PROJECT REPORT

ON

"Online Shopping Cart"

Submitted To: -
Ejaz Hussan

Submitted By:-
Abdul Hanan

Roll No. F1511461

ABSTRACT

This is a project report on "**ONLINE SHOPPING**". During the making/developing of this project we explored new ideas and functionality behind the working of a notepad.

This project is the output of our planning, schedule, programming skill and the hard work, and this report reflects our steps taken at various levels of programming skill, planning and schedule.

We have learnt a lot during this project and liked the improvement in our testing skills and deep concept related to these kinds of projects.

Our project is **ONLINE SHOPPING CART**. This is a web based application which helps people to find and buy latest fashion clothes, mobiles on internet. It is useful in the way that it makes an easier way to buy different product online.

In this application we have basically 2 modules. The first module includes the customer module.

The customer has to register for any enquiry related to clothes. The unregistered person can't make a order but view the products.

The admin module contains the access of admin on the application. The admin can change everything in the application. He has the ability to add, delete, update any information regarding the products.

CONTENTS ***PAGE NO***

PROJECT DESCRIPTION

In today's busy world, people don't have time for their personal needs. And the technology is so fast that anyone can do anything by just sitting in a room. The internet is the way that helps a person in all aspects. If someone wish to buy and view things, he can buy online with the help of internet.

Today there are very least organizations which are manual. Everything is going to be computerized and online whether it is banking, advertising or shopping. We are trying to help people to make their life easier by proving online clothes shopping.

In this we have introduced many modules like admin module and customer module. The unregistered person can't make the order successfully. The unregistered/register customer can view details of different products.

The admin module contains the access of admin on the application. The admin can change everything in the application. He has the ability to add, delete, update any information.

COMPONENTS OF THE PROJECTS

- **How to Login**

  In this module, the user will enter his username and password to order, view and buy latest products. There will be 2 types of users Administrator/Customer

- **How to be a member of this application**

  In this site, the candidate can join this application, if he is not a member yet by pressing sign up link. User should provide some details that are asked to join.

- **How to give order**

  First the user has to login, and then he will order the products. There he will select the products he wants to buy and then he will click on show products. There he can purchase that product.

- **How to pay money**

  The user can PayPal accounts.

REQUIREMENT ANALYSIS

Hardware Requirements

Number	Description
1	Intel core ,WIN xp/7/vista
2	780 MB RAM

Software Requirements

Number	Description
1	Windows XP –7
2	Php
3	MySQL
4	IIS server/ WAMPSERVER
5	HTML/Bootstrap /Ajax/JavaScript/Css

MODULES

Login Module –In this module user can enter the application by providing username and password and start shopping.

Admin Module – Admin can add, modify and delete the latest verities of clothes.

Join Module – In this module user can become a part of the site by providing some necessary information for example first name, last name, password, confirm password, email and other details.

Shopping Module – The customer can view and buy latest verities of product.

Administrator – **Can add, modify, and delete the cloth details.**

INTRODUCTION TO TOOLS

FRONT-END/BACK-END

Front-End: **Web Pages using HTML, Css , JavaScript etc.**

Back-End: **MYSQLi, PHP**

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

Required website is to provide online details to the customers of the specified products. The system should satisfy the following requirements:

General Aspects:

Authenticate User

- Show Products and their details with type Website should be able to register new user.
- Analysis: Authenticate user based on username & password. Keeping session track of user activity.
- Maintaining the record of products.

External Interface Requirements:

Hardware Interface

- Server side Hardware
- Hardware Recommended by the all the software needed
- Communication hardware to server client request.
- Client side Hardware
- Hardware recommended by respective client operating system & Web browser.
- Communication hardware to communicate the server.

Software Interface

- Server side software
- Web-Server software
- Server side scripting tools
- Database tools
- Compatible operating system

Client side software

Web browser supporting machines

Non Functional Requirements

- System should be able to handle multiple users.
- Login by username, password should be incorporated wherever necessary
- Should be user friendly and display easy to understand error messages

Reliability

Data validation & verification need to be done at every stage of activity. Validation user input

Availability

- The web application should be available anywhere and anytime.
- User Session should timeout after 20 minutes of inactivity.

Performance

- The system will be used by multiple users and may grow as time passes.
- Necessary measures need to be taken to make the system as fast as possible.

Software System Attributes

1. Usability: The links are provided for each form. The user is facilitated to view and make entries in the forms. Validations are provided in each field to avoid inconsistent or invalid entry in the databases. Some forms consist Hyper Links, which provides further details.

2. Security: Application will allow only valid users to access the system. Access to any application resource will depend upon user's designation. Security is based upon the individual username and password.

3. Maintainability: The products detail will be easily available for the user.

4. Availability: System will be available around the clock except for the time required for the backup of data.

Acceptance Criteria

- A user-friendly interface with proper menus.
- Data transfer should be accurate and within a reasonable amount of time keeping in mind the network traffic.
- The system should not allow entry of duplicate key values.
- System should have the ability to generate transactional Logs to avoid any accidental loss of data.

Aims and Objective

The main purpose behind the proposed system is to provide a comprehensive computerized system, which can capture, collate and analyze the data and evaluate the impact of the program.

