Detailed Design

Requested by: Dr. Eric Breimer
Web Master & Assistant Professor
Department of Computer Science
Siena College

Dr. Tim Lederman
Professor
Department of Computer Science
Siena College

Alumni Spotlight Web System
(ASWS)

Initrode Solutions

Prepared by: Anthony Angelucci
Charles Feltes
Elise Hearn
David Luck
Christopher McConnell

March 6th 2007
# Alumni Spotlight Web System
## Detailed Design

### Table of Contents

I. External Design Specifications ................................................................. 1

#### User displays and report formats ........................................................ 1
  Administrator Home Page Screen .............................................................. 1
  Administrator Access Control Screen ......................................................... 2
  Administrator Content Screen ................................................................. 3
  Administrator Main Menu Screen ............................................................ 4
  Administrator Menus Screen .................................................................. 5
  Administrator Menu Building Screen ...................................................... 6
  Administrator Recent Events Screen ....................................................... 7
  Administrator Settings Screen ................................................................ 8
  Administrator Modules Screen ............................................................... 9
  Administrator Users Screen ................................................................... 10
  Browse User List Screen ....................................................................... 11
  Comments Screen .................................................................................. 12
  Contact Us Screen ................................................................................. 13
  Edit Profile Screen ................................................................................. 14
  Edit Profile Screen Continued ................................................................. 15
  User Home Page Screen ......................................................................... 16
  Profile Page Screen ............................................................................... 18
  Profile Page Screen Continued ............................................................... 19
  Registration Screen ............................................................................... 20
  Registration Screen Continued .............................................................. 21
  Registration Screen Continued .............................................................. 22
  Search Yielding Results Screen .............................................................. 25

#### User command summary ................................................................. 28

II. Detailed Data Flow Diagrams ............................................................... 33

#### Symbols Key .................................................................................. 33

#### Context Diagram ............................................................................ 34

#### Level 0 Diagram ............................................................................. 35

#### Level 1 Diagram ............................................................................. 36

#### Level 2 Diagram ............................................................................. 37

#### Level 3 Diagram ............................................................................. 38

#### Logical Data Stores ......................................................................... 39

#### Logical Data Dictionary .................................................................... 49

#### Logical Format of Data Files and Databases ...................................... 72
II. Architectural Design Specification ................................................................. 72

Structure Diagrams ............................................................................................ 72

III. Testing Requirements .................................................................................... 73

Requirements Inventory ....................................................................................... 73
Use Case Scenarios .............................................................................................. 73
Functional Requirements Inventory .................................................................... 75
Non-Functional Requirements Inventory .......................................................... 77

Test Plan .............................................................................................................. 78

Documentation Prologue for Each Routine ......................................................... 78
Community User ................................................................................................. 78
1.1 Login ......................................................................................................... 78
1.2 Registration .............................................................................................. 78
1.3 Modifies Account ...................................................................................... 78
1.4 Forgotten Password .................................................................................. 78
Anyone (siena.edu) ........................................................................................... 78
2.1 Sends Request to View the Spotlighted Page ............................................ 79
Administrative User ........................................................................................... 79

Pseudocode for Each Routine ............................................................................ 80
Community User ................................................................................................. 80
1.1 Login ......................................................................................................... 80
1.2 Registration .............................................................................................. 80
1.3 Modifies Account ...................................................................................... 80
1.4 Forgotten Password .................................................................................. 80
Anyone (siena.edu) ........................................................................................... 80
2.1 Sends Request to View The Spotlighted Page ............................................ 80
Administrative User ........................................................................................... 80

Physical data structures and data file specifications ......................................... 82
Access ................................................................................................................. 82
Authmap ............................................................................................................. 82
Blocks Roles ....................................................................................................... 83
Blocks ................................................................................................................ 83
Boxes .................................................................................................................. 84
Cache .................................................................................................................. 84
Cache Filter ........................................................................................................ 84
Cache Page ....................................................................................................... 85
Cache Menu ....................................................................................................... 85
Comments ........................................................................................................ 86
Content Type Bio .............................................................................................. 86
Content Type Story .......................................................................................... 87
Content Type Page ............................................................................................ 87
Content Type User Node ................................................................................... 87
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Revisions</td>
<td>88</td>
</tr>
<tr>
<td>Files</td>
<td>88</td>
</tr>
<tr>
<td>Filter Format</td>
<td>89</td>
</tr>
<tr>
<td>Filters</td>
<td>89</td>
</tr>
<tr>
<td>Flood</td>
<td>89</td>
</tr>
<tr>
<td>History</td>
<td>90</td>
</tr>
<tr>
<td>Menu</td>
<td>90</td>
</tr>
<tr>
<td>Node Access</td>
<td>91</td>
</tr>
<tr>
<td>Node Comment Statistics</td>
<td>92</td>
</tr>
<tr>
<td>Node Counter</td>
<td>92</td>
</tr>
<tr>
<td>Node Field</td>
<td>92</td>
</tr>
<tr>
<td>Node Field Instance</td>
<td>93</td>
</tr>
<tr>
<td>Node Group</td>
<td>93</td>
</tr>
<tr>
<td>Node Group Fields</td>
<td>93</td>
</tr>
<tr>
<td>Node Revisions</td>
<td>94</td>
</tr>
<tr>
<td>Node Type</td>
<td>94</td>
</tr>
<tr>
<td>Node Family</td>
<td>95</td>
</tr>
<tr>
<td>Permission</td>
<td>95</td>
</tr>
<tr>
<td>Watchdog</td>
<td>95</td>
</tr>
<tr>
<td>Profile Values</td>
<td>96</td>
</tr>
<tr>
<td>Profile Fields</td>
<td>96</td>
</tr>
<tr>
<td>Roles</td>
<td>97</td>
</tr>
<tr>
<td>Search Dataset</td>
<td>97</td>
</tr>
<tr>
<td>Search Total</td>
<td>97</td>
</tr>
<tr>
<td>Search Index</td>
<td>98</td>
</tr>
<tr>
<td>Sequences</td>
<td>98</td>
</tr>
<tr>
<td>Sessions</td>
<td>98</td>
</tr>
<tr>
<td>System</td>
<td>99</td>
</tr>
<tr>
<td>Term Node</td>
<td>99</td>
</tr>
<tr>
<td>Term Hierarchy</td>
<td>99</td>
</tr>
<tr>
<td>Term Data</td>
<td>100</td>
</tr>
<tr>
<td>Term Relation</td>
<td>100</td>
</tr>
<tr>
<td>Term Synonym</td>
<td>100</td>
</tr>
<tr>
<td>URL alias</td>
<td>101</td>
</tr>
<tr>
<td>Usernode</td>
<td>101</td>
</tr>
<tr>
<td>User Roles</td>
<td>101</td>
</tr>
<tr>
<td>Variable</td>
<td>102</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>102</td>
</tr>
<tr>
<td>Vocabulary Node Types</td>
<td>102</td>
</tr>
</tbody>
</table>

Packaging specifications.............................................................................. 103

Glossary of Terms.............................................................................................. 104

Development Timeline – Gantt Chart................................................................ 106
I. External Design Specifications

User displays and report formats

Administrator Home Page Screen

Welcome to the Computer Science Alumni Spotlight Web System.

We at Siena College would like to thank you for taking the time to create your user account here with us. As you know from your experience here at Siena, we are a college that is focused on creating a community, not just an educational institution. Your help in greatly appreciated in this matter, as we hope to "spotlight" users from this system to show prospective students what a degree in Computer Science can do for them.

From all of us here in the Computer Science Department, we would like to thank you once again.

-- Chris McConnell
-- Head Developer | Siena College Alumni Spotlight Web System

Back to top
### Administrator Access Control Screen

#### Access Control

Permissions let you control what users can do on your site. Each user role (defined on the user roles page) has its own set of permissions. For example, you could give users classified as "administrators" permission to "administer nodes" but deny this power to ordinary, "authenticated" users. You can use permissions to reveal new features to privileged users (those with subscriptions, for example). Permissions also allow trusted users to share the administrative burden of running a busy site.

<table>
<thead>
<tr>
<th>Permission</th>
<th>admin user</th>
<th>anonymous user</th>
<th>authenticated user</th>
</tr>
</thead>
<tbody>
<tr>
<td>block module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>administrator blocks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>use PHP for block visibility</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>comment module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>access comments</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>administrator comments</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>post comments</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>post comments without approval</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>filter module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>administrator filters</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>menu module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>administrator menu</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>node module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>access content</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>administrator content types</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>administrator nodes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>create bio content</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>create page content</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>create story content</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>edit bio content</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>edit own bio content</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Administrator Content Screen

### Content

<table>
<thead>
<tr>
<th>Title</th>
<th>Type</th>
<th>Author</th>
<th>Status</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>contact</td>
<td>page</td>
<td>drupaladmin</td>
<td>published</td>
<td>edit</td>
</tr>
<tr>
<td>sd3242 new</td>
<td>usermod</td>
<td>sd3242</td>
<td>published</td>
<td>edit</td>
</tr>
<tr>
<td>seh5924 new</td>
<td>usermod</td>
<td>seh5924</td>
<td>published</td>
<td>edit</td>
</tr>
<tr>
<td>oak5677 new</td>
<td>usermod</td>
<td>oak5677</td>
<td>published</td>
<td>edit</td>
</tr>
<tr>
<td>Home Page</td>
<td>page</td>
<td>drupaladmin</td>
<td>published</td>
<td>edit</td>
</tr>
<tr>
<td>drupaladmin new</td>
<td>usermod</td>
<td>drupaladmin</td>
<td>published</td>
<td>edit</td>
</tr>
<tr>
<td>sw5557</td>
<td>usermod</td>
<td>sw5557</td>
<td>published</td>
<td>edit</td>
</tr>
<tr>
<td>My Life</td>
<td>biography</td>
<td>sw5557</td>
<td>published</td>
<td>edit</td>
</tr>
</tbody>
</table>

- Back to top
Administrator Main Menu Screen

- **Content management**: Manage your site's content.
  - Categories: Create vocabularies and terms to categorize your content.
  - Comments: List and edit site comments and the comment moderation queue.
  - Content: View, edit, and delete your site's content.
  - Content types: Manage fields by content type, including default settings, front page promotion, etc.
  - Node hierarchy: Manage relations between content types.
- **Post settings**: Control posting behavior, such as teaser length, requiring previews before posting, and the number of posts on the front page.
- **RSS publishing**: Configure the number of items per feed and whether feeds should be in title/teaser/full-text.
- **User management**: Manage your site's users, groups and access to site features.
  - Access control: Determine access to features by selecting permissions for roles.
  - Access rules: List and create rules to disallow usernames, e-mail addresses, and IP addresses.
- **Site building**: Control how your site looks and feels.
  - Blocks: Configure what block content appears in your site's sidebars and other regions.
  - Menus: Control your site's navigation menu, primary links and secondary links, as well as rename and reorganize menu items.
  - Modules: Enable or disable modules not used for your site.
  - Themes: Change which theme your site uses or allows users to set.
- **Site configuration**: Adjust basic site configuration options.
  - Administration theme: Settings for how your administrative pages should look.
  - Clean URLs: Enable or disable clean URLs for your site.
  - Date and time: Settings for how Drupal displays date and time, as well as the system's default timezone.
  - Error reporting: Control how Drupal deals with errors including 403/404 errors as well as PHP error reporting.
  - File system: Tell Drupal where to store uploaded files and how they are accessed.
  - Image toolkit:
### Administrator Menus Screen

![Administrator Menus Screen](image)

### Menus

Menus are a collection of links (menu items) used to navigate a website. The list(s) below display the currently available menus along with their menu items. Select an operation from the list to manage each menu or menu item.

