

P-3267



# **CONTENT MANAGEMENT SYSTEM - EXTRANET PORTAL**

## **PROJECT REPORT**

*Submitted By*

**G. SURESH**

**Register No.: 0720300050**

*in partial fulfillment for the award of the degree  
of*

**MASTER OF COMPUTER APPLICATIONS**

**in**

**COMPUTER APPLICATIONS**

**KUMARAGURU COLLEGE OF TECHNOLOGY**

**(An Autonomous Institution Affiliated to Anna University, Coimbatore)**

**May, 2010**

# KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution Affiliated to Anna University, Coimbatore)

**COIMBATORE – 641 006.**

Department of Computer Applications

**PROJECT WORK**

**MAY 2010**

This is to certify that the project entitled  
**CONTENT MANAGEMENT SYSTEM - EXTRANET PORTAL**  
is the bonafide record of project work done by

**G. SURESH**

**Register No: 0720300050**

of MCA (Computer Applications) during the year 2009-2010.



\_\_\_\_\_

Project Guide



\_\_\_\_\_

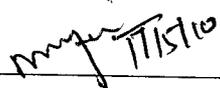
Head of the Department

Submitted for the Project Viva-Voce examination held on 17/05/2010



\_\_\_\_\_

Internal Examiner

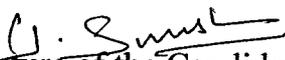


\_\_\_\_\_

External Examiner

**DECLARATION**

I affirm that the project work titled **CONTENT MANAGEMENT SYSTEM - EXTRANET PORTAL** being submitted in partial fulfillment for the award of **MASTER OF COMPUTER APPLICATIONS** is the original work carried out by me. It has not formed the part of any other project work submitted for award of any degree or diploma, either in this or any other University.

  
(Signature of the Candidate)

G. SURESH

072030050

I certify that the declaration made above by the candidate is true

  
Signature of the Guide,  
Parameswarar P., ST/MCA

With Name & Designation



14 – NITHYANANDA ST,  
IRUMBULIYUR,  
WEST TAMBARAM,  
CHENNAI.

Phone: +044 - 263625

[www.sparkglobalsolutions.com](http://www.sparkglobalsolutions.com)

April 23, 2010

### **To Whom It May Concern**

We are providing this certificate to **Mr.Suresh.G (Reg.No:0720300050)**, MCA department from "**Kumaraguru College Of Technology**" for the project "**Content Management System-Extranet Portal**" Completed with our extreme Organization.

**Project Duration: December 2009 to April 2010**

The developed software have been tested and forwarded to User Acceptance Testing. After Successful implementation of the application we are issuing this certificate.

**Swaminathan**  
**Vice President (Technology & global operations)**

## ACKNOWLEDGEMENT

First of all, I wish to thank the almighty who have blessed me with good health and wealth to carry out this project successfully.

I wish to express my sincere thanks to **Dr. S. Ramachandran**, Principal, Kumaraguru College of Technology, Coimbatore, for permitting me to undertake this project.

I wish to express earnest thanks to **Dr. Thangasamy**, Dean and **Dr. A. Muthukumar**, course co-ordinator, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore, for their support and encouragement.

My deepest acknowledgement to the project co-ordinator **Mr. S. Hameed Ibrahim**, Senior Lecturer, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore, for his timely remarks and comments about the project, which helps us to improve our knowledge and project.

I want to acknowledge my project guide, **Ms. P. Parameswari**, Senior Lecturer, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore, for his support and encouragement in making this project a success. His excellent suggestions and timely encouragement helped me complete the project and the project report.

I wish to express my gratitude to my Spark Global Solutions Project guide, **Mr. P. Swaminathan**, Vice President (Technology & global operations), Spark Global Solutions, Chennai, for his useful suggestions and support given throughout the project.

I wish to thank all my teaching and supporting staff members for their timely help to complete the project successfully.

My heartfelt thanks to my lovable family members and friends, who have provided the mental support for me throughout the course of this project work. Finally, I would like to thank all the people around me, who helped directly or indirectly to complete this project.

## ABSTRACT

This system “**CONTENT MANAGEMENT SYSTEM – EXTRANET PORTAL**” mainly aims at fetching the web content (articles) from the particular website and refreshes the content in order to sell the articles. It also focuses on the articles inward, outward, sales and inventory.

This system is maintained by the Administrator. Administrator can search the articles by category wise and that URL will be the input to our system. Administrator can also set the number of articles up to a limit.

Articles are fetched from the website and it is stored in to DB. Administrator inserted category to which the articles are belongs to. In rewrite of content section, category is chosen and the contents fetched in a category from the website are displayed with built in translators (Google, Microsoft translators). Using this, the content of the articles is optimized.

If the Administrator wants to insert content directly, then that is also possible. The articles belongs to a particular category which are rewritten are packed using title list of the articles. Also unpack of the packed content is also possible. Administrator downloads the articles in a particular format according to the client requirement such as ‘pdf’, ‘doc’ and ‘txt’ to sell the articles by pack wise. Re-download is also available for redundant sale.

Also this system focuses on reports such as whole report and also dates wise reports of sales, packs and categories.

## TABLE OF CONTENTS

Abstract	iv
List of Tables	vii
List of Figures	vii
1. Introduction	1
1.1 Organization Profile	1
1.2 Project Overview	3
2. Development Environment	4
2.1 Hardware Specification	4
2.2 Software Specification	4
2.3 About the Software	5
3. System Design and Analysis	14
3.1 Existing System	14
3.1.1 Drawbacks of the existing system	14
3.2 Proposed System	14
3.2.1 Advantages of Proposed System	15
4. System Design	16
4.1 Diagrams	16
4.1.1 Use-case diagram	16
4.1.2 Data Flow Diagram	18
4.2 Elements of Design	22
4.2.1 Input Design	22
4.2.2 Output Design	23
4.2.3 Database Structure	24
4.2.4 Modular Design	27

<b>5. System Implementation and Testing</b>	<b>30</b>
<b>5.1 System Implementation</b>	<b>30</b>
<b>5.2 System Verification</b>	<b>30</b>
<b>5.3 System Validation</b>	<b>31</b>
<b>5.4 Testing</b>	<b>31</b>
<b>5.4.1 Unit Testing</b>	<b>31</b>
<b>5.4.2 Integration Testing</b>	<b>32</b>
<b>6. Conclusion and Future Enhancement</b>	<b>33</b>
<b>6.1 Conclusion</b>	<b>33</b>
<b>6.2 Future Enhancement</b>	<b>34</b>
<b>Appendix</b>	<b>35</b>
<b>Screen Shots</b>	<b>35</b>
<b>References</b>	<b>40</b>

**List of Tables:**

<b>Table No.</b>	<b>Table Name</b>	<b>Page No.</b>
4.3(a)	cmp_searchurl	24
4.3(b)	cmp_category	25
4.3(c)	cmp_content	25
4.3(d)	cmp_package	26
4.3(e)	cmp_sales	26

**List of Figures:**

<b>No.</b>	<b>Diagram</b>	<b>Page No.</b>
1.	Use-case for CMS-Extranet Portal	17
2.	Dataflow Diagram for CMS – Extranet Portal (Level - 0)	19
3.	Dataflow Diagram for CMS – Extranet Portal (Level - 1)	20
4.	Dataflow Diagram for Pack / Unpack Activity (Level - 2)	21

**List of Abbreviations:**

<b>Abbreviation</b>	<b>Expansion</b>
IDE	Integrated Development Environment

## CHAPTER 1

### INTRODUCTION

#### 1.1. ORGANIZATION PROFILE

**Spark Global Solutions, Inc.** provides innovative, user-friendly software solutions to the legal industry and is committed to building easy-to-use cross-platform business and technology solutions.

