WEB PORTAL FOR THE CENTER FOR BIOPHARMACEUTICAL AND
BIODEVICE DEVELOPMENT WITH BLACKBOARD FEATURES

A Thesis
Presented to the
Faculty of
San Diego State University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
in
Computer Science

by
Venkata Sai Bharath Chadarajupalli
Fall 2011
SAN DIEGO STATE UNIVERSITY

The Undersigned Faculty Committee Approves the

Thesis of Venkata Sai Bharath Chadarajupalli:

Web Portal for the Center for Biopharmaceutical and Biodevice Development

with Blackboard Features

Carl Eckberg, Chair
Department of Computer Science

Joseph Lewis
Department of Computer Science

Dale Sevier
Center for Bio/Pharmaceutical and Biodevice Development

Oct 20, 2011
Approval Date
DEDICATION

This thesis is dedicated to my parents and to my sister for their unconditional love, support and encouragement throughout my graduate study.
ABSTRACT OF THE THESIS

Web Portal for the Center for Biopharmaceutical and Biodevice Development with Blackboard Features
by
Venkata Sai Bharath Chadarajupalli
Master of Science in Computer Science
San Diego State University, 2011

The Center for Biopharmaceutical and Biodevice Development (CBBD) provides education and training that augment the professional excellence and career opportunities of professionals in the pharmaceutical, biotechnology and those with the impetus to make it large. CBBD envisages the pinnacle of excellence in providing world-class accredited academic and practical education in various areas of the biomedical industry.

The CBBD website aims at providing a perfect interface to those aspiring individuals in the biomedical field by putting up all the necessary information at one place thereby making it the one-stop website for a biomedical striver to be a pioneer in the industry. The website is designed using a Content Management System (CMS) and provides easy accessibility for the students and instructors alike. The CMS facilitates effective communication between the faculty members and the students. It truly reflects the goals and the vision of the department.
# TABLE OF CONTENTS

PAGE

| ABSTRACT ............................................................................................................................... | v |
| LIST OF TABLES ................................................................................................................. | vii |
| LIST OF FIGURES ................................................................................................................. | ix |
| ACKNOWLEDGEMENTS ..................................................................................................... | xi |
| CHAPTER |
| 1 INTRODUCTION ........................................................................................................ | 1 |
| 1.1 Motivation ............................................................................................................... | 1 |
| 1.2 Background on Drupal Content Management System ............................................ | 2 |
| 1.3 CBBD Department .................................................................................................... | 3 |
| 1.4 Compression of CBBD, SDSU Blackboard and ClassTA ............................................ | 3 |
| 2 TECHNOLOGIES ......................................................................................................... | 5 |
| 2.1 PHP ...................................................................................................................... | 5 |
| 2.2 MYSQL .................................................................................................................. | 6 |
| 3 INSTALLATION .......................................................................................................... | 7 |
| 3.1 Basic Installation .................................................................................................... | 7 |
| 3.2 Create the CBBD Database .................................................................................... | 8 |
| 3.3 Logging into CBBD Site Admin Area ....................................................................... | 9 |
| 4 APPLICATION OVERVIEW ..................................................................................... | 13 |
| 4.1 High Level View ..................................................................................................... | 13 |
| 4.2 Implementation ........................................................................................................ | 14 |
| 4.2.1 Create Account ............................................................................................... | 14 |
| 4.2.2 Create a Course ............................................................................................... | 14 |
| 4.3 Content Management ............................................................................................ | 15 |
| 4.3.1 Create a New Content ................................................................................... | 16 |
| 4.3.2 Mega Menus .................................................................................................... | 16 |
| 4.3.3 Menus .............................................................................................................. | 19 |
| 4.3.4 Comments ...................................................................................................... | 20 |
4.3.5 Content Search ................................................................. 23
4.3.6 Working with Content Types and Fields .......................... 23
4.3.7 Forum/Discussion Board .................................................. 26
  4.3.7.1 Create a Forum .......................................................... 26
  4.3.7.2 Create a Forum Container ........................................ 27
  4.3.7.3 Adding Fields to Forums .......................................... 29
4.3.8 Post Settings ................................................................. 31
4.3.9 Taxonomy ...................................................................... 31
4.4 Features ............................................................................ 34
  4.4.1 Ease of Use ................................................................. 34
  4.4.2 Browser Compatibility ................................................ 35
5  CONCLUSION .......................................................................... 36
6  FUTURE WORK ..................................................................... 38
BIBLIOGRAPHY ........................................................................ 39
LIST OF TABLES

