

KXT209

Assignment 1 Specification

Contents

ASSIGNMENT 1 SPECIFICATION	2
INTRODUCTION	2
DUE DATE AND SUBMISSION METHOD	2
MARKING	3
SYSTEM DESCRIPTION	3
DATABASE STRUCTURE	12
MARKING INFORMATION	24
HELP AND HINTS	25
PLAGIARISM AND CHEATING	26
SUBMISSION DETAILS	26
CHECKLIST	27

ASSIGNMENT 1 SPECIFICATION

INTRODUCTION

The aim of this assignment is to begin the implementation of a “Computing and Information Systems Library” website. The implementation will be completed in assignment 2.

CIS Library website contains the following sections:

- Home – a brief introduction to the CIS Library
- Search results page – a list of the results of search queries
- Book details page – a list of the details of the selected book
- Sign up page – a form that a user can fill in to sign up as a member of CIS Library
- My Library page – a list of the books that each user marked to view later - will be implemented in assignment 2
- Admin page – a page enabling the administrator to manage registered books and users. (i.e. add a book, delete a book or edit an user) - will be implemented in assignment 2
- Extra section – a list of the popular components of the search results (i.e. popular authors, popular keywords, popular subjects)

In the website, users will be able to interact with the database at three levels:

- “Public” – for users who have not signed up or logged in to the CIS Library website.
- “General Member” – for users who have logged in to the CIS Library website as a member.
- “Administrator” - for users who have logged in to the CIS Library website as an administration.

For more detailed information about the components of the system, you should read ‘**SYSTEM DESCRIPTION**’ carefully from [page 3 to 11](#). Users will be given an interface in order to interact with this "data store" in various ways. You will be required to import the 'default.sql' file and create a missing table which is not in the 'default.sql' file. You will find more information in ‘**DATABASE STRUCTURE**’ on [page 12](#).

A demo video for assignment is available on MyLO. If you are unsure about the assignment specification, please watch the demonstration video.

DUE DATE AND SUBMISSION METHOD

Due date: Friday 26 April 3PM

NOTE: Submission will **not** be through MyLO, it will be made using the School of Computing & Information Systems submission system (see **SUBMISSION DETAILS** on [page 26](#)).

Late Submissions

- Late assignments will only be accepted in exceptional circumstances and provided that the proper procedures have been followed (see the School Office or the CIS web site for details). Assignments that are submitted late without lecturer's approval will be subject to mark penalties if they are accepted at all (see School office or web site for details on this as well).
- Forms to request extensions of time to submit assignments are available from the School of Computing office. Requests must be accompanied by suitable documentation and should be submitted before the assignment due date.

MARKING

This assignment carries 10% of the marks for this unit. Remember that you must obtain at least 45% of the available marks for the in-semester assessment component (assignment 1 + assignment 2) to be able to obtain a passing grade in this unit.

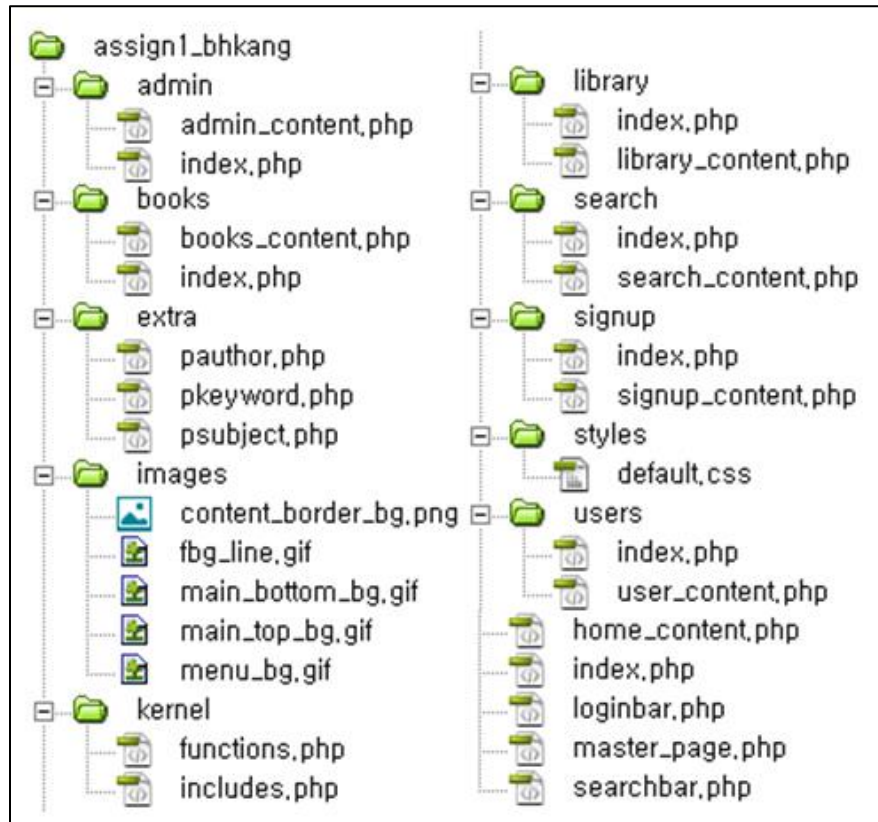
Details of the marking scheme are given in the '**MARKING INFORMATION**' on [page 24](#)).

