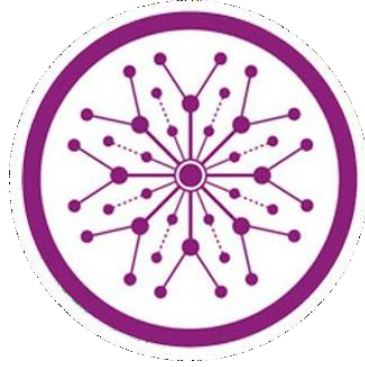


SUPERIOR UNIVERSITY LAHORE



Faculty of Computer Science & IT

Final Year Project PROJECT REPORT

[Online Shopping Management System]

Project ID: [FYP-F19-013]

Project Team

Student Name	Student ID	Program	Contact Number	Email Address
Muhammad Imran Shabbir	MCSW-F19-014	MCS (Weekend)	0321-4244076 0305-4477801 (WhatsApp)	Mcsw-f19-014@superior.edu.pk Imranshabbir.lhc@gmail.com
Hina Zaheer	MCSW-F19-011	MCS (Weekend)	0309-6774028	Mcsw-f19-011@superior.edu.pk
Amna Javed	MCSE-F19-008	MCS (Weekend)	0323-4525600	Mcse-f19-008@superior.edu.pk

[Project Supervisor]
([M. Arif Sb - Lecturer])

Project Report

[Online Shopping Management System]

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Muhammad Imran Shabbir	1.0	02.05.2021	<Original Draft>	
Hina Zaheer	2.0	05.05.2021	<Changes Based on Feedback from Supervisor>	
Amna Javed	3.0	06.05.2021	<Changes Based on Feedback From Faculty>	
Muhammad Imran Shabbir	4.0	08.05.2021	Backend Database designed using Strapi for testing purpose initially	
Amna Javed	5.0	12.05.2021	Products updated after Consultation with Project Supervisor	
Muhammad Imran Shabbir	6.0	15.05.2021	Product detail page was managed and tested with consultation of Res Sir M. Arif Sb	
Hina Zaheer	7.0	20.05.2021	Front end management entirely supervised by Hina Zaheer	
Muhammad Imran Shabbir	8.0	25.05.2021	Finally the project was synchronized with Mongo DB using Node JS and Express Frameworks	
Hina Zaheer	9.0		Draft compiled	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

Name: Muhammad Arif

Date: 11th May 2021 Signature: _____

PROJECT MANAGER

Comments: _____

Date: _____ Signature: _____

HEAD OF THE DEPARTMENT

Comments: _____

Date: _____ Signature: _____

Declaration

This is certifying that the project entitled "Online Shopping Management System" is submitted by ***Muhammad Imran Shabbir (MCSW-F19-014), Hina Zaheer (MCSW-F19-011) as well as Amina Javed (MCSE-F19-008)***, in partial fulfillment of the requirement for the award of degree of Masters of Computer Science to the department of Computer Science, Superior Group of Colleges (Gold Campus), Lahore, comprises our group members original work and due acknowledgement has been made in the text to all other material used. The result of the project has not been submitted to any other university or institute for the award of any degree or diploma before submission to Superior Group of Colleges. The task accomplished under the kind supervision and assistance of ***Respected Sir M. Arif Sb*** who always shown willingness and readily available round the clock to decipher any query raised by the group member during its completion process.

Acknowledgements

All thanks to Almighty ALLAH, the creator and the owner of this universe, the most merciful, beneficent and the most gracious, who provided us guidance, strength and abilities to complete the subject report.

First, we would like to thank Respected Sir M. Arif, our project supervisor, for guiding us through each and every step of the process with knowledge, suggestions and endless support. This project / report would not have been possible without his continuous support. Thank you for your advice, guidance and assistance.

Finally, we would like to thank my beloved parents, sisters, guardians and group members for their never ending support, motivation and believe on us during the entire process from inception till end.

Executive Summary

The Online Shopping Management System is a web based application intended for online customers and retailers across the country. The main objective of this application is to make it interactive and its ease of use. It would make searching, viewing and selection of a product easier. It contains a sophisticated search engine for user's to search for products specific to their needs. The search engine provides an easy and convenient way to search for products where a user can Search for a product interactively and the search engine would refine the products available based on the user's input. The user can then view the complete specification of each product. They can also view the product reviews and also write their own reviews. The application also provides a drag and drop feature so that a user can add a product to the shopping cart by dragging the item in to the shopping cart. The main emphasis lies in providing a user-friendly search engine for effectively showing the desired results and its drag and drop behavior.

TABLE OF CONTENT

Declaration	4
Acknowledgment.....	5
Executive Summary	6
Table of Contents	7-8
List of Figures.....	9
List of Tables	10
Chapter 01: Introduction	11
1.1. Motivation.....	12
1.2. Contribution Summary.....	12
1.3. Report Outline.....	13
2. Proposed System and Methodology	14
2.1. System Benefit	15
2.1.1. Existing system	15
2.1.2. Proposed System	15
2.1.3. Benefits of Proposed System.....	16
2.2. Software Process Model	17
2.2.1. The Systems Development Life Cycle (SDLC).....	17
2.2.2. Why following a model?	18
2.3. Methodology	18
2.3.1. Waterfall model.....	18
2.3.2. Advantage	19
2.3.3. Why choose WATERFALL mode I?.....	19
2.4. Feasibility study	19
3. Chapter 03: Requirement Analysis.....	21
3.1. User Requirement.....	22
3.2. Administrator Requirements	22
3.3. System Requirement	23
3.3.1. Functional Requirement	23
3.3.2. Non-Functional Requirements	23
3.4. Use Case Diagram	24
3.5. Platform Specification - Deployment	25
3.5.1. Hardware Specification	25
3.5.2. Software Specification.....	25
4. Chapter 04: Analysis	26
4.1. Activity / ER Diagrams.....	27
4.1.1. Admin Panel Login Activity Diagram	27
4.1.2. Add Product to List Activity Diagram	27
4.1.3. Order Process Activity Diagram	27
4.1.4. Checkout or Payment and Shipping Activity Diagram	27
4.2. Data Flow Diagram	28
4.3. Use Case Diagram	29
4.4. Class Diagram	30

