## 

## SWAMI VIVEKANAND INSTITUTE OF MODERN SCIENCE

## *(College code-264)*

## BACHELOR OF COMPUTER APPLICATION

**PROJECT REPORT**

**ON**

**ONLINE GARMENT STORE**

**SUBMITTED BY**

*SAYANTANI GANGULY REGISTRATION NUMBER:-162641O10044 OF 2016-2017*

*ROLL NO:-26401216017*

*BACHELOR OF COMPUTER APPLICATION*

*Under*

*MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY*

**DECLARATION**

I hereby declare that the project work tiled **“Online Garment Store”** submitted to the **“Swami Vivekananda** **Institution of Modern Science”.** College affiliated to **“Maulana Abul Kalam Azad University of Technology”.** As a partial fulfillment for the award of **“BCA”**. This is an original work done me and has been submitted to any other Institution.

Place: ***With sincere regards***

Date:

………………………………

(Sayantani Ganguly)

**ACKNOWLEDGEMENT**

This project entitled “***ONLINE GARMENT STORE*** ", though a result of own effort however; a subject of such a complicated nature could not have been addressed without the intellectual in sights.

It would not have been possible without the proper guidance of **Dr. Nandan Gupta, Director** andProf**. Gargi Gupta, Vice Principal** of **Swami Vivekananda Institute of Modern Science** who helped me immensely and revised this project thoroughly, providing details that lead to the completion of this project without any errors.

I would also like to thank our internal guide **Prof. Suvo Chakraborty**, **Head of the Department and Assistant Professor of Dept. Computer Application** and our external guide Mr**. Amar Banerjee**. Therefore I owe a great deal of my gratitude to my internal guide and external guide for their constant support and helping me to complete this project and my group members.

I am thankful to all the **Faculty Members** of Department **of Computer Application** and also the **Faculty Memberrs of SVIMS.**

I am grateful to each and every dear member of my project whose help, advise, guidance and assistance involved in completing this project.

I am also gretful to my**Parents** and **Friends** for supporting me directly during the project.

I am also grateful to Mritunjay Kumar Singh, Sohini Debnath, Mou Sarkar, Md. Arif Hussain for providing tremendous support without whose help the project would not have lasted today’s success.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

With sincere regards

**PREFACE**

This project is aimed at developing an Online Garment Store. Online shopping is a form of electronic commerce which allows consumers to directly buy goods or services from a website over the Internet using a web browser. The Online Garment Shop is an Internet based application that can be accessed throughout the Net and can be accessed by anyone who has a net connection. This application will automate the shopping of garments.

**CONTENTS**

**Chapter 1:** Introduction7

1.1: Main Objectives of the project8

**Chapter 2:** System Analysis

2.1: Existing System

2.2: Proposed System

2.3: Feasibility Study

2.3.1 Technical Feasibility

2.3.2 Operational Feasibility

2.3.3 Economic Feasibility

2.3.4 Social Feasibility

2.3.5 Market Research

2.3.6 Alternative Solution

2.3.7 Feasibility Report

2.4: Project, Category, Tools & Platform

2.5: Hardware Requirement

2.6: Software Requirement

**Chapter 3**: System Requirement and Specification

3.1: Overview

3.2: Module Description

3.3: Requirement Analysis

**Chapter 4**: System Planning and Design

4.1 Project Planning

4.2 Project Scheduling

4.3Entity –Relationship Diagram (E-R D)

4.4 Data Flow Diagrams (DFD)

4.4.1 Software Engineering Data Flow Diagram

4.4.1.1 1 Level 0 DFD

4.41.2 1 Level 1 DFD

4.4.1.3 1 Level 2 DFD

4.4.2 Schema / Database Design

**Chapter 5:** Technology Description

5.1 HTML

5.2 JavaScript

5.3 AJAX

5.4 jQuery

5.5 Microsoft SQL Server

**Chapter 6:** Coding and Outputs Screens

**Chapter 7**: Testing and Debugging Technique

7.1 Team Interaction

7.2 Test Objective

7.3 Process Overview

7.4 Testing Strategy

7.5 Unit Testing

7.6 White Box Testing

7.7 Black Box Testing

7.8 Gray Box Testing

7.9 Alpha Testing

7.10 Beta Testing

7.11 Integration Testing

7.12 Costs Estimation of the project along with cost

**Chapter 8**: Future Enhancement

**Chapter 9:** Conclusion

**Chapter 10:** Bibliography

**Chapter 11:** Reference

**Chapter 1** :

**Introduction**

This project is aimed at developing an ***“Online Garment Store”***. Online shopping is a form of electronic commerce which allows consumers to directly buy goods or services from a website over the Internet using a web browser. The Online Garment Shop is an Internet based application that can be accessed throughout the Net and can be accessed by anyone who has a net connection. This application will automate the shopping of garments.

