##### 

##### MULTI CONTENT MANAGEMENT SYSTEM

Session 2007-2011

#### DEVELOPED BY:

Syed Raza Ali BSIT-38

([hei@mastergraphiks.no](mailto:hei@mastergraphiks.no))

###### In the name of Almighty Allah,

###### The most Beneficent, the most Merciful.

###### And

###### In The Respect of Prophet Muhammad P.B.U.H, the Greatest Scholar and Human Reformer.

###### A project submitted to the Department

###### Of Computer Science & Information Technology,

###### University of Azad Jammu & Kashmir

###### In partial fulfillment of the requirements

###### For the award of the degree of

###### **Bachelor of Science in Information Technology**

PROJECT BRIEF

**PROJECT TITLE: Multi Content Management System**

**OBJECTIVE:** To provide better multi built-in sites solution for internet users.

**DEVELOPED BY: Syed Raza Ali (38)**

BS (Information Technology)

Session (2007-2011)

**TOOLS USED:** Developer Toolbox,

Photoshop, After Effect, Dreamweaver

**LANGUAGES USED:** PhP

MySql

JavaScript

HTML5

CSS3

**ABSTRACT**

Content Management System is basically an online software that provides the solution for internet users to manage their websites content directly from browser instead of manual editing the content therefor we have created web based application **Multi Content Management System** that not just provide Content management system solution, it also provide the readymade multi websites solution within one CMS. By using this **MULTI CMS** people can build their websites in less time and with no cost, as this CMS we have created is free of cost and anyone can download it and install into their server. This Content management system provides the 10+ built-in fully functional websites under the one CMS People can switch to any website is available under our CMS with one click.

# LIST OF ABBREVIATIONS

**ABBREVIATIONS DESCRIPTION**

**MULTI CMS** MULTI CONTENT MANAGEMENT SYSTEM

**CMS** CONTENT MANAGEMENT SYSTEM

**C&M** COMPONENTS & MODULES

**CUS-M** CUSTOMER MANAGEMENT

**USER-M** USERS MANAGEMENT

**CM** COMPONENTS MANAGEMENT

**MM** MODULE MANAGEMENT

**TM**  TEMPLATE MANAGEMENT

**M & C** MODULES & COMPONENTS

**PHP**  PERSONAL HOME PAGE

**MYSQL** MY STRUCTURE QUERY LANGUAGE

**XML**  EXTENSIBLE MARKUP LANGUAGE

**OE** OPERATING ENVIRONMENT

**DIC** DESIGN AND IMPLEMENTATION CONSTRAINTS

**UD** USER DOCUMENTATION

**AD** ASSUMPTIONS AND DEPENDENCIES

**HI** HARDWARE INTERFACES

**SI** SOFTWARE INTERFACES

**CI** COMMUNICATION INTERFACES

**EVO** EVOLUTIONARY DEVELOPMENT

CHAPTER 1

INTRODUCTION

# 1.1 INTRODUCTION

Now a day’s internet is a best place for doing business and promoting business worldwide, for that purpose people are taking more and more interest in this field. Everyone wants to create their own website to promote business or product around the world, but most of them cannot build websites due to lack of knowledge on how to create websites. They have to pay developers for creating websites. We are planning to create web based application **Multi Content Management System**. By using this **MULTI CMS** people will be able to create their websites without having the knowledge of programming or designing. They will not have to pay anything to developers and they will be able to launch their own website as they want using our advance user friendly **MULTI** **CMS**.

# 1.2 PROBLEM STATEMENT

The websites are now growing rapidly a lot of people are taking interest in creating own websites. But there are problems for new people who are new to web don’t know the programming. There are few online Content management systems that help people to use less knowledge in creating websites but still are difficult to use and enhanced. There must be a system that provides help in creating website quickly with different topics flavor.

# 1.3 EXISTING SYSTEMS

There are many online systems providing solutions like our proposed system. Some of the Popular Online CMS are:

## **1.3.1 JOOMLA**

 Joomla [1] is an Online Free Content Management System use to build different type of websites. It offers different extensions, components, modules to use in website according to need.

### **JOOMLA OFFERS**

1. Free Premade Components / Modules / Extensions
2. Free Templates

### **ADVANTAGES**

* Less Programming Knowledge
* Free Templates, Components, Modules, Extensions for Use

### **DIS-ADVANTAGES**

* Difficult to use
* No Built-in sites
* Required Special Knowledge to use the system
* To build website using premade components, modules, extensions we need to learn a lot about it.

1.3.2 WORDPRESS

C:\Users\Ali\Desktop\wordpress-logo.png Wordpress [2] is a popular online content management system, mostly used to create online blogs. But it also used to create personal websites. Wordpress have free plugins to use in there website.

### **WORDPRESS O0FFER**

1. Free Plugins to use in website
2. Free Themes

### **ADVANTAGES**

* Less Programming Knowledge
* Easy to install plugins

### **DIS-ADVANTAGES**

* Difficult to use
* No Built-in Sites during download
* To build personal website, required special skills.
* Installation of plugin is easy; to use those plugins is very difficult.
* To build website for specific topic, we need to work a lot and to give more time to understand the system.

# 1.4 PROPOSED SYSTEM

Our proposed system will be a CMS like joomla, wordpress. But it will offer more great features that are not yet provided by those popular CMS. Our proposed system will be called Multi CMS because it will have multi CMS features as well Pre-Made websites feature in it.

**Why MULTI CMS?**

We have suggested the name of our CMS to MULTI CONTENT MANAGEMENT SYSTEM, because our system will offer Multi CMS for pre-made 10+ websites flavors under one location. Our system will include 10+ Premade CMS based websites that can be change at just one click.

**How it will work, we have described two examples below:**

1. EXAMPLE 1**:** Incase user is interested to create a data search engine website; user will install our Multi CMS system in his server by following the steps. Once system is successfully installed, user will be requested to select the topic of his interest, user will just select search engine from drop down list and will click on change button. CMS will be change according to selected topic, all components, modules, extensions required by search engine website will be auto installed. Complete front-end website will be change to a data search engine website, which will be easy to manage from secure BACKEND CMS.
2. EXAMPLE 2**:** To create video streaming website, user will just select video streaming from drop down list and click on change. Complete backend CMS and frontend website will be change to a video streaming website. All components, modules, extensions that are required to use in video streaming website will be auto installed. Complete front-end website will be change to a video streaming website, which will be then easy to manage from online backend administrator area.

Same like above 2 examples, there will be around 10+ websites with different topics in Multi CMS:

* Online Blog
* Ad Posting Website
* Video Streaming Website
* Market Place
* Image Gallery Website
* Affiliate Shopping Store
* And Many Others

Only user will have to select the topic from drop down list in which he/she is interested and will click on change only. Complete website CMS will be change according to this topic. Not just 10+ websites, incase if user want website that is not in the list, user will be able to request for this website to include in the list our development team will design this system free of cost and will request the user to update the list file, not only that, user will be able to create website of his interest his self without having the high level skills of CMS or programming knowledge.

## **1.4.1 MULTI CMS OFFER**

* Premade 10+ CMS based Websites
* User will be easily change website to any other provided website in list in just 1 click.
* This system will provide to build custom websites as well.
* It will have Premade Components, Modules, Extensions, and Plugins.
* This system will have Premade Themes
* It will have very easy to use backend Admin System.

# 1.5 SCOPE OF SYSTEM

This **CMS** will help users to build websites as they want without having the knowledge of programming or designing. This **CMS** will be useful in different fields like:

* Personal Portfolio/Resume website.
* Affiliate Shopping Store For Earning Commission On Products.
* Ad Posting System to post free ads.
* Tutorials Website for posting different tutorials.
* Data Search engine for helping search the documents and files from data record.

Our Multi CMS will include professional tutorials for user’s support on using our system.

# 1.6 FEATURES

This MULTI **CMS** will have different great features, few of them are:

1. This **Multi Content Management System** will be user friendly
2. People will build different type of website from available list through this CMS like data search engine website, affiliate shopping, personal portfolio, resume, online blog, ad posting, video streaming, image gallery or similar.. We will provide 10+ Premade Websites in this CMS, users will have option to create own style website from available resources in our Multi CMS.
3. This **Multi Content Management System** will have a user friendly administrator area where users will be able to

* Secure Administrator Area to manage all data
* Manage User accounts
* Manage Content
* Manage Graphical Elements
* Easily Change to any website
* Easily Integration of built-in components.
* Manage Website templates
* Manage Components and Modules

## **1.6.1 DEFAULT FEATURES** IN ADMINISTARTOR AREA

There will be default features in administrator area that will be preset and available in any selected CMS from MULTI CMS section.

* Users Management (**USER-M**): **USER-M** will be a section in administrator area that will allow Administrator of **MULTI CMS** to manage the users of website, there data, information; limits, etc. (depend on what website is created).
* Components Management (**CM**): **CM** will be a section in administrator area, where user will easily add and remove important components according to website requirement.
* Module Management (**MM**): **MM** will be an administrator section that will allow **MULTI CMS** user to enable and disable all available modules according to website requirement. Those modules will perform most of action in front-end. Like category module showing, advertising area module within component or a website, and same like that others.
* Template Management (**TM**): **TM** will be a section available in administrator area that will be used to change the look and feel of the website and will use to change the widgets in website.

## 1.6.2 USER CLASSES

* System Administrator**:** Is generally the owner that takes care of maintenance for the **MULTI CMS**. The administrator will be in charge of assigning privileges of accounts, changing of data, maintain the website look and feel, adding and removing components and modules, etc. Suggested more than one individual can have administrator privilege to ensure advisability.
* Customer**:** A customer is an individual wishing to join the **MULTI CMS** based website. The customer that will join the website by creating free account in front-end (depend on if customer system is active in backend), then join the website to access the special features that will be available to members of website. Customer will use contact feature, order feature, record check feature, membership feature and etc.
* Components & Modules**:** These classes are available for backend user to enable and disable features in frontend. Adding and removing the components and modules, setting up C&M according to website requirement, adding limits in the **forms or parts** by editing modules or components in backend.

# 1.7 WEB APPLICATION ARCHITECTURE

## 1.7.1 INTRODUCTION

Web applications evolved from Web sites or Web systems. Documents are accessed and viewed with a piece of software called browser, a software application that runs on a client computer. With it, the user can request Web documents from other computers on the network and render the documents on the user’s display. To view a document, the user must start the browser and enter the name of the document and the name of the host computer on which it can be found. The browser sends a request for the document to the host computer. The request is handled by a software application called a Web Server; an application usually runs as a service. The Web server receives the request, locates the document on its final file system, and sends the documents back to the browser.

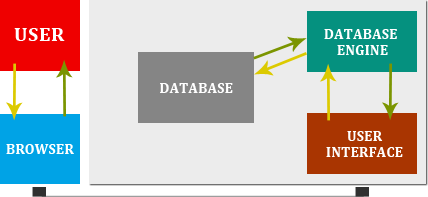


Figure 1. Website View Process

### WEB APPLICATIONS

Web applications use enabling technologies to make their content dynamic and to allow users of the system to affect business logic on the server. The distinction between Web sites and Web applications is subtle and relies on the ability of a user to affect the state of the business login on the server. If no business login exists on a server, the system certainly should not term as Web application. For those systems on which the Web server that allows business login to be affected via Web browsers, the system is considered a Web application.

### SESSION MANAGEMENT

One of the most common Challenges of Web applications is managing client state on the server. Due to connectionless nature of client and server communications, a server doesn’t have an easy way to keep track of the state of each client using the system. A session represents a single coherent use of the system. A session usually involves the use of many executable Web pages and lots of interaction with the business login on the application server.

### COOKIES

There are several ways to do session management, the most being use of cookies. Cookies, a feature of browsers, allow a web server to place a short string of characters-a cookie in the browser. Cookies are persistent and can even last beyond the lifetime of a browser’s execution.

## 1.7.2 PRESENTATION TIER

Connecting to tier means to accept and deliver the data and services from one tier to the other tier. The problem to look for the link with the presentation tier is the type recognition problem. The middle ware is written in PHP, which contain various data types as another programing and scripting languages, whereas the presentation tier is written in HTML that don’t possess any data type. Therefor an extra care to be made in the receiving and sending data to and from the Presentation Tier.

## 1.7.3 MIDDLE TIER

Middle tier consists of two parts

* Web Server
* Application Server

### WEB SERVER

A web server is a piece of software that manages the web pages and makes them available to client or we may say that a web server provides the functionality of a linking tier. It means that when the client requests for a static web page, it provides that page to the client. If the request needs some processing web server passes this request to application server and then receives the processed web pages from the application server and redirect this web page to the client. Browsers via a local network or over the internet. In the case of internet, the web server and the browser are usually on two different machines.

### APPLICATION SERVER

Application server receives the request from the server and parses this request for the processing. If the request demands some data, application server queries the database server for the particular data. Then the result is formatted in the form of an HTML web page and returns it to the web server.

## 1.7.4 DATABASE TIER:

All the data is cached on the server side into another database. The data is then transported to the backend (original) database. The system uses a message passing system. This system passes a message to the other database for any changes in any database. When the message is received by the destination database, the database is update instantly.

## 1.7.5 HOW ARE DYNAMIC WEB PAGES SERVED?

There are two ways of providing dynamic web page content

### DYNAMIC WEB PAGES FROM DATABASE:

The one way of dynamic web pages is to show them from the database using the specific coding methods. We specify the database quiry and then using php programming we get the specific record from database using POST / GET Method. Then we show the record from database to specific page. As soon the unique identify POST / GET Name change, the content of the page also change.

### DYNAMIC WEB PAGES FROM WITHIN PAGE:

Another way of making a static page to dynamic page is through defining the section with in one page and showing them by getting the Unique POST / GET Name.

# 1.8 PROCESS MODEL

We have used the Evolutionary Development Model **[7]** for our Project. By using Evolutionary Model we can update our designed software to new version with new features and supports of different latest browsers and systems. Below is some introduction about this Model

## 1.8.1 EVOLUTIONARY DEVELOPMENT MODEL

In the Evolutionary Model, development engineering effort is made first to establish correct, precise requirement definitions and system scope, as agreed by all the users across the organization. This is achieved through application of iterative processes to evolve a system most suited to the given circumstances.

Initiation

Deployment

Figure 1. 2 EVOLUTIONARY DEVELOPMENT MODEL

This model differs from the iterative enhancement model in the sense that this does not require a useable product at the end of each cycle. In evolutionary development, requirements are implemented by category rather than by priority.

### NEED OF AN EVOLUTIONARY DEVELOPMENT MODEL

The various reasons why there exists a need for an evolutionary model include:

* Business and product requirements often change as development proceeds.
* Tight market deadlines make completion of a comprehensive software product impossible but a limited version must be introduced to meet competitive and business pressures.
* A set of core product or system requirements is well understood, but the details of product or system extensions have yet to be defined.

USES OF EVOLUTIONARY DEVELOPMENT MODEL  
This model is useful for projects using new technology that is not well understood. This is also used for complex projects where all functionality must be delivered at one time, but the requirements are unstable or not well understood at the beginning.

CHAPTER 2

REQUIREMENT ANALYSIS & SPECIFICATION

# 2.1 SOFTWARE REQUIREMENT SPECIFICATION

## 2.1.1 OVERALL DESCRIPTION

### PRODUCT PERSPECTIVE

**MULTI CMS** is a new system designed for users new to the online business. The **CMS** will be a quick and easy means to setup and operate an online Website.

### PRODUCT FEATURES

Customer Management (**CUS-M**): **CUS-M** is a section in administrator area allow user of CMS to manage the customers of website, there data, information, orders, etc. (depend on what topic is selected).

Components Management (**CM**): **CM** is a section in administrator area, where user can easily add and remove important components according to website requirement.

Module Management (**MM**): **MM** is an administrator section allow **CMS** user to enable and disable all available modules according to website requirement. Those modules will perform most of action in front-end. Like category module showing, advertising area module within component or a website, and same like that others.

Template Management (**TM**): **TM** is a section available in administrator area used to change the look and feel of the website front-end.

### USER CLASSES

1. **SYSTEM ADMINISTRATOR:** Is generally the owner that takes care of maintenance for the **MULTI CMS**. The administrator will be in charge of assigning privileges of accounts, changing of data, maintain the website look and feel, adding and removing components and modules, etc. Suggested more than one individual can have administrator privilege to ensure advisability.
2. **CUSTOMER:** A customer is an individual wishing to join the **MULTI CMS** based website. The customer can join the website by creating free account in front-end (depend on if customer system is active in backend), then join the website to access the special features that are available to members of website. Customer can use contact feature, order feature, record check feature, membership feature and etc.
3. **COMPONENTS & MODULES:** These classes are available for backend user to enable and disable features in frontend. Adding and removing the components and modules, setting up components and modules according to website requirement, adding limits in the **forms or parts** by editing modules or components in backend.

### OPERATING ENVIRONMENT

**OE-1:** **MULTI CMS** shall operate with the following internet browsers: Microsoft IE7+, Firefox 3+, Google Chrome, and Netscape 5+.

**OE-2:** **MULTI CMS** shall operate on an Intel based system provides linux based hosting and Apache Web Server. Although maintenance documentation will be supplied and the operating system will be tested, the developers of this **MULTI CMS** are not responsible for the functionality of the operating system.

**OE-3:** The system shall use MYSQL based database to store Information’s, settings, changes.

**OE-4:** The system will be developing using the PHP Version 4.0, and will require a PHP supporting Hosting with minimum requirement of PHP 4.0.

### DESIGN AND IMPLEMENTATION CONSTRAINTS

**DIC-1:** Must use a MYSQL based database. MYSQL standard is the most widely used database format. Restricting to MYSQL allows easy of use and compatibility for **MULTI CMS** based Website.

**DIC-2:** Compatibility is only tested and verified for Microsoft Internet Explorer 7 and 8, Netscape 5+, Google Chrome, Firefox 3+. Other versions may not be 100% compatible.

### USER DOCUMENTATION

**UD-1:** Shall install online help for users via the web interface.

**UD-2:** Shall deliver Operations and Maintenance manual, Users Guide book, and Installation Instructions for the Administrator.

### ASSUMPTIONS AND DEPENDENCIES

**AD-1:** Assume hosting providers database can be lose a connection due to server poor quality.

**AD-2:** Assume the files hosted in the ftp of hosting account and data stored in database is loosed due to system damage.

**AD-3:** Assume the backup is not created and further data is losing due to some critical effects.

## 2.1.2 SPECIFIC REQUIREMENTS

### CONTENT ADDING, REMOVING, EDIT

1. **DESCRIPTION AND PRIORITY**

User will able to add, remove or change the content of website according to requirement. User can add categories, sections and articles on each category according to requirement. This is high priority system feature. Backend security will assure that the article will not be changeable except the administrator.

1. **STIMULUS / RESPONSE SEQUENCES**

* Server send request for changes
* System perform action in database
* Changes according in the system
* Web Browser will show changes result

1. **FUNTIONAL REQUIREMENTS**

User will be able to add and remove content into website from the backend area. User can add restrictions to the each content either a user’s of website can copy it, can change it in frontend or not.

### TEMPLATE SETTING

1. **DESCRIPTION AND PRIORITY**

User of **MULTI CMS** will be able to change the look and feel of website by using built-in templates. User can edit the template style as well to change the template setting as well. This is high priority system require to show a website graphical interface in professional style.

1. **STIMULUS / RESPONSE SEQUENCES**

* Web browser will perform action of changes
* System will check and set the required template
* Server will perform action of changes
* Template will replace with current theme

1. **FUNTIONAL REQUIREMENTS**

User can change the template according to site requirement, he can easily add, remove the specific parts of templates as he needs. He can change setting of template as well.

### MODULES & COMPONENTS (M & C)

1. **DESCRIPTION AND PRIORITY**

Modules and components are important part of **MULTI CMS**, through them user can add new system to website according to requirement; can remove parts and sections of website and so one. Priority of this system is also high. With the use of modules and components website can be make as more professional with more features.

1. **STIMULUS / RESPONSE SEQUENCES**

* User perform action in web browser of adding and remove M & C
* System will check the changes required by user
* Server will perform the action of changes in the backend
* The Web Browser will show the changes in frontend

1. **FUNTIONAL REQUIREMENTS**

Modules & Components can be enabling and disable according to needs, through it the user can make more functions in the website and can add more features in the website he is going to develop using **MULTI CMS**.

### CUSTOMER ACCOUNTS

1. **DESCRIPTION AND PRIORITY**

User in the backend can manage the customer accounts, can set limits to each user, can edit information of customers, can remove and block customers, can track and complete customer orders etc (depend on which topic is used). This is normal priority function of website.

1. **STIMULUS / RESPONSE SEQUENCES**

* Web browser perform the action
* Server will check the changes
* System will make changes
* Browser will show the changes made by user in backend.

1. **FUNTIONAL REQUIREMENTS**

User can completely manage the customer information, data, and orders in the backend area according to what component is used for system.

### CHANGES CONFIRMATION

1. **DESCRIPTION AND PRIORITY**

Any changes made by the user in backend will required a confirmation; through it changes on data unexpected will be come to 0%. This is low priority function.

1. **STIMULUS / RESPONSE SEQUENCES**

Web browser initiate request of change

* Confirmation appear of server will appear
* Changes will be made according to yes no method
* Web browser will return result according to it

1. **FUNTIONAL REQUIREMENTS**

This function will be helpful in expecting click made on any result. So when any change is made in backend the system will ask at least one time for confirm the change so user will be aware of what change is made.

### INTERFACE

1. **DESCRIPTION AND PRIORITY**

The interface will be presented to the customer in a web browser. The interface must remain consistent among various web browsers and be intuitive to the customer. This is a medium priority system feature.

1. **STIMULUS / RESPONSE SEQUENCES**

Users of website will able to see the website designed using **MULTI CMS** in web browser, where they will see different components and modules set by admin and will be able to see the contents and different parts of website, contact form, help and etc section that will be designed and set by the user of **MULTI CMS**.

### WEB FORM

1. **DESCRIPTION AND PRIORITY**

Forms can be set by admin in backend area, that could be registration form for membership, login form, or contact us form. This is normal priority function and is useful in membership based website mostly.

It comes into 3 parts

1. Registration form: here users provide their personal information, age, gender, address, username, password for membership required by various components of **MULTI CMS**.
2. Login Form: This form provide solution the users to get access to their account using username and password and then get access to secure areas that are for members only, place orders etc.
3. Contact Form: This form provides solution to user of website to communicate with the owner of website or teams.
4. **STIMULUS / RESPONSE SEQUENCES**

* Web browser initiate request in form
* Api transfer the request to server
* Server call the database for getting results
* Results are fetch or submit to database
* Web browser show the result of action at end

1. **FUNTIONAL REQUIREMENTS**

Forms help users to access member area, place orders, make contact with owner, searching the data and etc.

# 2.2 EXTERNAL INTERFACE REQUIREMENTS

## 2.2.1 USER INTERFACES



Figure 2. LOGIN BOX

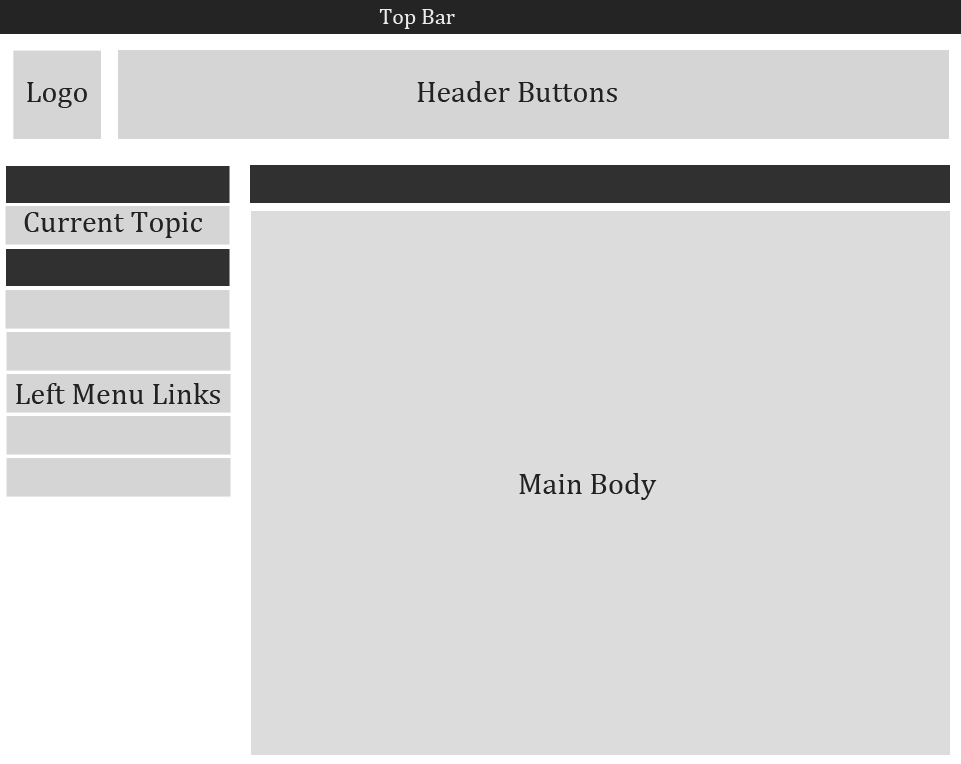


Figure 2. ADMIN DASHBOARD

Above screenshot is for Administrator Area. It contain 6 Different Locations that includes

* Top Bar
* Logo
* Header Buttons
* Current Topic
* Left Menu Links
* Main Body

That are the requirement sketches and can be change any in actual design.



Figure 2. FRONTEND

Above screenshot is an sketch of frontend view of 10+ websites. All 10+ sites have same style sketch with different options and content. The sketch showing above contain following things

* Top Bar
* Logo
* Main Menu
* Left Categories
* Left Widgets
* Main Body
* Right Categories
* Right Widgets

Above frontend design is for 1 theme only, our system support different themes and frontend view can be change according to themes designed for our Multi CMS>

## 2.2.2 HARDWARE INTERFACES

**HI-1:** Operating System with faster computer machine.  
**HI-2:** Linux Server contains php and mysql.

## 2.2.3 SOFTWARE INTERFACES

**SI-1:** **MULTI CMS** Browser Interface

**SI-1.1:** The MYSQL database of **MULTI CMS** will communicate with Administrator through a programmatic interface.

**SI-1.2:** Through programmatic interface, **MULTI CMS** will transfer the performed action detail into database and will make those changes in MYSQL Database.