4.5. Risk Protection	31
5. Chapter 05: Design	32
5.1. Design Goals	33
5.1.1. Architectural Design	33
5.1.1.1. Entity Relationship Diagram	33
5.2.2. Description of A.D	34
5.2. Database Design	35
5.3. Database Field Design	35
5.4. Modules of System	35
5.5. Interface Design	35
6. Chapter 06: Testing & Evaluation	36
6.1. Software Verification and Validation	37
6.1.1. Unit Testing	37
6.1.2. Integration Testing	38
6.1.3. Validation Testing	39
6.1.4. White Box Testing	39
6.1.5. Performance Testing	39
6.2. Software Quality Assurance (SQA)	42
5.3 Test Plan	42
7. Chapter 07: Implementation and Maintenance.....	43
7.1. Implementation	43
7.2. Maintenance	44
8. Chapter: 08: Summary, Conclusion & Future Plan	45
8. 1. Limitations and Future Plan	46
8.2 Experiences.....	46
8.2. Conclusion	46
9. Appendix	47
Appendix- A. Glossary	48
Reference	49-50

LIST OF FIGURE

Figure 01: Software Development Life Cycle (SDLC) 17

Figure 02: Waterfall model 18

Figure 03: Use Case Diagram 24

Figure 04: Add Product to List Activity Diagram 27

Figure 05: Context Level Diagram 28

Figure 06: Class Diagram..... 30

Figure 07: Architecture Diagram 33

LIST OF TABLE

Table 01: Requirement Analysis	22
Table 02: Functional Requirement	22
Table 03: Non Functional Requirement	23

Chapter 1

Introduction

Chapter 1: Introduction

1.1 Motivation

The main objective behind this project is to develop a web oriented application which can provide an online shopping feature to the users. In other words, this project aimed at creating a virtual shop environment for users, in some handy form, which will be available to them through internet. Although the idea of developing online shopping websites are not new in the electronic market and has been evolved soon after the World Wide Web (www). It is reasonable to say that the process of shopping on the web is becoming commonplace. The objective of this project is to develop a general purpose e-commerce store where any product (such as books, CDs, computers, mobile phones, electronic items, and home appliances) can be bought from the comfort of home through the Internet. Customers can browse the catalog and select products of interest. User can login into ecommerce website, once he logged in then automatically one shopping cart will be created, once user select an item for buy it will add to that cart. In case user thinks the selected item is not useful for him, then he can delete that item form the cart. More information will be needed to complete the transaction if customer want to buy item. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and select paymentoption. This system has been designed keeping in mind al) the aspects such as loading the data, complexity and maintaining the security of user credentials.

1.2 Contribution Summary

In current system user must go to shop and order products. It is difficult to identify the required product. Also there are expenses for travelling from house to shop. It requires lots of time and time means a lot to everyone. Now people trying to make shopping system more easier by using online shopping system. Previously many work has been done on online

shopping system but those are less user-friendly because user interface is not user friendly. All are focused some special types of customer that's why their interface is not user-friendly for all. Here my system ensures that every one can use this system more easily, we focused everyone, and interface is more user friendly for all. Here we focused on this problem and made a user friendly interface for all, here is no need to special knowledge for buying anything. Customer can buy product easily.

1.3 Report Outline

Chapter 02 contains existing system, proposed system and its benefit also discuss on SDLC and Methodology which is used in this project and that is waterfall model.

Chapter 03 discuss about all types of requirements of this system also use case diagram and all the tools which is used in this project.

Chapter 04 contain total analysis and diagrams also discuss on project risk, project schedule and all the function's description.

Chapter 05 shows the interface of this system

Chapter 06 shows some of the row code of the system

Chapter 07 discuss about testing phases of this project.

Chapter 08 discuss on maintenance and implementation.

Chapter 09 concludes the report also discusses about future opportunities from this project and experience.

Chapter 2

Proposed System & Methodology

Chapter 2: Proposed System & Methodology

2.1 System Benefit

2.1.1 Existing system

The present scenario for shopping is to visit the shops and market manually and then from the available product list one needs to choose the item he or she wants and then payment for the same item mainly in cash mode is done, as not every society is well educated and aware to use net banking or card modes or wallets etc. This system is not much user friendly as one needs to go to the market physically and then select items only from the available-list. So mostly it is difficult to get the product as per our desire. Description about the products are less available and are mostly verbal only. For this type of shopping one needs to have ample amount of free time. Also not really good markets exist everywhere, so many times good markets become out of reach for certain people. This existing system of buying goods has several disadvantages. It requires lots of time to travel to the particular shop to buy the goods. It is having lots of manual work. Since everyone is leading busy life now a days, time means a lot to everyone. Also there are expenses for travelling from house to shop. It is less user-friendly. In current system user must go to shop and order products. It is difficult to identify the required product. More over the shop from where we would like to buy something may not be open 24/7 and 365 day of the calendar. Hence we have to adjust our time with the shopkeeper's time or vendor's time. In current system user have to go shop to view the description of the product.

2.1.2 Proposed System

The proposed system helps in building a website to buy, sell products or goods online using internet connection. Unlike traditional commerce that is earned out physically with effort of a person to go and get products, my project will make it easier for human to reduce physical work and to save time. The basic concept of the application is to allow the customer to shop

virtually using the Internet and allow customers to buy the items of their desire from the store. In this proposed system customers need not to go to the shops for purchasing the products. Customer can order the product he/she wishes to buy through the use of this system. The shop owner can be the admin of the system. Shop owner can appoint officials particularly to handle this, who will help owner in managing the customers and product orders. The system also endorses a home delivery system for delivering the purchased products. Enables consumers to shop or do other transactions 24 hours a day, all year round from almost any location. It can be accessed over the Internet purchasing of goods online, user can choose different products based on categories, online payments, delivery services and hence covering the disadvantages of the existing system and making the buying easier and helping the vendors to reach wider market. It provides consumers with more choices. Customer can purchase products online. A good ecommerce site fulfill the following factors.

- Knowing when an item was saved or not saved in the shopping cart.
- Returning to different parts of the site after adding an item to the shopping cart. Easy scanning and selecting items in a list.
- Effective categorical organization of products.
- Simple navigation from home page to information and order links for specific products. Obvious shopping links or buttons.
- Minimal and effective security notifications or messages
- Consistent layout of product information
- Admin has the authority to add new particulars to the items list whenever needed. Permission to administrator to remove items, anytime.
- Allows the admin to modify the price of each item, whenever required or felt like. Admin has the authority to update description of each item.
- Permission to the admin to view information about each customer who checkouts the items list.