SYSTEM DESCRIPTION

Default pages and database

There are two files given for this assignment: `default.zip` and `default.sql` (can be downloaded from MyLO).

Some files are pre-written (i.e. `default.css`, `home_content.php` and `master_page.php`). You **DO NOT** need to use the provided pre-written codes. However, You **MUST NOT** make any changes to the file name or the structure of this file (Except files in the `kernel` folder). You are **ONLY** allowed to add PHP files that are used for creating your own functions. Those added files **MUST** be located in the `kernel` folder.

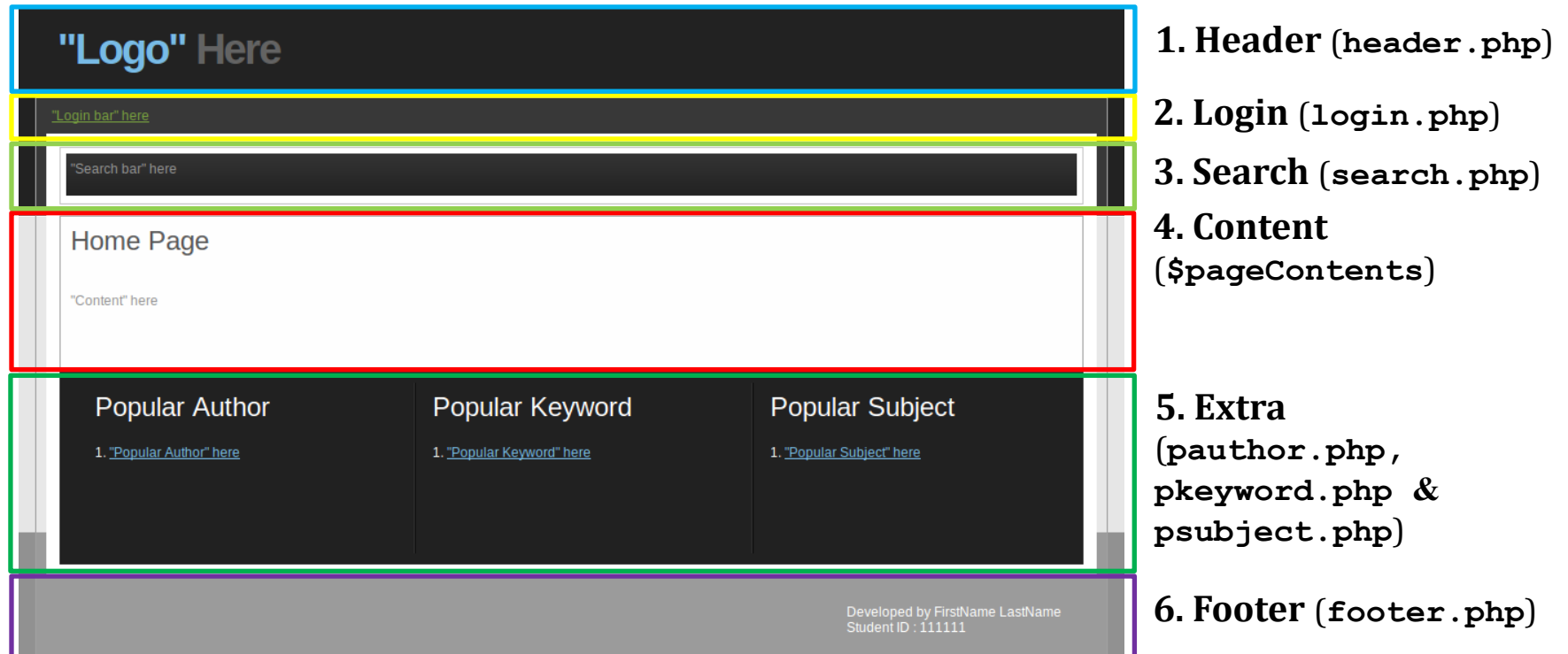
default.zip

NOTE: Your folder name should contain your username. (i.e. assign1_bhkang)