The online garment shop provides a website for shopping garments where any user of internet can access it. In the garment shop there are three different types of categories ---- Women, Men, Kids and each category has several subcategories. User is required to login to the system and they can pay on COD. Garments will reach to the door of every customer within a week.

All of us love to shop for our friends, family and of course for ourselves. It is very joyful moment to gift someone any dress through online who stays far away from us. But in the hectic schedule all this excitement vanishes after hopping many shops to find the correct one on our choice.

Our Online Garment Shop is one of the best opportunities for those who cannot afford enough time to get their garments by hopping various shops and standing in long queues. In this website all the products are under one roof that customer can shop everything on their choice. Customer can shop garments online at any time of day or night. Our online shopping for garments also provides option to cancel the products which are added previously.

**1.1: Main Objectives of this project:**

1. An User should be able to
   * View all the products.
   * Can search the needed product via product name, category, brand and size.
   * Login via email id & password.
   * Can sign in if the account is new.
   * Can cart any product on choice.
   * Can also cancel any product if needed.
2. An admin should be able to
   * Login to the Garment Shop.
   * Add Garments.
   * Update and Remove any record.
   * View all Details.
   * Admin will view the products which are added to cart.

**Chapter 2** :

**Functional Requirements:**

This project is aimed to provide the consumers to directly buy goods or services from a website over the Internet using a web browser where he/she can buy any product on his/her choice.

Online Shopping system is basically made for providing the customer an anytime and anywhere service for shopping the garment on their choice and to get the facility of all garments under a roof. The user can easily be able to know about the product size, color, date of delivery and they can cancel any product easily which was added to the cart previously.

The consumer can easily be able to know about the products detail provided by the website and then make the choice.

The front management application is the user visits online Garment Store web portal. Admin can use the system to insert and delete data (e.g. size, color) which will update the webpage (webpage dynamic page, changing according to the data in database). Also admin can check the statistic information from the system.

**The goals of our system are:**

* To reduce the hopping of stores to find the product on choice.
* To allow brands to promote products to more people.
* To increase the customer engagement.
* To provide anytime anyplace service for the customer.
* To provide a online payment procedure as well as COD for the customers, that’s why customers can pay their purchase amount.
* To increase the profit.
* To become an area of excellence in fashion research,regionally and internationally.

**2.1: Existing System**

* In the existing system all transactions, dealings of products, purchasing of products were done manually which is time consuming.
* Reports are prepared manually as and when needed. Maintaining of reports is very tedious task.
* To buy any product user has to collect information about it either by visiting the shop or asking people which is the better one.
* There is no computer system for handling payments. All calculations are performed manually which may not be accurate always. Maintaining the record is really a tedious task.
* Any internet user can use this existing website to search for any kind of products, select particular products from a wide range of products.
* Once they make of their mind to purchase any particular thing they can place an order and make a payment throw various available payment option.

**2.2: Proposed System**

* Online shopping portal is a specific requirement of the client that integrats the buying and selling services specifically to their customers.
* Need for the new system is due to major drawbacks of existing system.
* Reports can be generated at any time within few seconds, so that manual labour is not required, and also analysis can be performed much more frequently which helps in taking decision.
* The details regarding all users, products can also be maintained as their information is very helpful and sometimes becomes a critical requirement.
* Allows user to get registered from their places and transact for the required product.
* To overcome these problems in existing system we develop “Online Garments Shop”.

**2.3: Feasibility Study:**

The feasibility of the project is analyzed in the phase and project proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

* Technical feasibility
* Operational feasibility
* Echonomical feasibility
* Social feasibility

**2.3.1 Technical Feasibility:**

This involves questions such as whether the technology needed for the system exists, how difficult it will be to build, and whether the firm has enough experience using that technology. The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs and Procedures. This can be quantified in terms of volumes of data, trends, frequency of updating etc in order to estimate if the new system will perform adequately or not.

**2.3.2 Operational Feasibility:**

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project. Since the proposed system was to help reduce the hardships encountered. In the existing manual system, the new system was considered to be operational feasible.

**2.3.3 Economical Feasibility:**

Economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system. More commonly known as cost/benefit analysis, the procedure is to determine the benefits and saving that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system.

**2.3.4 Social Feasibility:**

Determines whether the proposed system conflicts with legal requirements, [e.g. a data processing system must comply with the local data protection acts]. When an organization has either internal or external legal counsel, such reviews are typically standard. However, a project may face legal issues after completion if this factor is not at this stage. It is about the authorization.