2.1.3 Benefits of Proposed System

The main advantage of this system over traditional commerce is the user can browse online shops, compare prices and order merchandise sitting at home on their PC. Secure registration and profile management facilities for customers. Shopping cart feature allows online shopping customers to "place" items in the cart. It decreases the cost of creating, processing, distributing, storing and retrieving paper-based information Expands the marketplace to national and

international markets. Upon "checkout" the software calculates as total for the order including shipping and handling postage, packing and taxes, if applicable. Reduces the time between the outlay of capital and the receipt of products and services.

2.2 Software Process Model

2.2.1 The Systems Development Life Cycle (SDLC)

It is a conceptual model used in project management that describes the stages involved in an information system development project from an initial feasibility study through maintenance of the completed application. Various SDLC methodologies have been developed to guide the processes involved including the waterfall model (the original SDLC method). Some methods work better for specific types of projects, but in the final analysis, the most important factor for the success of a project may be how closely particular plan was followed.

A Software Development Life Cycle is essentially a series of steps, or phases, that provide a model for the development and lifecycle management of an application or piece of software. The Software Development Life Cycle is a process that ensures good software is built. Each phase in the life cycle has its own process and deliverables that feed into the next phase. Phases are given below:

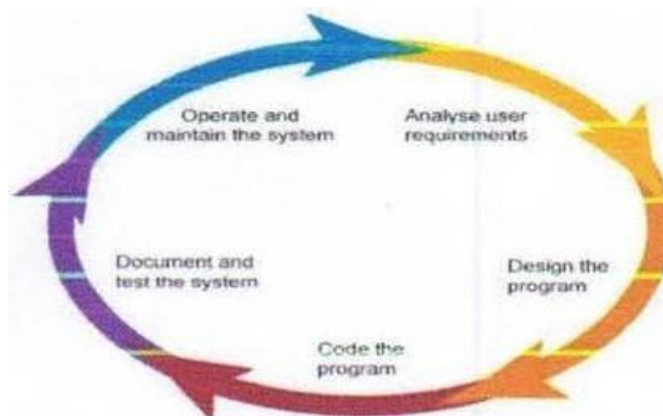


Figure 01: Software Development Life Cycle (SDLC)

SDLC Models are:-

- ▶ Water Fall Model
- ▶ The Prototyping model / Evolutionary development
- ▶ Spiral Model
- ▶ The Incremental model
- ▶ Agile model

2.2.2 Why following a model?

- ▶ Without using of a particular life cycle model the development of a software product would not be in a systematic and disciplined manner.
- ▶ When a software product is being developed by a team there must be clear understanding among team members about what to do and when to do. Not following - confusion and project failure.

2.3 Methodology

2.3.1 Waterfall model

Here using waterfall mod for this project.

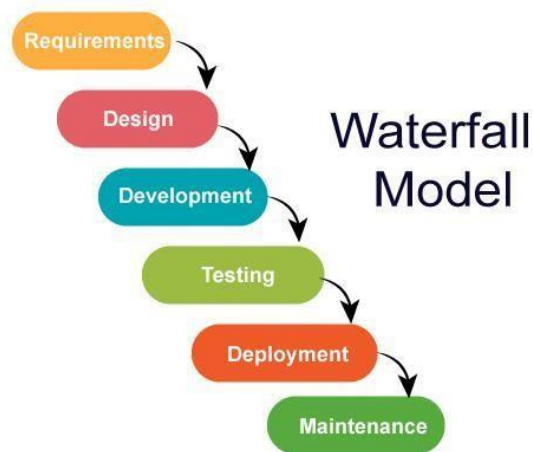


Figure 02: Waterfall model

2.3.3 Advantage

- ▶ The waterfall model are that documentation is produced at each phase and that it fits with other engineering process models.
- ▶ Disciplined approach
- ▶ Careful checking by the Software Quality Assurance Group at the end of each phase. (Or testing in each phase.)
- ▶ Documentation available at the end of each phase
- ▶ Linear model.
- ▶ Easy to understand and implement [1].
- ▶ Identifies deliverables and milestones

2.3.4 Why choose WATERFALL model?

For following reason we used this model

- ▶ My requirements are very well known clear and fixed.
- ▶ Product definition is stable.
- ▶ Technology is understood [1].
- ▶ There are no ambiguous requirements
- ▶ Ample resources with required expertise are available freely the project is not so long.

2.4 Feasibility study

The software is said to have life cycle composed of several phases. At the feasibility stage, it is desirable that two or three different configuration will be pursued that satisfy the key technical requirement but which represent different level of ambition and cost. Feasibility is the destination of whether or not a project is worth doing. A feasibility study is carried out select a best system that mate performance requirements. The data collected

during primary investigation examines system feasibilities that is likelihood that the system will be beneficial to the organization. Four tests for feasibility study are as follows:-

Technical Feasibility: This is concerned with specifying equipment and software that will successfully satisfy the use considerably, but might include the feasibility to produce output in a given time because system is fast enough to handle multiple users. Response time under certain circumstances and ability to process a certain volume of transaction of a particular speed. Feasibility to communicate data to distant location. All this are successfully fulfil this project.

Economic Feasibility: Economically this system is feasible.

Operational Feasibility: It is mainly related to human organizational as social aspects. The points to be considered are - this system interface is standard, user friendly and provides extensive help. Hence no special training is not required.

Social Feasibility: Social feasibility is determination of whether a proposed project will be acceptable to people or not, so this project is totally Social and Feasible.