Spark Global Solutions was founded with the vision that legal processes could be improved using tools that have been successfully applied in other business disciplines. Ongoing discussions with industry experts and clients ensure continued innovative solutions built to meet exacting legal process requirements.

By filling the capability gaps and investing in research and development, Spark Global Solutions has built the industry's foremost end-to-end platform for corporations and their outside counsel to holistically manage legal governance, risk and compliance processes.

Today, Spark Global Solutions is essential in ensuring faster, smarter, litigation response to e-discovery and regulatory compliance requests.

Spark Global Solutions looks at problems from new perspectives and incorporates best practices from disparate sources. Concepts and processes are constantly being scrutinized and re-examined using input from clients and industry experts to come up with better, faster and more efficient solutions. Through constant innovation and continual refinement, Spark Global Solutions remains the leading provider of robust software solution for today's complex legal world.

Spark Global Solutions is exceptionally accessible to its clients and partners, who are encouraged to provide creative feedback — ensuring a constant influx of fresh, new perspectives and ideas. Leveraging roundtable discussions, on-site client meetings, learning seminars and working groups, Spark Global Solutions is able to direct its research and development to address unmet needs in the legal industry.

In Spark Global Solution's agile work atmosphere, new concepts move quickly from idea to reality, keeping pace with the ever-changing legal industry. Spark Global Solution's uniquely flexible solutions adapt to multilayered business environments and integrate seamlessly into disparate technology frameworks.

Spark Global Solutions believes in fostering a sense of community within its offices' walls as well as in the greater global neighborhood. Spark Global Solutions provides its employees with the ability to leverage their skills in fresh ways, and to take ownership over their jobs, their communities, and their roles as responsible, innovative, creative individuals.

## 1.2. PROJECT OVERVIEW

**Content Management System- Extranet Portal** is a web-based application designed to automate the content management of associatedcontent.com website.

In this project contents from the website 'associatedcontent.com' are filtered and pushed into database. Contents based on the category search are inserted into database in a single click. If the articles are below the limit than the need, the articles also can be downloaded.

After the contents are inserted into database those contents are re-written to refresh it i.e., the contents are first translated into 'Russian' language using 'Microsoft Translator' and again it is translated into 'English' using 'Google Translator' and inserted into Database as rewritten content.

The needed re-written content's titles are checked for packing depending on the selected category according to user requirement. Also unpacking and blocking the package can be done.

This portal displays all the titles under the selected package with the word count of titles' corresponding content in order to know the size of the article.

The pack of contents can be downloaded in any desired formats such as '.pdf', '.txt', '.doc'. Also re-downloaded can be done any other chosen format.

The downloaded files can be sold and also customer information is stored. Sold articles are blocked in order to avoid duplication in sales.

This system enables to generate Jasper reports. The needed Jasper reports can be downloaded in PDF format or can be viewed using I-Report Viewer. It helps to view the category report and sales, articles package reports fully and as well as date wise.

## CHAPTER 2

### DEVELOPMENT ENVIRONMENT

#### 2.1. HARDWARE SPECIFICATION

##### Hardware Requirements

- AMD Turion(tm) X2 Dual-Core Mobile RM-70
- 2 GB Ram
- 250 GB HDD

#### 2.2. SOFTWARE SPECIFICATION

##### Software Requirements

- Apache-Tomcat-5.5.28 (Web Server)
- Eclipse Java EE IDE
- Jaspersoft I-Report 3.7.1

##### Database Requirements

- MySQL 5.0.89

##### Client Software Requirements

- Microsoft Internet Explorer
- Netscape Navigator
- Mozilla Firefox

##### Platform Requirements

- Windows 7
- J2EE (Struts Framework)
- Jsp (Server side scripting)
- Java Script (Client side scripting)

## 2.3. ABOUT THE SOFTWARE

### Eclipse IDE

**Eclipse** is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. It is written primarily in Java and can be used to develop applications in Java and, by means of the various plug-ins, in other languages as well, including **C, C++, COBOL, Python, Perl, PHP, and others**. The IDE is often called **Eclipse ADT** for Ada, **Eclipse CDT** for C, **Eclipse JDT** for Java and **Eclipse PDT** for PHP.

The initial codebase originated from VisualAge. In its default form it is meant for Java developers, consisting of the **Java Development Tools (JDT)**. Users can extend its capabilities by installing plug-ins written for the Eclipse software framework, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules

The stated goals of Eclipse are "to develop a robust, full-featured, commercial-quality industry platform for the development of highly integrated tools." To that end, the Eclipse Consortium has been focused on three major projects:

- **The Eclipse Project** is responsible for developing the Eclipse IDE workbench (the "platform" for hosting Eclipse tools), the Java Development Tools (JDT), and the Plug-In Development Environment (PDE) used to extend the platform.
- **The Eclipse Tools Project** is focused on creating best-of-breed tools for the Eclipse platform. Current subprojects include a Cobol IDE, a C/C++ IDE, and an EMF modeling tool.
- **The Eclipse Technology Project** focuses on technology research, incubation, and education using the Eclipse platform.

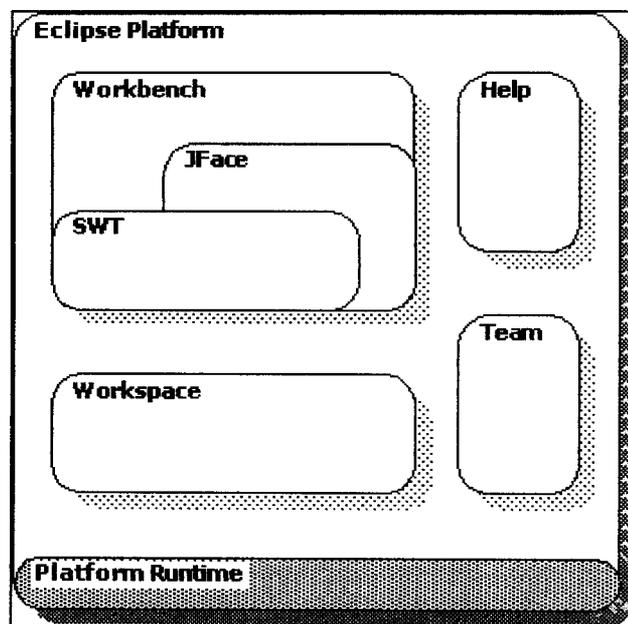
The Eclipse platform, when combined with the JDT, offers many of the features you'd expect from a commercial-quality IDE: a syntax-highlighting editor, incremental

code compilation, a thread-aware source-level debugger, a class navigator, a file/project manager, and interfaces to standard source control systems, such as CVS and ClearCase.

Eclipse also includes a number of unique features such as code refactoring, automatic code updates/installs (via the Update Manager), a task list, support for unit testing with JUnit, and integration with the Jakarta Ant build tool.

### **Eclipse platform overview**

The Eclipse platform itself is structured as subsystems which are implemented in one or more plug-ins. The subsystems are built on top of a small runtime engine. The figure below depicts a simplified view.



## Architecture

Eclipse employs plug-ins in order to provide all of its functionality on top of (and including) the runtime system, in contrast to some other applications where functionality is typically hard coded. The runtime system of Eclipse is based on Equinox, an OSGi standard compliant implementation.