**SI-1.3:** Plug-ins interface.

**SI-1.4:** Different Modules and Components will directly communicate with admin area.

## 2.2.4 COMMUNICATION INTERFACES

**CI-1:** The **MULTI CMS** system shall send data of customers registered at frontend and place order if the component is used (shopping cart) or related.

**CI-2:** The **MULTI CMS** system shall send an e-mail to System Administrator regarding any technical queries from customers or sales people.

# 2.3 OTHER NONFUNTIONAL REQUIREMENTS

## 2.3.1 PERFORMANCE REQUIREMENTS

* The system shall be able to handle unlimited users logged in concurrently at the same time. ( Depend on server )
* The system shall be able to add data to website in less than 2ms.
* The system shall be able to search for a specified data in less than 1 second.
* The system shall be able to email in less than 1 second.
* The system shall be able to validate data in less than 2 seconds.

## 2.3.2 SAFETY REQUIREMENTS

* The system will do periodic backups through a live internet connection.

## 2.3.3 SECURITY REQUIREMENTS

* The system shall validate data against fraud.
* The system shall encrypt all sensitive information via https.
* The system shall detect consecutive failed login attempts.
* The system shall be protected by open source firewall called Fire starter.

## 2.3.4 AVAILABILITY REQUIREMENTS

* The system shall have an availability of 99.99%.

## 2.3.5 EFFICIENCY REQUIREMENTS

* The system shall perform searches via algorithm.
* The system will employ on demand asynchronous loading for faster execution of pages.
* The system shall validate email address existence.

## 2.3.6 USABILITY REQUIREMENTS

* The system shall be easy to use
* The system shall be easy to learn
* The system shall utilize help bubbles to assist managers, customers, and administrators
* The system shall employ easy to locate buttons
* The system shall prompt customer with friend easy to read error messages.
* The system shall utilize consistent symbols and colors for clear notifications.

## 2.3.7 MAINTAINABILITY REQUIREMENTS

* The system shall utilize interchangeable plugins.
* The system shall be easily updatable for fixes and patches.
* The system shall create logs of all changes, updates, or fixes that are done to the site.
* The system shall be easy to upgrade.

## 2.3.8 TESTABILITY REQUIREMENTS

* The system should be able to run under debug mode.
* The system should be able to run test records.
* The system should be able to create test environment of **MULTI CMS**.

## 2.3.9 OTHER REQUIREMENTS

* The system shall adhere to the following hardware requirements:
* Linux Server with minimum 100 MB space.
* Apache web server
* Database: MySQL
* PHP 5.0

# 2.4 PROJECT PLAN

## 2.4.1 PLAN INTRODUCTION

This project is an online Content Management System used to build websites without the knowledge of programming and designing. This project will be prepare for company called Rayice,Co.

**HOW IT WILL BUILD?**

The project will build using server side script PHP stand for Personal Home Page and MySQL. Project will be run on Linux based server with apache support.

There are important things to be considered for Project to Develop and Complete.

### BUDGET

Total estimated budget for this project required is $309 / 26265 PKR

### DELIVERY TIME

The estimated time required to complete this project is around 3 Month

### STAFF AVAILABLE

Project will be assigned to Individual Person

## 2.4.2 PROJECT SCOPE

This **CMS** will help users to build websites as they want without having the knowledge of programming or designing. This **CMS** will be useful in different fields like:

* Affiliate Shop Business
* Market Place
* Personal Portfolio/Resume website.
* Image Gallery website.
* Video Streaming website.
* Ad posting website.
* Data Search Engine website.
* And many more websites.

Our Multi CMS will include professional tutorials for user’s support on using our system.

## 2.4.3 PROJECT ORGANIZATION

Alpha Team Members that involved in this project:-

Table 2. 1 Alpha Team Members

|  |  |
| --- | --- |
| ALPHA TEAM MEMBERS | ROLL |
| Syed Raza Ali | Programmer, Designer, Testing, Integration |

Syed Raza Ali: Role of syed raza ali in this project is as a manager of the project and the programmer, content writer, interface designer, resource locater, ITG.

Figure 2. 4 Role of Team

## 2.4.4 RISK ANALYSIS

There may be several risk types in CMS project.

### GENERIC RISKS

1. Project cost can be change which can cause the project to uncompleted.   
   **SOLUTION:** Project cost must include some extra charges to cover those issues.
2. Any team member if leave the project during the work can cause the failure of the project.  
   **SOLUTION:** there must be alternative members of same skills in organization that can replace if any person leaves the project because of any reason.

### PRODUCT SPECIFIED RISKS

Those risks can be special according to what system is designed. Our designed system is online Multi Content Management System so there can be several risks types according to system.

1. Admin Login may not works due to technical issue
2. Built-in Components and modules may not show in specific defined location
3. Cross-Browser support issue for HTML5/CSS3 versions.

And few others.

### KNOWN RISKS

Few known risks that will be considered during the development of CMS (Content management system)

1. Product should develop according to what size is mentioned on start.
2. Customer mind can be change during the development of project, so must be prepare for any changes that will come from the customer.
3. Technology can be change, so developing software must be support with new technology. Like window versions.
4. There must be alternative members of same skill that are involved in project.

## 2.4.5 HARDWARE AND SOFTWARE RESOURCE REQUIREMENT

### HARDWARE REQUIREMENT

This project is an online website called Multi Content Management System. This website will require a Hosting Server where the website will be host. So there are some of the hardware requirements included below:

1. Pentium 4 with minimum of 1 GB RAM and 40 GB Hard Disk Required.
2. Bandwidth Device with minimum of 1 GB Bandwidth Transfer.

### SOFTWARE REQUIREMENT

1. Linux Server with apache support
2. MySQL Server software
3. CPanel for managing Hosting
4. Domain name required for Server IP Mask

### COST REQUIRED

Table 2. Cost Required

|  |  |
| --- | --- |
| PRODUCT | COST |
| Pentium 4 – 1 GB RAM – 40 GB Hard Disk | $100 / 8500 PKR |
| Linux Server ( Apache Support ) | $30 / 2550 PKR For Month |
| Cpanel Software | $169 / 14365 PKR |
| Domain Name | $10 / 950 PKR For Year |
| TOTAL Supported HW/SW COST : | $309 / 26265 PKR |

## 2.4.6 WORK BREAKDOWN

The project will consist of different components; modules that will be develop, test and integrate for completion of the project.

Table 2. Work Breakdown

|  |  |  |
| --- | --- | --- |
| TYPE | LENGTH | MILESTONES |
| Estimated Components | 8+ | Current 90% Completion |
| Estimated Modules | 8+ | Current 90% Completion |
| Back-end Sections | 15+ | Current 90% Completion |
| Front-end Sections | 10+ | Current 90% Completion |

2.4.7 PROJECT SCHEDULE  
Below are the durations and dependencies for defining the PROJECT SCHEDULE to complete the project. That includes components, modules, backend sections, frontend sections and few others.

### COMPONENTS

Table 2. 4 Component Tasks

|  |  |  |
| --- | --- | --- |
| TASKS | DURATION / DAYS | DEPENDENCIES |
| Themes Management | 3 |  |
| Widgets | 2 |  |
| Ads Management | 3 | Widgets |
| Content Pages | 3 |  |
| External Links | 4 | Widgets |
| Categories | 3 | Widgets |
| News Portal | 8 |  |
| Contact System | 4 |  |

### MODULES

Table 2. 5 Module Tasks

|  |  |  |
| --- | --- | --- |
| TASKS | DURATION / DAYS | DEPENDENCIES |
| Top Menu | 1 |  |
| Left Menu | 1 |  |
| Right Menu | 1 |  |
| Footer Menu | 1 |  |
| Side Categories | 2 | Left Menu, Right Menu |
| Left Ads | 2 | Left Menu, Right Menu |
| Right Ads | 2 | Left Menu, Right Menu |

### BACKEND SECTIONS

Table 2. 6 Backend Section Tasks

|  |  |  |
| --- | --- | --- |
| TASKS | DURATION / DAYS | DEPENDENCIES |
| DashBoard | 2 |  |
| Multi CMS | 3 |  |
| Categories Section | 1 | Multi CMS |
| Topic Content Edit | 3 |  |
| Components Section | 8 |  |
| Modules Section | 5 | Components Section |
| Themes Section | 6 |  |
| Statistics Section | 5 | Multi CMS |
| Profile Section | 1 |  |
| Configuration Section | 2 |  |
| Media Section | 1 |  |
| Pages Section | 1 | Components Section |
| Contact Management Section | 1 | Components Section |
| Ads Management Section | 2 | Components Section |
| External Links Section | 1 | Components Section |
| Comments Section | 6 | Multi CMS |
| Widgets Section | 3 | Themes Section |
| Members Section | 9 |  |

### FRONTEND SECTIONS

Table 2. 7 Frontend Section Tasks

|  |  |  |
| --- | --- | --- |
| TASKS | DURATION / DAYS | DEPENDENCIES |
| Main Section | 3 |  |
| Content Section | 2 |  |
| Categories Section | 2 | Main Section |
| View Content Section | 2 |  |
| Login Section | 3 |  |
| Registeration Section | 3 | Main Section |
| Account Manage Section | 6 | Login Section |
| Content Manage Section | 5 | Account Manage Section |
| Widgets Section | 2 | Main Section |
| Advertisement Section | 1 | Main Section |
| Search Results Section | 3 | Main Section |

# 2.5 FEASIBILITY STUDY

In the feasibility study, we see that the project is feasible or not. After ensuring that this project is totally feasible to the organization, we accepted the project and start working on it. Feasibilities that were concerned are the following.

* Technical Feasibility.
* Economic Feasibility.
* Legal Feasibility.

## 2.5.1 TECHNICAL FEASIBILITY

System meets all the technical requirements i.e. run under specified hardware and software specification as described in previous section, look into section **2.4.5** for technical feasibility detail.

## 2.5.2 ECONOMICAL FEASIBILITY

The project is economically feasible to organization. Costs required to run this software has been discuss previously, please look into Table 2.2 for economical feasibility detail.

## 2.5.3 LEGAL FEASIBILITY

The tools that are used for the development of this project are totally legal, explanation about those tools well explained in section 5.1 of chapter 5.

# 2.6 CHANGE MANAGEMENT

**Q.** How does software team identify items that are to be changed?

**A.** Customer will submit the change request form with his project id if he is an existing customer.

**Q.** How does organization manage existing versions of a program in a manner that will enable change to be accommodated effectively?

**A.** Organization will pass the changes to project manager syed raza ali to focus on the changes and generate the report of current progress and new changes to development team.

**Q.** How to ensure that changes have been implemented properly?

**A.** Testing team comprising of syed raza ali will do an end-user testing to ensure that changes have been implemented properly.

## **2.6.1 CHANGE MANAGEMENT PROCESS**

CUSTOMER REQUESTED CHANGES THROUGH FORM

ACCEPT – AN ECO IS GENERATED

BASELINE ESTABLISHED

SENT TO PROJECT MANAGER

DENIDED

SYED RAZA ALI CHECK AND MADE CHANGES TO BASELINE AND FORWARD THE CHANGES TO DEVELOPMENT TEAM

TESTING

BASELINE

TESTING TEAM WILL CHECK ALL THE CHANGES AND WILL UPDATE THE CUSTOMER

CUSTOMER IS INFORMED  
ABOUT CHANGES

UPDATE PROJECT MANAGER

Figure 2. 5 Change Management Process

## **2.6.2 CHANGE MANAGEMENT PROCESS STEPS**

Customer requested the changes through Form

Changes send to the Change Control Authority ( Syed Raza Ali )

Current changes has been accepted and requested in queued for action

Engineering Change Order (ECO) is generated

Assigned changes to development team

Changes made

Baseline is established for testing

Testing team tested the changes and updated the project manager ( Syed Raza Ali ) about results.

Project Manager ( Syed Raza Ali ) Informed the customer about changes

Customer checked and reviewed the changes

Replied with status

## **2.6.3 CUSTOMER CHANGE REQUEST FORM**

Below is the form where customer send request to project manager about changes required.   
MAJOR FIELDS

* Project ID (provided when starting project with organization at first),
* Subject about changes,
* Current Status (current status of items),
* Change Description (changes required on current items),
* Additional Comments, If any other information want to send to Project Manager.

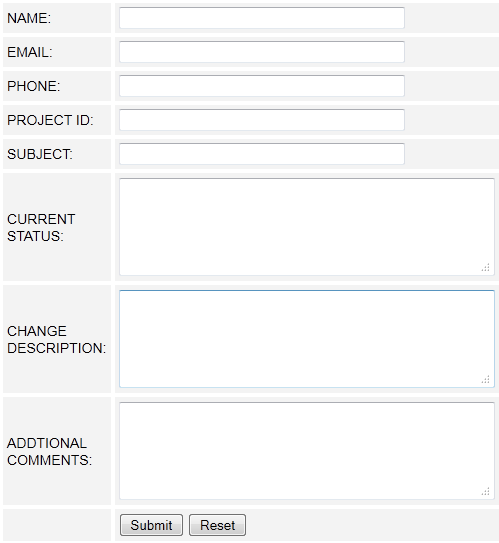


Figure 2. 6 Change Request Form

Can be access at http://www.rayice.com/changerequest.php

## **2.6.4 CHANGE MANAGEMENT TEAM**

Table 2. 8 Change Management Team

|  |  |
| --- | --- |
| ROLE | DESCRIPTION |
| PROJECT MANAGER | Syed Raza Ali is the in charge of the project. His goal will be to ensure that product will develop within time frame. So he will monitor the progress. |
| DEVELOPMENT TEAM | Syed Raza Ali. Following team will perform the development work including the changes comes from the customer |
| TESTING TEAM | Testing team will be involved Syed Raza Ali to perform testing on the changes to ensure all changes done perfectly. |

2.6.5 BASELINE

A) HARDWARE CHANGES  
No hardware Changes Comes from the customer side

### B) SOFTWARE CHANGES

Below are the software changes comes from customer, and need to be add into baseline in order to complete the project by using coming changes from customer.

Table 2. 9 Software Changes

|  |  |  |
| --- | --- | --- |
| NAME | CURRENT STATUS | CHANGE TO |
| Registration Area | Provide Registration System | It does not have username check feature, need to include |
| Support System | Provides phone feature, email feature | It does not have submit form, need to include |
| Header Menu | Good looking header menu with buttons | It does not have a buttons |
| Categories | Categories are showing in front-end | No sorting as ABC, need to include |

## 2.6.6 TIME

Table 2. Time

|  |  |
| --- | --- |
| TASK | DURATION |
| Statistics Section | **5 Day** |
| Configuration Section | **2 Day** |
| Additional Theme | **2 Day** |
| Categories Sorting | **1 Day** |
| DAYS: | **10 Days** |

CHAPTER 3

APPLICATION / LITERATURE REVIEW

# 3.1 ARTICLES

## 3.1.1 CMS AND A SINGLE WEB DESIGNER [3]

Content management is the next step in separating structure from design. What began with Cascading Style Sheets, is exploding with the CM environment, where billions were spent last year and more billions are expected to be spent in the years ahead.  
CM Systems come in many shapes: They can be huge or small, simple or very complex. They range from the very expensive. But they are all based on the same idea: CM allows designers to focus on design by building templates. Subject experts build content in a separate environment. The server takes the content, inserts it into the correct template and sends it all, neatly wrapped up, to end users.

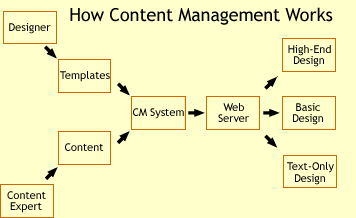


Figure 3. HOW CMS WORKS

But that’s just the technology side of CM systems. CM’s other aspect is the way it addresses your workflow. Sure, it’s great to separate the design from the content, but CM wants to streamline how your designs get approved and onto the server.

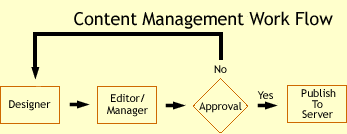


Figure 3. CMS WORK FLOW

Create a design in whatever tool/environment you’re comfortable with. Once it’s tested and ready to go, you pass it to your manager or editor or boss or whoever okays your design. If it’s approved, it’s sent on to the server. If not, you get notes and it is sent back to you, all within the CM environment: no email, no voice mail, no printouts of your design with red ink and yellow sticky notes all over it. The same process happens on the content side. The end result is that even though it’s easier for content and design to publish, there are still strict controls as to what makes it to the live server.

As a designer, you will need to find your place within the new CM world.

## Onward Specialization

With CM, designers get to specialize a bit. How many web designers moonlight in their organization as tech support, teachers, and troubleshooters for other people’s web pages? Why do so many do this? Because content experts add to the website via a WYSIWYG program they are not familiar with. So designers, the experts in such matters, become internal support for this bit of workflow jury—rigging. That’s money down the drain for an organization. Designers should design: it’s what we’re good at.

With CM systems, the content experts add content in a more simplified manner. Not through complex programs like FrontPage or GoLive, but through a simplified interface without all the design controls they don’t need. (Often, CM interfaces are browser–based and have a learning curve closer to Notepad than Word.)

Content people can author a page, add metadata to the page for cataloging and edit existing pages.

When an end user requests a page, the server takes the content, adds it into your template, and serves it on the web. Most CM systems allow you to design different templates for different browsers and serve the appropriate template based on the HTTP user agent info. Some systems allow for advanced personalization or even multi-language versions.

But what does this mean for designers?

## Implications for Designers (the good news)

Clearly, CM helps keep you from doing a lot of the non–design tasks that have become part of our job descriptions. Designers can spend time building templates for all parts of the site.

Moreover, a template–based environment allows changes you make to templates to be approved and seen by end users in a matter of minutes, not days or weeks. Changes you made to a template go through the normal workflow and are published to the server. The next user who requests a page with that template gets the new version; it’s a lot like changing a server–side include.

### CASCADING TECHNOLOGIES

Right now, most designers work with one hand tied behind their backs. Most enterprises have finally gotten it in their heads that using only the newest technologies like CSS2.They require designers to design a browser generation or two behind the state of the art. While that’s a good practice in some ways, it means that you are limited to using technology that may not have changed in years.

## Implications for Designers (the bad news)

Now that you can focus on design, a new issue appears: Will there be enough for a web designer to do?

A few years ago, the president of a mid–sized software company told me, “The purpose of any enterprise software is to make everything as efficient as possible, which means that some people will become unnecessary.” An organization using CM may decide that paying a full-time designer is more expensive than hiring an outside party to redesign templates every couple of months. So your survival may indeed be at stake.

## What to Do About the Bad News

So you may—surprise, surprise—have to do a little changing to survive. Web design specialists may consider taking on more special projects or print work. Scripters might consider a move towards web development.

Perhaps the best chance a generalist may have is to become a consultant/designer for the organization, or to move up to the management level, where a generalist who knows the workings of all the web–side technology will be very valuable.

## Chicken Little Was Wrong A Lot

None of what we’ve discussed takes into account what CM does best: namely, CM grafts the best of your design skills to the ceaseless production needs of content providers—and at a much lower cost than old–school hand production methods.

Given its ability to build efficient workflows and get the most out of designers and content experts alike, CM will likely be a favorite for managers and executives looking to increase their web–based productivity for some time to come.

# 3.2 REVIEWS

## 3.2.1 WORDPRESS VS JOOMLA VS DRUPAL

### WORDPRESS SYSTEM

WordPress [4] is a free, open source content management system intended for blogs. It is one of the most popular content management systems for blogs, and it can be adapted to serve a variety of purposes, such as displaying static web pages. Wp is the reincarnation of an older blogging system. But Word Press unquestionably has the higher market share. Wp was first released in 2003. Since then, it has progressed to its latest version (3.1) and won a Packet Open Source CMS Award in 2007.

As a result of the landslide of users adopting the Wp platform, there are websites with reams of free templates and plug-ins produced by professional programmer/designers and amateurs alike.

A major problem with Wp is security. It's not necessarily because the current release hasn't filled in the security holes—it's because most Wp blogs are running on older versions. Some are even running on 1.x versions! As a content management system progresses, it becomes (theoretically) more secure. Wp suffers from a compatibility problem with its plug-ins and the custom-coding a lot of people do to make their blog work as a website. Because of that, it becomes impossible to do the automatic updates through the administration area, and it's a huge hassle to re-code an entire website every time a new release of the CMS comes out.

A important thing of wordpress is that this is designed for creating blogging websites, to convert it to other websites required a lot of custom programming and designing. So if you are planning to use it and then stick with blogging and news based website with this system, you will face less problems for using it.

Another problem is with wp that we can use different variety of plugins and widgets to set into website but to build an systematically site with wordpress is hard enough and required special skills to build the website of your own interest. Any new user of wordpress must have to create website from scratch and must have knowledge of creating pages, categories, setting plugins, modules and too much customization for creating a normal website.

### DRUPAL SYSTEM

Drupal [5] is an open-source, free content management system programmed in PHP. It is cross-platform, supports multiple databases, and has become very popular as a web development framework among professional programmers and designers. Drupal started as a message board, and started provided many supported plugins, modules and widgets for the system.

Drupal was designed to be a community-built program, so it is set up to allow people to develop modules and themes easily. Many programmers and web designers feel that “giving back to the community”, a common theme in open source discussion, is a critical component of keeping open source alive. Therefore, they release many of the plug-ins they develop in the course of their professional work or those they've developed for the pure enjoyment of it. The Drupal community is highly supportive, and some programmers have been known to whip up solutions to your problems if you ask them nicely enough (or pay them).

One of the biggest complaints about this content management system is its control panel. It's been described as clunky, cryptic, and an all-around pain in the rear end to use—especially for those just jumping in on it. The original developer of Drupal, Dries Buytaert, says that he won't release the official Version 7.0 until the vast majority of the user-interface problems have been resolved. As it stands, though, it's certainly usable. The current control panel has a side-bar list of links leading to different areas in the back-end while the primary page displays the information and options. However, there are modules on the Drupal website to create a graphic user interface with icons and the side-bar list, like Joomla's panel.

Another problem is that Drupal sacrifices backward-compatibility for developing new core features. New features are always good, but sometimes the plug-in contributors have to completely rewrite their code every time a new version comes out. Unless you're a programmer working on Drupal modules, it works out in the end since you get more features, and the programmers put up with redesigning the code.

One bigger problem about this system is a common issue that we seem like in all other CMS is to understand the system in detail before going to put any website online, required special skills to customize the themes and logos and graphical elements from the coding. Still need to create website from scratch using different modules and widgets they offer, also there is a limitation of creating different style websites as this system is designed for message board and is good for that purposes only.

### JOOMLA SYSTEM

Joomla! [6] Is a free, open source content management system which is a fork, or off-shoot, of another CMS called Mambo. It's cross-platform, written in PHP, and contains all of the features you would expect from a modern CMS including: RSS feeds, page caching, blogs, searching, and language localization. It was officially separated from Mambo in 2005 when some of the developers stated that Mambo went against traditional open source values, so Joomla! Is supposed to follow the “totally free, totally open” mentality. So far, it's doing well.

It has so many extensions to install and modules to set on themes; it is too complicated software and required high skills of using this system more over then Wordpress and Drupal. Its backend admin system is too tuff to understand.

One problem you might encounter with this CMS is that the plug-ins created for earlier versions are not necessarily compatible with later versions. Also, the content hierarchy structure can't be changed without a major overhaul, some of the most troublesome problems with Joomla! Are its security issues. They're not crippling issues, but they certainly need to be addressed.

# 3.3 RESEARCH PAPERS

## 3.3.1 MULTI CONTENT MANAGEMENT SYSTEM

#### Abstract

Content Management System **[14]** Provides better solution to users for building online websites with less coding and expand their business easily. This paper researched the best features and techniques for CMS to make them more “stable” and “user-friendly”. *There are different Content Management Systems*; each Content Management System is designed for specific thing and very hard in usability. To overcome the problem of CMS usability and its features, we propose a (**Multi-CMS**) scheme which can help users to select Multi Ready Made CMS solutions with in one CMS, and select the appropriate topic of interested to get readymade website. *This means Multi-CMS will give user’s more readymade websites flavor instead of creating custom websites using CMS*, even users will have choice to select the topic of their interest and get website up within a minute with fully required features depend on what “topic” is selected. We are planning to apply this scheme on the practical project, **which will sooner release and will provide better and easiness Multi-CMS solution to users on the web.**

## 3.3.2 FROM WATERFALL TO EVOLUTIONARY DEVELOPMENT

How we rapidly created faster, more user-friendly, and more productive software products for a competitive multi-national market.