**2.3.5 Market Research:**

A comprehensive market research identifying a need for the product. Detail all market research you carried out, listing sources of information. Justify any conclusions you have drawn from your research. Identify the potential customer base for your product, together with evidence of customer need for the product. Describe how you propose to complete with similar products on the market.

**2.3.6 Alternative Solution**

Consideration of alternative solutions should be documented. At least two alternative business or technical systems options should be considered. Details the difference between these options and the proposed system. Justify your choice of the proposed system and the reasons for rejecting the alternative options.

**2.3.7 Feasibility Report**

|  |
| --- |
| At this point, all of the planning for the project has been done and if the feasibility study has shown that the project is likely to succeed within its constraints, then it only remains for us to start the requirements analysis and thus proceed with the project. **Feasibility Study** |
| System: Online Garments Store |
|  |
| **Product** |
| The project requires a web application to be developed that will allow a website for an online garment store. |

**2.4: Project category, Tools and platform**

This project is a web-based application develop on Java-JEE technology frameworks using the following tools:

• RDBMS (Relational Data base System)

• PHP

• MySQL

**2.5 Hardware Requirement Specification:**

* Intel Pentium and Celeron class processor
* Processor Speed – 1.2 GHz or above
* RAM – 512 MB
* HDD – 40 GB
* Monitor – 14”SVGA
* Printer – Laser Printer

**2.6 Software Requirement Specification:**

**Front-end Tool: Macromedia dreamweaver**

* User friendly
* Low Cost Solution
* GUI feature
* Better designing aspects

**Back-end Tool: Xamp server**

* Security
* Portability
* Quality

**Platform:**

Windows 10.

**Chapter 3**:

**System Requirement and Specification:**

**3.1 Overview**:

This system maintains the information about Online Grocery Shop, where shopkeeper can add his products. This information is maintained through two following modules:

* **Login Module**
* **Customer Module**
* **Order Module**
* **Product Module**
* **Shipping Module**
* **Payment Module**

**3.2 Modules and Description:**

**Login Module:**

This module is for both type of users (customers and admin). In this module according to the type of user (customer or admin) the further links and operations will be provided.

**Customer Module:**

As soon as a visitor register himself as a customer, the customer can now order the garment and pay for them online.

**Order Module:**

In this module product customer order a product. This module contains all the information related to ordered item. As soon as the customer’s request is complete, all the order details are displayed to him.

**Product Module:**

In this module customer can see multiple images of one product. And also can able to see customer reviews. And related product.

**Shipping Module:**

In this module product is shipped after ordering any product by the customer within 7 days of ordering any product. If any customer cancel any product which was ordered previously then the shipping will also be cancelled.

**Payment Module:**

This is the most important module because it deals with the payment of the garments ordered in the order module. The customer can pay for the garments after receiving the item on COD or by card.

**3.3 Requirement Analysis:**

Functional Requirements are those that refer to the functionality of the system, i.e. what services it will provide to the user. Non-functional (Supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

**Functional Requirements (UML):**

The Unified Modelling Language allows the software engineer to express an analysis model using the modelling notation that is governed by a set of syntactic semantic rules. A UML system is represented using five different views that describe the system from distinctly different perspective. Each view is defined by a set of diagrams, which is as follows.

• **User Model View:**

1. This view represents the system from the user’s perspective.

2. The analysis representation describes a usage scenario from the enduser’s perspective.

**• Structural model view:**

1. In this model the data and functionality are arrived from inside the system
2. . 2. This model view models the static structures.

**• Behavioural Model View:**

• It represents the dynamic of behavioural as parts of the system, depicting the interactions of collection between various structural elements described in the user model and structural model view.

• **Implementation Model View:**

• In this the structural and behavioural as parts of the system are represented as they are to be built.

**• Environmental Model View:**

• In this the structural and behavioural aspects of the environment in which the system is to be implemented are represented.

**UML is specifically constructed through two different domains they are:**

• UML Analysis modelling, this focuses on the user model and structural model views of the system.

• UML design modelling, which focuses on the behavioural modelling, implementation modelling and environmental model views.

**Chapter 4**:

**System Planning and Design:**

**4.1 Project Planning:**

Project planning is concerned with identifying the following for every project:

• Activities

• Milestones

• Deliverables.

A plan must be drawn up to guide the development towards the project goal. A plan is drawn up at the start of a project. This plan should be used as the driver for the project. The initial plan is not static and must be modified as the project progresses.

Planning is required for development activities from specification through to delivery of the system.