default.sql

- Download the file (default.sql) from MyLO. The database provided contains eleven different tables (see details in ‘**DATABASE STRUCTURE**’ on [page 12](#)). Import the default.sql file to MySQL in the alacritas/lawson server. You will be required to create a missing table (which is not in the default.sql file)

Basic Layout



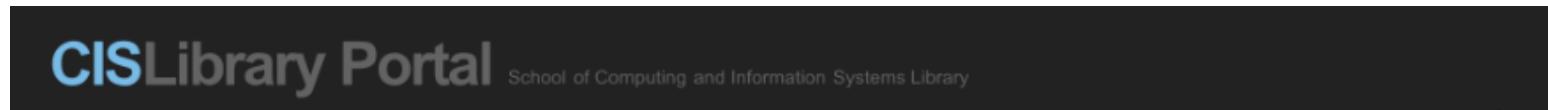
'Header', 'Login', 'Search', 'Extra' and 'Footer' section should be same in every page.

'Content' should be appropriate for each page.

Detailed specification for each section

There is a demonstration video clip in MyLO. If you are unsure what to do, please see the video clip.

1. **Header** (header.php): This section should be the same in every page. It should contain a logo for the CIS Library. From anywhere in the websites, the logo should redirect to the '**Home**' page. (You can change the logo name or appearance)



2. **Login bar** (loginbar.php): This section should be the same in every page.

2.1. **Before logging in**: There should be four components in this section:

- 1) A link that redirects to the '**Sign up**' page
- 2) Username field
- 3) Password field
- 4) 'Login' (submit) button.

The image shows a login bar with a dark grey background. On the left, there is a green link that says "If you haven't sign up, create your account". To the right of this link are two white input fields: one labeled "Username:" and one labeled "Password:". To the right of the password field is a grey button with the text "Login" in white.

If the username or password is invalid, the page should display the following message 'invalid username or password'.

The image shows the same login bar as before, but with an additional red error message at the bottom right that says "*invalid username or password".

If the user successfully logged in to the website, login bar should display as below (2.2. **After logging in**).

2.2. **After logging in**: There should be five components in this section:

- 1) Welcoming message with an account name (should be retrieved from database and facilitated with session)
- 2) A link that redirects to '**My Library**'
- 3) A link that redirects to '**My Page**'
- 4) A link that redirects to '**Admin page**'
- 5) 'Logout' (submit) button. When the 'Logout' button is clicked, the login bar should display as "2.1. **Before logging in**".

The image shows the login bar after a successful login. It has a dark grey background. On the left, it says "Welcome! bhkang" in white. To the right of this are three green links: "[My Library]", "[My Page]", and "[Admin Page]". To the right of these links is a grey button with the text "Logout" in white.

3. **Search bar** (`searchbar.php`): This section should be the same in every page. There should be three components in this section:
- 1) Drop-down list: it should contain 'Title', 'Keyword', 'Author' and 'Subject' (should be retrieved from the '**KXT209_SearchType**' table – see **DATABASE STRUCTURE** on [page 20](#)) - Each book has information, 'Title', 'Keyword', 'Author' and 'Subject'. Please check the **DATABASE STRUCTURE** on [page 12](#). The search only retrieves the information of selected search type. For example, when 'Title' is selected, the search result shows the list of books which has the typed query in the 'Title'.
 - 2) Text box: user can type search queries here. – It should display the previous search query when the content shows the search results(4.2 '**Search result**' page) *Please see Demo video clip
 - 3) Search button: by clicking this button the page should be redirected to the '**Search result**' page (see 4.2) with typed search queries.



The image shows a search bar interface. On the left is a dropdown menu with 'Title' selected, and options 'Title', 'Keyword', 'Author', and 'Subject' visible. To the right of the dropdown is a text input field. To the right of the text field is a button labeled 'Search'.

