**INTRODUCTION**

Today, the Internet is public, cooperative and self-sustaining facility accessible to hundreds of millions of people worldwide. This is due to its importance in everyday life. For instance, it allows information transferring, information sharing as well as communication. Sometimes, it also enables user from one location talk directly to users at other computers in a different location.

Then they inquire the shop for the material with the set of filters and then they have the option to order the material with specifying the number item and address or they can pick themself. The order will be registered with the system and then the shop vendor will collect items from the racks of shops according to the order. I have selected to do the project on E-Commerce.

**ABSTRACT**

This website of Jewellery shop allows users to check for various Jewellery items available at the store and help them to purchase online. The project consists of list of Jewellery products displayed in various models and designs. The user may browse through these products as per categories. If the user likes a product he may add it to his shopping cart. Once user wishes to checkout he must register on the site first. He can then login using same id password next time. Now he may pay through a credit card or cash on delivery. Once the user makes a successful transaction he gets a copy of the shopping receipt on his email id. Here we use inbuilt system framework to make the entire frontend. The middle tier or code behind model is designed for a strong processing support. And SQL serves as a backend to store Jewellery lists and inventory data .

Thus, this online Jewellery shopping project brings an entire Jewellery shop online and makes it easy for both buyers and sellers to make Jewellery deals.

**OBJECTIVES**

**PROJECT CATEGORY**

**RDBMS**

**The Project is developed using Relational Database Management System (RDBMS).**

A database system is essentially a sophisticated, computerized record keeping system, a repository for a collection of computerized data files. A database system maintains information and makes that information available on demands, for this purpose a database system provides see of facilities to perform such operations. The benefits of a database system over any traditional system are obvious as database is integrated as well as shared, thus a database eliminates redundancy and also as a consequence, database lets multiple users access the same piece of data.

The most important advantage of the database is to maintain the integrity i.e. it insures that the change made to the database by authorized users do not result in a loss of data consistency and guard against accidental damage to the database.

**Facilities Offered by RDBMS:**

* + Creation of files, Addition of data, Deletion of data, Modification of data.
  + Retrieving of collectively or selectively.
  + The data stored can be sorted or indexed at user’s discretion or direction.
  + Various reports can be produced from the system. These may either be standardized reports or that may be specifically generated according to specific user definition.
  + Mathematical function can be performed and the data stored in the database can be manipulated with functions to perform the desired calculations.
  + To maintain data integrity and database use.
  + Data integrity from multiple users.
  + Providing form based interface for easy accessibility and data entry.

**Tools/ Platform, Hardware & Software Requirement Specification**

**Software requirements:**

**Operating System:** Window XP, 2003, 2007

**Platform:** Java (JDK1.8)

**Front-End:** HTML5, CSS3, Bootstrap

**Back-End:** MySQL

**Server:** Apache Tomcat Server

**IDE:** Net Beans

**Browser:** Internet Explorer, Google Chrome, Mozilla Firefox etc.

**Hardware Requirements:**

**Monitor:** VDU

**Processor:** P4 Processor having (550 to 933 MHZ)

**RAM:** 128 MB RAM

**HDD:** 20 GB HD

**MODEM:** 36.6 KBPS

**PRINTER:** DeskJet

**CD-ROM DRIVE:** 52X

**SRS and Software Model**

**About PHP**

* The PHP Hypertext Preprocessor (PHP) is a programming language that allows web developers to create dynamic content that interacts with databases.
* PHP is basically used for developing web based software applications.
* It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.

**About HTML**

* HTML stands for Hyper Text Markup Language
* HTML is the standard markup language for creating Web pages
* HTML describes the structure of a Web page

**About CSS**

* CSS stands for Cascading Style Sheets
* CSS describes how HTML elements are to be displayed on screen, paper or in other media.
* External stylesheets are stored in CSS files

**About Bootstrap**

* Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development.
* It contains HTML, CSS and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

**SRS Document**

It is a reference document or contract between the customer and the development team. Once the customer agrees to the SRS document, the development team proceeds to develop the product conforming to all the requirements mentioned in the SRS document.

An SRS document should clearly document the following:

1. **Functional requirements of the system:** Each function of the system can be considered as performing a transformation of a set of input data to the corresponding set of output data. The functional requirements of the system should clearly describe each of the functions that the system needs to perform along with the corresponding input and output data set.
2. **Non-functional requirements of the system:** Non-functional requirements deal with the characteristics of the system that cannot be expressed functionally, e.g., maintainability, portability, Usability, etc. The non-functional requirements also include reliability issues, accuracy of results, human computer interface issues, operating and Physical constraints, etc.
3. **Constraints on the system:** The constraints on the use of the system may describe certain things that the system should or should not do.

