Contents

[1. Chapter Introduction 4](#_Toc528251057)

[1.1 Introduction 5](#_Toc528251058)

[1.2 Problem Statement 5](#_Toc528251059)

[1.3 Reasons and Motivations 5](#_Toc528251060)

[1.4 Aims and Objective 5](#_Toc528251061)

[1.5 Scope of Project 6](#_Toc528251062)

[1.6 SDLC Models 6](#_Toc528251063)

[1.7 SDLC Model Chosen 6](#_Toc528251064)

[1.8 Procedure 7](#_Toc528251065)

[1.9 Software Tools 7](#_Toc528251066)

[1.10 Project Outline 8](#_Toc528251067)

[2. Chapter Literature Review and Analysis 9](#_Toc528251068)

[2.1 Literature Review 10](#_Toc528251070)

[2.2 Existing Systems 10](#_Toc528251071)

[2.2.1 Existing System I 10](#_Toc528251072)

[2.2.2 Existing System II 11](#_Toc528251073)

[2.2.3 Existing System III 11](#_Toc528251074)

[2.3 Proposed System 11](#_Toc528251075)

[2.4 Comparison 12](#_Toc528251076)

[2.5 Feasibility Report 12](#_Toc528251077)

[2.5.1 Economic Feasibility 13](#_Toc528251078)

[2.5.2 Technical Feasibility 13](#_Toc528251079)

[2.5.3 Operation Feasibility 13](#_Toc528251080)

[2.5.4 Resource Feasibility 13](#_Toc528251081)

[2.6 Statement of Work 13](#_Toc528251082)

[2.7 Gantt Chart 14](#_Toc528251083)

[3. System Design 16](#_Toc528251084)

[3.1 Introduction 17](#_Toc528251086)

[3.2 Data Flow Diagram 17](#_Toc528251087)

[3.2.1 Context DFD 18](#_Toc528251088)

[3.2.2 Level 1 DFD 19](#_Toc528251089)

[3.3 Use Case Diagram 20](#_Toc528251090)

[3.4 Sequence Diagram 23](#_Toc528251091)

[3.5 Class Diagram 23](#_Toc528251092)

[3.6 ER Diagram 24](#_Toc528251093)

[3.7 Database 26](#_Toc528251094)

[3.7.1 Database Schema 26](#_Toc528251095)

[3.8 Normalization 29](#_Toc528251096)

[3.8.1 Normalization Types with Examples 30](#_Toc528251097)

[4. Development 33](#_Toc528251098)

[4.1 What is Incremental Model? 34](#_Toc528251100)

[4.2 Plate form 34](#_Toc528251101)

[4.3 Language 35](#_Toc528251102)

[5. Deployment 36](#_Toc528251103)

[5.1 Implementation 37](#_Toc528251105)

[5.2 Configuration 37](#_Toc528251106)

[5.2.1 Unit Testing 37](#_Toc528251107)

[5.2.2 Integration Testing 38](#_Toc528251108)

[5.2.3 User Acceptance Testing 38](#_Toc528251109)

[5.3 Test cases 38](#_Toc528251110)

[5.3.1 . Test Case 1: Admin Login 38](#_Toc528251111)

[5.3.2 5.5.2. Test Case 2: Add new Ads 38](#_Toc528251112)

[5.3.3 5.5.3. Test Case 3: Delete ads 39](#_Toc528251113)

[5.3.4 Test Case 4: View ads list 39](#_Toc528251114)

[5.3.5 5.5.5. Test Case 5: Update Ads 40](#_Toc528251115)

[5.3.6 Test cases 40](#_Toc528251116)

[5.3.7 Test cases 41](#_Toc528251117)

[5.4 User Interface 41](#_Toc528251118)

[5.5 Admin page 41](#_Toc528251119)

[5.6 Customer Tab 42](#_Toc528251120)

[5.7 Menu Tab 43](#_Toc528251121)

[5.8 Dashboard page 43](#_Toc528251122)

[5.9 Categories Page 43](#_Toc528251123)

[5.10 Contact us page 43](#_Toc528251124)

[5.11 Location page 44](#_Toc528251125)

[5.12 About us page 44](#_Toc528251126)

[6. Conclusion & References 45](#_Toc528251127)

[6.1. Conclusion 46](#_Toc528251128)

[6.2. References 46](#_Toc528251129)

List of figures

[Figure 1 of existing system I 10](#_Toc528251003)

[Figure 2 of existing System II 11](#_Toc528251004)

[Figure 3 of existing system III 11](#_Toc528251005)

[Figure 4of gannet chart of project 15](#_Toc528251006)

[Figure 5 of DFD 0 level 19](#_Toc528251007)

[Figure 6 of DFD Level 1 20](#_Toc528251008)

[Figure 7 Use Case Diagram 22](#_Toc528251009)

[Figure 8 of use case diagram for admin 22](#_Toc528251010)

[Figure 9 sequence diagram 23](#_Toc528251011)

[Figure 10 of class diagram of database 24](#_Toc528251012)

[Figure 11of Ads table in database 27](#_Toc528251013)

[Figure 12 of categories of database 27](#_Toc528251014)

[Figure 13 of countries table in database 28](#_Toc528251015)

[Figure 14 of states table in database 28](#_Toc528251016)

[Figure 15 of cities table in database 28](#_Toc528251017)

[Figure 16 of inbox table in database 28](#_Toc528251018)

[Figure 17 of admin table in database 28](#_Toc528251019)

[Figure 18 of user table in database 29](#_Toc528251020)

[Figure 19 of Schema diagram in database 29](#_Toc528251021)

[Figure 20 of types of normalization in database 30](#_Toc528251022)

[Figure 21 types of normalization 30](#_Toc528251023)

[Figure 22 of NF 30](#_Toc528251024)

[Figure 23 of 2NF 31](#_Toc528251025)

[Figure 24 After apply 2NF 31](#_Toc528251026)

[Figure 25 of 3NF 32](#_Toc528251027)

[Figure 26 of Incremental Model 34](#_Toc528251028)

[Figure 27 of login page 42](#_Toc528251029)

[Figure 28 of Customer services 42](#_Toc528251030)

[Figure 29of menu tab 43](#_Toc528251031)

[Figure 30 of dashboard of my project 43](#_Toc528251032)

[Figure 31 of categories page 43](#_Toc528251033)

[Figure 32 of contact us page 44](#_Toc528251034)

[Figure 33of location page 44](#_Toc528251035)

[Figure 34of about us page 44](#_Toc528251036)

List of Table

[Table 1 of comparison of existing and proposed system 12](#_Toc528182494)

[Table 2 of Statement work 14](#_Toc528182495)

[Table 3 of Gannt chart 15](#_Toc528182496)

# Chapter Introduction

## Introduction

Online Big Store are used to provide the customers with huge amount of information. This site provides the latest updates of the automobiles, Vehicles, Mobiles & Tablets, Electronics & Appliances, Real Estate, Home & Lifestyle, Jobs, Services and Animals.

An easy way to buy or sale a ads just sitting in front of your computers by registering into our site.

This is to facilitate all people who are busy with their works and have no time to get their desired goods. We are here to provide you all the best and suitable places for sale. If once you register into our site, then you are benefitted with our latest updates of the sales!

This is used to advertise the adss with images. One can easily login to get any kind of information. Here the user is also facilitated to directly interact with the consumer. He can get the desired ads with different rates and quality.

The main goal is to provide the customer with various goods just by sitting in front of a computer. He can get the goods easily without moving from place to place.

Consumers can also have a chance of introducing their ads not only in a single place but throughout the world using online adds. This system allows the user to interact directly just by sitting in front of the computer. Consumer have a chance of comparing the ads and purchase the desired one

## Problem Statement

This is an era of Information Technology where getting information is the base of each and every thing. Success of any business depends upon its popularity and good will. Today, market has been expanded and you have to be a global player, today every individual wants to access most of the information from his own place and then do transaction. The project would help in effective and systematic access to the Advertisements posted by the Poster and invalid access by any person will be caught at the time of registration of the Viewer.

## Reasons and Motivations

In the earlier times it was difficult to reach out with the classifieds to people spread over a wider area easily. If we publish it in a print media, the editions would be limited, in the case of television the audiences would be limited. Ever since internet has burst into the scene, much like everything advertisement industry has also picked up. Big Store are the need of the hour as a user can become both buyer and seller on the click of a button. More over since it is published online, any person can see it sitting anywhere in the world. All it takes is basic computer knowledge to browse this site.

## Aims and Objective

Following are the objectives of the Big Store

* To boost up the Process of sales, purchase and services
* To Decrease time consumption
* To Ease the Vendors
* To Reduce the cost for every vendor
* This project will manage large amount of data in short time.
* No data redundancy.
* To reduce the time consuming

## Scope of Project

Purchasing and selling goods and services over the internet without the need of going physically to the market is what Big Store all about. Big Store is just like a retail market that we do by going to the market, but it is done through the internet. Big Store has made purchase and sells painless and added more fun. Big Store offer Adss description, pictures, price and much more.