Chapter 3

Requirement Analysis

Chapter 3: Requirement Analysis

3.1 User Requirement

End user features that facilitate the enjoyment of Internet shopping include the following [8]:

- ▶ Users should be able to use the ecommerce application from any Web browser supporting I-ITML 3.2, HTML 5 (or later) and cookies.
- ▶ Visitors new to the site should be able to register by themselves. Users will be differentiated by unique user identifiers..
- ▶ Transactions should be secure. That is, a basic authentication mechanism must be built into the application to prevent unauthorized persons from making transactions on a user's behalf. Site visitors should be able to purchase goods or services via the electronic store.
- ▶ Users should be able to view a complete list of specified items available through the site.
- ▶ Users should be able to search for items by related attributes. For example, visitors might search for CDs by artist, album title and/or genre or search for books by author, title and/or ISBN number
- ▶ Site visitors should be able to search the database using relevant keywords to identify items of interest.
- ▶ Users should be able to select items of interest and add them to their shopping carts for future purchase.
- ▶ All selected items should be shipped to the user following purchase.
- ▶ Large numbers of users should be able to use the application simultaneously.

3.2 Administrator Requirements

Administrator who manage the site have specific requirements of their own [8]

Administrators should be able to manage ecommerce applications using web browsers.

Data managers should be able to delete users.

Site administrators should be able to change the status of goods purchased by users after

items have been shipped.

Administrators should be able to view all user transactions.

Site managers should be able to view all transactions for the day.

3.3 System Requirement

The System must provide following.

- User should have appropriate version of windows.
- System should have up to 2 GB ram minimum requirement for the application.
- Site should be uploaded on server.
- Need a web browser.
- Need an installed code editor.

So, this is the overall process of making the Online Shopping system worked, and user can get the policy without any headache of agent and sometime don not need to provide commission to them.

3.3.1 Functional requirement

The System must provide following functionalities [8]

- ▶ Keeping records of registration of customers.
- ▶ Keeping the records of products.
- ▶ Keeping the daily sell Information.
- ▶ Storing the feedback given by the customer.
- ▶ Keeping details about the product it is delivered or not etc.
- ▶ Storing the items selected by the customer in the temporary storage/Wish list.
- ▶ Full control of admin panel will be in admin hand.

3.3.2 Non- Functional Requirements

Following Non-functional requirements will be there in the Insurance on internet [8]:

- ▶ Secure access of confidential data (customer's details).

3.5 Platform Specifications – Deployment

3.5.1 Hardware Specification

Processor P IV, Intel or Later

RAM 2 GB

Minimum Space Required 50 GB

Display 16 bit color

3.5.2 Software Specification

Operating Environment Win 7/8/10/11 or later version

Platform / Languages

- NodeJS, Express, React,
- Mongo DB, Strapi

Chapter 4

Analysis

Chapter 4: Analysis

4. Analysis

After carefully analyzing the requirements and functionality of the web application, I had two important diagrams by the end of the analysis phase. They are the ER diagram and data flow diagram which were the basis for finding out entities and relationships between them, the flow of information.