This plug-in mechanism is a lightweight software componentry framework. In addition to allowing Eclipse to be extended using other programming languages such as C and Python, the plug-in framework allows Eclipse to work with typesetting languages like LaTeX, networking applications such as telnet, and database management systems. The plug-in architecture supports writing any desired extension to the environment, such as for configuration management. Java and CVS support is provided in the Eclipse SDK, with Subversion support provided by third-party plug-ins.

With the exception of a small run-time kernel, everything in Eclipse is a plug-in. This means that every plug-in developed integrates with Eclipse in exactly the same way as other plug-ins; in this respect, all features are created equal. Eclipse provides plug-ins for a wide variety of features, some of which are through third parties using both free and commercial models. Examples of plug-ins include a UML plug-in for Sequence and other UML diagrams, a plug-in for Database explorer, and many others.

The Eclipse SDK includes the Eclipse Java Development Tools (JDT), offering an IDE with a built-in incremental Java compiler and a full model of the Java source files. This allows for advanced refactoring techniques and code analysis. The IDE also makes use of a workspace, in this case a set of metadata over a flat file space allowing external file modifications as long as the corresponding workspace "resource" is refreshed afterwards. The Visual Editor project allows interfaces to be created interactively, thus allowing Eclipse to be used as an RAD tool.

Eclipse's widgets are implemented by a widget toolkit for Java called SWT, unlike most Java applications, which use the Java standard Abstract Window Toolkit

(AWT) or Swing. Eclipse's user interface also uses an intermediate GUI layer called JFace, which simplifies the construction of applications based on SWT.

### **About Eclipse Foundation**

Eclipse is an open source community, whose projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle. The Eclipse Foundation is a not-for-profit, member supported corporation that hosts the Eclipse projects and helps cultivate both an open source community and an ecosystem of complementary products and services.

The Eclipse Project was originally created by IBM in November 2001 and supported by a consortium of software vendors. The Eclipse Foundation was created in January 2004 as an independent not-for-profit corporation to act as the steward of the Eclipse community. The independent not-for-profit corporation was created to allow a vendor neutral and open, transparent community to be established around Eclipse. Today, the Eclipse community consists of individuals and organizations from a cross section of the software industry.

### **Introduction: Struts Framework**

Apache Struts is an open-source web application framework for developing Java EE web applications. It uses and extends the Java Servlet API to encourage developers to adopt a model-view-controller (MVC) architecture. It was originally created by Craig McClanahan and donated to the Apache Foundation in May, 2000. Formerly located under the Apache Jakarta Project and known as Jakarta Struts, it became a top level Apache project in 2005.

Although Struts is a well-documented, mature, and popular framework for building front ends to Java applications, it is facing new challenges from newer "light weight" MVC frameworks such as Spring MVC, Stripes, Wicket, and Tapestry. The new XForms standards and frameworks may also be another option to building complex web Form validations with Struts in the future.

The WebWork framework spun off from Apache Struts several years ago, aiming to offer enhancements and refinements while retaining the same general architecture of the original Struts framework. However, it was announced in December 2005 that Struts would re-merge with WebWork. WebWork 2.2 has been adopted as Apache Struts2, which reached its first full release in February 2007.

In 2004 Sun launched an addition to the Java platform, called JavaServer Faces (JSF). Aside from the original Struts framework, the Apache project also offers a JSF-based framework called Shale.

### **Advantages of Struts Framework**

The Struts Framework is a standard for developing **well-architected** Web applications. It has the following features:

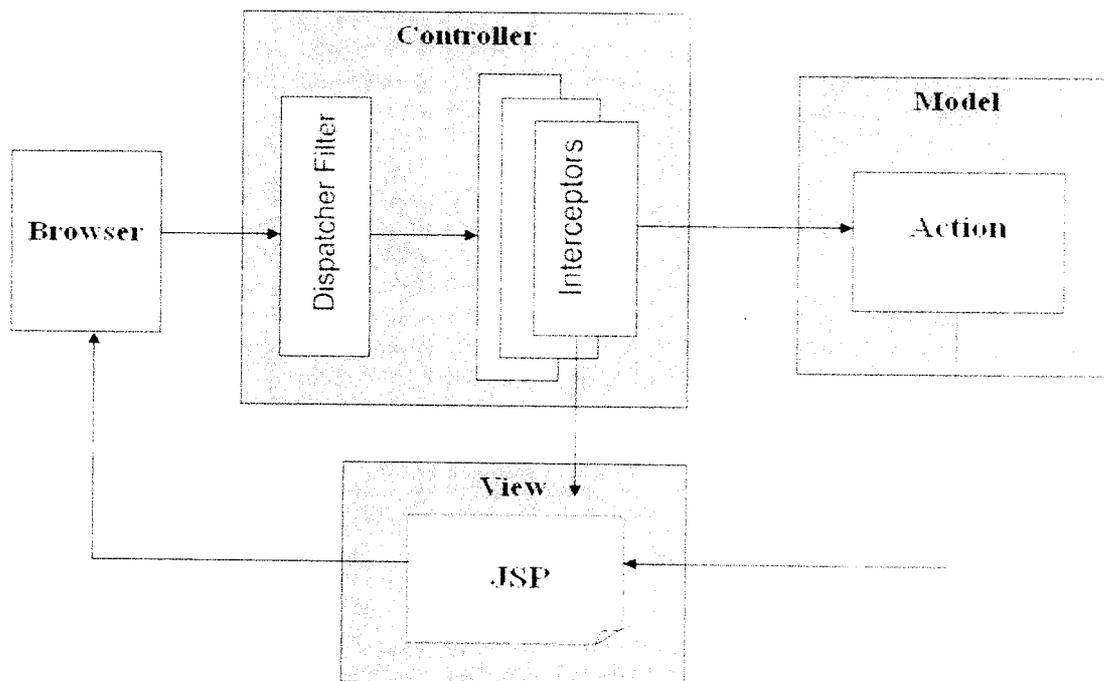
- Open source
- Based on the **Model-View-Controller (MVC)** architecture
- Implements the JSP Model 2 Architecture
- Stores application routing information and request mapping in a single core file, `struts-config.xml`

The Struts Framework, itself, only fills in the View and Controller layers. The Model layer is left to the developer.

## Architecture (MVC) –Struts

MVC design paradigm, distinctly separating as three levels:

- **Model** : application state
- **View** : presentation of data (JSP, HTML)
- **Controller** : routing of the application flow



### Model Components

Model components provide a "model" of the business logic or data behind a Struts program. For example, in a Struts application that manages customer information, it may be appropriate to have a "Customer" Model component that provides program access to information about customers.

It's very common for Model components to provide interfaces to databases or back-end systems. For example, if a Struts application needs to access employee information that is kept in an enterprise HR information system, it might be appropriate

to design an "Employee" Model component that acts as an interface between the Struts application and the HR information system.

Model components are generally standard Java classes. There is no specifically required format for a Model component, so it may be possible to reuse Java code written for other projects.

### **View Components**

View components are those pieces of an application that present information to users and accept input. In Struts applications, these correspond to Web pages.

View components are used to display the information provided by Model components. For example, the "Customer" Model component discussed above would need a View component to display its information. Usually, there will one or more View components for each Web page in a Struts application.

View components are generally built using JavaServer Page (JSP) files. Struts provides a large number of "JSP Custom Tags" (sometimes referred to as Struts Tags) which extend the normal capabilities of JSP and simplify the development of View components.

