# Task Two – Online Events Listing Application (Networks Task)

Introduction

Online events listings are typically 'database driven'. Building such an application will require a blend of HTML, CSS, PHP, MySQL and JavaScript. This task is worth 70% of the module.

Task

To design, build and test an events listing style web site.

The task breaks down into the following:

* Database Design and Modelling - design, structure and populate a MySQL backend database to hold the event information.
* HTML/CSS Front End - design a suitably attractive site front end using HTML5 and CSS. Extra marks are available for designs that adhere to concept of 'progressive enhancement' and those that are mobile friendly through the adoption of techniques such as responsive web design.
* Javascript - use JavaScript to enhance the design with animation and interaction through techniques such as AJAX. Extra marks are available for authoring Javascript as opposed to the use of simple plugins. This authored Javascript may include the use of popular Javascript frameworks and APIs such as jQuery and Google Maps.
* PHP Code - integrate the backend database using PHP to provide users with event information. Provide options such as filtering by date, genre and venue. Marks will be allocated for work that illustrates secure database techniques.
* Content Management System - build a bespoke CMS to allow the site owner to add, amend and delete event information. This should be password protected.

You are free to use JavaScript frameworks and plugins to enhance your design. All frameworks and plugins used must be clearly credited.

*Submission and Marking*

Upload a zip file containing all your files to the assignment handler on Blackboard by 9pm on 22 April 2016. Please also include a working URL. Marking will be by walkthrough.

Marking Scheme

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Task*** | ***Mark*** | ***40-49 (3rd)*** | ***50-59 (2:2)*** | ***60-69 (2:1)*** | ***70+ (1st)*** |
| HTML and CSS frontend design including site navigation | 20 | Simple page design to work in browser of student's choice. Basic navigation system. Use of at least 15 CSS properties to style the site. | Site tested in more than one browser of the student's choice. Consistent 'look and feel' between pages. Use of 20 plus CSS properties to style the site. | Well-presented and structured HTML and CSS. Browser support across IE11, Chrome, Safari and Firefox. Sensible sematic HTML structure. Intuitive navigation system. Use of 20 plus CSS properties including a range of CSS3 properties. Evidence of consideration of HTML/CSS to improve the user experience. | Browser support across IE8+, Chrome, Safari and Firefox. Use of media queries for responsive web design. Excellent UX delivered to a professional level. |
| MySQL Database Design | 20 | Basic one table database of at least 30 records. Use of at least 5 fields. | Basic one table database with appropriate data types. Data to include date/time stamps. | Correct use of data types. Sensible interrelated multi table database design. Some attempt to store login / user credentials. | Demonstrate understanding of database security. CMS login details stored securely. |
| Use of Javascript | 20 | At least one notable example of Javascript functionality added –for example a banner image slider. | At least three notable examples of Javascript functionality added – for example - date pickers, banner sliders, form validation as well as other appropriate UI enhancements such as tabs, accordions, modals. | Well structured and organized use of Javascript functionality. Evidence of at least one originally authored piece of Javascript. | Evidence of two or more originally authored and/or more advanced Javascript functionality. Examples may include use of AJAX to query data, interaction with third party APIs such as Google Maps. |
| Use of PHP for Dynamic functionality - searches, filters, categories | 20 | Simple event table style listings. | Data searchable by at least one search criteria. | Multiple search criteria and filters. Searchable data. Data categorised. Securely implemented SQL queries. | Use of AJAX queries. Result ordering and pagination. Advanced functionality such as a booking or reservation system. |
| Content Management System | 20 | Basic functionality only. Indicative example - the ability to update event information. | Extended functionality. Indicative examples - the ability to add, edit and delete event. CMS password protected. | Further functionality. Category management, enhanced CMS password security with credentials held in the database. | Extensive backend functionality available. Indicative examples - image uploading, user registration with email, homepage editing, hashed login credentials. |