**Nature of SRS**

The basic issues the SRS writer(s) shall address are the following:

1. **Functionality:** What the software is supposed to do?
2. **External Interfaces:** How does the software interact with people, the system’s hardware ,other hardware and other software.
3. **Performance:** What is the speed, availability, response time, recovery time, etc., of the various software fundamentals.
4. **Attributes:** What are the consideration for portability, correctness, maintainability, security, reliability, etc.
5. **Design constraints imposed on an implementation:** Are there any required standards or effect, implementation language, policies for database, integrity resource limits, operating environment, etc.

**Characteristics of a good SRS**

An SRS should be

* Correct
* Unambiguous
* Complete
* Consistent
* Verifiable
* Modifiable
* Traceable
* Ranked for Importance and for Stability

**Correct:** There is no tool or procedure that assures correctness. If the software must respond to all button presses within 5 seconds and the SRS stated that “the software shall respond to all button presses within 10 seconds”, then that requirement is incorrect.

**Unambiguous:** An SRS is unambiguous if and only if every requirement started therein has only are interpretation. In cases, where a term used in a particular context could have multiple meanings, the term should be included in a glossary where its meaning is made more specific.

**Complete:** An SRS is complete if and only if it includes of the following elements.

1. All significant requirements, whether relating to functionality, performance, design constraints, attributes or external interfaces.
2. Full labels and references to all figures, tables and diagram in the SRS and definition of all terms and units of measure.

**Consistent:** An SRS is consistent if no subset of individual requirements desorbed in it conflict. There are 3 types of likely conflicts in an SRS:

1. The specified characteristics of real word objects may conflict, e.g.

a. The Format of an output report may be described in are requirements as tabular but in another as textual.

b. One requirement may state that all lights shall be green while another states that all lights should be blue.

2. There may be logical or temporal conflict between two specified actions, e.g.,

a. Are requirement may specify that the program will add 2 inputs and another may specify that the program will multiply them.

b. Are requirement may state that ‘A’ must always follows B, while another requires that A&B occur simultaneously.

3. Two or more requirements may describe the same real word object but use different terms for that object. The use of standard terminology and definitions promotes consistency.