### ABSTRACT

Evolutionary development (Evo) **[7]** focuses on early delivery of high value to stakeholders, and on obtaining and utilizing feedback from stakeholders. This paper describes from a project manager’s viewpoint, the positive experiences that one organization rapidly achieved on switching from using the Waterfall method to Evo. Major benefit came from paying greater attention to the quality requirements as opposed to the previous practice of concentrating solely on the required functionality

CHAPTER 4

SYSTEM DESIGN OVERVIEW

In This chapter we will discuss about the Structure of the system and main parts of the system. This system is designed and included with different great features that can help the users to build and enhance the better style websites with online editing solution. Below is a wireframe Interface on Administrator and Frontend using this sketch we prepared our system template

# 4.1 USER INTERFACE DESIGN

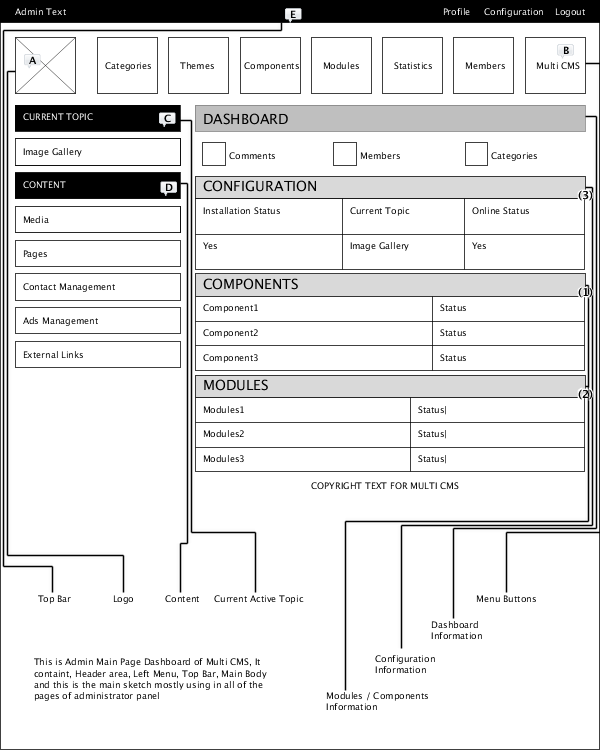


Figure 4. ADMINISTRATOR DASHBOARD SKETCH **[15]**

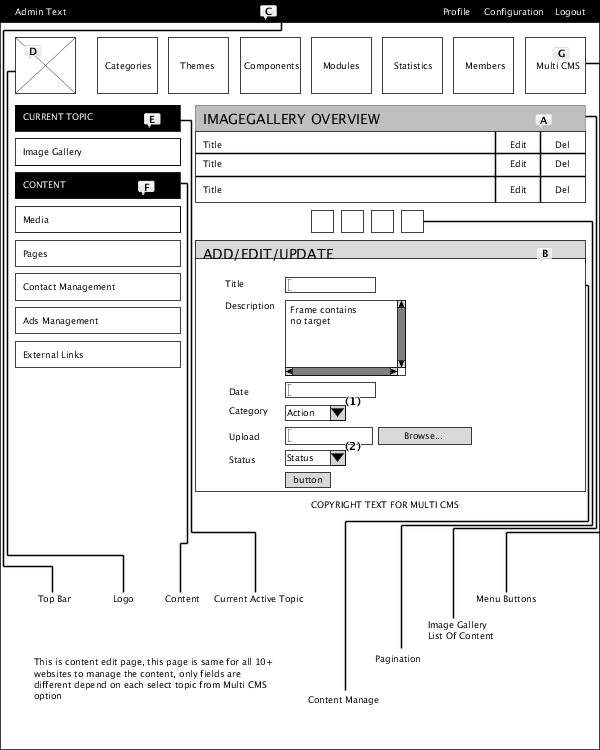


Figure 4. ADMINISTRATOR CONTENT EDITING SKETCH **[15]**

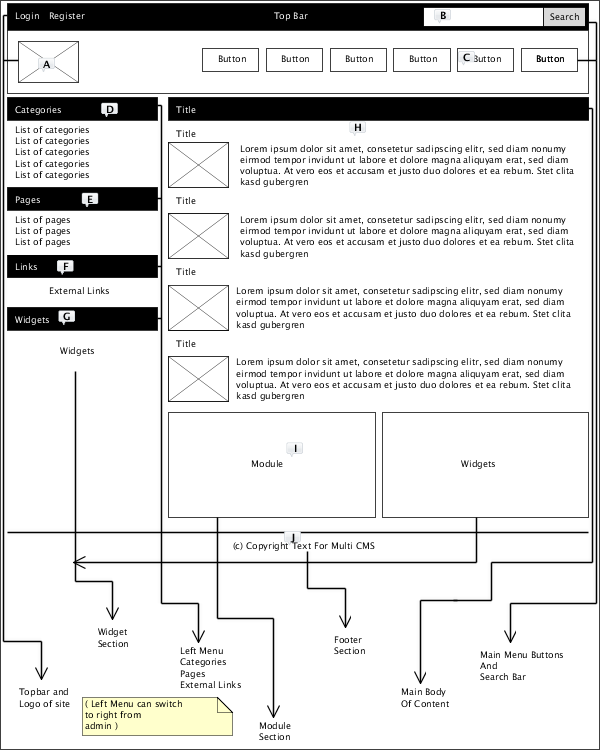


Figure 4. FRONTEND SKETCH **[15]**

# 4.2 SYSTEM PARTS

System parts are the most important parts in our system, system parts provide an enhancement of feature in software, they provide a more functionality on software, and they provide user friendly options in frontend. The main parts of system includes

* Components
* Modules
* Widgets

## 4.2.1 COMPONENTS

System component provide fully new functional parts of the system. These parts either include in backend of system or it is include in the frontend of the system. In Multi CMS we have built-in components in current version of the system. Those components included

* News
* Contact
* Ads Management
* Content Pages
* External Links
* Widgets

### NEWS

This component provides a fully functional news system that can be include in any Multi Topic Website. This news can be managed from the backend administrator area.

CONTACT  
this component provides a separate fully functional system that can be include in any Multi Topic Website. People can contact to website owner. All the messages can be viewable in administrator area under Contact Management.

### ADS MANAGEMENT

This component is an ad management component. That provides advertisement solution in website. Users of Multi CMS can use this component to add ads into website.

### CONTENT PAGES

This component provides a dynamic content solution to Multi Topics website. Users can add additional content pages to website using this component.

### EXTERNAL LINKS

This component provides additional system to website. It can be control from administrator area from where users can add external links that can be partner links or other website urls.

### WIDGETS

Widgets provide a way to embed the other sites parts or modules to website. Users can use other websites provided codes to show in Multi Topic websites using Widget Options.

## 4.2.2 MODULES

Modules are the parts of main website structures. Those modules mostly include in front-end area of the website. Current available modules of website are

* Top Menu
* Left Menu
* Right Menu
* Footer Menu
* Sidebar Categories
* Top Categories
* Left Ads
* Right Ads

## 4.2.3 WIDGETS

As discussed before widgets are other site parts or modules that can include into Multi CMS websites using Special Provide Codes. Those Widgets can be added from the backend administrator area into website using Embed codes. These widgets provide additional helpful stuff to website using external websites codes.

# 4.3 MULTI CMS OVERVIEW

Multi CMS is a special option in backend administrator area. Through this option website user can change the whole website into different type of website. User can select the from the available built-in topics in order to change the whole structure of the website according to selected topic. This is new function we have offer in our Content Management System and that is a main key for what we have decided to choose a name of our website from CMS to Multi CMS (Multi Content Management System). This is new feature and is our own research on Online Software, No one else online CMS Software is providing this feature in their website. At the moment we are offering the limit sites of popular interested topics that’s included in our Multi CMS. How this Multi CMS works we will explain to you through diagram for better understanding

### EXAMPLE OF CURRENT WEBSITE

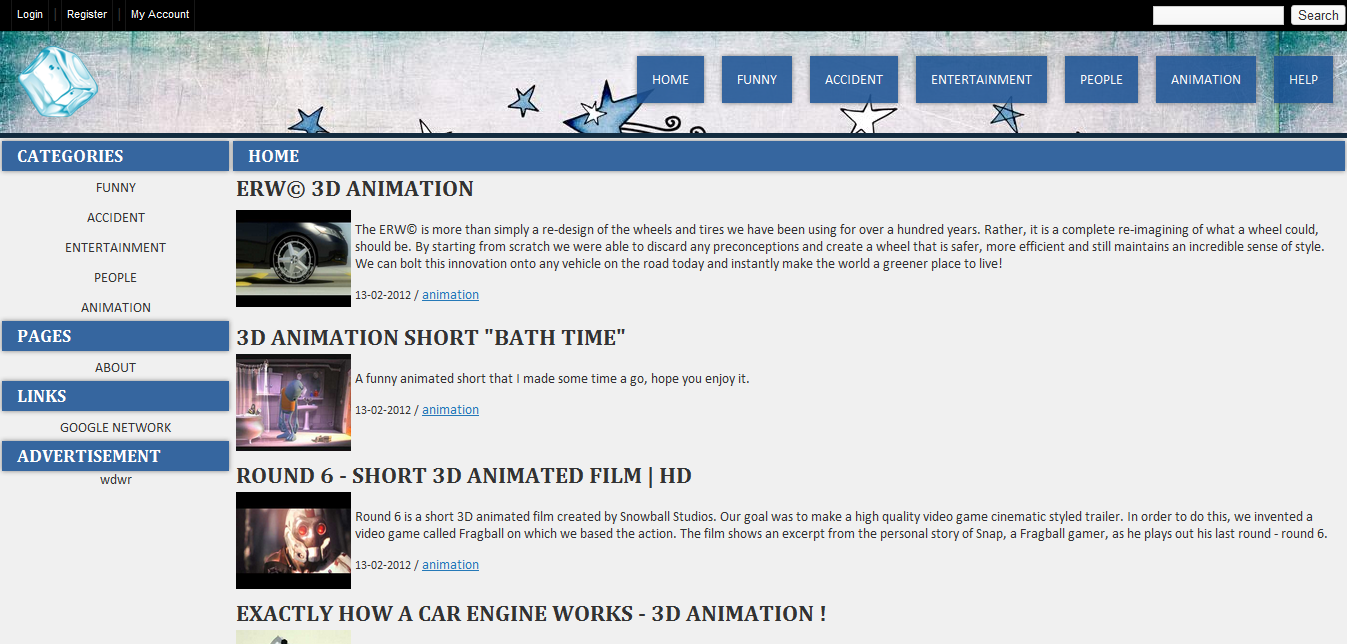


Figure 4. EXAMPLE PREVIEW OF WEBSITE

## 4.3.1 USING MULTI CMS

First Login to Administrator Area, then Click On Multi CMS Button As Seen Below

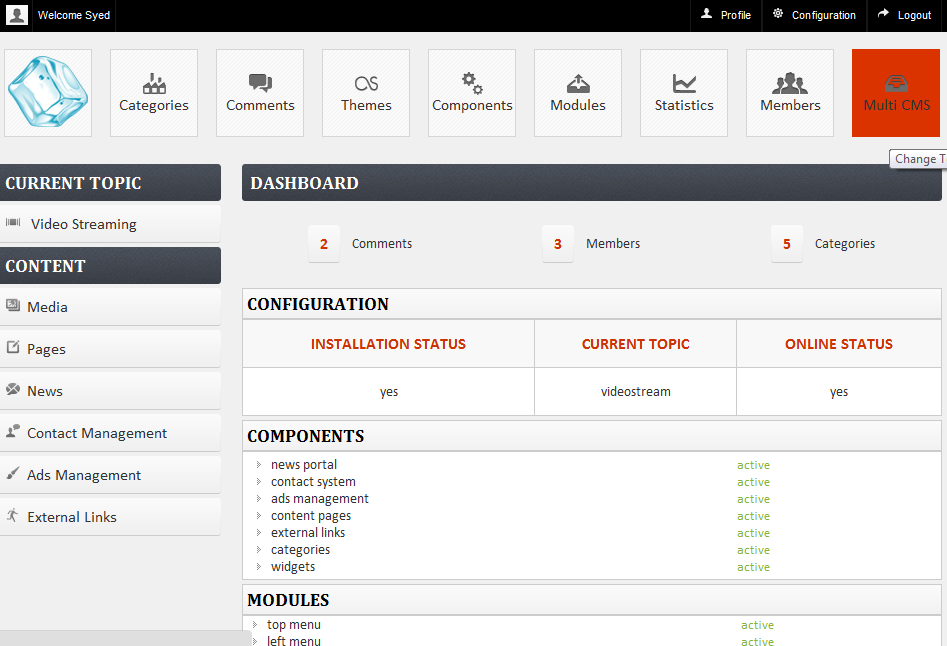


Figure 4. USING MULTI CMS

That will option the list of available topics in Multi CMS. Select one of the interested topic from the list and click change. See In Screenshot Below

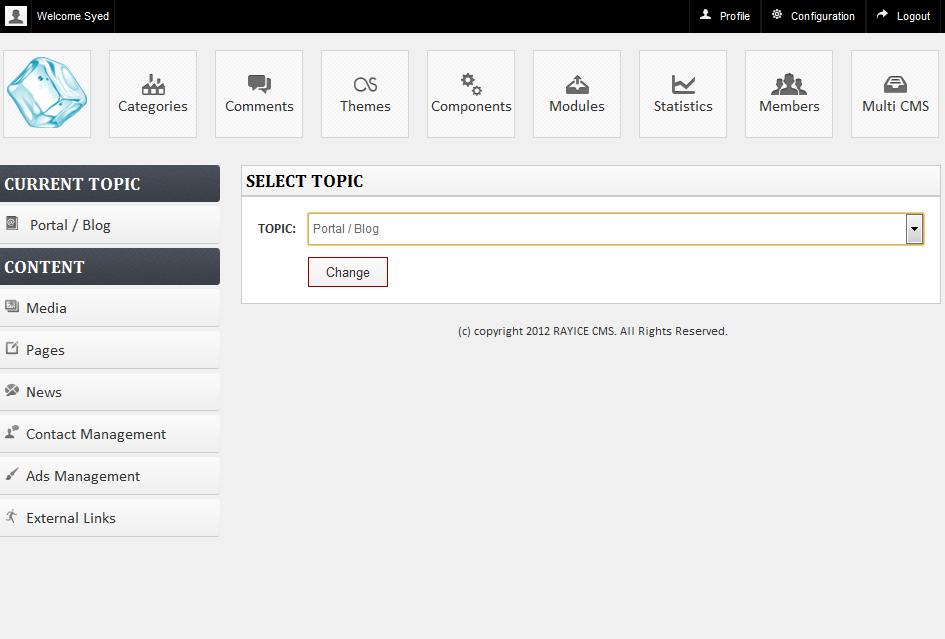


Figure 4. MULTI CMS (SELECT TOPIC)

One the Topic is Changed go to main website and you will see the whole new site is changed to selected topic as seen below

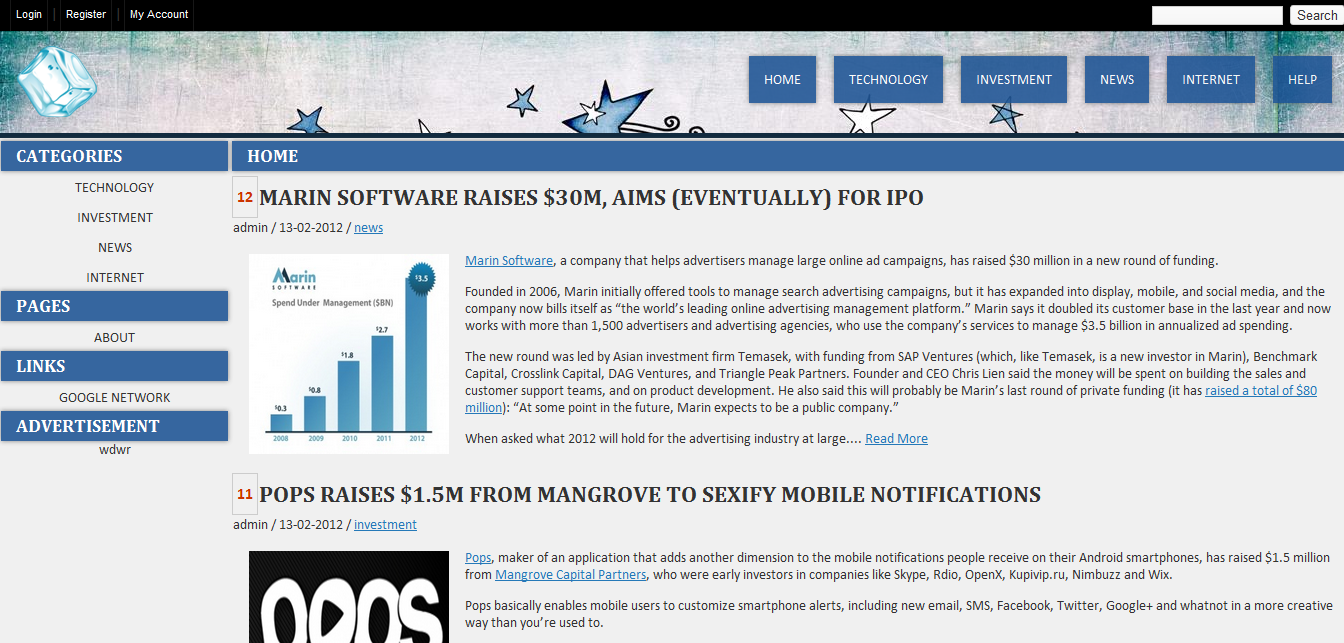


Figure 4. WEBSITE PREVIEW

# 4.4 DASHBOARD

Dashboard of website is a main page that appears after login to administrator area. This Dashboard provides user to access the different sections of admin area to manage the records and data of website. Below is the screenshot of Multi CMS Dashboard

### MAIN DASHBOARD

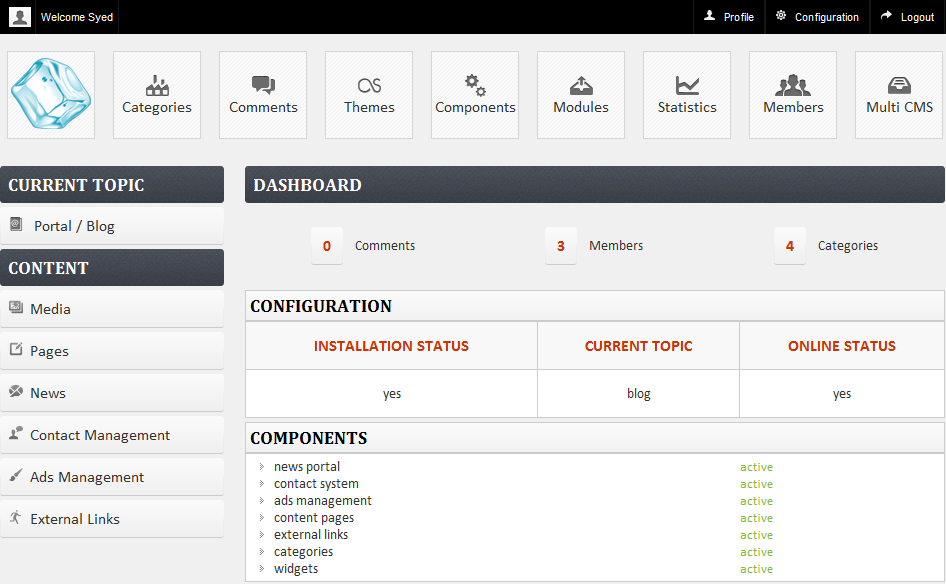


Figure 4. MAIN DASHBOARD

### ACCESS TO CONFIGURATION / PROFILE SETTING



Figure 4. TOP ADMIN BAR

This section includes Profile, Configuration, and Logout Options. Under Profile Administrator can update his personal detail. Configuration Option is a place to manage the basic configurations of the Backend / Frontend System.

### ACCESS TO MAIN SECTIONS

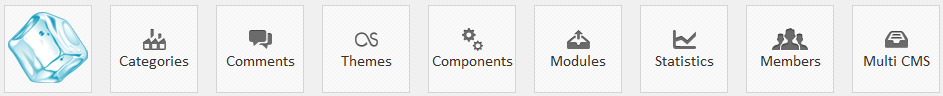


Figure 4. TOP ADMIN MENU

This includes the Access to Main Sections of Website.

* Categories : Managing the categories of Current Selected Topic
* Comments : Managing the comments of Current Selected Topic
* Themes : Managing the Themes for Frontend Look
* Components : Managing the Components of Backend/Front System
* Modules : Managing the modules of frontend Theme
* Statistics : Graphical Report of Select Topic Website Data
* Members : Managing the members of the system
* Multi CMS : Managing the Topics of Multi Content Management System

### ACCESS TO CURRENT SELECTED TOPIC



Figure 4. CURRENT TOPIC

This section shows the current select topic of the system. Once you changed the topic from Multi CMS option on Menu. The option will change to current selected topic name.

### ACCESS TO DIFFERENT CONTENTS FOR MANAGING

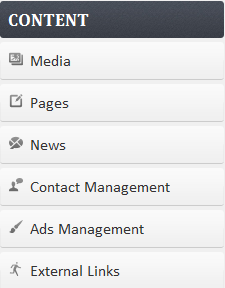


Figure 4. CONTENT SECTION

Content Section includes different options to manage the content of website that includes

* Media : Managing the Graphical Data Of website
* Pages : Managing the content pages of website
* News : Managing the news of website
* Contact Management : Managing the Messages Received From Peoples
* Ads Management : Managing the advertisement banners of website
* External Links : Managing the external links of website

### OVERVIEW OF CURRENT SELECTED TOPIC

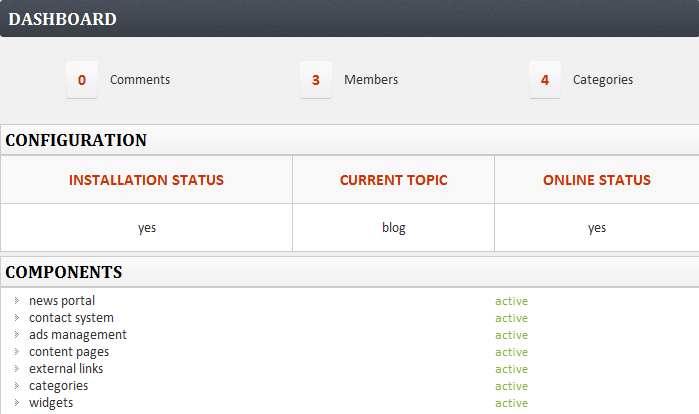


Figure 4. OVERVIEW

Above section show the information of current selected topic and overview of Main System. It includes

* Dashboard section : This shows the overview of current selected topic
* Configuration : this section show the overview of system
* Components : This show the overview of components
* Modules : This show the overview of modules

# 4.5 STATISTICS

Under Statistics we can see the graphical overview of the system that includes the

* Overview of Top10 Contents
* Overview of Pages
* Overview of Categories
* Overview of Members
* Overview of Overall System

Below are the screenshots of statistics area that show the graphical detail of the website contents according to user visits, graphical detail of website pages according to user visits, overview of categories / members according to list view.

### STATISITICS SECTION

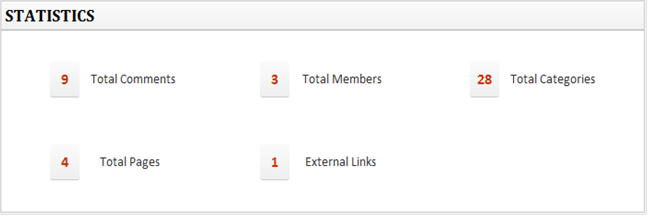


Figure 4. STATISTICS

Above screenshot show the overall detail of the system that includes

* Total Comments
* Total Members
* Total Categories
* Total Pages
* External Links

TOP10 CONTENTS

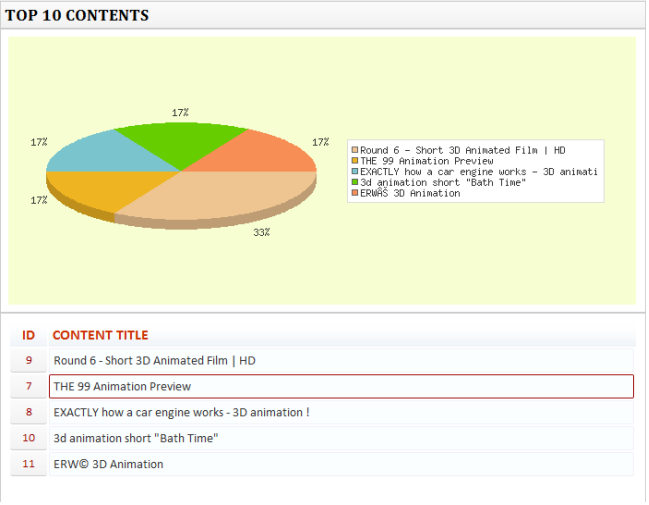


Figure 4. TOP10 CONTENTS

### CONTENT PAGES OVERVIEW

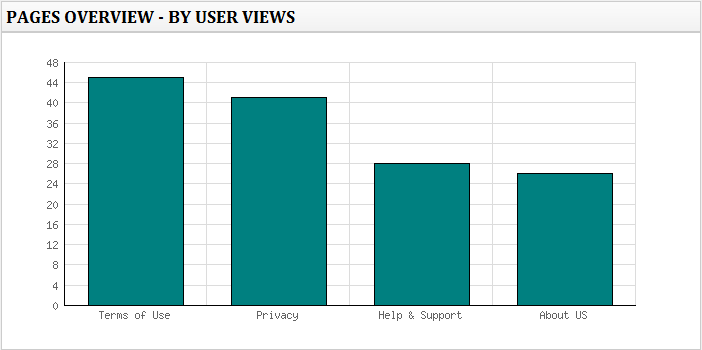
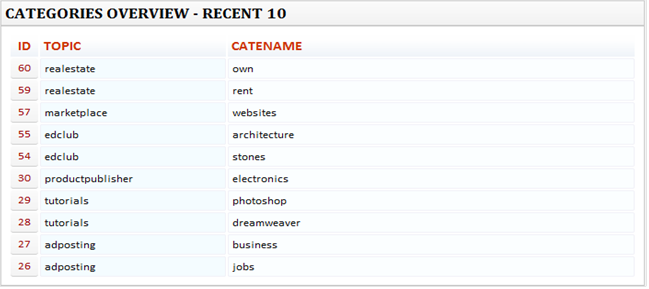


Figure 4. CONTENT PAGES OVERVIEW

Above shows the content pages overview by popularity.

### CATEGORIES OVERVIEW

  
  
Figure 4. CATEGORIES OVERVIEW

Above shows the recent 10 categories that have been added to system. This information is not for selected topic, it shows the categories from all topics available in the system.

### MEMBERS OVERVIEW

  
  
Figure 4. MEMBERS OVERVIEW

## 4.5.1 CONFIGURATION

This includes the configuration of main system. We can manage the main information of website example

* Username
* Password
* Site Title
* Site Url
* Changing installed status
* Changing Site Online Status
* Meta Description
* Meta Keywords
* Footer Text
* Logo of website
* Favicon of website and few others.

Below is screenshot of Configuration Page

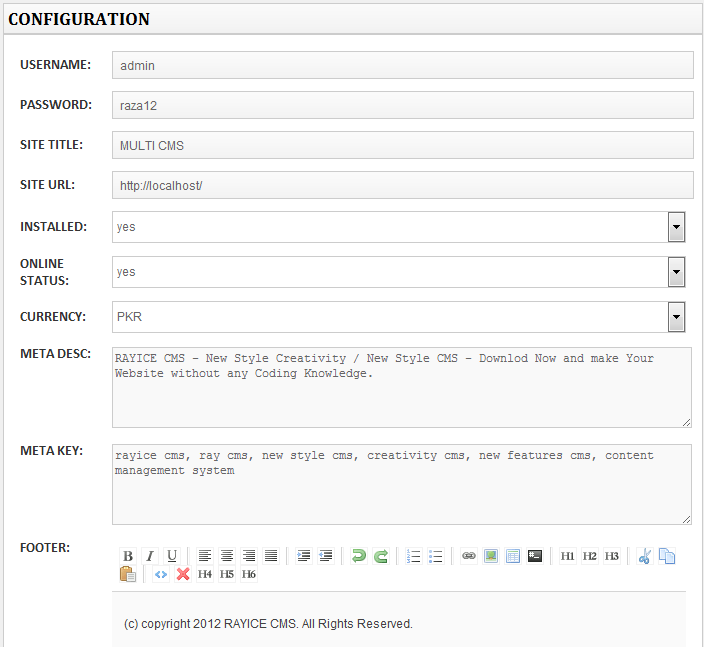


Figure 4. CONFIGURATIONS

# 4.6 SYSTEM DESIGN

Here we will discuss about the system design. We have provided the diagrammatically design of Multi Content Management System. We have created Entity Relation Ship Diagram and Data Flow Diagram for our system. ERD is drawn from the required tables and entities.