### **Controller Components**

Controller components coordinate activities in the application. This may mean taking data from the user and updating a database through a Model component, or it may mean detecting an error condition with a back-end system and directing the user through special error processing.

Controller components accept data from the users, decide which Model components need to be updated, and then decide which View component needs to be called to display the results.

One of the major contributions of Controller components is that they allow the developer to remove much of the error handling logic from the JSP pages in their application. (After all, if errors in processing occur, the Controller component forwards to an error-processing View component, not the primary results View component.) This can

P-3 267



significantly simplify the logic in the pages and make them easier to develop and maintain.

Controller components in Struts are Java classes and must be built using specific rules. They are usually referred to as "Action classes."

## **Back End**

MySQL 5.0.89

## **MySQL Features**

MySQL supports a set of features that result in the following benefits

### **Scalability**

The same database engine can be used across platforms ranging from laptop computers running Microsoft Windows® 95/98 to large, multiprocessor servers running Microsoft Windows NT®, Enterprise Edition.

### **Data warehousing**

MySQL includes tools for extracting and analyzing summary data for online analytical processing (OLAP). MySQL also includes tools for visually designing databases and analyzing data using English-based questions.

### **System integration with other server software**

MySQL integrates with e-mail, the Internet, and Windows.

## **Databases**

MySQL is a relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL is owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Sun Microsystems, a subsidiary of Oracle Corporation.

MySQL can be built and installed manually from source code, but this can be tedious so it is more commonly installed from a binary package unless special customizations are required. On most Linux distributions the package management system can download and install MySQL with minimal effort, though further configuration is often required to adjust security and optimization settings.

Though MySQL began as a low-end alternative to more powerful proprietary databases, it has gradually evolved to support higher-scale needs as well.

It is still most commonly used in small to medium scale single-server deployments, either as a component in a LAMP based web application or as a standalone database server. Much of MySQL's appeal originates in its relative simplicity and ease of use, which is enabled by an ecosystem of open source tools such as phpMyAdmin.

In the medium range, MySQL can be scaled by deploying it on more powerful hardware, such as a multi-processor server with gigabytes of memory.

There are however limits to how far performance can scale on a single server, so on larger scales, and multi-server MySQL deployments are required to provide improved performance and reliability. A typical high-end configuration can include a powerful master database which handles data write operations and is replicated to multiple slaves that handle all read operations. The master server synchronizes continually with its slaves so in the event of failure a slave can be promoted to become the new master, minimizing downtime. Further improvements in performance can be achieved by caching the results from database queries in memory using memcached, or breaking down a database into smaller chunks called shards which can be spread across a number of distributed server clusters.

## CHAPTER 3

### SYSTEM STUDY AND ANALYSIS

#### 3.1. EXISTING SYSTEM

Before our proposed system, everything was done manually. We used to collect articles from various searches from website. They searched for a content by open each and every URL, and copy the content and paste it in any editor and that content has to be changed as corresponding format such as .pdf, .txt, .doc.

##### 3.1.1 Drawbacks of the Existing System

- The time taking for the manual process is much more.
- High man power should be involved.
- Conversion software will take some cost.
- Difficult to maintain duplication in sales.
- Not possible for packing group of articles in order to sale packages of articles.
- Maintaining category of articles is very tedious in manual task.
- Also inventory is very tedious in manual.

#### 3.2. PROPOSED SYSTEM

This associated content-portal is a user friendly, GUI based client/server application that will automate the activity of extracting articles contents from a search URL. This application fetches up to a limit of articles from a website based on searched categories. The application completely automating the different activities of the various processes, such as rewriting article contents in order to refresh the content, for good delivery of article sale and packing, unpacking of group of articles and download in various formats such as .pdf, .txt, .doc.

The system will help in reducing the time and effort in fetching large number of articles quickly, and keep track of sale order, inwards, outward and date wise status of article packs and reports generated. It provides an error free generation of all the reports.

### **3.2.1. Advantages of Proposed System**

- Automate the activity of fetching articles contents from associated content website without opening all those article's URLs.
- By giving a search URL, up to some limits of articles can be fetch from a particular category. It consumes time much better.
- No need to buy any translating software to refresh the articles content and it is very difficult to rewrite all the content by open source translators. So inbuilt translators are used here in our system.
- Packing and unpacking of articles has been made easyly. By checking the articles titles it is packed together and also it can be unpacked.
- Download is a special and there is no need of any conversion software, this portal itself downloads the articles in pack wise in various formats.
- Re download option also avail in our application. In case of selling already sold packs it has been done.
- Reports also can be got in various formats according to need. It also taken care of sales that sales report can be got in date wise.