Table 5.1. Comparison of Interwork, Blackboard and Drupal ................................................37
**LIST OF FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 3.1</td>
<td>Installation options.</td>
<td>10</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>Language options.</td>
<td>10</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>Database type section.</td>
<td>10</td>
</tr>
<tr>
<td>Figure 3.4</td>
<td>Database name field.</td>
<td>10</td>
</tr>
<tr>
<td>Figure 3.5</td>
<td>Database username and database password fields.</td>
<td>11</td>
</tr>
<tr>
<td>Figure 3.6</td>
<td>Site information section.</td>
<td>11</td>
</tr>
<tr>
<td>Figure 3.7</td>
<td>Site maintenance account section.</td>
<td>11</td>
</tr>
<tr>
<td>Figure 3.8</td>
<td>Final setup list section.</td>
<td>12</td>
</tr>
<tr>
<td>Figure 3.9</td>
<td>Login and homepage.</td>
<td>12</td>
</tr>
<tr>
<td>Figure 4.1</td>
<td>High level view of the system.</td>
<td>13</td>
</tr>
<tr>
<td>Figure 4.2</td>
<td>Creating a new course.</td>
<td>15</td>
</tr>
<tr>
<td>Figure 4.3</td>
<td>Mega menus list.</td>
<td>17</td>
</tr>
<tr>
<td>Figure 4.4</td>
<td>Configuring mega menus.</td>
<td>18</td>
</tr>
<tr>
<td>Figure 4.5</td>
<td>Visibility settings mega menus (view 1).</td>
<td>18</td>
</tr>
<tr>
<td>Figure 4.6</td>
<td>Visibility settings mega menus (view 2).</td>
<td>19</td>
</tr>
<tr>
<td>Figure 4.7</td>
<td>Style settings for mega menus.</td>
<td>20</td>
</tr>
<tr>
<td>Figure 4.8</td>
<td>Adding a menu.</td>
<td>21</td>
</tr>
<tr>
<td>Figure 4.9</td>
<td>Comments approval queue.</td>
<td>21</td>
</tr>
<tr>
<td>Figure 4.10</td>
<td>Comments publishing options.</td>
<td>22</td>
</tr>
<tr>
<td>Figure 4.11</td>
<td>Editing comments.</td>
<td>22</td>
</tr>
<tr>
<td>Figure 4.12</td>
<td>Comments for comprehensive exam.</td>
<td>23</td>
</tr>
<tr>
<td>Figure 4.13</td>
<td>Content search and filter page.</td>
<td>24</td>
</tr>
<tr>
<td>Figure 4.14</td>
<td>Managing fields with the field UI.</td>
<td>24</td>
</tr>
<tr>
<td>Figure 4.15</td>
<td>Forum settings.</td>
<td>27</td>
</tr>
<tr>
<td>Figure 4.16</td>
<td>Creating a forum topic.</td>
<td>28</td>
</tr>
<tr>
<td>Figure 4.17</td>
<td>Creating a forum container.</td>
<td>29</td>
</tr>
<tr>
<td>Figure 4.18</td>
<td>Adding fields to a forum topic.</td>
<td>30</td>
</tr>
</tbody>
</table>
Figure 4.19. Post settings........................................................................................................32
Figure 4.20. Taxonomy...........................................................................................................33
Figure 4.21. Edit/add vocabulary............................................................................................34
ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to all the people who contributed for the success of this work. I would like to thank my thesis advisor and committee members Dr. Carl Eckberg, Dr. Dale Savier and Dr. Joseph Lewis. I am especially grateful to Dr. Dale Savier at the department of CBBD for his valuable time and support for his useful inputs from beginning to the completion of the project and to Dr. Carl Eckberg for his valuable suggestions.
CHAPTER 1

INTRODUCTION

1.1 MOTIVATION

The Center for Biopharmaceutical and Biodevice Development (CBBD) in the College of Sciences (COS) at San Diego State University (SDSU) department website contains information about that department such as related research work, faculty information, courses offered, student information, class tests, etc. This information is usually distributed multiple webhosts where the current website maintains the links for the respective section. The current website is completely designed in basic html using some java scripts. Any subsequent changes or additions to the website will require making a request for the same to the website administrators. This severely restricts the flexibility and usability of the website for the department and its faculty.

A department which related to computer science can easily create its own website, but a life sciences department might need external sources to maintain their website. It makes highly desirable if there was a content management system that could provide faculty members to edit or create content with the desired information on a website on their own according to access given to them whenever they want.

This type of content management system provides faculty members of the department to communicate with the students much more effectively. They can maintain up to date their course information on the departmental website for the students who enrolled for their courses such as change in curriculum, upcoming events for a class, project deadlines, exam schedules, class documents, assignments, etc. This system also support other features such as sending emails to students by class, a discussion board to interact with class, and forums etc. All these features were the motivation for this web based content management system. This system was designed for the department of CBBD at SDSU, which maintains a website with normal html pages, uses a simple access database, primarily for individual faculty to maintain current postings.
1.2 BACKGROUND ON Drupal Content Management System

Drupal is one of the best Content Management Systems (CMS). It is written in PHP and requires a MySQL database. Its basic installation can be easily turned into many different types of web sites - from simple web logs to large online communities.

Here is a list of the Drupal benefits:

- Easy to install - Drupal installation described here;
- Easy to use - no programming knowledge needed! Read this tutorial to learn the basics of Drupal;
- Lots of features including Search Engine Friendly URLs (SEF), categories, search function and many more;
- Lots of modules to extend your site’s functionality;
- Flexibility - you can easily turn your Drupal installation into a forum, blog, wiki and many other types of web sites;
- Free to use, open source. You can freely install Drupal and you can modify the source code to fit your needs;
- Lots of users and a large community - easy to find solutions to your problems;
- By enabling and configuring individual modules, an administrator can design a unique site which can be used for knowledge management, web publishing, community interaction purposes, etc.