#### Primary links

- **Edit**
- **Delete**
- **Add Item**

<table>
<thead>
<tr>
<th>Menu Item</th>
<th>Expanded</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
<td></td>
<td>edit</td>
</tr>
<tr>
<td>My Profile</td>
<td></td>
<td>edit</td>
</tr>
<tr>
<td>User list</td>
<td>No</td>
<td>edit</td>
</tr>
<tr>
<td>Contact us</td>
<td></td>
<td>edit</td>
</tr>
</tbody>
</table>

#### Navigation

- **Add Item**

<table>
<thead>
<tr>
<th>Menu Item</th>
<th>Expanded</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create page (enabled)</td>
<td></td>
<td>enable</td>
</tr>
<tr>
<td>Content (disabled)</td>
<td></td>
<td>enable</td>
</tr>
<tr>
<td>- Create content</td>
<td>No</td>
<td>edit</td>
</tr>
<tr>
<td>- Biography (disabled)</td>
<td></td>
<td>enable</td>
</tr>
<tr>
<td>- Story</td>
<td></td>
<td>edit</td>
</tr>
<tr>
<td>Search (disabled)</td>
<td></td>
<td>enable</td>
</tr>
<tr>
<td>Search content (disabled)</td>
<td></td>
<td>enable</td>
</tr>
<tr>
<td>Search users</td>
<td></td>
<td>edit</td>
</tr>
</tbody>
</table>
Blocks are boxes of content that may be rendered into certain regions of your website, for example, into sidebars. They are usually generated automatically by modules, but administrators can create blocks manually.

Only enabled blocks are shown. You can position blocks by specifying which area of the page they should appear in (e.g., a sidebar). Highlighted labels on this page show the regions into which blocks can be rendered. You can specify where within a region a block will appear by adjusting its weight.

If you want certain blocks to disable themselves temporarily during high server loads, check the "Throttle" box. You can configure the auto-throttle on the throttle configuration page after having enabled the throttle module.

You can configure the behaviour of each block (for example, specifying on which pages and for what users it will appear) by clicking the "configure" link for each block.

<table>
<thead>
<tr>
<th>Block</th>
<th>Region</th>
<th>Weight</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>header</td>
<td>0</td>
<td>configure</td>
</tr>
<tr>
<td>Left sidebar</td>
<td>left sidebar</td>
<td>0</td>
<td>configure</td>
</tr>
<tr>
<td>Right sidebar</td>
<td>right sidebar</td>
<td>0</td>
<td>configure</td>
</tr>
<tr>
<td>User login</td>
<td></td>
<td></td>
<td>configure</td>
</tr>
<tr>
<td>Disabled</td>
<td></td>
<td></td>
<td>configure</td>
</tr>
</tbody>
</table>

- **Search form**
- **Navigation**
- **Primary links**
- **Recent comments**
Administrator Recent Events Screen

The image shows a screenshot of the Siena College Alumni Spotlight System. The screen displays recent log entries for the system. The entries are categorized by type, date, and message. The log entries include actions such as session opening, closing, and page updates, as well as user interactions and system status updates.

Log entries are displayed in a table format with columns for Type, Date/Time, Message, User, and Operations. Each entry provides a brief description of the action taken within the system.
Administrator Settings Screen

Site configuration

- Administration theme: Settings for how your administrative pages should look.
- Clean URLs: Enable or disable clean URLs for your site.
- Date and time: Settings for how Drupal displays date and time, as well as the system's default timezone.
- Error reporting: Control how Drupal deals with errors including 403/404 errors as well as PHP error reporting.
- File system: Tell Drupal where to store uploaded files and how they are accessed.
- Image just: Choose which image toolkit to use if you have installed optional toolkits.
- Input formats: Configure how content input by users is filtered, including allowed HTML tags, PHP code tags. Also allows enabling of module-provided filters.
- Performance: Enable or disable page caching for anonymous users, and enable or disable CSS preprocessor.
- Search settings: Configure relevance settings for search and other indexing options.
- Site information: Change basic site information, such as the site name, slogan, e-mail address, mission, front page and more.
- Site maintenance: Take the site offline for maintenance or bring it back online.

Back to top
Administrator Modules Screen

Modules are plugins for Drupal that extend its core functionality. Here you can select which modules are enabled. Click on the name of the module in the navigation menu for their individual configuration pages. Once a module is enabled, new permissions might be made available. Modules can automatically be temporarily disabled to reduce server load when your site becomes extremely busy by enabling the throttle module and checking throttle. The auto-throttle functionality must be enabled on the throttle configuration page after having enabled the throttle module.

It is important that update.php is run every time a module is updated to a newer version.

You can find all administration tasks belonging to a particular module on the administration by module page.
**Administrator Users Screen**

Drupal allows users to register, login, log-out, maintain user profiles, etc. Users of the site may not use their own names to post content until they have signed up for a user account.

### Update options

Unlock the selected users

<table>
<thead>
<tr>
<th>Username</th>
<th>Status</th>
<th>Roles</th>
<th>Member since</th>
<th>Last access</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>5455454</td>
<td>active</td>
<td>2 weeks 1 day</td>
<td>2 weeks 1 day ago</td>
<td>edit</td>
<td></td>
</tr>
<tr>
<td>sx665454</td>
<td>active</td>
<td>2 weeks 1 day</td>
<td>2 weeks 1 day ago</td>
<td>edit</td>
<td></td>
</tr>
<tr>
<td>dx555555</td>
<td>active</td>
<td>2 weeks 3 days</td>
<td>5 min 6 sec ago</td>
<td>edit</td>
<td></td>
</tr>
<tr>
<td>sx55555</td>
<td>active</td>
<td>admin user</td>
<td>3 weeks 10 hours</td>
<td>edit</td>
<td></td>
</tr>
</tbody>
</table>

Back to top
Browse User List Screen

Siena College | Alumni Spotlight System

Home

user list

swn5557
Microsoft
CEO
Albany

Microsoft totally rocks

2007
Computer Science
Math
Web Design
Dr. Breiter
Algorithms
None

It was totally nifty

Anthony Angelucci
swn5557@siena.edu
135 Barbra Lane
555-555-5555
I love software engineering

drip@electr

Chris McConnell
dt140mca@siena.edu

c254dt

2007
Computer Science
Charles Fettes
c254dt@siena.edu
Comments Screen
## Edit Profile Screen

### Siena College | Alumni Spotlight System

#### Account Information

**Username:**
- saa5557

Your preferred username; punctuation is not allowed except for periods, hyphens, and underscores.

**Email Address:**
- saa5557@siena.edu

A valid email address. All emails from the system will be sent to this address. The email address is not made public and will only be used if you wish to receive a new password or if emails need to be sent to a different email address on your behalf.

**Password:**

Confirm password:

To change the current user password, enter the new password in both fields.

### Picture

**Upload Picture:**
- [Image]

Your virtual file or picture. Maximum dimensions are 800x800 and the maximum size is 400 kb.

### Comment Settings

**Signature:**

---

Initrode Solutions      Detailed Design – Alumni Spotlight Web System      Page 14
Welcome to the Computer Science Alumni Spotlight Web System.

We at Siena College would like to thank you for taking time to create your user account here with us. As you know from your experience here at Siena, we are a college that is focused on creating a community, not just an educational institution. Your help is greatly appreciated in this matter, as we hope to "Spotlight" users from this system to show prospective students what a degree in Computer Science can do for them.

From all of us here in the Computer Science Department, we would like to thank you once again.

-- Chris McConnell
-- Head Developer | Siena College Alumni Spotlight Web System

Back to top
Forgotten Password Screen

Siena College | Alumni Spotlight System

User account  Create new account  Log in  Request new password

Username or e-mail address:

Email new password

Back to top
Profile Page Screen
Profile Page Screen Continued

Major
Computer Science

Minor
Math

Favorite Class
Web Design

Favorite Teacher
Dr. Breiter

Least Favorite Class
Algorithms

Graduate School
None

Sienna Experience
It was totally nifty

Personal Information

Full Name
Anthony Angiulli

E-mail Address
sam5557@sienna.edu

Home Address
13 Barbara Lane

Phone Number
555-555-5555

Favorite Quote
I love software engineering

History

Member for
2 weeks 3 days

Back to top
Registration Screen

Please note: All users who are registering with this system are willing to allow Siena College the rights to display certain specified fields.

**Career**
- **Company**: Enter the company you are currently employed with.
- **Job Title**: Enter your current job title.
- **Location**: Enter the location of your company/job. (State, City as an example)
- **Salary**: Enter your current salary. Note: This field is used for Siena College purposes only, and will not be viewable by other site members.

**Career Experience**
Enter your experience with your job(s). Perhaps good jobs, bad jobs, how you found yours, etc.

**Account Information**
- **Username**: Your preferred username; punctuation is not allowed except for periods, hyphens, and underscores.
Registration Screen Continued
### Registration Screen Continued

![Registration Screen](image)

**College**

- **Graduation Year:** Enter the year you graduated from Sierra.
- **Major:** Enter your major while at Sierra College.
- **Minor:** Enter any minor(s) you had while at Sierra College.
- **Favorite Class:** Enter your favorite class while at Sierra.
- **Favorite Teacher:** Enter your favorite teacher while at Sierra.
- **Least Favorite Class:** Enter your least favorite class at Sierra.
- **Graduate School:** Enter the name of the school you attended or are attending.
- **Sierra Experience:** Talk about your college experience at Sierra College.

**Personal Information**

- **Full Name:** Enter your full name here.
- **E-mail Address:** Enter your email address here.
- **Home Address:**
Search Users Screen
Search Yielding No Results Screen

Your search yielded no results
- Check if your spelling is correct.
- Remove quotes around phrases to match each word individually: "blue smurf" will match less than blue smurf.
- Consider loosening your query with OR: blue OR smurf will match less than blue OR smurf.

Back to top
Search Yielding Results Screen
Search Site Screen
Welcome Screen

Welcome to the Computer Science Alumni Spotlight Web System.

We at Siena College would like to thank you for taking time to create your user account here with us. As you know from your experience here at Siena, we are a college that is focused on creating a community, not just an educational institution. Your help is greatly appreciated in this matter, as we hope to “Spotlight” users from this system to show prospective students what a degree in Computer Science can do for them.

From all of us here in the Computer Science Department, we would like to thank you once again.

- Chris McConnell
--- Head Developer | Siena College Alumni Spotlight Web System

Back to top
User command summary

Administrator Home Page Screen

The administrative user can locate this screen after he/she has successfully logged into the Alumni Spotlight Web System. This page simply notifies the administrator that they have successfully logged into the system.

Administrator Access Control Screen

This screen can be located from the Administrative Control Panel screen by clicking on the "Access Control" option located as a sub category of "administer" on the left hand menu bar. This screen breaks down every aspect of the site and gives the administrator the option to allow or disallows each option for each user type. For example, community users can create profiles and web blogs but the “anyone” user cannot.

Administrator Content Screen

This screen can be located from the Administrative Control Panel screen by clicking on the "Content" option located as a sub category of "administer" on the left hand menu bar. From this location, the administrator can edit different types of content that will be available such as photo albums, blogs and forms.