## CHAPTER 4

### SYSTEM DESIGN

#### 4.1. DIAGRAMS

↷ Use-case Diagram

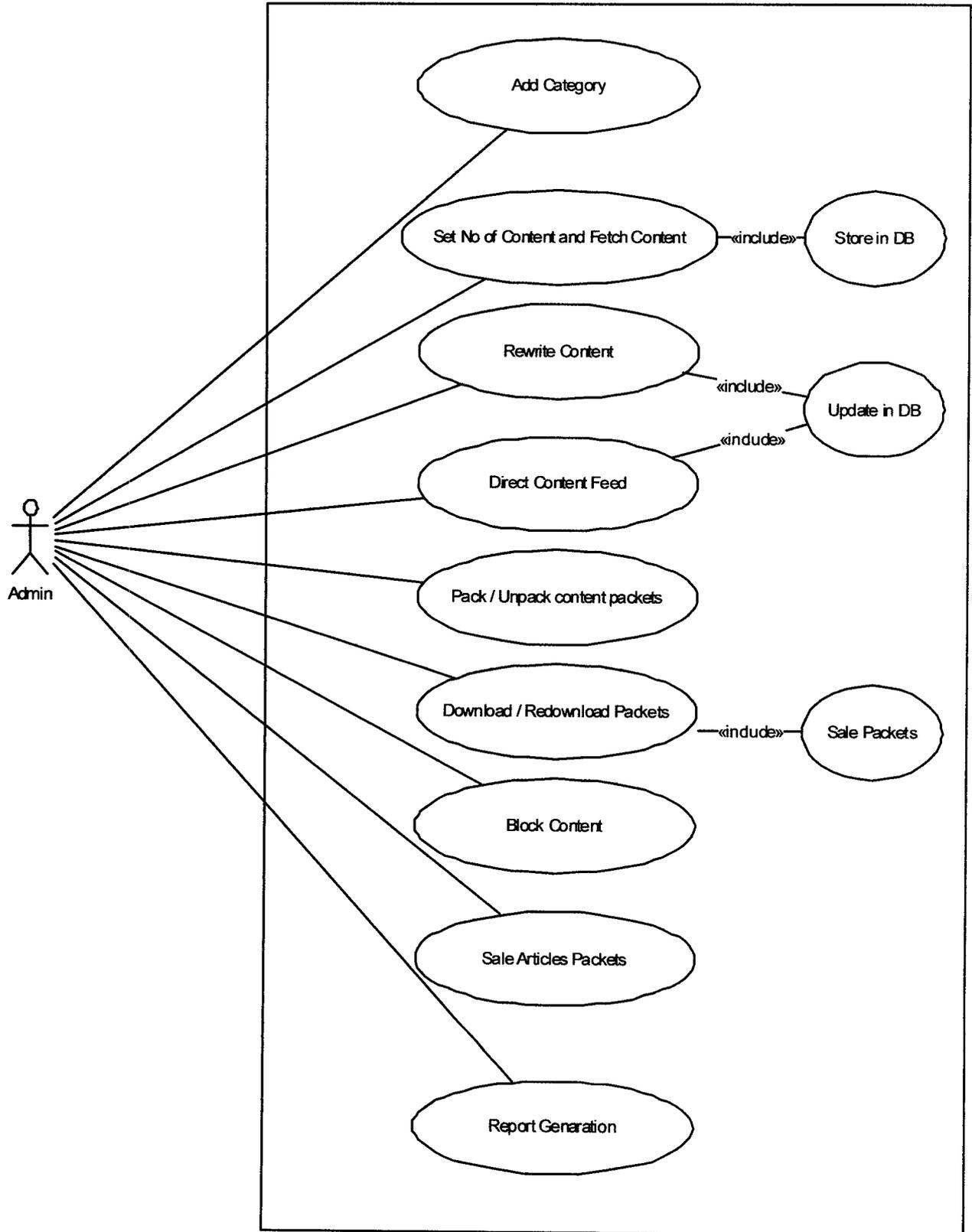
↷ Dataflow Diagram

- Level – 0
- Level – 1
- Level - 2

##### 4.1.1. USE-CASE Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

### Use-case for CMS-Extranet Portal



### 4.1.2. Data Flow Diagram

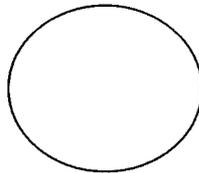
The data flow diagram is one of the most improvement tools used by the system analyst DeMacro (1978) Nad Gane Sarson (1979) popularized the use if the data flow diagrams as modeling tools through their structured system analysis methodologies.

A data flow diagram should be the first tool used by system analyst to model system components. These components are the system processes; the data used by this processes and external entities that interact with the system and the information flows in the system.

There are four kinds of system components

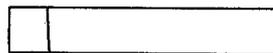
#### Process

Process show what system does. Each process has one or more data inputs and produces one or more data outputs, Circles in a data flow diagram represent process. Each process has unique name and number. This name and number appear inside the circle that represents the processes in a data flow diagram. This process is represented as circle



#### Data Stores

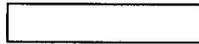
File or data store is depositary of data. They contain data that is retained in the system. Processes can enter the data into a data store or retrieve data from the data store. Each data store is represented by thin line in the data flow diagram and each data store has a unique name. The data store is represented in form of a line



## External Entities

External entities are outside the system but they either supply input data into the system or use the system output. They are entities which the designer has no control. Square or rectangle may represent external entities that supply data into a system or sometimes called sources. External entities that use the system data are sometimes called sinks.

External entity is represented as rectangle



## Data Flows

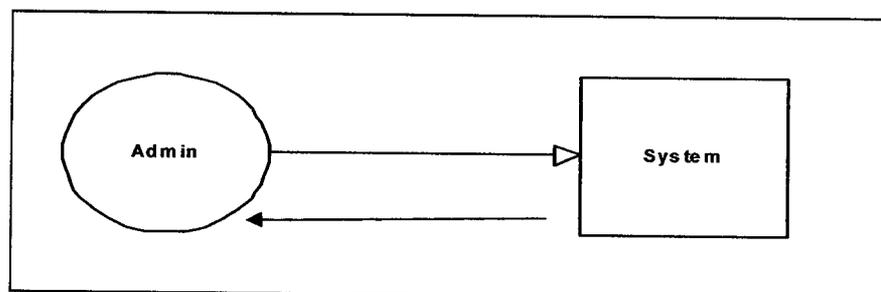
Dataflow model the passage of data in the system and are represented lines joining system components. An arrow indicates the direction of the flow and the line is labeled by the name of the data flow.



Flow of data in the system can take place.

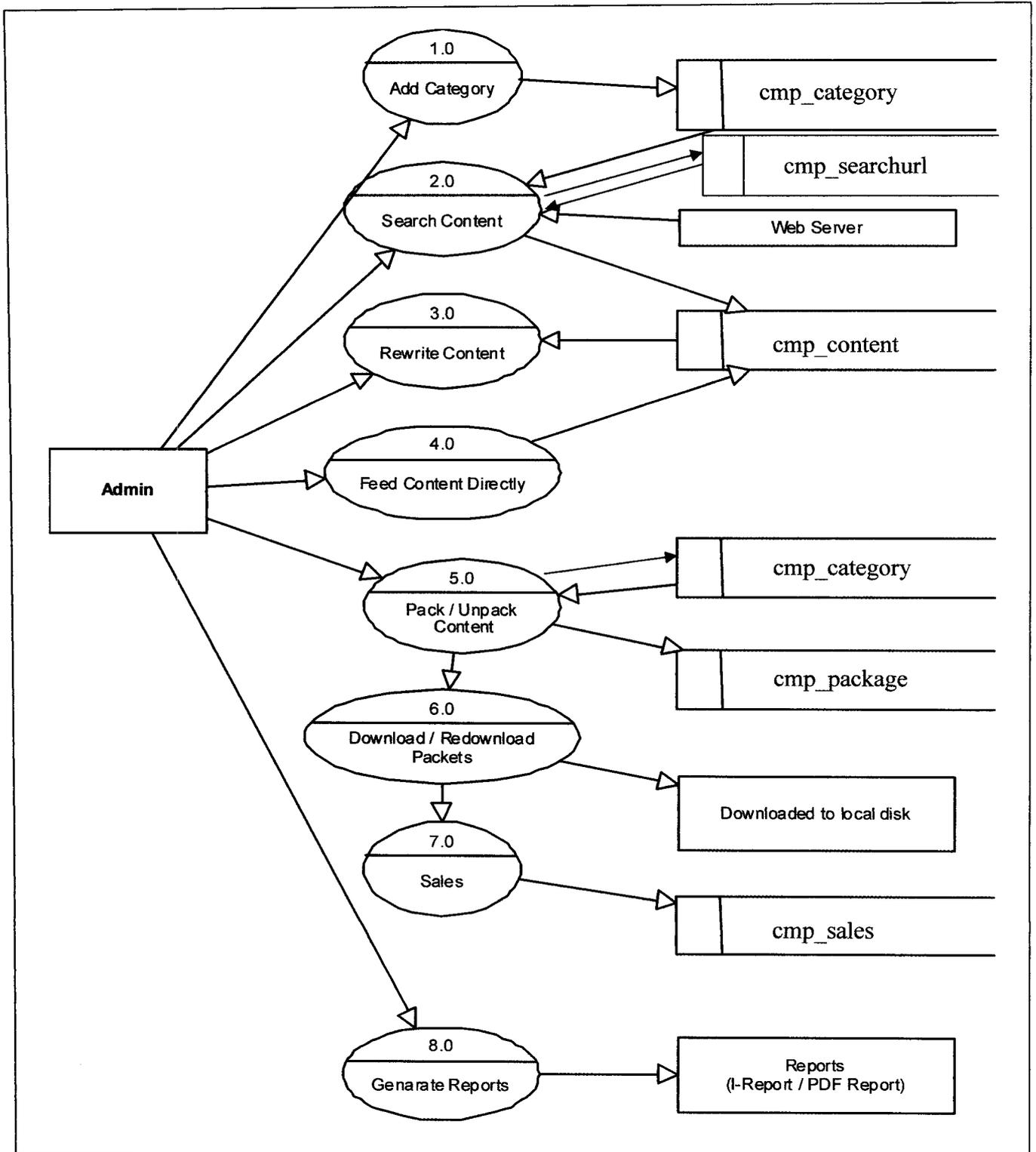
- Between two processes
- From a data store to process
- From a process to a data store
- From a source to process
- From a process to link