Here are some typical Drupal usages:

- Content management - Via a simple, browser-based interface, members can publish stories, blogs, polls, images, forums, etc. Administrators can easily customize the design of their Drupal installation.
- The Drupal classification system allows hierarchical ordering, cross-indexing of posts and multiple category sets for most content types. Access to content is controlled through administrator-defined user roles. A search option is also available.
- Weblog - A single installation can be configured as an individual personal weblog site or multiple individual weblogs. Drupal supports the Blogger API, provides RSS feeds for each individual blog and can be set to ping weblog directories when new content is posted on the home page.
- Discussion-based community - A Drupal web site can be successfully used as a discussion forum. Comment boards, attached to most content types, make it simple for members to discuss new posts. Administrators can control whether content and comments are posted without approval, with administrator approval or through community moderation. With the built-in news aggregator, communities can subscribe to and then discuss content from other sites.
• Collaboration - Used for managing the construction of Drupal, the project module is suitable for supporting other open source software projects. The wiki-like collaborative book module includes versions control, making it simple for a group to create, revise and maintain documentation or any other type of text.

• FAQ - You can use the Drupal installation as a frequently asked questions platform.

1.3 CBBD DEPARTMENT

The CBBD in the COS at SDSU provides education and training that enhance the professional excellence and career opportunities of professionals in the pharmaceutical, biotechnology and medical device industries, and those interested in entering the field. Programs and interactive online courses are designed to meet the educational needs of working professionals already possessing at least a BA or BS undergraduate degree. The primary objective is for students to establish a foundation of basic education and knowledge with a practical understanding to effectively address and mediate the real world challenges encountered during the discovery, development, manufacture, and commercialization of therapeutic biologics, drugs, and medical device products as regulated by the European Medicines Agency (EMA), the Food and Drug Administration (FDA), the International Committee on Harmonization (ICH), the International Standards Organization (ISO), and other global regulatory authorities.

CBBD mission is to provide fully accredited academic and practical education, knowledge, and training programs and interactive online courses for the Biotechnology, Pharmaceutical, and Medical Device Industries, which will meet the educational needs, and enhance the professional excellence and career opportunities of professionals in the biomedical industries, or those who are interested in this field.

1.4 COMPRESSION OF CBBD, SDSU BLACKBOARD AND CLASSTA

Blackboard is a type of content management system used by SDSU and most other local colleges, including two year programs. Content can be created and made visible to the students. In blackboard, the data is backed up after each semester and the student or faculty cannot view their previous semester’s content. For a student, preparing for a comprehensive test would require going through all the courses taken up previously. It would be helpful for a student to view all the past classes taken. All the class lectures and assignments could be
glanced through for a quick revision. This feature is provided by this thesis and is supported so that a student has the ability to go through past classes.

CBBD website requires proper authentication from the administrator, for the students or instructors to login. ClassTA [1] allows student registration, from which a password is obtained for future authentication. By contrast the tool in this thesis requires an administrator or faculty to authorize the addition of an user. The tradeoff is an interesting one, and might be able to compromise by allowing a class instructor to permit the registration approach by a checkbox. The existence of online classes often justifies authorizing student sign-ups, as opposed to self-registration. The ClassTA [1] website provides no authentication and a student can directly sign up for the website. Instructors can manually add students to the class roster or they can add to the list by uploading a spreadsheet which contains the student information whereas a centralized database [2] is used for the CBBD website. All the information is stored in the database and the content is displayed dynamically using the data driven from the tables and presented as a webpage.
CHAPTER 2

TECHNOLOGIES

2.1 PHP

PHP, A general purpose scripting language which is designed for web development. It is originally designed to produce dynamic pages by embedding PHP code into the HTML source documents. To generate webpage documents, HTML Source documents are interpreted with a PHP processor Module using a web server. PHP also includes command-line interface capability. PHP can also be used in the stand-alone graphical operating system and can be deployed as a standalone interpreter on almost every operating system on any web server.

PHP was originally created by Rasmus Lerdorf in 1995. The main implementation of PHP is now produced by The PHP Group and serves as the de facto standard for PHP as there is no formal specification.

PHP can be used with many relational database management systems (RDBMS). PHP primarily acts as a filter, [3] taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. Basic object-oriented programming functionality was added in PHP 3 and improved in PHP 4 [4]. Object handling was completely rewritten for PHP 5, expanding the feature set and enhancing performance [5]. In previous versions of PHP, objects were handled like value types. Since PHP 4, the PHP parser compiles input to produce bytecode for processing by the Zend Engine, giving improved performance over its interpreter predecessor. It also introduced a standard way of declaring constructors and destructors, similar to that of other object-oriented languages such as C++, and a standard exception handling model.

PHP includes free and open source libraries with the core build. PHP is a fundamentally Internet-aware system with modules built in for accessing FTP servers, many database servers, embedded SQL libraries such as embedded PostgreSQL, MySQL and SQLite, LDAP servers, and others. Many functions familiar to C programmers such as those in the studio family are available in the standard PHP build [6].
2.2 MYSQL

MySQL is a RDBMS [4] that runs as a server providing multi-user access to a number of databases. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP web application software stack - LAMP is an acronym for “Linux, Apache, MySQL, Perl/PHP/Python”.

MySQL is written in C and C++. Its SQL parser is written in yacc, and a home-brewed lexical analyzer named sql_lex.cc. MySQL is primarily an RDBMS and therefore ships with no GUI tools to administer MySQL databases or manage data contained within. Users may use the included command-line tools [7], or download MySQL frontends from various parties that have developed desktop software and web applications to manage MySQL databases, build database structure, and work with data records.

