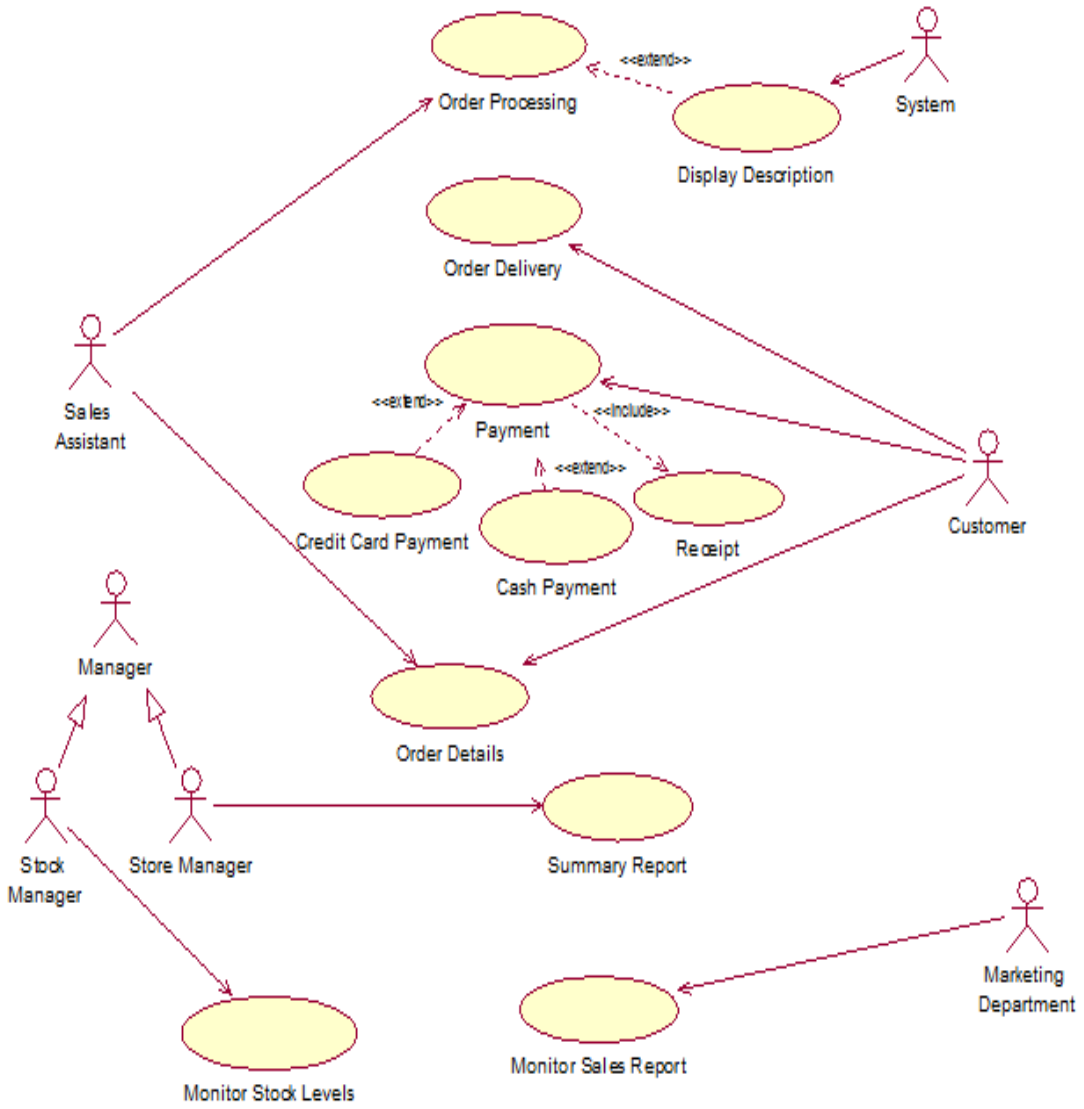


Fig.4. Generic Use case diagram for Store Stock Control based web Application

Table 1 to table 6 demonstrates the Use Case Description of Major Use Cases for above secure Web Application –

Table 1. Display Stock Details

Brief Description	The system displays info to Sales assistant
Actors	System
Flow of Events	1. Sales assistant enters the product numbers (Pro. No.) and required quantities into system. 2. System displays the description of the product.
Alternative Flow	If the stocks control system in not functioning, it will not start.
Precondition	Product numbers must be entered
Post condition	Information regarding product is displayed

Table 2. Order Delivery Use Case

Brief Description	This use case enables the customer to get the delivery of their order from the stock control system.
Actors	Customer
Flow of Events	1. Places order 2. Enter delivery details
Alternative Flow	If the processing doesn't get successful, generate error report to the sales assistant.
Precondition	The order should be successfully processed on the system
Post condition	If the use case was successful the order shall be delivered to the customer. If not, the system state is unchanged.