## SDLC Models

One of the basic notions of the software development process is SDLC models which stands for Software Development Life Cycle models. SDLC – is a continuous process, which starts from the moment, when it’s decided to launch the project, and it ends at the moment of its full remove from the exploitation. There is no one single SDLC model. They are divided into main groups, each with its features and weaknesses.

Evolving from the first and oldest “waterfall” SDLC model, their variety significantly expanded. The SDLC model’s diversity is predetermined by the wide number of ads types – starting with a web application development to a complex medical software. And if you take one of the SDLC models mentioned below as the basis – in any case, it should be adjusted to the features of the ads, project, and company. The most used, popular and important SDLC models are given below:

* Waterfall model
* Iterative model
* Spiral model
* V-shaped model
* Agile model

## SDLC Model Chosen

An iterative life cycle model does not start with a full specification of requirements. In this model, the development begins by specifying and implementing just part of the software, which is then reviewed in order to identify further requirements. Moreover, in iterative model, the iterative process starts with a simple implementation of a small set of the software requirements, which iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed. Each release of Iterative Model is developed in a specific and fixed time period, which is called iteration.

Furthermore, this iteration focuses on a certain set of requirements. Each cycle ends with a usable system i.e., a particular iteration results in an executable release. Iterative Model allows accessing previous phases, in which the changes are made accordingly. The final output of the ads is revived at the end of the Software Development Life Cycle (SDLC). Typically iterative development is used in conjunction with incremental development, in which a longer software development cycle is split into smaller segments that are built upon each other. Hence, iterative model is used in following scenarios:

* When the requirements of the complete system are clearly defined and understood.
* The major requirements are defined, while some functionalities and requested enhancements evolve with the process of the development process.
* A new technology is being used and is being learnt by the development team, while they are working on the project.
* If there are some high-risk features and goals, which might change in the future.
* When the resources with needed skill sets are not available and are planned to be used on contract basis for specific iterations.

Process of Iterative Model:

The process of Iterative Model is cyclic, unlike the more traditional models that focus on a rigorous step-by-step process of development. In this process, once the initial planning is complete, a handful of phases are repeated again and again, with the completion of each cycle incrementally improving and iterating on the software. Other phases of the iterative model are described below:

Iterative model is best for web projects. Because many times change in project.

## Procedure

User can view detail of all ads that are advertised on Big Store. User will register and login to add new ads on Big Store

* Manage Profile
* Manage Adss
* Manage Status of Adss
* Manage Feedback

## Software Tools

* Hardware Used
  + Dell Laptop
  + Core i3 2.4 GHz
  + 4GB RAM
* Software Used
* OS Related Information
* Windows 10 x64
* Resolution 1280x1024
* Front End Related Information
  + Html, Css
  + Bootstrap,
  + Javascript,
  + JQuery
* Back End Related Information
  + Php
  + Database
  + MYSQL SERVER
* Other Software
  + Xampp Server,
  + Notpad++
  + Microsoft Word 2016: For report making

## Project Outline

Chapter Name in Heading 1, Font Times New Roman, Size 18 Bold

Main Heading in Heading 2, Font Times New Roman, Size 15 Bold

Sub Heading in Heading 3, Font Times New Roman, Size 12 Bold

Paragraph in Normal, Justified, Font Times New Roman, Size 12

# Chapter Literature Review and Analysis



## Literature Review

Big Store play a great importance in the modern business environment. Big Store has opened the door of opportunity and advantage to the firms. Marketing is basically helping the consumer’s needs more effectively and efficiently with good service and ads with best price. A good marketer Continuously satisfying consumers’ needs in better way. Sometimes opportunity to give the consumers in better way is designed by marketers himself and sometimes it is offered by the technology. Internet is changing the way consumers shop for goods and services and has rapidly evolved into a global event. It is examined that internet is becoming a hotbed of advertising, shopping and commercial activity. It is stated that internet is influencing people’s daily life more so as compared to past. People’s daily activities have gradually shifted from physical conditions to virtual environment. The shopping and payment surroundings have also changed from physical store into online stores. It is investigated that online technologies provide many competitive advantages like agility, selectivity, individuality and interactivity. Customers can take enjoy online shopping for 24 hour per day. Consumers can purchase ro salle any goods and services anytime at everywhere. Online shopping is user friendly compare to an store shopping because consumers can just complete his requirements just with a click of mouse without leaving their home.

## Existing Systems

### Existing System I

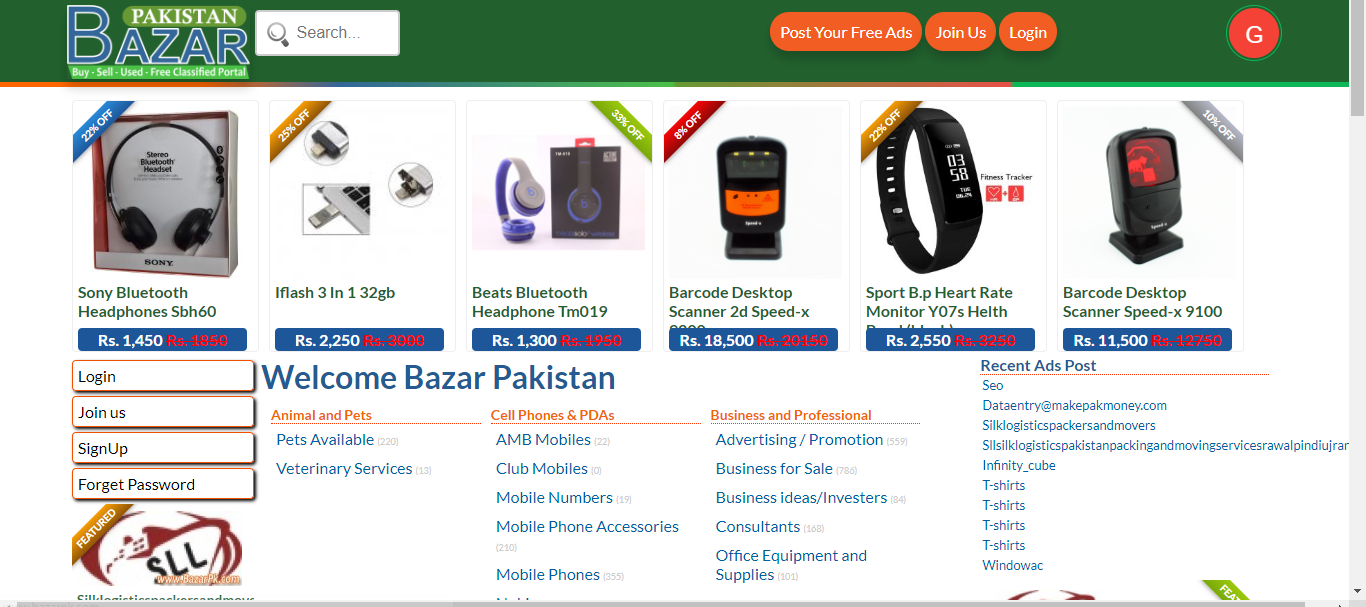


Figure 1 of existing system I

Problem is that this system doesn’t responsive and it doesn’t have advance option.

### Existing System II

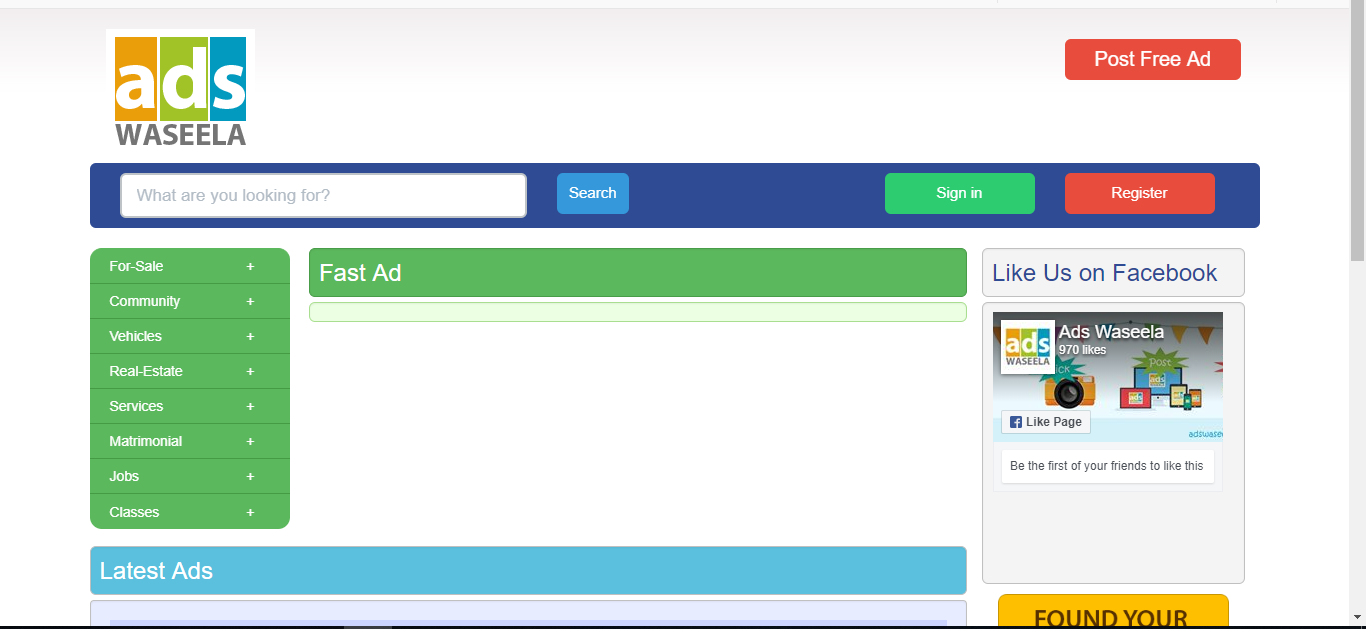


Figure 2 of existing System II

This is very old and not responsive and Categories are not managing not good search.