MySQL implements the following features, which some other RDBMS systems may not:

- Multiple storage engines, allowing one to choose the one that is most effective for each table in the application (in MySQL 5.0, storage engines must be compiled in; in MySQL 5.1, storage engines can be dynamically loaded at run time).
- Native storage engines (MyISAM, Falcon, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, Cluster, Berkeley DB, EXAMPLE, Maria, and InnoDB, which was made the default as of 5.5).
- Partner-developed storage engines (solidDB, NitroEDB, Infobright (formerly Brighthouse), Kickfire, XtraDB, IBM DB2 [8]). InnoDB used to be a partner-developed storage engine, but with recent acquisitions, Oracle now owns both MySQL core and InnoDB.
- Community-developed storage engines (memcache engine, httpd, PBXT, Revision Engine).
- Custom storage engines.
- Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.
CHAPTER 3
INSTALLATION

This chapter discusses about the installation of Drupal in a webhost.

3.1 BASIC INSTALLATION

Drupal provides an installation script that automatically populates database tables and configures the correct settings in the settings.php file. This section covers preparing for installation, running the installation script itself, and the steps that should be done after running the installation script has completed. It also explains how to do a "multi site" installation, where a number of different Drupal sites run off the same code base.

- Download Drupal - Using your web browser, navigate to http://Drupal.org/project/Drupal. Click the appropriate “tar.gz” or “zip” file link beside the version of Drupal you want to download. Upload the Drupal files to your web hosting account.
  1. Log in to your hosted account’s control panel and navigate to your HTML directory.
  2. Upload your compressed .tar.gz or .zip file and uncompress the file on the server.
  3. Rename the folder from its current name: “Drupal-6.x” to the name you want to use to refer to your site, such as “CBBD”.

- Set up a Domain and point it to the directory containing Drupal’s files.
  1. Register a Domain Name e.g. http://example.com.
  2. Configure your hosting account to use this domain name.
  3. Point the domain name to your new folder/directory, which contains the uncompressed Drupal files.

- Create the configuration file and grant permissions.
  In order to set up your new site, it is necessary to be able to modify the settings.php file via your browser. By default when you unarchive these files, only the default.settings.php file exists and the permissions are set to Read Only.
You will need to copy the file, rename it to settings.php, and then temporarily change the permissions so that the server can Read and Write to this file when a user makes changes via a web browser.

You then need to move up one level to the /sites directory and change the permission on the /default directory (folder), as this is the place where files created or uploaded via the web interface will be stored.

1. Copy the file default.settings.php and rename it to settings.php.
   - Use your ISP’s file manager to navigate to the /sites/default directory and select the default.settings.php file.
   - Copy the file, naming the new copy ‘settings.php’.

2. Change the permissions of the settings.php file and /default directory.
   - Add the permissions for the file to be written by the web server and save your changes.
   - In your ISP’s file manager select the /default folder and click on permissions.
   - Add the permissions for the folder to be written by the web server and save your change.

3.2 CREATE THE CBBD DATABASE

You must create a new, empty database for CBBD site to use. You must also add a user who has full access to this newly created database.

One of the most popular tools used to administer databases is “phpMyAdmin”. It is possible that you will have privileges to create new databases using this tool. However, it is more likely that your ISP provides access to phpMyAdmin so that you can work with content in your database.

- Create a new Database using phpMyAdmin.
  1. From your ISP’s control panel, open phpMyAdmin.
  2. In the Create new database field, type the name you want to use for your new Drupal database and then click Create.
  3. Navigate to Privileges tab-> Add a new User.
     1. In the User name field, type the username that you want to add.
     2. In the drop-down menu beside the Host field, select Local.
     3. In the Password and Re-type fields, type a password to use for the new user.
4. In the “Database for user” section, select “Grant all privileges” on the database you just created.

5. In the “Global privileges” section, leave all of the global privileges checkboxes unselected.

6. Click Go.

To run the Drupal installation script:

- Using your web browser, navigate to the base URL of your new website, e.g. http://example.com When you go to your new website, you should see the Drupal installation page. The installation wizard will guide you through the process of setting up your Drupal website.
  - Step 1: Choose language. The default for this is English. Choose that.
  - Step 2: Verify requirements. This page just asks you for basic requirements.

4. Select either the Standard or Minimal installation (see Figure 3.1).

5. Select the Save and continue button (see Figure 3.2).

6. On the Choose language page select the Save and continue button.

7. In the Database type section (see Figure 3.3) select MySQL, MariaDB, or equivalent.

8. In the Database name field (see Figure 3.4) enter the database name that you created on phpMyAdmin.

9. In the Database username and database password fields (see Figure 3.5) enter the user name and password that you created on phpMyAdmin.

10. Select the Save and continue button.

11. Complete the SITE INFORMATION section (see Figure 3.6).

12. Complete the SITE MAINTENANCE ACCOUNT section (see Figure 3.7).

13. Choose the SERVER SETTINGS and UPDATE NOTIFICATIONS preferences.

14. Select the Save and Continue button.

15. Select the Visit your new site link. See Figure 3.8.

### 3.3 LOGGING INTO CBBD SITE ADMIN AREA

To login your Drupal Admin area, after installing the Drupal application, you simply need to enter your administrator credentials into the User login form on your Drupal site. See Figure 3.9.
Figure 3.1. Installation options.

Figure 3.2. Language options.

Figure 3.3. Database type section.
Figure 3.4. Database name field.

Figure 3.5. Database username and database password fields.

Figure 3.6. Site information section.