Table 3. Payment Use Case

Brief Description	Payment Use Case allows the customer to make payment for his order in the stock control system.
Actors	Customer
Flow of Events	1. Places order 2. Enter delivery details 3. Makes payment
Alternative Flow	If the processing doesn't get successful, generate error report to the customer and order is unsuccessful.
Precondition	The order should be successfully processed on the system
Post condition	If the Payment Use Case was successful the payment shall be made by the customer. If not, the order is incomplete.

Table 4. Credit Card payment use case

Brief Description	The credit/debit card is the mode of payment for the customer.
Actors	Customer, Sales Assistant
Flow of Events	1. The customer pays for the goods with credit card. 2. The card payment is verified using online transaction system.
Alternative Flow	If the processing doesn't get successful, generate error report to the customer and order is unsuccessful.
Precondition	The order should be successfully processed on the system
Post condition	If the Credit Card payment use case was successful the payment shall be made by the customer. If not, the order is incomplete.

Table 5. Make Refunds

Brief Description	The sales assistant makes the refunds to the customer by initiating this use case.
Actors	Sales Assistant, Customer
Flow of Events	1. The customer produces a valid receipt. 2. The refunds are made by the sales assistant. 3. The use case ends.
Alternative Flow	Invalid Receipt The customer doesn't produce a valid receipt. He is asked for a valid receipt. If he is able to produce a valid receipt, the basic flow step 'REFUND' is resumed. Otherwise the use case ends.
Precondition	Customer has made all the payments.
Post condition	Customer has got refunds.

Table 6. Sales Summary Report Use Case

Brief Description	This use case enables the store manager to view the summary report
Actors	Sales manager
Flow of Events	1. Logs into the system 2. View summary report
Alternative Flow	If the processing doesn't get successful, generate error report to the sales manager.
Precondition	The stock information along with sales details should be present on the system
Post condition	If the use case was successful the summary report is delivered to the sales manager. If not, the error is generated.

For novel modeling and full UML notation, we also can use some predefined stereotypes like <<include>> and <<extend >>can also be used in extended use case diagram, as we can see in figure 4. This solution can be complemented by designing a sequence diagram. The interaction or sequence diagrams are used when we desire to know the message flow and of the software systems. Figure 5; demonstrate the sequence diagram for Store Stock Control system. Nowadays visualizing interaction is a complex task. Thus the solution is to use special types of models to capture the diverse aspects of the interaction. The following are the usages of interaction diagrams:

- To confine dynamic performance of a software system. As we know, in object oriented language, object will communicate through message passing approach
- To express the communication flow (API, RMI and message flow etc.) in the software system.
- Message flows (synchronization and a synchronization etc) among the objects in web application.
- The sequence in which the messages are flowing within the web application.

During implementations, Web Developers can implement these Message flows, Object organization, structural association of the objects , ordering of member functions, forward engineering, reverse engineering and round trip engineering of software systems. The generic Sequence diagram of projected case study is as pursues in figure 5:–