**4.2 Project Scheduling:**

**Grant Chart:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Task** | **Person Responsible** | **Week 1** | **Week 2** | **Week 3** | **Week 4** | **Week 5** | **Week 6** |
| **Communication** |  |  |  |  |  |  |  |
| **Quick Plan** |  |  |  |  |  |  |  |
| **Modelling Quick Design** |  |  |  |  |  |  |  |
| **Construction of Prototype** |  |  |  |  |  |  |  |
| **Deployment, Delievery and Feedback** |  |  |  |  |  |  |  |

**4.3 Entity- Relation Diagram (E-R-D):**

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. It illustrates the logical structure of databases. At first glance an entity relationship diagram looks very much like a flowchart. It is the specialized symbols, and the meanings of those symbols, that make it unique. There are five main components of an ERD:

1. **Entities, which are represented by rectangles. An entity is an object or concept about which you want to store information:**

**Entity**

**2. Actions, which are represented by diamond shapes, show how two entities share information in the database.**

**Relation**

**3. Attributes, which are represented by ovals. A key attribute is the unique, distinguishing characteristic of the entity.**

1. **Connecting lines, solid lines that connect attributes to show the relationships of entities in the diagram.**

**5. Cardinality specifies how many instances of an entity relate to one instance of another entity. Ordinality is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinality describes the relationship as either mandatory or optional. In other words, cardinality specifies the absolute minimum number of relationships.**

**Entity**

**Relation**

**Entity**

**Entity - Relationship Diagram (E-R Diagram):**

Admin

User

**Ih**

has many

**order**

Order

has many

Order\_item

product

has one

**4.4 Data Flow Diagram (DFD):**

A graphical tool used to describe and analyse the moment of data through a system manual or automated including the process, stores of data, and delays in the system. Data Flow Diagrams are the central tool and the basis from which other components are developed. The transformation of data from input to output, through processes, may be described logically and independently of the physical components associated with the system. The DFD is also known as a data flow graph or a bubble chart. DFDs are the model of the proposed system. They clearly should show the requirements on which the new system should be built. Later during activity this is taken as the basis for drawing the system’s structure charts. The Basic Notation used to create a DFD’s are as follows:

**4.4.1 Software Engineering Data Flow Diagram:**

**4.4.2 Schema / Database Design:**

**Chapter 5**:

**Technology Description:**

**5.1 HTML**

HTML, Hypertext Mark-up Language, is the predominant mark-up language for web pages. It provides a means to describe the structure of text-based information in a document—by denoting certain text as headings, paragraphs, lists, and so on—and to supplement that text with interactive forms, embedded images, and other objects. HTML is written in the form of labels (known as tags), surrounded by angle brackets. HTML can also describe, to some degree, the appearance and semantics of a document, and can include embedded scripting language code which can affect the behaviour of web browser and other HTML processors

Hypertext Mark-up Language (HTML), the languages of the world wide web (WWW), allows users to produces Web pages that include text, graphics and pointer to other web pages (Hyperlink).

HTML is not a programming language but it is an application of ISO standard 8879, SGML (Standard Generalized Mark-up Language), but specialized to hypertext and adapted to the web. The idea behind hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point. We can navigate through the information based on our interest

**HTML** provides tags (special codes) to make the document look attractive.

**HTML** tags are not case-sensitive. Using graphics, fonts, different sizes, colour, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

**Basic HTML Tags:**

• <! -- -- > specifies comments

• <A>……. </A> Creates hypertext links

• <B>……. </B> Formats text as bold

• <BODY>…<BODY> Contains all tags and text in the HTML document

• <FRAME>…</FRAME>Defines a particular frame in a set of frames

• <FORM>…</FORM>Encloses a fill-out form

• <H#>…</H#> Creates headings of different levels (1- 6)

• <HEAD>…</HEAD>Contains tags that specify information about a document

• <HTML>…</HTML>Contains all other other HTML tags

• <META>…</META>Provides meta- information about a document

• <SCRIPT>…</SCRIPT>Contains client-side or server-side script

**Advantages**

➢ A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.

➢ HTML is platform independent.

➢ HTML tags are not case-sensitive

**5.2 Javascript**

JavaScript is a script-based programming language that was developed by Netscape communication Corporation. JavaScript was originally called live script to indicate its relationship with Java. Java script supports the development of both silent and server components of web-based applications. On the silent side, it can be used to write programs that are executed by a web browser and then update the browser’sdisplay accordingly.

**Advantages**

➢ JavaScript can be used for server-side and client-side scripting.

➢ It is more flexible than VBScript.

➢ JavaScript is the default scripting languages at Client-side since all the browsers supports it.

**5.3 AJAX**

AJAX stands for Asynchronous Java Script and XML.It is the use of the XMLHttpRequest object to communicate with server-side scripts. It can **send as well as** receive information in a verity of formats, including JSON, XML, HTML, and even text files. AJAX’S most appealing characteristic is its “asynchronous” nature, which means it can do all of this without having to refresh the page. The two major features of AJAX are:

• Make requests to the server without reloading the page • Receive and work with data from server

**Advantages:**

➢ Reduce the traffic travels between the client and the server.