4.1 Activity / ER Diagram

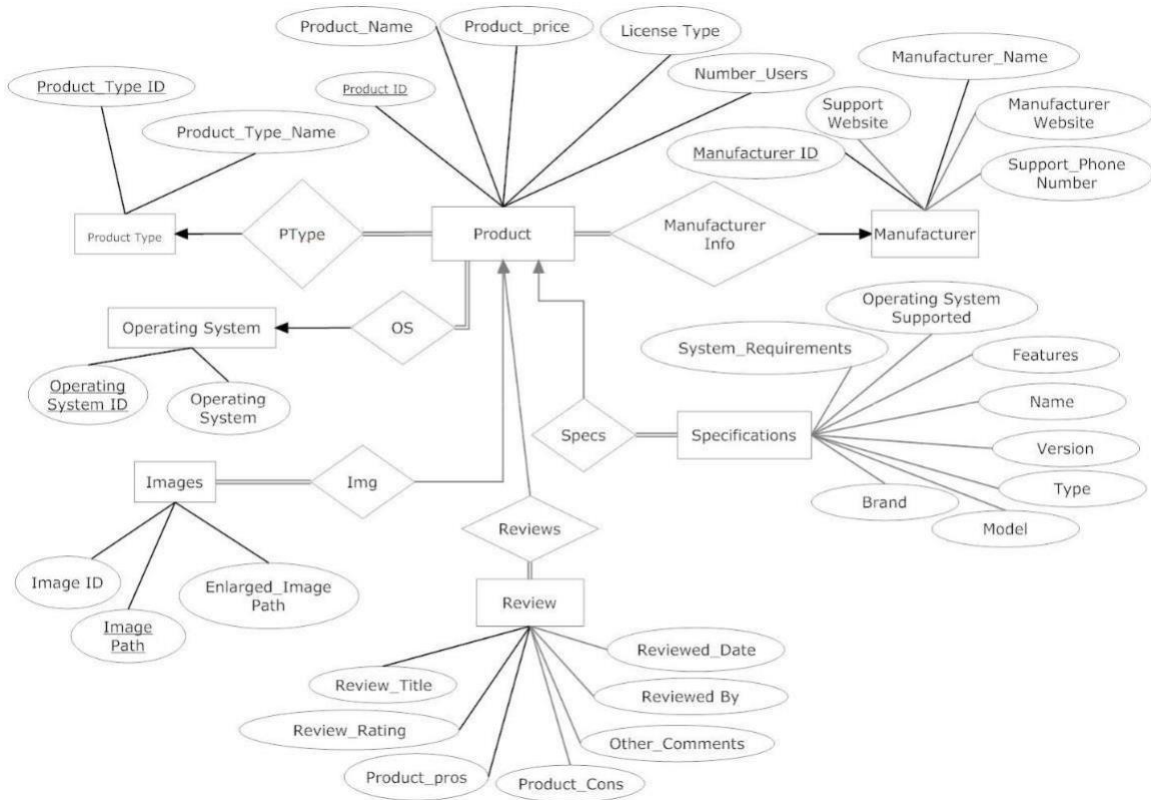


Figure 4.1 Entity Relation Ship Diagram

4.2 Data Flow Diagram

CONTEXT LEVEL DIAGRAM



Figure 4.2 : A Context Level Diagram

First Level DFD

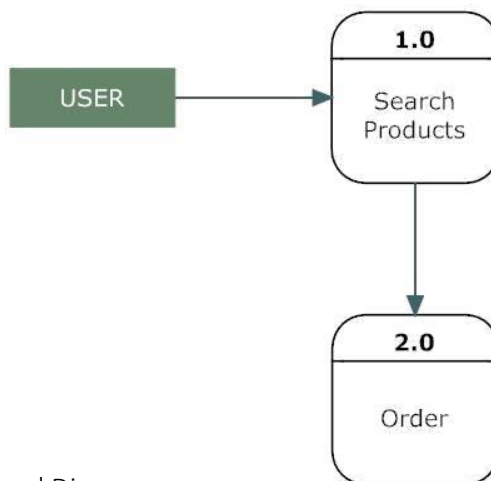


Figure 4.3: A First Level Diagram

SECOND LEVEL DFD

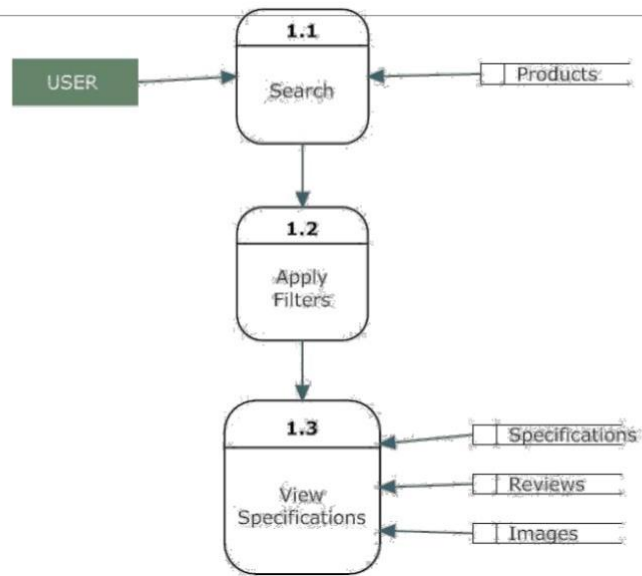


Figure 3.4: A Second Level Diagram

4.3 Use Case Diagram

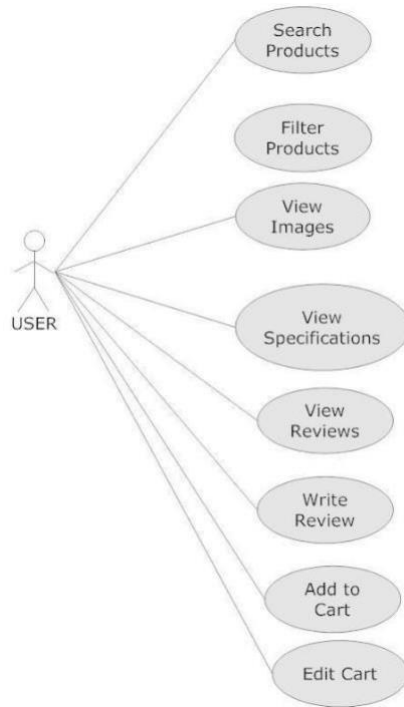
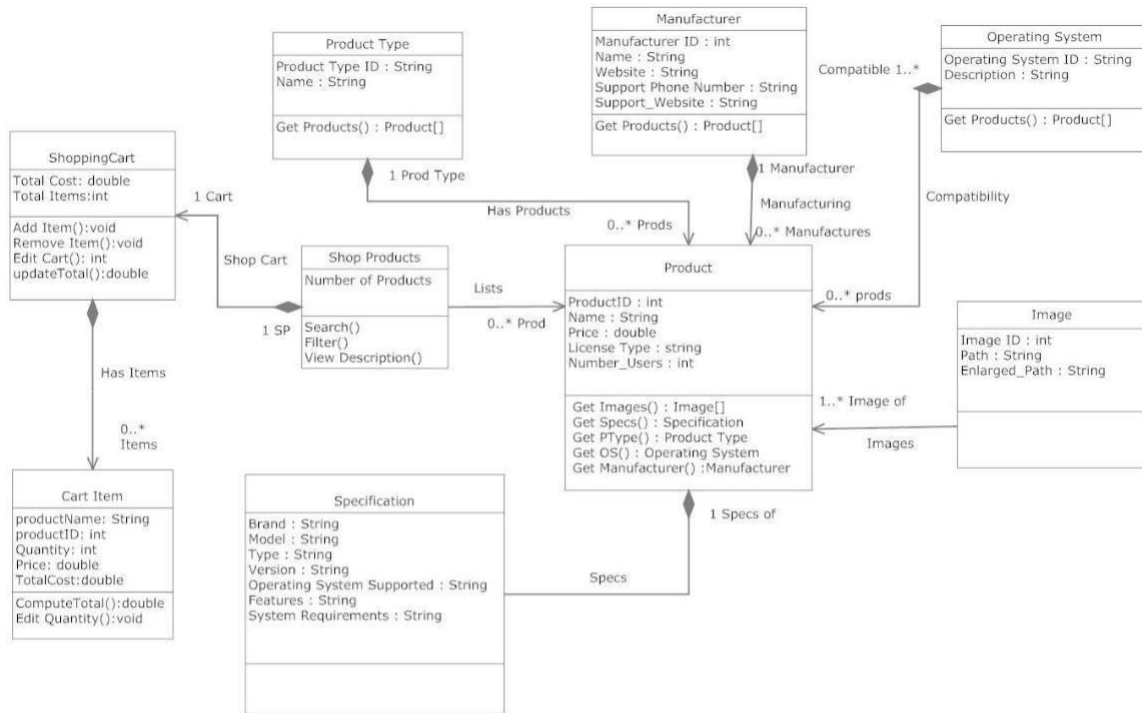


Figure 4.6: Use Case Diagram

4.4 Class Diagram



CLASS DIAGRAM

Figure 3.7: Class Diagram

Following Table 01 shows the risk projection

Table 01: Risk Projection

▶ Technology	No Internet connection	1
	The database used in the system cannot process as many transactions per second as expected	2
▶ People	It is impossible to hire staff with the skills required.	3
	Key staffs are ill and unavailable at critical times.	4

	Required training for staff is not available.	5
▶ Tool	The code generated by software code generation tools is inefficient.	6
	Software tools cannot work together in an integrated way.	7
▶ Requirement	Changes to requirements that require major design rework are proposed.	8
	Customers fail to understand the impact of requirements changes.	9
▶ Estimation	The time required to develop the software is underestimated.	10
	The rate of defect repair is underestimated.	11
	The size of the software is underestimated.	12

Chapter 5

Design

Chapter 5: Design

5.1 Design Goals

- The design of the web application involves the design of the forms for listing the products, search for products, display the complete specification for the product, and design a shopping cart that is easy to use.
- Design of an interactive application that enables the user to filter the products based on different parameters.
- Design of an application that has features like drag and drop etc.
- Design of application that decreases data transfers between the client and the server.