Figure 3.7. Site maintenance account section.
Figure 3.8. Final setup list section.

Figure 3.9. Login and homepage.
CHAPTER 4

APPLICATION OVERVIEW

4.1 HIGH LEVEL VIEW

Figure 4.1 gives a high level overview of this web content management system.

Figure 4.1. High level view of the system.
4.2 IMPLEMENTATION

This chapter describes about the implementation of this web content management system. The thesis aims to meet the following requirements and the implementation of these is discussed in this chapter:

1. Allow faculty members to create their accounts with the system and use this account to access and update their information whenever they want.
2. Support for faculty to maintain their personal information on their respective home pages.
3. Support for faculty to add classes to the website and manage related information.
4. Faculty should be able to upload images for each class.
5. Allow students to register for desired classes to receive emails from the respective faculty. Faculty should be able to send emails to registered students per class using the website interface.
6. Have a discussion board for each class. Both faculty and students should be able to access and update the contents of this section. Also, the concerned faculty should be notified via email when the content gets updated.
7. The website should be fully web accessibility compliant.

Implementation of these requirements is explained next.

4.2.1 Create Account

The first step in using this website to manage information is to create an account. This feature is password protected, as only the faculty members should be allowed to create an account and not any students or visitors. On the login page, there is a link as “New User” using which an account can be created, after proper authentication. To create an account, the user needs to enter a valid email address, name and password. The email address and password will be required for subsequent login. In case the user forgets his password, there is an option “Forgot Password” on the login page; using which he can have the password emailed to him at this email address. The user can delete his account when no longer needed; it will permanently delete all information associated with it.

4.2.2 Create a Course

The faculty members can add classes to this website, once they have an account. The interface to add a class is exactly the same as that for entering personal information. Each faculty member can add and delete classes as needed. When a class is deleted, all related
information such as a description of the class, uploaded images, contents on the discussion board and information on registered students are permanently deleted. When a class is added, several features for that class become available such as sending emails, uploading images and interacting on a discussion board. These features are discussed next. Figure 4.2 gives you how to create a new course.

Figure 4.2. Creating a new course.

4.3 CONTENT MANAGEMENT

In order to manage the content of your Drupal web site, you need to navigate to the Content management section under your administrator area. In the following lines we will give a brief description of the options under this section:
1. Drupal Comments - With the default installation, only users that have logged in can access/post comments. This can be changed from Administer -> User management -> Permissions. The comments settings are self-explanatory. By default, registered users can post comments (they don’t need to wait for moderator approvals). If you want to let anonymous users post comments, go to Administer -> User management -> Permissions -> comment module.

2. Content - You can manage, view, edit and delete the site content. You can publish and unpublish it.

3. Content types - Manage posts by content type, including default status, front page promotion. You can include new content types besides the Page and the Story.

4. Post settings - You can control the post behavior, the length of the posts and the number of posts per page.

5. Taxonomy - You can manage the tagging, the categorization and classification of your content.

4.3.1 Create a New Content

You can create new content through the Create content section under the site administrator area:

- The Page option allows you to create a static page.
- The Story option is suitable for a blog or a discussion usage.

4.3.2 Mega Menus

Mega menus are a type of menu drop-down, which seek to overcome some of the usability drawbacks to traditional style drop-downs. Primarily, mega menus have two advantages:

- The information architecture of navigation is made clearer and is visible all at once.
- The amount of clicking, searching, and navigating is reduced.

Mega menu takes a site menu that is 3-levels deep, and converts it into a categorized 3-level mega-dropdown-style menu, which can be placed into the theme as a block.

Each menu is allowed to act as a mega menu by just enabling it and it could also be configured. The list of mega menus created to the CBBD site is listed in Figure 4.3.

Various settings can be applied to mega menus and they are block specific settings. Each mega menu block is given a block title. Unless given a title, the default title is used. There are user specific visibility settings which could be customized as well. It has options to
Figure 4.3. Mega menus list.

let users control whether or not they see a particular block, hide the block by default but let individual users show it or show the block by default but let individual users hide it.

Role specific visibility settings specify the roles that can access a particular block. If no roles are selected, the block will be visible to all the users as shown in Figure 4.4.

Page specific visibility settings can also be configured. The administrator is allowed to show the block on every page excepted the listed pages or show on only the listed pages or show if some code block returns true as shown in Figure 4.5.

The order of the mega menu could also be specified. Each mega menu can be given a position and can be configured accordingly as shown in Figure 4.6 by dragging.
Figure 4.4. Configuring mega menus.

Figure 4.5. Visibility settings mega menus (view 1).
Figure 4.6. Visibility settings mega menus (view 2).

The mega menus could be set based on the orientations and Style settings. The Menu orientation could either be horizontal and drops down or it could be vertical and flies out. The slot orientation could be either columnar, in which the slots will sit next to each other or stack on top of each other.

The Style settings start by selecting a skin type which could be supplied with the module or skin which could be customized. A custom name would be requested for customized skins. Once the “save configuration” button is clicked, the orientations and the style settings are applied as shown in Figure 4.7.

4.3.3 Menus

Menus could be added by the user and the user is allowed to select from a list of menus added. A menu name is requested for the menu to be created and the text entered would be used for constructing the URL of the menu overview page for the menu created. The menu name entered has to be unique and it must contain only lowercase letters, numbers and hyphens. A title and a precise description is expected for every menu which provides an apt overview for the menu. Creating or adding a menu is shown in Figure 4.8.