### Existing System III

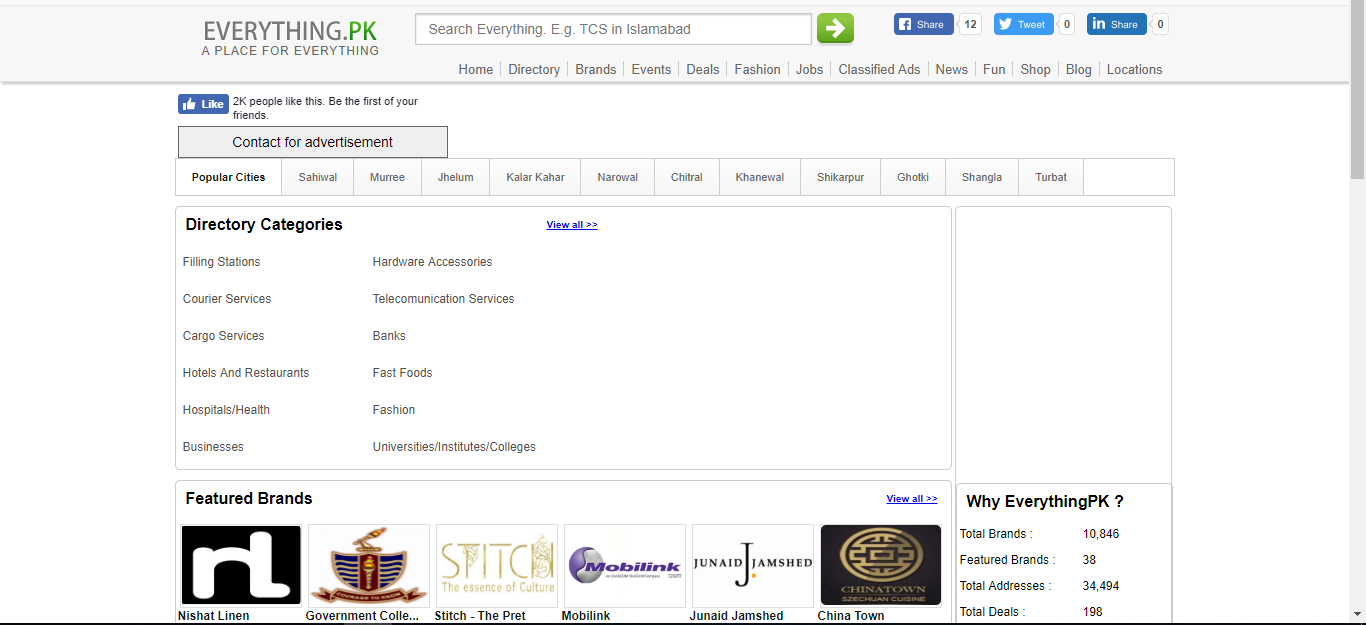


Figure 3 of existing system III

This system just has basic interface not much more function and design is not effective.

## Proposed System

The proposed system is a computerized one. This has greater accuracy and efficiency. This takes only limited time for retrieval of desired result from the user. The proposed system can be used to maintain efficiently the postings and retrieval of any type of Advertisement posted by a Poster. The proposed system is also useful and helpful in checking out different Advertisements by a Viewer so that he/she can easily categorize and select the appropriate AD for him/herself depending on the requirement and his/her choice. He/she can also create his/her mail account using my system. Since my system is allowing Banner Advertisements to obtain revenue out of it, a user can post his/her company’s or firm’s Advertisement using my system.

The system includes two users:

1. Administrator Poster (post AD’s)
2. General User Viewer (view AD’s)

The user gets into the system using user name and a unique password. Each user has his own accessibility permission to accomplish his task flawlessly. The administrator also has an ID and password to get access to the system so that no unauthorized person is able to keep an eye on the working of the complete system. Advantages of the proposed system are

1. Easy access to the data
2. The new system is more user friendly, reliable and flexible.
3. Timely Report generation.

## Comparison

|  |  |
| --- | --- |
| Existing systems | Proposed system |
| Old and Unfriendly Interface | Friendly and Eye-Catching Interface |
| Workflow was not fluid | Workflow is fluid |
| No Advance Reporting | Has Advance Reporting |
| Hard to Operate | Easy to Operate |
|  |  |

Table 1 of comparison of existing and proposed system

## Feasibility Report

The feasibility study is the important step in any software development process. This is because it makes analysis of different aspects like cost required for developing and executing the system, the time required for each phase of the system and so on. If these important factors are not analyzed then definitely it would have impact on the organization and the development and the system would be a total failure. So, for running the project and the organization successfully this step is a very important step in a software development life cycle process.

By making analysis this way it would be possible to make a report of identified area of problem. By making a detailed analysis in this area a detailed document or report is prepared in this phase which has details like project plan or schedule of the project, the cost estimated for developing and executing the system, target dates for each phase of delivery of system developed and so on. This phase is the base of software development process since further steps taken in software development life cycle would be based on the analysis made on this phase and so careful analysis has to be made in this phase.

The feasibility study that was done for this project included the following considerations –

### Economic Feasibility

This is a very important aspect to be considered while developing a project. We decided the technology based on minimum possible cost factor. All hardware and software cost have to be borne by the organization. We analyzed the existing system and concluded that the existing systems with the organization only needed to be updated to newer configuration instead of going for new hardware setups. Overall, we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

### Technical Feasibility

This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system. or this feasibility study, we studied complete functionality to be provided in the system, as described in the System Requirement Specification (SRS), and checked if everything was possible using Php and MySQL Server. After the study we came to conclusion that we can proceed further with the tools and development environment chosen by us. This was important in our case as we were working on two various phases of the department that will need to be integrated in future to make an extended system.

### Operation Feasibility

No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken all self-explanatory even to a layman. Besides, a proper training has been conducted to let know the essence of the system to the users so that they feel comfortable with new system. As far our study is concerned the clients are comfortable and happy as the system has cut down their loads and doing all the complex activities itself.

### Resource Feasibility

This is also an important check to be done that the required resources will be available or not. As far as Software and hardware were considered, there was no such constraint using Php and MySQL Server as Front-end and Back-end respectively.

## Statement of Work

A statement of work (SoW) is a document routinely employed in the field of project management. It defines project-specific activities, deliverables and timelines for a vendor providing services to the client. The SOW typically also includes detailed requirements and pricing, with standard regulatory and governance terms and conditions.

|  |  |
| --- | --- |
| ORGANIZATION | |
| NAME |  |
| CONTACT |  |
| ADDRESS |  |

|  |  |
| --- | --- |
| PROJECT | |
| NAME |  |
| BRAND |  |
| ADS |  |
| BEGIN DATE |  |
| END DATE |  |
| DURATION |  |

|  |
| --- |
| ASSUMPTIONS |
|  |

|  |  |
| --- | --- |
| GOALS | |
| OBJECTIVE |  |
| SCOPE |  |
| DELIVERABLES |  |

|  |  |
| --- | --- |
| PAYMENTS | |
| ADVANCE |  |
| MID |  |
| TOTAL COST |  |

Table 2 of Statement work

## Gantt Chart

A Gantt chart is a type of bar chart that illustrates a project schedule, named after its inventor, Henry Gantt (1861–1919), who designed such a chart around the years 1910–1915. Modern Gantt charts also show the dependency relationships between activities and current schedule status.

A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity.



Figure of gannet chart of project

# System Design



## Introduction

**Systems design** is the process of defining the [architecture](https://en.wikipedia.org/wiki/Systems_architecture), modules, interfaces, and [data](https://en.wikipedia.org/wiki/Data) for a [system](https://en.wikipedia.org/wiki/System) to satisfy specified [requirements](https://en.wikipedia.org/wiki/Requirement). Systems design could be seen as the application of [systems theory](https://en.wikipedia.org/wiki/Systems_theory) to [ads development](https://en.wikipedia.org/wiki/Product_development). There is some overlap with the disciplines of [systems analysis](https://en.wikipedia.org/wiki/Systems_analysis), [systems architecture](https://en.wikipedia.org/wiki/Systems_architecture) and [systems engineering](https://en.wikipedia.org/wiki/Systems_engineering).