Administrator Main Menu Screen

This is the main screen where the administrative user will be able to make changes to different modules, themes, roles and other issues concerned with the site. From here, the administrative user will be able to make all site changes and manage the users as well.

Administrator Menus Screen

This screen can be located from the Administrative Control Panel screen by clicking on the "Menus" option located as a sub category of "administer" on the left hand menu bar. This screen allows the administrator to enable or disable certain accessories that are associated with the website. These accessories can be disabled completely or selectively. If a feature is disabled completely, no user can use that feature. If the feature is disabled selectively, a particular user's blog entry may be removed or that user's individual ability to use the blog feature can be removed.
**Administrator Menu Building Screen**

In this segment of the administrative control panel, the admin can add, edit and delete links to certain menus, add new menus, or move/edit/delete current menus based upon the designated blocks, or created boxes. To get to this area, the admin will click to “administer” then “site building” then “menus.”

**Administrator Recent Events Screen**

This screen can be accessed by clicking on the "Administer" option on the left hand menu bar. This screen is another version of the Administrator - Logs screen. This screen allows the administrator to view the most recent occurrences of the system. This screen differs only slightly from the Logs screen in that it holds events for a shorter amount of time and it only displays the most recent events.

**Administrator Settings Screen**

This screen can be located from the Administrative Control Panel screen by clicking on the "Settings" option located as a sub category of "administer" on the left hand menu bar. This screen allows the administrator to set general configuration settings for the site. Specifically the administrator can change general settings, error handling settings, file system settings, and can perform site maintenance and string handling. General settings include basic fields and options for the website and database. Error handling takes care of different errors that may arise during the use of the website. This will allow the administrator to access and change portions of the code if necessary. File system settings are simply in regards to the uploading of user photos. Site maintenance includes the theme and appearance of the site as a whole and string handling deals with checking to ensure that input that goes into the database is acceptable.

**Administrator Modules Screen**

This screen can be located from the Administrative Control Panel screen by clicking on the "Modules" option located as a sub category of "administer" on the left hand menu bar. Modules are plug-ins for Drupal that extends its core functionality. This screen contains a list of all available modules as well as descriptions of each. From this screen, the administrator can chose which modules are enabled. Once a module is enabled, new permissions might be made available. Modules can be automatically temporarily disabled to reduce server load when the site becomes extremely busy. This can be done by checking the "throttle" check box instead of the "Enabled" check box.

**Administrator Users Screen**

This screen can be located from the Administrative Control Panel screen by clicking on the "Users" option located as a sub category of "administer" on the
left hand menu. This screen gives the administrator a list of all users that are registered within the system including how long they have been members, and the date and time of their last visit. Clicking on the "edit" option to the right of each statement will allow the administrator to edit the content of that user’s profile.

**Browse User List Screen**

This screen is viewable to all registered users. To get to this screen, the user can click on “user list.” This will return a list of all the users that have been registered with the system, including their profile picture, and some information from their profile for easy viewing/browsing of the alumni in our system.

**Comments Screen**

This is a test module to possibly be added to the user(s)’ profile(s). We are attempting to allow users to “comment” on each other’s profiles, leaving messages for them to see. This module is currently buggy, and does not always work; therefore the designated path has not been created. This will load occasionally in the correct manner within the users’ profile page.

**Edit Profile Screen**

The purpose of this page is to allow the user to change any information in their profile. This page will be accessed from the user’s Home/My Account screen. This screen contains information about the user, only some of which is required. The required fields are labeled with an asterisk. Upon submission, if no errors are found, the user will be redirected to their Home/My Account screen where the changes to their profile will be updated. If errors are found, the screen will be reloaded with a reason for the error.

**Contact Us Screen**

This screen will be accessed when the user clicks the **Contact Us** link on their User Home Page Screen This feature is used in order to allow the user to ask the administrator a question or just leave the administrator a comment. When the user hits the **Submit** button, an e-mail will be sent to the administrator.

**User Home Page Screen**

The users will see this screen once they successfully sign in to the system. After the user first registers, the profile will be blank. This screen consists of facts about the user including their careers, personal information, and their college life. This screen will also contain picture of the user that he/she chooses. From this screen the user will have multiple menu options which can be activated or deactivated by the Administrative user.
Forgotten Password Screen

The purpose of this screen is to give the user the ability to recover a lost or forgotten password. This screen contains two fields, via default, but will contain only one for our purposes. The user must enter their registered email address, and we will send them a new password which will act as the registration process has, where they will be given a link, and can then change their password to something they will remember.

Profile Page Screen

The users will see this screen once they successfully sign in to the system. After the user first registers, the profile will be blank. This screen consists of facts about the user including their careers, personal information, and their college life. This screen will also contain picture of the user that he/she chooses. From this screen the user will have multiple menu options which can be activated or deactivated by the Administrative user.

Registration Screen

The purpose of this screen is to allow new users to join the community. This screen is where new users will register for an account. The user’s email address will serve as their user name, or account ID for the web system. The user is then given a temporary password, which will confirm their account, and an email will be sent with the user’s password. Upon successful submission, the user will be directed to a confirmation screen. If the submission is unsuccessful, the screen will be reloaded with a note stating the reason for the error. This screen will be accessed from the sign on screen and possibly from other locations undetermined at this time.

Search Users Screen

Using this screen, registered users will be able to search for other members in the community. The user will search for other users by entering any of the following information (regarding the user they are searching for): name, e-mail address, or graduation year. The more information the user enters, the more specific the results that are returned will be. This screen will be accessed by clicking on the Search link located on the users Home/My Account Screen.

Search Yielding No Results Screen

This is the screen that will be returned to a user who has done a search – both for users and content – and nothing has been found.

Search Yielding Results Screen
This is the screen that will be returned to a user who has done a search – both for users and content – and something has been found.

**Search Site Screen**

This screen can be found by clicking on the “search” button at the top of the site without entering any information in the text box. From this screen the user can select to search through the users, or the site’s content to find their string. There is also addition advanced options that the user can select from.

**Welcome Screen**

This is the main screen that anyone will see upon visiting the site and not being logged in. From here the person can read what the site is about, then choose to register, or log in if they have already registered with the system.
Detailed Data Flow Diagrams

A detailed data flow diagram is a graphical representation of the “flow” of data through some sort of computer aided system. This is a precursor to a Structure diagram, which shows/represents the data processing that goes on throughout the system. Data flows utilize three basic symbols, as shown below to represent the structure, movement and interaction among the users, data, processes, and data stores – both internal and external.

Below is a key for the basic symbols used for our specific Detailed Data Flow diagrams which follow the key.

Symbols Key

- **Processes**
- **Sources/Sinks**
- **Data Store**
- **Dataflows**
Context Diagram
Level 3 Diagram

Anyone (siena.edu)

Web Page

Page Request

3.1

Create page

Profile Information

Content Management System
**Logical Data Stores**

This is a set of tables that represent the different fields for the MySQL styled database. The first part of this is the textual description, followed by a graphical representation of the tables.

### Access

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>aid</td>
<td>tinyint(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>mask</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>tinyint(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Authmap

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>aid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>authname</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>module</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Blocks

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>module</td>
<td>varchar(64)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>delta</td>
<td>varchar(32)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>theme</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>tinyint(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>region</td>
<td>varchar(64)</td>
<td>Yes</td>
<td>left</td>
<td></td>
</tr>
<tr>
<td>custom</td>
<td>tinyint(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>throttle</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>visibility</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>pages</td>
<td>text</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Boxes

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>bid</td>
<td>tinyint(4)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>varchar(64)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Null</td>
<td>Default</td>
<td>Links to</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------</td>
<td>------</td>
<td>---------</td>
<td>-------------------</td>
</tr>
<tr>
<td>body</td>
<td>longtext</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>info</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>format</td>
<td>int(4)</td>
<td>Yes</td>
<td>0</td>
<td>filter_formats -&gt; format</td>
</tr>
</tbody>
</table>

**Cache**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>cid</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>data</td>
<td>longtext</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>expire</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>created</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>headers</td>
<td>text</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>

**Comments**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>cid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>pid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>comments -&gt; cid</td>
</tr>
<tr>
<td>nid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>node -&gt; nid</td>
</tr>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>subject</td>
<td>varchar(64)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>comment</td>
<td>longtext</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hostname</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>timestamp</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>score</td>
<td>mediumint(9)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>format</td>
<td>int(4)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>thread</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>users</td>
<td>longtext</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(60)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>mail</td>
<td>varchar(64)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>homepage</td>
<td>varchar(255)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>

**Contact**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>cid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>category</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>recipients</td>
<td>longtext</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>reply</td>
<td>longtext</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
### Files

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>fid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>nid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>node -&gt; nid</td>
</tr>
<tr>
<td>vid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>node_revisions -&gt; nid</td>
</tr>
<tr>
<td>filename</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>filepath</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>filemime</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>filesize</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>list</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Filter_Formats

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>int(4)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>roles</td>
<td>varchar(255)</td>
<td>Yes</td>
<td>role -&gt; rid</td>
<td></td>
</tr>
<tr>
<td>cache</td>
<td>tinyint(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Filters

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>int(4)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>module</td>
<td>varchar(64)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>delta</td>
<td>tinyint(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>tinyint(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Flood

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>event</td>
<td>varchar(64)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hostname</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>timestamp</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
### History

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>nid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>node_revisions -&gt; nid</td>
</tr>
<tr>
<td>timestamp</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Menu

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>mid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>menu -&gt; mid</td>
</tr>
<tr>
<td>pid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>path</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>tinyint(4)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>int(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Node

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>nid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>vid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>node_revisions -&gt; vid</td>
</tr>
<tr>
<td>type</td>
<td>varchar(32)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>status</td>
<td>int(4)</td>
<td>Yes</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>created</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>changed</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>comment</td>
<td>int(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>promote</td>
<td>int(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>moderate</td>
<td>int(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>sticky</td>
<td>int(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Node_Access

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>nid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>node -&gt; nid</td>
</tr>
<tr>
<td>gid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>realm</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>grant_view</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
grant_update  tinyint(1)  Yes  0
grant_delete  tinyint(1)  Yes  0

**Node_Comment_Statistics**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>nid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td>node -&gt; nid</td>
</tr>
<tr>
<td>last_comment_timestamp</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>last_comment_name</td>
<td>varchar(60)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>last_comment_uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>comment_count</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**Node_Counter**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>nid</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>totalcount</td>
<td>bigint(20)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>daycount</td>
<td>mediumint(8)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>timestamp</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**Permission**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>rid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>role -&gt; rid</td>
</tr>
<tr>
<td>perm</td>
<td>longtext</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>tid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**Profile_Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>fid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>varchar(255)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(128)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>explanation</td>
<td>text</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>category</td>
<td>varchar(255)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>page</td>
<td>varchar(255)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(128)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>required</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
### Profile_Values

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>fid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>profile_fields -&gt; fid</td>
</tr>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>value</td>
<td>text</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>

### Roles

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>rid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(32)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Search_Dataset

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>sid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(16)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>data</td>
<td>longtext</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Search_Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>varchar(50)</td>
<td>Yes</td>
<td></td>
<td>search_total -&gt; word</td>
</tr>
<tr>
<td>sid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(16)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>fromsid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>fromtype</td>
<td>varchar(16)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>score</td>
<td>float</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>

### Search_Total

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>varchar(50)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Sequences

