

CSE2ICX

Internet Client Engineering

Assessment 3 – Server-Side Web Programming

Assessment Objectives

- Apply the principles of Web user interface design in analysing and building Web sites.
- Construct Web sites using a combination of DHTML technologies and simple server-side scripting.

This is an INDIVIDUAL assignment. Students are not permitted to work in a group when writing this assignment.

Copying, Plagiarism

This is an individual assignment. Students are not permitted to work in a group when writing this assignment. Plagiarism is the submission of another person's work in a manner that gives the impression that the work is their own. La Trobe University treats plagiarism seriously. When detected, penalties are strictly imposed.

Further information can be found on <http://www.latrobe.edu.au/students/academic-integrity/explanation/plagiarism>

Submission Guidelines

- Your assignment submission should be typed, not written/drawn by hand.
- Submit the electronic copy of your assignment through the subject LMS.
- Submission after the deadline will incur a penalty of 5% of the available marks for that task per day capped at 5 days. No assignment will be accepted after 5 days. If you have encountered difficulties that lead to late submission or no submission, you should apply for special consideration.

Contents

CSE2ICX Internet Client Engineering	1
Assessment 3 – Server-Side Web Programming	1
Assessment Objectives	1
Copying, Plagiarism	1
Submission Guidelines	1
Background	3
Task 1: Home (index.aspx)	3
Task 2: Edit record (edit.aspx)	3
Task 3: Invoice (invoice.aspx and print.aspx)	3
Task 4: Stocktake (stocktake.aspx).....	5
Task 5: Chart (chart.aspx).....	6
Task 6: Organisation	6
Task 7: Layout.....	7
Task 8: Navigation	7
Task 9: Presentation	7
Task 10: Code.....	8
Submission	8
Assessment marking criteria	8

Background

You have been approached to create two **conference** websites for a conference in **Sydney**. Your choice will be the same choice you made as per your Assignment 1 and 2 works. You must create the websites to be as realistic as possible for a normal organisation.

The organisation has requested two websites: A public website with information about the organisation and an internal website that members of the organisation will use for day-to-day management. For **assignment 3**, you are now tasked to create the **internal** website.

This assignment is part of a running scenario for all assignments in this subject, where the Sydney conference will continue to be your client. Please also refresh your memory regarding the additional background information provided previously in Assignments 1 and 2.

The tasks of this assignment are organised as follows. Tasks 1-5 first describe the functional requirements of the website, regarding what each of the web pages will do. Then Tasks 6-10 describe the non-functional requirements regarding the behaviour of all parts of your solution. Note that some of the non-functional requirements have similarities to Assignment 2, so please read these carefully in particular.

Task 1: Home (index.aspx)

The home page is the entry point for the website. As this is the first page that will be viewed by the staff, it should be designed in an attractive manner.

The home page includes a GridView control that has a list of registrations. The data will be taken from a XML data file and populated in the grid via the data binding technique. The fields will be for the following data: id, full name, email address, registration type, attending social event (yes/no). Note that the value of the id field must always be unique. A sample data file with dummy data can be downloaded from where you obtained this assignment document. You are welcome to change the dummy data to sensible data, provided that you maintain a minimum of 4 records.

The home page also includes a Label control with a one-sentence summary about the following: number of registrants attending the social event. Your data file should therefore have a mixture of records to make this output meaningful.

Task 2: Edit record (edit.aspx)

This page is a form that allows the user to make changes to the XML data file by editing records one at a time. An appropriate form field is given for each field of the record. Regarding the id field, this is implemented as a DropDownList instead to allow the user to select the record to edit. The remaining fields will always be maintained to match the record indicated in the DropDownList. When the user visits the page for the first time, the data of the first record will be pre-populated.

Validation must be implemented in all form fields. In addition, a submit button will save the new data to the data file. The contents of a label will indicate a success message when done.

Task 3: Invoice (invoice.aspx and print.aspx)

This feature will allow administrators to prepare a printable invoice. It is divided into two pages: invoice.aspx and print.aspx.

invoice.aspx page:

This page collects details about the invoice items. The form is designed to accept 3 rows of data for the invoice. A mock-up of the interface is below:

Full name of invoicee:			
Item name:	Unit price:	Quantity:	
	0.00	0	
	0.00	0	
	0.00	0	
		Submit	

Some notes about the interface:

- **Full name of invoicee:** Must be no longer than 50 characters to be valid. Required.
- **Item name:** Must be no longer than 100 characters to be valid.
- **Unit price and quantity:** Text boxes that must be validated to receive non-negative numbers only. A maximum of “1000.00” is allowed for unit price and a maximum of “100” is allowed for quantity.
- **Submit:** Validates the form and submits the form if valid. User is taken to the print.aspx page after submission.