## Data Flow Diagram

A two-dimensional diagram that explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output.

Individuals seeking to draft a data flow diagram must

(1) identify external inputs and outputs

(2) determine how the inputs and outputs relate to each other, and

(3) explain with graphics how these connections relate and what they result in.

This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspects. A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. As its name indicates its focus is on the flow of information, where data comes from, where it goes and how it gets stored.

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system. The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams. DFD has often been used due to the following reasons:

* Logical information flow of the system
* Determination of physical system construction requirements
* Simplicity of notation
* Establishment of manual and automated systems requirements

There are four basic symbols that are used to represent a data-flow diagram.

**Process**

A process receives input data and produces output with a different content or form. Processes can be as simple as collecting input data and saving in the database, or it can be complex as producing a report containing monthly sales of all retail stores in the northwest region.

**Data Flow**

A data-flow is a path for data to move from one part of the information system to another. A data-flow may represent a single data element such the Customer ID or it can represent a set of data element (or a data structure).

**Data Store**

A data store or data repository is used in a data-flow diagram to represent a situation when the system must retain data because one or more processes need to use the stored data in a later time.

**External Entity**

An external entity is a person, department, outside organization, or other information system that provides data to the system or receives outputs from the system. External entities are components outside of the boundaries of the information systems.

### Context DFD

A context diagram is a top level (also known as "Level 0") data flow diagram. It only contains one process node ("Process 0") that generalizes the function of the entire system in relationship to external entities.

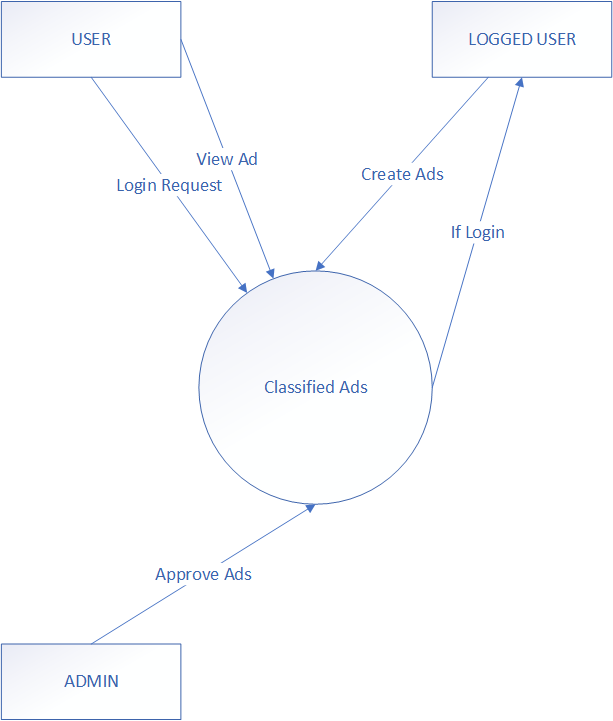


Figure 5 of DFD 0 level

This is simple Level 0 DFD with only one process

### Level 1 DFD

A level 1 data flow diagram (DFD) is more detailed than a level 0 DFD but not as detailed as a level 2 DFD. It breaks down the main processes into subprocesses that can then be analyzed and improved on a more intimate level.

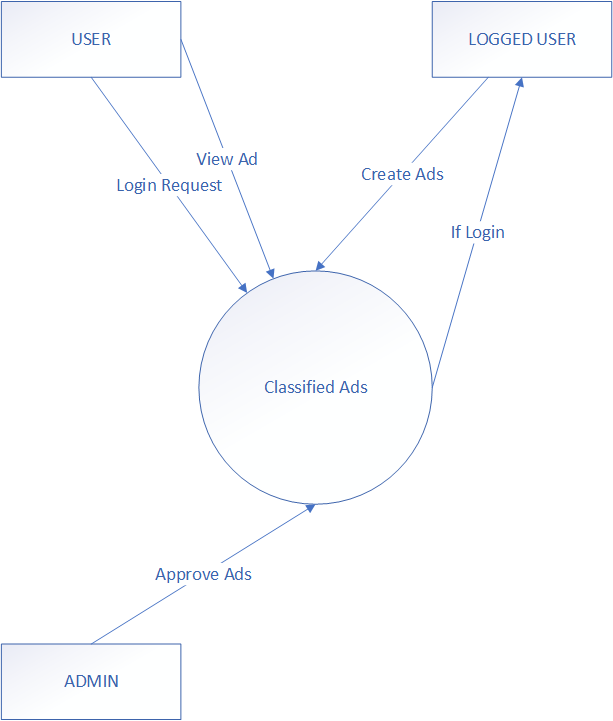


Figure 6 of DFD Level 1

A more detailed DFD of Big Store

## Use Case Diagram

The purpose of a use case diagram in UML is to demonstrate the different ways that a user might interact with a system. In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

* Scenarios in which your system or application interacts with people, organizations, or external systems
* Goals that your system or application helps those entities (known as actors) achieve
* The scope of your system

When to apply use case diagrams

A use case diagram doesn't go into a lot of detail—for example, don't expect it to model the order in which steps are performed. Instead, a proper use case diagram depicts a high-level overview of the relationship between use cases, actors, and systems. Experts recommend that use case diagrams be used to supplement a more descriptive textual use case.

UML is the modeling toolkit that you can use to build your diagrams. Use cases are represented with a labeled oval shape. Stick figures represent actors in the process, and the actor's participation in the system is modeled with a line between the actor and use case. To depict the system boundary, draw a box around the use case itself.

UML use case diagrams are ideal for:

* Representing the goals of system-user interactions
* Defining and organizing functional requirements in a system
* Specifying the context and requirements of a system
* Modeling the basic flow of events in a use case

User

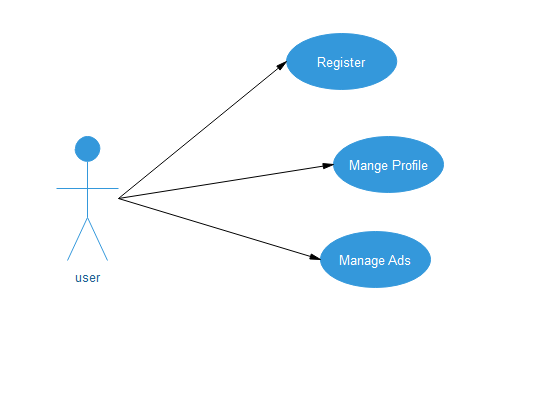


Figure 7 Use Case Diagram

Admin

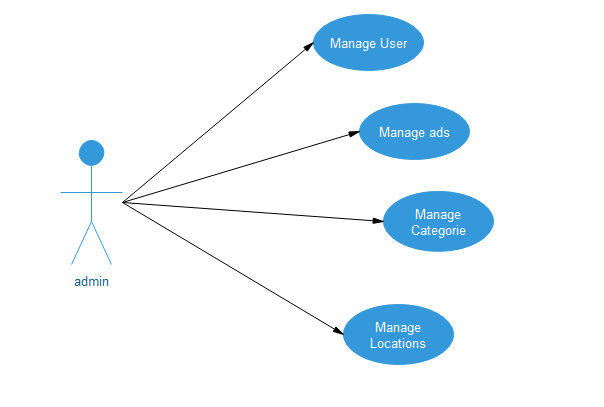


Figure 8 of use case diagram for admin

## Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

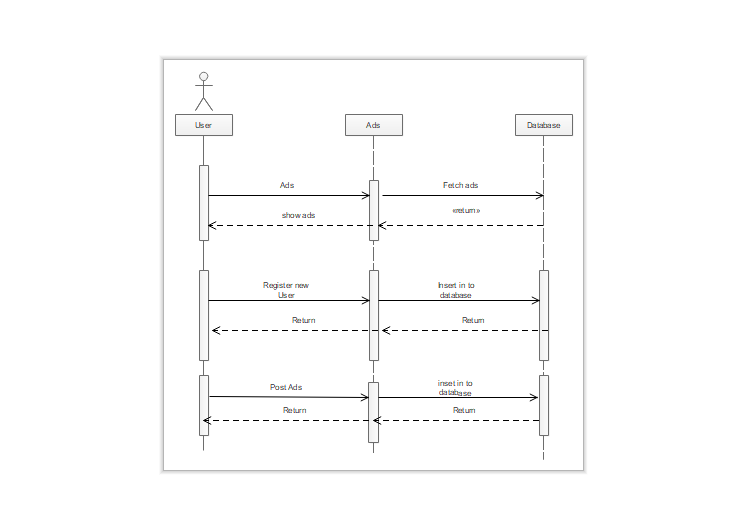


Figure 9 sequence diagram

## Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

In the diagram, classes are represented with boxes that contain three compartments:

The top compartment contains the name of the class. It is printed in bold and centered, and the first letter is capitalized. The middle compartment contains the attributes of the class. They are left-aligned and the first letter is lowercase. The bottom compartment contains the operations the class can execute. They are also left-aligned and the first letter is lowercase.