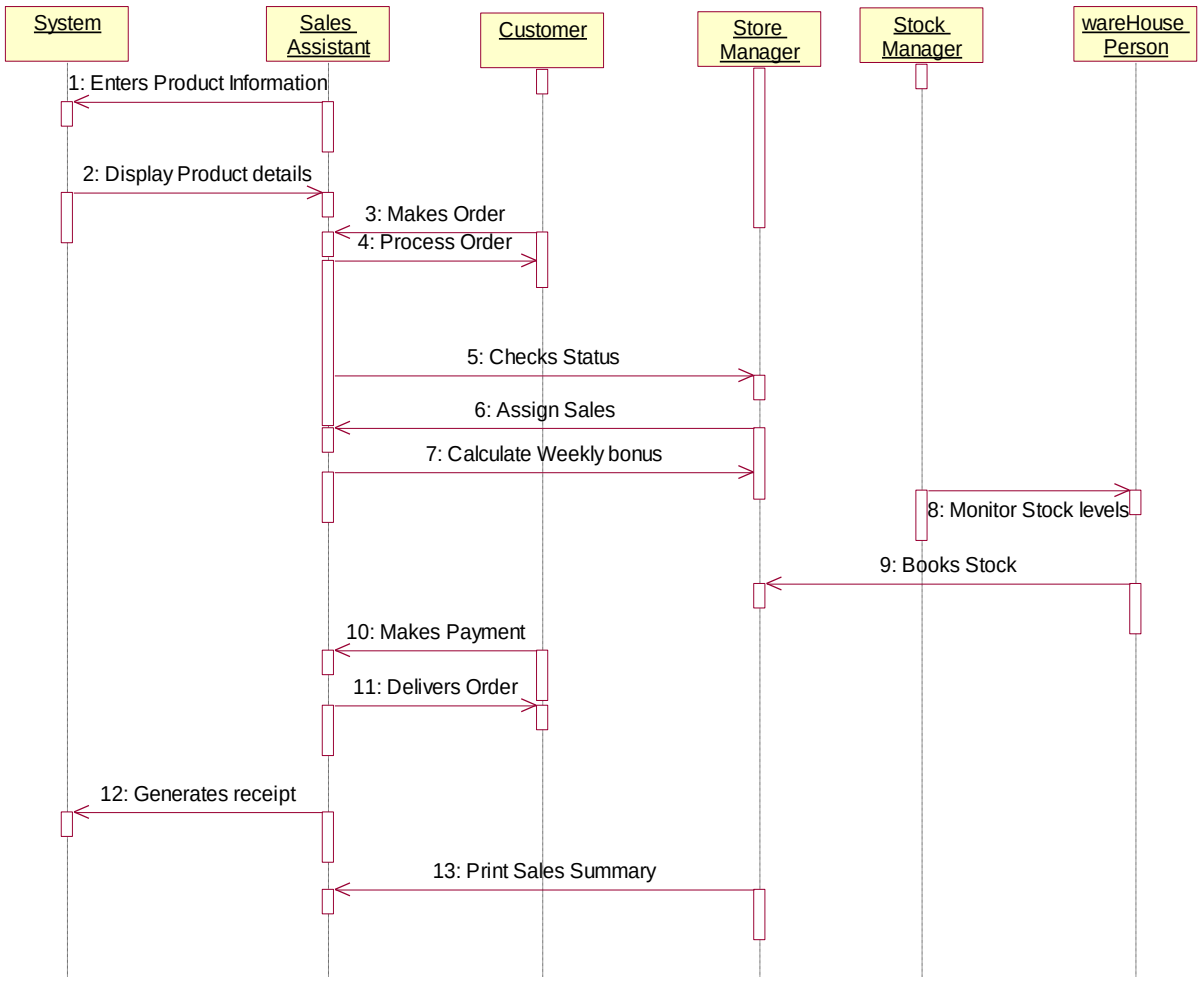


Fig.5. Sequence diagram for Store Stock Control based web Application

This paper focuses on empirical relationships between SPF model, the use case, sequence diagram and class diagrams. Web developers need to create a well-connected object oriented classes, where classes have high cohesion and low coupling between them.

By using above use case diagram, we develop the component based classes. After this we generate source code in C++ and execute forward engineering and reverse engineering for the same web application with the help of forward engineering tool. When we want to model the structure of a system or a web application, we can make use of class diagram. When we want to model the interaction among objects in runtime, with the sequence of method invocation, we can make use of sequence diagram.Class diagram obviously shows the mapping with object oriented languages like Java, C++ etc. So from realistic experience class diagram is normally used for construction purpose of software development process.

As source code is in text form, it is difficult and is hard for human to read or analyze, especially when the logic is difficult and involves a large number of classes. "A picture is worth a thousand words", by visualizing source code with diagram, we can simply realize the classes involve as well as their association in run time. so that we can gain a better understanding of C++ source code by reading diagram instead of looking to a possibly thousand lines of source code.