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Sessions

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td></td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>sid</td>
<td>varchar(32)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hostname</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>timestamp</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>cache</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>session</td>
<td>longtext</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>

### System

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>filename</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>int(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>throttle</td>
<td>tinyint(1)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>bootstrap</td>
<td>int(2)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Term_Data

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>tid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>vid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>vocabulary -&gt; vid</td>
</tr>
<tr>
<td>name</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>longtext</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>tinyint(4)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>term_hierarchy</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Field | Type | Null | Default | Links to
--- | --- | --- | --- | ---
tid | int(10) | Yes | 0 | term_data -> tid
parent | int(10) | Yes | 0 | term_hierarchy -> tid

**Term_Node**

### Field | Type | Null | Default | Links to
--- | --- | --- | --- | ---
nid | int(10) | Yes | 0 | node -> nid
tid | int(10) | Yes | 0 | term_data -> tid

**Term_Relation**

### Field | Type | Null | Default | Links to
--- | --- | --- | --- | ---
tid1 | int(10) | Yes | 0 | term_data -> tid
tid2 | int(10) | Yes | 0 | term_data -> tid

**Term_Synonym**

### Field | Type | Null | Default | Links to
--- | --- | --- | --- | ---
tid | int(10) | Yes | 0 | term_data -> tid
name | varchar(255) | Yes | | |

**Url_Alias**

### Field | Type | Null | Default | Links to
--- | --- | --- | --- | ---
pid | int(10) | Yes | NULL | |
src | varchar(128) | Yes | | |
dst | varchar(128) | Yes | | |

**Users**

### Field | Type | Null | Default | Links to
--- | --- | --- | --- | ---
uid | int(10) | Yes | 0 | |
name | varchar(60) | Yes | | |
pass | varchar(32) | Yes | | |
mail | varchar(64) | Yes | | |
mode | tinyint(1) | Yes | 0 | |
sort | tinyint(1) | Yes | 0 | |
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>rid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users_roles -&gt; rid</td>
</tr>
</tbody>
</table>

### Variable

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>varchar(48)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>value</td>
<td>longtext</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Vocabulary

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>int(10)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>longtext</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>help</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>relations</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>hierarchy</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>multiple</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>required</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>tags</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>module</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>tinyint(4)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
### Vocabulary_Node_Types

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>vocabulary -&gt; vid</td>
</tr>
<tr>
<td>type</td>
<td>varchar(16)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Watchdog

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Default</th>
<th>Links to</th>
</tr>
</thead>
<tbody>
<tr>
<td>wid</td>
<td>int(5)</td>
<td>Yes</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>uid</td>
<td>int(10)</td>
<td>Yes</td>
<td>0</td>
<td>users -&gt; uid</td>
</tr>
<tr>
<td>type</td>
<td>varchar(16)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>message</td>
<td>longtext</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>severity</td>
<td>tinyint(3)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>link</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>location</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>referer</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hostname</td>
<td>varchar(128)</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>timestamp</td>
<td>int(11)</td>
<td>Yes</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Logical Data Dictionary

---

**Account Changes**

*Description:* Modifying the users account in the database.

*Location:* Level (1)

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify Account</td>
<td>Content Management System</td>
</tr>
</tbody>
</table>

Date Created: 11/13/2006  Date Modified: 11/13/2006

---

**Accounts Data**

*Description:* This data flow represents the users account information, such as type, fields, pictures and access.

*Location:* Level (0)

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Management System</td>
<td>Manage Community User Requests</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Management System</td>
<td>View Page</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Management System</td>
<td>Register</td>
</tr>
</tbody>
</table>

Date Created: 11/13/2006  Date Modified: 11/13/2006

---

**Account Login Info**

*Description:* This data flow represents a query call that returns a list of all user e-mail accounts, and passwords to check for correct log-in information from the user.

*Location:* Level (0)

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Management System</td>
<td>Manage Community User Requests</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Management System</td>
<td>View Page</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Management System</td>
<td>Verify Login</td>
</tr>
</tbody>
</table>

Date Created: 11/15/2006  Date Modified: 11/15/2006

---

**Account Request**

*Description:*
Prompts the database for the Community User’s profile.

**Location:**

Level (0)

Source: View Page  
Destination: Content Management System

Date Created: 11/13/2006     Date Modified: 11/13/2006

**Administrative User**  

**Description:**

This user has total control over the web system, including user accounts, the layouts, database, and modules that can be added or removed. For a more in-depth description see the use case for the Administrative user.

**Location:**

Context Diagram

Inflows:  
Error Page  
Profile Homepage

Outflows:  
Admin Login Information

Level (1)

Inflows:  
Admin Control Panel
Error Page
Changes
Profile Homepage

Outflows:  
Admin Login Information

Level (2)

Inflows:  
Admin Control Panel
Error Page
Changed Material

Outflows:  
Admin Login Information
Website Properties
Database Properties

Level (3)

Inflows:  
Admin Control Panel
Error Page
Profile Homepage
Removal Confirmation
Spotlight Confirmation

Outflows:  
Admin Login Information
Search Request
Ban Request
Spotlight Request

Date Created: 11/15/2006     Date Modified: 11/15/2006

-------------------------------------------------------------------------------------------------------------------
Admin Control Panel

Description:
The page containing the administrator's control options.

Location:
Level (0)
Source: Maintain Web System
Destination: Administrative User

Source: Manage Admin Request
Destination: Administrative User

Level (2)
Source: Verify Login
Destination: Administrative User

Level (3)
Source: Verify Login
Destination: Administrative User

Date Created: 11/13/2006     Date Modified: 11/13/2006

------------------------------------------------------------------------------------------------------------

Admin Login Information

Description:
The Administrators login name and password.

Location:
Context Diagram

Source: Administrative User
Destination: Alumni Spotlight Web System

Level (0)
Source: Administrative User
Destination: Maintain Web System

Level (0)
Source: Administrative User
Destination: Manage Admin Request

Level (2)
Source: Administrative User
Destination: Verify Login

Level (3)
Source: Administrative User
Destination: Verify Login

Date Created: 11/13/2006     Date Modified: 11/13/2006

------------------------------------------------------------------------------------------------------------

Admin Login Info

Description:
This data flow represents a query call that returns all of the administrative menus, control panel, log in information – such as password(s) – and then if verified is sent to the administrative user.

**Location:**

Level (0)

**Source:** Content Management System  
**Destination:** Maintain Web System

**Source:** Content Management System  
**Destination:** Manage Admin Request

Level (2)

**Source:** Content Management System  
**Destination:** Verify Login

Level (3)

**Source:** Content Management System  
**Destination:** Verify Login

Date Created: 11/13/2006  
Date Modified: 11/13/2006

---

**Admin Login Info Request**

**Description:**

The log in attempt by the administrative user, this data flow represents a call to the system for the Admin Login Info.

**Location:**

Level (0)

**Source:** Maintain Web System  
**Destination:** Content Management System

**Source:** Manage Admin Request  
**Destination:** Content Management System

Level (2)

**Source:** Verify Login  
**Destination:** Content Management System

Level (3)

**Source:** Verify Login  
**Destination:** Content Management System

Date Created: 11/15/2006  
Date Modified: 11/15/2006

---

**Admin Maintenance**

**Description:**

Any changes that an administrative user makes.
**Alumni Spotlight Web System (0)**

**Process**

This is the basic; overall process that represents the entire Alumni Spotlight Web System. This will be further broken down in each level of our Detailed Data Flow Diagrams, in order to describe this process/entity in greater detail.

**Inflow:** Registration Info, Login Info, E-mail Address, Admin Login Information, Page Request, Data

**Outflow:** Error Page, Profile Homepage, Spotlight Page, Login Data Request, Registration Data Request, Password Info Request, Admin Maintenance, Profile Request

---

**Anyone**

**Source/Sink**

This represents the “anyone” user, who has limited options in what they can do with the Alumni Spotlight Web System. For a greater description take a look at the use case scenarios below.

**Inflow:** Spotlight Page

**Outflow:** Page Request

---

**Level (0)**

**Inflow:** Spotlight Page
Ban Request

Description:
Administrator chooses to ban a user from the community.

Location:
Level (3)

Source: Administrative User
Destination: Ban User

Ban User (3.3)

Description:
This process is the action that is taken by the administrative user to remove, or ban a specific e-mail account and profile from the system. This is used to prevent “joke” profiles from being created and hurting the integrity of our system.

Location:
Level (3)

Inflows: Ban Request
Success/Error Message

Outflows: Removal Confirmation
Error Page
Remove Request

Initrode Solutions  Detailed Design – Alumni Spotlight Web System  Page 54
### Changed Material

**Description:**
This data flow represents the accepted or rejected changes that the administrative user has made. He will see the new changes instantly on his own screen, and can browse the site to see if his changes should be accepted or rejected.

**Location:**
- **Level (2)**
  - Source: Maintain Database
  - Destination: Administrative User
  - Source: Maintain Website
  - Destination: Administrative User

**Date Created:** 11/15/2006
**Date Modified:** 11/15/2006

### Changes

**Description:**
Changes made by any Community User.

**Location:**
- **Level (0)**
  - Source: Maintain Web System
  - Destination: Administrative User

**Date Created:** 11/13/2006
**Date Modified:** 11/13/2006

### Community User

**Description:**
This represents the “community user” who self registers with the system. This user has limited access, but owns an editable account. For a more detailed description view the use case scenarios below.

**Location:**
- Context Diagram
  - **Level (0)**
    - Inflows: Error Page
    - Outflows: E-mail Address
    - Login Info
    - Registration Info
  - **Level (1)**
    - Inflows: User Homepage
    - Registration E-mail
    - Password E-mail
    - Error Page
  - Outflows: Login Info
  - Registration Request
Password Email
Outflows: Login Info
Registration Request
Modification Request
Password Info

Date Created: 11/15/2006   Date Modified: 11/15/2006

---

**Confirmation**

*Description:*
Confirm the data changes that were made to the website or the database.

*Location:*
Level (2)

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Management System</td>
<td>Maintain Website</td>
</tr>
<tr>
<td>Content Management System</td>
<td>Maintain Database</td>
</tr>
</tbody>
</table>

Date Created: 11/13/2006   Date Modified: 11/13/2006

---

**Content Management System**

*Description:*
Open source system that will create and maintain MYSQL database and website.

*Location:*
Context Diagram

<table>
<thead>
<tr>
<th>Inflows</th>
<th>Outflows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Data Request</td>
<td>Data</td>
</tr>
<tr>
<td>Registration Data Request</td>
<td></td>
</tr>
<tr>
<td>Password Info Request</td>
<td></td>
</tr>
<tr>
<td>Admin Maintenance</td>
<td></td>
</tr>
<tr>
<td>Profile Request</td>
<td></td>
</tr>
</tbody>
</table>

Level (0)

<table>
<thead>
<tr>
<th>Inflows</th>
<th>Outflows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Data Request</td>
<td>Account Login Info</td>
</tr>
<tr>
<td>Registration Data Request</td>
<td>Accounts Data</td>
</tr>
<tr>
<td>Password Info Request</td>
<td>Password</td>
</tr>
<tr>
<td>Admin Login Info Request</td>
<td>Admin Login Info</td>
</tr>
<tr>
<td>Admin Maintenance</td>
<td>Success/Error Message</td>
</tr>
<tr>
<td>Profile Request</td>
<td>Profile</td>
</tr>
</tbody>
</table>

Initrode Solutions Detailed Design – Alumni Spotlight Web System Page 56
Level (1)

Inflows:  Login Data Request
          Registration Data Request
          Account Changes
          Password Info Request

Outflows: Account Login Info
          Accounts Data
          New Account Information
          Password

Level (2)

Inflows:  Admin Login Info Request
          Data Changes

Outflows: Admin Login Info
          Confirmation

Level (3)

Inflows:  Admin Login Info Request
          Profile Request
          Remove Request
          Generation Request

Outflows: Admin Login Info
          Profile
          Success/Error Message
          Profile Information

Date Created: 11/17/2006          Date Modified: 11/17/2006

------------------------------------------------------------------------------------------------------------

Data

Description:
  A data flow that represents all types of user account data, password information, request calls from both administrative, and community users. This data flow is then broken down further in the following diagrams to better represent what each user is looking for.

