## Chapter: 1

## Introduction

## Chapter: 2

## Introduction

## Chapter: 2

## Introduction

## Chapter: 2

## Introduction

## Chapter: 2

## Introduction

## Chapter: 2

## Introduction

**1.1 Background of Study**

Nowadays the concept of e-commerce is getting popular day by day. This is a form of shopping; some form of electronic communications technology is used at the offering, ordering and/or payment stage. E-commerce takes place between companies between companies and their customers, or between companies and public administrations. E-commerce is the process of managing online financial transactions by individuals and companies." Usually the key components of e-commerce are distinguished between electronic commerce between businesses (B2B), between consumers and businesses (B2C) and that between consumer each other. These three key models of E-commerce: B2B (e.g.: Walmart) is viewed as dealing with manufacturing and wholesale activity, and B2C (e.g.: Amazon) e-commerce or electronic commerce is used to describe a transaction conducted over the Internet between a business and a consumer for his/her personal use.C2C e-commerce is consist of online customer auctions, (e.g.: eBay), with providing a platform for buyer and seller to engage in the selling and purchase of retail goods.

* 1. **Objectives**

Around in world, everything is technically sophisticated. The ultimate objective of the system is to provide facility to the user for management of a company. Security of this system is very high and the possibility of doing wrong in the calculation is low. Since, now-a-days every system become increasingly technically advanced, the proposed system will make all processes involving the system much faster and easier for the users. The main objective of this system is to record all information including stocks and profit of daily monthly, which is essential to reduce the paperwork and safe time of products management. This software provides to find out which products are available which are not.

* 1. **Broad Objectives**

The broad objective of this project is to use our institutional educational experience in the real life working.

**1.4 Specific Objective**

1. Reach out to a larger audience - internet access is becoming so mainstream now that your product/service can reach almost everyone on the planet with a internet-enabled device.

2.To make an automated system that can handle update of every product information including product availability ,product quantity, product details, product selling price, Customers handling.

3. To manage organizational information

4. You build your brand more quickly - as more people will know and talk and post and blog about you on social networks.

5. For most part; setting up a website and maintaining it is lots cheaper given the place of hosting services available.

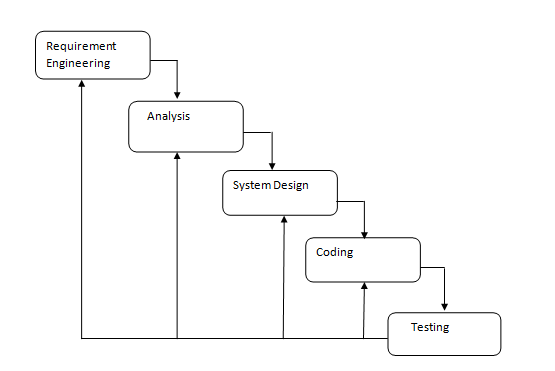
6. Reducing the wasting of time that occurred in manual system.

**1.5 System Benefits:**

* Faster and efficient processing.
* Easily insert and manage result.
* History Backup.
* Easily accessible.
* Maintain privacy

**1.6 Software Process Model**

To do this project, I have chosen waterfall model as software process model.



**Fig 1.2: Waterfall Process Model**

**Advantages of Waterfall Model:**

* This model is simple and easy to understand and use.
* It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
* In this model phases are processed and completed one at a time. Phases do not overlap.
* Waterfall model works well for smaller projects where requirements are very well understood

**Disadvantages of waterfall model:**

* Once an application is in the testing stage, it is very difficult to go back and change something that was not well-thought out in the concept stage.
* No working software is produced until late during the life cycle.
* High amounts of risk and uncertainty.
* Not a good model for complex and object-oriented projects.
* Poor model for long and ongoing projects.
* Not suitable for the projects where requirements are at a moderate to high risk of changing.

**When to use the waterfall model:**

* This model is used only when the requirements are very well known, clear and fixed.
* Product definition is stable.
* Technology is understood.
* There are no ambiguous requirements
* Ample resources with required expertise are available freely
* The project is short

**1.7. Methodology**

The development process on “E-commerce Management System” will complete following the structure described later on Software Analysis & Design. It is tentative in nature. The variables identified to manipulate through a handy inspection and from primary and secondary data

**1.8. Data Sources**

For this project in data collection phase we collected two types of data i.e.

1. Primary Data

2. Secondary Data

**Primary Data**

Primary data are generated within the organization. The organizations practical experience, observation, and face-to-face interview with our own web administrators helped us generate the primary data.

**Secondary Data**

Secondary data are collected by real life experience and studying different articles, newspapers, and research papers and of course information collected via Internet. Data, facts and statistics collected from different web sites and sources made us understand the project better.

**1.9. Feasibility Study**

Feasibility study determines whether that solution is feasible or achievable for the organization. There are three major areas of feasibility study. On studying the feasibility of the system, three major considerations are dealt with, to find whether the automation of the system is feasible.

* Technical feasibility
* Economic feasibility
* Operational feasibility

**Technical Feasibility**

For completed this project I used Notepad++ with Zen coding MySQL database system. I’ve also used HTML5 CSS and calculations are done in Ajax. And to run this project a simple configurable computer is needed. All this technology which I have mentioned above is ready to use. So I can consider this project is technically feasible.

To use the system users will require personal computer with windows operating system, Microsoft SQL Server database. All of these are available technologies now a day. So, the project is technically feasible.

**Economic Feasibility**

By using this system authority will be benefited. They save their time in managing their product. They can sell many products within short time. They need not to pay extra money for any software (except windows operating system). Their hardware is enough to run our software. So we can say it is economically feasibility.

**Operational feasibility**

User will feel comfortable to use this system. There is no complexity that can confuse the user. The proposed system is design for such type of user’s who have minimum knowledge to operate computer. So the system is operationally feasible.

## Chapter: 2

## Organizational Overview

* 1. **Organizational Overview**

DHAKA SOLUTION is a web and marketing solutions provider at its core with the highly qualified designers having experience of more than 5 years in various and complex designs. Other than our core service like web design and development DHAKA SOLUTION has satisfied the clients with the services like Mobile app Design and development, Software and Mobile Testing, SEO and Social Media Designing & Development. We are customer centric and divert our efforts to act as a one stop solution provider in the area of IT. In every area of our operations we work hard in understanding the Client’s requirement and providing the DHAKA SOLUTION made solution.

**Our Philosophy**

We firmly believe in the philosophy of ‘Our skill Our identity’. We take pride in a team of highly qualified, skilled and motivated Professionals who are encouraged to lead, innovate and excel. Our team consists of top professionals who share a common vision and passion, providing our clients with critical insights and advise to succeed in today's competitive environment. We believe in delivering Expertise, Excellence Services through our past Experience and providing the highest and best end use of services to our client.

**Our Services**

* Web Design
* Web Application
* Software Development
* **Web-Design**

At DHAKA SOLUTION, we focus on creating search engine friendly, aesthetically appealing and interactive website designs.

It is a known fact that to build a strong web presence and to secure the countless marketing opportunities available on the internet, a good website is imperative, thus triggering a race for Website Design while designing and developing your website, our professionals keep in mind key factors like easy-navigation, overall consistency and content quality, stipulated timeframes and budget. Higher for the company is to be ranked on the Search Engine result pages.

* **Software Development**

At DHAKA Info Care Limited we offers fully integrated software development and technical support solutions.