## Level – 0



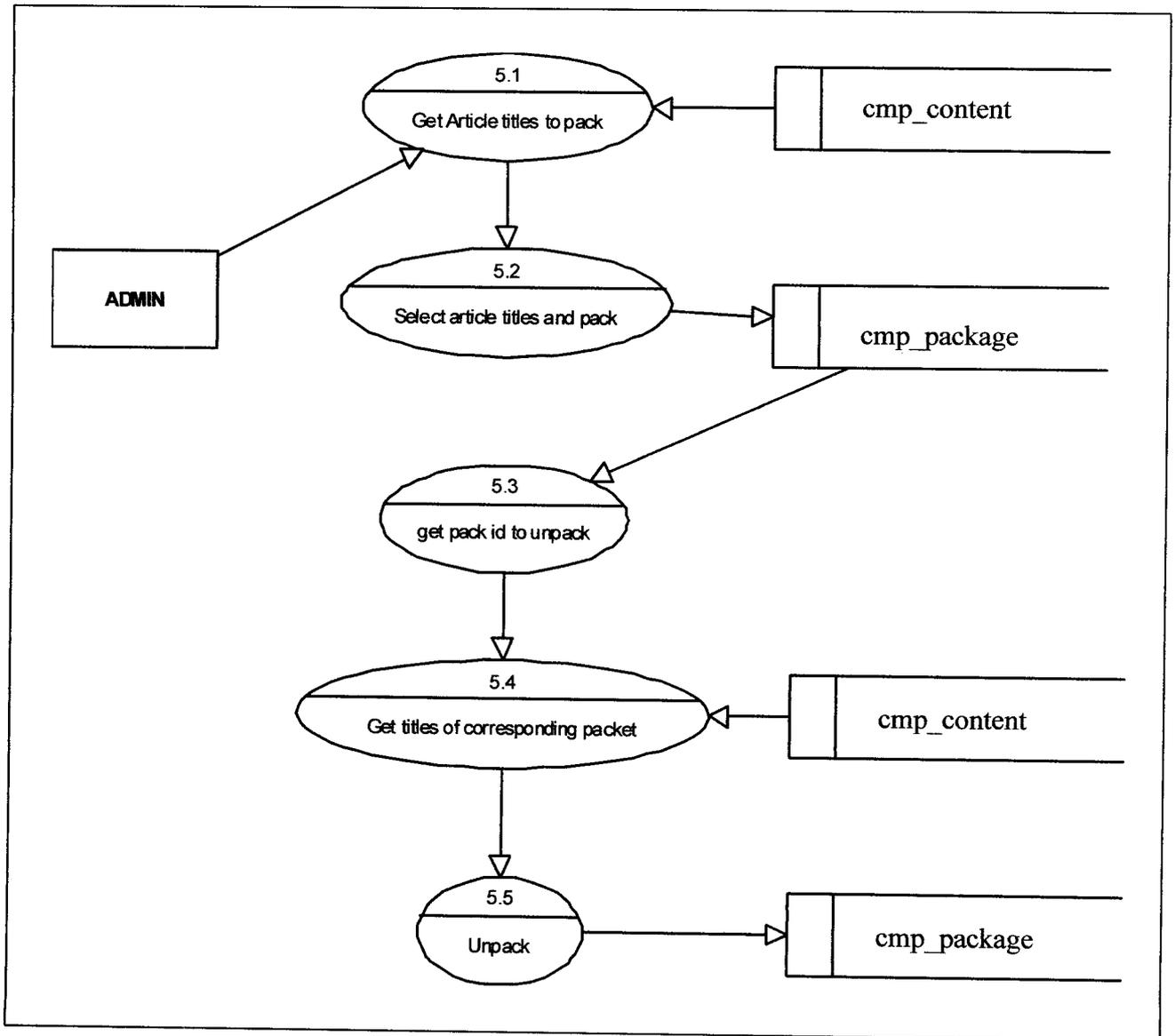
Level - 1

Dataflow Diagram for CMS - Extranet Portal



## Level – 2

## Dataflow Diagram for Pack / Unpack Activity



## 4.2. ELEMENTS OF DESIGN

System Design is the most creative and challenging phase in the development of a software system. The first step is to determine what input data is needed for the system and then to design a database that will meet the requirements of the proposed system. The next step is to determine what outputs are needed from the system and the format of the output to be produced.

During the design of the proposed system some areas where attention is required are:

- ⌘ How are the inputs required and the outputs produced?
- ⌘ How should the data be organized?
- ⌘ What will be the processes involved in the system?
- ⌘ How should the screen look?

The steps carried out in the design phase are as follows:

- ✓ Input Design
- ✓ Output Design
- ✓ Database Design

### 4.2.1 INPUT DESIGN

Input design is a part of the system design and hence must be carefully designed which otherwise lead to serious errors in the later stages of development. Inaccurate input data is the most common cause of errors in data processing. The main objective of designing input focus on

- ↻ Controlling the amount of input required
- ↻ Avoiding delayed responses
- ↻ Keeping process simple
- ↻ Controlling and avoiding errors

## 4.2.2 OUTPUT DESIGN

Output generally refers to the results and information that are generated by the system. For many end-users, output is the main reason for developing the system and the basis on which they will evaluate the usefulness of the application. Most end-users will not actually operate the information system or enter data through workstations, but they will use the output from the system. When designing output, system analysis must accomplish the following.

- ☞ Determine what information to present
- ☞ Decide whether to display, print or speak the information and select the output medium
- ☞ Arrange the presentation of information in an acceptable format.
- ☞ Decide how to distribute the output to intended recipients.

The arrangement of information on a display or printed document is termed as layout. Accomplish the general activities listed above will require specific decisions, such as whether to use pre-printed forms when preparing reports and documents, how many lines to plan on a printed page or whether to use graphics and colour.

The output design is specified on layout performs, sheets that describe the location characteristics, and format of the column headings and pagination.

The output must be provided in a format easily understandable even by a novice user. After analyzing the operations of the system, output information required for each jobs are determined. In addition to this, these outputs may be in format suitable as inputs for subsequent processing.

### 4.2.3 DATABASE DESIGN

A database is a collection of inter-related data stored with minimum redundancy to serve many users quickly and efficiently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user. An elegantly designed database can play a strong foundation for the whole system.

The details about the relevant data for the system are first identified. According to their relationship, tables are designed through the following method.

- The data type for each data item in the table is decided.
- The tables are then normalized.

The tables are normalized so that they can provide better response time, have data integrity, avoid redundancy and be secure.

#### DATABASE STRUCTURE:

This system uses many numbers of tables to store the details of agents and clients. It also contains the table about transactions.

#### 4.2.3(a): Table Name: cmp\_searchurl

This table contains the lists of all the processed urls. Every time when input gets contents from web server it refers this table to avoid redundant urls.

Field Name	Data type	Constraint
id	Int(10)	Primary
cmp_url	Longtext	

**4.2.3(b): Table Name: cmp\_category**

This table contains the lists categories which are all categories undergone search in this portal.

<b>Field Name</b>	<b>Data type</b>	<b>Constraint</b>
id	Int(10)	Primary Key
category	Varchar(45)	

**4.2.3(c): Table Name: cmp\_content**

This table contains all the extracted and re-written contents from website. Also contains which package the articles belong to.