4. **Content**: It should contain the relevant information for each page. Each page should have an appropriate '**Content Title**' (i.e. Home, Search Result, Book details, etc.).
- 4.1. '**Home**' page (`home_content.php`)
- The content section of this page should contain a brief introduction to the application: a list of pages (i.e. '**Home**', '**Search result**', '**Book details**', '**Sign up**', '**My Page**', '**My Library**' and '**Admin Page**' with a brief description of each page). By clicking the logo in the '**Header**', page should be redirected to this page from anywhere in the website.



The image shows the content of the Home page. It starts with the title 'Home' and a welcome message 'Welcome to CIS Library Portal Website.' followed by a green link 'A brief introduction.' Below this is a list of links, each followed by a brief description in green text: 'Home - A brief description', 'Search result - A brief description', 'Book details - A brief description', 'Sign up - A brief description', 'My Page - A brief description', 'My Library - A brief description', and 'Admin Page - A brief description'.

4.2. 'Search result' page (search_content.php)

The page should retrieve a list of books that match a search query in the CIS Library. At this stage, all queries should be considered as one word when the system retrieves the data. For example, 'web services' has two words: 'web' and 'services', but we consider it as one word 'web services' (i.e. `SELECT ... FROM ... WHERE ... LIKE '%web services%'`).

The following image indicates how the content section of 'Search result' page should be.



Following functions should be implemented.

- **Redirecting to the 'Book details' page**

When the title is selected, the page should be redirected to 'Book details' page (4.3).

- **Redirecting to the 'Search result' page**

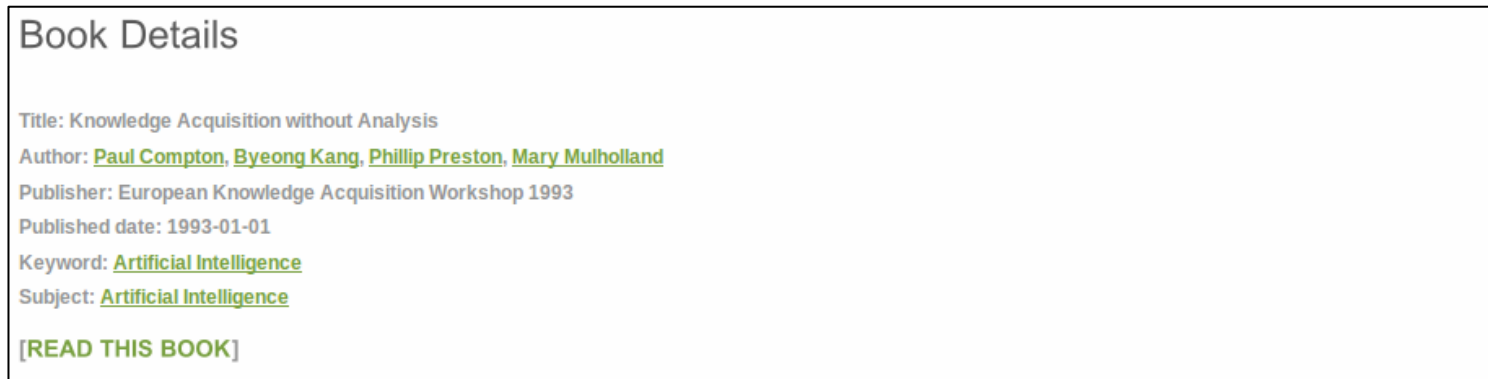
When the author is selected, the page should be redirected to 'Search result' page (4.2) with an author's name as a query and 'Author' as a search type. Please note that the outcome of this is same as using search bar (**3. Search bar**).

- **Popularity counting**

When the title is selected, selected book is counted as being given a popularity vote. The click-count made by public user should be counted as well. Note that previously selected book should not be counted until the user logout (for general member or administrator ONLY) or the browser is closed (for all users including public users). The counts should be stored in the table '**KXT209_Popularity**'. Three factors (Keyword, Author and Subject) should be counted. These counts will be used in '5.Extra' section.

4.3. 'Book details' page (book_content.php)

The following image indicates how the content section of 'Book details' page should be.



Following functions should be implemented.

- **Redirecting to the 'Search result' page**

When the author is selected, the page should be redirected to 'Search result' page (4.2) with the selected author's name as a query and 'Author' as a search type. Please note that the outcome of this is same as using search bar (**3. Search bar**).