A class with three compartments.

In the design of a system, a number of classes are identified and grouped together in a class diagram that helps to determine the static relations between them. With detailed modelling, the classes of the conceptual design are often split into a number of subclasses.

In order to further describe the behavior of systems, these class diagrams can be complemented by a state diagram or UML state machine.

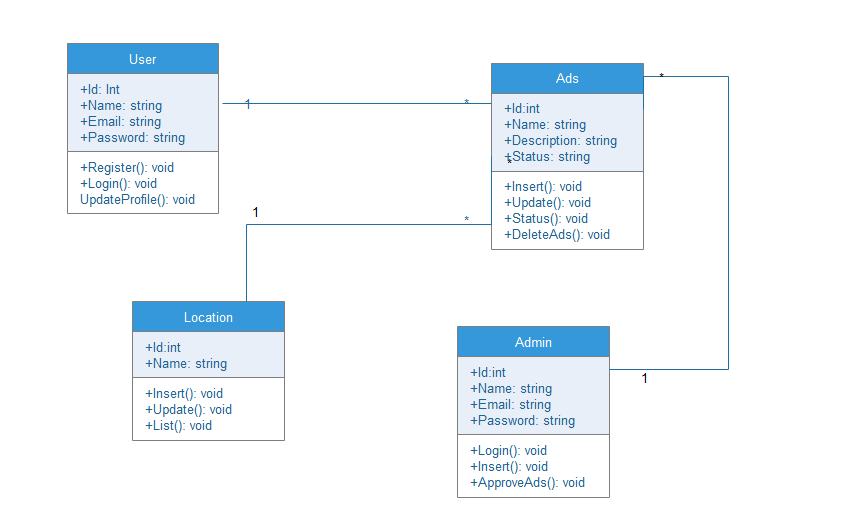


Figure 10 of class diagram of database

## ER Diagram

An entity-relationship diagram (ERD) is a graphical representation of an information system that shows the relationship between people, objects, places, concepts or events within that system. An ERD is a data modeling technique that can help define business processes and can be used as the foundation for a relational database.

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 numbers.

For example, an ER diagram representing the information system for a company's sales department might start with graphical representations of entities such as the sales representative, the customer, the customer's address, the customer's order, the ads and the warehouse. (See diagram) Then lines or other symbols can be used to represent the relationship between entities, and text can be used to label the relationships.

Finally, cardinality notations define the attributes of the relationship between the entities. Cardinalities can denote that an entity is optional (for example, a sales rep could have no customers or could have many) or mandatory (for example, the must be at least one ads listed in an order.)

Relationships: A relationship, in the context of databases, is a situation that exists between two relational database tables when one table has a foreign key that references the primary key of the other table. Relationships allow relational databases to split and store data in different tables, while linking disparate data items.

Cardinality and ordinality, respectively, refer to the maximum number of times an instance in one entity can be associated with instances in the related entity, and the minimum number of times an instance in one entity can be associated with an instance in the related entity. Cardinality and ordinality are

When it comes to notation, data modelers have many options to choose from. While crow's foot notation is widely accepted as the most intuitive style, some developers use OMT, IDEF, Bachman, or UML notation to indicate cardinality. Since crow's foot notation shows both minimum and maximum cardinality in an easy-to-read graphic format.

The three main cardinal relationships are:

One-to-one (1:1). For example, if each customer in a database is associated with one

mailing address.

One-to-many (1:M). For example, a single customer might place an order for multiple adss. The customer is associated with multiple entities, but all those entities have a single connection back to the same customer.

Many-to-many (M:N). For example, at a company where all call center agents work with multiple customers, each agent is associated with multiple customers, and multiple customers might also be associated with multiple agents.

While there are tools to help draw entity-relationship diagrams, such as CASE (computer-aided software engineering) tools, some relational database management systems also have design capabilities built in.

Here are some best practice tips for constructing an ERD:

* **Identify the entities.** The first step in making an ERD is to identify all of the entities you will use. An entity is nothing more than a rectangle with a description of something that your system stores information about.
* **Identify relationships**. Look at two entities, are they related? If so draw a solid line connecting the two entities.
* **Describe the relationship**. How are the entities related? Draw an action diamond between the two entities on the line you just added. In the diamond write a brief description of how they are related.
* **Add attributes.** Any key attributes of entities should be added using oval-shaped symbols.
* **Complete the diagram.** Continue to connect the entities with lines, and adding diamonds to describe each relationship until all relationships have been described.

**Tips for Effective ER Diagrams**

* Make sure that each entity only appears once per diagram.
* Name every entity, relationship, and attribute on your diagram.
* Examine relationships between entities closely.

## Database

A database is an organized collection of data, stored and accessed electronically. Database designers typically organize the data to model aspects of reality in a way that supports processes requiring information, such as (for example) modeling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

### Database Schema

A database schema of a database system is its structure described in a formal language supported by the database management system (DBMS). The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database. These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realization of the database language. The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modeled in the database.

"A database schema specifies, based on the database administrator's knowledge of possible applications, the facts that can enter the database, or those of interest to the possible end-users." The notion of a database schema plays the same role as the notion of theory in predicate calculus. A model of this "theory" closely corresponds to a database, which can be seen at any instant of time as a mathematical object. Thus, a schema can contain formulas representing integrity constraints specifically for an application and the constraints specifically for a type of database, all expressed in the same database language. In a relational database, the schema defines the tables, fields, relationships, views, indexes, packages, procedures, functions, queues, triggers, types, sequences, materialized, views, synonyms, database links, directories, XML schema, and other elements.

A database generally stores its schema in a data dictionary. Although a schema is defined in text database language, the term is often used to refer to a graphical depiction of the database structure. In other words, schema is the structure of the database that defines the objects in the database.

**Table**

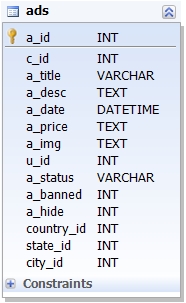


Figure 11of Ads table in database

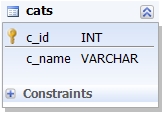


Figure of categories of database

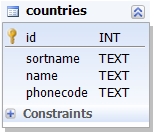


Figure 13 of countries table in database

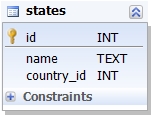


Figure 14 of states table in database

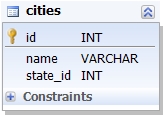


Figure 15 of cities table in database

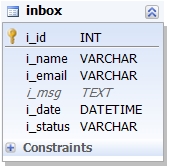


Figure 16 of inbox table in database

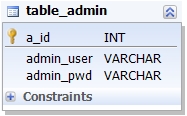


Figure 17 of admin table in database

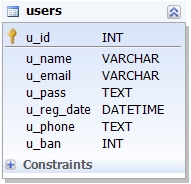


Figure 18 of user table in database

**Schema**

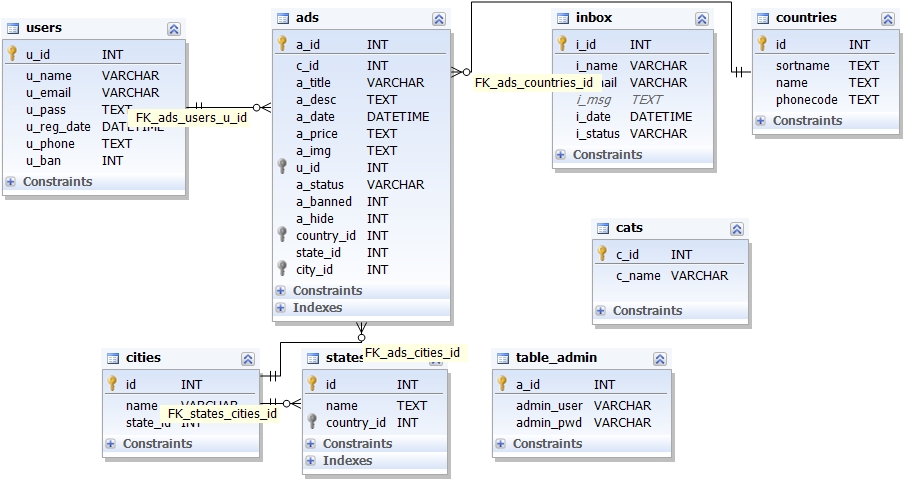


Figure 19 of Schema diagram in database

## Normalization

Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

Normalization is used for mainly two purposes,

* Eliminating redundant(useless) data.
* Ensuring data dependencies make sense i.e data is logically stored.

The inventor of the relational model Edgar Codd proposed the theory of normalization with the introduction of First Normal Form, and he continued to extend theory with Second and Third Normal Form. Later he joined with Raymond F. Boyce to develop the theory of Boyce-Codd Normal Form.