## 4.6.1 DATABASE TABLES:

**Please Look at APPENDIX (B) For Database Tables**

## 4.6.2 ENTITIES:

**Please Look at APPENDIX (C) For Entities**

## 4.6.3 ENTITY RELATIONSHIP DIAGRAM:

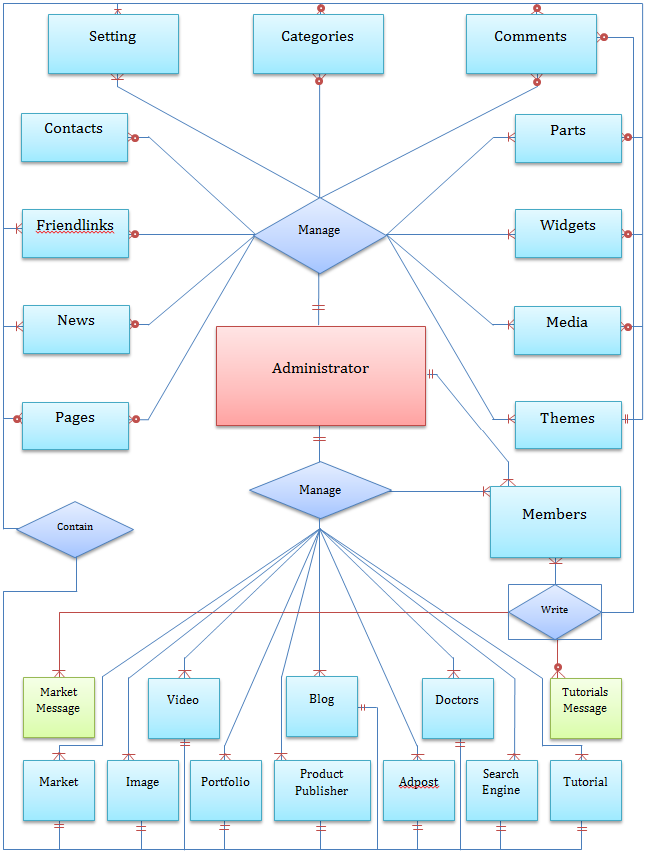


Figure 4. ERD Diagram

## 4.6.4 DATA FLOW DIAGRAM: LEVEL 0

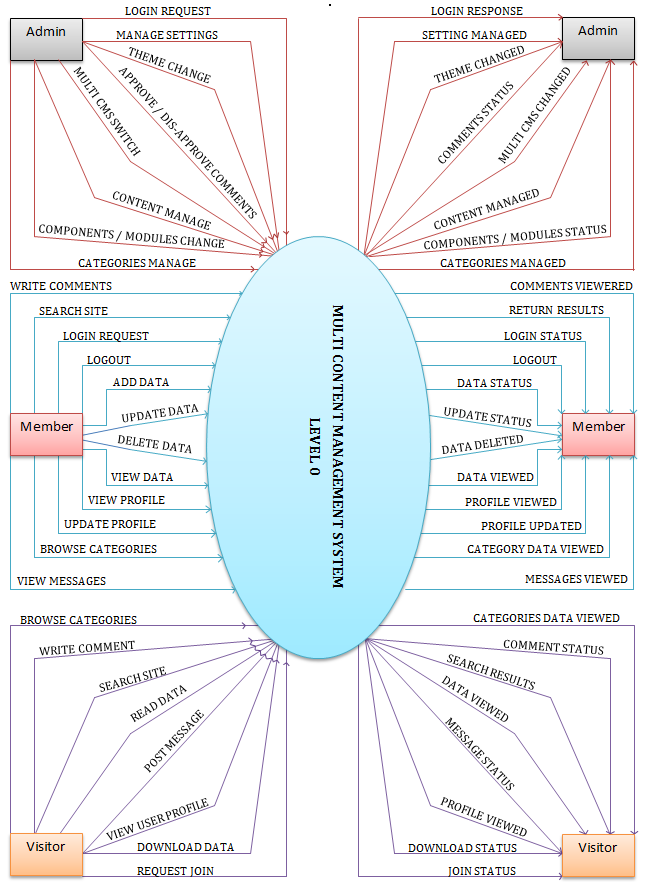


Figure 4. DFD: Level 0 Diagram

## LEVEL 1.1

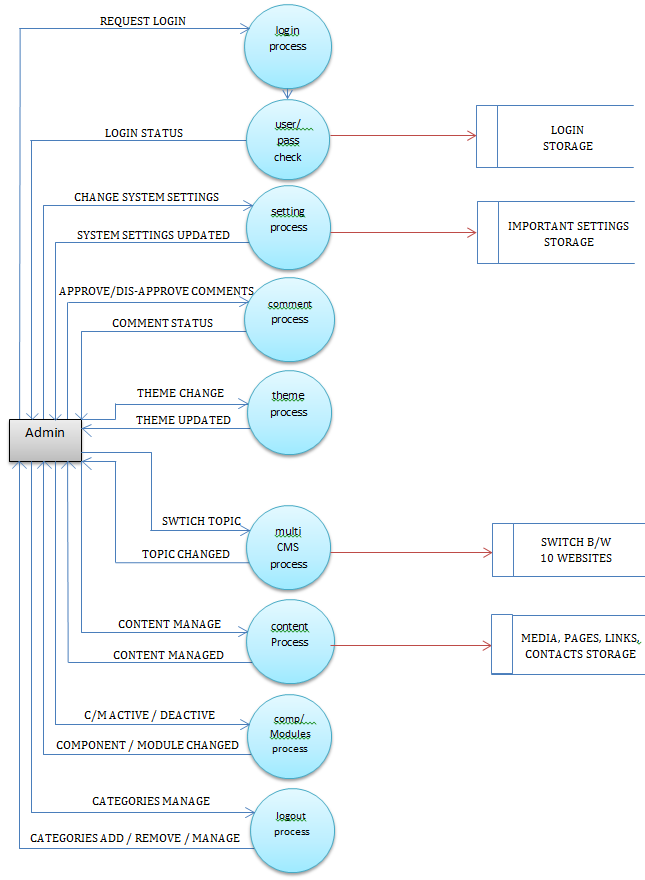


Figure 4. DFD: Level 1.1

## LEVEL 1.2

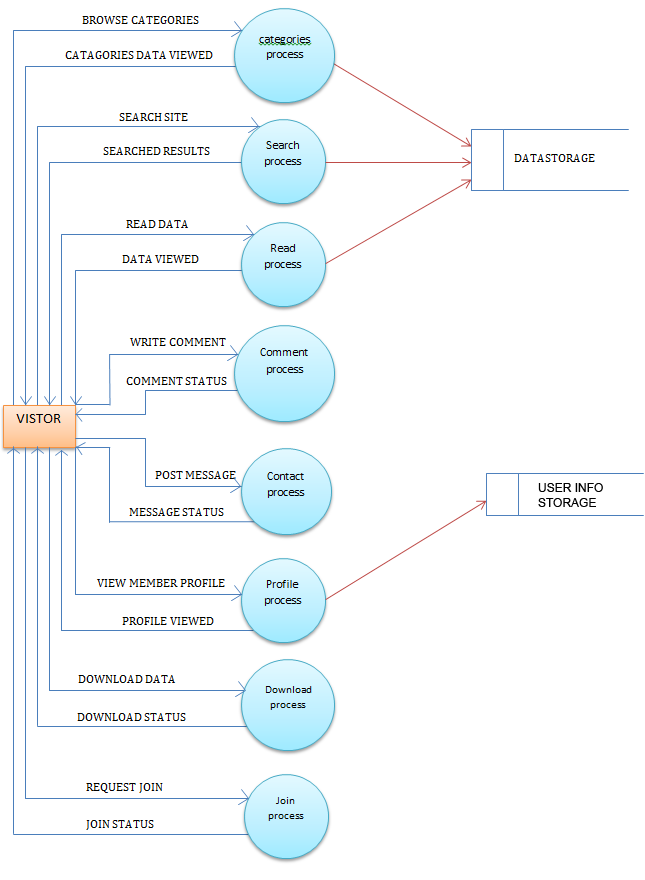


Figure 4. DFD: Level 1.2

## LEVEL 1.3

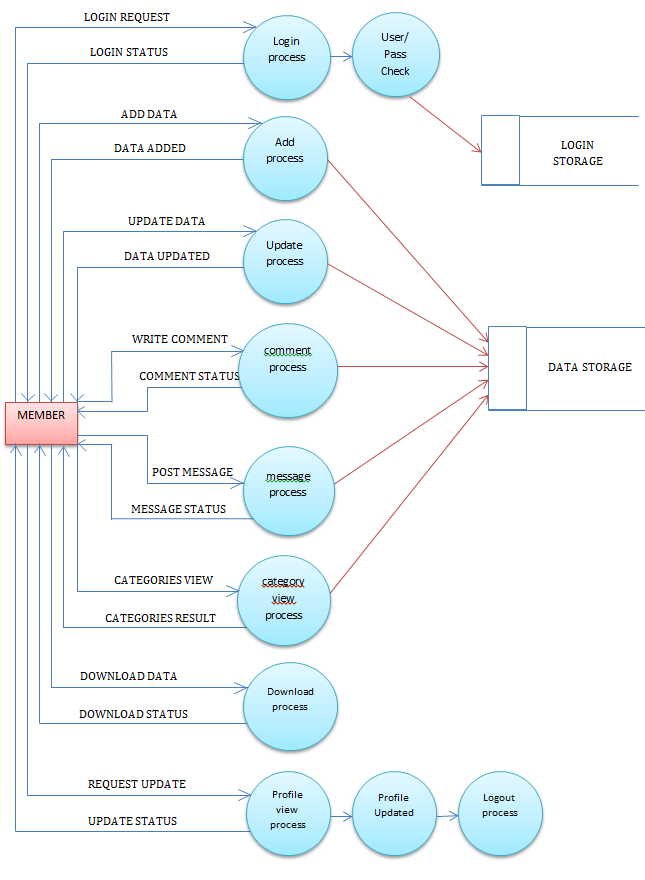


Figure 4. DFD: Level 1.3

## 4.6.5 SITEMAP

MultiCMS  
Frontned

External Links

Categories

Search

Account

Pages

Redirect  
to Links

View  
Pages

List of   
Content by  
Categories

List of  
Content by  
Search

Register

Login

Manage  
Content

View Content

Write  
Comment

Figure 4. Sitemap

## 4.6.6 FRONTEND FLOW CHART

FRONTEND WEBSITE

Register

Categories

Contents

Comments

Search

Login

If No

LoginCheck

Main  
Content

If Yes

Manage  
Content

Figure 4. Frontend Flow Chart

## 4.6.7 BACKEND FLOW CHART

Admin Login

LoginCheck

If No

If Yes

Multi CMS

Statistics

Modules

Current  
Topic

Components

Configs

Themes

Manage

List of   
Components / Modules

Select  
Topic

Select

Active / Deactive

Install

If Yes

If No

Dashboard

Check

Figure 4. Backend Flow Chart

CHAPTER 5

IMPLEMENTATION

Below are the few tools introduction, these tools and languages has been used to implement our system.

# 5.1 DESIGN & DEVELOPMENT TOOLS

We have used several Design & Development tools to complete our project, those tools are professional and need some professional skills to use them deeply in order to build and design a professional and good looking stuff or a system. We have used few of the popular Designing and Development tools for our Multi CMS that are:

## 5.1.1 ADOBE DREAMWEAVER CS 5.5

### INTRODUCTION

Adobe® Dreamweaver® CS5.5 is the industry-leading web authoring and editing software that provides both visual and code-level capabilities for creating standards-based websites and designs for the desktop, smartphones, tablets, and other devices.

****

Figure 5. 1 Dreamweaver

### PERSONAL EFFORTS

Above Dreamweaver CS5.5 is a professional Editor that helps the developers to write the codes and view there visual looks during work progress, this help the developers to safe the time in coding and help achieving the goals faster. We used this Editor to build our Multi CMS, we created our Multi CMS programming’s that contain HTML, PHP and JavaScript using Abode Dreamweaver CS5.5.

## 5.1.2 ADOBE PHOTOSHOP CS 5.5

### INTRODUCTION

The newest version of Adobe® Photoshop® CS5 software redefines digital imaging with breakthrough tools for photography editing, superior image selections, realistic painting, and more. And now, use it with creativity-boosting mobile apps.



Figure 5. 2 Photoshop

### PERSONAL EFFORTS

Above Photoshop CS5.5 is a professional Designing Software that helps the designers to design the banners, logos, website graphics, and few other elements at high quality levels. We have used this professional software to create Multi CMS graphical elements, themes, icons, banners, logos and other things.

## 5.1.3 PHPMyAdmin

PHPMyAdmin is a web based Tool that help creating database without having to enter the queries. Databases directly create in PHPMyAdmin for mysql database and then connect to php, where user call records from mysql database using php queries. We have used this software to create our Multi CMS database.

## 5.1.4 HTML5

### INTRODUCTION

HTML or Hypertext Markup Language is a formatting language that programmers and developers use to create documents on the Web. The latest edition HTML5 has enhanced features for programmers such as <video>, <audio> and <canvas> elements. You view a Web page written in HTML in a Web browser such as Internet Explorer, Mozilla Firefox or Google Chrome. The HTML5 language has specific rules that allow placement and format of text, graphics, video and audio on a Web page. Programmers use these programming tags or elements to produce web pages in unique and creative ways. Tags such as <section>, <article>, <header> enable the creator to make a more efficient and intelligent web page. Users will not have to use a Flash plug-in for video and audio content. Visual Studio users typically write code in HTML5 when creating web site content.



Figure 5. Html5

### PERSONAL EFFORTS

Html5 is an enhanced version of html; it provides great features like Javascript. It has new programming methods defined for providing more friendly websites. We have used this programming for our Multi CMS website in order to give users better Multi CMS solution.

## 5.1.5 CSS3

the latest and greatest standard that’s all the rage with modern websites and the browsers we use to view them. The rest of us may have heard the buzz about CSS3 -- but do you know what it is, how it works and why it matters? Read on to find out.

“CSS” is an acronym for Cascading Style Sheets, a web-based markup language used to describe the look and formatting of a website to the browser, most commonly used in HTML or XHTML web pages but also applicable to XML documents, including plain XML, SVG and XUL. “CSS3” simply refers to the latest incarnation of CSS, with additional capabilities far beyond the scope of the first two generations.



Figure 5. CSS3

### PERSONAL EFFORTS

CSS3 is a new enhancement of old CSS style sheets, css3 include different great options to provide better look and feel into your web pages. We have used CSS3 for our Multi CMS themes, where we have added new transparent, dropdown shadows and few other great effects in themes for making look of sites professional and simple.

# 5.2 PSEUDOCODE

## 5.2.1 ADMINISTRATOR LOGIN

Access Login Page

Get Username

Get Password

If (Username Equal EnteredUsername AND Password Equal EnteredPassword) THEN

Login Success

Redirect to DashBoard

Else

Login Failed

Redirect to Login Page

EndIf

## 5.2.2 ADMINISTRATOR DATA MANAGE

Access Selected Topic

### 3 TYPES:

**Add Data**

Get Title

Get Description

Get Image

Get Other Details

If (All Fields Fill && Data Types Are Correct)

Data Added

Else

Data No Added

Endif

**Update Data**

Get Data

If (isset (Dollar\_Get Data))

Data Updated

Else

Data Not Updated

EndIf

**Delete Data**

Get Data Id (GET From Url)

Deleted Data

Return Success

## 5.2.3 FRONTEND SEARCH

Get Query

If (Get Query == Database Query)

Show Results

Else

Show Empty Message

EndIf

# 5.3 CODING

In this section we have described the coding we have used for our system. We have used most powerful programming languages for our system to build it and prepare it using advance functions based method.

## 5.3.1 MYSQL Language

Mysql is a database language through which we can create the databases in which we can save the records and results of our system. There are few other databases available like Database Access and Database Sql. As we have used PHP Programming for our Multi CMS and Mysql is more familiar and supported language with PHP so we have decided to use this language for our system. Below are some queries examples we have used in our system.

**Please Look at Appendix (D) For Database Queries**

## 5.3.2 PHP

PHP stands for personal home page. This is most powerful web based language through which we can create online software’s, online tools and graphs, charts and some great things. Recently we discussed about the most popular online Content management a system that involves Wordpress, Drupal, and Joomla. All of these are created using PHP programming. The reason behind the popularity of PHP programming is that it is really easy to use and is very powerful language that provides high level of security on the system depend on how much programming has been done and what conditions has been applied on different results. We have used PHP programming language for our Multi CMS, and we have used different coding patterns to complete the programming of our system. Below are some of the important codes containing in our Multi CMS.

Below is php query code fetching record from setting table

**Please Look at Appendix (E) For Php Codes**

CHAPTER 6

TESTING / RESULTS

# 6.1 TESTING INTRODUCTION

Testing is the systematic attempt to find errors in a planned way in the implemented software. In other words, testing is the process of demonstrating that errors are not exists.

# 6.2 TESTING STRATEGY

Two basic strategies that were used for testing were

* BlackBox Testing
* WhilteBox Testing

## 6.2.1 BLACKBOX TESTING

Even after the code testing has been performed exclusively, it does not ensure against program failure. Code testing does not indicate whether the code meets the specifications or whether all the aspects are even implemented. Therefore examining specifications starting what program should do and how it should perform under various conditions performed specification testing.

### TEST CASES

A test case is a set of input data and expected results that exercises a component with the purpose of causing failures and detecting results. Test cases were developed to test the range of values expected including both valid and invalid data. It helped in finding discrepancies between the system and its original objectives, current specification and system documentation. During testing phase, all efforts were made to remove programming bugs and minor designed faults.

Table 6. TEST CASES

|  |  |  |
| --- | --- | --- |
| TEST CASE | TEST CASE ID | RESULT |
| CASE 1 | Verify that user can view the login page | Test 1 Passed |
| CASE 2 | Verify that user redirect to login page when entered wronge password | Test 2 Passed |
| CASE 3 | Verify that user gets error message on leaving an empty space in the form. | Test 3 Passed |
| CASE 4 | Verify that user redirect to categories list after enter new category. | Test 4 Passed |
| CASE 5 | Verify that administrator can view the Multi CMS to change topic | Test 5 Passed |
| CASE 6 | Verify that topic is changed and site is changed as well on frontend | Test 6 Passed |
| CASE 7 | Verify that administrator do not leave any field emty. | Test 7 Passed |
| CASE 8 | Verify that all the record is showing on each section of administrator | Test 8 Passed |
| CASE 9 | Verify that all the edit pages displayed for each record in administrator | Test 9 Passed |
| CASE 10 | Verify that administrator can Submit the edited records | Test 10 Passed |
| CASE 11 | Verify that administrator can Delete the records | Test 11 Passed |
| CASE 12 | Verify that administrator can change the theme of the site | Test 12 Passed |
| CASE 13 | Verify that Member can edit record on each site | Test 13 Passed |
| CASE 14 | Verify that Member can remove record on each site | Test 14 Passed |
| CASE 15 | Verify that Member can update his profile | Test 15 Passed |
| CASE 16 | Verify that Member can Adding new Record to each site | Test 16 Passed |
| CASE 17 | Verify that system components are activating correctly | Test 17 Passed |
| CASE 18 | Verify that system modules are activating and displaying in frontend | Test 18 Passed |
| CASE 19 | Verify that widgets are activating and display in frontend | Test 19 Passed |

### FUNTIONAL TESTING

## That is applied into Registeration Page of Multi CMS

## Username: Username Minimum Limit is 4 and Maximum Limit is 20

## Password: Password Minimum Limit is 8 and Maximum Limit is 30

## Full Name: Full Name Minimum Limit is 4 and Maximum Limit is 20

## Address: Address Minimum Limit is 6 and Maximum Limit is 60

## Email: This type is email and its min Limit is 6 and Maximum Limit is 60

## Zip: Zip Minimum Limit is 6 and Maximum Limit is 10

## City: City minimum Limit is 2 and Maximum Limit is 25

## State: State minimum Limit is 3 and Maximum Limit is 15

## Country: Country Select List

## Phone: This type is number and its min Limit is 10 and Maximum Limit is 25

## Yahooid: YahooId maximum limit is 50

## Twitter: Twitter maximum limit is 50

## Facebook: Facebook maximum limit is 50

#### EQUIVALENCE PARTITIONING

## We are going to generate test case for the fields Username, State, Phone, and Email.

## Green For: Valid Class Red For: Below Invalid Class Orange For: Above Invalid Class

|  |  |  |
| --- | --- | --- |
| **GREEN FOR** | **RED FOR** | **ORANGE FOR** |
| **Valid Class** | Below Invalid Class | Above Invalid Class |

## 1. USERNAME LIMITATION BETWEEN

## .....0,1,2,3 < 4 TO 20 > 21,22,23,24.....

## 2. STATE FIELD WORDS LENGTH LIMIT

## ..... 1, 2 < 3 TO 15 > 16, 17, 18, 19.....

## 3. PHONE FIELD DIGITS LIMIT

## .....7,8,9 < 10 to 25 > 26,27,28,29.....

## 4. EMAIL FIELD WORDS LENGTH LIMIT

## .....3, 4, 5, 6 < 6 TO 60 > 61, 62, 63, 64.....

## *TEST CASES GENERATE:*

## USERNAME LIMITATION: By Characters Limit / Special Characters

|  |  |  |
| --- | --- | --- |
| Below Invalid Class | Valid Class | Above Invalid Class |
| 1 Char’s Entered (Invalid) **Message Return :** Minimum allow char’s 4 | 4 Char’s Entered (Valid) Status : Accepted | 21 Char’s Entered (Invalid) Message Return : Maximum allow char’s 20 |
| 2 Char’s Entered (Invalid) **Message Return :** Minimum allow char’s 4 | 5 Char’s Entered (Valid) Status : Accepted | 45 Char’s Entered (Invalid) Message Return : Maximum allow char’s 20 |
| 3 Char’s Entered (Invalid) **Message Return :** Minimum allow char’s 4 | 15 Char’s Entered (Valid) Status : Accepted | 30 Char’s Entered (Invalid) Message Return : Maximum allow char’s 20 |
| 0 Char’s Entered (Invalid) **Message Return :** Minimum allow char’s 4 | 20 Char’s Entered (Valid) Status : Accepted | 25 Char’s Entered (Invalid) Message Return : Maximum allow char’s 20 |

## STATE FIELD WORDS LENGTH LIMIT: By Characters Limit

|  |  |  |
| --- | --- | --- |
| Below Invalid Class | Valid Class | Above Invalid Class |
| 1 Word Entered (Invalid) **Message Return :** Minimum allow char’s 3 | 5 Words Entered (Valid) Status : State is accepted | 16 Words Entered (Invalid) Message Return : Maximum Allow Char’s 15 |
| 2 Words Entered (Invalid) **Message Return :** Minimum allow char’s 3 | 9 Words Entered (Valid) Status : State is accepted | 40 Words Entered (Invalid) Message Return : Maximum Allow Char’s 15 |
| 2 Words Entered (Invalid) **Message Return :** Minimum allow char’s 3 | 14 Words Entered (Valid) Status : State is accepted | 60 Words Entered (Invalid) Message Return : Maximum Allow Char’s 15 |
| 0 Words Entered (Invalid) **Message Return :** Minimum allow char’s 3 | 15 Words Entered (Valid) Status : State is accepted | 88 Words Entered (Invalid) Message Return : Maximum Allow Char’s 15 |

## PHONE FIELD DIGITS LIMIT: By Numbers Limit

|  |  |  |
| --- | --- | --- |
| Below Invalid Class | Valid Class | Above Invalid Class |
| 2 Digits Entered (Invalid) **Message Return :** Minimum allow Digit’s 10 | 16 Digits Entered (Valid) Status : Accepted | 45 Digits Entered (Invalid) Message Return : Minimum allow Digit’s 25 |
| 1 Digits Entered (Invalid) **Message Return :** Minimum allow Digit’s 10 | 12 Digits Entered (Valid) Status : Accepted | 33 Digits Entered (Invalid) Message Return : Minimum allow Digit’s 25 |
| 8 Digits Entered (Invalid) **Message Return :** Minimum allow Digit’s 10 | 15 Digits Entered (Valid) Status : Accepted | 26 Digits Entered (Invalid) Message Return : Minimum allow Digit’s 25 |
| 0 Digits Entered (Invalid) **Message Return :** Minimum allow Digit’s 10 | 20 Digits Entered (Valid) Status : Accepted | 27 Digits Entered (Invalid) Message Return : Minimum allow Digit’s 25 |

## EMAIL FIELD WORDS LENGTH LIMIT: By Email Type

|  |  |  |
| --- | --- | --- |
| Below Invalid Class | Valid Class | Above Invalid Class |
| 12as Entered (Invalid) **Message Return :** Email Format is invalid or Too Short | a@asd.com (Valid) Status : Entered Mail is accepted | 61 Words Entered (Invalid) Message Return : Email Format invalid or Too Long |
| 123 Entered (Invalid) **Message Return :** Email Format is invalid or Too Short | ask@rayice.com (Valid) Status : Entered Mail is accepted | 66 Words Entered (Invalid) Message Return : Email Format is invalid or Too Long |
| dfa5sa Entered (Invalid) **Message Return :** Email Format is invalid or Too Short | rayice@gmail.com (Valid) Status : Entered Mail is accepted | 89 Words Entered (Invalid) Message Return : Email Format is invalid or Too Long |
| Aslf2 Entered (Invalid) Message Return : Email Format is invalid or Too Short | rayice@ymail.com (Valid) Status : Entered Mail is accepted | 105 Words Entered (Invalid) Message Return : Email Format is invalid or Too Long |

#### BOUNDARY VALUE ANALYSIS

## We are going to analyze the boundary values of fields Username, State, Phone, and Email.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GREEN FOR** | **RED FOR** | **ORANGE FOR** | **BLUE FOR** | **PINK FOR** |
| **Valid Class** | Below Invalid Class | Above Invalid Class | Lower Boundary | High Boundary |

## 1. USERNAME LIMITATION BETWEEN

## .....0, 1, 2, 3 < 4 TO 20 > 21, 22, 23, 24.....

## 1, 2, 3, 4, 5, 6, 7 -------- 17, 18, 19, 20, 21, 22, 23

## IMMEDIADE BACK: 3 and 19 IMMEDIATE FRONT: 5 and 21

## 2. STATE FIELD WORDS LENGTH LIMIT

## .....1, 2, 3, 4 < 3 TO 15 > 16, 17, 18, 19.....

## 2, 3, 4, 5, 6, 7, 8 -------- 12, 13, 14, 15, 16, 17, 18

## IMMEDIADE BACK: 3 and 14 IMMEDIATE FRONT: 6 and 16

## 3. PHONE FIELD DIGITS LIMIT

## .....6, 7, 8, 9 < 10 TO 25 > 26, 27, 28, 29.....

## 7, 8, 9, 10, 11, 12, 13 -------- 22, 23, 24, 25, 26, 27, 28

## IMMEDIADE BACK: 9 and 24 IMMEDIATE FRONT: 11 and 26

## 4. EMAIL FIELD WORDS LENGTH LIMIT

## .....2, 3, 4, 5 < 6 TO 60 > 61, 62, 63, 64.....

## 1, 2, 4, 5, 6, 7, 8, 9------------57, 58, 59, 60, 61, 62, 63

## IMMEDIADE BACK: 5 and 59 IMMEDIATE FRONT: 7 and 61

#### CAUSE EFFECT GRAPHING

## We are going to test cause effect graphing in same module account creation form. We will test it in 3 different statements that’s will be used in account creation form.