5.1.1 Architectural Design

5.2.1 Architectural Context Diagram

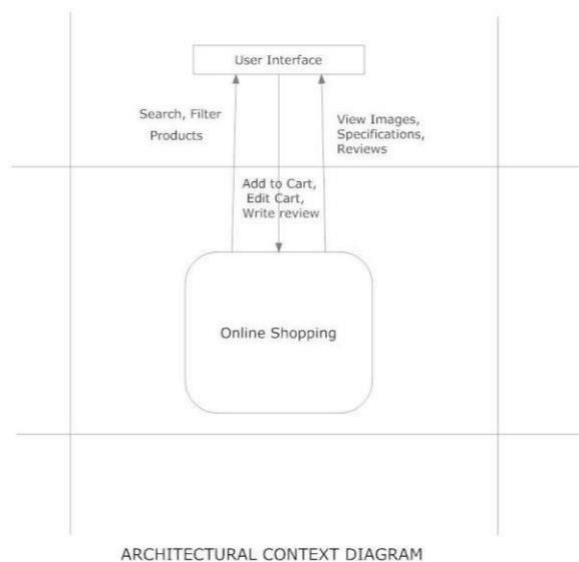


Figure 5.1 Architectural Context Diagram

5.2.2 Description of Architectural Design

In this context diagram, the information provided to and received from the 'Online Shopping' is identified. The arrows represent the information received or generated by the application. The closed boxes represent the set of sources and sinks of information.

In the system, we can observe that the user interacts with the application through a graphical user interface. The inputs to the system are the Search and Filter criteria provided by the user and a new review written by the user. Also, the output is in the form of Repeater and grid views which present the users with list of Products available. The users can view complete specification, view Images and reviews by other users.

5.3 Procedural/Modular Approach

Following are all the modules designed for the Online Shopping System.

5.3.1 Shop Products Module

This module starts when the user visits the home page or when a user searches for a product by entering a search term. This part of the application includes displaying all the products that are available or the products that match the search term entered by the user. The user can then filter these products based on various parameters like manufacturer, product type, operating system supported or a price range. The user browse through the products and each product would be displayed with an image and its features like operating system supported, number of user licenses and if it is a full version or an upgrade version. A user can add a product to the cart either by dragging the product and dropping it in the cart or by clicking a button. The user would be able to see the shopping cart summary.

5.3.2 Product Description Module

This module starts when a user visits the product description page. A user can view various images of the product of different sizes. The user can see an enlarged image in a popup window. The user can view the complete specification of the product like its features, operating system supported, system requirements etc. A user can also view the manufacturer information and also information about rebates, exchange policies etc. A user can also view the reviews of the product. A user can also write a review for the product.

5.3.3 Shopping Cart Module

This module starts when the user views the shopping cart. All the products that have been added to the shopping cart by the user are listed along with their price and the quantity. The total price of all the products added to cart is displayed. A user can edit the quantity of each product or remove the product from the shopping cart. A user can remove the product from the cart by clicking a button or by dragging the product and dropping it outside the cart. The total price changes accordingly when a user edits the quantity of a product or when a product is removed from the cart.

(Database Design & Interface will be provided at the time of Designing)

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

6. Testing

Software testing is a process of running with intent of finding errors in software. Software testing assures the quality of software and represents final review of other phases of software like specification, design, code generation etc.

6.1.1 Unit Testing

Unit testing emphasizes the verification effort on the smallest unit of software design i.e.; a software component or module. Unit testing is a dynamic method for verification, where program is actually compiled and executed. Unit testing is performed in parallel with the coding phase. Unit testing tests units or modules not the whole software.

I have tested each view/module of the application individually. As the modules were built up testing was carried out simultaneously, tracking out each and every kind of input and checking the corresponding output until module is working correctly.

The functionality of the modules was also tested as separate units. Each of the three modules was tested as separate units. In each module all the functionalities were tested in isolation.

In the Shop Products Module when a product has been added to cart it has been made sure that if the item already exists in the shopping cart then the quantity is increased by one else a new item is created in the shopping cart. Also the state of the system after a product has been dragged in to the shopping cart is same as the state of the system if it was added by clicking the add to cart button. Also it has been ensured that all the images of the products displayed in the shop products page are drag gable and have the product property so that they can be dropped in the cart area.

In the Product Description Module it has been tested that all the images are displayed properly. Users can add review and the as soon as a user adds a review it is updated in the view customer review tab. It has been checked to see if the whole page refreshes or a partial page update happens when a user writes a review.

In the Cart Details it has been tested that when a user edits a quantity or removes a product from the cart, the total price is updated accordingly. It has been checked to see if the whole page refreshes or a partial page update happens when a user edits the cart.

Node JS LTD, Express has in built support for testing the application. The unit testing can be done using Express without the need of any external application. Various methods have been created for the purpose of unit testing. Test cases are automatically generated for these methods. The tests run under the NodeJS context which means settings from Web. config file are automatically picked up once the test case starts running.

6.1.2 Integration Testing

In integration testing a system consisting of different modules is tested for problems arising from component interaction. Integration testing should be developed from the system specification. Firstly, a minimum configuration must be integrated and tested.

In my project I have done integration testing in a bottom up fashion i.e. in this project I have started construction and testing with atomic modules. After unit testing the modules are integrated one by one and then tested the system for problems arising from component interaction.

6.3 Validation Testing

It provides final assurances that software meets all functional, behavioral & performance requirement. Black box testing techniques are used.

There are three main components

Validation test criteria (no. in place of no. & char in place of char)

Configuration review (to ensure the completeness of s/w configuration.)

Alpha & Beta testing-Alpha testing is done at developer's site i.e. at home & Beta testing once it is deployed. Since I have not deployed my application, I could not do the Beta testing.

Test Cases- I have used a number of test cases for testing the product. There were different cases for which different inputs were used to check whether desired output is produced or not.