The content management tab presents the user with a list of navigation links to manage the website’s content that the user is authorized for. The links include comments, content and the various types of content, tools to modify and present date, forums to participate in active discussions and post threads, post settings, RSS publishing and taxonomy.
4.3.4 Comments

The comments section allows the user to list and edit the site comments. The content section allows the user to view, edit and delete the website’s content.

Every comment posted in the comment queue must be approved as shown in Figure 4.9. To approve a comment, click on edit and then change its “moderation status” to Approved. The author’s name could be clicked and the author’s information could be edited. User submission could be deleted by clicking “delete”. Selected comments could be updated upon selection by either publishing or deleting them as shown in Figure 4.10.
Figure 4.8. Adding a menu.

Figure 4.9. Comments approval queue.
The comments section’s usage could be explained with a well-defined example on Comprehensive exams. Figure 4.11 explains the comments on the topic of comprehensive exam. A Comprehensive exam topic is created and every student is allowed to comment on that topic. The comments include their views on the exam, the questions and their possible answers. The instructor could intervene and edit the comments by correcting them or can also delete the comments if it does not relate to the questions on the exam. See Figure 4.11 and 4.12.

The content page allows the user to view all the content properties. It also allows the user to customize the manner in which the content page is displayed based on status, type and category. The content could also be filtered depending upon the user options. The filtered content could also be published. The content page displays the contents in a table which includes the type of the content, its author, the status of the content and also the operations that could be performed on that particular content which might include editing the content as well.
4.3.5 Content Search

The contents could be filtered using a wide variety of options presented to the user. The update options include publishing the contents, unpublish, promoting the content to the front page, demoting from the front page or delete the content. The type drop down also has a list of options that let the user the type as a story, page, poll, new course or allow the user to submit assignment. See Figure 4.13.

4.3.6 Working with Content Types and Fields

The Field UI module provides an administrative user interface (UI) for attaching and managing fields. Fields can be defined at the content-type level for content items and comments, at the vocabulary level for taxonomy terms, and at the site level for user accounts. Other modules may also enable fields to be defined for their data. Field types (text, image, number, etc.) are defined by modules, and collected and managed by the Field module. See Figure 4.14.

There are several decisions you will need to make before defining a field for content, comments, etc.:
Figure 4.13. Content search and filter page.

Figure 4.14. Managing fields with the field UI.
• **What type of data the field will store**
  Each field can store one type of data (text, number, file, etc.). When you define a field, you choose a particular field type, which corresponds to the type of data you want to store. The field type cannot be changed after you have created the field.

• **How the data will be input and displayed**
  Each field type has one or more available widgets associated with it; each widget provides a mechanism for data input when you are editing (text box, select list, file upload, etc.). Each field type also has one or more display options, which determine how the field is displayed to site visitors. The widget and display display options can be changed after you have created the field.

• **How many values the field will store**
  You can store one value, a specific maximum number of values, or an unlimited number of values in each field. For example, an employee identification number field might store a single number, whereas a phone number field might store multiple phone numbers. This setting can be changed after you have created the field, but if you reduce the maximum number of values, you may lose information.

• **Reusing fields**
  There are two main reasons for reusing fields. First, reusing fields can save you time over defining new fields. Second, reusing fields also allows you to display, filter, group, and sort content together by field across content types. For example, the contributed Views module allows you to create lists and tables of content. So if you use the same field on multiple content types, you can create a View containing all of those content types together displaying that field, sorted by that field, and/or filtered by that field.

• **Fields on content items**
  Fields on content items are defined at the content-type level, on the Manage fields tab of the content type edit page (which you can reach from the Content types page). When you define a field for a content type, each content item of that type will have that field added to it. Some fields, such as the Title and Body, are provided for you when you create a content type, or are provided on content types created by your installation profile.

• **Fields on user accounts**
  Fields on user accounts are defined on a site-wide basis on the Manage fields tab of the Account settings page. When you define a field for user accounts, each user account will have that field added to it. For example, you could add a long text field to allow users to include a biography.

• **Fields on comments**
  Fields on comments are defined at the content-type level, on the Comment fields tab of the content type edit page (which you can reach from the Content types page). When you add a field for comments, each comment on a content item of that type will have that field added to it. For example, you could add a website field to the comments on forum posts, to allow forum commenter’s to add a link to their website.
4.3.7 Forum/Discussion Board

The Forum core module lets you create threaded discussion boards, called forums, on your site. This is similar to a message board system such as phpBB. Forums are very useful because they allow community members to discuss topics with one another, and the discussions are archived for future reference.

This option appears as ‘Ask a Question’ on the visitor/student interface of the website. This feature applies to every class added to the website. As only students should be allowed to post in this section and not any random visitor. If this is found to be inadequate security, more elaborate authentication can be added later. After this authentication, students can post any question or information on the Forum and it will be visible to all who have access to it.

Topics, or threaded discussions, are posted to a forum. Each topic is comprised of an initial post and replies, or comments. Forum topics can have their own URLs.

Multiple forums are often set up for different areas of discussion. Forums can also be placed inside other forums for a discussion area that is a subset of the parent forum, in which case they are often called child boards or child forums.

This feature can be very useful for interactions or discussions between students. Students can ask questions on current project or homework and other students or faculty can reply to them.

As an administrator one of the first things to do is to add appropriate permissions.