## 1. USERNAME LIMITATION BETWEEN

## .....0, 1, 2, 3 < 4 TO 20 > 21, 22, 23, 24.....

## Normal Statement:

## If ((Min >= 4) AND (Max <= 20)) { Echo “Username Accepted” } else { Echo “Characters Limitation Error” //(Error can be change) }

## Decision Table:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test #1** | **Test #2** | **Test #3** |
| CAUSE: |  |  |  |
| Min >= 4 | T | F | T |
| Max <= 20 | F | T | T |
| EFFECT: |  |  |  |
|  | Characters Limitation Error | Characters Limitation Error | Username Accepted |

## 2. STATE LIMITATION BETWEEN

## .....1, 2, 3, 4 < 5 TO 15 > 16, 17, 18, 19.....

## 

## Normal Statement:

## If ((Min >= 3) AND (Max <= 15)) { Echo “State Accepted” } else { Echo “Characters Limitation Error” //(Error can be change) }

## Decision Table:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test #1** | **Test #2** | **Test #3** |
| CAUSE: |  |  |  |
| Min >= 3 | T | F | T |
| Max <= 15 | F | T | T |
| EFFECT: |  |  |  |
|  | Characters Limitation Error | Characters Limitation Error | State Accepted |

## 3. PHONE LIMITATION BETWEEN

## .....6, 7, 8, 9 < 10 TO 25 > 26, 27, 28, 29.....

## 

## Normal Statement:

## If ((Min >= 10) AND (Max <= 25)) { Echo “Phone Accepted” } else { Echo “Characters Limitation Error” //(Error can be change) }

## Decision Table:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test #1** | **Test #2** | **Test #3** |
| CAUSE: |  |  |  |
| Min >= 10 | T | F | T |
| Max <= 25 | F | T | T |
| EFFECT: |  |  |  |
|  | Characters Limitation Error | Characters Limitation Error | Phone Accepted |

## 4. EMAIL LIMITATION BETWEEN

## .....2, 3, 4, 5 < 6 TO 60 > 61, 62, 63, 64.....

## 

## Normal Statement:

## If ((Min >= 6) AND (Max <= 60)) { Echo “Email Accepted” } else { Echo “Characters Limitation Error” //(Error can be change) }

## Decision Table:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test #1** | **Test #2** | **Test #3** |
| CAUSE: |  |  |  |
| Min >= 6 | T | F | T |
| Max <= 60 | F | T | T |
| EFFECT: |  |  |  |
|  | Characters Limitation Error | Characters Limitation Error | Phone Accepted |

## 6.2.2 WHITEBOX TESTING

The whole system coding was tested through Dreamweaver Environment to avoid any code issues. The testing was made on each step during writing of code for Multi CMS. There are few White Box Testing Reports for few components of Multi CMS and on each unit of those components.

### 1. PHONE UNIT

## Field name of this Unit within Register Account form is “phone”, where the limit has been applied to this unit. We have limited the digits to enter minimum 10 and maximum 25. So we have set the min=10 and max=25.

#### FIGURE CODE DRAWS A FLOWGRAPH

## CODE:

**1** **2**

if (min >= “min”) AND (max <= “max”)

{

echo “Phone Accepted”; **3**

}

else

{

echo “Error Return”; **4**

}

endif ; **5**

#### FLOW GRAPH

T

F

T

F

## endif endif

**PHONE UNIT FLOW GRAPH**

#### CYCLOMATIC COMPLEXITY

**NODES** = 5

**EDGES =** 6

**REGIONS =** 3

## VERTICES OF GRAPHS V(G)

**Formula:**

**1) e-n+2**

6-5+2 = 3

**2) p+1**

2+1 = 3

#### IDENTIFY PATHS

**Path 1** = **1** – **2** – **3** – **5**

**Path 2** = **1** – **2** – **4** – **5**

**Path 3** = **1** – **4** – **5**

### 2. USERNAME UNIT

## The field name of this Unit within Account Creation form is “username”, where the limit has been applied to this unit. We have limited the age to minimum 4 and maximum 20. So we have set the min=4 and maxlength=20.

#### FIGURE CODE DRAWS A FLOWGRAPH

## CODE:

**1** **2**

If (( min >= “min”) AND ( max <= “maxlength”))

{

echo “User Accepted”; **3**

}

else

{ **4**

echo “Error Return”;

}

end if;

**5**

#### FLOW GRAPH

T

F

T

F

## 

## endif endif

**USERNAME FLOW GRAPH**

#### CYCLOMATIC COMPPLEXITY

**NODES** = 5

**EDGES =** 6

**REGIONS =** 3

## VERTICES OF GRAPHS V(G)

**Formula:  
1) e-n+2**

6-5+2 = 3

**2) p+1**

2+1 = 3

#### IDENTIFY PATHS

**Path 1** = **1** – **2** – **3** – **5**

**Path 2** = **1** – **2** – **4** – **5**

**Path 3** = **1** – **4** – **5**

### 1. EMAIL UNIT

## Field name of this Unit within Register Account form is “phone”, where the limit has been applied to this unit. We have limited the digits to enter minimum 10 and maximum 25. So we have set the min=6 and maxlength=60.

#### FIGURE CODE DRAWS A FLOWGRAPH

## CODE:

**1** **2**

if (min >= “minemail”) AND (max <= “maxemail”)

{

echo “Email Accepted”; **3**

}

else

{

echo “Error Return”; **4**

}

endif ; **5**

#### FLOW GRAPH

T

F

T

F

## endif endif

**EMAIL UNIT FLOW GRAPH**

#### CYCLOMATIC COMPLEXITY

**NODES** = 5

**EDGES =** 6

**REGIONS =** 3

## VERTICES OF GRAPHS V(G)

**Formula:**

**1) e-n+2**

6-5+2 = 3

**2) p+1**

2+1 = 3

#### IDENTIFY PATHS

**Path 1** = **1** – **2** – **3** – **5**

**Path 2** = **1** – **2** – **4** – **5**

**Path 3** = **1** – **4** – **5**

### 4. STATE UNIT

## The field name of this Unit within registration form is “state”, where the limit has been applied to this unit. We have limit of minimum characters to enter 5 and maximum is 15. So we have set the min=3 and maxlength=15.

#### FIGURE CODE DRAWS A FLOWGRAPH:

## CODE:

**1** **2**

If(state>=” min”)AND(state<=” maxlength”) F T

{

echo “Accepted”; **3** F T

}

else

{ endif endif

**STATE UNIT FLOW GRAPH**

echo “less than 3 or more than 15 characters are not allowed”;

**4**

}

endif;

**5**

#### CYCLOMATIC COMPPLEXITY:

**NODES** = 5

**EDGES =** 6

**REGIONS =** 3

## VERTICES OF GRAPHS V(G)

**Formula:  
1) e-n+2**

6-5+2 = 3

**2) p+1**

2+1 = 3

#### IDENTIFY PATHS:

**Path 1** = **1** – **2** – **3** – **5**

**Path 2** = **1** – **2** – **4** – **5**

**Path 3** = **1** – **4** – **5**

# 6.3 SECURITY

This system is designed for open source access to use. This system has registration system on different locations, on each site separate registration. Registered users can add the records to selected Multi CMS site from Account Area. It has high level security defined to avoid any hacking issue or wronged account access issue.

# 6.4 SYSTEM EVALUATION

When the system is implemented successfully, the designer evaluates the system to see whether the objective f the system is accomplished or note. Also none of the developed system is a complete one. There is always need for improvements. An exercise may achieve an immediate goal and arrive at a state that logically seems an appropriate point for one project may be a good beginning of another. So, discussing features of the developed system and future enhancement carryout the evaluation of the system.

The system fulfilling the following objectives:

* The Project is fulfilling the following objectives:
* User can join the website and can post the data
* User can write message to other members
* User can Download the Data
* User can Watch the Videos and can View The Images
* User can Create his own Portfolio site
* User can add links to data search engine
* Administrator can manage the Content of site
* Administrator can Switch the Multi CMS sites
* Administrator can active/deactivate Components/Modules
* Administrator can change site setting
* Administrator can manage categories
* Administrator can delete members
* Visitors can search the site
* Visitors can read the data
* Visitors can comment on data
* Visitors can download resources
* Visitors can watch the videos
* Visitors can view the images

# 6.5 FUTURE ENHANCEMENTS

In the future we have decided to improve this system to provide more great readymade websites within one Content Management System. We are planning to provide the better solution to internet users and a planning to provide more and greater features like more themes, components, modules, widgets and much more with this Multi CMS in feature. This project is specially designed for Final Project however we are planning to use it for our Online Business to make our position better on the internet as well to provide more easy solution to internet users so they can run sites of their own choice without having to face the tuff situations.