The standard class diagram of Store Stock Control web application is as follows in figure 6:

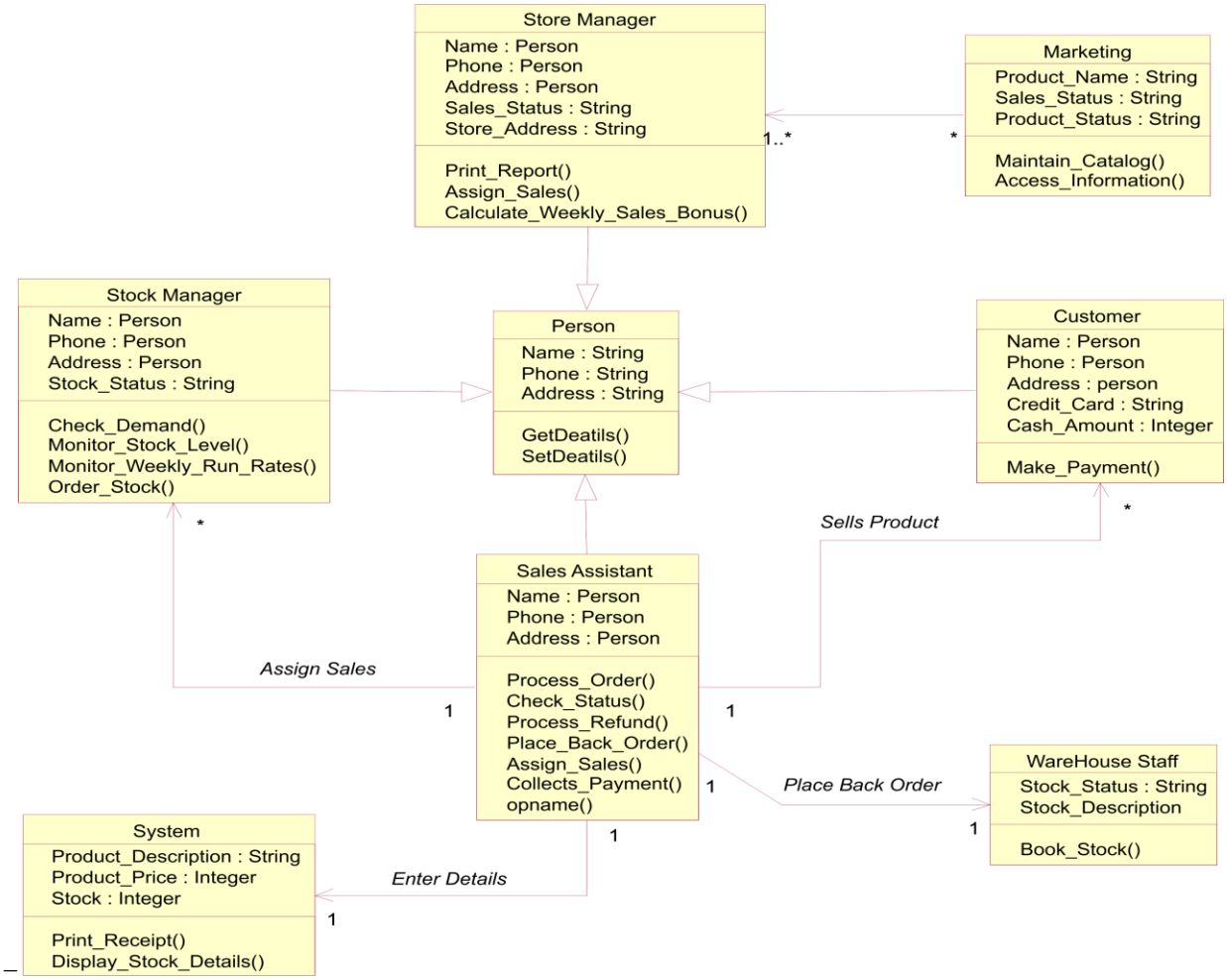


Fig.6. Generic Class Diagram for online Store Stock Control web application

Logical and technical Designers can build a high level object oriented class model, then pass to web developer to perform more lower-level software system modeling, and ultimately follow the forward engineering to generate source code in any object oriented language. With IBM rational Rose, we can simply generate source code in object oriented language from class diagram. This source code will work as blue print for software development team.