To access the appropriate permissions you can:

- Modules >> Forum >> Permissions (in the “operations” column on the right); or
- People >> Permissions (tab on the top right)

Scroll down the Permissions administration panel until you reach the Forum item. Here you may want to add Forum administrations privileges to specific user types by letting them access the Forum administration panel. There are five more Forum permission settings to set under the Node category. See Figure 4.15.

4.3.7.1 CREATE A FORUM

Users can post topics to forums. Forums can appear below other forums in a hierarchy, or can appear in containers. The following are the steps for creating or starting
thread discussion boards. The Forum creation page with all the fields need to create Forum is shown on Figure 4.16.

1. Navigate to the Forums administration page (/admin/structure/forum).
2. Click Add Forum.
3. In the Forum Name field, enter the name of the forum.
4. In the Description field, enter an optional description.
5. In the Parent list, select the parent of the forum. You can select the root of the site, another forum, or a forum container.
6. In the Weight list, enter an optional weighting. A smaller number will weigh the forum higher in a list.
7. Click Save.

4.3.7.2 CREATE A FORUM CONTAINER

Containers can also be created to hold a group of forums. However, unlike in parent forums, topics cannot be posted to containers. Both containers and forums can be placed
inside other containers and forums. By planning the structure of your containers and forums well, you make it easier for users to find a discussion area of interest to them. The steps for creating forum contains are listed below. The fields that should be added to the forum container are thrown in Figure 4.17.

1. Navigate to the Forums administration page (/admin/structure/forum).
2. Click Add Container.
3. In the Forum Name field, enter the name of the container.
4. In the Description field, enter an optional description.
5. In the Parent list, select the parent of the forum. You can select the root of the site, another forum, or a forum container.

6. In the Weight list, enter an optional weighting. A smaller number will weigh the container higher in a list.

7. Click Save.

**4.3.7.3 ADDING FIELDS TO FORUMS**

Adding fields and groups to the forum content type and arrange them on forum content display and input forms. We can add a field to a particular group by dragging it below and to the right of the group. Figure 4.18 shows all the fields which are created and how to add new fields in ADD block. We can also change order of all the fields by dragging them to required position. We can also add/change field type as shown in Figure 4.17.

It is easy to add fields to the forum. It is similar to adding fields as in normal content type. We can also add existing field to the forum topic from various content types. We can page, in XML feeds, etc. To disable teasers, set to ‘Unlimited’. Note that this setting also get
Figure 4.18. Adding fields to a forum topic.
the values from the other content types by adding its fields with corresponding query which should be written in PHP for this content itself while creating a page.

4.3.8 Post Settings

Under Administer >> Content Management >> Post Settings you may access the following options (see Figure 4.19):

- Rebuild permissions: If the site is experiencing problems with permissions to content, you may have to rebuild the permissions cache. Possible causes for permission problems are disabling modules or configuration changes to permissions. Rebuilding will remove all privileges to posts, and replace them with permissions based on the current modules and settings. Rebuilding may take some time if there is a lot of content or complex permission settings. After rebuilding has completed, posts will automatically use the new permissions.

- Number of posts on main page: Provides a dropdown box that determines the default maximum number of posts to display per page on overview pages such as the main page.

- Length of trimmed posts: Provides a dropdown box that determines the maximum number of characters used in the trimmed version of a post. Drupal will use this setting to determine at which offset long posts should be trimmed. The trimmed version of a post is typically used as a teaser when displaying the post on the main will only affect new or updated content and will not affect existing teasers.

- Preview post: Provides two radio buttons that determine whether it is optional or required for users to preview their posts before submitting.

4.3.9 Taxonomy

Taxonomy is the practice of classifying things. In Drupal, the taxonomy module is how you classify your website content, and it is an important piece of many sites’ information architecture.

It is helpful to think of taxonomy as the use of “categories”. In fact, this is what it was called in earlier versions of Drupal. Taxonomy lets you gather together content under one term or another. It has become advanced enough to give you as much flexibility as you need in designing classification schemes. See Figure 4.20.

Taxonomy module categorizes the content using some predefined terms as well as tags. Content could be classified using a wide variety of advanced features. Initially, a vocabulary could be created that comprises of one set of terms or tags. A vocabulary can be associated with a type and could be related to a specific content. When a vocabulary is
Figure 4.19. Post settings.

associated with a particular type, it would be displayed when creating or editing posts of that type. Multiple vocabularies could also be tied to the same content type and they would be displayed as well. The order of the vocabulary could be changed by using the drag and drop handle. However, the changes would be saved only after the save button is clicked at the bottom of the page.

- Adding a Vocabulary
  1. Choose a name for the vocabulary.
  2. Give the vocabulary a description. Modules may use this description in different ways. (For example, a module may show the description when users hover over a link.)
3. The vocabulary could be given a help text to help the users choose. Note that when creating a new content type, existing vocabularies do not have the new content type selected; you need to edit them to add the type.

4. Finally, the vocabulary could be deleted altogether, thereby also deleting all its terms, but not the nodes to which they were assigned. See Figure 4.21.

- Principles that Apply to CBBD Site Taxonomy Module
  1. Each vocabulary consists of a set of terms.
  2. A site can have an unlimited number of vocabularies.
  3. Each vocabulary has an associated id number. Each taxonomy term has an associated id number.
  4. Each vocabulary can contain an unlimited number of terms.
  5. Within a vocabulary, terms can be ordered into hierarchies.