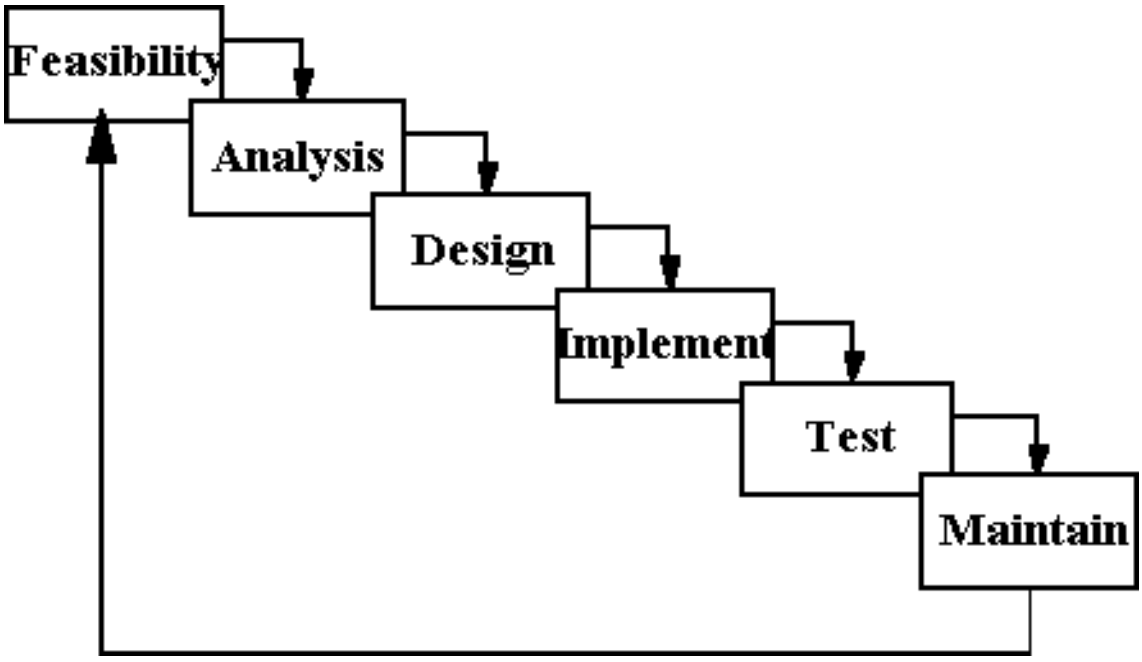
Characteristics of the proposed system

The web application has following features:

- In comparison to the present system the proposed system will be less time consuming and is more efficient.
- Analysis will be very easy in proposed system as it is automated.
- The proposed system is very secure as no chances of loss of data as it is dependent on the administrator only.

SYSTEM DEVELPOMENT LIFE CYCLE (SDLC)

The Systems Development Life Cycle **(SDLC)** is a conceptual model used in project management that describes the stages involved in an information system development project from an initial feasibility study through maintenance of the completed application. Various SDLC methodologies have been developed to guide the processes involved including the waterfall model (the original SDLC method). Documentation is crucial regardless of the type of model chosen or devised for any application, and is usually done in parallel with the development process. Some methods work better for specific types of projects, but in the final analysis, the most important factor for the success of a project may be how closely particular plan was followed.



PROBLEM ANALYSIS

The main applications of the On Line Placement System is the ability of the website to properly show enroll the artists and manage information about them. The administrator has the ability to change, modify, view and delete the various details regarding the users and arts. The users have the ability to log in and post their queries and download arts.

Challenges

The challenges mainly lie in detecting attacks like viruses, hacking and also in the implementation of firewall. A virus can enter the system and can disrupt the working of the website. Hacking can be done by some people who want to access some restricted sections of the website (e.g. administrator's area) and to modify or taper some aspects of the website.

Scanning attacks may yield:

(i) The method used by viruses to enter the system.

(ii) The types of database allowed through a firewall.

(iii) The paths or ways used by hackers to enter the system

(iv) The loopholes remaining in the system (or website) which are used by attackers.

(v) The server from where the viruses or hackers are gaining access to the system.

(vi) The types of viruses able to affect the website.

REQUIREMENT ANALYSIS

Goal of Thesis

The goal of project is to develop a website that can be used as an online shopping website with the features of interaction and problem solving. The whole project will be based on PHP with MYSQL as the database with certain security constraints added to it.

Our aim is also to implement the Administrator part in to the project so that the server or administrator himself can view, add, delete and modify products information.

A. Administrator

He has to see whether the website is working properly and whether the details available in the system are relevant and correct. He can view, add, modify, delete details.

B. Database

The database keeps all the records of all the users i.e. name, course, phone no.,dob, city, country etc. For creating such records it takes the help of tables which is created in the MYSQL. The tables can have infinite entries of all the registered users as well as administrators.

C. Clients

Our aim will also to provide efficient way by which client can enter to see his profile, ask questions to his teacher and download softwares.

D. Security Constraints

There need to be certain constraints which have to be implemented on the database

as well as on the administrator in order to work properly the whole system, such as declaring the primary key, or such constraints in order to keep the database work properly.

Some of such constraints are as follows:

1. Each user has a field called username which can be used to differentiate between different users.

2. The administrator has his own password known to no one else to access a unique page.

3. Only the administrator has the right to delete, modify users and questions.

4. Administrator has to first login to get access to my account part and modify some data available in the website.

5. A new user cannot have same username and password as of some already registered user.

6. When user or administrator click on logout the session of the user ends and he has to again login using his/her username and password to access the same functionality.

7. Checks are implemented so that the mandatory fields are filled by users when the user is entering some information in the system.

FEASIBILTY STUDY

From the inception of ideas for software system, until it is implemented and delivered to customer and even after that the system undergoes gradual developments and evaluations.

- The software is said to have life cycle composed of several phases.
- At the feasibility stage, it is desirable that two or three different configuration will be pursed that satisfy the key technical requirement but which represent different level of ambition and cost.
- Feasibility is the determination of whether or not a project is worth doing. A feasibility study is carried out select a best system that mate performance requirements.
- The data collected during primary investigation examines system feasibilities that is likelihood that the system will be beneficial to the organization. Four tests for feasibility study are as follows: -
- **Technical Feasibility:** This is concerned with specifying equipment and software that will successfully satisfy the use considerably, but might include
    - The feasibility to produce output in a given time because system is fast enough to handle multiple users.
    - Response time under certain circumstances and ability to process a certain volume of transaction of a particular speed.
    - Feasibility to communicate data to distant location.
- **Economic Feasibility:** Economic analysis is the most frequently used technique used for evaluating the effectiveness of a proposed system. More commonly known as cost/benefit analysis the procedure is to determine the benefits and savings that are

- expected from a proposed system and compared them with cost. Though the cost of installing the system may appear high, it is one time investment. The resulting benefits is that automation results in turnaround time. The resulting cost/benefit ratio is favorable.

