CS 4WW3/6WW3 - Project Part 3 - Server-side			Course:	4WW3	6WW3
Student name:			**Student number:**		
Criteria	**Performance Level**			**Score**	
	Excellent	**Satisfactory**	**Unsatisfactory**		
Core programming tasks	The search form successfully returns a dynamically generated page with a table and live map with markers based on searching by name, rating, or location.	The search form successfully returns a dynamically generated page with at least a map with markers, on search by at least two of name/rating/location.	Submitting the search form does return a dynamically generated page, or it does not include a map, or it searches on only of name/rating/location.		
	The search results include links to dynamically generated individual object pages, which contain the object information, live map, and ratings and reviews submitted by users.	The search results include links to dynamically generated individual object pages, but the pages do not contain all of the required components.	The search results do not link to individual object pages, or the individual object pages are not dynamically generated.		
	The individual object page includes a form for logged-in users to submit a rating and review of the object, and this is successfully added to the database upon submission and displayed when the page is viewed later		There is no mechanism for users to submit ratings/reviews, or the database and page is not updated after submission.		
	The user registration page allows a user to register.		Users cannot register for accounts.		
	For the user registration page, data of different formats, at least numeric, alphabetic, email and date formats, is validated correctly on the server side using PHP.	Data validation of some different formats OR data validation is present but the validation does not execute correctly	Different data formats are not validated - only validates presence of input OR No data validation		
	The object submission page allows a logged-in user to submit a new object, including uploading an image	The object submission page allows a logged-in user to submit a new object, but image upload is not supported or implemented correctly.	Users cannot submit new objects.		
	For the object submission page, data is validated correctly on the server side using PHP.		No server-side validation.		
	PHP pages post back to themselves to avoid duplicating code for re-displaying completed forms when displaying an error to the user.	PHP pages post back to themselves, but do not include full error handling.	PHP pages do not post back to themselves.		
	Query string is used to pass and receive information between search results and individual object pages (e.g., showobject.php?id=17)		Individual object pages are not dynamically generated.		
	Client-side display and functionality remains satisfactory in terms of part 1 and part 2 requirements.		Client-side display and functionality is unsatisfactory in terms of part 1 and part 2 requirements.		
Comments on core programming tasks:				**/ 20**	
PHP	PHP code has been structured with a number of functions with little or no repeated code	PHP code has a few functions OR the functions contain a large amount of repeated code	PHP code has no structure		
	PHP code perfectly laid out and easy to read with consistent formatting	PHP code is readable but has a number of formatting inconsistencies	PHP has no formatting or a large number of formatting inconsistencies		
	PHP code contains a large number of good explanatory comments	PHP code contains a few useful comments	PHP code contains no comments or comments are not useful		
	PHP includes used to avoid repeating common HTML components (header, menu, etc.)	Some common HTML components are de-duplicated into include files.	Common HTML components are repeated across many HTML/PHP source files.		
	PHP data objects (PDO) used for accessing the database	The mysql_ or mysqli_ API is used for accessing the database.	No database access.		

	Files, functions, and variables have meaningful names.		Files, functions, and variables do not have meaningful names.	
Comments on PHP code:				**/ 10**
Database design	Database model uses appropriate column types for each data field.		No data model provided or very poor data design	
	Database model uses ids to link reviews to individual objects.	Rows in the reviews table replicate fields from another table.	No data model provided or no table for reviews	
Comments on database design:				**/ 3**
Security and deployment	Database access code always uses countermeasures for SQL injection attacks (prepared statements)	Database access code uses some countermeasures for SQL injection attacks (some prepared statements or string escaping/validation)	No SQL injection countermeasures in PHP code.	
	Website and all page elements (external JS/CSS/images) are served over SSL/TLS without errors.	Some portions of the website served over SSL/TLS, but some errors or mixed content warnings.	SSL/TLS not used.	
	Passwords are not stored in the database in clear text -- they are hashed and salted.	Passwords are not stored in the database in clear text -- they are hashed but not salted.	Passwords are stored in clear text.	
	Website is hosted on Amazon Web Services EC2 virtual server.		No live website provided.	
	Images uploaded via the object submission page are stored in an Amazon S3 bucket. When the user views a page containing the image, the image is served from the S3 bucket directly to the user.		User uploaded images are not stored or served from an S3 bucket.	
Comments on security and deployment:				**/ 5**
Add-on task #3	On individual item pages, AJAX is used to submit a user's rating and review, and the new rating and review are dynamically inserted into the DOM using Javascript, without reloading the page.	AJAX used for submitting rating and review, but page is not dynamically updated via Javascript.	AJAX not used.	
	AJAX code includes error handling in the case of an error.		AJAX code does not include error handling.	
Comments on add-on task #3:				**/ 5**