DHAKA Info Care Limited is well-experienced custom software Development Company and software outsourcing company. We have great expertise in the development of custom software applications due to our professional team efforts in performing the work according to the need of our offshore clients. We strive to focus on the customer and deliver solutions designed around their requirements rather than focusing on a specific technology and expecting the customer to adapt to the technology and platform of our choice. We use to fulfill specific needs of our clients as per their convenience for their business.

**Location :**

87, BNS Center, Level 5, Room 618,

Sector 7, Uttara, Dhaka-1230.

Bangladesh.

**2.2 Vision**

* To build a trusted IT Companies in Bangladesh
* To be the largest Software Companies in Bangladesh
* To be the largest online retailer
* To be the best choice for people when they like to Apps Development
* To be the largest SEO (search engine optimization) Marketing Company in Bangladesh
* To be the largest IT company in World.

**2.3 Mission**

Produce excellent service in the field of IT Service, Software Development, Website Design & Development, Apps Development, SEO (Search Engine Optimization), SMM (Social Media Marketing), Online Advertisement, e-database systems and banking home and abroad, E-commerce and Consultancy with maximum effort driven toward customer satisfaction.

**Our Mission at a glance:**

* To achieve maximum customer satisfaction over the entire life cycle of our customer solution via our excellence of products and solutions.
* To consistently enhance our competitiveness and deliver profitable growth.
* To practice highest standards of corporate governance and be a financially sound company.
* To be a partner in nation building and contribute towards Bangladesh economic growth.
* To encourage ideas, talent and value systems and become the customers of choice.
* To earn the trust and confidence of all customers, exceeding their expectations.
* To uphold the guiding principles of trust, integrity and transparency in all aspects of interactions and dealings. Vision to build upon a reputation of being one of the most innovative IT Solution and Service provider. We believe in doing our work in the most efficient way with robust and structured methodology, with gradual evolution from hard-work to smart- work culture, at client’s end also.

**2.4 Organization Structure:**



Figure: 2.1 Organizational Structure

## Chapter: 3

**Requirement Engineering**

**3.1 Requirement Engineering**

Requirements engineering is the process of conforming engineering designs to a set of core software requirements. This is critically important for creating accurate results in software engineering. Requirements engineering is also known as requirements analysis. Designing and building an elegant computer program that solves the wrong problem serves no one’s need. That’s why it is important to understand what the customer wants before we begin to design and build a computer-based system. Requirement engineering encompasses the tasks that lead to an understanding of what the business impact of the software will be, what the customer wants, and how end-users will interact with the software.

* User Requirements
* System Requirements
* Functional Requirements
* Non-Functional Requirements

**3.1.1 User and System Requirements:**

**User requirement#1: Admin will manage all the system in dashboard.**

**System requirement#1.1:** Admin will login the System

**System requirement#1.2:** Admin panel can fully accessed by admin

**System requirement#1.3:** Admin get some options to change the contents from admin dashboard.

**User requirement#2: Admin can manage customer through online.**

**System requirement#2.1:** Admin must login through his/her valid username and password.

**System requirement#2.2:** After login admin get a option to see **registered user**. Through this option admin can see the information about the customer.

**Systemrequirement#2.3:** If admin cancel the order request, then also user can get a notification with an explanation of what is the problem about his/her order.

**User requirement #3: Admin can update (add/delete) product.**

**System requirement #3.1:** Admin Must Login the system.

**System requirement #3.2:** After login admin get update product. Through this option admin can add or delete product.

**System requirement #3.2:** Admin can see and make all kind of product order and sell related report.

**User requirement #4: Admin can receive order request through online.**

**System requirement #4.1** Admin must login the system.

**System requirement #4.2:** After login, Admin will get Order request. Admin can see applicant details.

**System requirement #4.3:** After checking all information, Admin can confirm or cancel the order of customer.

**System requirement4.4:** If admin confirm the order request then transaction will be confirmed and customer must get a notification about their order through order tracking procedure

**User requirement #5: Admin add the category of products**.

**System requirement #5.1:** Admin add the category name of product.

**System requirement #5.2:** Enter the product name, picture, price and quantity of the products.

**System requirement #5.3:** Add the subcategory of products under category.

**User requirement #6: Admin can generate Report.**

**System requirement #6.1:** Admin can receive Customers order

**System requirement #6.2:** Admin can get order information and generate report by using customers and their order Information

**User requirement #7: Customer can do login in the system**.

**System requirement #7.1:** For login into the system user have to put his name and password on login panel

**System requirement #7.2:** If customer not sign in this system, they customer need to register the system.

**System requirement #7.3:** In register enter name, email, phone number, and password and click sign up to the system.

**System requirement #7.4**: If Customer login successfully then system will show a message “thanks for login”.

**User requirement #8: Every customer will get a separate profile.**

**System requirement #8.1:** Every customer must login through ID, username, password.

**System requirement #8.2:** After login, every customer will get his/her own profile.

**System requirement #8.3:** Customer can see only their own result and information.

**User requirement #9: Customer will view the products list**

**System requirement #9.1:** After login customer show the products category in this system

**System requirement #9.2:** Search the category of products

**System requirement #9.3:** Show the products with price and details.

**User requirement #10: Customer will give order.**

**System requirement #10.1:** Login into the System

**System requirement #10.2:** Choose the product, size, quantity

**System requirement #10.3:** If the products is stock out then show message

”stock out of products”.

**System requirement #10.4:** If product is available user can add them to cart.

**System requirement #10.5:** System will show the total bill to the customer.

**System requirement #10.6:** Enter the type of payment method.

a) Cash on Delivery

b) Bkash payment

c) Rocket Payment

**User requirement #11: Customer will give feedback on Product.**

**System requirement #11.1:** Enter customer name, mail.

**System requirement #11.2:** Customer give feedback if they have any suggestion

## 3.1.2 Functional Requirements:

A functional requirement document characterizes the usefulness of a system or one of its subsystems. It also relies on the type of software, expected users and the type of system where the software is used. The functional requirements of this system are given below.

* Add, update and delete product.
* Add customer information.
* Add category and subcategory of products.
* View product
* Add to cart
* Make order and payment
* View the customer feedback
* Search Product details.
* View Customer details with their order information.
* Calculate and process Order, Product price.
* Generate Reports

## 3.1.3 Non Functional Requirements

Non-functional requirements are not straightforward requirement of the system rather it is related to usability, security, reliability, performance etc.

* System will have secure login.
* System will need email and password information from customer and admin
* Password will be encrypted.
* System will need all the username and password stored in database.
* Usability, reliability & availability requirements must be ensured
* Give notification for every kind of mismatch information.
* Admin can manage whole management module

**3.1.4 Specification of Each Requirement:**

**3.1.4.1 Admin specification:**

**Function:** Log in, add, delete. Update product, Receive order request

**Description:** All the access of the system

**Input:** Admin input his information in his criteria.

**Output:** Information submits successfully.

**Action:** Information accepted or rejected.

**Side effects:** None

**3.1.4.2User specification:**

**Function:** Log in, add information view and order product or cancel order/

**Description:** Easily use the system for his useful purpose.

**Input:** User input his information in his criteria

**Output:** Information submits successfully.

**Action:** Request & information accepted or rejected.

**Side effects:** None

**3.1.4.3 Database specification:**

**Function:** Store whole information.

**Input**: assign data

**Output:** Progress and provide information

**Action:** Support Data

**Side effects**: none

**3.2. USE CASE Diagram**

A use case diagram is a graphic depiction of the interactions among the elements of a system. This is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.A use case diagram contains four components:

* The boundary, which defines the system of interest in relation to the world around it.
* The actors, usually individuals involved with the system defined according to their roles.
* The use cases, which the specific roles are played by the actors within and around the system.
* The relationships between and among the actors and the use cases.