1. Addition of a new product to the cart should create a new row in the shopping cart.
2. Addition of an existing product to the cart has to update the quantity of the product.
3. Any changes to items in the cart have to update the summary correctly.
4. Because same page is inserting data into more than one table in the database atomicity of the transaction is tested.
5. The state of the system after a product has been dragged in to the cart should be same as the state of the system if the same product is added to the cart by clicking a button.

6.4 White Box Testing

In white box testing knowing the internal working of the product, tests can be conducted to ensure that internal operations are performed according to specification

and all internal components have been adequately exercised. In white box testing logical path through the software are tested by providing test cases that exercise specific sets of conditions and loops.

Using white-box testing software developer can derive test case that

- Guarantee that all independent paths within a module have been exercised at least once.
- Exercise all logical decisions on their true and false side.
- Exercise all loops at their boundaries and within their operational bound.
- Exercise internal data structure to ensure their validity.

At every stage of project development I have tested the logics of the program by supplying the invalid inputs and generating the respective error messages. All the loops and conditional statements are tested to the boundary conditions and validated properly.

6.5 Performance Testing

Jakarta JMeter, a tool for testing applications was used to simulate the virtual users (clients) and test the performance of the system. It can be used to test performance both on static and dynamic resources (files, Servlets, Perl scripts, Java Objects, Data Bases and Queries, FTP Servers and more). It can be used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types. It can be used to make a graphical analysis of performance and test the server/script/object behavior under heavy concurrent load.

I have done performance testing to achieve an estimate of the peak and sustained load the application. This has done with few pages like the Shop Products (extensive Database access, business logic Intensive and more Images) and the Cart Details (simple page). A few sample Screenshots of test results are shown below. The tests have been conducted by running the application (server) and JMeter on same machine. These test results do

not include factors like network bandwidth etc as the server is running on the same machine along with JMeter.

Photo will be pasted here

Comparison of Constant Users vs. Constant Loop-Count

In the following test, the number of users has been kept constant and the Loop Count has been increased.

Users	Loop Count	Ramp Up period(sec)	Cart Details Page Average Response(ms)
100	150	10	1060
100	300	10	1071
100	600	10	1021
100	1000	10	1033

The above data shows high response average which increases with the loop count.

Observations

Response Time increases rapidly with number of users but not very much when the users are kept constant and only loop-count is increased. This is because, if the number of users is kept constant and only the loop count is increased, the number of requests handled by the server per second remains constant for every execution of the loop count and for every increase in the loop count. Hence the response time will not increase drastically in this case. Whereas, if the users are increased and loop count is kept constant, the requests handled by the server per second increases with increasing users and hence the longer response time.

Local Testing:

Users	Loop Count	Ramp Up period(sec)	Cart Details Page(ms)	Shop Products page(ms)
100	150	10	1060	28177
500	150	10	8075	105388
1000	150	10	11993	135273

Observations:

Response Time of a complex webpage with database and business logic functions is far more than a simple webpage.

Factors affecting Response Time:

- Limited System Hardware Resources (CPU, RAM, Disks) and Configuration
- JMeter Tests and Application running on the same machine.

Remote Testing:

Users	Loop Count	Ramp Up period(sec)	Cart Details Page(ms)	Shop Products page(ms)
100	150	10	792	8312
500	150	10	6392	99069
1000	150	10	20457	227056

6.5. Software Quality Assurance (SQA)

After examine and change the software engineering process itself to reduce the amount of faults that end up in the delivered software: the so-called defect rate. It has also been checked by the specialist

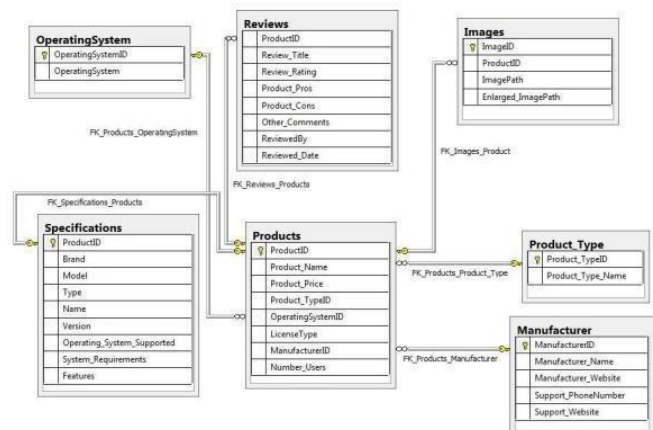
Chapter 7

Implementation & Maintenance

7.1 Implementation

Though this project was only for education purpose, implement it properly by the help of some of friends and users. The implementation phase involves installing approved applications into production environments. Primary tasks was include announcing the implementation schedule, training end

users, and installing the project. Additionally, my fiends input and verify data, configure and test system and security parameters, and conduct post-implementation reviews. After installing the project, Input pre-existing data manually and electronically to the new system.



Verifying the accuracy of the input data and security configurations was a critical part of the implementation process.

Organizations often run a new system in parallel with an old system until they verify the accuracy and reliability of the new system. Employees should document any programming, procedural or configuration changes made during the verification process.

For implementation of the website project

The website can be installed on a computer or a server which has NodeJS, Mongo DB installed in it.

The owners of the website are to be properly trained to use all the features of the website, giving details of each features of the website.

To show the accuracy of the website and conformance of the website to the requirements of the owners or users of the website

7.2 Maintenance

The maintenance phase involves making changes to hardware, software, and documentation to support its operational effectiveness. It includes making changes to improve a system' s performance, correct problems, enhance security, or address user requirements. To ensure modifications do not disrupt operations or degrade a system's performance or security,

Organizations should establish appropriate change management standards and procedures. Routine changes are not as complex as major modifications and can usually be implemented in the normal course of business. Routine change controls should include procedures for requesting, evaluating,

Approving, testing, installing, and documenting software modifications. Maintaining accurate, up- to-date hardware and software inventories is a critical part of all change management processes, Management should carefully document all modifications to ensure accurate system inventories . Management should coordinate all technology related changes through an oversight committee and assign an appropriate party responsibility for administering software patch management programs. Quality assurance, security, audit, regulatory compliance, network, and end-user personnel should be appropriately included in change management processes. For maintenance of the website:

Chapter 8

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

8.1 Future Plan

Some of the features that will be modified and added to this system in future. Online transaction/Credit card validation is not done. Third party proprietary software can be used for validation check. All e-mail notification is sent to the customer as soon as the order is placed.