Location:
  Context Diagram
  Source:    Content Management System
  Destination: Alumni Spotlight Web System

Date Created: 11/15/2006          Date Modified: 11/15/2006

------------------------------------------------------------------------------------------------------------

Data Changes

Description:
  Any type of changes that the administrative user has made to either the website, or the database. Examples of such would be adding modules, or changing a CSS, or adding a new field to the database table.

Location:
  Level (2)
---

**Database Properties**

*Data Flow*

*Description:*
Changing the Database settings, such as adding a new table, changing the length on an instance in the database, or changing relationships – either adding or deleting.

*Location:*
Level (2)

*Source:*
Administrative User

*Destination:*
Maintain Database

---

**E-mail Address**

*Data Flow*

*Description:*
The E-mail address of the user.

*Location:*
Context Diagram

*Source:*
Community User

*Destination:*
Alumni Spotlight Web System

---

**Error Page**

*Data Flow*

*Description:*
The page that is sent to the user if he/she makes a mistake logging in.

*Location:*
Context Diagram

*Source:*
Alumni Spotlight Web System

*Destination:*
Community User

*Level (0)*

*Source:*
Manage Community User Requests

*Destination:*
Community User

*Source:*
Maintain Web System

*Destination:*
Administrative User
Level (1)

Source: Manage Admin Request
Destination: Administrative User

Source: Verify Login
Destination: Community User

Source: Register
Destination: Community User

Source: Retrieve Password
Destination: Community User

Level (2)

Source: Verify Login
Destination: Administrative User

Source: Maintain Website
Destination: Administrative User

Source: Maintain Database
Destination: Administrative User

Level (3)

Source: Verify Login
Destination: Administrative User

Source: View Profile
Destination: Administrative User

Source: Ban User
Destination: Administrative User

Source: Spotlight User
Destination: Administrative User

---

**Generation Request**

*Description:*
Web Server tells the database what profile information to compile.

*Location:*
Level (3)

Source: Spotlight User
Destination: Content Management System

Date Created: 11/15/2006   Date Modified: 11/15/2006
Login Data Request

Description:
Spotlight system requests list of usernames and passwords for verification of login.

Location:
Context Diagram
Source: Alumni Spotlight Web System
Destination: Content Management System

Level (0)
Source: Manage Community User Request
Destination: Content Management System

Level (1)
Source: Verify Login
Destination: Content Management System

Date Created: 11/15/2006     Date Modified: 11/15/2006

Login Info

Description:
The users E-mail address and password to logon to the system.

Location:
Context Diagram
Source: Community User
Destination: Alumni Spotlight Web System

Level (0)
Source: Community User
Destination: Manage Community User Requests

Level (1)
Source: Community User
Destination: Verify login

Date Created: 11/13/2006     Date Modified: 11/13/2006

Manage Admin Request (3)

Description:
This process represents all types of actions that the administrative user can take. This is a basic; overall process that will be broken down further in the following diagrams.

Location:
Level (0)
Inflow: Admin Login Information
Admin Login Info
Profile
Outflows: Admin Control Panel
Manage Community User Requests (1)

Description:
This process represents all types of actions that the community user can take. This is a basic; overall process that will be broken down further in the following diagrams.

Location:
Level (0)

Inflow:
- Login Info
- Registration Request
- Password Info
- Account/Login Info
- Accounts Data
- Password

Outflows:
- User Homepage
- Registration E-mail
- Password E-mail
- Error Page
- Login Data Request
- Registration Data Request
- Password Info Request

Maintain Database (2.3)

Description:
The process that the administrative user takes part in when he makes any type of modification to the database itself.

Location:
Level (2)

Inflows:
- Database Properties
- Confirmation

Outflows:
- Error Page
- Changed Material
- Data Changes

Maintain Website (2.2)

Description:
The process that the administrative user takes part in when he makes any type of modification to the website itself.
Maintain Web System (2)

Description:
This process represents all types of actions that the administrative user can take in relationship to the web system. This is a basic; overall process that will be broken down further in the following diagrams.

Inflow:
- Admin Login Information
- Admin Login Info
- Success/Error Message

Outflows:
- Admin Control Panel
- Error Page
- Changes
- Admin Login Info Request
- Admin Maintenance

Modification Request

Description:
User changes their profile.

Source: Community User
Destination: Modify Account

Modify Account (1.3)

Description:
A basic; overall process that represents the user’s ability to modify their account, either with changes/deletion/addition of information, photos, or blogs (if added).

Inflows:
- Modification Request
- New Account Information

Outflows:
- User Homepage
Account Changes

Date Created: 11/15/2006     Date Modified: 11/15/2006

------------------------------------------------------------------------------------------------------------

New Account Information

Description: Sends the edited account information back to the web server.

Location: Level (1)
Source: Content Management System
Destination: Modify Account

------------------------------------------------------------------------------------------------------------

Page Request

Description: The "anyone" user looks at the spotlighted profile on the Siena website.

Location: Context Diagram
Source: Anyone (siena.edu)
Destination: Alumni Spotlight Web System

Level (0)
Source: Anyone
Destination: View Page

------------------------------------------------------------------------------------------------------------

Password

Description: Holds the user’s password.

Location: Level (0)
Source: Content Management System
Destination: Manage Community User Requests

Level (1)
Source: Content Management System
Destination: Retrieve Password

------------------------------------------------------------------------------------------------------------

Password E-mail

Description: The e-mail that is sent to the user containing their current password.

Location: Level (0)
Source: Manage Community User Requests
Password Info

Description:
The password that the user types in.

Location:
Level (0)
Source: Community User
Destination: Manage Community User Requests

Level (1)
Source: Community User
Destination: Retrieve Password

Profile

Description:
The Community User’s profile.

Location:
Level (0)
Source: Content Management System
Destination: Manage Admin Request

Level (3)
Source: Content Management System
Destination: View Profile
Profile Homepage

**Description:**
The community users profile the administrator requests.

**Location:**

- **Context Diagram**
  - Source: Alumni Spotlight Web System
  - Destination: Administrative User

- **Level (0)**
  - Source: Manage Admin Request
  - Destination: Administrative User

- **Level (3)**
  - Source: View Profile
  - Destination: Administrative User

Profile Information

**Description:**
The spotlighted user’s information from their profile.

**Location:**

- **Level (3)**
  - Source: Content Management System
  - Destination: Spotlight User

Profile Request

**Description:**
Obtaining profile data from the database.

**Location:**

- **Context Diagram**
  - Source: Alumni Spotlight Web System
  - Destination: Content Management System

- **Level (0)**
  - Source: Manage Admin Request
  - Destination: Content Management System

- **Level (3)**
  - Source: View Profile
  - Destination: Content Management System
Receive Page

Description:
The "anyone" user receives the requested page.

Location:
Context Diagram
Source: Alumni Spotlight Web System
Destination: Anyone (siena.edu)

Date Created: 11/13/2006     Date Modified: 11/13/2006

Register (1.2)

Description:
The basic; overall process that represents a community user’s attempt at registering. The user will enter the necessary information to register their account with the system.

Location:
Level (1)
Inflows: Registration Request
Accounts Data
Outflows: Registration E-mail
Error Page
Registration Data Request

Date Created: 11/15/2006     Date Modified: 11/15/2006

Registration Data Request

Description:
This data flow represents a query call that asks the CMS for the list of current e-mail addresses to be compared to the current e-mail address the user is attempting to register with.

Location:
Context Diagram
Level (0)
Source: Manage Community User Requests
Destination: Content Management System
Level (1)
Source: Register
Destination: Content Management System

Date Created: 11/15/2006     Date Modified: 11/15/2006

Registration E-mail

Description:
Confirmation e-mail that the user receives when their registration is complete.

**Location:**
Level (0)
Source: Manage Community User Request
Destination: Community User

Level (1)
Source: Register
Destination: Community User

Date Created: 11/15/2006     Date Modified: 11/15/2006

---

**Registration Info**

**Description:**
An HTML form which the user fills out to complete the registration process.

**Location:**
Context Diagram
Source: Community User
Destination: Alumni Spotlight Web System

Date Created: 11/15/2006     Date Modified: 11/15/2006

---

**Registration Request**

**Description:**
User fills out a registration form in order to become a community member.

**Location:**
Level (0)
Source: Community User
Destination: Manage Community User Request

Level (1)
Source: Community User
Destination: Register

Date Created: 11/13/2006     Date Modified: 11/13/2006

---

**Remove Request**

**Description:**
The Web Server tells the database to remove user from the database.

**Location:**
Level (3)
Source: Ban User
Destination: Content Management System

Date Created: 11/15/2006     Date Modified: 11/15/2006

---

**Removal Confirmation**

**Description:**
A confirmation message received by the Administrative User informing him/her that the user was banned from the community.

*Location:* Level (3)
*Source:* Ban User
*Destination:* Administrative User

---

**Retrieve Password (1.4)**

*Description:* This process represents the ability for a community user to retrieve a password if they have forgotten what their password is.

*Location:* Level (1)
*Inflows:* Password Info
*Password*
*Outflows:* Password E-mail
>Error Page
>Password Info Request

---

**Search Request**

*Description:* The administrator searches for a Community User’s profile.

*Location:* Level (3)
*Source:* Administrative User
*Destination:* View Profile

---

**Spotlight Confirmation**

*Description:* A message received by the Administrative User confirming that the chosen profile was spotlighted.

*Location:* Level (3)
*Source:* Spotlight User
*Destination:* Administrative User

---

**Spotlight Page**

*Description:*
Appears on the Siena College homepage with a Community User’s profile featured.

**Location:**
Level (0)

**Source:** View Page

**Destination:** Anyone

Date Created: 11/15/2006     Date Modified: 11/15/2006

----------------------------------------

**Spotlight Request**

**Data Flow**

**Description:**
The Administrative User chooses a profile to spotlight on the Siena Webpage.

**Location:**
Level (3)

**Source:** Administrative User

**Destination:** Spotlight User

Date Created: 11/15/2006     Date Modified: 11/15/2006

----------------------------------------

**Spotlight User (3.4)**

**Process**

**Description:**
This process is what allows the administrative user to select a specific user profile to be “spotlighted” on the CS home page.

**Location:**
Level (3)

**Inflows:** Spotlight Request
Profile Information
Success/Error Message

**Outflows:** Spotlight Confirmation
Error Page
Generation Request

Date Created: 11/15/2006     Date Modified: 11/15/2006

----------------------------------------

**Success/Error Message**

**Data Flow**

**Description:**
A data flow that represents a simple confirmation or error message sent to the user if the changes made did not take by the CMS.