**Figure 3.1** Use case diagram of ONLINE SHOPING MANAGEMENT SYSTEM

## Chapter: 4

**Project Estimation**

**4.1 Functions of Proposed System**

Table 4.1: Function of Proposed System

|  |  |
| --- | --- |
| Login the System | [F1] |
| Register into the System | [F2] |
| Customer Information | [F3] |
| View Products | [F4] |
| Make Orders | [F5] |
| Make Payment | [F6] |
| Give Feedback | [F7] |
| Mange Products | [F8] |
| Manage orders | [F9] |
| Mange users | [F10] |
| Generate Report | [F11] |

**4.1.1 Function Description**

Function description descriptive the function in details. It concerns on three factors: what is the possible input, possible output for a particular function and which table of the database uses by that function

.

1. Login into the System

* Input: Email and Password
* Output: If login data is valid then set authorization as Super admin or admin or client to the login person as defined in database otherwise will show error message.
* Use table of the database: user.

2. Register into the System

* Input: Username, email, phone, password,address,token
* Output: Successfully registered into the system.
* Use table of the database: users.

3. Customer information

* Input: Input new user information Update user information can be through this function.
* Output: The detailed record of all users and employee profile for an individual users.
* Use table of the database: Users

4. View Product

* Output: Product Name,Brand,Availability,Size,Descriptition,Price,Discount,Wishlist
* Use table of the database: Products.

5. Make Orders

* Input: Name, Email, Phone, Address, Products Name, Product Quantity, Product Size, Total Price
* Output: Confirmed time will be displayed to both client and admin.
* Use table of the database: P\_order.

6. Make Payment

* Output: Cash on Delivery,Bkash,Rocket
* Use table of the database: Pro-orders.

7. Give Feedback

* Input: Username, Email, Message.
* Output: Confirmed time will be displayed For customers.
* Use table of the database: Product\_comments.

8. Manage products

* Input:name,category,sub category,price,size,discount,quantity
* Output: List of all information of products for different user.
* Use table of the database: products

9. Manage orders

* Output: Name, Email, Phone, Description, Status.
* Use table of the database: p\_order

10. Manage users

* Output: Name, Email, Mobile Number, Address, Action.
* Use table of the database: users

11. Report generation

* Input: Generate order details report, Sells Report
* Output: Detailed orders information, Sells information will be generated
* Use table of the database: P-order, products

**4.2 System Project Planning:**

Software project management commences with a set of activities that collectively called software

project planning. Before starting any project, it is compulsory to estimate the work to be done, the resources that will be required, the time will elapse from start to finish and to analyze the project to determine whether it is feasible or not.

The following activities of software project planning that have followed in this project are:

* System Project Estimation
* Function Oriented Metrics
* Process Based Estimation
* Effort Distribution
* Task Scheduling
* Project Schedule Chart
* Cost Estimation

**4.2.1 System Project Estimation**

* The accuracy of a software project estimate predicated based on a number of things:
* Properly estimated the size of the product to build.
* The ability to translate the size estimation into human effort, calendar time and money.
* The degree to which the project plan reflects the abilities of the software team or engineer.
* The stability of the product requirements and the environment that supports the software

Engineering effort.

**4.2.2 Function Point Estimation**

Function point based estimation focuses on information domain values rather that software

values. Function points are computed by comparing five information domain characteristics. The

Five Components of Function Points-

**Data Functions**

* Internal Logical Files
* External Interface Files

**Transactional Functions**

* External Inputs
* External Outputs
* External Inquiries

**Number of external inputs** – Each user input that provides distinct application-oriented data to

the software is counted inputs should be distinguished from inquires.

**Number of external outputs** – Each user output that provides application-oriented information to the user is counted.

**Number of external inquires** – An inquiry defined as an on-line input those results in the generation of some immediate software response in the form of an on-line output. Each distinct

inquiry counted.

**Number of Internal Logical files** – Each logical internal file is a logical grouping of data that

resides within the application’s boundary and is maintained via external inputs.

**Numbers of external interfaces** – All machine-readable interfaces that used to transmit

Table 4.2: Complexity Matrix for FP Function Components

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ILF/EIF** |  | DET |  | **EI** |  | DET |  | **EO/EQ** |  | DET | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| RET | 1-19 | 20-50 | 51+ | FTR | 1-4 | 5-15 | 16+ | FTR | 1-5 | 6-19 |  | 20+ |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 | Low | Low | Avg | 0-1 | Low | Low | Avg | 0-1 | Low | Low |  | Avg |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2-5 | Low | Avg | High | 2 | Low | Avg | High | 2-3 | Low | Avg |  | High |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6+ | Avg | High | High | 3+ | Avg | High | High | 4+ | Avg | High |  | High |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

Table 4.3**.** Function Component Complexity Weight Assignment

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Low** | **Average** | **High** |
|  |  |  |  |
| External Inputs | 3 | 4 | 6 |
|  |  |  |  |
| External Outputs | 4 | 5 | 7 |
|  |  |  |  |
| External Inquiries | 3 | 4 | 6 |
|  |  |  |  |
| Internal Logical Files | 7 | 10 | 15 |
|  |  |  |  |
| External Interface Files | 5 | 7 | 10 |
|  |  |  |  |

Table 4.4**.** Function Point Count

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Transaction** |  | **FTRS** | **DETS** | **Complexity** | **UFP** |
| Login the System (EI) |  | 1 | 2 | Low | 3 |
| Register into the System (EI) |  | 1 | 8 | Low | 3 |
| Customer Information(EQ) |  | 1 | 5 | Low | 3 |
| View Product (EQ) |  | 1 | 10 | Low | 3 |
| Make Orders (EI) |  | 1 | 7 | Low | 3 |
| Make payment (EI) |  | 1 | 3 | Low | 3 |
| Remove Product from Cart (EQ) |  | 1 | 1 | Low | 3 |
| Invoice Details (EQ) |  | 1 | 7 | Low | 3 |
| Order Details (EI) |  | 1 | 5 | Low | 3 |
| Bill Calculation(ILF) |  | 1 | 2 | Low | 7 |
| Add Product (EI) |  | 1 | 12 | Low | 3 |
| Update Product(EI) |  | 1 | 12 | Low | 3 |
| Make Admin (EI) |  | 1 | 2 | Low | 3 |
| Remove Admin (EI) |  | 1 | 2 | Low | 3 |
| Update Layouts 3(EI) |  | 1 | 15 | Low | 9 |
| View Report Date wise (EQ) |  | 2 | 3 | Low | 3 |
| View Report Product Wise(EQ) |  | 2 | 5 | Average | 4 |
|  |  |  |  | Total | 62 |

**Unadjusted Function Point Contribution for Data Function**

Table 4.5 Unadjusted Function Point Contributions for Data Function

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Function | RETs | DETs | Complexity | UFP |
| Admin (ILF) | 1 | 5 | Low | 7 |
| User (ILF) | 1 | 13 | Low | 7 |
| Category (ILF) | 1 | 4 | Low | 7 |
| Contact (ILF) | 1 | 6 | Low | 7 |
| Feedback (ILF) | 1 | 6 | Low | 7 |
| Cart (ILF) | 1 | 3 | Low | 7 |
| Search (ILF) | 1 | 4 | Low | 7 |
| Product (ILF) | 1 | 15 | Low | 7 |
| Payment (ILF) | 1 | 3 | Low | 7 |
| Orders (ILF) | 1 | 16 | Low | 7 |
| Total |  |  |  | 70 |