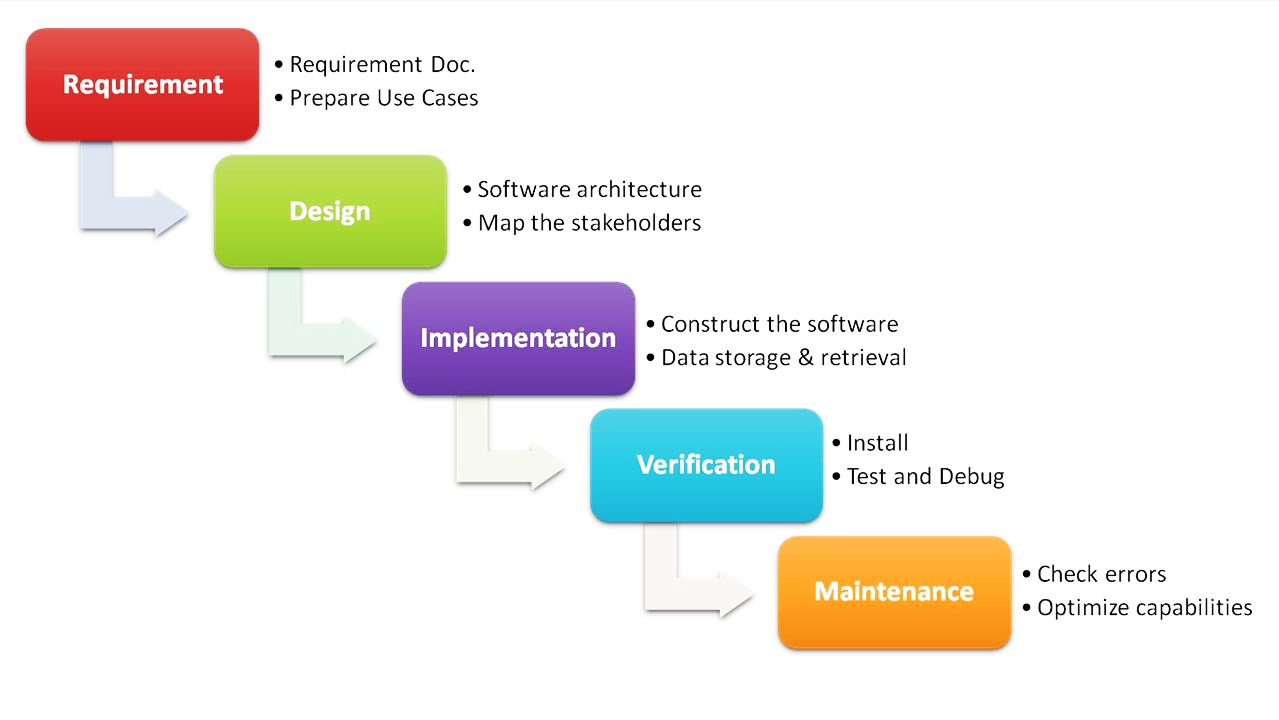
**Water Fall Model**

The waterfall model is a sequential design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall model) through the phases of Conception, Initiation, analysis, design, Construction, testing.

Production/implementation and maintenance.

The waterfall development model originates in the manufacturing and construction industries; highly structured physical environments in which after-the-fact changes are prohibitively costly, if not impossible. Since no formal software development methodologies existed at the time, this hardware-oriented model was simply adapted for software development.

This presentation was about the development of software for SAGE.



**Data Flow Diagrams (DFD)**

**ADMIN**

**CUSTOMER**

**DATABASE**

Register, Login and provide product details

Register and Login

Get Customer’s Details

Get product details

Store Customers and Products details

Retrieve Customers and Product details

**INPUT**  **OUTPUT**

**-------------------------------------** 0 Level DFD **----------------------------------**

**User Control**

**Maintained Information**

If Valid then

Check Retrieved Information

User ID and Password

**Show Details**

**------------------------------------** 1st Level DFD **----------------------------------**

**Information Maintained**

**Maintained Information**

**Show Details**

**Information Maintained**

**Show Details**

**Show Details**

**Information Maintained**

**Show Details**

**Information Maintained**

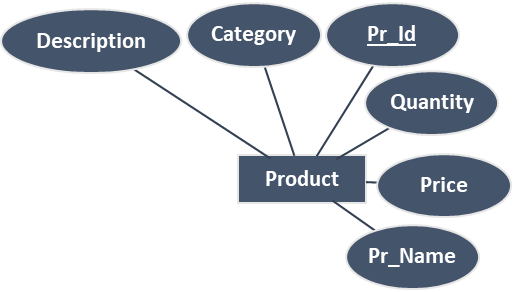
**Show Details**

**Information Maintained**

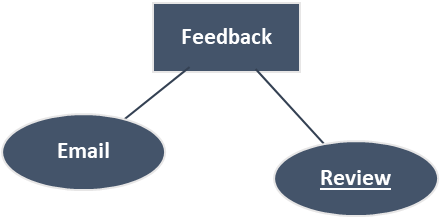
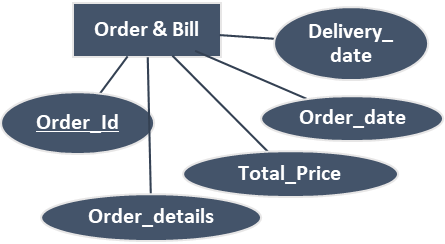
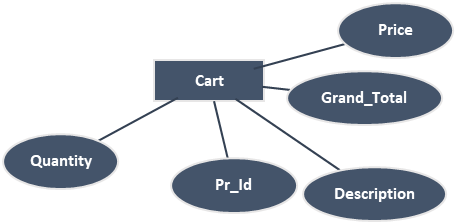
**------------------------------------** 2nd Level DFD **--------------------------------**

**Entity Relationship (ER) diagram**

**Maintains**



**AddedTo**

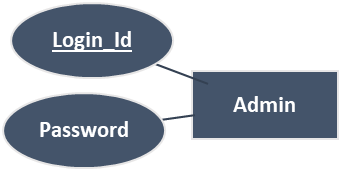


**View**

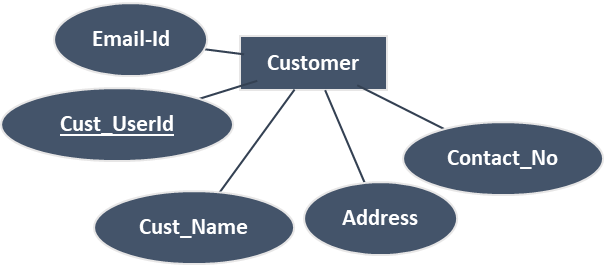
**Purchased**

**Received By**

**Gives**



**Searches**



**Conclusion**