➢ Response time is faster so increases performance and speed.

➢ It can use JSON which is alternative to XML. JSON is key value pair and works like an array. ➢ It can use Firefox browser with an add – on called as Firebug to debug all Ajax calls.

➢ Ready Open source JavaScript libraries available for use – jQuery, Prototype, Scriptaculous, etc.

➢ AJAX communicates over HTTP Protocol

**5.4 jQuery**

JQuery is a cross platform java scripts library designed to simplify the clientside scripting of HTML. JQuery is the most popular JavaScript library in use today, with installation on 65% of the top 10 million highest trafficked site on the web. It is free, open source software licenced under the MIT License. Its syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. It also provides capabilities for developers to create abstraction for low level interaction and animation, advanced effects and high level, theme-able widgets. The modular to the jQuery library allows the creation of powerful dynamic and web applications. The set of JQuery core features- DOM element selections, traversal and manipulating-enabled by its selector engine, create a new “Programming style”, fusing algorithm and data structures.

**Advantages:**

➢ Ease of use: This is pretty much the main advantage of using JQuery, it is a lot easier to use compared to standard JavaScript and another JavaScript libreries apart from simple syntax, it also requires much less lines of code to achieve the same feature in comparison

➢Large library: JQuery enables us to perform hordes of functions in comparison to other java scripts libraries.

➢ Strong open source community: jQuery, while relatively new has a following that religiously devote their time to develop and enhance the functionality of jQuery. Thus, there are hundreds of prewritten plugs in available for download to instantly speed up our development process. Another advantage behind this is the efficiency and security of the script.

➢ Great documentation and tutorials: The JQuery website has a comprehensive documentation and tutorials to get even an absolute beginner in programming to get the ball rolling with this library.

➢ Ajax support: JQuery support Ajax enables a sleeker interface where actions can be performed on pages without requiring to entire page to be reloaded.

**5.5 Microsoft SQL Server**

Databases help the business access information more quickly. This, in turn, allows employees to get more done in less time, which allows business to grow more rapidly. When businesses grow more rapidly, there are generally more jobs and higher pay and the economy dose better. This all sounds wonderful, doesn’t it? So then why don’t we see database helping companies and organizations progress their business faster? Productivity is main contributor, but there are many others. In fact databases are very difficult to understand on a holistic level. From programming to maintaining and from deploying to managing, databases can be tricky to keep up with when thinking about the business. In this post, we’ll take a closer look at what challenges are faced by virtualization admins who are,in one way or another, connected to Microsoft SQL Server (MSSQL) in addition to some of the advantages of using the software. The main reason Microsoft SQL Server is a favorite of developers and virtualization admins alike is its ease of use. Development and troubleshooting are typically the toughest aspects to perfect when thinking about getting a SQL project into production. MSSQL comes with excellent tools that will save you a lot time in these areas – tools like SQL Server Profiler, SQL Server management Studio, BI tools and Database Tuning Advisor. Setting up almost everything, from installing on a VM to initial Query writing and editing, is incredible easy with MSSQL – especially in comparison to other SQL products. If there are problems in any stage of development, there is a plethora of online support and Documentation in Addition to live product support, whereas the support option for other SQL products are not nearly as robust.

**MySQL Memory Management Challenges**:

Although we can always program code better to be a little (or a lot) more efficient, the real struggles with MSSQL happen when managing it in a virtual environment. MSSQL tends to “hog” the allocated memory, at least from the VM’s perspective, seemingly using 100% of its virtual memory. A common practice is to simply ignore this monitored utilization and takethe DBA’s recommendation for VM memory configuration. This only leads to inefficiency or evenperformance problems and doesn’t give any application-tier insight to the infrastructure. By giving an arbitrary amount of resource capacity to the VM running the application, resources not used are wasted. On the flip side, when the capacity isn’t big enough, application performance can suffer. Without the proper insight or visibility into the application to find out actual resource utilization, ventilation admins are often the scapegoat for when something goes wrong.

**The Finger-pointing Scenario:**

So, what if performance dose become an issue? Who’s at fault, and how do you resolve it? It the coding, the resource supply on the virtual machine or internal to the application? Without proper communication or awareness of all parties involved, it’s often difficult to pinpoint the bottleneck. This is when the finger-pointing begins. If the application’s middleware components were aware of the sizing of the underling VM infrastructure and vice versa, then there wouldn’t be any bottlenecks from a resource allocation perspective. How else is the VM supposed to know how much memory to give to the application if the database memory isn’t sized correctly which is leading into an unacceptable response time or QOS? In this situation, rather than spending timetroubleshooting, it would benefit the performance output of the application if there was a proper supply of computing resource to the consuming database server. Conversely, if the DBA had proof/justification that the underling infrastructure was free of resource contention bottlenecks, s/he would be able to troubleshoot performance issues much more quickly. Eliminating bottlenecks on the infrastructure reduces the number of places a finger can point in a degraded performance situation and enables teams to refocus their time on innovative projects for the business.