**Performance and Environmental Impact**

**Table 4.6:** Performance and Environmental Impact

|  |  |  |
| --- | --- | --- |
| Number | Factor | Value |
| 1 | Data communications | 2 |
| 2 | Distributed data processing | 0 |
| 3 | Performance | 5 |
| 4 | Heavily used configuration | 2 |
| 5 | Transaction rate | 2 |
| 6 | Online data entry | 3 |
| 7 | End user efficiency | 5 |
| 8 | Online update | 0 |
| 9 | Complex processing | 3 |
| 10 | Reusability | 3 |
| 11 | Installation ease | 4 |
| 12 | Operation ease | 3 |
| 13 | Multiple sites | 0 |
| 14 | Facilitate change | 3 |
| **Total Degree of Influence(TDI)** | | **35** |

Value Adjustment Factor (VAF) = (0.65 + (.01xTDI))

= (0.65 + (.01x35))

= 1

UFP = UFP (Transaction Function) + UFP (Data Function)

= 62+70

=132

Adjustment Function Point Count (AFP) = UFP \* VAF

= 132\*1

= 132

Effort for PHP = AFP \* Productivity

= 132\* 15.5 [PHP productivity=15.5]

= 2046.5 person hours / 6 hours

= 341person days / 4 persons

= 85person days / 24 days

= 3.55 months

Approx. 4 months

**4.2.3 Process Based Estimation**

Software planning involves estimating how much time, effort, money, and resources will be required to build a specific software system. After the project scope is determined and the problem is decomposed into smaller problems, software managers use historical project data (as well as personal experience and intuition) to determine estimates for each. The final estimates are typically adjusted by taking project complexity and risk intoaccount. The resulting work product is called a project management plan. In process-based estimation, process is decomposed into a relatively small set of tasks and the effort required to accomplish each task is estimated. Process based estimation begins with a delineation of software functions obtained from the project scope. A series of software process activities must be performed for each function.

**Table 4.6** Process Based Estimation

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Activity | | CC | Planning | Risk  Analysis | Engineering | | Construction | | CE | Total |
| Function | 0.0 | | 0.0 | 0.0 | Analysis | Design | Code | Test | N/A |  |
| F1 | 0.0 | | 0.0 | 0.0 | 0.15 | 0.30 | 0.30 | 0.15 | N/A | 0.9 |
| F2 | 0.10 | | 0.10 | 0.05 | 0.35 | 0.25 | 0.25 | 0.20 | N/A | 1.3 |
| F3 | 0.10 | | 0.10 | 0.05 | 0.35 | 0.30 | 0.35 | 0.28 | N/A | 1.48 |
| F4 | 0.15 | | 0.15 | 0.10 | 0.45 | 0.25 | 0.30 | 0.25 | N/A | 1.65 |
| F5 | 0.15 | | 0.10 | 0.05 | 0.25 | 0.15 | 0.25 | 0.20 | N/A | 1.15 |
| F6 | 0.24 | | 0.20 | 0.15 | 0.45 | 0.25 | 0.40 | 0.28 | N/A | 1.97 |
| F7 | 0.25 | | 0.25 | 0.3 | 0.45 | 0.48 | 0.25 | 0.12 | N/A | 2.1 |
| F8 | 0.12 | | 0.15 | 0.12 | 0.40 | 0.22 | 0.28 | 0.26 | N/A | 1.55 |
| Total | 1.11 | | 1.05 | 0.82 | 2.83 | 2.2 | 2.38 | 1.74 | N/A | 12.15 |
| Effort | 9.1% | | 8.64% | 6.74% | 23.29% | 18.10% | 19.5% | 14.32% | N/A | 100% |

**4.2.4 Time Line Calculation**

Process Based Estimation = 12.15man months

Estimated time for the project = Estimated Man Month / 4.

= (12.15 / 3) months

= 4.05 ≈ 4 months need for 3 people to complete this system.

**4.2.5 Effort Distribution**

The project estimation technique leads to estimates of work units required to complete the software development. A recommended distribution of effort across the definition and development phases referred as the 40-20-40 rule. Forty percent of all effort allocated to front-end analysis and design, twenty percent allocated to coding and the remaining forty percent allocated to back-end testing. This rule used as a guideline only.

In this project, 34% of full software development has been allocated to analysis and design, 29%

has allocated to coding and the remaining 37% is allocated to software testing and support.

**4.2.6 Effort Based Estimation**

**Figure 4.1: Effort based Estimation**

**Description:**

* 1 (5% - Planning)
* 2 (25% - Customer Communication)
* 3 (20% - Analyzing)
* 4 (14% - Designing)
* 5 (29% - Coding)
* 6 (4% - Testing).
* 7 (5% - Risk analysis).

**4.2.7 Task Scheduling**

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

Duration. There are some basic rules for project scheduling. They are as follows –

**Compartmentalization** – The project must compartmentalize into a number of manageable

activities and tasks.

**Interdependency** – The interdependency of each compartmentalized activity or task must be

determined. Some tasks must occur in sequence while others can occur in parallel.

**Time allocation** – Each task to be scheduled must allocated some number of work units.

**Effort validation** – Every project has a defined number of staff members. It should ensure that no

more than the allocated number of people has scheduled at any given time.

**Defined responsibilities** – Every task that is scheduled should assign to a specific team member.

**Defined outcomes** – Every task that is scheduled should have a defined outcome. The outcome is normally a work product or a part of a work product.

**4.2.8 Project Schedule Chart**

Total system development is a combination of set of tasks. These set of tasks should done

sequentially and timely. Project schedule works as the guideline of the system developer. The

following is the schedule chart of this project:

 **Figure 4.1: Effort based Estimation**

**4.2.9 Cost Estimation**

Cost analysis represents the total cost to complete any project. In this project, there are

four factors to analyze and calculate the cost. The factors are personnel cost, software

cost, hardware cost and other cost

* Personnel cost: Personnel cost is the salary of the customer communicator, system

analyst and designer, coder and tester. For estimating the cost the analyzer used the

minimum industrial average.