The interface must also have appropriate validation controls and a ValidationSummary control at the end. Validation for individual inputs will be sufficient, and you may omit validation for whole rows of inputs in combination. A mock-up of the interface showing these is below:

Full name of invoicee:			*
Item name:	Unit price:	Quantity:	
	a	0	*
	0.00	1.5	*
	-1	0	*
		Submit	

You must correct the following errors:

- Full name is required.
- Unit price (row 1) must be currency between 0.00 and 1000.00.
- Quantity (row 2) must be an integer between 0 and 100.
- Unit price (row 3) must be currency between 0.00 and 1000.00.

Note that if the provided validation rules are not suitable for your particular organisation, then you may negotiate changes with an academic. Common sense will generally prevail. Note the changes in the “notes” part of your submission on the LMS.

print.aspx page:

This is the “destination” page after the form is submitted. This page will be a print-friendly version of the invoice form with simple labels and totals. All non-essential header, navigation and footer page components are omitted for print-friendliness. The invoice must include the name and contact details of the organisation at the top. A mock-up of the remaining output format is below:

Invoice for: Joe Bloggs

Date issued: 12/07/2018

Item:	Unit price:	Quantity:	Subtotal:
name of first item	\$700.00	1	\$700.00
name of second item	\$550.00	2	\$1,100.00
—	\$0.00	0	\$0.00
Total:			\$1,800.00

Terms: Invoice is due to be paid in full in 14 days.

Due date: 26/07/2018

Task 4: Stocktake (stocktake.aspx)

This page allows administrators to take a general stock inventory for any day-to-day items related to the organisation. Examples could be food, beverages, consumables, stationery, equipment, etc.

The stocktake form will allow the user to enter up to three rows of items. The form will be “pre-loaded” with dummy data to *simulate* taking the data from a data source.

The stocktake form also has a text input for the “reorder level”. The reorder level is the level that the stock must be replenished to. Only stock that has dipped below the reorder level needs to be replenished back to the reorder level.

A mock-up of the interface is below:

Reorder level:

Item name:	Unit price:	Stock quantity:	Reorder quantity:	Subtotal:
placeholder for item 1	5.50	2	8	44.00
placeholder for item 2	16.00	7	3	48.00
placeholder for item 3	29.90	15	0	0.00
Calculate			Total:	92.00

Some notes about the interface:

- **Reorder level:** Read only. Default 10. Modified with “-” and “+” buttons. Range is 0 to 1000.
- **Item name:** Must be no longer than 100 characters to be valid.

- **Stock quantity** and **unit price**: Required. Text boxes that must be validated to receive non-negative numbers only. A maximum of “1000.00” is allowed for unit price and a maximum of “100” is allowed for stock quantity. May be set to zero to indicate unused rows.
- **Reorder quantity, subtotal** and **total**: Derived data in labels.
- **Calculate**, “-” and “+”: Validates the form and recomputes calculations (if valid).

The interface must also have appropriate validation controls and a ValidationSummary control at the end. A mock-up of the interface showing these is below:

Reorder level: - 10 + Calculate					
Item name:	Stock quantity:	Unit price:	Reorder quantity:	Subtotal:	
placeholder for item 1		5.50	8	\$44.00	*
placeholder for item 2	7	1001.00	3	\$48.00	*
placeholder for item 3	a	29.90	0	\$0.00	*
			Total:	\$92.00	

You must correct the following errors:

- Stock quantity (row 1) is required.
- Unit price (row 2) must be currency between 0.00 and 1000.00.
- Stock quantity (row 3) must be an integer between 0 and 100.

Note that if the provided validation rules are not suitable for your particular organisation, then you may negotiate changes with an academic. Common sense will generally prevail. Note the changes in the “notes” part of your submission on the LMS.

Task 5: Chart (chart.aspx)

This page includes an ASP.NET Chart control to model data for the organisation with the Chart Type is set to Bar. The data will model the number of each registration type.

You only need to populate the chart with static data. I.e.: You are not expected to perform any data binding. It does not matter if your data does not automatically update to remain consistent with any other page in the site.

The chart must be given attention to detail for its presentation, including all components that you may expect for this type of chart: title, axis labels, data labels.

Task 6: Organisation

The following organisational aspects must be put in place for your solution:

(1) Programming language:

The website will be a server-side website using ASP.NET for the front-end and C# for the back-end. Some HTML and CSS may also be used where appropriate.