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

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

SYSTEM ANALYSIS

The Analysis model:

The analysis model must achieve three primary objectives:

- To describe what the customer requires.
- To establish the basis for the enhancement of a software design.
- To define a set of requirements that can be validated once the software is completely enhanced. The main elements of the analysis model are briefly described below.
- At the core of the model lies the ***data dictionary***, which is a repository that contains descriptions of all the data objects consumed or produced by the software. Three different diagrams surround the core.
- The **entity relation diagram** depicts relationships between data objects.

- The **data flow diagram** provides an indication of how the data is transformed as they move through the system.
- The **state transition diagram** indicates how the system behaves as a consequence of external events.

DESIGN PHASE

The design phase involves converting the informational, functional, and network requirements identified during the initiation and planning phases into unified design specifications that developers use to script programs during the development phase. Program designs are constructed in various ways. Using a top- down approach, designers first identify and link major program components and interfaces, then expand design layouts as they identify and link smaller subsystems and connections. Using a bottom-up approach, designers first identify and link minor program components and interfaces, then expand design layouts as they identify and link larger systems and connections.

Designers should carefully document completed designs. Detailed documentation enhances a programmer's ability to develop programs and modify them after they are placed in production. The documentation also helps management ensure final programs are consistent with original goals and specifications. Organizations should create initial testing, conversion, implementation, and training plans during the design phase. Additionally, they should draft user, operator, and maintenance manuals.

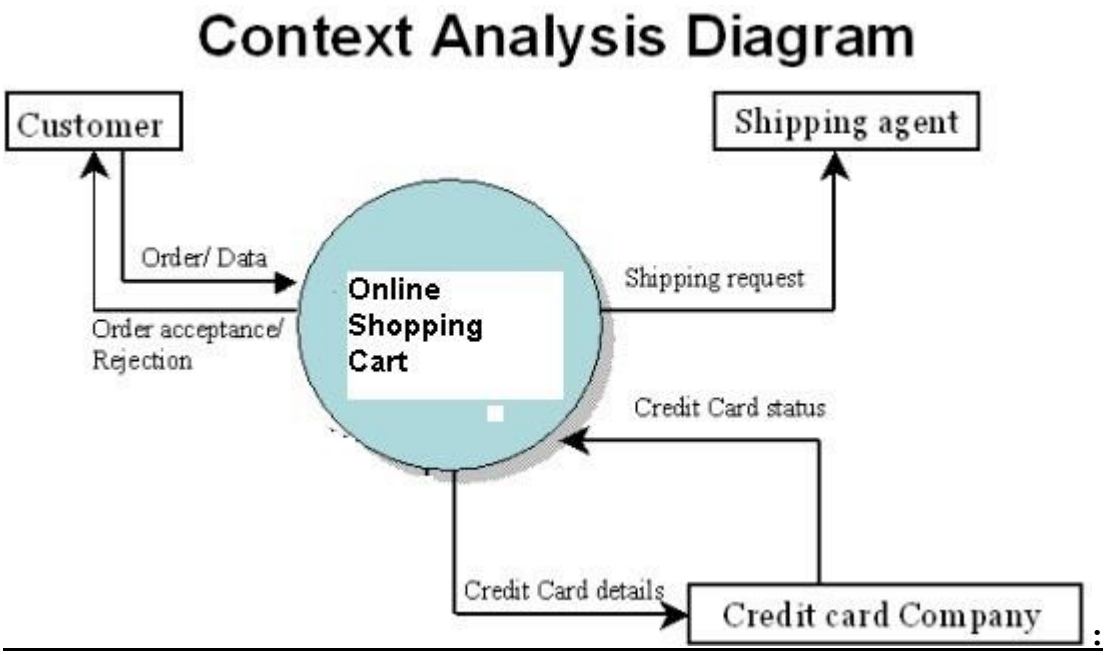
For design of the website project:

1. First Database has to be designed which can be used to handle all the requirements of the users.
2. The basic structure of the website has to be designed.
3. The main template to be used for the website is designed.