* Software cost: It is the cost of the software is which used in this project
* Hardware cost: cost of the computer that used to complete the project.
* Other cost: Other cost includes the cost of the house rent, telephone bill, electricity and so on

Number of days in a year = 365

Number of government holidays in a year=24

24 Number of weekly holidays in a year =52

Total number of working days to develop the project =365-(52+24) =289 days

Total number of working days per months to develop the project =289 /12 =24.08 days

Organization’s working hours per day = 8 hours

Organization’s working hours per month=24.08 \*8= 192.64 hours

|  |  |  |
| --- | --- | --- |
| **Position** | **Salary/Month (BDT)** | **Salary /Hour (BDT)** |
| **System Analyst** | 19200 |  |
| **Designer** | 15360 | 80.00 |
| **Programmer** | 11520 | 60.00 |

Duration of the project = 4 months

Total working hours per month = 192 hours

Total working hours for the project = 192\*4 = 768 hours

**Table 4.7** Personnel Cost Estimation

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Designation | Number Of person | Working Hours | Total salary | First  Payment  at 60%  of  Salary | Remaining  20%  Payment of  Salary | Remaining  20%  Salary  Distributed  Each  Month | Total  Salary |
| System  Analyst | 1 | 190 | 19000 | 11400 | 3800 | 3800 | 19000 |
| Designer | 1 | 265 | 21200 | 12720 | 4240 | 4240 | 21200 |
| Coder | 1 | 310 | 18600 | 11160 | 3720 | 3720 | 18600 |
| Total |  |  |  |  |  |  | 58800 |

**Depreciated Hardware Cost**

The first step is to sum the digits or numbers starting with the life and going back to one. For

Example, an asset with a life of 5 would have a sum of digits as follows: 5+ 4+ 3 +2 + 1 = 15. To

Find the percentage for each year divide the year's digit by the sum. In the example above the

Percentage would be calculated as follows:

|  |  |
| --- | --- |
| Year 1 | 5 / 15 = 33.34% |
| Year 2 | 4 / 15 = 26.67% |
| Year 3 | 3 / 15 = 20 % |
| Year 4 | 2 / 15 = 13.33 % |
| Year 5 | 1/ 15 = 6.67% |

**Table 4.8**Depreciated Hardware Cost

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SL** | **Hardware** | **Number** | **Depreciation**  **Calculation** | **Depreciation**  **Expense** | **Total** |
| 1 | Laptop | 1 | 34000\*33.34% | 11334 | ((34000-11334)/48)\*4=  1888 Tk |
| 2 | Modem | 1 | 2800\*33.34% | 934 | ((2800-934)/48)\*4= 155Tk |
| 3 | Printer | 1 | 3200\*33.34% | 1066 | ((3200-1066)/48)\*4=177Tk |
| Total= 2220 Tk |

**Table 4.9**Depreciated Software Cost

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SL.** | **Software** | **Number** | **Depreciation**  **Calculation** | **Depreciation**  **Expense** | **Total** |
| 1 | Windows 7 | 1 | 12000\*33.34% | 39999 | ((12000-3999)/48)\*4  = 666Tk |
| 2 | Microsoft  Office | 1 | 8000\*33.34% | 2666 | ((8000-2666)/48)\*4  = 444 Tk |
| 3 | Xampp | 1 | free | free |  |
| 4 | Notepad++ | 1 | free | free |  |
| Total= 1110 Tk |

**Other Cost**

**Table 4.10:** Other Cost

|  |  |
| --- | --- |
| **Particular** | **Cost(for 4 Month)** |
| Office rent | 22000Tk |
| Electric Bills | 7000Tk |
| **Total** | **29000** |

**Total Cost:**

Total Cost (BDT) **=**Personal Cost+Hardware + Software +Others

= 58800+2220+1110+29000

= 91130 TK

In word: Ninety One Thousand One Hundred Thirty TK Only

## Chapter: 5

Risk Management

**5.1 Risk Management**

A risk is a potential problem that might or might not happen. It is necessary to analyze the potential risks in a project. If the risks of a software project are not properly analyzed and estimated, many problems can plague the software project. Risk analysis and management are a series of steps that help a software team to understand and manage uncertainty.

To establish a risk management model the following phases are followed:

* Risk identification is the process of detecting potential risks or hazards through data collection. A range of data collection and manipulation tools and techniques exists. The team is using both automated and manual techniques to collect data and begin to characterize potential risks to Web resources. Web crawling is one effective way to collect information about the state of Web pages and sites.
* Risk classification is the process of developing a structured model to categorize risk and fitting observable risk attributes and events into the model. The team combines quantitative and qualitative methods to characterize and classify the risks to Web pages, Web sites, and the hosting servers.
* Risk assessment is the process of defining relevant risk scenarios or sequences of events that could result in damage or loss and the probability of these events. Many sources focus on risk assessment. Rosenthal describes the characteristics of a generic standard for risk assessment as "transparent, coherent, consistent, complete, comprehensive, impartial, uniform, balanced, defensible, sustainable, flexible, and accompanied by suitable and sufficient guidance.
* Risk analysis determines the potential impact of risk patterns or scenarios, the possible extent of loss, and the direct and indirect costs of recovery. This step identifies vulnerabilities, considers the willingness of the organization to accept risk given potential consequences, and develops mitigation responses.
* Risk management implementation defines policies, procedures, and mechanisms to manage. The implemented program should balance the value of assets and the direct and indirect costs of preventing or recovering from damage or loss. To take comprehensive care of a web based system we must consider the following points
* Hardware and software environment, including any upgrades to the operating system and Web server, the installation of security patches, the removal of insecure services, use of firewalls, etc.
* Administrative procedures, such as contracting with reputable service providers, renewing domain name registration, etc.
* Network configuration and maintenance, including load balancing, traffic management, and usage monitoring.
* Backup and archiving policies and procedures, including the choice of backup media, media replacement interval, number of backups made and storage location.
* Physical location of the server and its vulnerability to fire, flood, earthquake, electric power anomalies, power interruption, temperature fluctuations, theft, and vandalism.

There are different categories of risks that should be considered in any software project. The following categories of risks have been considered in this software project:

**Project risks:** These risks threaten the project plan. If these risks become real, it is likely that the project schedule will slip and that costs will increase. Project risks identify potential budgetary, schedule, personnel, resource, customer and requirement problems and their impact on the software project.

**Technical risks:** These risks threaten the quality and timeliness of the software to be produced. If a technical risk becomes a reality, implementation may become difficult or impossible. Technical risks identify potential design, implementation, interface, verification and maintenance problems. Moreover, specification ambiguity, technical uncertainty, technical obsolescence are also risk factors.

**Business risks**: These risks threaten the viability of the software to be built. The business risks

Can be –

a. Building a system that no one really wants – market risks.

b. Building a system that no longer fits into the overall business strategy for the

Company – strategic risks.

c. Building a system whose business needs have been changed.

d. Losing the support of senior management due to a change in focus or a change in

People – management risks.

e. Losing budgetary or personnel commitment – budget risks.

**5.2 The RMMM Plan**

**5.2.1 Risk Identification**

**Table 5.1** Risk Identification

|  |  |
| --- | --- |
| **Risk Type** | **Possible Risk** |
| Technology | 1. Security of the system  2. Reusable software components may  contain defects and cannot be reused as  Planned. |
| People | 3. Key staff is ill and unavailable at critical  Times.  4. Required training for staff is not  Available. |
| Organizational | 5. Organizational financial problems force  Reductions in the project budget. |
| Requirement | 6. Changes to requirements that require  major design rework are proposed |

**5.2.2 Risk Analysis**

**Table 5.2** Risk Analysis

|  |  |  |
| --- | --- | --- |
| **Risk** | **Probability** | **Effects** |
| Organizational financial problems  force reductions in the project  budget | Low | Catastrophic |
| Security of the system. | High | Serious |
| Reusable software components  contain defects that mean they  cannot be reused as planned | Moderate | Serious |
| Changes to requirements that  require major design rework are  Proposed. | Moderate | Serious |
| Required training for staff is not  Available. | Moderate | Tolerable |
| Customers fail to understand the  Impact of requirements changes. | Moderate | Tolerable |