(2) Third-party components:

The website will be hand-coded by yourself. Therefore, you may not include any third-party components except for those mentioned in the assignment.

(3) Images:

Store the images in a subfolder called “images”. Note that the website cannot use copyrighted images. To avoid this problem, you are welcome to use images with an open licence. You can get these from [free image websites](#) or by performing an image search on your search engine with a “usage rights” filter set to “labelled for reuse” (or equivalent). Maintain a list of references in a file called **references.txt** in your solution with the name, author, location and access date of the images. You may reuse images from Assignment 2, provided these also follow the conventions above.

(4) Integrated development environment:

The website must be built using Visual Studio 2017.

(5) Template:

The website will be built from scratch beginning with an empty folder. Follow these steps:

- a) Create a new folder called **xxx_cse2icx_assignment3**, where xxx is your student ID.
- b) Open Visual Studio 2017. Select “File” -> “Open” -> “Website...” and select the new folder.
- c) Select “File” -> “Close Solution” immediately. This will prompt you to save a solution file (*.sln suffix). This must be saved in your new folder. Close Visual Studio 2017.
- d) Double-click on the solution file to open Visual Studio 2017 and your solution again. Continue to use this method for opening your solution going forward for convenience.
- e) You are now ready to go. Add your project assets next.

Task 7: Layout

The body of the website pages must be organised to include distinct header, navigation, main and footer semantic tags with the following content:

1. Header: Site name and branding.
2. Navigation: Navigation bar.
3. Main: This is the main content of each page.
4. Footer: Copyright statement.

Task 8: Navigation

The navigation bar appears on every page of the website and contains the following aspects:

1. The navigation bar has links to the five main pages of the website (Tasks 1-5).
2. Each navigation item has a fixed-size design.
3. The link for the current page is disabled and has a different colour/appearance.
4. The links for the non-current pages have a different colour/appearance when hovering.
5. The navigation bar is responsive, meaning that the menu options gracefully wrap to the next line when they cannot all fit horizontally on the screen.

Mock-up examples are as per Assignment 2.

Task 9: Presentation

The presentation of the website will largely be controlled by CSS with the following aspects:

1. All CSS code will be stored in a site-wide style file called **styles.css** and referenced on each web page.
2. The styling will include a variety of styling for layout.
3. The styling will include a variety of styling for content.
4. The presentation will endeavour to give the site a contemporary appearance, including HTML version 5 and CSS version 3 where practical.

Task 10: Code

This task will award marks according to the quality of your code according to these aspects:

1. Code indentation is consistent with one level of indentation per block.
2. Code consistently uses best-practice casing (lowercase, camelCase, TitleCase as needed).
3. All files each use some comments (<%-- comment --%>, <!--comment -->, /* comment */ or // comment as appropriate).
4. The Visual Studio 2017 Error List reports zero errors and zero warnings.

Submission

When you have completed, submit your answers via the link on the LMS. You should submit the following archive:

- a) Zip your Visual Studio 2017 solution folder to a file called **xxx_cse2icx_assignment3.zip**, where xxx is your student number. Take care to zip your solution folder rather than the contents of the folder to avoid a [tarbomb effect](#). The following files will be expected in your folder:
 - ASP.NET × 6: One file for each of Tasks 1-5 plus print.aspx (Task 3).
 - C# × 2: Data model classes used with serialisation code (Tasks 1-2).
 - XML × 1: data file (Tasks 1-2).
 - Newtonsoft.JSON assembly files (Tasks 1-2).
 - Subfolder: "images" subfolder with graphic assets (Task 6c).
 - Text file: references.txt for image references (Task 6c).
 - Solution file: xxx_cse2icx_assignment3.sln.
 - Web.config: Visual Studio may generate this file, which you can leave alone.

Assessment marking criteria

The marks for this assignment will be awarded as per the following marking rubric:

Task	Marks	Description	Total
1	0	The home page design could be improved.	7
	1	The home page design was attractive.	
	0	The data file was malformed.	
	1	The data file was well-formed.	
	0	The data file records did not meet the requirements.	
	1	The data file records had 4+ records with non-default data.	
	0	The code-behind was poor.	
	1	The code-behind could be improved.	
	2	The code-behind was done well: file reading, deserialisation and data binding.	
	0	The data grid data was not rendered.	
	1	The data grid data was rendered.	
	0	The label summary was incomplete.	
	1	The label summary code and output were correct.	
2	0	The form controls were incomplete.	7
	1	The form controls need improvement.	
	2	The form controls were complete.	
	0	The form control validation was incomplete.	
	1	The form control validation needs improvement.	