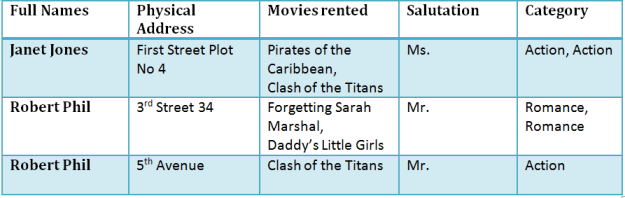
Theory of Data Normalization in SQL is still being developed further. For example, there are discussions even on 6th Normal Form. However, in most practical applications, normalization achieves its best in 3rd Normal Form. The evolution of Normalization theories is illustrated below-

[What is Normalization? 1NF, 2NF, 3NF & BCNF with Examples](https://www.guru99.com/images/NormalizationProcess(1).png)

Figure 20 of types of normalization in database

### Normalization Types with Examples

Assume a video library maintains a database of movies rented out. Without any normalization, all information is stored in one table as shown below.

[[](https://www.guru99.com/images/NormalizationTable1.png)](https://www.guru99.com/images/NormalizationTable1.png)

[Figure 21 types of normalization](https://www.guru99.com/images/NormalizationTable1.png)

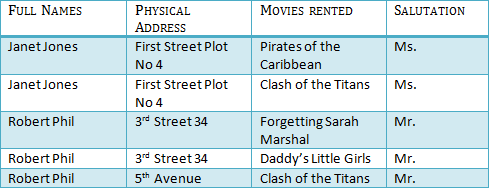
**1NF (First Normal Form) Rules**

Each table cell should contain a single value.

Each record needs to be unique.

The above table in 1NF-

1NF Example

[[](https://www.guru99.com/images/1NF.png)](https://www.guru99.com/images/1NF.png)

[Figure 22 of NF](https://www.guru99.com/images/1NF.png)

2NF (Second Normal Form) Rules

Rule 1- Be in 1NF

Rule 2- Single Column Primary Key

It is clear that we can't move forward to make our simple database in 2nd Normalization form unless we partition the table above.

[[](https://www.guru99.com/images/Table2.png)](https://www.guru99.com/images/Table2.png)

[Figure 23 of 2NF](https://www.guru99.com/images/Table2.png)

[[](https://www.guru99.com/images/Table1.png)](https://www.guru99.com/images/Table1.png)

[Figure 24 After apply 2NF](https://www.guru99.com/images/Table1.png)

We have divided our 1NF table into two tables viz. Table 1 and Table2. Table 1 contains member information. Table 2 contains information on movies rented.

We have introduced a new column called Membership which is the primary key for table 1. Records can be uniquely identified in Table 1 using membership id

**3NF (Third Normal Form) Rules**

Rule 1- Be in 2NF

Rule 2- Has no transitive functional dependencies

To move our 2NF table into 3NF, we again need to again divide our table.

3NF Example

[[](https://www.guru99.com/images/2NFTable1.png)](https://www.guru99.com/images/2NFTable1.png)

[[](https://www.guru99.com/images/2NFTable2.png)](https://www.guru99.com/images/2NFTable2.png)

[](https://www.guru99.com/images/2NFTable3.png)

[Figure 25 of 3NF](https://www.guru99.com/images/2NFTable3.png)

We have again divided our tables and created a new table which stores Salutations.

There are no transitive functional dependencies, and hence our table is in 3NF

In Table 3 Salutation ID is primary key, and in Table 1 Salutation ID is foreign to primary key in Table 3

Now our little example is at a level that cannot further be decomposed to attain higher forms of normalization. In fact, it is already in higher normalization forms. Separate efforts for moving into next levels of normalizing data are normally needed in complex databases.  However, we will be discussing next levels of normalizations in brief in the following.

# Development



## What is Incremental Model?

Incremental Model is a process of software development where requirements are broken down into multiple standalone modules of software development cycle. Incremental development is done in steps from analysis design, implementation, testing/verification, maintenance. Each iteration passes through the requirements, design, coding and testing phases. And each subsequent release of the system adds function to the previous release until all designed functionality has been implemented.

[](https://www.guru99.com/images/6-2015/052615_1049_WhatisIncre2.png)

Figure 26 of Incremental Model

The system is put into Adson when the first increment is delivered. The first increment is often a core ad where the basic requirements are addressed, and supplementary features are added in the next increments. Once the core ads are analyzed by the client, there is plan development for the next increment.

## Plate form

Any windows-based operating system (windows 7 or 8). XAMPP Sever for running PHP and MYSQL database. IDE (Dreamweaver) for developing code. XAMPP Server is a Windows web development environment. It allows to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your databases.

One of the most difficult tasks after system requirements are known is determining whether particular whether particular software is capable of meeting the system requirements or not. For those that do so further scrutiny is needed to determine the desirability in comparison with other conditions. The choice of software is a very important factor to be considered during the development phase of the new system because new system is totally dependent on the software it using. This decision depends upon many factors including the current environment, amount of data to handle and the cost of programming.

HTTP stand for Hypertext Transfer Protocol is a transaction or oriented client/server protocol between web browser & a Web Server.

## Language

**PHP**

PHP is the server-side web language platform. PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical application.

**HTML**

HTML stands for Hyper Text Markup Language, is a markup language for web pages. It provides a means to create structured documents including headings, pictures, objects, lists, links, and other items and can be used to create interactive pages. It can include or can load scripts in languages such as JAVA SCRIPT which affects the behavior of HTMML processors like Web Browsers.

**CSS**

CSS (Cascading Style Sheets) is used to style and lay out web pages — for example, to alter the font, color, size and spacing of your content, split it into multiple columns, or add animations and other decorative features. This module gets you started on the path to CSS mastery with the basics of how it works, including selectors and properties, writing CSS rules, applying CSS to HTML, how to specify length, color, and other units in CSS, cascade and inheritance, and debugging CSS.

**JavaScript**

JavaScript is a cross-platform, object-oriented scripting language used to make webpages interactive (e.x. having complex animations, clickable buttons, popup menus, etc.). There are also more advanced server side versions of JavaScript such as Node.js which allow you to add more functionality to a website than simply downloading files (such as Realtime collaboration between multiple computers). Inside a host environment (for example, a web browser), JavaScript can be connected to the objects of its environment to provide programmatic control over them.

JavaScript contains a standard library of objects, such as Array, Date, and Math, and a core set of language elements such as operators, control structures, and statements. Core JavaScript can be extended for a variety of purposes by supplementing it with additional objects.

# Deployment



## Implementation

Implementation is any method, execution, or a repetition of a plan, and any design, or idea and requirement, standard and the way of doing something is call implementation process. we can say that the implementation is the any action that must be follow any initial thinking in order to actually happened. As such, implementation is the action that must follow any preliminary thinking in order for something to actually happen. In this chapter we explain the making steps of designing and implementation phase of project Big Store. And the system testing has to be done to minimize the programming and system error in the project Big Store. Then at the implementation phase the system requirements in which hardware and the software will be clear in detail in the project Big Store. Alongside that the system prototype interfaces and function (modules) will be fully established to the users whom use the system of Big Store for all internet users.

The proposed Big Store can perform tasks in effective management of internet users. The major aim of the proposed system is to increase the salles or buy service fast, comfortable, reliable and effective way. The system will record data in the database, display details of goods and services.

## Configuration

Testing of the system

System testing is a process where we check the problem and where complete and integrated software are tested. The drive of this stem is to assess the system agreement with the exact requirement. System testing is a very important step where we check all the problems of software which have requirement and errors accrue in system completion. System testing of software can be done in some stages. The first step is called the unit testing and the component testing. This step is done when the system is developing. Each components script or module test isolate from the other components or the unit will be checking for the input and output of the system of developing. The second stage is called integration of testing. In this step the integration of system will be check. And if any error accrues the components will be check again. The third step is called user acceptance testing and this testing done by user who will request to develop the system. and next stage is called security testing. this is the final stage in any system developing. it also called user acceptance testing this testing will done by the user who will request to develop the system

### Unit Testing

Unit testing is a software development process in which the smallest testable parts of an application, called units, are individually and independently analyzed for proper operation. Unit testing is often automated but it can also be done manually.  Unit testing is a [software testing](https://en.wikipedia.org/wiki/Software_testing) method by which individual units of [source code](https://en.wikipedia.org/wiki/Source_code), sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. This approach was taken to ensure the modules meet the objectives defined and the units work appropriately.

### Integration Testing

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. This testing approach was conducted once individual program units had been tested.

### User Acceptance Testing

In software development, user acceptance testing (UAT) - also called application testing, and end user testing - is a phase of software development in which the intended audience tests the software in the “real world”.The purpose of this test is to evaluate the system’s compliance with the business requirements and assess whether it is acceptable for delivery.Acceptance testing or user testing is the final testing procedure in Information System. Users will be actively involved in testing of Management System to ensure that it meets their requirements.