**5.2.3 Risk Planning**

**Table 5.3** Risk Planning

|  |  |
| --- | --- |
| **Risk** | **Risk Strategy** |
| Security | Investigate the possible security leaks and  measurements |
| Organizational financial  problems | Prepare briefing documents for senior  management showing how the project is  making a very important contribution to  the goals of business and presenting  reasons why cuts to the project budget  Would not be cost-effective. |
| Requirements problem | Alerts customer to potential difficulties  and possibility of delays; investigate  Buying in component. |
| Staff illness | Reorganize them so that there is more  overlap work and people therefore  Understand each other jobs. |
| Defective component | Replace defective potential component  with bought in component of know  Reliability. |
| Requirements changes | Replace defective potential component  with bought in component of know  Reliability. |
| Requirements changes | Derive traceability information to access  requirements change impact; maximizing  information hiding in the design |

**5.2.4 Risk Monitoring:**

* A re-planning of the project occurs. New task schedule and milestones are defined. Staffs work on their assigned jobs within the new timeframe.
* In order to prevent this from happening, the software will develop with the end user in mind.
* The user-interface will design in a way to make use of the program convenient and pleasurable.
* Meetings (formal and informal) will be held with the stakeholders regularly. This insures that the product we are producing solves a problem.
* The development cost of the software may increase by 20%.Consult with the System Analyst during the system analysis, design and testing phase of the software project.
* Proper coding grammar is followed to make sure that the codes are easily understandable and reusable.

**Table 5.4** Project Risk (P01)

|  |  |
| --- | --- |
| Project Risk (P01) | Date: 28-10-2018 |
| Name | **Insufficient Budget** |
| Probability | Moderate (35%) |
| Impact | Marginal (2) |
| Description | If the budget is low project may not complete |
| Mitigation & Monitoring | The project needs server that is costly to set-up. We find several alternative streaming services to reduce the budget risk. |
| Management | Refinement in project goal. A new plan for regulate the budget. |
| Status | Problem resolved. |

**Table 5.5** Business Risk (B01)

|  |  |
| --- | --- |
| Business Risk (B01) | Date: 5-11-2018 |
| Name | **Insufficient Budget** |
| Probability | Moderate (35%) |
| Impact | Marginal (2) |
| Description | If the budget is low project may not complete. |
| Mitigation & Monitoring | The project needs server that is costly to set-up.  We find several alternative streaming services to reduce the budget  risk. |
| Management | Refinement in project goal. A new plan for regulate the budget. |
| Status | Problem resolved. |

**Table 5.6** Business Risk (B02)

|  |  |
| --- | --- |
| **Business Risk (B02)** | Date: 12-11-2018 |
| Name | **End Users Accept System** |
| Probability | Low (15%) |
| Impact | Critical (4) |
| Description | The system fails to gain user’s faith |
| Mitigation & Monitoring | In order to prevent this from happening, the software will develop with the end user in mind. The user-interface will design in a way to make use of the program convenient and pleasurable. |
| Management | Training the users to familiarize them with the new system. Releasing patches/bug fixes for greater user satisfaction. |
| Status | The risk has not been arisen yet. |

**Table 5.7** Technical Risk (T01)

|  |  |
| --- | --- |
| **Technical Risk (T03)** | **Date: 23-11-2018** |
| Name | **Computer Crash** |
| Probability | High (60%) |
| Impact | Tolerable (3) |
| Description | Computer can be crash. |
| Mitigation & Monitoring | We should take proper follow up of computers. We also take regular Data backup every day, We can use IPS to stop unexpected shutdown. |
| Management | If our computer has been crashed then we will restore backup. |
| Status | The risk has not been faced yet. |

## Chapter: 6

**Interface Design**

**6.1 Activity Diagram:**

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system.

**Activity diagram for Admin Login**



**Figure 6.1:** Activity Diagram for Admin login

**Activity for User login**



**Figure 6.2:** Activity Diagram for User login

**Activity Diagram For Show the Product List**



**Figure 6.3:** Activity Diagram for Product list

**Activity Diagram for Buying Product**



**Figure 6.4:** Activity Diagram for buying product

**Activity Diagram for managing Category Of The product**



**Figure 6.5:** Activity Diagram for manage category

**Activity Diagram for Managing Feedback**



**Figure 6.6:** Activity Diagram for Managing Feedback

**Activity Diagram for Managing Orders**



**Figure 6.7:** Activity Diagram for Managing orders

**Activity Diagram For Generate Sales Report**



**Figure 6.8:** Activity Diagram for Generate sells report

**6.2 Swim Lane Diagram**

A swim lane diagram is a type of flowchart that delineates who does what in a process.  Using the metaphor of lanes in a pool, a swim lane diagram provides clarity and accountability by placing process steps within the horizontal or vertical “swim lanes” of a particular employee, work group. It shows connections, communication and handoffs between these lanes, and it can serve to highlight waste, redundancy and inefficiency in a process

|  |  |  |
| --- | --- | --- |
| Customer | System | Admin |
|  | Yes  No  In stock |  |

**Figure 6.9:** Swim lane Diagram for the system

**6.3 Entity Relationship Diagram:**

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation of an information system that depicts the relationships among people, objects, places, concepts or events within that system. An ERD is a data modeling technique that can help define business processes and be used as the foundation for a relational database. It provides an excellent graphical representation of the data structures and Relationship. While useful for organizing data that can be represented by a relational structure, an entity relationship diagram can't sufficiently represent semi-structured or unstructured data, and an ERD is unlikely to be helpful on its own in integrating data into a pre-existing information system. Three main components of an ERD are the entities, which are objects or concepts that can have data stored about them, the relationship between those entities, and the cardinality which defines that relationship in terms of number.



**Figure 6.10**Entity Relationship Diagram for Online Shopping Management

**6.4 Data Flow Diagram:**

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO.

**Context Level Diagram:**

A System Context Diagram (SCD) in software engineering and systems engineering is a diagram

that defines the boundary between the system, or part of a system, and its environment, showing

the entities that interact with it. Context level diagram of my system is given below-