V. Implementation of Forward Engineering in C++ for Store Stock Control Based Web Application

In this Store Stock Control Based Web Application, storing objects may be sales clerk, inventory, Credit card, Cheque, store manager, Payment, Person, marketing, stock manager, person, warehouse person, invoice, system, customer etc. (See Fig. 6). The class wise equivalent C++ code of this case study is as follows-

Code:

```
#ifndef
CUSTOMER_H_HEADER_INCLUDED_AAC32C6F
#define
CUSTOMER_H_HEADER_INCLUDED_AAC32C6F
#include "Person.h"
//##ModelId=553C9B4C0295
class Customer : public Person
{
public:
//##ModelId=553C9B890326
Make_Payment();
private:
//##ModelId=553C9B5A003E
Person Name;
//##ModelId=553C9B5E0028
Person Phone;
//##ModelId=553C9B64019E
person Address;
//##ModelId=553C9B680189
String Credit_Card;
//##ModelId=553C9B6F0007
Integer Cash_Amount;
};
#endif /*
CUSTOMER_H_HEADER_INCLUDED_AAC32C6F
*/
#ifndef
MARKETING_H_HEADER_INCLUDED_AAC354F0
#define
MARKETING_H_HEADER_INCLUDED_AAC354F0
//##ModelId=553C983603A7
```

```
Class Marketing
{
public:
//##ModelId=553C987002CA
Maintain_Catalog();
//##ModelId=553C98770138
Access_Information();
private:
//##ModelId=553C984200B3
String Product_Name;
//##ModelId=553C984C02E5
String Sales_Status;
//##ModelId=553C985E02BD
String Product_Status;
};
#endif /*
MARKETING_H_HEADER_INCLUDED_AAC354F0
*/
#ifndef
PERSON_H_HEADER_INCLUDED_AAC30F36
#define
PERSON_H_HEADER_INCLUDED_AAC30F36
//##ModelId=553C970F00AC
class Person
{
public:
//##ModelId=553C976E01C4
GetDeatils();
//##ModelId=553C97750038
SetDeatils();
private:
//##ModelId=553C971B02D3
String Name;
//##ModelId=553C97380226
String Phone;
//##ModelId=553C975B0001
String Address;
};
#endif /*
PERSON_H_HEADER_INCLUDED_AAC30F36  */
#ifndef
SALES_ASSISTANT_H_HEADER_INCLUDED_AAC
352A
6
#define
SALES_ASSISTANT_H_HEADER_INCLUDED_AAC
352A
6
#include "Person.h"
//##ModelId=553C997601E5
class Sales Assistant : public Person
{
public:
//##ModelId=553C9A21007E
Process_Order();
//##ModelId=553C9A260121
Check_Status();
//##ModelId=553C9A300352
Process_Refund();
//##ModelId=553C9A3E007C
Place_Back_Order();
//##ModelId=553C9A4A0191
Assign_Sales();
//##ModelId=553C9A530207
Collects_Payment();
//##ModelId=553C9A70013A
opname();
private:
//##ModelId=553C99860376
Person Name;
//##ModelId=553C998A0350
Person Phone;
//##ModelId=553C998E035F
Person Address;
};
#endif /*
#include "Person.h"
//##ModelId=553C99010386
class Stock Manager : public Person
{
public:
//##ModelId=553C99340150
Check_Demand();
//##ModelId=553C99380262
Monitor_Stock_Level();
//##ModelId=553C9940011C
Monitor_Weekly_Run_Rates();
//##ModelId=553C994F005E
Order_Stock();
private:
//##ModelId=553C9912016D
Person Name;
//##ModelId=553C991601B4
Person Phone;
//##ModelId=553C991B0229
Person Address;
//##ModelId=553C9920005D
String Stock_Status;
};
#endif /*
#include "Person.h"
//##ModelId=553C97A103B6
class Store Manager: public Person
{
public:
//##ModelId=553C9806018D
Print_Report();
//##ModelId=553C980D0338
Assign_Sales();
//##ModelId=553C981A01B5
Calculate_Weekly_Sales_Bonus();
private:
//##ModelId=553C97B30348
Person Name;
//##ModelId=553C97BD029E
Person Phone;
//##ModelId=553C97C5007C
Person Address;
//##ModelId=553C97D101AE
String Sales_Status;
```

```
//##ModelId=553C97E402A7
String Store_Address;
};
#endif /*
STORE_MANAGER_H_HEADER_INCLUDED_AAC
332A6
*/
#endif /*
 SYSTEM_H_HEADER_INCLUDED_AAC3366C  */
#ifndef
#define
WAREHOUSE_STAFF_H_HEADER_INCLUDED_A
AC32A1B
//##ModelId=553C9C2202FC
class WareHouse Staff
{
public:
//##ModelId=553C9C4B0368
Book_Stock();
private:
//##ModelId=553C9C2B02DE
String Stock_Status;
//##ModelId=553C9C330009
Stock_Description;
};
#endif /*
#include "Marketing.h"
//##ModelId=553C987002CA
Marketing::Maintain_Catalog  ()
{
}
```

Through roundtrip engineering, we develop the understanding level of complex software. In the above experimental case, we have transformed the object oriented models into C++ source code. With the help of above software development process, developers can identify software Metrics like no. of data member per class, friend functions and friend classes, data structure metrics, coupling between objects, information flow etc.