<b>Field Name</b>	<b>Data type</b>	<b>Constraint</b>
id	Int(10)	Primary Key
category_id	Int(10)	Foreign key references cmp_category(id) table
title	Longtext	
cmp_url	Longtext	
content	Longtext	
content_rew	Longtext	
created_date	Date	
edited_date	Date	
package_id	Int(10)	Foreign Key References cmp_package(id) table
sold_flag	Tinyint(1)	
rew_flag	Tinyint(1)	

**4.2.3(d): Table name: cmp\_package**

This table contains the details of articles packages and flag value for sold packets, downloaded packets and blocked packets.

Field Name	Data type	Constraint
id	Int(10)	Primary Key
date_packed	Date	
block_flag	Tinyint(1)	
sold_flag	Tinyint(1)	
download_flag	Tinyint(1)	

**4.2.3(e): Table name: cmp\_sales**

This table contains the sales details and customer information.

Field Name	Data type	Constraint
sales_id	Int(10)	Primary key
pack_id	Int(10)	Foreign Key References cmp_package(packid) table
sold_Date	Date	
customer_info	Varchar(100)	
price	Double	

#### 4.2.4. MODULAR DESIGN

**Modular design** — or "modularity in design" — is an approach that subdivides a system into smaller parts (modules) that can be independently created and then used in different systems to drive multiple functionalities. Besides reduction in cost (due to lesser customization, and less learning time), and flexibility in design, modularity offers other benefits such as augmentation (adding new solution by merely plugging in a new module), and exclusion.

This system is also modularized to reduce the complexity of the system. This contains various modules.

##### A) Category Modules:

- Category module contains the list of categories whose contents are to be extracted.
- Category names are identified only by its id. Category id will flow over the project.

##### B) Content Extraction Module:

- ☞ Data extraction module extracts articles' contents from a website named 'Associated Content' which contain lakhs of articles and it will list them depend on the search by user.
- ☞ In this module user want to give url resulted from search and number of contents needed as input.
- ☞ Each page of the website contains 30 articles. If the user request more than 30, then the requested number of contents will trim to 30.
- ☞ Already inserted articles will not get inserted into database.
- ☞ If all the contents of requested page are already processed the system will automatically display the last processed URL so that the user know what to give as next input.

**C) Re-write Module:**

- ☞ All the filtered content from website need to be refreshed to avoid duplication under search engines.
- ☞ The extracted contents are first translated from English to Russian using 'Microsoft Translator' and again used 'Google Translator' to convert from Russian to English.
- ☞ The re-written contents are again pushed into database.

**D) Direct Feed Module:**

- ☞ The user may wish to write his own article and sell it.
- ☞ The user wants to just give title, category and content of article and push it into database. Those contents also come under sales on request.

**E) Pack Modules:**

- ☞ All the re-written content's titles are displayed for packing according to requested category.
- ☞ The user may select the needed content's titles and pack it.
- ☞ There is also provision for unpacking.
- ☞ If the user thinks that there is some unwanted contents he may select them, pack as one and may block it. The blocked package will not come under download and sales.

**F) Download Modules:**

- ☞ The Contents of a package can be downloaded in formats like '.pdf', '.txt', '.doc'.
- ☞ The downloaded contents can re-download in other needed format.

**G) Sales Module:**

- ☞ All the downloaded and re-downloaded files can be sold according to the selected package.
- ☞ Sales module deals with storing of customer information, price, etc.,
- ☞ In sales page all the titles under selected package are displayed for conformation.
- ☞ Duplication of sales is avoided.

**H) Reports Module**

- ☞ Jasper reports are created by the system.
- ☞ Reports are either created as a PDF file or viewed using IReportViewer according to the user demand.
- ☞ The report file is generated with report name, date and time of report generation for uniqueness.
- ☞ Various reports such as
  - Category List
  - Articles Pack List
  - Date wise Sales, etc., are generated.

## CHAPTER 5

### SYSTEM IMPLEMENTATION AND TESTING

#### 5.1. IMPLEMENTATION

The system is implemented using Eclipse IDE, Tomcat Server and MYSQL. The business logic is implemented using Eclipse J2EE Galileo. For database communication, MYSQL is used along with PHP. MYSQL is the most popular Open Source SQL database management system. MYSQL has many inbuilt functions to carry out the operations with database. These functions are used as a part of PHP file. Using these functions the queries are executed and the operations on database are carried out.

The inputs are received from action script file of IDE. The operations are performed according to the command received from IDE. Those operations are performed and the results are returned to the IDE using XML files.

#### 5.2. SYSTEM VERIFICATION

**System Verification** is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. Verification is ensuring that the product has been built according to the requirements and design specifications- i.e., you built it right. Verification is the assurance that the products of a particular development phase are consistent with the requirements of that phase and preceding phase(s).

In this project, review of interim work steps is done to ensure they are acceptable. In data access, it verifies whether the right data is being accessed in terms of the right place and in the right way.

### 5.3. SYSTEM VALIDATION

**System Validation** is the process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. Validation checks that the product design satisfies or fits the intended usage (high-level checking) — i.e., you built the right product. This is done through dynamic testing and other forms of review. Validation ensures that the product actually meets the user's needs, and that the specifications were correct in the first place.

In this project validation is achieved using java script. Any wrong entry display error messages or warnings. All the fields are validated to make user to know about wrong data usage. Validation also determines if this project complies with the requirements and performs functions for which it is intended and meets the organization's goal and user needs.

### 5.4. TESTING

Testing is a critical element of software quality and assurance and represents the ultimate review of specification design and coding. It is a vital activity that has to be enforced in the development of any system. This could be done in parallel during all the phases of system development. The feedback received from these tests can be used for further enhancement of the system under consideration. The main type of test carried out in Personal Financial Advisor is Unit Testing and Integration Testing.

#### 5.4.1. Unit Testing

A series of stand-alone tests are conducted during Unit Testing. Each test examines an individual component that is new or has been modified. A unit test is also called a module test because it tests the individual units of code that comprise the application. Unit tests focus on functionality and reliability, and the entry and exit criteria can be the same for each module or specific to a particular module. Unit testing is done in a test environment prior to system integration. If a defect is discovered during a unit test, the severity of the defect will dictate whether or not it will be fixed before the module is approved.

In Content Management Portal each component i.e. each form is tested individually to verify that the detailed design for unit has been correctly implemented. Initially the flow of control and data through that page is checked. In a page, each control is further tested in unit testing. The process is done in all the forms of the system.

#### **5.4.2. Integration testing**

Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. A component, in this sense, refers to an integrated aggregate of more than one unit. In a realistic scenario, many units are combined into components, which are in turn aggregated into even larger parts of the program. The idea is to test combinations of pieces and eventually expand the process to test your modules with those of other groups. Eventually all the modules making up a process are tested together. Beyond that, if the program is composed of more than one process, they should be tested in pairs rather than all at once. Integration testing identifies problems that occur when units are combined.

## **CHAPTER 6**

### **CONCLUSION AND FUTURE ENHANCEMENT**

#### **6.1. CONCLUSION**

The project 'Content Management System- Extranet Portal' enables the user to extract some given number of articles in a single click. All the contents are refreshed, needed contents are packed into one, download the package contents and sell it.