**Figure 6.11:** Context level diagram for System

**Level 1 Diagram: (Check spelling)**



**Figure 6.11.1**Level 1 Diagram

**Level 2 Process 1 DFD Admin Login:**



**Figure 6.11.2**Level 2 DFD of Process 1

**Level 2 process 2 DFD (manage category)**



**Figure 6.11.3:** Level 2 DFD of Process 2

**Level 2 process 3 DFD (manage Product)**



**Figure 6.11.3**Level 2 DFD of Process 3

**Level 2 process 4 DFD (manage Payment)**



**Figure 6.11.4 Level** 2 DFD of Process 4

**Level 2 DFD of Process 5 (User Registration):**



**Figure 6.11.5**Level 2 DFD of Process 5

**Level 2 DFD of Process 6 (User login):**



**Figure 6.11.6**Level 2 DFD of Process 6

**Level 2 process 7 DFD (View Post)**



**Figure 6.11.7**Level 2 DFD of Process 7

**Level 2 process 8 DFD (Make Order)**



**Figure 6.11.8**Level 2 DFD of Process 8

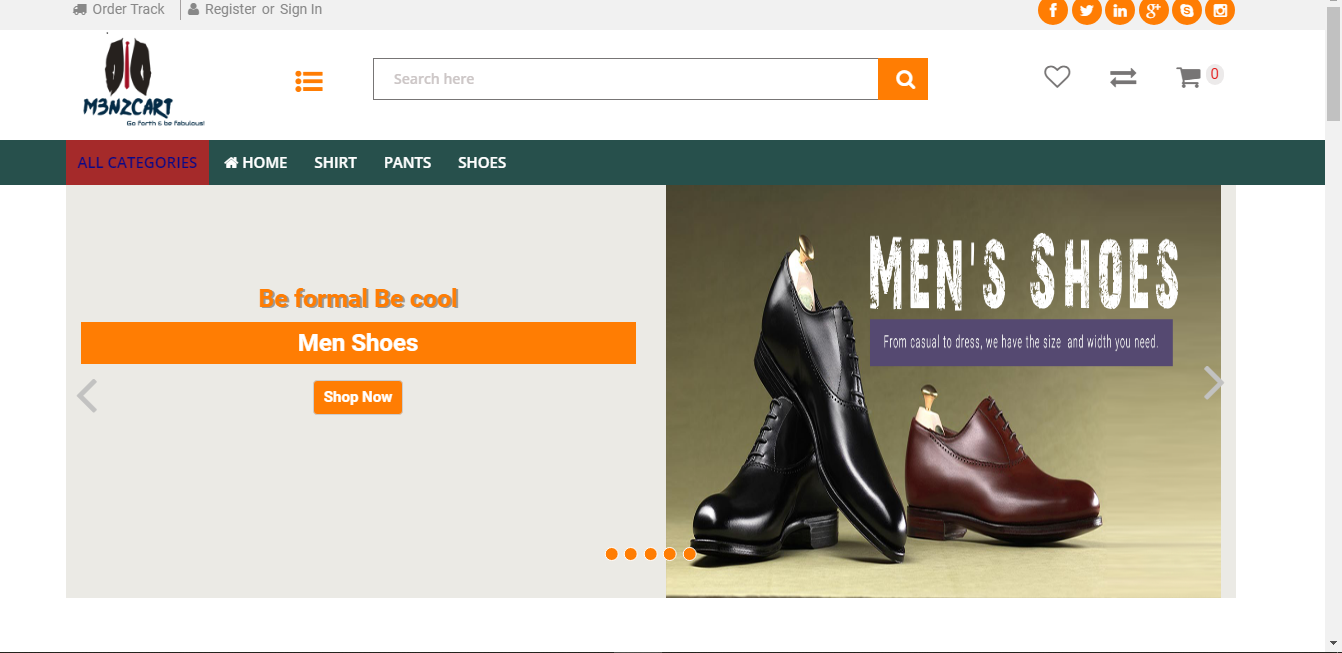
**Level 2 process 9 DFD (Manage Order)**



**Figure 6.11.9**Level 2 DFD of Process 9

**6.5 Interface Design for System**

**Home page:**

****

**Figure 6.12.1:** Home page

**Product Details**

**Figure 6.12.2:** Product Details

## Chapter: 7

**System Testing**

**7.1 System Testing:**

According to the Common Process Framework (CPF), the software testing is the final activity that has to initiate after testing. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation.

The objectives of software testing are:

* Testing is a process of executing a program with the intent of finding an error.
* A good test case is one that has a high probability of finding an as-yet-undiscovered error.
* A successful test is one that uncovers an as-yet-undiscovered error.

The design of tests for software can be challenging as the initial design of the product itself. Software can be tested in one of two ways:

* Knowing the specified function that the software has been designed to perform, tests can be conducted that demonstrate each function fully while at the same time searching for errors in each function. This approach is known as black-box testing.
* Knowing the internal workings of software, tests can be conducted to ensure that internal operations are performed according to specifications and all internal components have been adequately exercised. This approach is known as white-box testing

A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of a software. The strategy provides a road map that describes the steps to be conducted as part of testing.

Testing strategy that will be followed in this software project

* Unit testing
* Integration testing
* Validation testing

The first step in software testing is unit testing. Unit testing concentrates on each unit of the software as implemented in source code. Unit testing focuses on each component individually. The unit test is white-box oriented. Thus, unit testing of this library software will be done after completion of every module or component.

The next step is integration testing. Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective of integration testing is to take unit tested components and build a program structure that has been dictated by design.

**7.2 System Testing Methodology**

* **Black - Box Testing:**

Black-box testing which is also known as behavioral testing focuses on the functional requirements of the software. It enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black-box testing method will be applied to test the modules of LMS.



**Figure 7.1** System Testing

* **White Box Testing:**

White-box testing, which also known as glass-box testing, is a test case design method that uses the control structure of the procedural design to derived test cases. Using white-box testing methods, software engineer can derive test cases that

1. Guarantee that all independent paths within a module have been exercised at least once.
2. exercise all logical decisions on their true and false sides
3. execute all loops at their boundaries and within their operational bounds
4. Exercise internal data structures to ensure their validity.

The modules that contain some complex calculations or decision making code such as check the availability of the library item will be tested using white-box method.

**7.3 Testing Design what type of testing**

**Table 7.1** Testing Scenario No 1

|  |  |
| --- | --- |
| **Scenario** | Admin, User Login testing scenario of my system |
| **Inputs** | Email, password of admin, user for login |
| **Desired Outputs** | When enter email, password then get access level define. |
| **Actual Outputs** | For login my system work correctly |
| **Verdict** | Getting result from Desired Output’s and Actual Output’s decided this system is successful for login. |

**Table 7.2** Testing Scenario No 2

|  |  |
| --- | --- |
| **Scenario** | Admin can add category, subcategory of products. |
| **Inputs** | Category, subcategory of product info will be stored into the database. |
| **Desired Outputs** | When enter all basic info correctly, display category, subcategory of product details in the system. |
| **Actual Outputs** | For add category and subcategory info displays my system work correctly. |
| **Verdict** | Getting result from Desired Output’s and Actual Output’s decided this system is successfully added. |

**Table 7.3** Testing Scenario No 3

|  |  |
| --- | --- |
| **Scenario** | Admin can edit and delete category and subcategory details. |
| **Inputs** | Category and subcategory basic info updated and deleted by the admin. |
| **Desired Outputs** | When changed basic information for the user and products by the admin, display category, subcategory details in the system. |
| **Actual Outputs** | For update and delete category, subcategory info displays my system works correctly |
| **Verdict** | The process is worked correctly and successfully |

**Table 7.4** Testing Scenario No 4 FONT

|  |  |
| --- | --- |
| **Scenario** | Admin can add discount for product. |
| **Inputs** | Discount info will be stored into the database. |
| **Desired Outputs** | When user view product it will display discount details in the system. |
| **Actual Outputs** | For viewing my system work correctly |
| **Verdict** | Getting result from Desired Output’s and Actual Output’s decided this system is successful for viewing. |

**Table 7.5** Testing Scenario No 5

|  |  |
| --- | --- |
| **Scenario** | Admin can edit and delete product order. |
| **Inputs** | Post basic info updated and deleted by the admin. |
| **Desired Outputs** | When changed basic information for the suppliers and products by admin, display post details in the system. |
| **Actual Outputs** | For update and delete post info displays my system works correctly |
| **Verdict** | The process is worked correctly and successfully |

**Table 7.6** Testing Scenario No 6

|  |  |
| --- | --- |
| **Scenario** | User can write review for a product. |
| **Inputs** | Comment info will be stored into the database. |
| **Desired Outputs** | User will view their comments |
| **Actual Outputs** | For writing comment for a product displays my system work correctly |
| **Verdict** | The process is worked correctly and successfully. |