DATA FLOW DIAGRAM

DATA FLOW DIAGRAM OF ONLINE SHOPPING SYSTEM

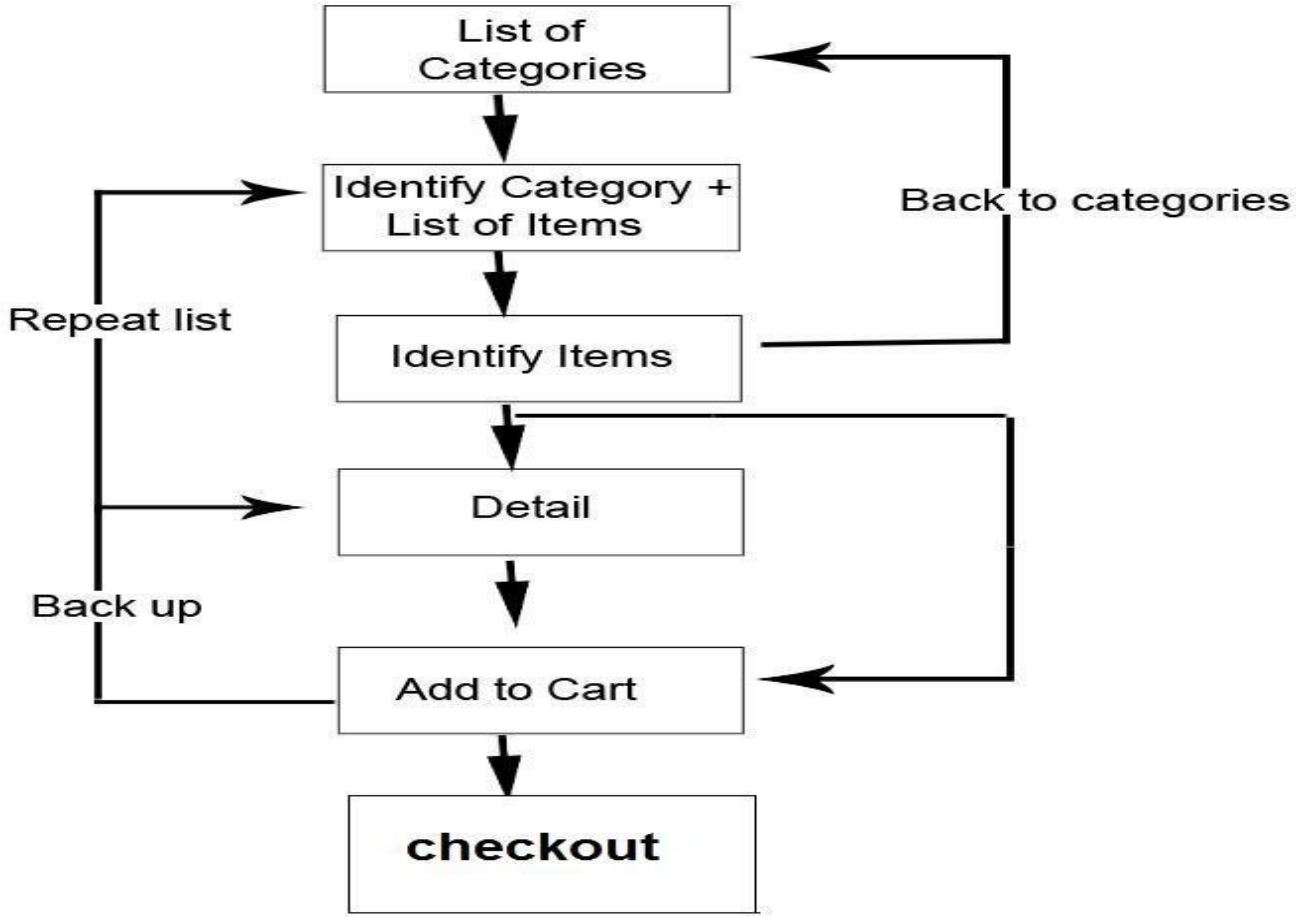
Context Level Diagram



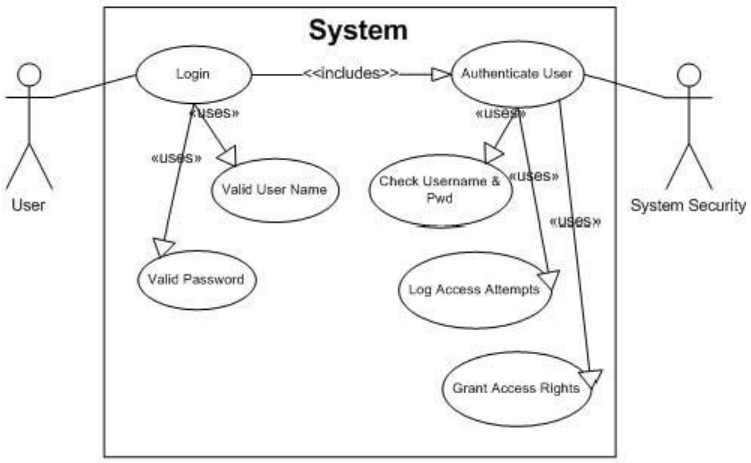
:

First Level DFD Diagram:

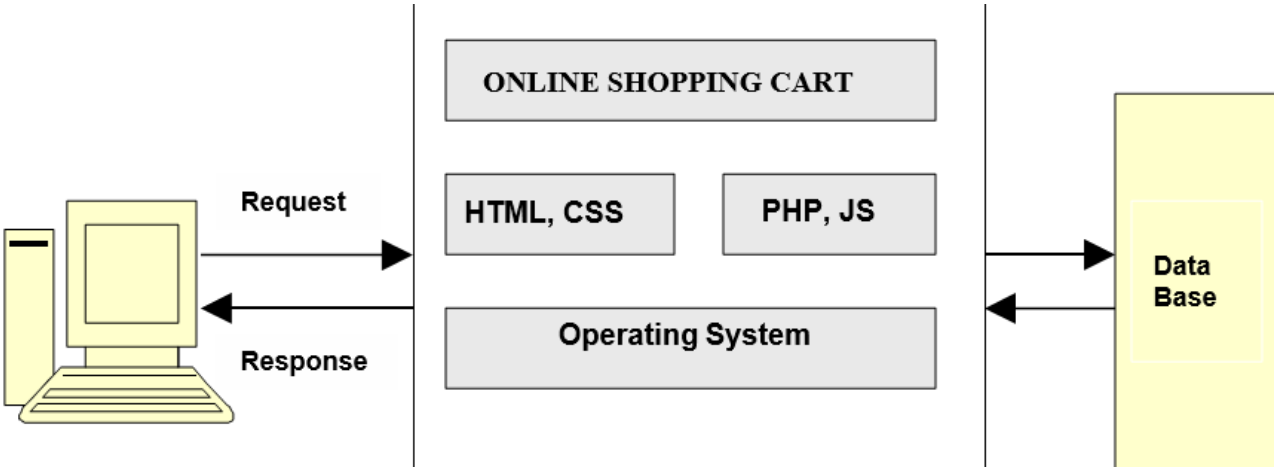
Second level DFD Diagram:



Login Activity Diagram:

