# Milestones

# Phase 0: Ongoing (5%)

## Weekly timesheet

### Use provided template

### Weekly push to GitHub; available to instructors via URL in Moodle notes

# Phase 1: Brainstorming (2.5%)

## Due beginning of the 2nd week

## Peer review

## A clear presentation; as if you are presenting to a client

## Five minute presentation and five minutes for feedback

## Must be in class to present and receive/offer feedback

## Deliverables:

### On time attendance

### Presentation with a visual aid showing the user experience

### Audience participation

# Phase 2: Planning (10%)

## Due beginning of the 4th week

## Project description (project\_description.docx)

### Use the following headings:

#### Clients

##### Who are the clients? (people paying for site development)

#### Users

##### Who are the users? (general public, registered users, possibly the clients)

#### Problem/Need

##### What problem or need does your web application address?

#### General Functionality

##### What is the general functionality?

##### What will each level of user be able to do?

### Refrain from describing technology (JavaScript will be used to…, PHP/MySQL will be used to…)

### Submitted via Moodle file upload

## Domain name and live hosting

### One link that links to a page on your domain.

### One link that links to the whois information.

### Submitted as a Moodle note with Project Description file upload

## Database

### Minimum of 4 tables (though you will likely have MANY more)

### Database optimization

#### Normalized tables

#### Indexes

#### Appropriate data types

### Database diagram

#### Hand drawn or a document similar to Access’ Relationships window.

##### If hand drawn, scan as **erd\_phase2.jpg**

##### If created with software, export as a JPEG with the file name **erd\_phase2.jpg**

#### Must indicate optimization items listed above

## Deliverables:

### project\_description.docx

### erd\_phase2.jpg

### two links

# Phase 3: Functioning Production Environment (5%)

## Due beginning of the 6th week

## Generated Database

### SQL dump (upload via Moodle, database\_phase3.sql)

## Database connection PHP include file (upload via Moodle, database\_connection.php)

## Proof that PHP and the database are working

### A PHP page on the server that displays content from the database. The database can contain data that is hard coded. (Post link in Moodle note with dump and connection file uploads)

## Deliverables:

### database\_phase3.sql

### database\_connection.php

### one link

# Phase 4: Demonstration Checkpoint (10%)

## Due during the 9th week of the term

### Start usability testing during 8th week

## Single multi-user login

### Single login that determines the user level and redirects appropriately.

#### Form validation

##### Must have both client (HTML5 and/or JavaScript) and server side validation

##### Username and password required

## Some dynamic data displayed

### Personalized logout link

### User-specific data (three possible views)

## Page framework complete

### All pages are built though not necessarily with their content

### 100% functional navigation

### Layout scaffolding is complete (shows complete design)

#### CSS

#### Colors

#### Non-essential graphics

## Students demonstrate live to instructor

## Usability test

### Application is tested by at least three other students

### Student has tested at least two other student’s application

### Submit usability form (provided by instructor) in class

## Deliverables:

### Demonstration of login

### Demonstration of dynamic data

### Demonstration of user-level specific dynamic data

### Demonstration of complete framework

### Completed usability form(s)

# Phase 5: Final Usability Testing (2.5%)

## Due middle of the 12th week

## Second round of usability testing

### Application is tested by at least three other students

### Student has tested at least two other student’s application

### Submit usability form (provided by instructor) in class

## Deliverables:

### Completed usability form(s)

# Phase 6: Final Submission (50%)

## Both a live demo and localhost demo must be available

### You will present using the live server

### Write the following – largely and clearly – on the board:

#### URL

#### Member username and password

#### Admin username and password

### Be sure to have a localhost version available as a backup in case the Internet is not available or you are experiencing live server problems

### You will have 5 minutes before your presentation to write the information on the board and get yourself ready or – if necessary – to install/configure the localhost demo. Practice this! Make sure your SQL script works and inserts the required user accounts and content.

### You will have 15 minutes to present. Do not talk about technology! Just demonstrate the general public, member, and admin views/functionality.