**Table 7.7**Testing Scenario No 7

|  |  |
| --- | --- |
| **Scenario** | User can place an order. |
| **Inputs** | Order info will be stored into the database. |
| **Desired Outputs** | When enter all basic info correctly, display pending order detail in the system. |
| **Actual Outputs** | For place order info displays my system work correctly. |
| **Verdict** | Getting result from Desired Output’s and Actual Output’s decided this system is successful for login. |

**Table 7.8**Testing Scenario No 8

|  |  |
| --- | --- |
| **Scenario** | Admin can manage orders |
| **Inputs** | Order that placed by user can be confirmed or cancelled by admin. |
| **Desired Outputs** | When changed basic information for orders by admin user can view the time of confirmation |
| **Actual Outputs** | For confirm and cancel info displays my system works correctly. |
| **Verdict** | The process is worked correctly and successfully |

**Table 7.9** Testing Scenario No 9

|  |  |
| --- | --- |
| **Scenario** | Admin can Create invoice for user orders |
| **Inputs** | Order that placed by user can be confirmed admin and then admin can create an invoice of order and print it |
| **Desired Outputs** | When Confirmed the orders admin can see the user and their order details and print the ordering invoice with these order id |
| **Actual Outputs** | For creating invoice my system works correctly. |
| **Verdict** | The process is worked correctly and successfully |

**Table 7.10** Testing Scenario No 10

|  |  |
| --- | --- |
| **Scenario** | Users Can track their order |
| **Inputs** | Order that placed by user can be confirm and rejected by admin and then admin.by using order tracking users can know about situation of their order |
| **Desired Outputs** | Customer can get immediate updating about their order. can know is their order is accepted or rejected. |
| **Actual Outputs** | For order tracking my system works correctly. |
| **Verdict** | The process is worked correctly and successfully |

## Chapter: 8

**Conclusion**

**8.1 Practicum & Its Value**

The practicum was the first time I felt like I had the opportunity to explore the Practical field of my studies. It helps me to improve my professional skills. This field teach me to communicating with different types of people and encountering situations gives practical orientation to life. As a Student of College of Engineering and Technology (CEAT) at International University of Business Agriculture and Technology go for this practicum program carrying 9 credit hours weight, which goes for a semester long and usually after the completion of the course work.

**8.2 The Experience**

Working in this project was a big opportunity for me. I have learned a lot about the technical and

Learning Environment of Dhaka solution Ltd where I have done my internship. While doing my internship I learned how to work under pressure, requirement collection of real life project, how to do the analysis of a project, testing, quality ensuring.

**8.3 Impact of Project**

I desire this project will bring a positive platform for OnlineShopping Management system. This will bring a new platform for people also. They will buy their product easily by using this platform that also gives the seller an opportunity to sell their goods and make profit and maintaining whole selling and buying process easily and securely.

Software will manage the online shop.

* Software will generate bill of the product with additional charge.
* Ordering various product is possible from anywhere in Bangladesh.
* Notify admin about the new order.
* Notify registered customer about progress of their order

Developing this software was great opportunity for me. It helps me to learn a lot which was not possible without the continuous supervision of my supervisor.

**8.4 Future Work**

This is a web based application developed with a view to providing with the facility of online gift shop system along with safe and secure online shop. As an initial version there are still a lot of limitations that will be overcoming over the time. As of now this system only provides some

Categories of product and also product color and size cannot insert completely, that facility will be introduced to this system. Currently this system only provides Cash on Delivery (COD) Bkash and Rocket payment facility in a manual way. This process will be automated soon and other payment methods like internet banking and credit card system will be introduced as well.

**8.5 Conclusion**

“Online ShoppingManagement System” application develop for human being to achieve

maximum efficiency in shopping online and to reduce the time taken to purchase items. It is

designed for people to shop online rather than searching for things by visiting the shop. It is a

365/24 service. The biggest experience working at Dhaka Solution is indeed being a part of designing and implementing software. My most experience was round the designing issue. I have learnt a lot of new things which was so much unknown to us. By this application system uses HTML, CSS, Java Script, PHP and jQuery as front end and MySQL as back end for the database. The system is strong enough to withstand regressive daily operations under conditions where the database is maintained and cleared over a certain time of span. The implementation of the system in the organization will considerably reduce data entry, time and also provide readily calculated reports. To provide better and uninterrupted services to the customers, a group of engineers at Dhaka solution is relentlessly working to keep the services towards customers. It was a wonderful experience working with them. My earnest thanks, gratitude and salutations to these wonderful people from the deep down inside my heart.

## Chapter: 9

**References**

**References**

1. Pressman, Roger S. and Maxim Bruce R. (2004). *Software Engineering: A Practitioner’s*

*Approach. 5th edition*. Boston: McGraw Hill.

2. Silberschattz, Abraham, Korth, Henry F and Sudrashan S. (2002). *Database System*

*Concepts*. *4th edition.* Boston: McGraw Hill.

3. Kendall, Kenneth E and Kendall, Jullie E. (1999) *System Analysis and Design. 4thedition.*

New Delhi: Prentice Hall.

4. O’Brien, James A and M. Markas, George. (2008). *Management Information Systems.*

Boston: McGraw Hill.

5. Freelancer. (2016). Feasibility analysis matrix jobs.

Available at: http://www.freelancer.com.bd/projects/Project-Management/Feasibility-

Analysis-Matrix.html (Accessed 29 May, 2018).

6. Slide Share. (2015).Risk management, mitigation and monitoring.

Available at: http://www.slideshare.net/naparnanayak/rmmmrisk-managementmitigationand-

monitoring (Accessed 6 June, 2018).

7. Jacurtis Alexander. (2016). Depreciation Calculations.

Available at: http://ccba.jsu.edu/accounting/DEPRECIATION.HTML (Accessed 8 June,

2018).

8. Kwak, Y.H. and Stoddard, Jared. (2015). Project risk management.

Available at: http://home.gwu.edu/~kwak/Project\_Risk\_Management.pdf (Accessed 18 June,

2018).

9. Softwaremetrices. (2014). Introduction to Function Point Analysis.

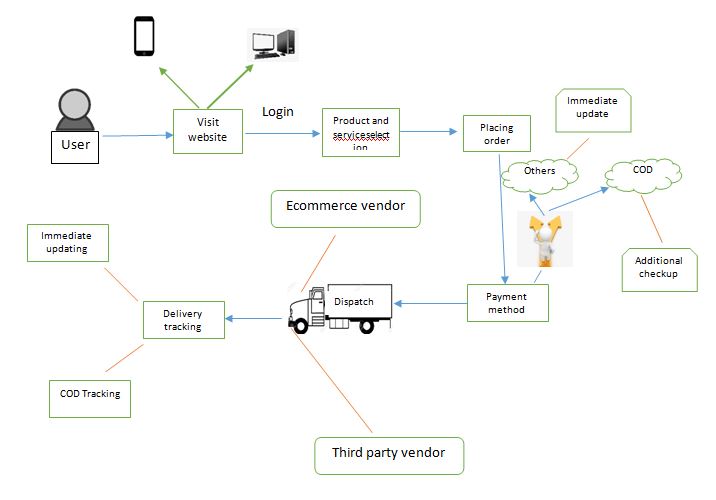
Available at: http://www.softwaremetrics.com/fpafund.htm (Accessed: 1 July, 2018).

10. Jonathan Prabhat. (2007). Estimation Techniques: Function Point Analysis

Available at: http://geekswithblogs.net/Prabhats/archive/2007/03/01/107632.aspx (Accessed:

18 July, 2018).

**Business Logic Diagram**

****