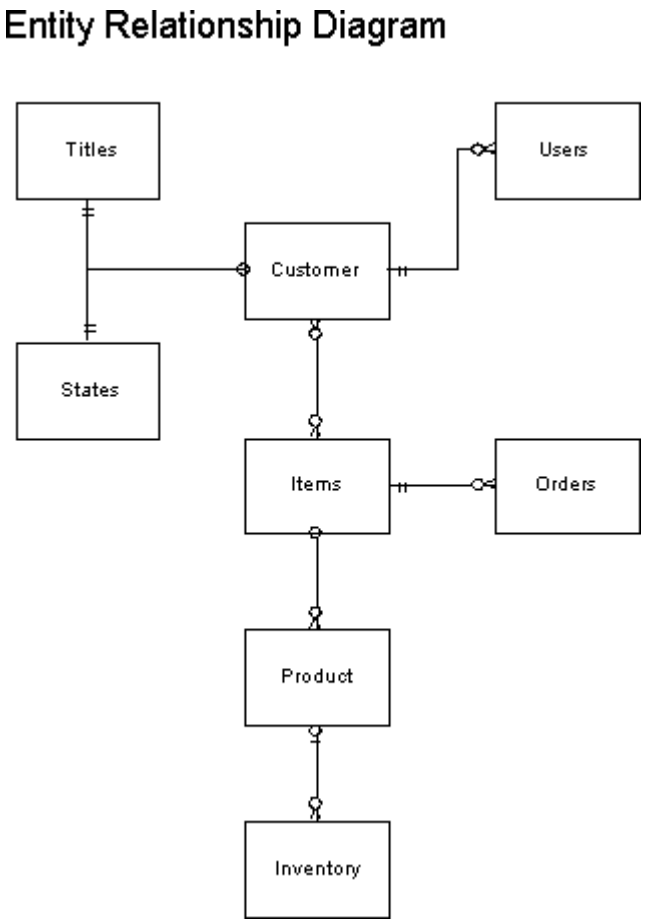


Implementation View:



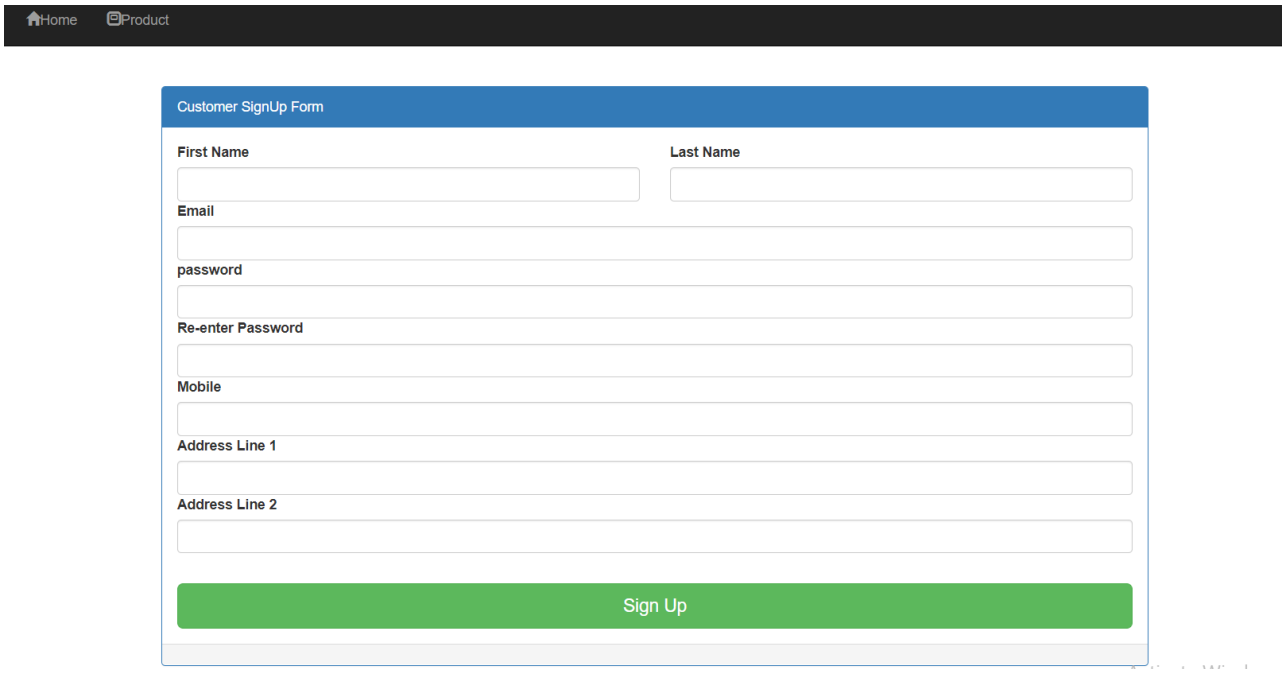
E-R Diagram:

-

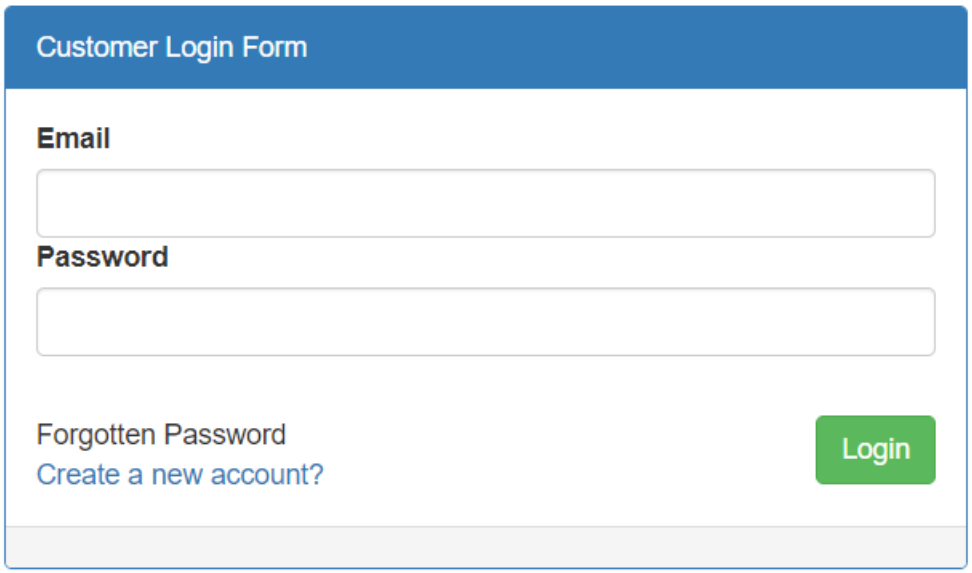

Entity Relationship Diagram
Titles
Users
Customer
States
Items
Orders
Product
Inventory