### There will be a 10 minute Q and A period after you present where reviewers will likely ask technical questions.

### Your entire presentation will last 30 minutes including set up, presentation, and questions.

## Expo Presentation on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Deliverables due on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Final Presentation on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Style guide (styleGuide.pdf)

### Typography

### Border treatments

#### Images

#### Content boxes

### Color scheme and usage

## Media sources documentation (mediaSources.docx or .pdf)

### Word or PDF document

#### Thumbnail images with URLs

## Database

### Database optimization

#### Normalized tables

#### Indexes (not limited to primary/foreign keys)

#### Appropriate data types

### Database diagram (ERD, erd.jpg)

#### Hand drawn or a document similar to Access’ Relationships window.

##### If hand-drawn, scan as erd.jpg

#### Must indicate indexes, relationships, and data types

### SQL script file (to generate tables and default content, database\_final.sql)

#### Must include at least one admin account and one registered user account

#### Must include all initial content

#### Store the file as .sql with comment at the top that indicates the included logon credentials.

## Security

### Prevent SQL injection

### Prevent XSS

### Encrypted passwords

### CAPTCHA (choose one that is usable! Not impossible/frustrating for users.)

### Protected member/admin pages

## Usability

### Fully responsive

### Consistent navigation

### Intuitive navigation

### Intuitively usable

### Current page marker (page heading, navigation)

### Form validation throughout (client and server side)

#### Validation errors display next to problem fields (not as a disjointed block)

### Sticky forms (client and server side)

### Login/personalized logout on every page

### No horizontal scroll

### Works in versions of browsers installed at A-B Tech

### Custom 404 pages

## Search feature

### Some content is searchable

#### Users

#### Content

#### Prices

#### Dates

#### Etc…

## Code functionality and best practices

### HTML validates (for HTML in your control)

### CSS validates (for CSS in your control)

### Runs without errors

### Appropriate modularity

## Code documentation (codeDocumentation.pdf)

### PHP

#### Consistent formatting and appropriate whitespace for readability

#### Comment block at the top of each file

##### Author

##### Revision date

##### File name

##### Description

#### Function Library

##### Each function has a comment block

###### Function name

###### Argument list

###### Return data (if any)

###### Description

###### Function dependency (if applicable)

### CSS

#### Consistent formatting and appropriate whitespace for readability

#### Organization of rules

#### Minimal code bloat

#### Minimal classitis/divitis

#### Shorthand where appropriate

#### Non-essential images as backgrounds

### JS

#### Consistent formatting and appropriate whitespace for readability

#### Appropriate abstraction

#### Reference the javascript framework

##### List the framework/plugins

##### No need to print out the js library

### All code (PHP/CSS/HTML/JS/SQL dump…) is pushed to GitHub and available to instructors/reviewers

## Files to Submit via Moodle

### Last-First-WEB-289-2015SP (zipped)

#### assets (folder)

##### index.php (or index.htm) – contains links to the following, opening each in a new window/tab:

###### directoryWireframe.txt (tree /f/a)

###### Original unedited artwork (folder)

###### erd.jpg

###### database\_final.sql

###### styleGuide.pdf

###### mediaSources.docx

###### codeDocumentation.pdf with Table of Contents (bookmark links)

#### css (folder)

##### style sheets

#### images (folder)

##### optimized essential/non-essential images

#### js (folder, optional)

##### optional javascript

#### optional folders (admin, login, lib, functions …)

#### php files

## Presentation/demonstration

### 5 minutes to set up localhost if needed and write live server URL and credentials on board

### 15 minute presentation, using localhost (reviewers use live demo)

### 10 minutes for questions and answers

### Show all three views

#### Walk through each of these views to demonstrate the user experience

### All assets are available to reviewers as a URL from your domain (http://yourdomain.com/assets/)

### What part did you find the most challenging?

### What would you do differently/add in version 2.0?

### What aspect did you enjoy the most?