- Setting Up Taxonomy
  1. Administer taxonomies at administer > content management > Taxonomy.
  2. Decide who else can administer the Taxonomy module at Administer > User management > Permissions.
  3. Add a vocabulary at Administer > Content management > Taxonomy > Add Vocabulary.
4.4 FEATURES

This section discusses some key features of this system.

4.4.1 Ease of Use

The website is carefully designed to be user friendly. Its design is very intuitive and informative so that even people with little experience in computers can easily use this system. A user friendly feature of this website is ‘title’ information for each item in each web page. Moving the mouse pointer over any item such as buttons, text boxes, images, etc will display this title information below the mouse pointer indicating its function or intent. This greatly simplifies interaction with the website as the use of each item on each page becomes known.

Figure 4.21. Edit/add vocabulary.
Also, the design of the interface through which the faculty will manage their information, is exactly the same as of the interface to the visitors of the website. So while entering information, the faculty will get a close idea of how it will actually look on the website. In addition, a user manual with step by step instructions on using this system has also been provided to the department.

4.4.2 Browser Compatibility

The website is carefully designed and implemented so that it is compatible with different browsers. The website is tested and verified to be working fine on Firefox 3+ and onwards, Internet Explorer 7+ and onwards, Safari 5+ and Google Chrome 8+. 
CHAPTER 5

CONCLUSION

Table 5.1 briefly lists all the features and compares them with the interworks, blackboard and Drupal. The next paragraphs discuss more in detail.

On the CBBD website, a public profile is provided which is visible to all the students. They could only see the basic information about the students (First Name, Last Name).

An online quiz or comprehensive exam feature is also available in the CBBD website. After a thorough comparison of CBBD, ClassTA exhibits several nice features including online assignment grading, adding weights to the questions, penalty points if the deadline isn’t met, adding comments to each question of the assignment or the entire assignment. These features are listed in the future work of the thesis. And other issues are visibility grades, current rank in class, current movable letter grade.

The website has been used by the students and instructors of the CBBD department during Summer 2011 for three classes.
Table 5.1. Comparison of Interwork, Blackboard and Drupal

<table>
<thead>
<tr>
<th>Feature</th>
<th>Interwork</th>
<th>Blackboard</th>
<th>Drupal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sign on</td>
<td>One user/password</td>
<td>User specific</td>
<td>User specific</td>
</tr>
<tr>
<td>Security</td>
<td>Weak</td>
<td>Strong</td>
<td>Strong</td>
</tr>
<tr>
<td>Look and Feel</td>
<td>Easy to use</td>
<td>Easy to use</td>
<td>Easy to use</td>
</tr>
<tr>
<td>Handling of Rosters</td>
<td>Manual</td>
<td>Automatic from web portal</td>
<td>Parsed from web portal</td>
</tr>
<tr>
<td>Handling of student informational web pages</td>
<td>HTML</td>
<td>HTML</td>
<td>Text editing of fields in data tables</td>
</tr>
<tr>
<td>Video, audio, slide presentations</td>
<td>Excellent, lacks security</td>
<td>Excellent, secure</td>
<td>Excellent, secure</td>
</tr>
<tr>
<td>External links to FDA</td>
<td>Paste URL</td>
<td>Paste URL</td>
<td>URL from database</td>
</tr>
<tr>
<td>Handling of student input(Web exercises)</td>
<td>Email responses</td>
<td>Upload doc file</td>
<td>Direct entry into data table</td>
</tr>
<tr>
<td>Discussion forums</td>
<td>Proprietary file</td>
<td>Proprietary file</td>
<td>Direct entry into data table</td>
</tr>
<tr>
<td>Assignment Upload</td>
<td>Upload doc file</td>
<td>Upload doc file</td>
<td>Upload doc file</td>
</tr>
<tr>
<td>Peer review</td>
<td>Email responses</td>
<td>Upload doc file</td>
<td>Direct entry into data table</td>
</tr>
<tr>
<td>Online Meetings</td>
<td>None</td>
<td>WIMBA</td>
<td>None</td>
</tr>
<tr>
<td>Grading - Analyzing and comments</td>
<td>Pedantic</td>
<td>Powerful</td>
<td>Not Complete</td>
</tr>
<tr>
<td>Instructors ability to edit content</td>
<td>None</td>
<td>Some</td>
<td>All</td>
</tr>
<tr>
<td>Information for Instructors</td>
<td>None</td>
<td>Some</td>
<td>Detailed</td>
</tr>
<tr>
<td>Information for Administrators</td>
<td>None</td>
<td>Some</td>
<td>Detailed</td>
</tr>
<tr>
<td>Cost per Student</td>
<td>$100</td>
<td>$15</td>
<td>$0</td>
</tr>
</tbody>
</table>
CHAPTER 6

FUTURE WORK

Any project has the potential to get enhanced and some of the enhancements for my work are:

• The instructor could also be allowed to authenticate the user.
• Password restrictions could be added to password creation and expiration.
• Grade evaluation based on all the exams and assignments for a student.
• Students could be allowed to pick their selected schedule from the general class schedule.
• The students would be given access to program of study which lists out the courses took by a student towards the completion of the respective degree.
• Allow students to poll on some selected issues and let the instructor decide based on the poll result.
• The administrator or faculty could notify the events to the students on the home page.
• Email notifications could also be included which would remind the student of the deadlines ahead related to exams, assignments and any other course related information.
BIBLIOGRAPHY