**Data definition:** Defining tables and structure in the database (DDL used to create, alter and drop schema objects such as tables and indexes).

**Data manipulation:** Used to manipulate the data within those schema objects (DML Inserting, Updating, Deleting the data, and Querying the Database). A schema is a collection of database objects that can include: tables, views, indexes and sequences List of SQL statements that can be issued against an Oracle database schema are:

• **ALTER** – Change an existing table, view or index definition (DDL)

• **AUDIT**- Track the changes made to a table (DDL)

**• COMMENT**- Add a comment to a table or column in a table (DDL)

• **COMMIT**- Make all recent changes permanent (DDL – transactional)

• **CREATE**- Create new database objects such as tables or views (DDL)

**• DELETE**- Delete rows from a database table (DML)

**• DROP**- Drop a database object such as a table, view or index (DDL)

**• GRANT**- Allow another user to access database objects such as tables or views (DDL)

**• INSERT**- Insert new data into a database table (DML)

**• No AUDIT**- Turn off the auditing function(DDL)

• **REVOKE**- Disallow a user access to database objects such as tables and views (DDL)

• **ROLLBACK**- Undo any recent changes to the database (DMLTransactional)

• **SELECT**- Retrieve data from a database table (DML)

• **TRUNCATE**- Delete all rows from a database table (cannot be rolled back)(DML)

• **UPDATE-** Change the values of some date item in a database table (DML)

**Advantages**

➢ High Speed: SQL Queries can be used to retrieve large amounts of records from a database quickly and efficiently.

➢ Well Defined Standards Exist:SQL database use long – established standard, which is being adopted by ANSI & ISO. Non – SQL database do not adhere to any clear standard.

➢ NO Coding Required:Using standard SQL it is easier to manage database systems without having to write substantial amount of code.

➢ Emergence of ORDBMS:Previously SQL database were synonymous with Relational database. With the emergence of Object Oriented DBMS, object storage capabilities are extended to relational database**.**

**Chapter 6**:

**Codding and Output Screen**

**Chapter 7**:

**Testing and Debugging Technique:**

**7.1 Team Interaction:**

The following describes the level of team interaction necessary to have a successful product.

• The Test Team will work closely with the Development Team to achieve a high quality design and admin interface specifications based on modeler requirements. The Test Team is responsible for visualizing test cases and raising quality issues and concerns during meetings to address issues early enough in the development cycle.

• Since the application interacts with a back-end system component, the Test Team will need to include a plan for integration testing. Integration testing must be executed successfully prior to system testing.

**7.2 Test Objective:**

The objective our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. We will be testing a Binary Search Tree Application utilizing a pre-order traversal format. There will be eight key functions used to manage our application: load, store, clear, search, insert, delete, list in ascending order, and list in descending order. Our admin interface to utilize these functions is designed to be admin-friendly and provide easy manipulation of the tree. The application will only be used as a demonstration tool, but we would like to ensure that it could be run from a variety of platforms with little impact on performance or usability**.**

**7.3 Process Overview:**

The following represents the overall flow of the testing process:

1. Identify the requirements to be tested. All test cases shall be derived using the current Program Specification.

2. Identify which particular test(s) will be used to test each module.

3. Review the test data and test cases to ensure that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit.

4. Identify the expected results for each test.

5. Document the test case configuration, test data, and expected results.

6. Perform the test(s).

7. Document the test data, test cases, and test configuration used during the testing process. This information shall be submitted via the Unit/System Test Report (STR).

8. Successful unit testing is required before the unit is eligible for component integration/system testing.

9. Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, its possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.

10. Test documents and reports shall be submitted. Any specifications to be reviewed, revised, or updated shall be handled immediately.

**7.4 Testing Strategy:**

The following outlines that types of testing that will be done for unit, integration and system testing. While it includes what be tested, the specific use cases that determines how the testing is done will be detailed in the Test Design Document. The test cases that will be used for changing use cases in shown and onwards.

**7.5 Unit Testing:**

Unit Testing is done at the source or code level for language-specific programming errors such as bad syntax, logic errors, or to test particular functions or code modules. The unit test cases shall be designed to test the validity of the programs correctness.