When the keyword is selected, the page should be redirected to 'Search result' page (4.2) with the selected keyword as a query and 'Keyword' as a search type. Please note that the outcome of this is same as using search bar (**3. Search bar**).

When the subject is selected, the page should be redirected to 'Search result' page (4.2) with the selected subject as a query and 'Subject' as a search type. Please note that the outcome of this is same as using search bar (**3. Search bar**).

- **Redirecting to the URL of book.**

When '[READ THIS BOOK]' is clicked, the page should be redirected to the URL which is stored in the table 'KXT209_Books'.

4.4. 'Sign up' page (signup_content.php)

This page should contain a form that a user can fill in to "sign up" as a member of CIS Library. Each field of the form should be styled to indicate whether it is required and, if applicable, a short description of the type of information to be entered.

To do so, you should utilize the table 'KXT209_Users' in the provided database 'default.sql'.

The following image indicates how the content section of 'Sign up' page should be.

The following **validations** are required:

- Username – This is a required field. To be valid the user must enter a value that contains only letters or digits and is not a username that has been taken already.
- Password – This is a required field. To be valid the user must enter a value that contains at least 6 characters which must not include any spaces (all other characters are acceptable).
- Re-type Password– This is a required field. To be valid the user must enter a value that is exactly the same entry as the first password entry.
- First Name, Last Name – This is a required field.
- Middle Name – This is an optional field.
- Date of Birth – This is a required field. Users should be given 3 drop down lists to enter this (it must be a valid date) – the options in the drop down lists must make it possible to enter no date. <day (1 – 31), month ('Jan' - 'Dec'), year (1910 – 2010)>
- Email – This is an optional field. To be valid, it must enter @ sign.
- Affiliation– This is a required field.

If the any of entered data is not valid, the error message should appear next to the related field and the valid data should remain (Please see Demo).

4.5. 'My Page', 'My Library' and 'Admin Page' (user_content.php, library_content.php & admin_content.php)

These pages will be implemented in Assignment 2. The page should indicate that the implementation is not completed.

5. **Extra** (pauthor.php, pkeyword.php & psubject.php):

This section should contain three columns: 'Popular Authors', 'Popular Keywords' and 'Popular Subjects'. Each section displays the top 5 most viewed keywords in the CIS Library.

The keywords in these sections should be retrieved from the table '**KXT209_Popularity**' (see details in 4.2). Each keyword redirects to 'Search result' page with keyword (as a query). Please note that the outcome of this is same as using search bar (**3. Search bar**).

Popular Authors	Popular Keywords	Popular Subjects
1. Paul Compton	1. Artificial Intelligence	1. Artificial Intelligence
2. Byeong Kang	2. Artificial Intelligence (incl. Robotics)	2. Artificial Intelligence (incl. Robotics)
3. Phillip Preston		3. Mathematical Logic and Formal Languages
4. Mary Mulholland		4. Data Mining and Knowledge Discovery
5. Ivan Bindoff		5. Information Systems Applications (incl. Internet)

6. **Footer** (footer.php): It should contain the following information:

- The name of the developer
- The student number of the developer

Developed by Byeong Ho Kang Student ID : 000000
--

7. **Others**

7.1. **Read me** text file

The file readme.txt should contain the following:

- 1) Student details: full name, username, id number, and campus
- 2) A list of resources you used in preparing this assignment
- 3) A list of known problems that you have identified when testing your implementation and have not been able to solve.

DATABASE STRUCTURE

The file 'default.sql' can be downloaded MyLo. You should import the 'default.sql' file to the MySQL of alacritas and lawson. After you import it, you can see eleven different tables are imported in your database. There is one table missing in the file 'default.sql'. You should create the table. The structure of each table is provided as follows. For your information, we put some sample data in each table.

NOTE: You must not make any changes to the structure of those tables and the database. You should not delete the provided sample data, which was already saved in the tables. If you do not follow these instructions as to database structure, your mark for this assignment will be reduced by between 1 and 2 grade levels.

1. 'KXT209_Author' table

The structure of the 'KXT209_Author' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
FName	text	No		None	
MName	text	No		None	
LName	text	No		None	
Affiliation	text	No		None	
Updated_time	datetime	No		None	

* Updated_time :This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Author' should be:

ID	FName	MName	LName	Affiliation	Updated_time
1	Byeong	Ho	Kang	University of Tasmania	2012-08-08 20:40:00
2	First	Middle	Last	University of Tasmania	2012-08-08 20:40:00

2. 'KXT209_Books' table

The structure of the 'KXT209_Books' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Title	Text	No		None	
Pub_id	int(11)	No		None	
URL	Text	No		None	
Published_date	Date	No		None	
Updated_time	datetime	No		None	

* Pub_id: Foreign key that links to table 'KXT209_Publisher'.