Screenshot

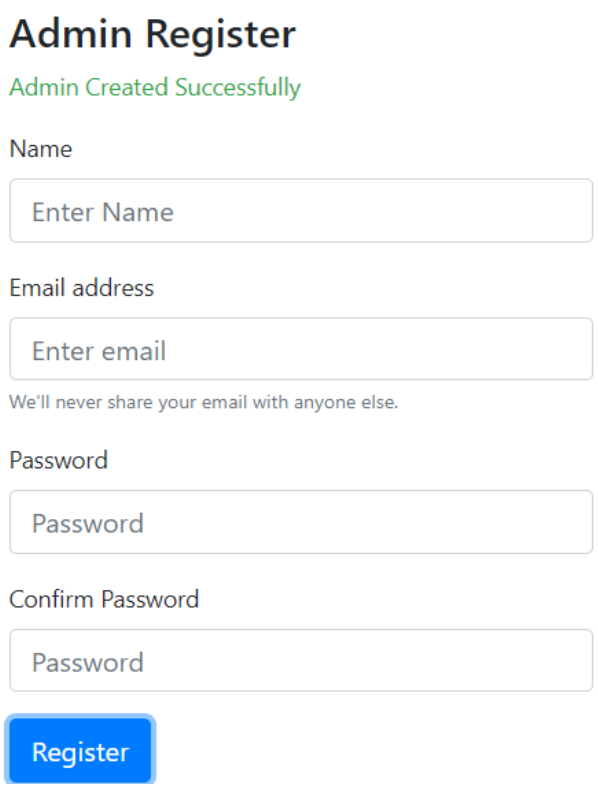
Sign up Screen



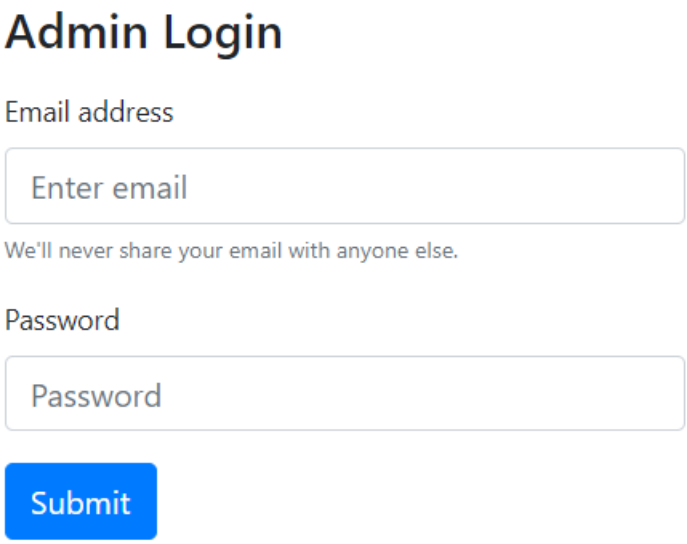
Login Screen



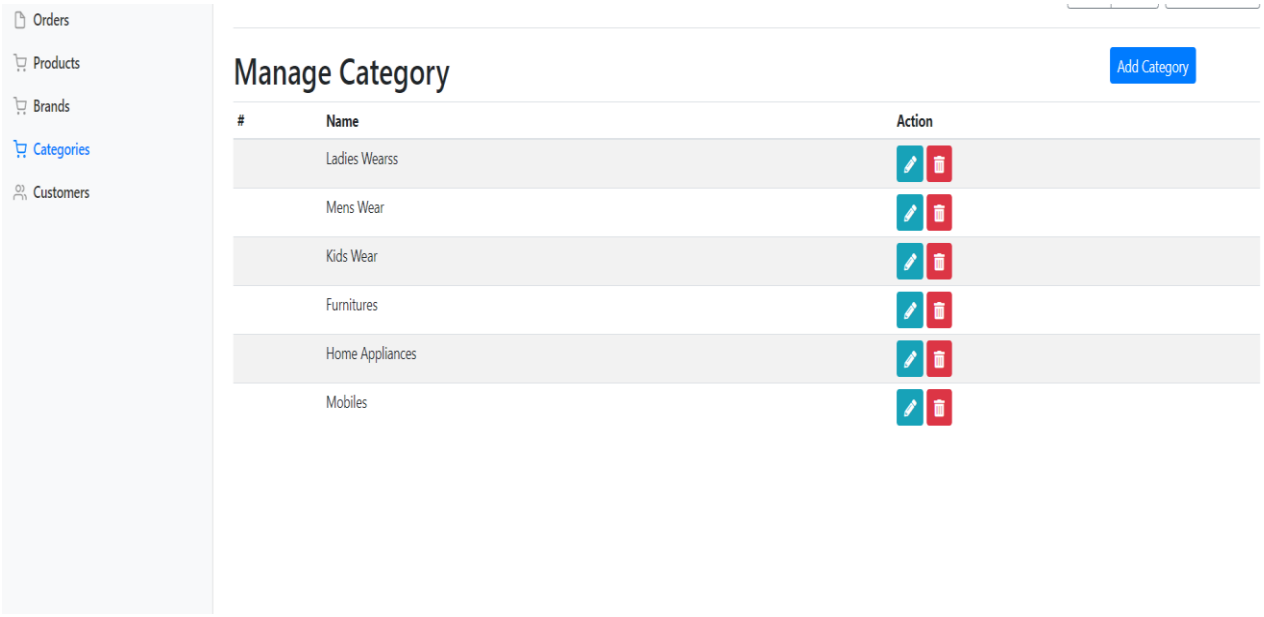
Admin Register Screen



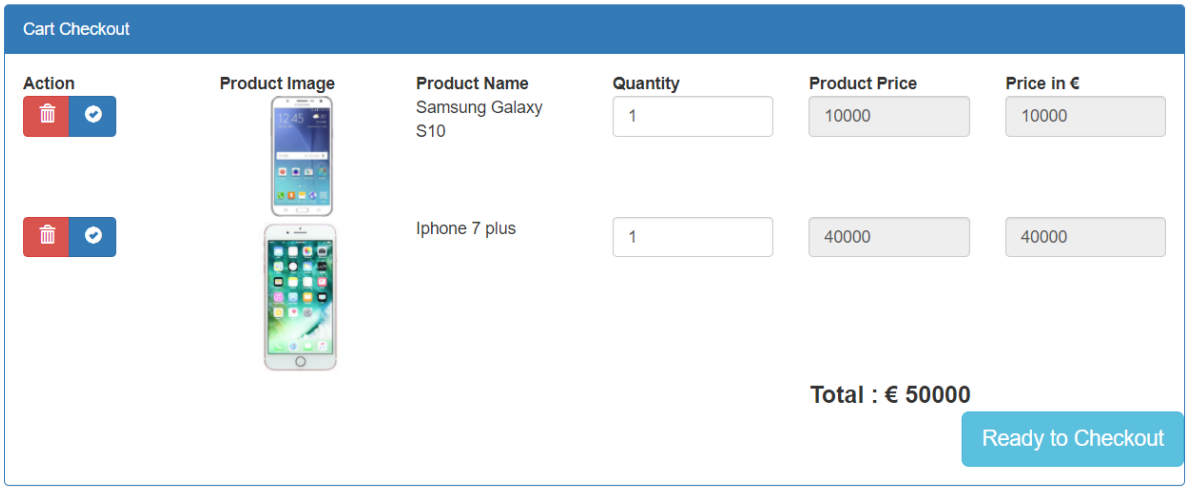
Admin Login Screen



View screen of Admin



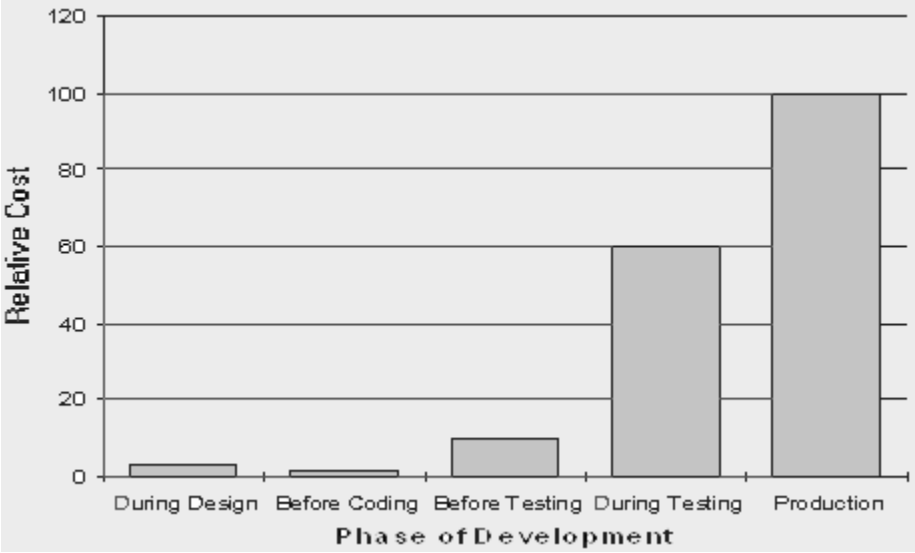
User's Cart View



TESTING

Software testing

Software testing is the process used to measure the quality of developed computer software. Usually, quality is constrained to such topics as correctness, completeness, security, but can also include more technical requirements as described under the ISO standard ISO 9126, such as capability, reliability, efficiency, portability, maintainability, compatibility, and usability. Testing is a process of technical investigation, performed on behalf of stakeholders, that is intended to reveal quality-related information about the product with respect to the context in which it is intended to operate.



White box, black box, and grey box testing

White box and black box testing are terms used to describe the point of view that a test engineer takes when designing test cases. **Black box** testing treats the software as a black-box without any understanding as to how the internals behave. Thus, the tester inputs data and only sees the output from the test object. This level of testing usually requires thorough test cases to be provided to the tester who then can simply verify that for a given input, the output value (or behavior), is the same as the expected value specified in the test case.

White box testing, however, is when the tester has access to the internal data structures, code,

and algorithms. For this reason, unit testing and debugging can be classified as white-box testing and it usually requires writing code, or at a minimum, stepping through it, and thus requires more skill than the black-box tester. If the software in test is an interface or API of any sort, white-box testing is almost always required.

In recent years the term grey box testing has come into common usage. This involves having access to internal data structures and algorithms for purposes of designing the test cases, but testing at the user, or black-box level. Manipulating input data and formatting output do not qualify as grey-box because the input and output are clearly outside of the black-box we are calling the software under test. This is particularly important when conducting integration testing between two modules of code written by two different developers, where only the interfaces are exposed for test.

Grey box testing could be used in the context of testing a client-server environment when the tester has control over the input, inspects the value in a SQL database, and the output value, and then compares all three (the input, sql value, and output), to determine if the data got corrupt on the database insertion or retrieval.

Verification and Validation

Software testing is used in association with verification and validation (V&V). *Verification* is the checking of or testing of items, including software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, inspections, and walkthroughs. *Validation* is the process of checking what has been specified is what the user actually wanted.

- Verification: Have we built the software right? (i.e. does it match the specification).
- Validation: Have we built the right software? (i.e. Is this what the customer wants?)