## Test cases

### . Test Case 1: Admin Login

|  |  |
| --- | --- |
| **Identifier** | Admin login TC-1 |
| **Purpose** | Admin will provide with user name and password in the login form textbox fields. |
| **Priority** | High |
| **Pre-Conditions** | User name and password is necessary to login. |
| **Input Data** | User name and password. |
| **Detail Steps** | Admin will enter the:  User name and   1. Password |
| **Expected Results** | 1. User validation 2. User name validation 3. Password validation |
| **Post-Conditions** | Admin will be taken to the next form: Home Page. |

### 5.5.2. Test Case 2: Add new Ads

|  |  |
| --- | --- |
| **Identifier** | Admin can add ads TC-2 |
| **Purpose** | Admin will add ads. |
| **Priority** | High |
| **Pre-Conditions** | User name and password is necessary to login. |
| **Input Data** | Fill the required form |
| **Detail Steps** | Admin will enter the:  User name and  Password |
| **Post-Conditions** | After add is successful message will be display: Welcome. |

Table 5 of test case 2 of admin login

### 5.5.3. Test Case 3: Delete ads

|  |  |
| --- | --- |
| **Identifier** | Admin can delete ads TC-3 |
| **Purpose** | Admin will delete ads |
| **Priority** | High |
| **Pre-Conditions** | Ads ID must be known by admin to delete. |
| **Input Data** | Ads admin ID |
| **Detail Steps** | Admin will enter the:  Ads ID |
| **Expected Results** | Ads will be delete  Database will be updated |
| **Post-Conditions** | After delete is successful message will be display: ads deleted successfully! |

Table 6 of test case 3 of delete ads

### Test Case 4: View ads list

|  |  |
| --- | --- |
| **Identifier** | Admin can view ads list TC-4 |
| **Purpose** | View total numbers of ads working in the website |
| **Priority** | High |
| **Pre-Conditions** | Admin will have to be logged in to the website to view the ads list. |
| **Input Data** | Button Click |
| **Detail Steps** | Admin will click the:  Ads List |
| **Expected Results** | Ads List will be displayed |
| **Post-Conditions** | After the list has been displayed admin can do whatever he/she decides! |

Table 7 of Test Case 4: View ads list

### 5.5.5. Test Case 5: Update Ads

|  |  |
| --- | --- |
| **Identifier** | Admin can update Ads |
| **Purpose** | Admin will update Ads |
| **Priority** | High |
| **Pre-Conditions** | Ads ID must be known by admin to update. |
| **Input Data** | Ads ID |
| **Detail Steps** | Admin will enter the:  Ads ID |
| **Expected Results** | Ads will be updated  Database will be updated |
| **Post-Conditions** | After update is successful message will be display: Ads updated successfully! |

Table 8 of Test Case 5: Update Ads

### Test cases

Test case title: ads post

|  |  |
| --- | --- |
| **Test case1: ads post** | **Priority (H, L):** High |
| **Test objective:** user insert post to system | |
| **Test descriptions: “**user enter the required field, press add button”, client program contactto the server, server contact with database, database insert data in to the database. | |
| **Requirements verified: No**  **xz** | |
| **Test environment: Apache** must be in running state, Database should contain appropriate table andlink must be established between server and client program. | |
| **Test setup/ pre-condition: Apache** must be in running state. The entire mandatory field must beentered. | |
| **Location** | **Expected result** |
| The user show the post in his/her panel to application | “Post ads successfully”. Display main menu. |
| **Pass:** Yes  **Condition**: Pass **Fail:** No | |
| **Problem /issues:** NIL | |
| **status:** successfully executed | |

Table 5 of test case for add post

Table 6 of test case for add post

### Test cases

Test case title: Ads Status

|  |  |
| --- | --- |
| **Test case1: Change ads status** | **Priority(H,L):** High |
| **Test objective: Change status of ads** | |
| **Test descriptions: user open the dashboard of Big Store after registration and login. User can change status of ads.** | |
| **Requirements verified:** Yes  **xz** | |
| **Test environment:** Apachemust be in running state, Database should contain appropriate table andlink must be established between server and client program. | |
| **Test setup/ pre-condition:**  Apache must be in running state. The entire mandatory field must beentered. | |
| **Location** | **Expected result** |
| The user will change the status of ads | “Change status successfully”. Display all ads. |
| **Pass:** Yes  **Condition**: Pass **Fail:** No | |
| **Problem /issues:** NIL | |
| **status:** successfully executed | |

Table 7 of test case for Ads Status

## User Interface

There are some screen shots mention to show my project pages.

## Admin page

There is admin page to use admin login.

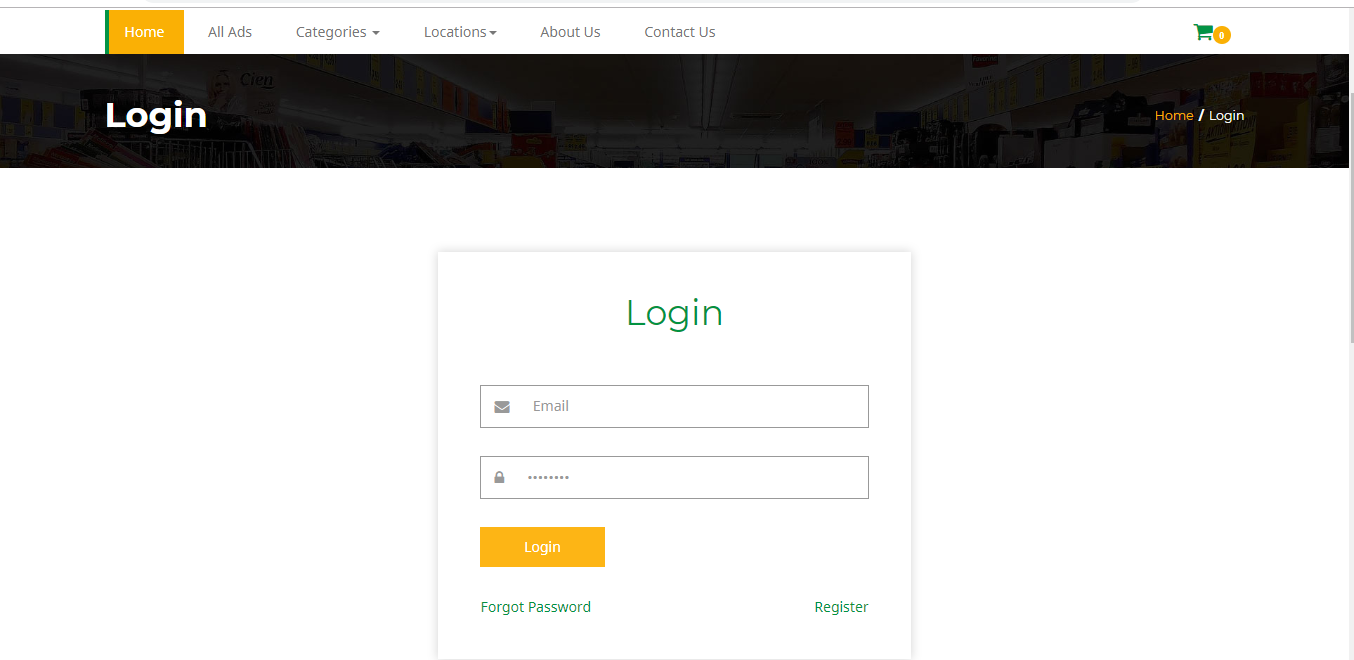


Figure 27 of login page

## Customer Tab

In this picture Show Customer services.

