

Project IV - System Development

CISC3140, Fall 2011

PART 1 - System Development

Project can be found at <http://146.245.250.215/~achikin/project/index.php>
Full source can be downloaded at <http://146.245.250.215/~achikin/project.tgz>
Database dump: <http://146.245.250.215/~achikin/achikin.sql>

PART 2 - System Demonstration

Presentation performed in class on Thursday, 8th December.

PART 3 - Quality Control

Administrator interface:

Test script 1 - access admin interface (success):

0. Log in to website under **administrator user**
1. Click on “ADMIN” menu item in main menu
2. The page header should say BigNightOut “Administrator View” and content should show **complete user list** with pages

Test script 1 - access admin interface (failure):

0. Log in to website under **regular user** (or skip log in)
1. Attempt to go to administrator by using `./admin/index.php` url (<http://146.245.250.215/~achikin/project/admin/index.php>)
2. The page header will say BigNightOut “Administrator View” and content will say “**Access Denied**”

Test script 2 - modify user access level (success):

0. In admin interface (users section) enter **username** or **name** in search box (“Ben”)
1. Click “Search”
2. Page now only displays users with name “Ben”
3. Click on this user username (“Balcomb9”)
4. Page displays user details that can be modified by administrator
5. Select rank from dropdown list (was “admin” - select “moderator”)
6. Click “Save changes”
7. Header will say “**Changes saved**” and new details will indicate new rank

Test script 2 - modify user access level (failure):

0. In admin interface (users section) enter your **own username** or **user with higher privileges** username in search box (“Alexey”)
1. Click “Search”
2. Page now displays only your profile in list
3. Click on your profile username
4. Page will give an error “**Cannot moderate this user's profile profile.**”

Test script 3 - delete user (success):

0. Search for **user with lower privileges** as in test 2a
1. Check “Delete user” checkbox in profile details
2. Press “Save changes”
3. Website will ask to confirm deletion
4. Press “Delete permanently”
5. Message should say “**User deleted successfully.**“

PART 4 - Programmer Documentation

All website components are loaded through index.php in root folder, except for admin interface which has its own index.php located in /admin/ folder. 'index.php' includes required system files from /include/ folder such as header, footer and helper.php by 'require_once' method. 'index.php' determines which “view” to load from /view/ folder by \$_GET parameter extracted from URL (loads default.php if none selected or unexistent selected).

User authorization is processed by /include/user.php which is included in all cases. It uses \$_SESSION and Cookies to determine whether the user is already logged in and loads user profile to \$user global variable, which can be accessed from all places (example <?php echo \$user[“username”];?>)

Views themselves load main section of website (content), they might require additional function and load more files from /include/. That way /view/user.php loads /include/profile.php and uses functions from there to process profile form and save changes to profile.

/css/ folder contains CSS stylesheet template.css which sets all styles for website (administrator interface has its own css folder) and fonts folder which contains fonts

/js/ folder contains javascripts which are used to make website dynamic, such as highlighting, data verification, AJAX requests and banners change.

More advanced information represented on the next page in tree form (most files covered).

File structure tree:

/admin/	administrator interface root
- css/	administrator stylesheet and fonts
- images/	administrator background and images
- include/	administrator functions and system files
- view/	administrator views
- index.php	administrator main php file

/css/	styles
- fonts/	font files
- templace.css	main stylesheet

/images/	images, icons, banners
- avatars/	user profile pictures
- banners/	banners
- icons/	various icons
- bg.jpg	main background

/include/	system files containing functions to:
- comment.php	access comments with AJAX
- footer.php	display footer
- header.php	display header
- helper.php	prepare most views such as DB (Database) class
- profile.php	display/edit user profile view
- subscribe.php	access subscriptions with AJAX
- user.php	authenticate user from \$_SESSION or Cookies, prepare \$user data

/js/	javascript files used to:
- attend.js	activate Attend button AJAX call
- banner.js	change banner picture every 20 seconds
- comment.js	post new comment and reload comments display with AJAX
- subscribe.js	activate Subscribe button AJAX call
- verify.js	verify various fields format (“numbers only” or “price”, “email”, etc.)

/view/	php files that determine content display
- create.php	create event view
- default.php	default (home) view - default event search by IP/zip
- event.php	event view - event details, attend event, Google Map API
- forgot.php	forgot password form submission, reset password form
- search.php	search view - search by keyword/zip or advanced search + results
- user.php	user profile view - register/login or user details + edit user details

/xml/	xml folder contains xml files which are required due to php.ini limits
- eventful.xml	xml downloaded from eventful.com by API

index.php	main php file
-----------	---------------

PART 5 - Team Assessment

Alexey (“project manager”):

- ✓ Design system core files (functions and classes) and folders, main menu
- ✓ Assign and Collect group members code; integrate it into main project on homepage
- ✓ Provide examples of usage of core functions
- ✓ Find and prepare graphics/icons
- ✓ Eventful.com API
- ✓ Facebook 'Like' button API
- ✓ Implement following pages:
 - home (default search),
 - create (create event form),
 - administrator profile
- ✓ Take part in implementing: Administrator user list (with Ben), AJAX requests, pagination
- ✓ Take part in DB design

Ben

- ✓ Provide a lot of ideas on how to enhance user experience
- ✓ Implement and design following pages:
 - search, advanced search
- ✓ Take part in implementing: Admin user list (with Alexey), search pagination (with Alexey), event page features
- ✓ Take part in DB design

Chris

- ✓ Javascript verification scripts
- ✓ Captcha verification
- ✓ Mootools date javascript plugin integration
- ✓ Implement and design following pages:
 - user (preview/edit),
 - register (registration form/action),
- ✓ Take part in implementing: Administrator user list (with Ben)
- ✓ Take part in DB design

Roger

- ✓ Google maps API
- ✓ Implement and design following pages:
 - event (details/preview/edit)
- ✓ Take part in implementing: AJAX Attending button (with Alexey)
- ✓ Take part in DB design