Level of testing

- **Unit testing** tests the minimal software component, or module. Each unit (basic component) of the software is tested to verify that the detailed design for the unit has been correctly implemented. In an Object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.
- **Integration testing** exposes defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.
- **Functional testing** tests at any level (class, module, interface, or system) for proper functionality as defined in the specification.

- **System testing** tests a completely integrated system to verify that it meets its requirements.
- **System integration testing** verifies that a system is integrated to any external or third party systems defined in the system requirements.
- **Acceptance testing** can be conducted by the end-user, customer, or client to validate whether or not to accept the product. Acceptance testing may be performed as part of the hand-off process between any two phases of development.

    - ***Alpha testing*** is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.
    - ***Beta testing*** comes after alpha testing. Versions of the software, known as beta versions, are released to a limited audience outside of the company. The software is released to groups of people so that further testing can ensure the product has few faults or bugs. Sometimes, beta versions are made available to the open public to increase the feedback field to a maximal number of future users.

It should be noted that although both Alpha and Beta are referred to as testing it is in fact use immersion. The rigors that are applied are often unsystematic and many of the basic tenets of testing process are not used. The Alpha and Beta period provides insight into environmental and utilization conditions that can impact the software.

After modifying software, either for a change in functionality or to fix defects, a regression test re-runs previously passing tests on the modified software to ensure that the modifications haven't unintentionally caused a *regression* of previous functionality. Regression testing can be performed at any or all of the above test levels. These regression tests are often automated.

SMOKE TESTING

Smoke testing is a term used in plumbing, woodwind repair, electronics, and computer software development. It refers to the first test made after repairs or first assembly to provide some assurance that the system under test will not catastrophically fail. After a *smoke test* proves that the pipes will not leak, the keys seal properly, the circuit will not burn, or the software will not crash outright, the assembly is ready for more stressful testing.

- In plumbing, a *smoke test* forces actual smoke through newly plumbed pipes to find leaks, before water is allowed to flow through the pipes.
- In woodwind instrument repair, a smoke test involves plugging one end of an instrument and blowing smoke into the other to test for leaks. (This test is no longer in common use)
- In electronics, a *smoke testing* is the first time a circuit is attached to power, which will sometimes produce actual smoke if a design or wiring mistake has been made.
- In computer programming and software testing, *smoke testing* is a preliminary to further testing, which should reveal simple failures severe enough to reject a prospective software release. In this case, the smoke is metaphorical.

IMPLEMENTATION PHASE

The implementation phase involves installing approved applications into production environments. Primary tasks include announcing the implementation schedule, training end users, and installing the product. Additionally, organizations should input and verify data, configure and test system and security parameters, and conduct post-implementation reviews. Management should circulate implementation schedules to all affected parties and should notify users of any implementation responsibilities.

After organizations install a product, pre-existing data is manually input or electronically transferred to a new system. Verifying the accuracy of the input data and security configurations is a critical part of the implementation process. Organizations often run a new system in parallel with an old system until they verify the accuracy and reliability of the new system. Employees should document any programming, procedural, or configuration changes made during the verification process.

For implementation of the website project:

1. The website can be installed on a computer or a server which has PHP and MYSQL installed in it.

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

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

TEST PLAN

The testing phase requires organizations to complete various tests to ensure the accuracy of programmed code, the inclusion of expected functionality, and the interoperability of applications and other network components. Thorough testing is critical to ensuring systems meet organizational and end-user requirements. Test plans created during initial project phases enhance an organization's ability to create detailed tests.

A bottom-up approach tests smaller components first and progressively adds and tests additional components and systems. A top-down approach first tests major components and connections and progressively tests smaller components and connections.

Bottom-up tests often begin with functional (requirements based) testing. Functional tests should ensure that expected functional, security, and internal control features are present and operating properly. Testers then complete integration and end-to-end testing to ensure application and system components interact properly. Users then conduct acceptance tests to ensure systems meet defined acceptance criteria. Organizations should review and complete user, operator, and maintenance manuals during the testing phase. Additionally, they should finalize conversion, implementation, and training plans.

For testing of the website:

1. All the features of the website are tested by running each function available in the website.

2. The results of the tests conducted on the website are analyzed properly. Only after getting satisfactory results of testing the website can be uploaded on the network i.e. internet.

MAINTENANCE

The maintenance phase involves making changes to hardware, software, and documentation to support its operational effectiveness. It includes making changes to improve a system's performance, correct problems, enhance security, or address user requirements. To ensure modifications do not disrupt operations or degrade a system's performance or security, organizations should establish appropriate change management standards and procedures. Routine changes are not as complex as major modifications and can usually be implemented in the normal course of business. Routine change controls should include procedures for requesting, evaluating, approving, testing, installing, and documenting software modifications. Maintaining accurate, up-to-date hardware and software inventories is a critical part of all change management processes. Management should carefully document all modifications to ensure accurate system inventories. Management should coordinate all technology related changes through an oversight committee and assign an appropriate party responsibility for administering software patch management programs. Quality assurance, security, audit, regulatory compliance, network, and end-user personnel should be appropriately included in change management processes. Risk and security review should be done whenever a system modification is implemented to ensure controls remain in place.

For maintenance of the website:

1. The database has to be updated regularly according to new available information.
2. Redundant and false information must be removed from the database.
3. Newer version of PHP and MYSQL can be used for up gradation of website and to improve the overall performance of the system.

FUTURE ENHANCEMENTS

The Online Shopping cart website will have the prominent features including:

- Providing personalized inbox to the user.
- Providing video conferencing with the Artists
- Providing links to news which will elaborate more information about them.

Conclusions

We have successfully implemented the site 'Online Shopping Cart'. With the help of various links and tools, we have been able to provide a site which is live and running on the web. We have been successful in our attempt to take care of the needs of both the customers as well as the administrator. Finally, we hope that this will go a long way in popularizing the organization and making its work of enrollment, keeping track of Artist's Arts, problem solving, etc much more efficient.