**7.6 White box Testing:**

**White-box testing** (also known as **clear box testing**, **glass box testing**, **transparent box testing**, and **structural testing**) is a method of testing [software](https://en.wikipedia.org/wiki/Software) that tests internal structures or workings of an application, as opposed to its functionality (i.e. [black-box testing](https://en.wikipedia.org/wiki/Black-box_testing)). In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the expected outputs. This is analogous to testing nodes in a circuit, e.g. [in-circuit testing](https://en.wikipedia.org/wiki/In-circuit_test) (ICT). White-box testing can be applied at the [unit](https://en.wikipedia.org/wiki/Unit_testing), [integration](https://en.wikipedia.org/wiki/Integration_testing) and [system](https://en.wikipedia.org/wiki/System_testing) levels of the [software testing](https://en.wikipedia.org/wiki/Software_testing)process. Although traditional testers tended to think of white-box testing as being done at the unit level, it is used for integration and system testing more frequently today. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it has the potential to miss unimplemented parts of the specification or missing requirements

**7.7 Black Box Testing:**

Black-box testing is a method of [software testing](https://en.wikipedia.org/wiki/Software_testing) that examines the functionality of an application without peering into its internal structures or workings. This method of test can be applied virtually to every level of software testing: [unit](https://en.wikipedia.org/wiki/Unit_test), [integration](https://en.wikipedia.org/wiki/Integration_testing), [system](https://en.wikipedia.org/wiki/System_testing) and [acceptance](https://en.wikipedia.org/wiki/Acceptance_test). It is sometimes referred to as specification-based testing..

**7.8 Gray Box Testing:**

Gray-box testing is a combination of white-box testing and black-box testing. The aim of this testing is to search for defects if ant due to improper structure or improper usage of application.

**7.9 Alpha Testing:**

This is a form of internal acceptance testing performed mainly by the in-house software QA and testing teams. Alpha testing is the last testing done by the test teams at the development site after the acceptance testing and before releasing the software for Beta Testing.

Alpha testing can also be done by the potential users or customers of the application. But still, this is a form of in-house acceptance testing.

**7.10 Beta Testing:**

This is a testing stage followed by the internal full alpha test cycle. This is the final testing phase where the companies release the software to few external user groups outside the company test terms or employees. This initial software version is known as the beta version.

**7.11 Integration Testing:**

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group .It occurs after unit testing and before validation testing . Integration testing takes as its input modules that have been unit tested , groups them in larger as its output the integrated system ready for system testing. The purpose of integration testing is to verify functional , performance ,and reliability requirements placed on major design items. These “design items”, group of units ,are exercised through their interfaces using black-box testing . success and error cases being simulated via appropriate parameter and data

**7.12 Cost Estimation of the Project along with Cost** **Estimation Model:**

**Analogous estimate of effort or cost**

Used for Early Estimate or Individual Activity Estimate

Sample example shown below is for two major deliverables of a software project. You use a previous project as a benchmark for analogous estimation. Using your experience you will estimate a multiplier.

Multipliers:

1. Prototyping: 0.75.
2. Testing: 0.5
3. Deployment: 0.5

Finally, if you want to convert to cost, you would use current rates for the resources.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **WBS** | **Previous** | **Previous** | **Current** | **Multiple** | **Effort** | **Cost** |
| **ID** | **Similar Project Activity** | **Effort** | **Project Estimate** | **R** | **(Previous Effort \* 0.75)** | **(Rs. 500/hr)** |
| 1 | Prototyping | 40 Work-Hours | Prototyping | 0.75 | 30 Work Hours | Rs.  15000/- |
| 2 | Testing | 20 Work-Hours | Testing | 0.50 | 10 Work-Hours | Rs. 5000/- |
| **Total** |  |  |  |  | **40 Work-Hours** | **Rs. 20000/-** |

**Chapter 8**:

**Future Enhancement:**

The project had covered almost all the requirements. Further requirement and improvements can easily done since the coding is mainly structured or modular in nature. Improvements can be appended by changing the existing modules or adding new modules. One important development that can be added to the project in future is SMS verification.

* The size of the database increases day-by-day, increasing the load on database backup and data maintenance activity**.**
* We need to add the Return Policy and Refund System.
* We need to add e-banking system to make it more user friendly.
* Training needed for using the server side of this project.
* We need to improve the GUI of this application to make more attractive to the user.
* **Chapter 9**:

**Conclusion:**

Our project is only a humble venture to satisfy the needs to manage their project work. Several user friendly coding have also adopted. This package shall prove to be a powerful package in satisfying all the requirements of the school. The objective of software planning is to provide a frame work that enables the manger to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses. Made statement of the aims and objectives of the project. The description of Purpose, Scope, and applicability. We define the problem on which we are working in the project. We describe the requirement Specifications of the system and the actions that can be done on these things. We understand the problem domain and produce a model of the system, which describes operations that can be performed on the system. We included features and operations in detail, including screen layouts. We designed user interface and security issues related to system. Finally the system is implemented and tested according to test cases.