CHAPTER 7

APPENDICES

# APPENDIX (A) SCREENSHOTS

## LOGIN PAGE

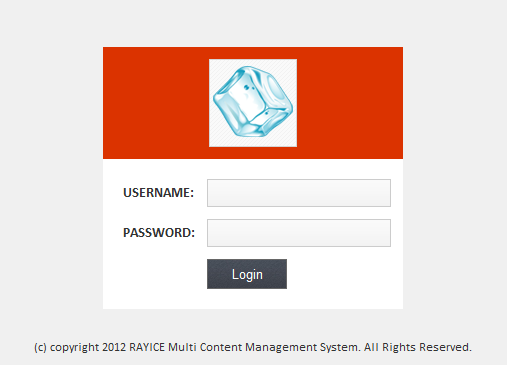


Figure 7. Login Screen

## ADMINISTRATOR MAINPAGE

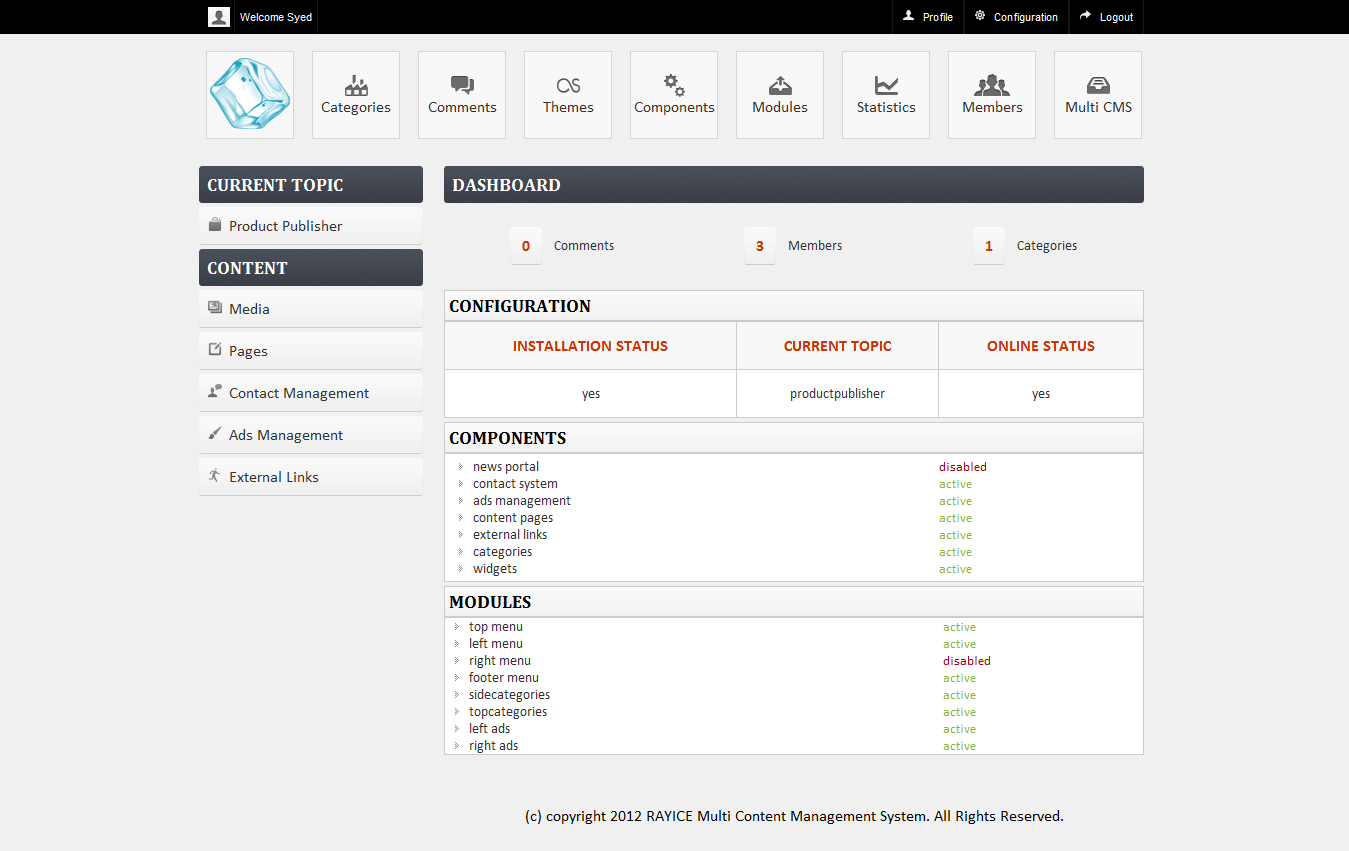


Figure 7. Administrator MainPage

## MULTI CMS

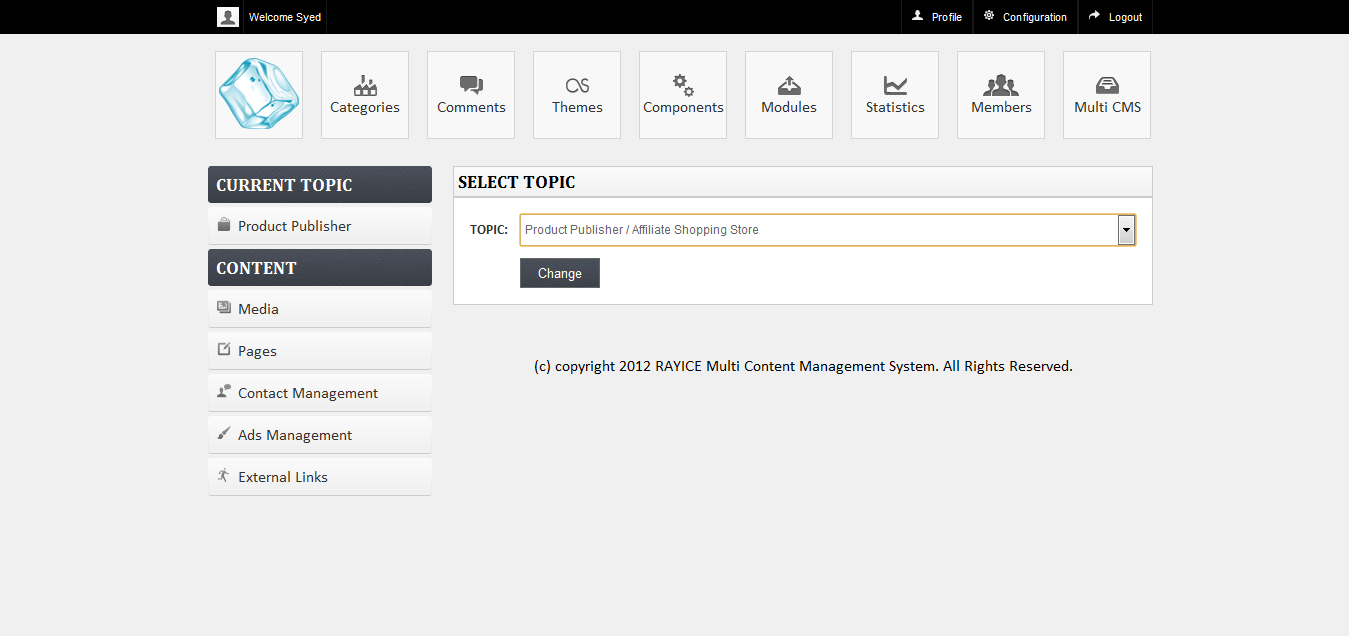


Figure 7. Multi CMS Topics List

## PROFILE

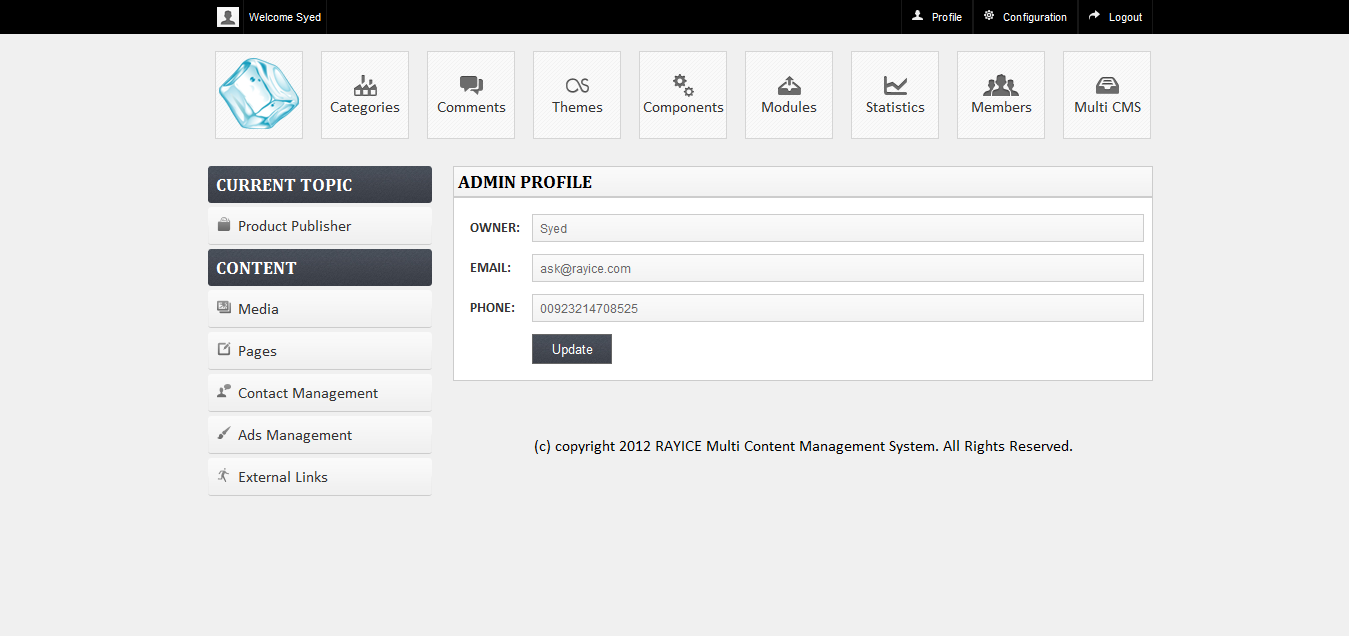


Figure 7. Administrator Profile

## CONFIGURATION

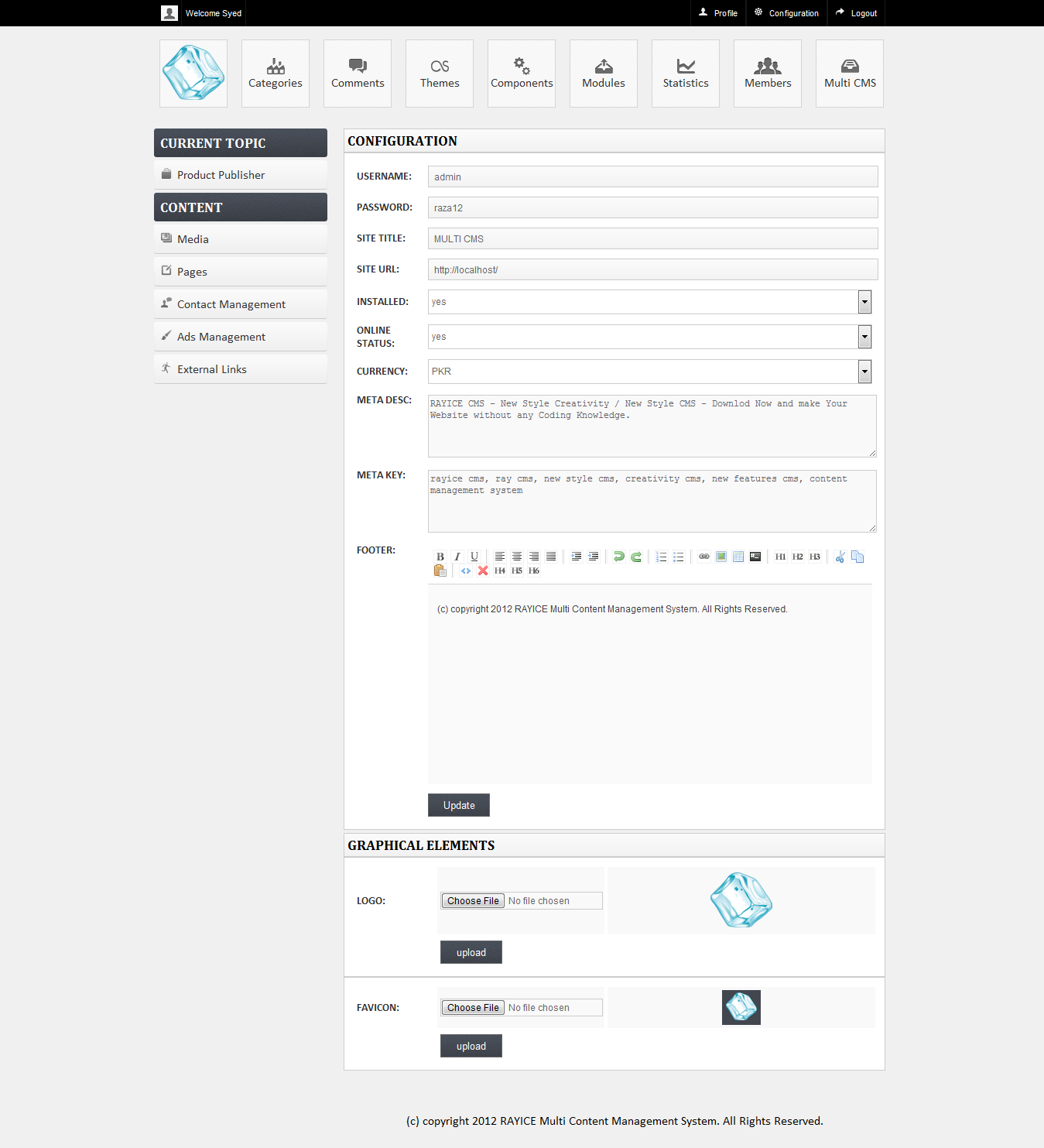


Figure 7. Configuration

## CATEGORIES

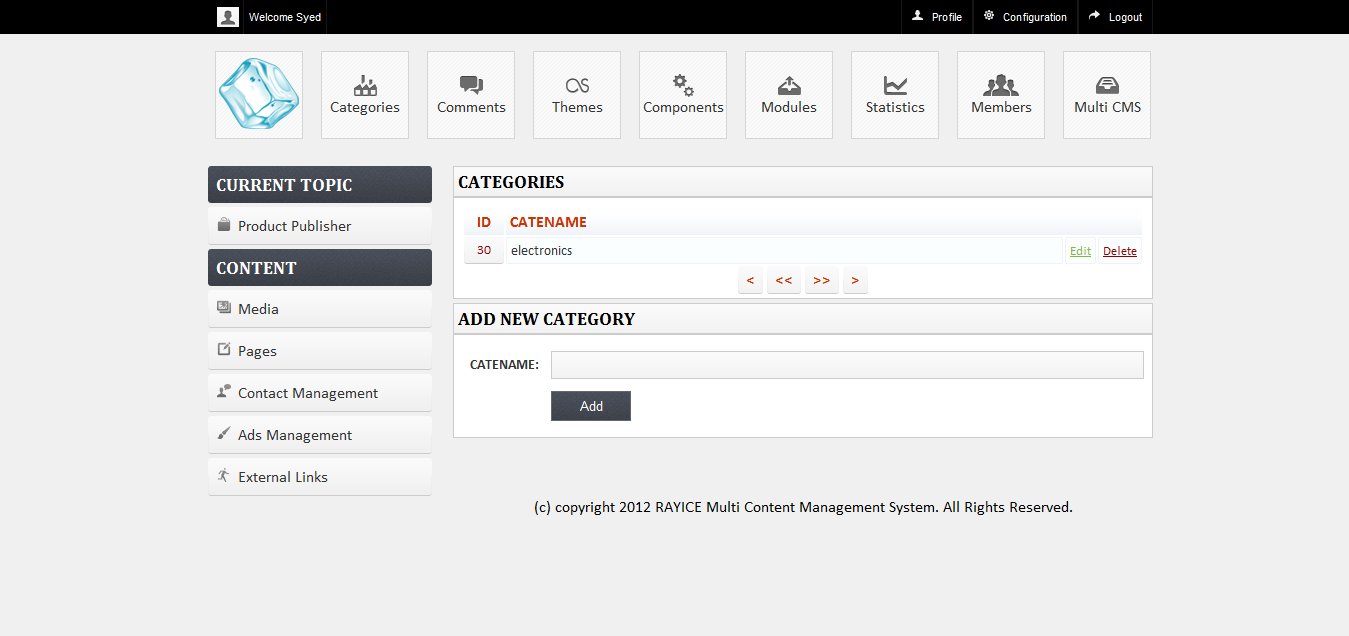


Figure 7. Categories

## THEMES

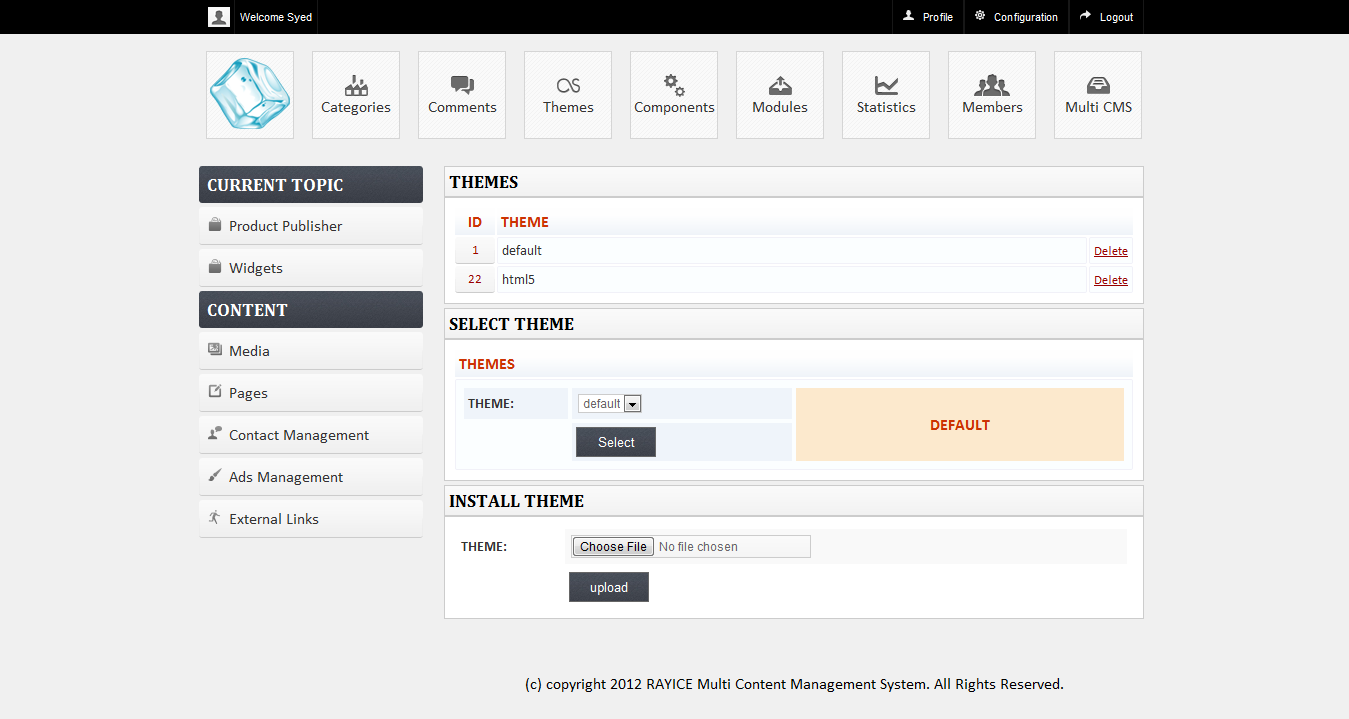


Figure 7. 7 Themes

## COMPONENTS

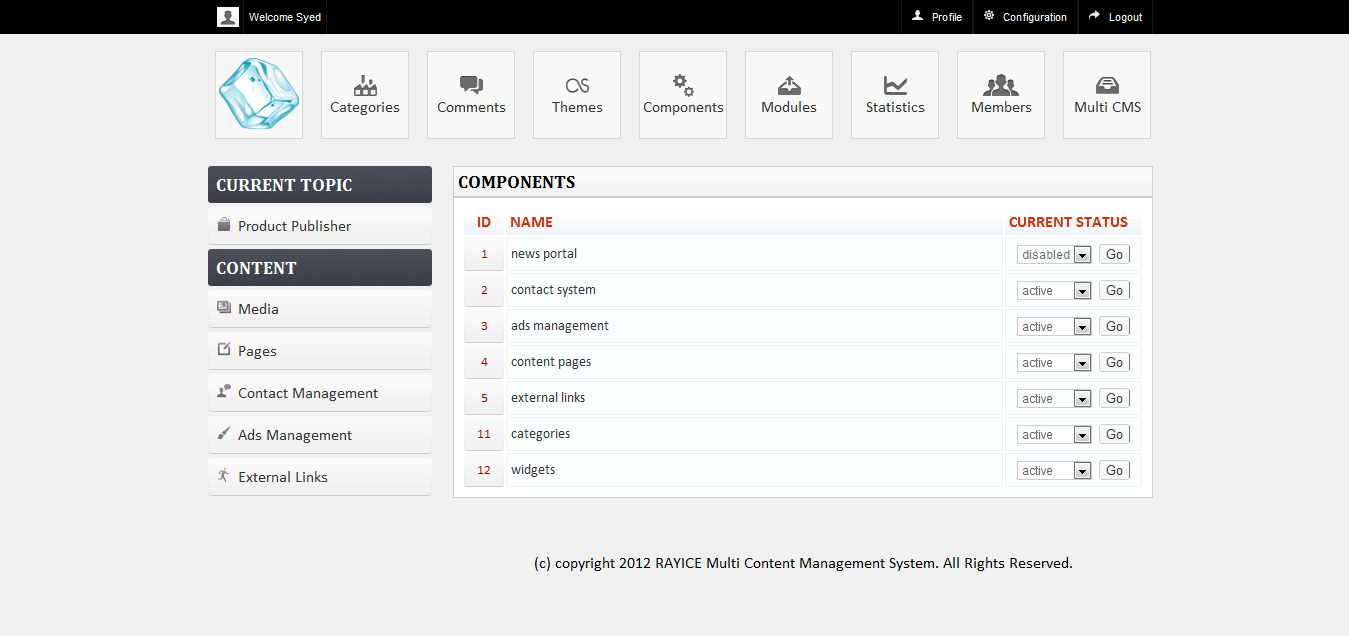


Figure 7. Components

## MODULES

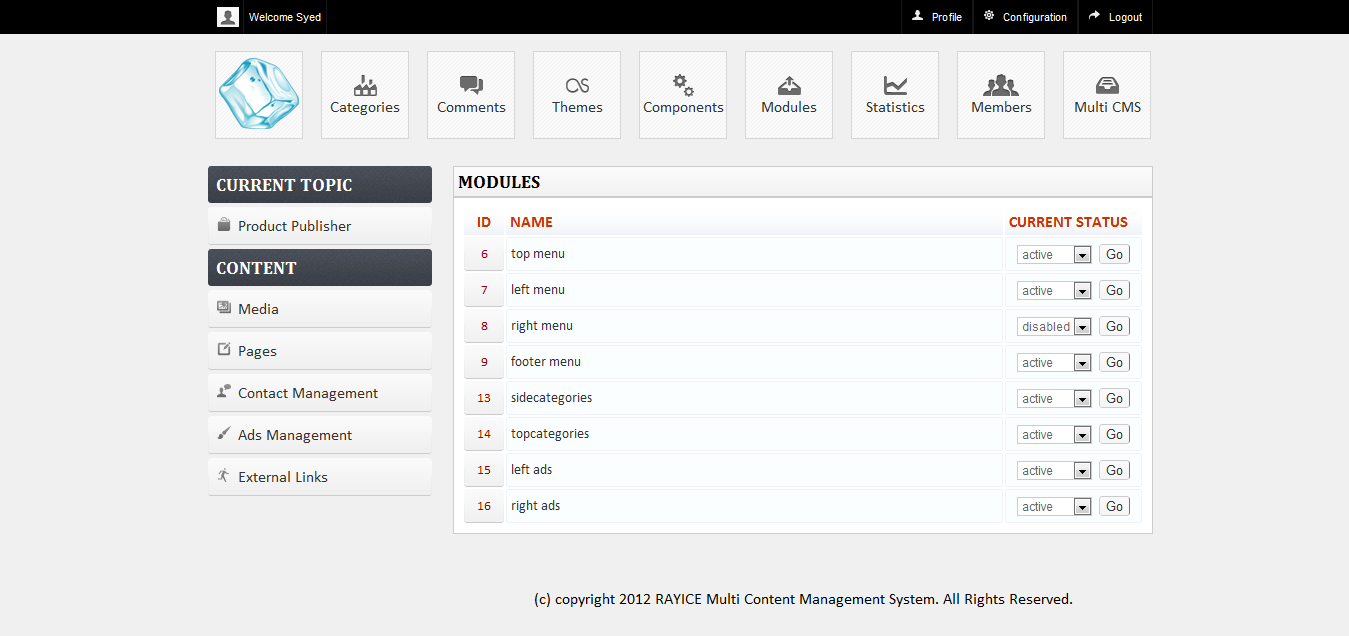


Figure 7. Modules

## WIDGETS

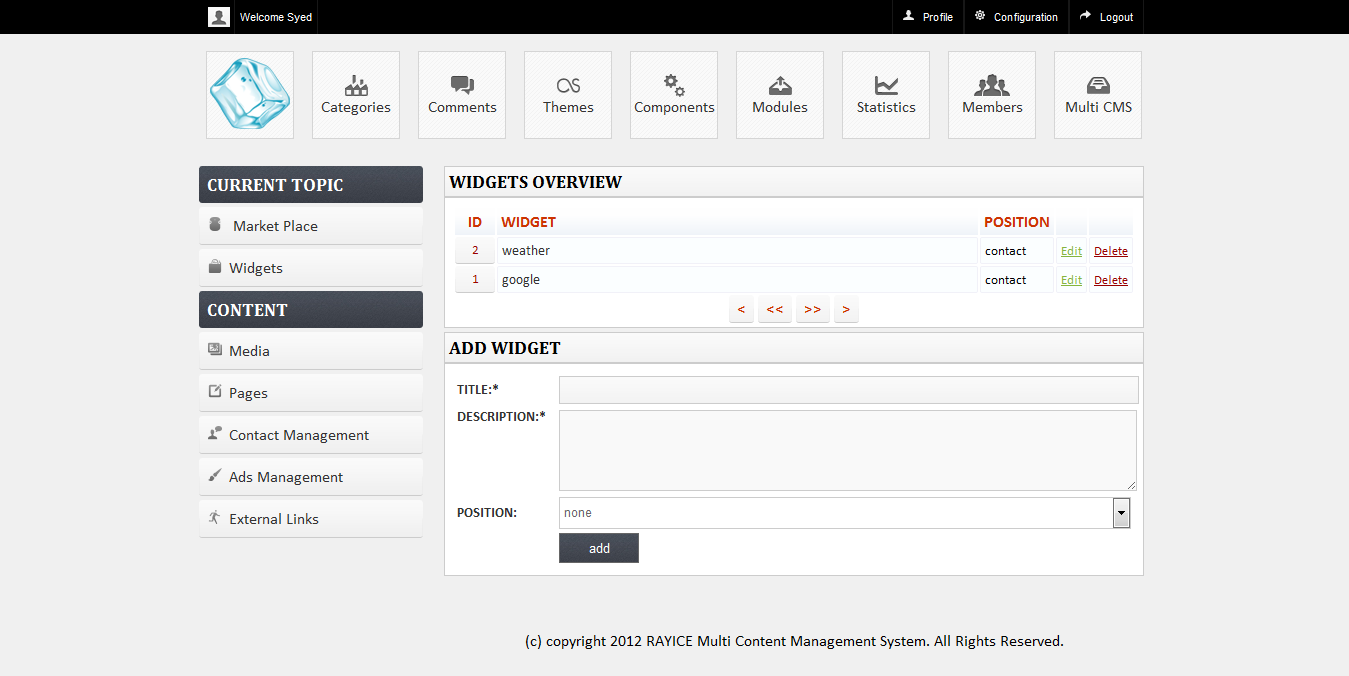


Figure 7. Widgets

## MEDIA

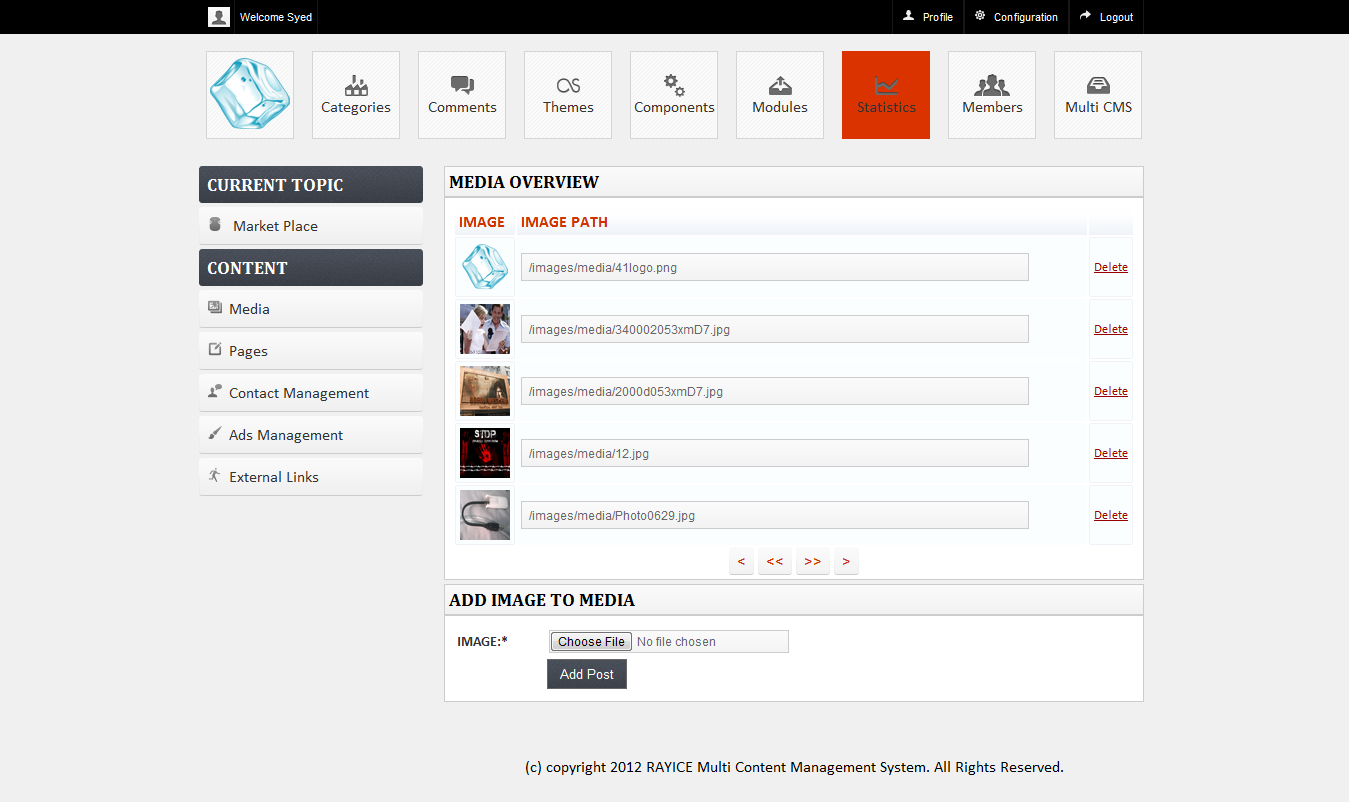


Figure 7. Media

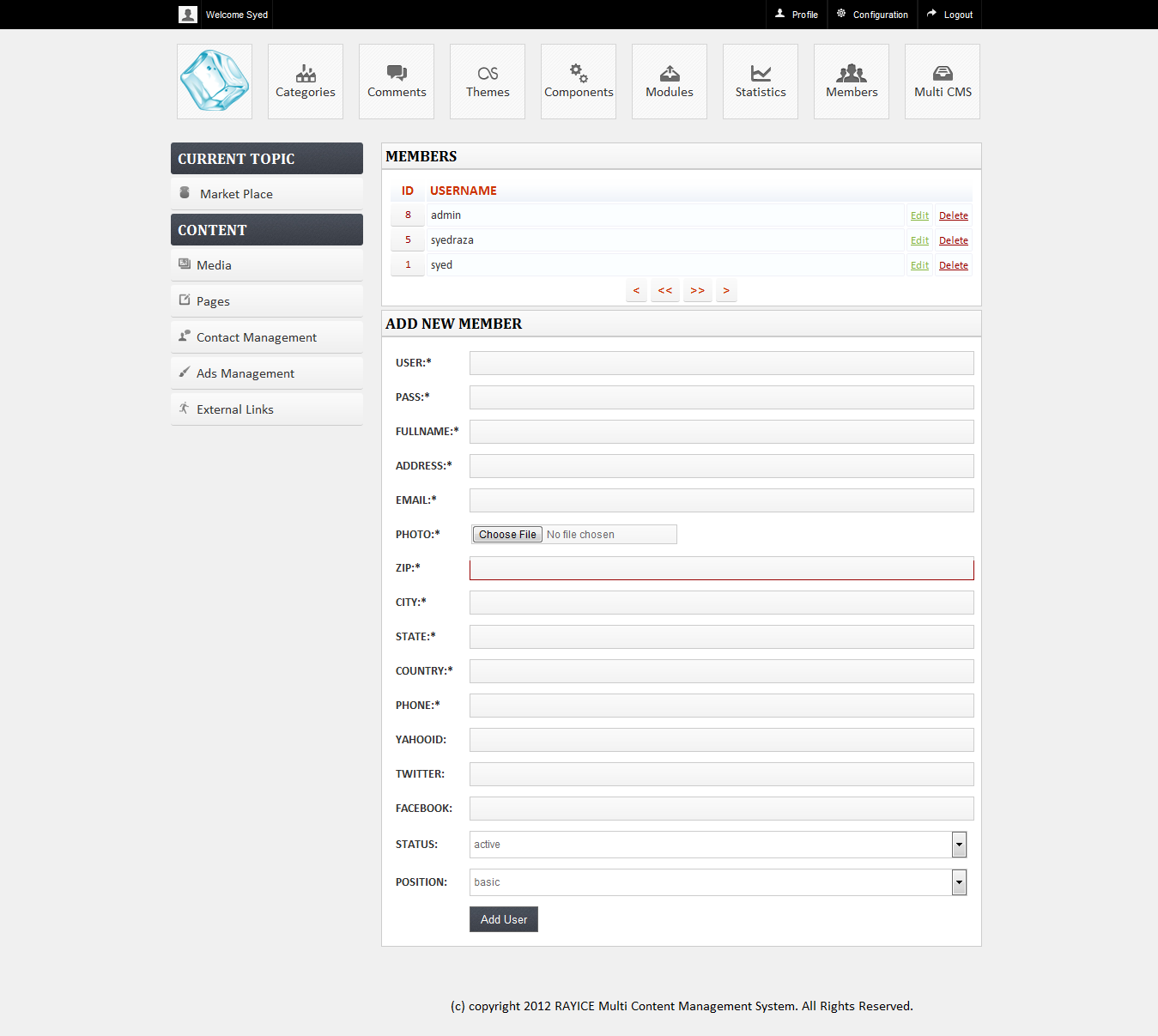
MEMBERS

Figure 7. Members

## STATISTICS

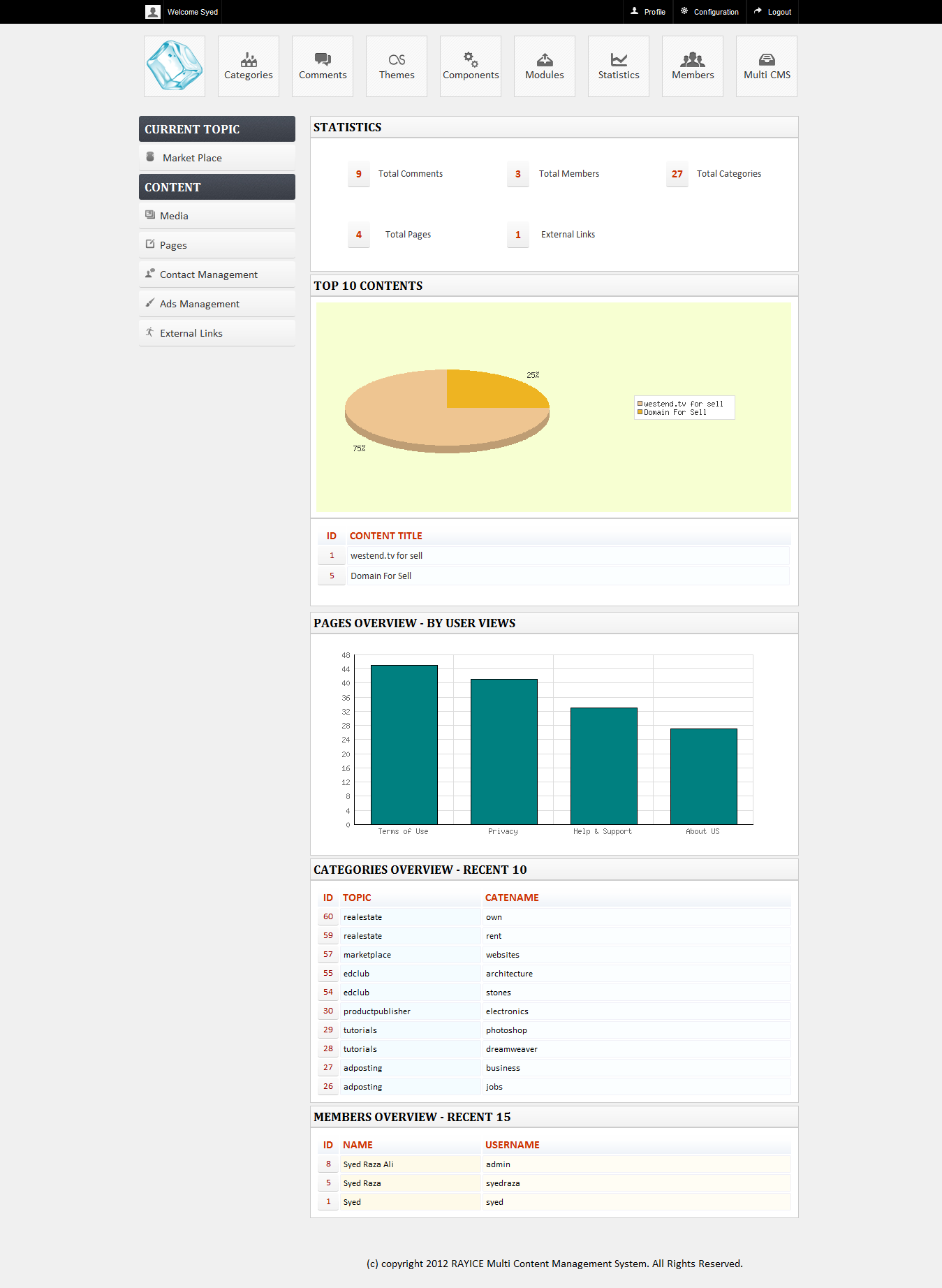


Figure 7. Statistics

PAGES

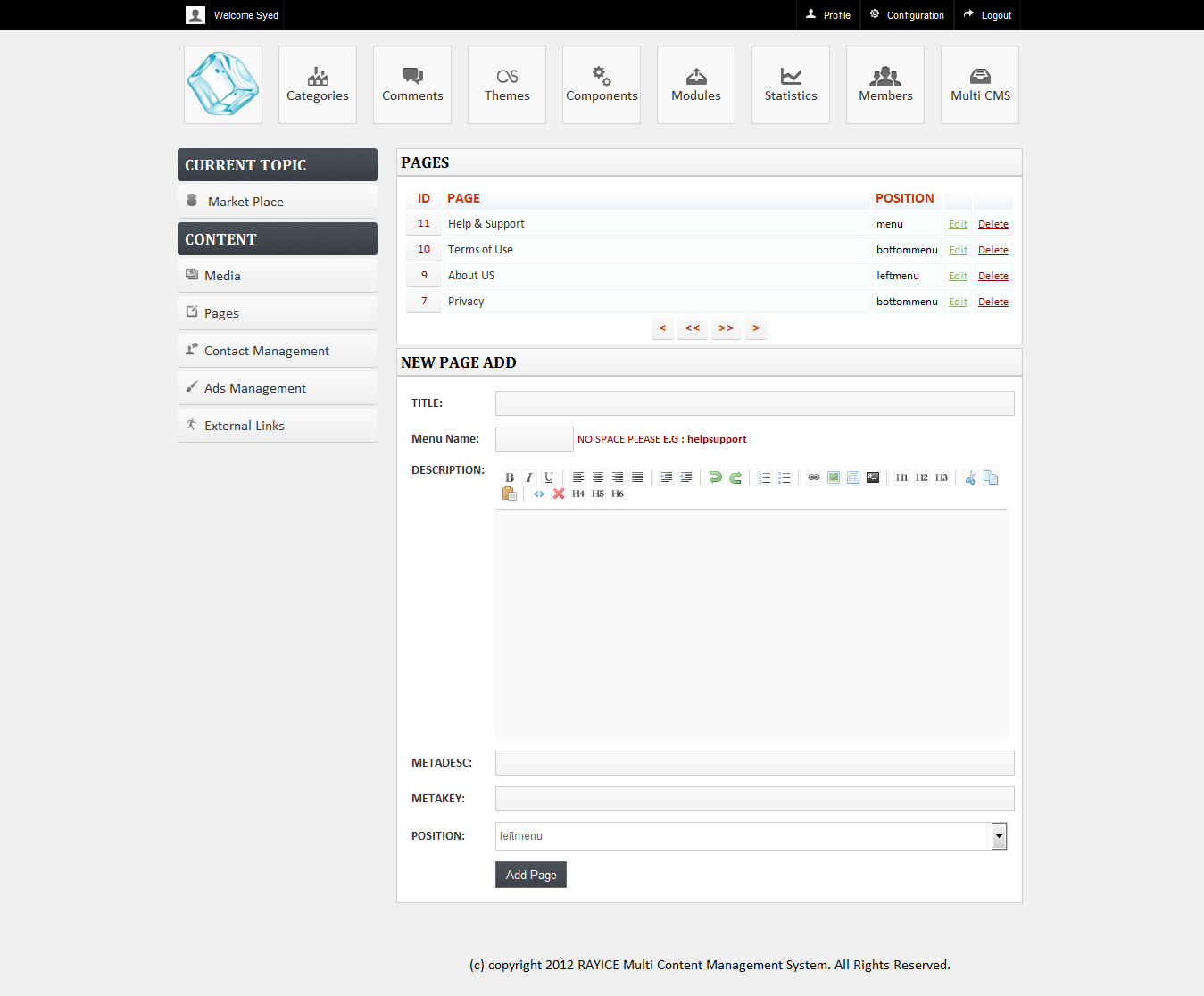


Figure 7. Pages

## MESSAGES

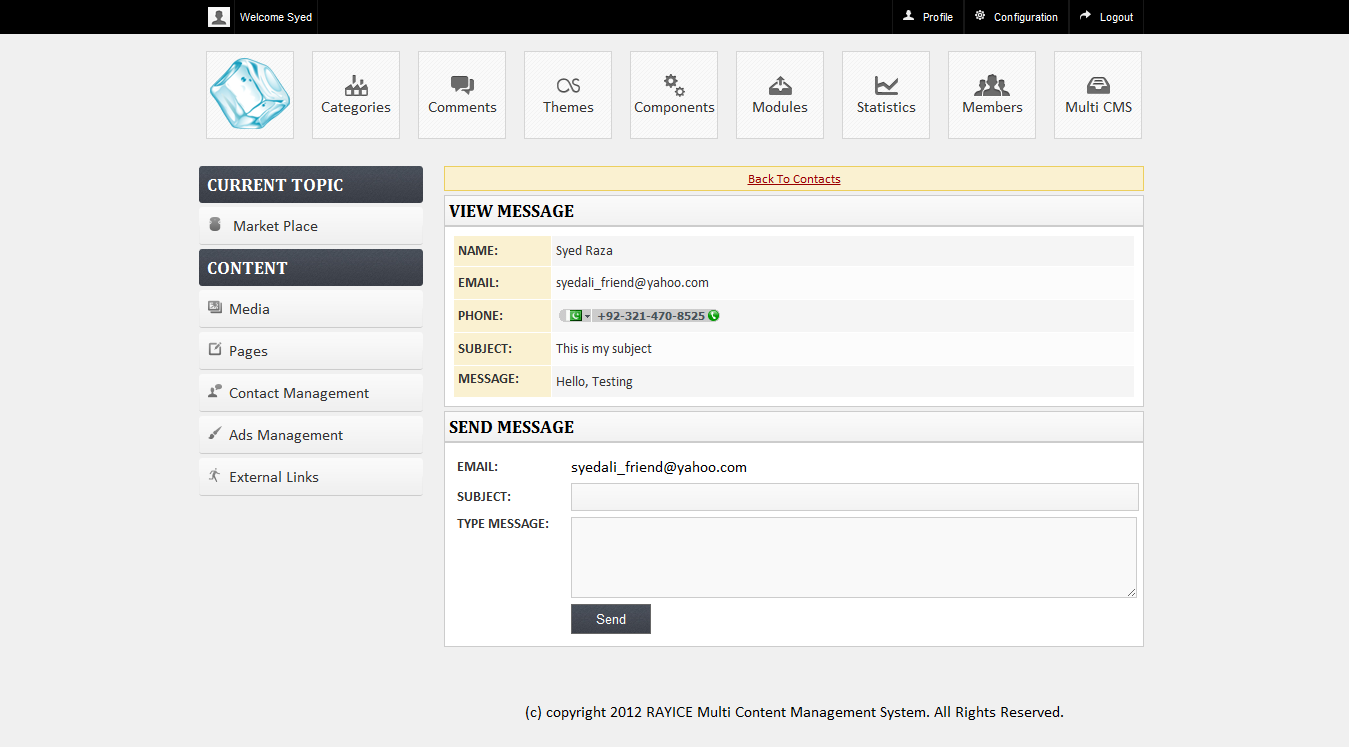


Figure 7. Messages

## EXTERNAL LINKS

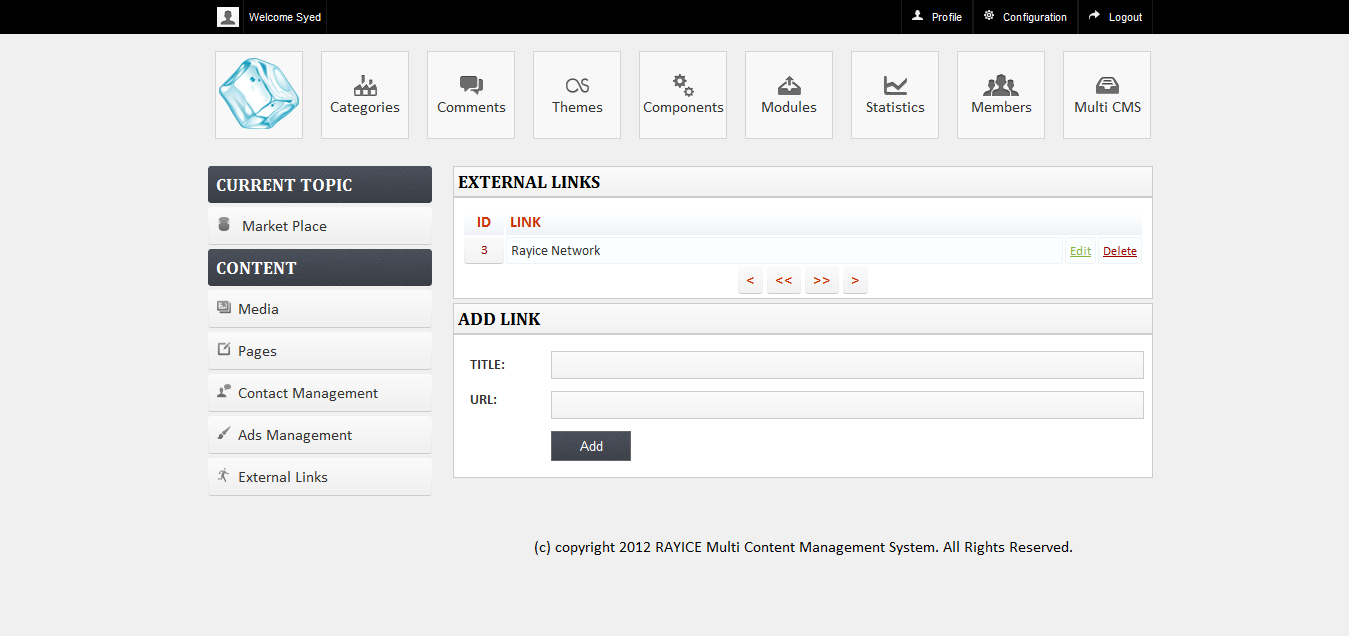


Figure 7. External Links

## MUTLI CMS CONTENT EDITING

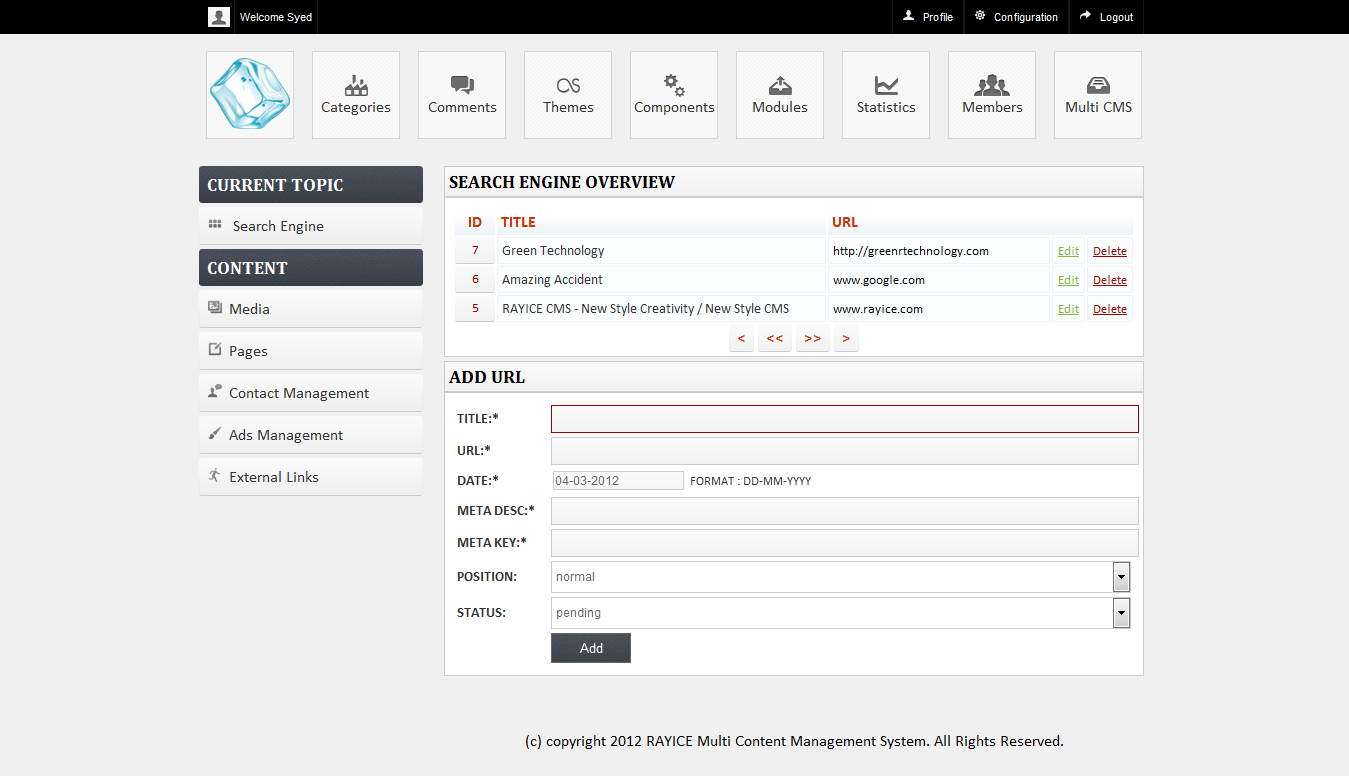