* Published_date: published date of book/paper

* Updated_date: updated date (This value should not be extracted from the 'sign up' form. This can be extracted by using current time function)

Sample data in this table 'KXT209_Books' should be:

ID	Title	Pub_id	URL	Published_date	Updated_time
1	Knowledge Acquisition without Analysis	1	http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.31.7951&rep=rep1&type=pdf	1993-01-01	2012-08-08 20:40:00

*You should not delete sample data.

3. 'KXT209_Books_author' table

The structure of the 'KXT209_Books_author' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Book_id	int(11)	No		None	
Author_id	int(11)	No		None	
Updated_time	datetime	No		None	

* Book_id: Foreign key that links to table 'KXT209_Books'.

* Author_id: Foreign key that links to table 'KXT209_Author'.

* Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Books_author' should be:

ID	Book_id	Author_id	Updated_time
1	1	1	2012-08-08 20:40:00
2	1	2	2012-08-08 20:40:00

4. 'KXT209_Books_keyword' table

The structure of the 'KXT209_Books_keyword' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Book_id	int(11)	No		None	
Keyword_id	int(11)	No		None	
Updated_time	datetime	No		None	

* Book_id: Foreign key that links to table 'KXT209_Books'.

* Keyword_id: Foreign key that links to table 'KXT209_Keyword'.

* Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Books_keyword' should be:

ID	Book_id	Keyword_id	Updated_time
1	1	1	2012-08-08 20:40:00
2	1	2	2012-08-08 20:40:00

5. 'KXT209_Books_subject' table

The structure of the 'KXT209_Books_subject' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Book_id	int(11)	No		None	
Subject_id	int(11)	No		None	
Updated_time	datetime	No		None	

* Book_id: Foreign key that links to table 'KXT209_Books'.

* Subject_id: Foreign key that links to table 'KXT209_Subject'.

* Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Books_subject' should be:

ID	Book_id	Subject_id	Updated_time
1	1	1	2012-08-08 20:40:00
2	1	2	2012-08-08 20:40:00

6. 'KXT209_Keyword' table

The structure of the 'KXT209_Keyword' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Keyword	varchar(255)	No		None	
Updated_time	datetime	No		None	

* Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Books_keyword' should be:

ID	Keyword	Updated_time
1	Artificial Intelligence	2012-08-08 20:40:00
2	Artificial Intelligence (incl. Robotics)	2012-08-08 20:40:00

7. 'KXT209_Popularity' table – This table is not in the default.sql file. You will be required to create this table.

The structure of the 'KXT209_Popularity' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Type	varchar(255)	No		None	
Element_id	int(11)	No		None	
Count	int(11)	No		None	
Created_time	datetime	No		None	
LUpdated_time	datetime	No		None	

*Type: type of element (i.e. Author, Keyword or Subject)

*Element_id: Foreign key that links to relevant table (i.e. 'KXT209_Author', 'KXT209_Keyword' or 'KXT209_Subject')

*Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Popularity' should be:

ID	Type	Element_id	Count	Created_time	LUpdated_time
1	Author	1	1	2012-08-08 20:45:00	2012-08-08 20:45:00
2	Author	2	1	2012-08-08 20:45:00	2012-08-08 20:45:00
3	Keyword	1	1	2012-08-08 20:45:00	2012-08-08 20:45:00
4	Subject	1	1	2012-08-08 20:45:00	2012-08-08 20:45:00

8. 'KXT209_Publisher' table**The structure of the 'KXT209_Publisher' table**

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Name	text	No		None	
City	text	No		None	
Country	text	No		None	
Updated_time	datetime	No		None	

*Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Publisher' should be:

ID	Name	City	Country	Updated_time
1	European Knowledge Acquisition Workshop 1993	Toulouse and Caylus	France	2012-08-08 20:45:00
2	20th Australian Joint Conference on Artificial Intelligence	Gold Coast	Australia	2012-08-08 20:45:00

9. 'KXT209_SearchType' table

The structure of the 'KXT209_SearchType' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Type	Varchar(255)	No		None	

*Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_SearchType' should be:

ID	Subject
1	Title
2	Keyword
3	Author
4	Subject

10. 'KXT209_Subject' table**The structure of the 'KXT209_Subject' table**

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Subject	varchar(255)	No		None	
Updated_time	datetime	No		None	

*Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Subject' should be:

ID	Subject	Updated_time
1	Artificial Intelligence	2012-08-08 20:40:00
2	Data Mining and Knowledge Discovery	2012-08-08 20:40:00

11. 'KXT209_Users' table**The structure of the 'KXT209_Users' table**

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Ulevel	varchar(255)	No		None	
Username	datetime	No		None	
Password	varchar(32)	No		None	
FName	varchar(255)	No		None	
MName	varchar(255)	No		None	
LName	varchar(255)	No		None	
Birthdate	date	No		None	
Email	text	No		None	
Phone	varchar(255)	No		None	
Website	text	No		None	
Affli	text	No		None	
Updated_time	datetime	No		None	

*Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_Users' should be:

ID	Ulevel	Username	Password	FName	MName	LName	Birthdate	Email	Phone	Website	Affli	Updated_time
1	2	bhkang	0d0f0c60802a0106e8fbac7c2f7d6e90	Byeong	Ho	Kang	1960-09-09	bhkang@utas.edu.au			University of Tasmania	2012-08-08 20:40:00

12. 'KXT209_User_Library' table – this table will be used in assignment 2.

The structure of the 'KXT209_User_Library' table

Field	Type	Null	Key	Default	Extra
ID	int(11)	No	PRI	None	auto_increment
Book_id	int(11)	No		None	
User_id	int(11)	No		None	
Updated_time	datetime	No		None	

* Book_id: Foreign key that links to table 'KXT209_Books'.

* User_id: Foreign key that links to table 'KXT209_Users'.

* Updated_time: This value should not be extracted from the 'sign up' form. This can be extracted by using current time function.

Sample data in this table 'KXT209_User_Library' should be:

ID	Book_id	User_id	Updated_time
1	1	1	2012-08-08 20:40:00

MARKING INFORMATION

This assignment will be marked out of 10. The performance standards required for full marks for each section are shown below. Where your performance standard is below the levels shown here your mark for that section will be reduced.

Your submitted work will be tested on Firefox for your marking. Please test your work before your submission.

Criterion	Performance standards	Maximum mark for this criterion
Follows acceptable programming practice	<ul style="list-style-type: none"> All references (URI values) within the site are <u>relative</u> Display is not seriously broken No other serious violations of the programming practices required for this unit Application behaves correctly when the content of the data-store is changed The database and tables are used as required (*database on school server is <u>case-sensitive</u>) Assignment coversheet is included 	N/A This criterion is important. If you do not match the required performance standard for this criterion your mark for this assignment will be reduced by between 1 and 2 grade levels. (i.e. DN → CR)
Correct submission of work as specified	<ul style="list-style-type: none"> Files are correctly named Directory structure follows as specified Read me file contains all required information 	0.5 mark (5%)
Overall structure of application is as required	<ul style="list-style-type: none"> Layout of pages essentially as specified Appropriate 'Title' for each page is used 'Extra' section appears correctly on all of the page 'Footer' section appears correctly (student info). 	1.5 mark (15%)
Database usage: import default.sql file and create a table.	<ul style="list-style-type: none"> The provided 'default.sql' file is imported correctly to the MySQL in the alacritas/lawson Required table is created correctly 	0.5 mark (5%)
Search result section functions correctly	<ul style="list-style-type: none"> Displays the list of the results correctly Links work correctly 	1.0 mark (10%)

Popularity function works correctly	<ul style="list-style-type: none"> Count is correctly stored in database Distinct public and admin/general user correctly 	1.0 mark (10%)
Book details section functions correctly	<ul style="list-style-type: none"> Displays the details of the book correctly Links work correctly 	1.0 mark (10%)
Extra section correctly	<ul style="list-style-type: none"> Displays top five keywords correctly Links work correctly 	1.0 mark (10%)
Login section correctly	<ul style="list-style-type: none"> Username and Password check works correctly Retrieves a correct username correctly Links work correctly 	1.0 mark (10%)
Member signup section functions correctly	<ul style="list-style-type: none"> Form is displayed correctly (shows required fields) Validation work correctly The value of the password should be encrypted The submitted data should be inserted into the database correctly 	1.5 mark (15%)
Code is well written	<ul style="list-style-type: none"> Code can be easily read and the logic follows Uses built-in PHP functions where possible Follows correct naming conventions for variables and functions Object-oriented programming is used. 	1.0 mark (10%)