VI. Transition Process of Forward and Reverse Engineering

As we know, Round-trip engineering is the capability to create object oriented UML model from source code and generate source code from these object oriented UML model, and keep them synchronized. For any web application, we can use round-trip engineering to remain our implementation model and source code up-to-date. Object oriented design information is improved from the source code and some obtainable design documentation. As we know, the software roundtrip engineering i.e. forward engineering and backward engineering (as we can notice in Fig.7) plays a vital role in software development life cycle.

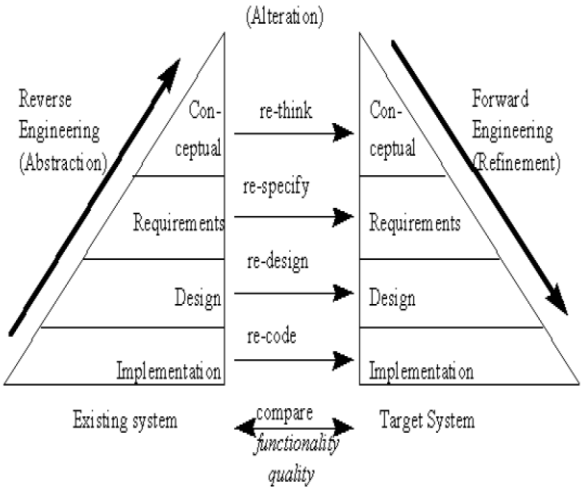


Fig.7. General Model for Software Round Trip engineering

In this paper, we have used round trip engineering implementation for the same stock control system web application. In Figure.7 indicates the implemented model of Software Round trip engineering process for complex software development. There are various business object oriented tools that provide the reverse engineering abilities. In this paper we have used IBM Rational Rose for the same purpose. UML tools mostly apply reverse engineering to execute round-trip engineering.

VII. Conclusion and Future Scope

This research paper creates experiential support to showcase the effectiveness and efficiency of object oriented modeling in software development, with primary focus of security through the process SPF and system level SPF based security in web applications. For maintaining the balance between security and performance, this paper covers the concept of security based forward engineering and round-trip engineering where round-trip engineering maintains the synchronization between your C++ source code and software design.

The mining restrictions imposed by proposed information securing system, includes complex processing steps, so it becomes necessary to visualize and design its model accordingly before its actual implementation. In this paper, we describe experiential round trip engineering, object oriented modularization, source code structuring and restructuring of secure software system.

Presently this research paper is related to static models, which are logically very close to the development and execution. In forthcoming study, this work can be extending for dynamic modeling for more and more refine and high qualitative software.

REFERENCES

[1] PEDRO VALDERAS and VICENTE PELECHANO "A Survey of Requirements Specification in Model-Driven Development of Web Applications" ACM Transactions on the Web, Vol. 5, No. 2, Article 10, Publication date: May 2011

[2] James P. Davis "Propositional Logic Constraint Patterns and Their Use in UML-Based Conceptual Modeling and Analysis" IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 19, NO. 3, MARCH 2007

[3] Andrian Marcus and Denys Poshyvanyk "Using the Conceptual Cohesion of Classes for Fault Prediction in Object-Oriented Systems" IEEE TRANSACTIONS ON SOFTWARE ENGINEERING,VOL.34,NO.2,MARCH/APRIL 2008

[4] BARBARA POBLETE, MYRA SPILIOPOULOU "Privacy-Preserving Query Log Mining for Business Confidentiality Protection" ACM Transactions on the Web, Vol. 4, No. 3, Article 10, Publication date: July 2010.

[5] SARA COMAI and DAVIDE MAZZA "A Model-Driven Methodology to the Content Layout Problem in Web Applications" ACM Transactions on the Web, Vol. 6, No. 3, Article 10, Publication date: September 2012.

[6] Thiago Jabur Bittar, Renata P. M. Fortes and Luanna Lopes Lobato "Web Communication and Interaction Modeling using Model-Driven Development" SIGDOC'09, October 5–7, 2009, Bloomington, Indiana, USA. Copyright 2009 ACM 978-1-60558-559-8/09/10

[7] Ibéria Medeiros, Nuno F. Neves and Miguel Correia "Automatic Detection and Correction of Web Application Vulnerabilities using Data Mining to Predict False Positives" WWW'14, April 7–11, 2014, Seoul, Korea. ACM 978-1-4503-2744-2/14/04.http://dx.doi.org/10.1145/2566486.2568024.