Figure 7. MultiCMS Content Editing

## ADS MANAGEMENT

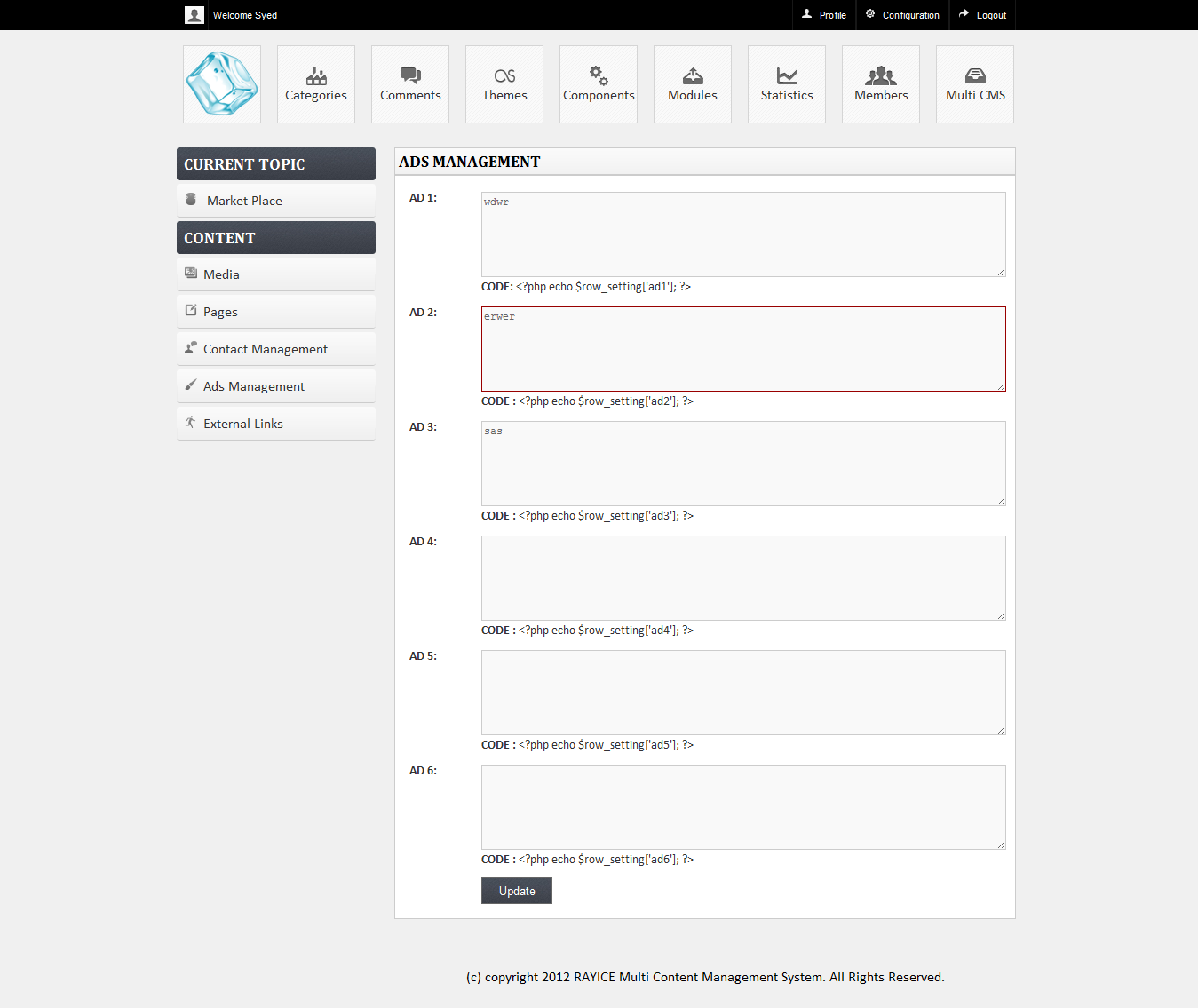


Figure 7. Ads Management

## MULTI CMS MARKETPLACE



Figure 7. MultiCMS MarketPlace Content Editing

## MULTI CMS PORTFOLIO/RESUME:

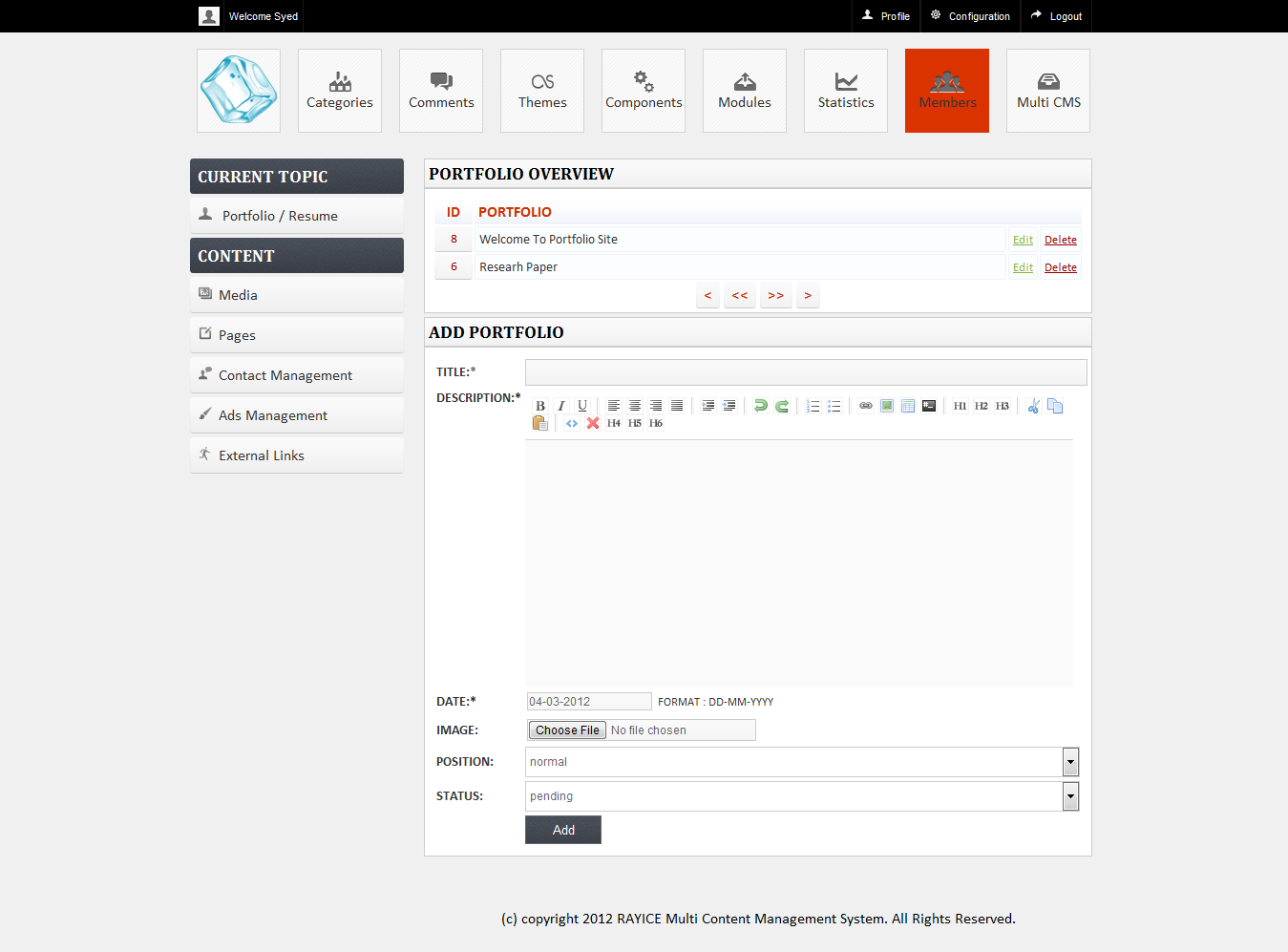


Figure 7. MultiCMS Portfolio Content Editing

## MULTI CMS TUTORIALS:

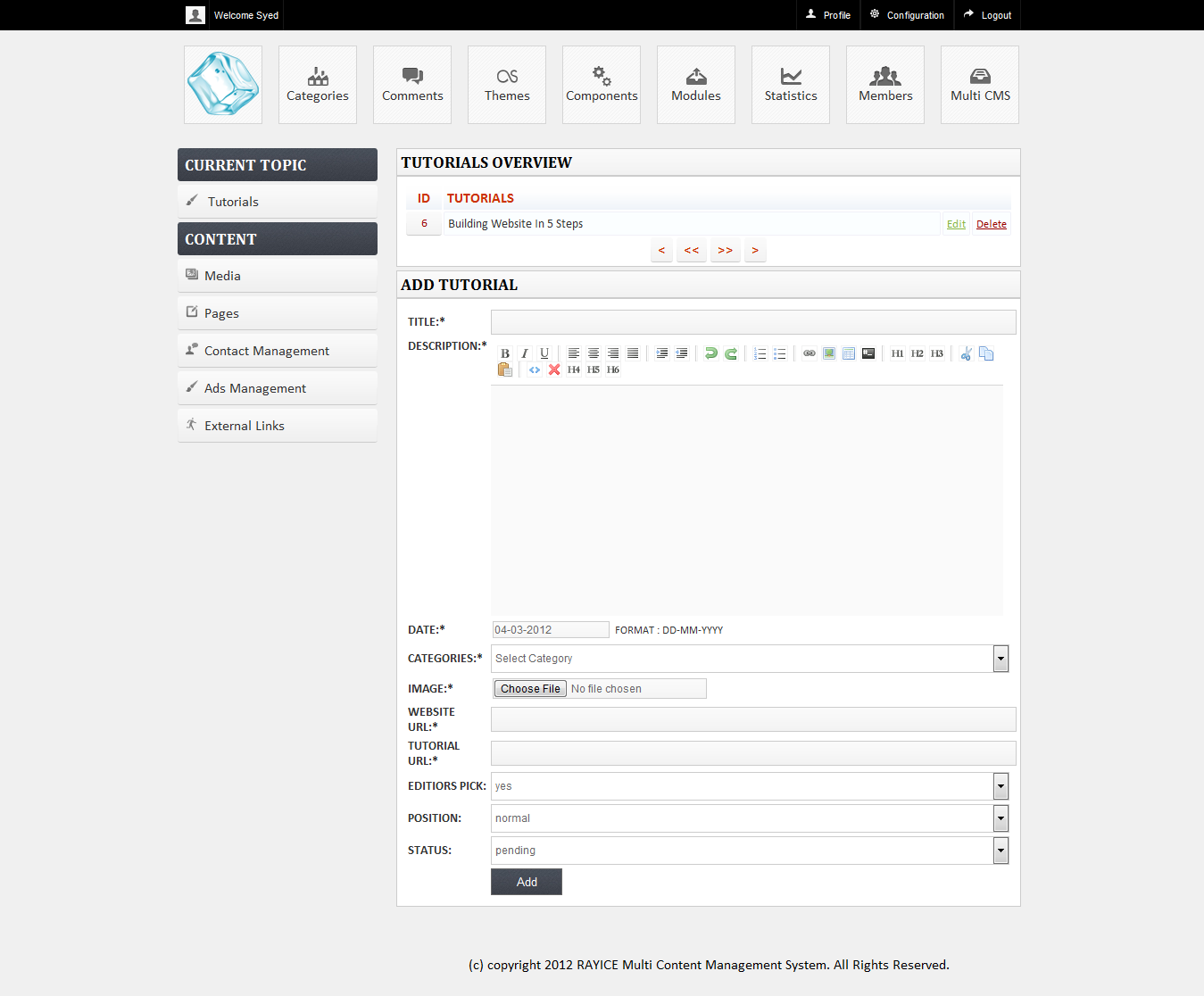


Figure 7. MultiCMS Tutorials Content Editing

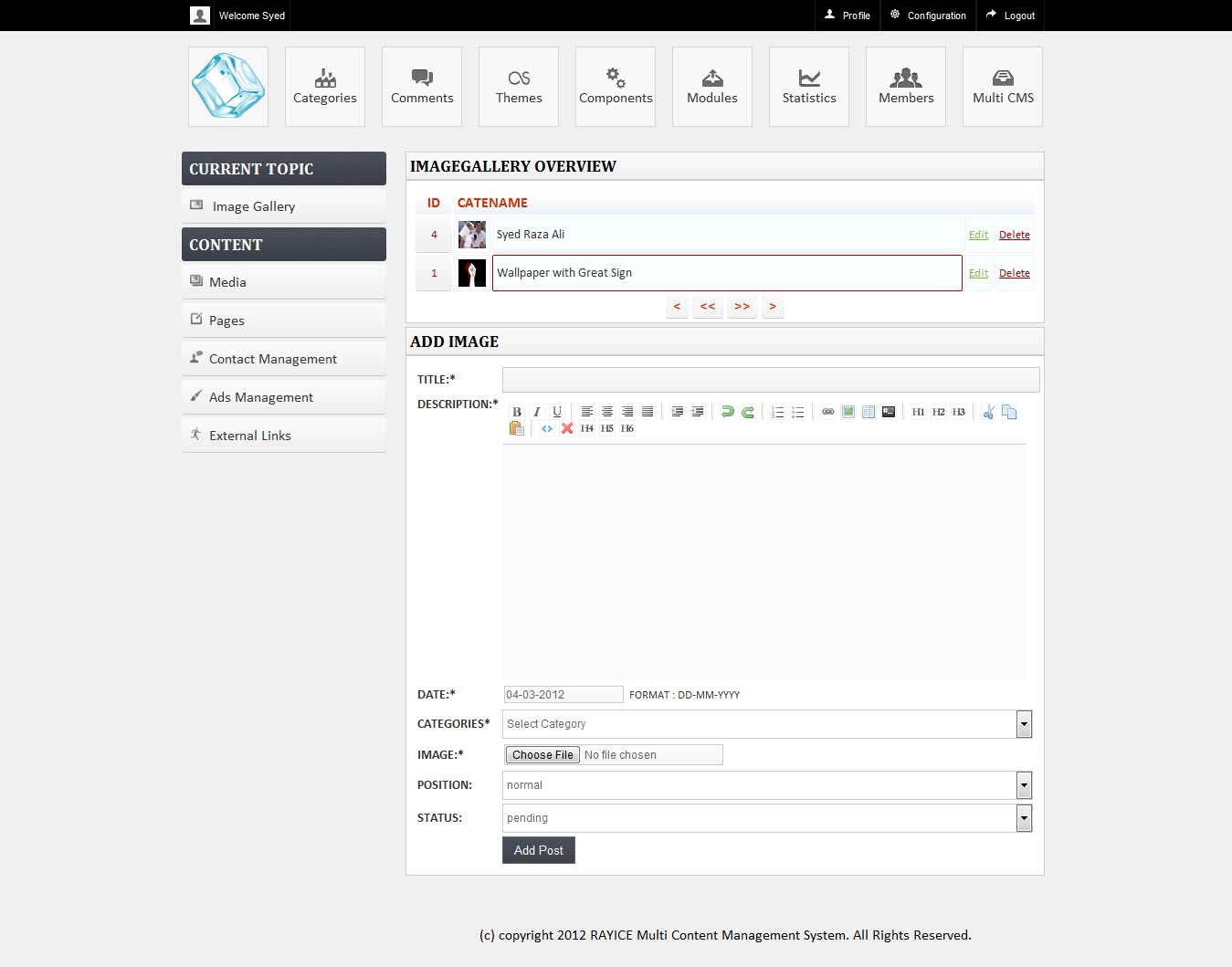
MULTI CMS IMAGE GALLERY: 

Figure 7. MultiCMS Image Gallery Content Editing

# APPENDIX (B) DATABASE TABLES

### SETTINGS TABLE

Table 7. 1 Database Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| settingid(AI) | Int | 100 | PRIMARY KEY |
| Datauser | Varchar | 100 |  |
| Datapass | Varchar | 100 |  |
| Level | Varchar | 100 |  |
| Title | Varchar | 100 |  |
| Siteurl | Varchar | 100 |  |
| Logo | Varchar | 100 |  |
| Favicon | Varchar | 100 |  |
| Selecttopic | Varchar | 100 |  |
| Installed | Varchar | 100 |  |
| Currency | Varchar | 100 |  |
| Theme | Varchar | 100 |  |
| Owner | Varchar | 100 |  |
| Missingimage | Varchar | 100 |  |
| Email | Varchar | 100 |  |
| Phone | Varchar | 100 |  |
| Metadesc | Varchar | 100 |  |
| Metakey | Varchar | 100 |  |
| onlinestatus | Varchar | 100 |  |
| footer | Text |  |  |
| ad1 | Text |  |  |
| ad2 | Text |  |  |
| ad3 | Text |  |  |
| ad4 | Text |  |  |
| ad5 | Text |  |  |
| ad6 | Text |  |  |