**Location:**
Level (0)

**Source:** Content Management System

**Destination:** Maintain Web System

Level (3)

**Source:** Content Management System

**Destination:** Ban User

Date Created: 11/15/2006     Date Modified: 11/15/2006
**User Homepage**

*Description:*
The user’s personalized homepage, which displays their profile information.

*Location:*
Level (0)
Source: Manage Community User Request
Destination: Community User

Level (1)
Source: Verify Login
Destination: Community User

Source: Modify Account
Destination: Community User

---

**Verify Login (1.1)**

*Description:*
The process in which the user’s entered e-mail address is compared to the list of e-mail addresses, and then the password that was entered is compared. If no matches are found, the login does not occur.

*Location:*
Level (1)

**Inflows:**
- Login Info
- Account Login Info

**Outflows:**
- User Homepage
- Error Page
- Account Login Info

Level (2)

**Inflows:**
- Admin Login Information
- Admin Login Info

**Outflows:**
- Admin Control Panel
- Error page
- Admin Login Info Request

Level (3)

**Inflows:**
- Admin Login Information
- Admin Login Info

**Outflows:**
- Admin Control Panel
- Error Page
- Admin Login Info Request
View Page

**Description:**
The process in which an “anyone” user goes to the Siena College Computer Science home page and is able to view the spotlighted alumni page.

**Location:**
Level (1)

**Inflows:**
- Page Request
- Accounts Data

**Outflows:**
- Spotlight Page
- Account Request

---

View Profile (3.2)

**Description:**
This process represents the ability for the administrative user to look at (view) specific profiles of community users registered within the system.

**Location:**
Level (3)

**Inflows:**
- Search Request
- Profile

**Outflows:**
- Profile Homepage
- Error Page
- Profile Request

---

Website Properties

**Description:**
Any changes that the administrative user might have changed, such as the CSS, or modules that are sent from the administrative user to the website, and then to the CMS.

**Location:**
Level (2)

**Source:**
Administrative User

**Destination:**
Maintain Website

---
Logical Format of Data Files and Databases

All database tables and relationships are being created by our team specifically for this project. None of our database information is already in existence or being derived from an external source such as our client. Further information regarding our database design can be found in the Detailed Design portion of this document.

II. Architectural Design Specification

Structure Diagrams

A data structure diagram, or structure diagram for short, is a data model that is used to describe other conceptual data models (the Detailed Data Flow Diagrams) by providing a hierarchical view of how all the users, processes, and data flow will occur. This sort of diagram also allows the viewer to see certain processes that are utilized by more than one member/user of the system if such an event occurs.

Alumni Spotlight Web System:
Structure Diagram
III. Testing Requirements

Requirements Inventory

Use Case Scenarios

The Alumni Spotlight Web System will include three different user cases, each of which are described below including check lists of specific needs for each.

Administrative User:

- This user will be the first user in the database, and first one created on the CMS.
- The user name will be “Administrator” and a password that we have yet to determine.
- This user will have complete access to all accounts.
  - The administrator will be able to see all registered accounts.
  - The administrator should be able to delete a user from the community.
  - The administrator should be able to edit a user’s account in the community.
- Will be able to modify the website as well as the database behind it.
  - Will have access to the MySQL database table to edit it.
  - Will be able to change the table settings in the database.
  - Will be able to change the style sheets for the website.
  - Will have the ability to change functions for the website.
- Administrative users will be able to select specific community members to be spotlighted on the Siena College Computer Science Home Page.
- The administrative user will be able to delete the account.
- No profile homepage instead has the control panel.
- Will not be able to change the log in name of “Administrator”

Community User:

- User will self register with the community
- The user will use his/her E-mail address as their User name (Log In).
- The password will consist of 6-12 characters, and must contain at least one number.
- If the user forgets their password, the user can retrieve it automatically via a link “Forgotten Password” which will e-mail the user their current password.
- The user will have their own account profile which will be customizable with personal information.
  - The following is a list of what will be included in a User’s profile:
- First Name – User’s first name.
- Last Name – User’s last name.
- City - Current city the user lives in.
- State – Current state the user lives in.
- E-mail – Current E-mail address the user is using.
- Graduation Year – What year the user graduated Siena.
- Major – What the user’s major was.
- Minor(s) – Any minor(s) the user had, if applicable.
- Degree Earned – The user’s current degree.
- Favorite Class – The user’s favorite class at Siena.
- Favorite Professor – The user’s favorite professor at Siena.
- Least Favorite Class – The user’s least favorite class at Siena.
- Graduate School – A Yes or No question if the user has gone to, or is currently attending graduate school.
- What School if applicable – The name of the school if the previous field was yes.
- Current Job Location – The user’s current company they work for.
- Current Job Title – The user’s current job title at work.
- Salary – The user’s current salary at their current job.
- Description on Siena Experience – A small description of what the user felt about their time spent at Siena college.
- Personal Information – Any other personal information the user might want to share such as family status.
- Favorite Quote – A quote the user might have found inspiring in their life or simply something they have heard.

- Will be able to upload a photo depicting them.
  - There will also be a sub part, containing photo albums.
  - The user can create, update, delete and add new photo albums.
  - The user can also update, delete, and add to old albums as well.
- Will have their own photo Album that can be updated.
- Will be able to search other community members that have registered for the Alumni Spotlight Web System.

**Anyone:**

- This user does not need to register for the community.
- Since the user is not registered, there is no account associated with the user on the community.
• Represents any user that visits the Siena College Computer Science Home page.
• Will only be able to view the current Alumni Spotlight that is active on the page above.

Functional Requirements Inventory

• The Alumni Spotlight Web System must be web based, so anyone can access it from anywhere.
• The system must work with all major Browsers, including but not limited to:
  o Internet Explorer 6.0 as well as 7.0
  o Mozilla Firefox 1.5.07
  o Netscape 7.1
  o Safari 2.0.4
• The system should be controlled by some sort of CMS to facilitate the administrator’s job.
• Allow an administrator to access, monitor, and maintain the web site.
  o Ability to review, accept, and deny initial profiles and changes made to those profiles later on.
  o Ability to select and use a profile for the Alumni Spotlight.
• Allow the administrator to access the database to maintain the user’s data, and make sure the system runs correctly.
• The administrator must be able to select specific community user(s) for the Alumni Spotlight.
• Allow the administrator to generate a page automatically for the Siena College Computer Science Home Page displaying the Alumni Spotlight.
• In accordance to the point above, this selection must generate the page to be loaded on the Siena College Computer Science Home Page.
• Allow secure sign in to the system via a sign on screen.
• According to the Use Case scenarios, we will ensure that each user type is correctly established.
• Allow self-registration into the system.
• Allow users to change/add new personal information to their profile in the following areas:
  ▪ First Name
  ▪ Last Name
  ▪ City
  ▪ State
  ▪ E-mail
  ▪ Graduation Year
  ▪ Major
  ▪ Minor(s)
  ▪ Degree Earned
  ▪ Favorite Class
• Favorite Professor
• Least Favorite Class
• Graduate School
• What School if applicable
• Current Job Location
• Current Job Title
• Salary
• Description on Siena Experience
• Personal Information
• Favorite Quote

• Finally, all page links will be tested and ensure that they redirect users to their appropriate link.

Welcome Screen

• Upon pressing the “Log On” button we will ensure that the user name and password is correct, and then redirects the user to their home page.
• We will ensure that the link for registration brings the user to the “Registration Screen.”
• Lastly, we will ensure that the link for a forgotten password works correctly as well, placing the user at the “Forgotten Password Screen.”

Registration Screen

• We will make sure that all indicated fields are filled in, and check the fields to ensure they can handle specific characters, such as spaces or single quotes.
• We will also make sure that upon submission, the account will be activated, and a confirmation email will be sent telling the user that their registration was a success. If the account exists already, we will redirect the user to a page informing them that there is an account with that email address already.

Profile Page Screen

• We will make sure the email account entered is contained in our database, and if it is, send the user their password.

Home / My Account Screen

• We will make sure the correct information loads with the specified profile.
• A check of all the links will be done, ensuring that they all lead to their correct, respective pages.
Edit Profile Screen

• This page will check to make sure all required fields have been filled in above all else.
• We will ensure specific characters such as spaces and single quotes are acceptable.
• Upon submission, we will make sure that each field is saved in the database, actually updating the specified profile.

Search Site Screen

• Upon pressing “Search” we will make sure the fields entered check the database, and return the user to the results screen, showing correct information according to the search criteria.

Browse User(s) Screen

• We will ensure that the profiles displayed here are correct in accordance with the search criteria.

Contact Us Screen

• We will ensure this screen will send an email to the systems administrator correctly including the user who sent it, as well as the comments that the user filled out within the comments area.

Admin Control Panel

• Due to the complexity of the control panel itself, we will summarize the basic operations that are held within the system. Since our open source CMS will create the basic functionality of the Admin Control Panel the basic things to check will be
  o That each link on the page works.
  o Any change made by the administrative user is reflected on the site, or data base depending upon which has been modified.
  o That the administrative user will be able to select a community user as a spotlighted member, creating a page for the Siena College Computer Science homepage.

Non-Functional Requirements Inventory

• The system should be fast, using fast servers and databases.
• This system should be easily adaptable, to enable Siena College to experience this community if desired.
Test Plan

Please note that the Test Plan has been included as a separate document. This document can be found on the web at the following address: http://oraserv.cs.siena.edu/%7Eperm_initrode/documents/project/TestPlan/TestPlanVer1.1.html

IV. Detailed Design Specification

Documentation Prologue for Each Routine

Community User

1.1 Login
   1. The user submits login information to the CMS
   2. The CMS verifies the login information
   3. If login information contains correct user name and password combination, the user is directed to his/her homepage. If the information is incorrect, the user receives an error page stating that the login attempt failed.

1.2 Registration
   1. The user clicks on the New User link
   2. The user fills out the information required to register
   3. The user may fill in the optional information or leave it unfilled
   4. The user clicks the register button
   5. The user is automatically logged in and directed to his/her homepage

1.3 Modifies Account

   1. The user modifies his/her account
      a. Login Information
      b. Profile Content
      c. Photo Gallery Content
   2. The changes are submitted to the CMS
   3. The changes are reflected immediately in the system

1.4 Forgotten Password
   1. The user clicks on the forgotten password link
   2. An e-mail is sent to the user’s registered e-mail account with a one-time use login link.
   3. The user changes his/her password

Anyone (siena.edu)
2.1 Sends Request to View the Spotlighted Page

1. The user views the page where the Alumni Spotlight is located
2. A request to load the Spotlighted page is sent to the CMS
3. The CMS returns the Spotlighted profile data to be viewed

Administrative User

3.1 Login

1. The user submits login information to the CMS.
2. The CMS verifies the login information
3. If login information contains correct user name and password combination, the user is directed to the administrative control panel. If the information is incorrect, the user receives an error page stating that the login attempt failed.

3.2 View Profile

1. The user submits information about a community user to the CMS.
2. The CMS searches the database for the community user.
3. If the information is found in any community users profile then the administrator will receive a list of the users whose profiles contain the searched information. If no profiles are found to contain the searched information then the user receives an error page saying no profiles found.
4. The user selects the profile and is directed to that users homepage.