[8] Andrea De Lucia, Carmine Gravino, Rocco Oliveto and Genoveffa Tortora "An experimental comparison of ER and UML class diagrams for data modeling" Empir Software Eng (2010) 15:455–492 DOI 10.1007/s10664-009-9127-7, © Springer Science+Business Media, LLC 2009

[9] R.W. Selby and V.R. Basili, "Cleanroom Software Development: An Empirical Evaluation," IEEE Trans. Software Eng., vol. 13, no. 9,pp. 1027-1037, Sept. 1987.

[10] http://www.ijric.org/volumes/Vol8/Vol8No8.pdf

[11] http://cairo.cs.uiuc.edu/publications/papers/thesis-achu.pdf

[12] Keng Siau and Lihyunn Lee "Are use case and class diagrams complementary in requirements analysis? An experimental study on use case and class diagrams in UML" Requirements Eng (2004) 9: 229–237, DOI 10.1007/s00766-004-0203-7, Springer-Verlag London Limited 2004.

[13] Nitish Pathak, Girish Sharma and B. M. Singh "Trusted Operating System based Model-Driven Development of Secure Web Applications" Paper accepted for CSI - 2015; CSI - 50th Golden Jubilee Annual Convention, International Conference.

[14] Per Runeson & Martin Höst "Guidelines for conducting and reporting case study research in software engineering" Empir Software Eng (2009) 14:131–164 DOI 10.1007/s10664-008- 102-8, open access at Springerlink.com, December 2008

[15] Nitish Pathak, Girish Sharma and B. M. Singh "Towards Designing of SPF based secure web application using UML 2.0" published in International Journal of Systems Assurance Engineering and Management, Springer, DOI: 10.1007/s13198-015-0338-6 , 28-jan-2015.

[16] MARCO BRAMBILLA, STEFANO CERI, and PIERO FRATERNALI "Process Modeling in Web Applications" ACM Transactions on Software Engineering and Methodology, Vol. 15, No. 4, October 2006.

[17] Georgia M. Kapitsaki, Dimitrios A. Kateros, Christos A. Pappas, Nikolaos D. Tselikas, Iakovos S. Venieris "Model-Driven Development of Composite Web Applications"iiWAS2008, November 24–26, 2008, Linz, Austria.2008 ACM 978-1-60558-349 5/08/0011

[18] Nitish Pathak, Girish Sharma and B. M. Singh "Experimental Designing of SPF based secure web application using Forward Engineering" accepted in IEEE and IETE Sponsored 9th International Conference (INDIACom-2015) on Computing for Nation Development at BVICAM, New Delhi (ISSN 0973-7529 ISBN 978-93-80544-00-7) include in IEEE Xplore.

[19] PETER DESNOYERS, TIMOTHY WOOD, PRASHANT SHENOY "Modellus: Automated Modeling of Complex Internet Data Center Applications" ACM Transactions on the Web, Vol. 6, No. 2, Article 8, Publication date: May 2012.

[20] H. Kim, Y. Zhang, S. Oussena, T. Clark,"A Case study on Model Driven Data Integration for Data Centric SoftwareDevelopment",ACM,2009.

[21] Nitish Pathak, Girish Sharma and B. M. Singh "Experimental analysis of SPF based Secure Web application" published in International Journal of Modern Education and Computer Science (IJMECS), ISSN: 2075-0161 (Print), ISSN: 2075-017X (Online) DOI: 10.5815/ijmecs, Volume 7, No. 2, February 2015, 2, Page.48-55,Hong Kong.

[22] Nikolay Karabutov ,"Frameworks in Problems of Structural Identification Sy stems", International Journal of Intelligent Sy stems and Applications(IJISA), Vol.9, No.1, p p .1-19, 2017. DOI: 10.5815/ijisa.2017.01.01

[23] N. Karabutov, "Structural M ethods of Estimation Lya-punov Exp onents Linear Dy namic Sy stem," International journal intelligent systems and applications, 2015, vol, 7, no.10, p p . 1-11.