### THEMES TABLE

Table 7. 2 Themes Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| themeid(AI) | Int | 100 | PRIMARY KEY |
| theme | Varchar | 100 |  |
| status | Varchar | 100 |  |

### PAGES TABLE

Table 7. 3 Pages Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| pageid(AI) | Int | 100 | PRIMARY KEY |
| name | Varchar | 100 |  |
| Title | Varchar | 100 |  |
| Description | Text |  |  |
| Image | Varchar | 100 |  |
| Metadesc | Varhcar | 100 |  |
| Metakey | Varchar | 100 |  |
| Selecttopic | Varchar | 100 |  |
| Position | Varchar | 100 |  |
| Views | int | 100 |  |

### WIDGETS TABLE

Table 7. 4 Widgets Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| wid(AI) | Int | 100 | PRIMARY KEY |
| Widget | Varchar | 100 |  |
| Content | Text |  |  |
| Position | Varchar | 100 |  |
| Status | Varchar | 100 |  |

### COMMENTS TABLE

Table 7. 5 Comments Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| commentid(AI) | Int | 100 | PRIMARY KEY |
| Comment | Text |  |  |
| name | Varchar | 100 |  |
| email | Varchar | 100 |  |
| website | Varchar | 100 |  |
| status | Varchar | 100 |  |
| selecttopic | Varchar | 100 |  |
| id | Int | 100 |  |

### CATEGORIES TABLE

Table 7. 6 Categories Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| cateid(AI) | Int | 100 | PRIMARY KEY |
| catename | Varchar | 100 |  |
| selecttopic | Varchar | 100 |  |

### CONTACTS TABLE

Table 7. 7 Contacts Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| cid(AI) | Int | 100 | PRIMARY KEY |
| name | Varchar | 100 |  |
| email | Varchar | 100 |  |
| phone | Varchar | 100 |  |
| subject | Varchar | 100 |  |
| message | Text |  |  |

### FRIENDLINKS TABLE

Table 7. 8 FriendLinks Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| linkid(AI) | Int | 100 | PRIMARY KEY |
| Linktitle | Varchar | 100 |  |
| Linkurl | Varchar | 100 |  |

### MEDIA TABLE

Table 7. 9 Contacts Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| id(AI) | Int | 100 | PRIMARY KEY |
| Imageurl | Varchar | 100 |  |
| Selecttopic | Varchar | 100 |  |

### MEMBERS TABLE

Table 7. 10 Contacts Tables

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| memberid(AI) | Int | 100 | PRIMARY KEY |
| Users | Varchar | 100 |  |
| Passs | Varchar | 100 |  |
| Fullname | Varchar | 100 |  |
| Address | Varchar | 100 |  |
| Email | Varchar | 100 |  |
| Photo | Varchar | 100 |  |
| zip | Varchar | 100 |  |
| city | Varchar | 100 |  |
| state | Varchar | 100 |  |
| country | Varchar | 100 |  |
| phone | Varchar | 100 |  |
| yahooid | Varchar | 100 |  |
| twitter | Varchar | 100 |  |
| facebook | Varchar | 100 |  |
| status | Varchar | 100 |  |
| position | Varchar | 100 |  |
| selecttopic | Varchar | 100 |  |

### NEWS TABLE

Table 7. 11 News Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| newsid(AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| selecttopic | Varchar | 100 |  |
| views | Varchar | 100 |  |

### PARTS TABLE

Table 7. 12 Parts Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| partsid(AI) | Int | 100 | PRIMARY KEY |
| Part | Varchar | 100 |  |
| Status | Varchar | 100 |  |
| Type | Varchar | 100 |  |

### ADPOSTING TABLE

Table 7. 13 Adposting Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| adpostingid(AI) | Int | 100 | PRIMARY KEY |
| Title | Varchar | 100 |  |
| Description | Text |  |  |
| Imageurl | Varchar | 100 |  |
| Adpostingurl | Varchar | 100 |  |
| Owner | Varchar | 100 |  |
| Date | Varchar | 100 |  |
| Catename | Varchar | 100 |  |
| Status | Varchar | 100 |  |
| Rating | Int | 100 |  |
| Views | Int | 100 |  |
| position | Varchar | 100 |  |
| editorpick | Varchar | 100 |  |
| users | Varchar | 100 |  |

### BLOG TABLE

Table 7. 14 Blog Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| blogid(AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| photo | Varchar | 100 |  |
| dates | Varchar | 100 |  |
| metadesc | Varchar | 100 |  |
| metakey | Varchar | 100 |  |
| catename | Varchar | 100 |  |
| status | Varchar | 100 |  |
| position | Varchar | 100 |  |
| users | Varchar | 100 |  |
| rating | Int | 100 |  |
| views | Int | 100 |  |

### DOCTORS TABLE

Table 7. 15 Doctors Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| doctorsid(AI) | Int | 100 | PRIMARY KEY |
| Title | Varchar | 100 |  |
| Description | text |  |  |
| Metadesc | Varchar | 100 |  |
| Metakey | Varchar | 100 |  |
| Author | Varchar | 100 |  |
| Dates | Varchar | 100 |  |
| Image | Varchar | 100 |  |
| Status | Varchar | 100 |  |
| Position | Varchar | 100 |  |
| Users | Varchar | 100 |  |
| Rating | Int | 100 |  |
| Views | Int | 100 |  |
| Selecttopic | Varchar | 100 |  |

### IMAGE TABLE

Table 7. 16 Image Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| imageid(AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| imageurl | Varchar | 100 |  |
| date | Varchar | 100 |  |
| catename | Varchar | 100 |  |
| status | Varchar | 100 |  |
| rating | Int | 100 |  |
| views | Int | 100 |  |
| position | Varchar | 100 |  |
| users | Varchar | 100 |  |

### MARKET TABLE

Table 7. 17 Market Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| marketid(AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| imageurl | Varchar | 100 |  |
| owner | Varchar | 100 |  |
| date | Varchar | 100 |  |
| price | Varchar | 100 |  |
| Catename | Varchar | 100 |  |
| Status | Varchar | 100 |  |
| Rating | Int | 100 |  |
| Views | Int | 100 |  |
| Position | Varchar | 100 |  |
| Users | Varchar | 100 |  |
| Editorpick | Varchar | 100 |  |
| Current | Varchar | 100 |  |

### MARKETMESSAGE TABLE

Table 7. 18 MarketMessage Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| marketmessageid(AI) | Int | 100 | PRIMARY KEY |
| Title | Varchar | 100 |  |
| Description | Text |  |  |
| Users | Varchar | 100 |  |
| From | Varchar | 100 |  |
| status1 | Varchar | 100 |  |
| Status | Varchar | 100 |  |
| Marketed | Int | 100 |  |

### PORTFOLIO TABLE

Table 7. 19 Portfolio Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| portfolioid(AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| metadesc | Varchar | 100 |  |
| metakey | Varchar | 100 |  |
| author | Varchar | 100 |  |
| dates | Varchar | 100 |  |
| image | Varchar | 100 |  |
| status | Varchar | 100 |  |
| position | Varchar | 100 |  |
| users | Varchar | 100 |  |
| rating | Int | 100 |  |
| views | Int | 100 |  |
| selecttopic | Varchar | 100 |  |

### TUTORIALS TABLE

Table 7. 20 Tutorials Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| tutorialsid(AI) | Int | 100 | PRIMARY KEY |
| Title | Varchar | 100 |  |
| Description | Text |  |  |
| Imageurl | Varchar | 100 |  |
| Owner | Varchar | 100 |  |
| date | Varchar | 100 |  |
| catename | Varchar | 100 |  |
| tutorialsurl | Varchar | 100 |  |
| tutorialpathurl | Varchar | 100 |  |
| status | Varchar | 100 |  |
| rating | Int | 100 |  |
| views | Int | 100 |  |
| position | Varchar | 100 |  |
| users | Varchar | 100 |  |
| editorpick | Varchar | 100 |  |

### TUTORIALSMESSAGE TABLE

Table 7. 21 TutorialsMessage Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| tutorialsmessageid (AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| users | Varchar | 100 |  |
| from | Varchar | 100 |  |
| status1 | Varchar | 100 |  |
| status | Varchar | 100 |  |
| tutorialsid | Int | 100 |  |

### PRODUCTPUBLISHER TABLE

Table 7. 22 ProductPublisher Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| productpublisherid (AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| imageurl | Varchar | 100 |  |
| productpublisherurl | Varchar | 100 |  |
| owner | Varchar | 100 |  |
| date | Varchar | 100 |  |
| catename | Varchar | 100 |  |
| status | Varchar | 100 |  |
| rating | Int | 100 |  |
| views | Int | 100 |  |
| position | Varchar | 100 |  |
| users | Varchar | 100 |  |
| editorpick | Varchar | 100 |  |
| current | Varchar | 100 |  |

### SEARCHENGINE TABLE

Table 7. 23 SearchEngine Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| searchengineid(AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| siteurl | Varchar | 100 |  |
| dates | Varchar | 100 |  |
| metadesc | Varchar | 100 |  |
| metakey | Varchar | 100 |  |
| status | Varchar | 100 |  |
| position | Varchar | 100 |  |
| views | Int | 100 |  |

### VIDEO TABLE

Table 7. Video Table

|  |  |  |  |
| --- | --- | --- | --- |
| COLUMN | TYPE | LENGTH/VALUES | INDEX |
| videoid(AI) | Int | 100 | PRIMARY KEY |
| title | Varchar | 100 |  |
| description | Text |  |  |
| imageurl | Varchar | 100 |  |
| embedcode | Text | 100 |  |
| date | Varchar | 100 |  |
| videotype | Varchar |  |  |
| catename | Varchar | 100 |  |
| status | Varchar | 100 |  |
| rating | Int | 100 |  |
| views | Int | 100 |  |
| position | Varchar | 100 |  |
| users | Varchar | 100 |  |

# APPENDIX (C) ENTITIES

## SETTINGS TABLE

**SETTINGS**

## THEMES TABLE

**THEMES**

## PAGES TABLE

**PAGES**

## WIDGETS TABLE

**WIDGETS**

## COMMENTS TABLE

**COMMENTS**

## CATEGORIES TABLE

**CATEGORIES**

## CONTACTS TABLE

**CONTACTS**

## FRIENDLINKS TABLE

**FRIENDLINKS**

## MEDIA TABLE

**MEDIA**

## MEMBERS TABLE

**MEMBERS**

## NEWS TABLE

**NEWS**

## PARTS TABLE

**PARTS**

## ADPOSTING TABLE

**ADPOSTING**

## BLOG TABLE

**BLOG**

DOCTORS TABLE

**DOCTORS**

IMAGE TABLE

**IMAGE**

MARKET TABLE

**MARKET**

MARKETMESSAGE TABLE

**MARKET  
MESSAGE**

PORTFOLIO TABLE

**PORTFOLIO**

**TUTORIALS TABLE:**

**TUTORIALS**

TUTORIALS TABLE

**TUTORIALS**

PRODUCTPUBLISHER TABLE

**PRODUCT  
PULISHER**

SEARCHENGINE TABLE

**SEARCH ENGINE**

VIDEO TABLE

**VIDEO**

# APPENDIX (D) DATABASE QUEIRIES

### TO CREATE DATABASES

Create database rayicecms;

### TO CREATE TABLES

Below are some database queries to create tables for **rayicecms** database

* **ADPOSTING:**CREATE TABLE IF NOT EXISTS `adposting` ( `adpostingid` int(100) NOT NULL AUTO\_INCREMENT, `title` varchar(100) NOT NULL, `description` text NOT NULL, `imageurl` varchar(100) NOT NULL, `adpostingurl` varchar(100) DEFAULT NULL, `owner` varchar(100) DEFAULT NULL, `date` varchar(100) NOT NULL, `catename` varchar(100) NOT NULL, `status` varchar(100) NOT NULL, `rating` int(100) DEFAULT NULL, `views` int(100) DEFAULT NULL, `position` varchar(100) NOT NULL, `editorpick` varchar(100) DEFAULT NULL, `users` varchar(100) NOT NULL, PRIMARY KEY (`adpostingid`));
* **CATEGORIES:**

CREATE TABLE IF NOT EXISTS `categories` ( `cateid` int(100) NOT NULL AUTO\_INCREMENT, `catename` varchar(100) NOT NULL, `selecttopic` varchar(100) DEFAULT NULL, PRIMARY KEY (`cateid`));

* **COMMENTS:**

CREATE TABLE IF NOT EXISTS `comments` ( `commentid` int(100) NOT NULL AUTO\_INCREMENT, `comment` text, `name` varchar(100) DEFAULT NULL, `email` varchar(100) DEFAULT NULL, `website` varchar(100) DEFAULT NULL, `status` varchar(100) DEFAULT NULL, `selecttopic` varchar(100) DEFAULT NULL, `id` int(100) DEFAULT NULL, PRIMARY KEY (`commentid`));

* **CONTACTS:**

CREATE TABLE IF NOT EXISTS `contacts` ( `cid` int(100) NOT NULL AUTO\_INCREMENT, `name` varchar(100) NOT NULL, `email` varchar(100) NOT NULL, `phone` varchar(100) NOT NULL, `subject` varchar(100) NOT NULL, `message` text NOT NULL, PRIMARY KEY (`cid`));

* **FRIENDLINKS:**

CREATE TABLE IF NOT EXISTS `friendlinks` (`linkid` int(100) NOT NULL AUTO\_INCREMENT, `linktitle` varchar(100) NOT NULL, `linkurl` varchar(100) NOT NULL, PRIMARY KEY (`linkid`)) ;

* **MARKET:**

CREATE TABLE IF NOT EXISTS `market` ( `marketid` int(100) NOT NULL AUTO\_INCREMENT, `title` varchar(100) NOT NULL, `description` text NOT NULL, `imageurl` varchar(100) NOT NULL, `imageurl1` varchar(100) DEFAULT NULL, `imageurl2` varchar(100) DEFAULT NULL, `owner` varchar(100) DEFAULT NULL, `date` varchar(100) NOT NULL, `price` varchar(100) NOT NULL, `catename` varchar(100) NOT NULL, `status` varchar(100) NOT NULL, `rating` int(100) DEFAULT NULL, `views` int(100) DEFAULT NULL, `position` varchar(100) NOT NULL, `users` varchar(100) NOT NULL, `editorpick` varchar(100) NOT NULL, `current` varchar(100) NOT NULL, PRIMARY KEY (`marketid`));

* **MEDIA:**

CREATE TABLE IF NOT EXISTS `media` ( `id` int(100) NOT NULL AUTO\_INCREMENT, `imageurl` varchar(100) NOT NULL, PRIMARY KEY (`id`));

* **MEMBERS:**

CREATE TABLE IF NOT EXISTS `members` ( `memberid` int(100) NOT NULL AUTO\_INCREMENT, `users` varchar(100) NOT NULL, `passs` varchar(100) NOT NULL, `fullname` varchar(100) NOT NULL, `address` varchar(100) NOT NULL, `email` varchar(100) NOT NULL, `photo` varchar(100) DEFAULT NULL, `zip` varchar(100) NOT NULL, `city` varchar(100) NOT NULL, `state` varchar(100) NOT NULL, `country` varchar(100) NOT NULL, `phone` varchar(100) NOT NULL, `yahooid` varchar(100) DEFAULT NULL, `twitter` varchar(100) DEFAULT NULL, `facebook` varchar(100) DEFAULT NULL, `status` varchar(100) NOT NULL, `position` varchar(100) NOT NULL, `selecttopic` varchar(100) DEFAULT NULL, PRIMARY KEY (`memberid`));

* **PAGES:**

CREATE TABLE IF NOT EXISTS `pages` ( `pageid` int(100) NOT NULL AUTO\_INCREMENT, `name` archar(100) NOT NULL, `title` varchar(100) NOT NULL, `description` text NOT NULL, `image` varchar(100) DEFAULT NULL, `metadesc` varchar(100) DEFAULT NULL, `metakey` varchar(100) DEFAULT NULL, `selecttopic` varchar(100) NOT NULL, `position` varchar(100) NOT NULL, `views` int(100) DEFAULT NULL, PRIMARY KEY (`pageid`));

* **PARTS:**

CREATE TABLE IF NOT EXISTS `parts` ( `partsid` int(100) NOT NULL AUTO\_INCREMENT, `part` varchar(100) NOT NULL, `status` varchar(100) NOT NULL, `type` varchar(100) DEFAULT NULL, PRIMARY KEY (`partsid`));

* **PORTFOLIO:**

CREATE TABLE IF NOT EXISTS `portfolio` ( `portfolioid` int(100) NOT NULL AUTO\_INCREMENT, `title` varchar(100) NOT NULL, `description` text NOT NULL, `metadesc` varchar(160) DEFAULT NULL, `metakey` varchar(250) DEFAULT NULL, `author` varchar(100) DEFAULT NULL, `dates` varchar(100) NOT NULL, `image` varchar(100) DEFAULT NULL, `status` varchar(100) NOT NULL, `position` varchar(100) NOT NULL, `users` varchar(100) NOT NULL, `rating` int(100) NOT NULL, `views` int(100) NOT NULL, `selecttopic` varchar(100) DEFAULT NULL, PRIMARY KEY (`portfolioid`));

* **THEMES:**

CREATE TABLE IF NOT EXISTS `themes` (`themeid` int(100) NOT NULL AUTO\_INCREMENT, `theme` varchar(100) NOT NULL, `status` varchar(100) NOT NULL, PRIMARY KEY (`themeid`));

* **WIDGETS:**

CREATE TABLE IF NOT EXISTS `widgets` ( `wid` int(100) NOT NULL AUTO\_INCREMENT, `widget` varchar(100) NOT NULL, `content` text NOT NULL, `position` varchar(100) DEFAULT NULL, `status` varchar(100) NOT NULL, PRIMARY KEY (`wid`));

### TO SELECT RECORD

SELECT \* FROM `categories` LIMIT 0, 30;   
SELECT \* from themes;  
SELECT \* from pages WHERE pageid = ’3’;  
SELECT \* FROM `themes` LIMIT 0 , 30;

### TO INSERT RECORD

* **ADPOSTING:**

INSERT INTO `adposting` (`adpostingid`, `title`, `description`, `imageurl`, `adpostingurl`, `owner`, `date`, `catename`, `status`, `rating`, `views`, `position`, `editorpick`, `users`) VALUES(1, 'My Mobile For Sale', 'I want to sell my mobile, the price for my mobile is 1500, my mobile set is Nokia 500, Contact me on 00923214708525.', 'mobile.jpg', 'http://www.adposting.com', 'syed', '12/12/2012', 'electronics', 'published', 0, 15, 'featured', NULL, 'razashah');

* **CATEGORIES:**

INSERT INTO `categories` (`cateid`, `catename`, `selecttopic`) VALUES

(2, 'technology', 'blog'),

(3, 'investment', 'blog'),

(4, 'news', 'blog'),

(7, 'internet', 'blog'),

(8, 'funny', 'videostream'),

(9, 'accident', 'videostream'),

(10, 'entertainment', 'videostream'),

(12, 'animation', 'videostream'),

(13, '3d wallpapers', 'imagegallery'),

(19, 'electronics', 'marketplace');

* **COMMENTS:**INSERT INTO `comments` (`commentid`, `comment`, `name`, `email`, `website`, `status`, `selecttopic`, `id`) VALUES(2, 'that is good post;, 'ali', 'falconali2004@hotmail.com', 'http://rayice.com/', 'published', 'edclub', 1);
* **FRIENDLINKS:**

INSERT INTO `friendlinks` (`linkid`, `linktitle`, `linkurl`) VALUES(3, 'Rayice Company’, 'http://www.rayice.com');

* **MEDIA:**

INSERT INTO `media` (`id`, `imageurl`) VALUES(1, 'Photo0629.jpg');

* **MEMBERS:**

INSERT INTO `members` (`memberid`, `users`, `passs`, `fullname`, `address`, `email`, `photo`, `zip`, `city`, `state`, `country`, `phone`, `yahooid`, `twitter`, `facebook`, `status`, `position`, `selecttopic`) VALUES(1, 'syed', '123456', 'Syed Raza Ali', 'mian mohamad town', 'ask@rayice.com', '1000b053xmD7.jpg', '10250', 'mirpur', 'azad kashmir', 'pakistan', '923214708525', '', '', '', 'active', 'gold', NULL);

* **THEMES:**

INSERT INTO `themes` (`themeid`, `theme`, `status`) VALUES

(1, 'default', 'active'),

(22, 'html5', 'disabled');

### TO DELETE RECORD

* DELETE TABLE `rayicecms`.`categories`;
* DELETE FROM `rayicecms`.`categories` WHERE `categories`.`cateid` = 8;
* DELETE FROM `rayicecms`.`members` WHERE `members`.`membersid` = 2;
* DELETE \* FROM `rayicecms`.`themes`;

### TO UPDATE RECORD

* UPDATE `rayicecms`.`settings` SET `ad3` = 'sas' WHERE `settings`.`settingid` =1;
* UPDATE `rayicecms`.`pages` SET `name` = 'privacy',`description` = 'Privacy Data Changed',`metadesc` = 'All about privacy of Multi CMS',`metakey` = 'privacy , detail, page' WHERE `pages`.`pageid` =7;
* UPDATE `rayicecms`.`parts` SET `status` = 'disabled' WHERE `parts`.`partsid` =1;

UPDATE `rayicecms`.`widgets` SET `content` = 'Add Google Widget To Multi CMS' WHERE `widgets`.`wid` =1;

# APPENDIX (E) PHP CODES

### SETTING

<!--Setting Table -->

mysql\_select\_db($database\_rayicecms, $rayicecms);  
**$query\_setting** = "SELECT \* FROM settings";  
**$setting** = mysql\_query($query\_setting, $rayicecms) or die(mysql\_error());  
**$row\_setting** = mysql\_fetch\_assoc($setting);  
**$totalRows\_setting** = mysql\_num\_rows($setting);

### MEMBERS

Below is a php query code fetching record from members table

<!--Members Table-->

mysql\_select\_db($database\_rayicecms, $rayicecms);  
**$query\_totalusers** = "SELECT \* FROM members";  
**$totalusers** = mysql\_query($query\_totalusers, $rayicecms) or die(mysql\_error());  
**$row\_totalusers** = mysql\_fetch\_assoc($totalusers);  
**$totalRows\_totalusers** = mysql\_num\_rows($totalusers);

### COMPONENTS

Below is a php query code fetching record from components table

<!-- Parts Table -->

mysql\_select\_db($database\_rayicecms, $rayicecms);  
**$query\_parts** = "SELECT \* FROM parts WHERE type = 'component'";  
**$parts** = mysql\_query($query\_parts, $rayicecms) or die(mysql\_error());  
**$row\_parts** = mysql\_fetch\_assoc($parts);  
**$totalRows\_parts** = mysql\_num\_rows($parts);

### USER SESSION CODE

Below is the code that is starting the session of the login user

<!--Component Table-->

if (!isset($\_SESSION)) {  
session\_start();  
}

### PAGES

Below is a php query code fetching record from pages table

<!--PAGESLIST on Left Menu -->

mysql\_select\_db($database\_rayicecms, $rayicecms);  
**$query\_pages** = "SELECT \* FROM pages WHERE `position` = 'leftmenu'";  
**$pages** = mysql\_query($query\_pages, $rayicecms) or die(mysql\_error());  
**$row\_pages** = mysql\_fetch\_assoc($pages);  
**$totalRows\_pages** = mysql\_num\_rows($pages);

### THEMES

Below is a php query code fetching record from setting table

<!-- Themes -->

mysql\_select\_db($database\_rayicecms, $rayicecms);  
**$query\_theme** = "SELECT \* FROM themes";  
**$theme** = mysql\_query($query\_theme, $rayicecms) or die(mysql\_error());  
**$row\_theme** = mysql\_fetch\_assoc($theme);  
**$totalRows\_theme** = mysql\_num\_rows($theme);

### IMPORTANT CODES

<?php include("../configuration.php"); ?>

<!-- code start -->

<?php  
if(($row\_setting['installed'] == "yes") && ($row\_setting['selecttopic'] == "portfolio"))  
{  
?>

<!-- index start -->

<?php include($theme\_path."".$row\_setting['theme']."/topdocs.php"); ?>

<!--Status Check -->

<?php

if($row\_setting['onlinestatus'] == "yes")

{

?>

<?php

}

else

{

?>

<?php header( 'Location: /install.php' ) ; ?>

<?php

}

?>

# APPENDIX (F) REFERENCES

**[1]** Joomla System, “http://joomla.org”, [5/march/2012]

**[2]** Wordpress System, "Wordpress.com”, [5/march/2012]

**[3]** CMS and A Single Web Designer, "http://www.alistapart.com/articles/cms1/", [5/march/2012]

**[4]** Wordpress Reviews, "http://webmasterformat.com/tools/cms/wordpress", [5/march/2012]

**[5]** Drupal Reviews,"http://webmasterformat.com/tools/cms/drupal", [5/march/2012]

**[6]** Joomla Reviews, "http://webmasterformat.com/tools/cms/joomla", [5/march/2012]

**[7]** Evolutionary Development, "http://www.compaid.com/caiInternet/ezine/FIRM-FromWaterfall2Evo.pdf”, [5/march/2012]

**[8]** Rayice,Co, S.R Ali, “http://www.rayice.com”, [5/march/2012]

**[9]** Exhedra Solutions, Inc, “http://www.vworker.com/RentACoder/DotNet/SoftwareCoders/showBioInfo.aspx?lngAuthorId=1597908”, [5/march/2012]

**[10]** Odesk Corporation, “https://www.odesk.com/users/Rayice-Web-Designer-Developer-PHP-MYSQL-CMS-Joomla-SEO\_~~d65355477d39385c”, [5/march/2012]

**[11]** Wikipedia Organization “http://en.wikipedia.org/wiki/Functional\_testing”, [22/4/2012]

**[12]** Evolutionary Process “http://www.ricklapenna.net/choosing-development-models-methodologies/”, [21/4/2012]

**[13]** Evolutionary Prototyping “http://en.wikipedia.org/wiki/Software\_prototyping”, [14/4/2012]

**[14]** Multi Content Management System, “http://www.blogscompany.com/”, [12/april/2012]