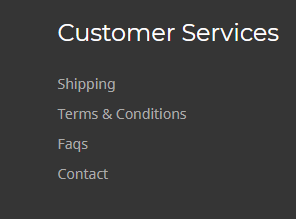


Figure 28 of Customer services

## Menu Tab

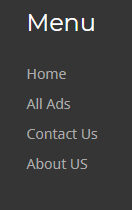


Figure 29of menu tab

## Dashboard page

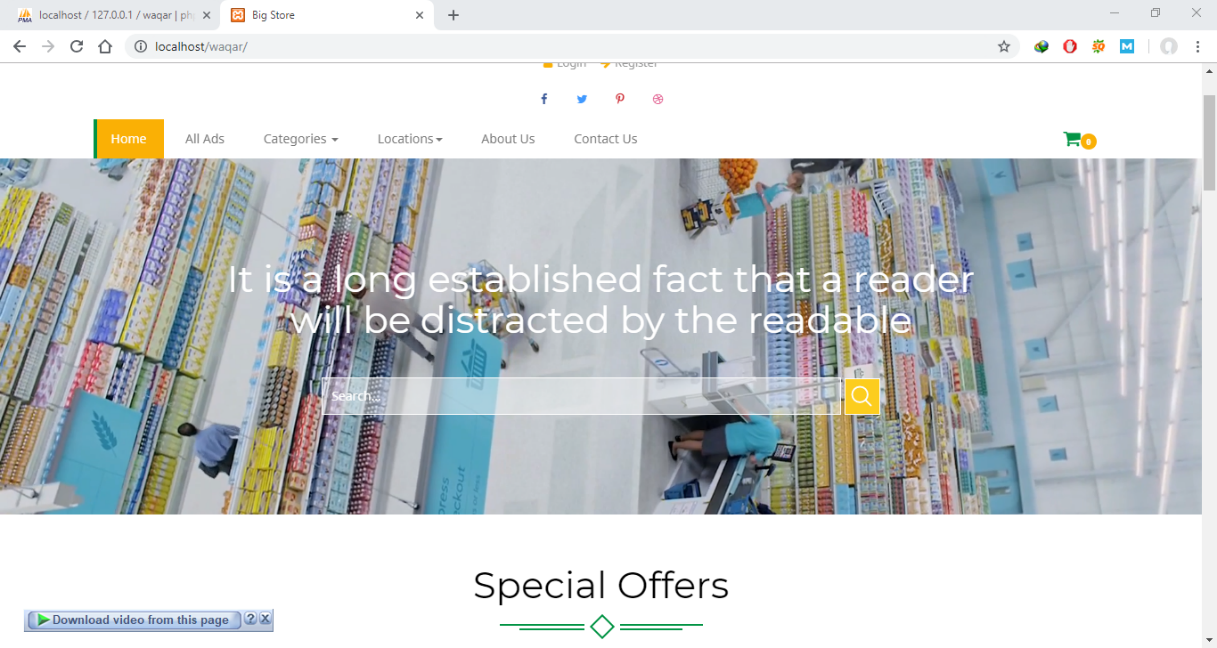


Figure 30 of dashboard of my project

## Categories Page

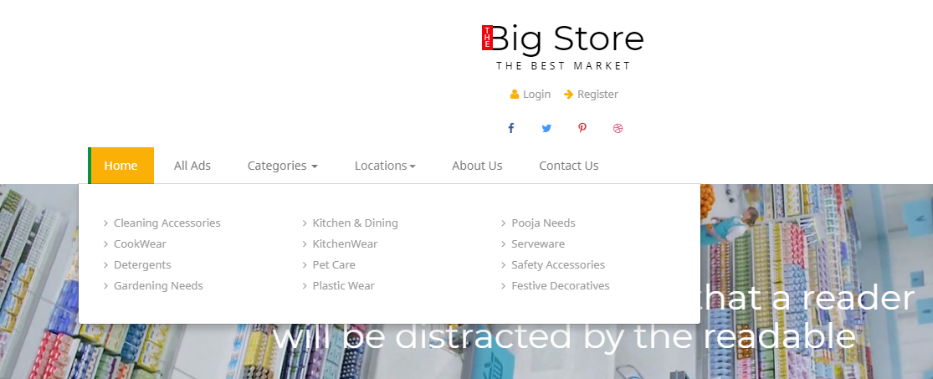


Figure 31 of categories page

## Contact us page

In this screen shot show Contact us page



Figure 32 of contact us page

## Location page

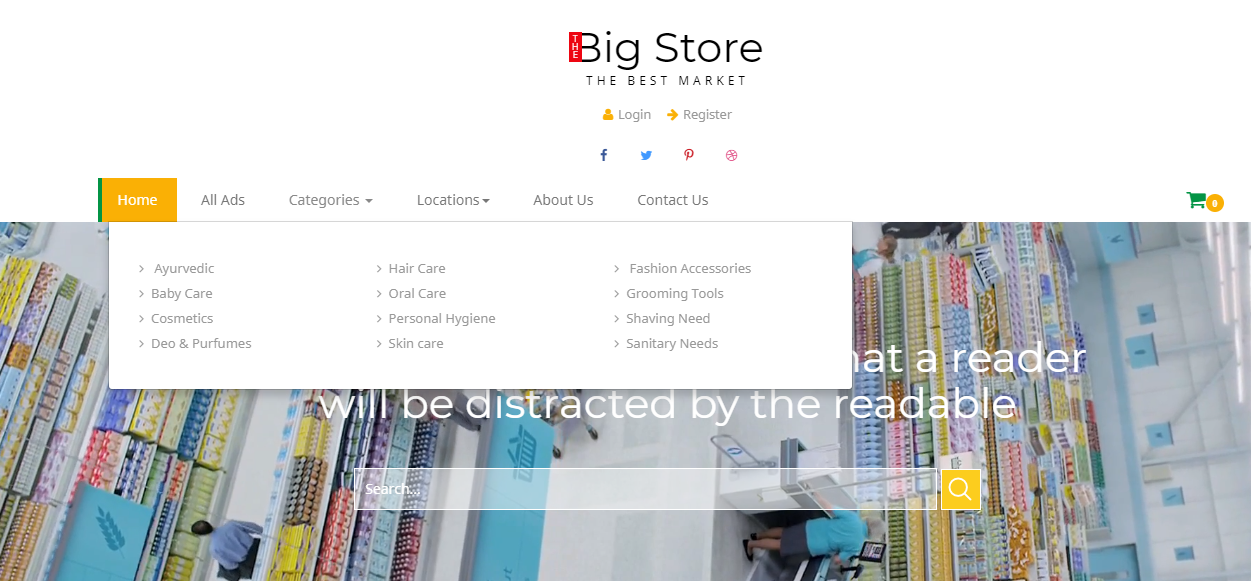


Figure 33of location page

## About us page

About us page show in this picture

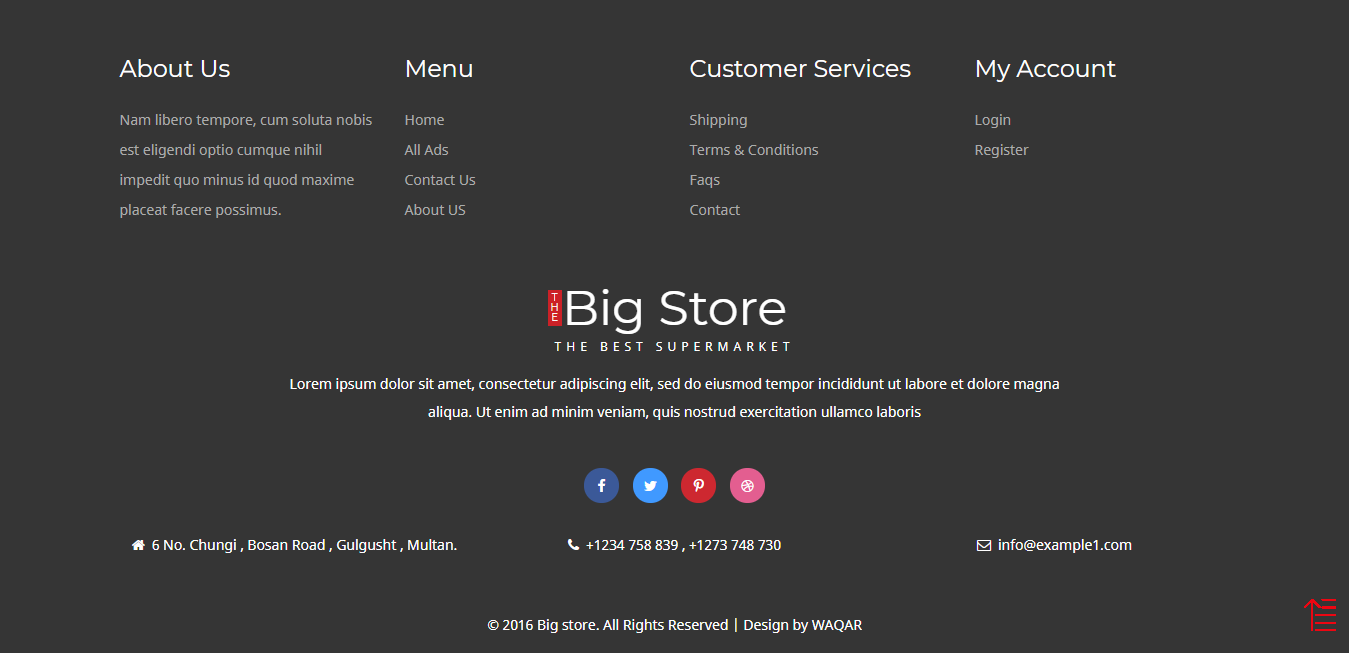


Figure 34of about us page

## 

# Conclusion & References

## 6.1. Conclusion

It was first time I worked on actual project on big scale. It was a hard task working on interface and design. I took help from video tutorials and teacher’s guidance helped a lot.

It needs a lot practice. It needs a lot of time. Since giving proper time to the language and practicing it over and over again I now know how it works and have grip over it. Development skills and testing practice has enhanced the project scope. Creativity and idea generation is developed by design and user interface of the project.

I have enhanced my programming skills and my error detecting skill is enhanced as well. I now know how to get authenticate information and provide good results using that information in an appropriate way. Sir/Ma’am you are honorably welcome to suggest changes and amendments need to be made in this project for further reasons.

## 6.2. References

<https://www.tutorialspoint.com/dbms/database_normalization.htm>

<https://en.wikipedia.org/wiki/Systems_development_life_cycle>

<http://softwaretestingfundamentals.com/unit-testing/>

<https://erdplus.com/#/>