HELP AND HINTS

- Read the specification carefully and make sure that you know what the PHP application needs to do (Demo video is also available).
- Look at the examples in self-studies, tutorials, and lectures, to see whether you have seen similar tasks before.
- Try out techniques that you think will do the job.
- DO NOT neglect your self-study and tutorial work in the unit to work on the assignment. Some of the activities may lead you to see what needs to be done in the assignment.
- You may seek help with this assignment in normal consultation times for this unit or via email to your lecturer or tutor. Here are some hints to make best use of help.

- Make sure you know (or think you know) what your problem is.
 - Have details of the work you have done so far and the progress you have made on hand when you seek help.
 - The more specific you can be in your request for help the more immediately useful the help is likely to be.
- It is highly recommended to develop the websites in school machine. Student who uses his/her own server for assignment, check the programs with school machine before you submit the assignment. (Student must take responsibility for using his/her own server).

PLAGIARISM AND CHEATING

Practical assignments are used by the School of Computing and Information Systems for students to both reinforce and demonstrate their understanding of material which has been presented in class. They have a role both for assessment and for learning. It is a requirement that work you hand in for assessment is substantially your own.

Cheating

- Cheating occurs if you claim work as your own when it is substantially the work of someone else.
- Cheating is an offence under the Ordinance of Student Discipline within the University. Furthermore, the computing profession has ethical standards in which cheating have no place.
- Cheating involves two or more parties.
- If you allow written work, computer listings, or electronic version of your workbook to be borrowed or copied by another student you are an equal partner in the act of cheating.
- You should be careful to ensure that your work is not left in a situation where it may be stolen by others.
- Where there is a reasonable cause to believe that a case of cheating has occurred, this will be brought to the attention of the unit lecturer. If the lecturer considers that there is evidence of cheating, then no marks will be given to any of the students involved. The case will be referred to the Head of School for consideration of further action.

SUBMISSION DETAILS

- Make sure that the folder in which you have all the files for this assignment (as specified in the page 3) is enclosed in a folder that is labelled with your username.
- Export your database in sql file format. The file name should be labelled with your username (i.e. `assign1_bhkang.sql`).
- Download the word version of the School of Computing & Information Systems “assignment cover sheet” (follow the links from the home page), edit it to contain your details, and copy the edited version into the folder containing all the files for this assignment. In doing this you are deemed to have submitted a signed version of the cover sheet.

- Make a .zip archive of the folder with your exported sql file(assign1_bhkang.sql). On a MAC this is very easy – in the finder select the folder to “zip”, right click and select the relevant option from the drop down list you will see (this will be labelled “compress ...” or possibly “create archive”), the system will then create a .zip file. If your username is ‘bhkang’ and all of the files for the assignment submission were in a folder and labelled “assign1_bhkang.zip” then the archiving (zipping) process would produce a file called “assign1_bhkang.zip”.
- Go to your home directory on alacritas (Hobart students) or lawson (Launceston students).
- You will find there a directory (folder) called kxt209submit (the “submit” folder) (If you do not have a submit folder, please contact the Computing Help desk.)
- Copy your .zip file into this directory. Open the submit folder to see that your file has been copied. If you want to resubmit, open the submit folder, remove the old version of your submitted work and copy in the new version. Please keep the submitted files.

CHECKLIST

When you can answer yes to all of these questions, you are ready to submit the assignment

Q: Are all the files contained in a folder (directory) that is labelled with your user name?

Q: Are all the files that are required present?

Q: Have you checked the function of all aspects of the application and documented any problems in the readme.txt file?

Q: If you have changed any file names or locations, have you rechecked the function?

Q: Have you have checked that all of the URIs are relative references? Your application will be moved to a different location for testing – any links back to your alacritas / lawson account will be noticed and penalised.

Q: Have you exported your database in sql file format (assign1_username.sql)?

Q: Have you filled in a copy of the School of Computing & Information Systems assignment cover sheet and added it to the folder that you will submit?

Q: Have you made a .zip file of the final version of your folder?

Q: Do you have a submission folder for this unit (kxt209) in your alacritas / lawson account?

Q: Have you checked whether your program is working properly in the school machine (Firefox in Mac)?