3.3 Ban User

1. The user submits a request to delete a profile to the CMS.
2. The CMS deletes the user from the database.
3. If the profile is successfully deleted from the database the user will receive a confirmation page saying that the profile has been deleted successfully. If the profile is not deleted then the user will receive an error message stating that the profile has not been deleted.

3.4 Spotlight User

1. The user submits a spotlight request to the CMS.
2. The CMS gathers the proper information from the users profile.
3. The CMS copies the information to the Computer Science Departments Homepage.
4. If the spotlight is correct then the spotlighted users information will be shown on the Computer Science Departments Homepage and the user will receive a message saying the operation was completed successfully. If the spotlight does not work the user will receive an error message stating the spotlight was unsuccessful.
Pseudocode for Each Routine

The Alumni Spotlight Web System uses the Drupal Content Management System to provide the majority of its features and functions. The following is the logic that determines the course of action that is being taken for each routine.

Community User

1.1 Login

if (login(user name, password) = = TRUE)
    display(user’s my account page);
else display(error page);

1.2 Registration

// User clicks New User link
boolean required_information_missing = true;
display_registration_page();
while( required_information_missing ) {
    // User enters information
    // User clicks register button
    if( all required fields entered )
        required_information_missing = false;
    else display_registration_page(missing_fields);

1.3 Modifies Account

submit_changes( changes made to account );
update_database();
display_page( current_page );

1.4 Forgotten Password

registered_user_email = lookup_email( user_name );
send_mail( registered_user_email );

Anyone (siena.edu)

2.1 Sends Request to View The Spotlighted Page

display_spotlight( );

Administrative User
3.1 Login

    If (login(name, password) = TRUE)
        display(administrator control panel);
    else
        display(error page);

3.2 View Profile

    If (Search(user) = TRUE)
        display(users homepage) ;
    else
        display(error page);

3.3 Ban User

    If (Search(user) = TRUE)
        delete(users profile);
        display(success message);
    else
        display(error page);

3.4 Spotlight User

    copy_profile_to_spotlight( selected_alumni_user );
Physical data structures and data file specifications

Access

This table deals with the rules governing the use of email addresses, user names and originating hostnames.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>aid</td>
<td>Number</td>
</tr>
<tr>
<td>mask</td>
<td>Text</td>
</tr>
<tr>
<td>type</td>
<td>Text</td>
</tr>
<tr>
<td>status</td>
<td>Number</td>
</tr>
</tbody>
</table>

Authmap

This table stores information about loading user data from external sources.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>aid</td>
<td>Number</td>
</tr>
<tr>
<td>uid</td>
<td>Number</td>
</tr>
<tr>
<td>authname</td>
<td>Text</td>
</tr>
<tr>
<td>module</td>
<td>Text</td>
</tr>
</tbody>
</table>
**Blocks Roles**

This table holds the assignments between blocks and their permissions among themes and users.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>module</td>
<td>Text</td>
</tr>
<tr>
<td>delta</td>
<td>Text</td>
</tr>
<tr>
<td>rid</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Blocks**

This table holds the block configuration settings for specific themes. This includes things such as the status, region, and weight.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>module</td>
<td>Text</td>
</tr>
<tr>
<td>delta</td>
<td>Text</td>
</tr>
<tr>
<td>theme</td>
<td>Text</td>
</tr>
<tr>
<td>status</td>
<td>Number</td>
</tr>
<tr>
<td>weight</td>
<td>Number</td>
</tr>
<tr>
<td>region</td>
<td>Text</td>
</tr>
<tr>
<td>custom</td>
<td>Number</td>
</tr>
<tr>
<td>throttle</td>
<td>Number</td>
</tr>
<tr>
<td>visibility</td>
<td>Number</td>
</tr>
<tr>
<td>pages</td>
<td>Memo</td>
</tr>
<tr>
<td>title</td>
<td>Text</td>
</tr>
</tbody>
</table>
Boxes

This table holds user defined custom blocks, such as a block in the “Blocks” table, but is created by the administrative user so is held separately and can be placed inside of a block.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>bid</td>
<td>Number</td>
</tr>
<tr>
<td>body</td>
<td>Memo</td>
</tr>
<tr>
<td>info</td>
<td>Text</td>
</tr>
<tr>
<td>format</td>
<td>Number</td>
</tr>
</tbody>
</table>

Cache

This table stores data that may be cached to speed up page loading, especially for anonymous visitors.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>cid</td>
<td>Text</td>
</tr>
<tr>
<td>data</td>
<td>OLE Object</td>
</tr>
<tr>
<td>expire</td>
<td>Number</td>
</tr>
<tr>
<td>created</td>
<td>Number</td>
</tr>
<tr>
<td>headers</td>
<td>Memo</td>
</tr>
</tbody>
</table>

Cache Filter

The use of this table is yet to be determined.
**Cache Page**

This table allows specified page data to be cached to speed up page loading, especially for anonymous visitors.

![Cache Page Table]

**Cache Menu**

This table allows specified menu data may be cached to speed up page loading, especially for anonymous visitors.

![Cache Menu Table]
Comments

This table holds the comment(s) that have been added to any type of node, such as a person leaving a comment on a user’s profile.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>cid</td>
<td>Number</td>
</tr>
<tr>
<td>pid</td>
<td>Number</td>
</tr>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
<tr>
<td>uid</td>
<td>Number</td>
</tr>
<tr>
<td>subject</td>
<td>Text</td>
</tr>
<tr>
<td>comment</td>
<td>Memo</td>
</tr>
<tr>
<td>hostname</td>
<td>Text</td>
</tr>
<tr>
<td>timestamp</td>
<td>Number</td>
</tr>
<tr>
<td>score</td>
<td>Number</td>
</tr>
<tr>
<td>status</td>
<td>Number</td>
</tr>
<tr>
<td>format</td>
<td>Number</td>
</tr>
<tr>
<td>thread</td>
<td>Text</td>
</tr>
<tr>
<td>users</td>
<td>Memo</td>
</tr>
<tr>
<td>name</td>
<td>Text</td>
</tr>
<tr>
<td>mail</td>
<td>Text</td>
</tr>
<tr>
<td>homepage</td>
<td>Text</td>
</tr>
</tbody>
</table>

Content Type Bio

This table holds two ID numbers, one which is a ID for the table itself, and another which is used as a relation to the actual content to be held in the “bio” page that has been created.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>Number</td>
</tr>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
</tbody>
</table>
**Content Type Story**

This table holds two ID numbers, one which is a ID for the table itself, and another which is used as a relation to the actual content to be held in the “story” page that has been created.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>Number</td>
</tr>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Content Type Page**

This table holds two ID numbers, one which is a ID for the table itself, and another which is used as a relation to the actual content to be held in the page that has been created.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>Number</td>
</tr>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Content Type User Node**

This table holds two ID numbers, one which is a ID for the table itself, and another which is used as a relation to the actual content to be held in the “user node” page that has been created. The user node is considered a page that holds the user’s profile which they have filled out, and can edit.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>Number</td>
</tr>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
</tbody>
</table>
**File Revisions**

This table holds details about file revisions to the system.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fid</td>
<td>Number</td>
</tr>
<tr>
<td>vid</td>
<td>Number</td>
</tr>
<tr>
<td>description</td>
<td>Text</td>
</tr>
<tr>
<td>list</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Files**

This table holds details of any files uploaded by the users of the system. This table also holds information such as title, the name, the path, and the type of the file.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fid</td>
<td>Number</td>
</tr>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
<tr>
<td>filename</td>
<td>Text</td>
</tr>
<tr>
<td>filepath</td>
<td>Text</td>
</tr>
<tr>
<td>filemime</td>
<td>Text</td>
</tr>
<tr>
<td>filesize</td>
<td>Number</td>
</tr>
</tbody>
</table>
Filter Format

This table defines filter formats like "PHP code" and “Filtered "HTML.” This table will take a user’s entered data, and decipher it depending upon the specified style of entry. A user could write a page using PHP, or HTML – both filtered and full. This table links directly to the “filters” table.

Filters

This is the table that the actual filters use for each “filter format” and their order of operation.

Flood

This table is used for tracking how often the contact page (and other resources) are used.
History

This table maintains a record of page (node) visits per user.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>uid</td>
<td>Number</td>
</tr>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
<tr>
<td>timestamp</td>
<td>Number</td>
</tr>
</tbody>
</table>

Menu

This table holds the menu(s) data including the parent, path, title, description and destination of the menu link.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>mid</td>
<td>Number</td>
</tr>
<tr>
<td>pid</td>
<td>Number</td>
</tr>
<tr>
<td>path</td>
<td>Text</td>
</tr>
<tr>
<td>title</td>
<td>Text</td>
</tr>
<tr>
<td>description</td>
<td>Text</td>
</tr>
<tr>
<td>weight</td>
<td>Number</td>
</tr>
<tr>
<td>type</td>
<td>Number</td>
</tr>
</tbody>
</table>
**Node**

This table stores the basic node information such as title, type, status, user id, created on date(s), when something was modified, or is “sticky.” This is also the central hub for all the node based tables.

![Node Table](image)

**Node Access**

This is the table where any module that needs access to the node table(s) can register their interest in gaining access to the tables. Works like a queue, first in, and first out.

![Node Access Table](image)
**Node Comment Statistics**

This table stores special information about each of the node’s latest comment, and how many comments there are.

![Node Comment Statistics Table](image)

**Node Counter**

This table holds the basic statistics of about how often a node has been accessed.

![Node Counter Table](image)

**Node Field**

This table acts as a temporary holding place for fields that are related to the user node/node page(s) that once created is moved to the permanent place in the node table when combined with a node_field_instance.

![Node Field Table](image)
**Node Field Instance**

This table acts as a temporary holding place for the type of a field that is related to the user node/node page(s) that once created is connected with a node_field entry to be moved into the node table.

![Node Field Instance Table]

**Node Group**

This table acts as a holding place for a group name, type, and weight that is then linked with the node_group_fields.

![Node Group Table]

**Node Group Fields**

This table contains sields that are linked to a node_group which are accessed by the user node itself.

![Node Group Fields Table]
**Node Revisions**

This table is where the node teaser and body is stored along with previous versions of the node if applicable by the administrative user.

<table>
<thead>
<tr>
<th>node_revisions: Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Name</td>
</tr>
<tr>
<td>nid</td>
</tr>
<tr>
<td>vid</td>
</tr>
<tr>
<td>uid</td>
</tr>
<tr>
<td>title</td>
</tr>
<tr>
<td>body</td>
</tr>
<tr>
<td>teaser</td>
</tr>
<tr>
<td>log</td>
</tr>
<tr>
<td>timestamp</td>
</tr>
<tr>
<td>format</td>
</tr>
</tbody>
</table>

**Node Type**

This is the table which holds the different nodes, descriptions, types, and the module which the node is from.