[24] M utlaq B. Alotaibi,"M odelling Electronic Sh arin g (E-Sharin g) Beh aviour: A Persp ective on Location- aware Technolo gy ", International Journal of Intelligent Systems and Applications (IJISA), Vol.9, No.1, p p .20-29, 2017. DOI: 10.5815/ijisa.2017.01.02

[25] Siddu P. Algur, Prashant Bhat,"Web Video Object Mining: A Novel Approach for Knowledge Discovery", International Journal of Intelligent Systems and Applications (IJISA), Vol.8, No.4, pp.67-75, 2016. DOI: 10.5815/ijisa.2016.04.08

Authors' Profiles

Mr. Nitish Pathak is an Assistant Professor in Bharati Vidyapeeth's ICAM, GGSIPU, Delhi. Presently, He is pursuing his Ph.D. in Computer Science and Engineering from Uttarakhand Technical University, Dehradun. He received his M.Tech. (CSE), M.Phil. (CS) and did his MCA with Honors from Dr. A.P.J. Abdul Kalam Technical University, Lucknow formerly UPTU. He has More than eleven

Years of extensive Corporate as well as Professional Teaching Experience at Graduate and Post Graduate level. He got Directorate Award (Three times on teacher's Day) for best teaching performance since 2007 to 2009, at ABES Engineering College, Ghaziabad. Further, Got CSI Outstanding Contribution Award-2015 and INDIACom- Outstanding Contribution Award-2015 and INDIACom- Outstanding Contribution Award-2016 for conducting CSI - 50th Golden Jubilee Annual Convention, International Conference and IEEE international conference respectively. These Conference proceedings published by Springer and IEEE respectively, which is indexed with the world's leading Abstracting & Indexing (A&I) databases, including ISI, SCOPUS, DBLP, EI-Compendex, Google Scholar, etc. He has been pro-actively involved with professional associations and is Life member of CSI and ISTE. He has published more than 25 research papers in various reputed International Journals and Conference Proceedings (e.g. IEEE, Springer and ACM etc.).He is the reviewer of various International journals and jointly edited five conference proceedings. His major interests include: Trusted Operating Systems; Software Engineering, OOSE, Data Mining, Technology and Innovation Management etc.

Nitish pathak Got Best Research Paper Publication Award-2017, paper entitled "Towards Designing of SPF based secure web application using UML 2.0" published in International Journal of Systems Assurance Engineering and Management, Springer, and index with ESCI, DBLP, EI-Compendex, OCLC, SCImago, and SCOPUS. Publisher: Springer-Verlag Berlin Heidelberg 2015

Dr. Girish Sharma is currently working as Professor& Head of BPIBS, GGSIP University, Government of NCT OF DELHI. He has More than 21 Years of academic/research and administrative experiences in reputed organizations like Hindu College, Sonepat, ABES Engineering College, Ghaziabad, and College of Engineering Roorkee. He completed his PhD (2005) in Mathematics (Inter Disciplinary) at Department of Mathematics, Gurukul Kangri University, Haridwar. He completed his M.Tech in Computer Technology and Application (2003) from Delhi College of Engineering (DCE), Delhi University, Completed his MCA and M.Sc(Physics) in 1993 and 1996 respectively from Gurukul kangri University, Haridwar. His current research interests include different aspects of Distributed systems,Trusted operating Systems, Object oriented based Forward Engineering and Data Mining. He has published more than 65 research papers in various reputed national / international journals.

Dr. Brijmohan Singh is currently working as Associate Professor& Head of computer science and engineering department at COER,Roorkee. He has More than 15 Years of academic/research and administrative experiences in reputed organizations. He completed his PhD in Computer science and Engineering from Uttrakhand Technical University, Dehradun, UK. He completed his Master of Technology (Computer Science) with First Class from Rajasthan Vidhya peeth Deemed University, Udaipur Rajasthan, in the session 2005 to 2007. He did his Bachelor of Engineering (Computer Science & Engineering) with First Class from College of Engineering Roorkee, in the session 1999 to 2003.His current research interests include different aspects of Digital Image Processing, Mobile Computing, Artificial Intelligence, Pattern Recognition; Multimedia Systems etc. He has published more than 40 research papers in various reputed national / international journals.

How to cite this paper: Nitish Pathak, B. M. Singh, Girish Sharma,"An Empirical Perspective of Roundtrip Engineering for the Development of Secure Web Application Using UML 2.0", International Journal of Intelligent Systems and Applications(IJISA), Vol.9, No.5, pp.43-54, 2017. DOI: 10.5815/ijisa.2017.05.06