**Chapter 10**:

**Bibliography:**

1. Learning PHP, MySQL & JavaScript: With JQuery, CSS & HTML5,

Book by Robin Nixon

2. PHP and MySQL Web Development, Book by Luke Welling

3. The Joy of PHP: A Beginner's Guide to Programming Interactive Web ...Book

by Alan Forbes

4. jQuery in Action,Book by Bear Bibeault

5. PHP for the Web: Visual QuickStart Guide, Book by Larry Ullman

6. Head First JQuery, Book by Ronan Cranley and Ryan Benedetti

7. JQuery Mobile: Up and Running, Book by Maximiliano Firtman

8. Creating Mobile Apps with jQuery Mobile\, Book by Shane Gliser

9. Pro PHP and JQuery, Book by Jason Lengstorf

10. HTML5, JavaScript, and JQuery 24-Hour Trainer, Book by Dane

Cameron

11. High Performance MySQL, Book by Baron Schwartz, Derek J.Balling,

Jeremy Zawodny, and Peter Zaitsev

12. Beyond JQuery, Book by Ray Nicholus

13. Pro PHP Security, Book by Chris Snyder and Michael Southwell

14. A Web‑based Introduction to Programming: Essential Algorithms, Syntax and

Book by Mike O'Kane

15. Web Developer's Cookbook, Book by Robin Nixon

16. Object-Oriented PHP, Book by Peter Lavin

17. Database Systems: Models, Languages, Design, and Application

Programming, Book by Ramez Elmasri and Shamkant Navath

18. JQuery Essentials, Book by Troy Miles

19. Mastering jQuery, Book by Alex Libby

20. A Smarter Way to Learn JQuery: Learn It Faster, Remember It Longer,

Book by Mark Myers

21. A Smarter Way to Learn JavaScript: The New Approach That Uses,

Book by Mark Myers

22. Learning JQuery 3 - Fifth Edition, Book by Adam Boduch, Jonathan

**Chapter 11**:

**References:**

1. https://www.w3schools.com/js/tryit.asp?filename=tryjs\_myfirst
2. https://www.w3schools.com/jquery/tryit.asp?filename=tryjqueryhide
3. https://www.w3schools.com/jquery/jquery\_ref\_selectors.asp
4. https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_fadein
5. https://www.w3schools.com/jquery/jquery\_syntax.asp
6. https://www.w3schools.com/jquery/jquery\_get\_started.asp
7. https://www.w3schools.com/jquery/jquery\_selectors.asp
8. https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_lib\_microsoft
9. https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_lib\_google

10. https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_lib\_google

11. https://stackoverflow.com/questions/50129967/laravel-jquery-ajax-file-upload

12. https://stackoverflow.com/questions/50129760/how-to-insert-data-from-array-into-mysql-using-php

13. https://stackoverflow.com/questions/50129430/how-can-i-firebase-database-reference-sending-other-page

14. https://stackoverflow.com/questions/50129421/how-to-fetch-value-from-mysql-table-using-multiple-condition-with-optional-value

15. https://stackoverflow.com/questions/50130283/pass-the-value-from-the-current- page-display-those-values-in-new-tab

16. https://stackoverflow.com/questions/50130163/datatable-class-id-is-not-getting-called-from-js-into-module-file

17. https://www.w3schools.com/php/showphp.asp?filename=demo\_form\_validation

18. https://www.w3schools.com/php/showphp.asp?filename=demo\_filter1

19. https://www.w3schools.com/php/showphp.asp?filename=demo\_session1

20. https://www.w3schools.com/php/php\_file\_upload.asp

21. https://www.w3schools.com/php/php\_datatypes.asp

22. https://www.w3schools.com/php/showphp.asp?filename=demo\_form\_validation\_

Complete

23. https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_hide\_p

24. https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery\_slide\_down

25. https://github.com/laravel/framework/blob/5.6/.github/ISSUE\_TEMPLATE.md

26. https://stackoverflow.com/questions/50130280/sort-array-by-name-php

27. https://stackoverflow.com/questions/50129421/how-to-fetch-value-from-mysql-table-using-multiple-condition-with-optional-value

28. https://stackoverflow.com/questions/50129260/create-pairs-with-2-array-so- there-is-no-repeated-pairs

29. <https://stackoverflow.com/questions/50129789/editing-the-contents-of-a-dwitha-dynamic-form>

30. https://stackoverflow.com/questions/50129484/redirect-with-mvc-and-https