<table>
<thead>
<tr>
<th>node_type: Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Name</td>
</tr>
<tr>
<td>type</td>
</tr>
<tr>
<td>name</td>
</tr>
<tr>
<td>module</td>
</tr>
<tr>
<td>description</td>
</tr>
<tr>
<td>help</td>
</tr>
<tr>
<td>has_title</td>
</tr>
<tr>
<td>title_label</td>
</tr>
<tr>
<td>has_body</td>
</tr>
<tr>
<td>body_label</td>
</tr>
<tr>
<td>min_word_count</td>
</tr>
<tr>
<td>custom</td>
</tr>
<tr>
<td>modified</td>
</tr>
<tr>
<td>locked</td>
</tr>
<tr>
<td>orig_type</td>
</tr>
</tbody>
</table>
**Node Family**

This table contains two numbers that establish the child of the current node, as well as the parent to be able to link from one node to another in a linked list type of organization.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>parent_nid</td>
<td>Number</td>
</tr>
<tr>
<td>child_nid</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Permission**

This table assigns each role permissions like “access content”, “post comments”.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>rid</td>
<td>Number</td>
</tr>
<tr>
<td>perm</td>
<td>Memo</td>
</tr>
<tr>
<td>tid</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Watchdog**

This table deals with modules that can record events here. These events range from quite normal activities to serious problems.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>wid</td>
<td>Number</td>
</tr>
<tr>
<td>uid</td>
<td>Number</td>
</tr>
<tr>
<td>type</td>
<td>Text</td>
</tr>
<tr>
<td>message</td>
<td>Memo</td>
</tr>
<tr>
<td>severity</td>
<td>Number</td>
</tr>
<tr>
<td>link</td>
<td>Text</td>
</tr>
<tr>
<td>location</td>
<td>Memo</td>
</tr>
<tr>
<td>referer</td>
<td>Text</td>
</tr>
<tr>
<td>hostname</td>
<td>Text</td>
</tr>
<tr>
<td>timestamp</td>
<td>Number</td>
</tr>
</tbody>
</table>
**Profile Values**

This table stores user data associated with custom fields.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fid</td>
<td>Number</td>
</tr>
<tr>
<td>uid</td>
<td>Number</td>
</tr>
<tr>
<td>value</td>
<td>Memo</td>
</tr>
</tbody>
</table>

**Profile Fields**

This table stores custom fields that have been set up with the profile module.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fid</td>
<td>Number</td>
</tr>
<tr>
<td>title</td>
<td>Text</td>
</tr>
<tr>
<td>name</td>
<td>Text</td>
</tr>
<tr>
<td>explanation</td>
<td>Memo</td>
</tr>
<tr>
<td>category</td>
<td>Text</td>
</tr>
<tr>
<td>page</td>
<td>Text</td>
</tr>
<tr>
<td>type</td>
<td>Text</td>
</tr>
<tr>
<td>weight</td>
<td>Number</td>
</tr>
<tr>
<td>required</td>
<td>Number</td>
</tr>
<tr>
<td>register</td>
<td>Number</td>
</tr>
<tr>
<td>visibility</td>
<td>Number</td>
</tr>
<tr>
<td>autocomplete</td>
<td>Number</td>
</tr>
<tr>
<td>options</td>
<td>Memo</td>
</tr>
</tbody>
</table>
**Roles**

This table is where the roles are defined. Roles are user groups, like ’Admin’ and ‘Editors’.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>rid</td>
<td>Number</td>
</tr>
<tr>
<td>name</td>
<td>Text</td>
</tr>
</tbody>
</table>

**Search Dataset**

Each row of this table is the simplified content of a node – or any other type of data object.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>sid</td>
<td>Number</td>
</tr>
<tr>
<td>type</td>
<td>Text</td>
</tr>
<tr>
<td>data</td>
<td>Memo</td>
</tr>
</tbody>
</table>

**Search Total**

This table contains a unique list of words from search_index, with frequency statistics.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>Text</td>
</tr>
<tr>
<td>count</td>
<td>Number</td>
</tr>
</tbody>
</table>
**Search Index**

This table contains an index of every word in the content in the search_dataset table.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>Text</td>
</tr>
<tr>
<td>sid</td>
<td>Number</td>
</tr>
<tr>
<td>type</td>
<td>Text</td>
</tr>
<tr>
<td>fromsid</td>
<td>Number</td>
</tr>
<tr>
<td>fromtype</td>
<td>Text</td>
</tr>
<tr>
<td>score</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Sequences**

This table records the latest ID(s) for various tables in the database. This is mainly for compatibility with older MySQL versions.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Text</td>
</tr>
<tr>
<td>id</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Sessions**

This table stores data related to the user's visit to the site.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>uid</td>
<td>Number</td>
</tr>
<tr>
<td>sid</td>
<td>Text</td>
</tr>
<tr>
<td>hostname</td>
<td>Text</td>
</tr>
<tr>
<td>timestamp</td>
<td>Number</td>
</tr>
<tr>
<td>cache</td>
<td>Number</td>
</tr>
<tr>
<td>session</td>
<td>Memo</td>
</tr>
</tbody>
</table>
**System**

This table stores data about a list of modules and themes. Holds information such as name, path, status among other items.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>filename</td>
<td>Text</td>
</tr>
<tr>
<td>name</td>
<td>Text</td>
</tr>
<tr>
<td>type</td>
<td>Text</td>
</tr>
<tr>
<td>description</td>
<td>Text</td>
</tr>
<tr>
<td>status</td>
<td>Number</td>
</tr>
<tr>
<td>throttle</td>
<td>Number</td>
</tr>
<tr>
<td>bootstrap</td>
<td>Number</td>
</tr>
<tr>
<td>schema_version</td>
<td>Number</td>
</tr>
<tr>
<td>weight</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Term Node**

This table links the terms from the term_data table to each node in the system.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>nid</td>
<td>Number</td>
</tr>
<tr>
<td>tid</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Term Hierarchy**

This table stores data regarding vocabulary hierarchies which are achieved by defining parent-child relationships between terms.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>tid</td>
<td>Number</td>
</tr>
<tr>
<td>parent</td>
<td>Number</td>
</tr>
</tbody>
</table>
**Term Data**

This table stores term data. A term is a label that can be applied to nodes. This table is where the terms are defined. This is also the central table of all the “term_*” tables. All the term tables are connected from through this main table.

![Term Data Table](image)

**Term Relation**

This table stores any terms that have a relationship can be linked to eachother. These links are defined here.

![Term Relation Table](image)

**Term Synonym**

This table defines alternatives to existing terms. Eg. "bike" might be given a synonym of "bicycle".

![Term Synonym Table](image)
**URL alias**

This table holds the alias, and actual links for user-defined, as well as module-defined alternative URLs.

![URL alias table](image)

**Usernode**

This table holds two numbers, one to hold a key for the database table itself, and one to link the user node id with the user node content.

![Usernode table](image)

**User Roles**

This table holds a list of assignments between users and roles.

![User Roles table](image)
**Variable**

This table is a flexible table for storing site settings.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Text</td>
</tr>
<tr>
<td>value</td>
<td>Memo</td>
</tr>
</tbody>
</table>

**Vocabulary**

This table stores a vocabulary which is list or group of related terms.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>Number</td>
</tr>
<tr>
<td>name</td>
<td>Text</td>
</tr>
<tr>
<td>description</td>
<td>Memo</td>
</tr>
<tr>
<td>help</td>
<td>Text</td>
</tr>
<tr>
<td>relations</td>
<td>Number</td>
</tr>
<tr>
<td>hierarchy</td>
<td>Number</td>
</tr>
<tr>
<td>multiple</td>
<td>Number</td>
</tr>
<tr>
<td>required</td>
<td>Number</td>
</tr>
<tr>
<td>tags</td>
<td>Number</td>
</tr>
<tr>
<td>module</td>
<td>Text</td>
</tr>
<tr>
<td>weight</td>
<td>Number</td>
</tr>
</tbody>
</table>

**Vocabulary Node Types**

This table will allow you to choose which vocabularies can be used with various node types. Thus when you create a new node, you will only have certain vocabulary terms to choose from.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>vid</td>
<td>Number</td>
</tr>
<tr>
<td>type</td>
<td>Text</td>
</tr>
</tbody>
</table>
Packaging specifications

Initrode Solutions will install all the necessary files and features of the Alumni Spotlight Web System on the Siena Computer Science network, Ares. There will then be a web address supplied for users to access our application.
Glossary of Terms

Cascading Style Sheets (CSS) – These are used to describe a specific style for a web page or set of pages, a website.

Code - The symbolic arrangement of data or instructions in a computer program or the set of such instructions.

Content Management System (CMS) – Web based management system that will allow the team and webmaster to edit, modify, and remove profiles and web pages.

Data Flow Diagram – A representation of the functional decomposition of a system.

Database - A collection of data arranged for ease and speed of search and retrieval.

Drupal – An open-source content management system that is versatile and allows the user to personalize a network of profiles. There are several different themes and colors for visual effects. Drupal comes with the ability to handle self-registration pages, blogs, and easy file uploading. It allows the profiles to be organized into separate categories and to set permissions for the profiles. Lastly, Drupal has the ability to track changes made to the user profiles. The administrator will be able to view all of these changes

Extensible Hypertext Markup Language (XHTML) – has same capabilities of HTML but is more restricted to allow for conformity among multiple platforms.

Gantt chart - A chart that depicts progress in relation to time, often used in planning and tracking a project.

Hypertext Markup Language (HTML) – Predominant markup language used for the creation of webpages.

Integration Test - the process in Software Engineering when individual modules are combined and tested as a group.

Internet - An interconnected system of networks that connects computers around the world via the TCP/IP protocol.

Internet Explorer (IE) – Microsoft based program used for browsing the internet.

Linear Sequential Model / Classic Waterfall Model – A systematic, sequential approach to software development that begins at the system level and progresses through analysis, design, coding, testing, and support.
**Markup Languages** – Combines a set of text, along with extra information about this text, such as formatting.

**MySQL** – A Multithreaded, multi-user SQL Database Management System.

**PHP Hypertext Preprocessor (PHP)** – An open source (free) programming language that is used for dynamic pages and is a high-level scripting language.

**Portal** – a specific kind of Content Management System Web site, password protected to allow Site Administrators to edit text, images and other content as and when necessary. This allows easy updating of the website content without the need to learn programming code.

**Profiles** – User created accounts that will be customizable according to each specific person.

**Pseudocode** - A notation resembling a programming language but not intended for actual compilation. It usually combines some of the structure of a programming language with an informal natural-language description of the computations to be carried out.

**Regression Testing** - testing that is meant to uncover errors that may occur when one feature of an application is changed.

**Transmission Control Protocol/Internet Protocol (TCP/IP)** - A communications protocol for computer networks, the main protocol for the Internet.

**Unit Test** - a procedure used to validate that individual modules or units of source code are working properly.

**Web log** - an online journal comprised of links and postings in reverse chronological order, meaning the most recent posting appears at the top of the page. Also known as a blog.
Development Timeline – Gantt Chart

<table>
<thead>
<tr>
<th>ID</th>
<th>Task Name</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Team Meetings</td>
<td>71 days</td>
</tr>
<tr>
<td>17</td>
<td>Client Meetings</td>
<td>71 days</td>
</tr>
<tr>
<td>33</td>
<td>Detailed Design</td>
<td>30 days</td>
</tr>
<tr>
<td>34</td>
<td>Detail Design Presentation</td>
<td>1 day</td>
</tr>
<tr>
<td>35</td>
<td>Acceptance Test</td>
<td>41 days</td>
</tr>
<tr>
<td>36</td>
<td>Acceptance Test Presentation</td>
<td>1 day</td>
</tr>
<tr>
<td>37</td>
<td>Academic Celebration</td>
<td>1 day</td>
</tr>
</tbody>
</table>

Project: Spring Gantt Chart
Date: Fri 2/13/07

<table>
<thead>
<tr>
<th>Task</th>
<th>Milestone</th>
<th>External Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Split</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Progress</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:
- Task
- Milestone
- External Tasks
- Project Summary
- Deadline