8.2 Experiences

During the making/developing of this project explored new ideas and functionality behind the working of a notepad. This project is the output of our planning, schedule , programming skill and the hard work, and this report reflects our steps taken at various levels of programming skill, planning and schedule. Learnt a lot during this project and liked the improvement in my testing skills and deep concept related to these kinds of projects.

8.3 Conclusions

Finally successfully develop and implement the site ' Online Shopping Management System' . With the help of various kinds of links and tools, includes all the features which was basic requirement for an ecommerce web site. Provide an ecommerce site which is live and running on the web. Finally got success in our attempt to take care of the needs of both the customers as well as the administrator which was our main objectives.

Appendix

Appendix- A. Glossary

Project scheduling - Project scheduling is an activity of distributing the estimated efforts within the planned project duration

Entity Relationship (ER) diagram - The Entity Relationship diagram describes data as entities, relationships and attributes. ERD is known as the graphical representation of the database

Entity - An entity is an object with physical existence or may be an object with conceptual existence.

Attribute - Attributes are the particular properties which are used to describe the entity.

Relationship - A relationship is a logical linkage between two or more entities which describes how the entities are associated with each other.

Primary Key: A primary key is an attribute or collection of attributes that allow us to identify an entity uniquely.

Foreign key: A foreign key is an attribute of a relation which refers to an existing attribute of another relationship

HTML: Hypertext Markup Language, is the predominant markup language for web pages

CSS-for design front end need CSS.

ReactJS: is a widely-used open source general-purpose scripting language that is especially suited for web development / web app and can be embedded into HTML.

MongoDB: MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License.

JavaScript: It is a scripting language most often used for client-side web development.

Reference

1. I Putu Dody Lesmana, Rinda Nurul Kari.mah, Beni Widiaw an, "Agile-Waterfall hybrid for prevention information system of dengue viral infections: A case study in Health Department of Jember: East Java Indonesia", *JC T and Knowledge Engineering (IC T&KE) 2016 14th international Conference on*, pp. 1-6, 2016, ISSN 2157-099X.
2. Adel Mohammad, Ja'far Alqatawna, Mohammad Abushariah., "Secure software engineering: Evaluation of emerging trends", *Information Technology (JC IT) 2017 8th International Conference on*, pp. 814-818, 2017.
3. P. M. Khan, Kaleem A. Quraishi, "Impact of RACI on Delivery and Outcome of Software Development Projects", *Advanced Computing & Co mmuni cation Technologies (ACCT) 2014 Fourth International Conference on*, pp. 177-184, 2014.
4. Liu Meilu, Li Xiaoping, "Under the environment of B2B logistics service quality and relationship quality", *An Empirical Study of Commercial Economic Research*, pp. 41-42, 2015.
5. W. H. DeLone, E. R. McLean, "Measuring e-Commerce Success: Applying the DeLone & McLean InfonationSystems Success Model", *InJernational Jo1Jrnal of Electronic Commerce*, vol. 9, no. 1, pp. 31-47, 2004.
6. Y. Bakos, "The Emerging Landscape for Retail E -Commerce", *Journal of Economi c Perspectives*, vol. 15, no. 1, 2001, pp. 69-80.
1. S. Morasca, "Software Measurements" in Handbook of Software Engineering and Knowledge Engineering 1, World Scientific Publishing Co. Pvt. Ltd, 2001.
8. C. Ebert, "Understanding the Product Life Cycle: Four Key Requirements Engineering Techniques", *IEc" E Sof tware*, vol. 23, no. 3, pp. 19-25, May 2006 .
9. Richard R. Maiti, Frank J. Mitropoulos, "Capturing eliciting predicting and prioritizing (CEPP) non-functional requirements metadata during the early stages of agile software development", *Southeast Con 2015*, pp. 1-8, 2015, ISSN 1558-058X.
10. Abubaker Wahaballa, Osman Wahballa, Majdi Abdellatief, Hu Xiong, Zhlguang Qin, "Toward unified DevOps model", *Software Engineering and Service Science {ICSESS} 2015 6th I EEE InternationalConference on*, pp. 211-214, 2015, ISSN 2327-0594.
11. Sharon McGee, Des Greer, "A Software Requirements Change Source Taxonomy", *Software Engineering Advances 2009. ICS EA '09. Fourth International Conference on*, pp. 51-58, 2009.

12. Lin Shi, Qing Wang, Mingshu Li, "Learning from evolution history to predict future requirement changes", *Requirements Engineering Conference (RE) 2013 21st IEEE International*, pp. 135--144, 2013.
13. Ismail Keshta, Mahmood Niazi, Mohammad Alshayeb, "Towards Implementation of Requirements Management Specific Practices (SPI.3 and SPI.4) for Saudi Arabian Small and Medium Sized Software Development Organizations", *Access IEEE*, vol. 5, pp. 24162-24183, 2017, ISSN 2169-3536.
14. D. Anderson, M. Hills, "Query Construction Patterns in ReactJS", *Proceedings of SANER 2017.IEEE*, pp. 452-456, 2017.
15. P. Klint, T. van der Storm, J. Vinju, "EASY Meta-programming with Rascal", *Post-Proceedings ofGITSE'09 ser. LNCS. Springer*, vol. 6491, pp. 222-289, 2011.
16. Barry Norton, Carlos Pedrinaci, Laurent Henocque, Mathias Kleiner, "3-Level Behavioural Models for Semantic Web Services", *International Transactions on Systems Science and Applications*, pp. 340-355, 2008.
17. B. Anda, K. Hansen, I. Gullesen, H.K. Thorsen, "Experiences from introducing UML-based development in a large safety-critical project", *Empirical Software Engineering*, vol. 11, no. 4, pp. 555-581, 2006.
18. D. K. Deeptimahanti, R. Sanyal, "Static UML model generator from analysis of requirements(SUGAR)", *Proceedings of the 2008 Advanced Software Engineering and Its Applications*, pp. 77-84, Dec. 2008.
19. K. Barker, "A logical model and schema integration architecture for determining equivalence in ER-diagrams", *Computing and Information 1992. Proceedings. JCCJ '92. Fourth International Conference on*, pp. 320-324, 1992.