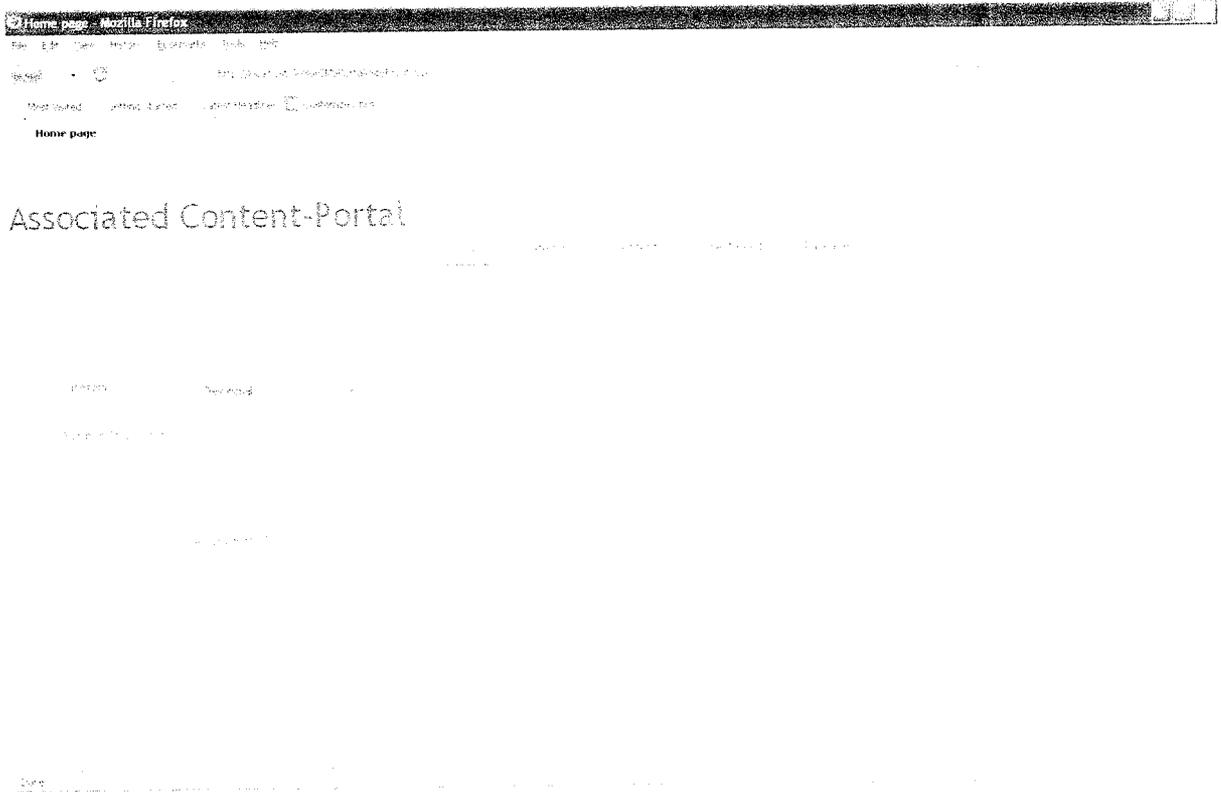
Thus this project saves the time and work of user from manually copying and using the content by opening all the urls from the websites.

## **6.2. FUTURE ENHANCEMENT**

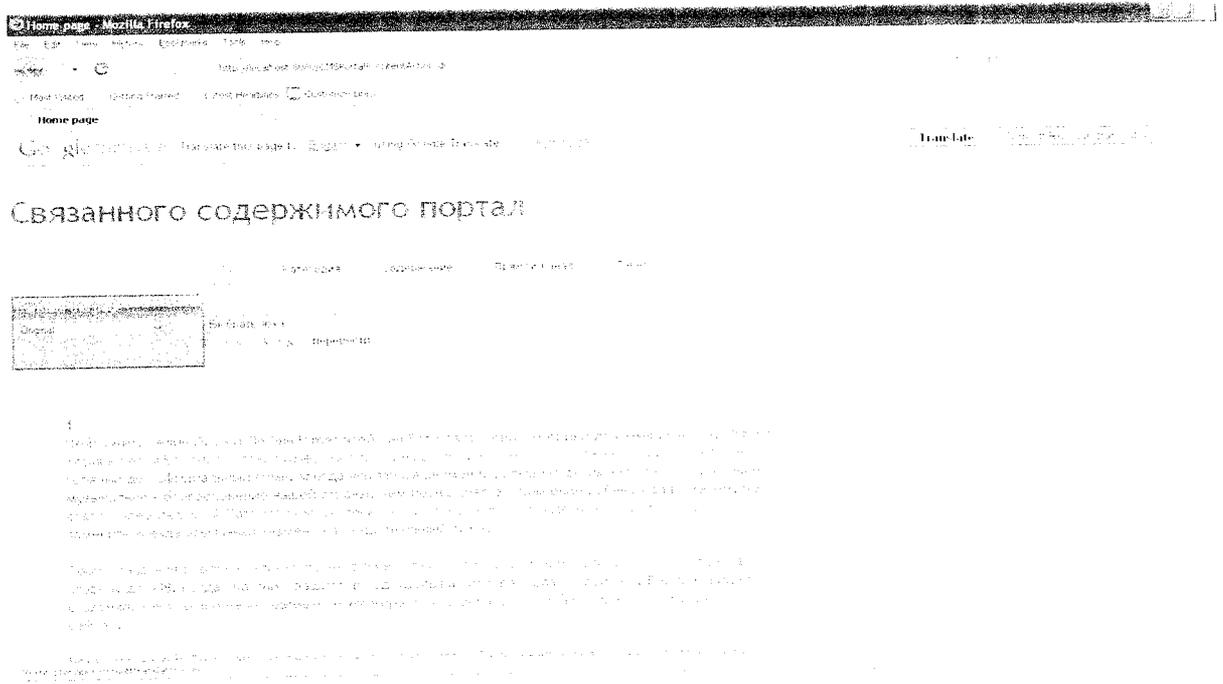
- ☞ The project 'Content Management System- Extranet Portal' enables the user to translate the content manually. In future these things are done automatically without user interaction.
- ☞ In this project already available translation software such as 'Microsoft Translator' and 'Google Translator'. In Future the contents are translated using own algorithm to optimize the contents more.
- ☞ User request is now receiving by separate mails, in future it is going to be implemented with mail service in order to do this service automatically once receives mail.

## APPENDIX-SCREEN SHOTS:

### Home:



### Translated to Russian Language:









Insert title here - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://localhost:8080/CMS/Portal/job/Package

Welcome to Facebook

Insert title here

## Associated Content-Portal

Category: Content

Select Package ID: [ ] Report: [ ] Refresh: [ ]

PackageID selected between: From Date: [ ] To Date: [ ]

PackageID: [ ] Downloaded into [ ]

Job: [ ]

With Content: [ ]

1. Downloaded into [ ]

PackageID	No of Articles	Category Name
PA011	4	Job
PA014	2	Job
PA015	2	Flowchart
PA017	1	Job
PA019	2	Job

## REFERENCES

1. [www.roseindia.com](http://www.roseindia.com)
2. [www. www.packtpub.com/ JasperReports/book](http://www.packtpub.com/JasperReports/book)
3. <http://struts.apache.org/>
4. <http://www.java-tips.org/java-se-tips/java.io/>
5. [http://www.java4less.com/print\\_java\\_e.html](http://www.java4less.com/print_java_e.html)
6. <http://dev.mysql.com/tech-resources/articles/>
7. <http://www.ibm.com/developerworks/library/j-struts/>