	2	The form control validation was complete.	
	0	The form did not submit correctly.	
	1	The form submits correctly.	
	0	The data was serialised and saved incorrectly.	
	1	The data was serialised and saved with errors.	
	2	The data was serialised and saved correctly.	
3	0	The form controls were incomplete.	7
	1	The form controls need improvement.	
	2	The form controls were complete.	
	0	The form control validation was incomplete.	
	1	The form control validation needs improvement.	
	2	The form control validation was complete.	
	0	The form did not submit correctly.	
	1	The form submits correctly.	
	0	The printable invoice data had errors.	
	1	The printable invoice data was correct.	
	0	The printable invoice presentation needs improvement.	
	1	The printable invoice was well presented.	
4	0	The form controls were incomplete.	8
	1	The form controls need improvement.	
	2	The form controls were complete.	
	0	The form control validation was incomplete.	
	1	The form control validation needs improvement.	
	2	The form control validation was complete.	
	0	The increment/decrement buttons did not behave correctly.	
	1	The increment/decrement buttons behaved correctly.	
	0	The calculate button did not behave correctly.	
	1	The calculate button behaved correctly.	
	0	The stocktake calculations were incorrect.	
	1	The stocktake calculations were correct in part.	
	2	The stocktake calculations were correct.	
5	0	The chart data was poor.	5
	1	The chart data could be improved.	
	2	The chart data was good.	
	0	The chart components were poor.	
	1	The chart components could be improved.	
	2	The chart components were good.	
	0	The chart presentation was unsatisfactory.	
	1	The chart presentation was satisfactory.	
6	0	Image references were incomplete.	1
	1	A list of image references correctly acknowledged the name, author, location and access date of the third-party images.	
7	0	The header areas were incomplete.	8
	1	The header areas could be improved.	
	2	A header area was included on all pages with the site name and branding.	
	0	The navigation areas were incomplete.	
	1	The navigation areas could be improved.	
	2	A navigation area was included on all pages with a navigation bar.	
	0	The main areas were incomplete.	
	1	The main areas could be improved.	
	2	A main area was included on all pages with the main content.	
	0	The footer areas were incomplete.	
	1	The footer areas could be improved.	

	2	A footer area was included on all pages with the copyright statement.	
8	0	The navigation bar consistency was poor.	9
	1	The navigation bar consistency could be improved.	
	2	The navigation bar links to the 5 main pages were on all pages.	
	0	Each navigation item did not have a fixed-size design.	
	1	Each navigation item had a fixed-size design.	
	0	The navigation item for the current pages was not always disabled.	
	1	The navigation item for the current pages was always disabled.	
	0	The links for the non-current pages did not have a different colour/appearance when hovering.	
	1	The links for the non-current pages had a different colour/appearance when hovering.	
	0	The navigation bar responsive component was incomplete.	
	1	The navigation bar responsive component needs improvement.	
	2	The navigation bar was responsive, meaning that the menu options gracefully wrapped to the next line when they cannot all fit horizontally on the screen.	
	0	The navigation bar links did not go to the correct locations.	
	1	The navigation bar links sometimes went to the correct locations.	
	2	The navigation bar links always went to the correct locations.	
9	0	All CSS code was not stored in a site-wide style file.	8
	1	All CSS code was stored in a site-wide style file.	
	0	The style file was not correctly referenced on each web page.	
	1	The style file was correctly referenced on each web page.	
	0	The styling for layout was poor.	
	1	The styling for layout could be improved.	
	2	The styling included a variety of good styling for layout.	
	0	The styling for content was poor.	
	1	The styling for content could be improved.	
	2	The styling included a variety of good styling for content.	
	0	The presentation was poor.	
	1	The presentation could be improved.	
	2	The presentation gave the site a contemporary appearance, including HTML version 5 and CSS version 3 where practical.	
10	0	Indentation was poor.	10
	1	Indentation could be improved.	
	2	Code indentation was consistent with one level of indentation per block.	
	0	Casing was poor.	
	1	Casing could be improved.	
	2	Code consistently used best-practice casing for each language as appropriate.	
	0	Commenting was poor.	
	1	Commenting could be improved.	
	2	All code files used appropriate comments.	
	0	The Error List reported 5+ errors.	
	1	The Error List reported 1-4 errors.	
	2	The Error List reported 0 errors.	
	0	The Error List reported 5+ warnings.	
	1	The Error List reported 1-4 warnings.	
	2	The Error List reported 0 warnings.	
–	0	Feedback will also be given regarding submission timeliness, length and format. Penalties may apply.	0
		Total available marks	70

