**Install, use and develop an application which uses server-side scripts (PHP), to access and update an SQL database.**

### Table of Contents:

[Title 2](#_Toc31111860)

[Assignment Task 4](#_Toc31111861)

[References 10](#_Toc31111863)

## Assignment Task

GitHub Repository: <https://github.com/JamieChopra/bnu-php-example>

Task 1-5 Video Demonstration: <https://www.youtube.com/watch?v=HYOnwXED40k>

1. Text

   Description automatically generated(task1.php) Timestamp: 0:12

Inserts 5 student records into the database and redirects to student.php (table of student records)

Text

Description automatically generated

1. (students.php) Timestamp: 0:51

Creates a HTML table and displays the database rows

Text

Description automatically generated

1. (students.php) Timestamp: 1:18

HTML form that redirects to delete.php, HTML checkbox formatted in table that stores the selected checkbox values in an array.

Onclick HTML method that produces javascript prompt for user validation when deleting an item from the table/database.

Text

Description automatically generated (delete.php)

When “DELETE” button is clicked the checked items in the checkbox are deleted from the database

1. Text

   Description automatically generated(addstudent.php) Timestamp: 2:23

HTML form to take user input for data, when the “Submit” button is clicked, the POST action is to savestudent.php, all inputs require user input via HTML ‘require’

Text

Description automatically generated

(savestudent.php)

Saves the user input (from addstudent.php) to PHP variables, PHP variables are inserted into SQL database via “INSERT INTO student

$passwordhash encrypts inputted password to be stored in database.

Text

Description automatically generated

1. (addstudent.php) Timestamp: 3:22

Prompts user for input of a file, only accepts file types of .png and .jpg (File Type Validation)

Text

Description automatically generated (savestudent.php)

Image files are stored to $image PHP variable, image data is extracted and stored into $imagedata, which is then stored into the database via “INSERT INTO student”

Text

Description automatically generated (getjpg.php)

Pulls image data linked to each student id in the database, the images are then displayed using $displayimage

(students.php)

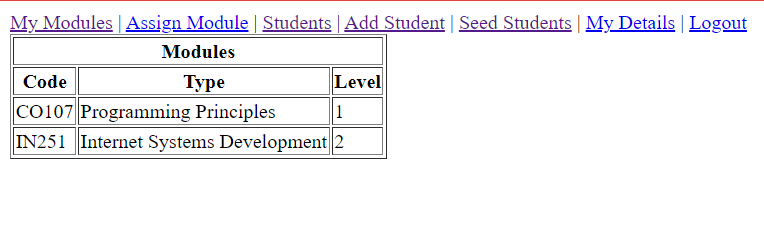
Uses the getjpg.php to display the student images within the HTML table in students.php



1. <https://github.com/JamieChopra/bnu-php-example/tree/master/templates/partials>

Table

Description automatically generated with low confidenceBefore After



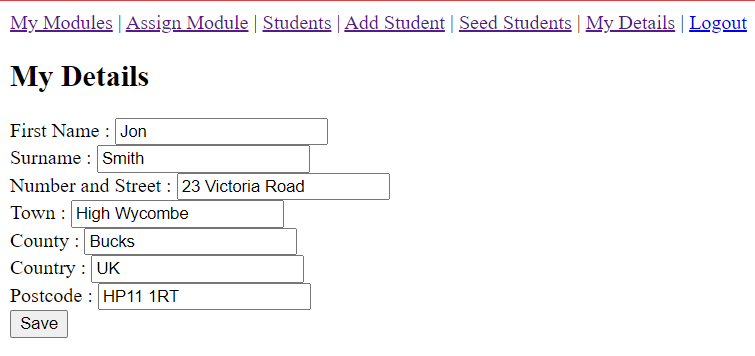
Graphical user interface, text, chat or text message

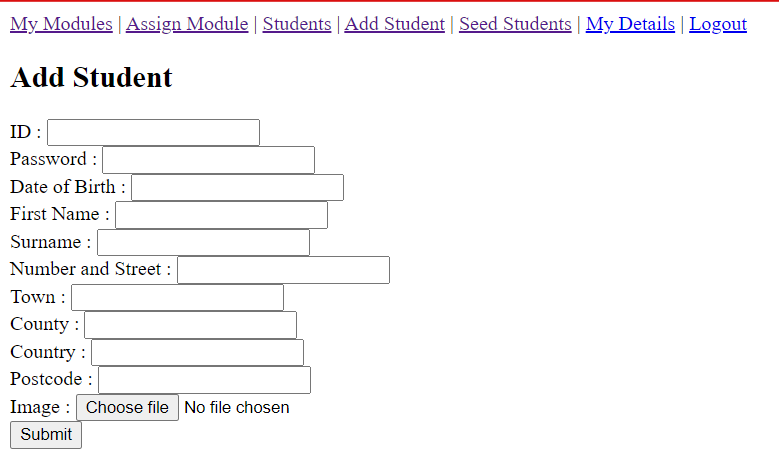
Description automatically generated

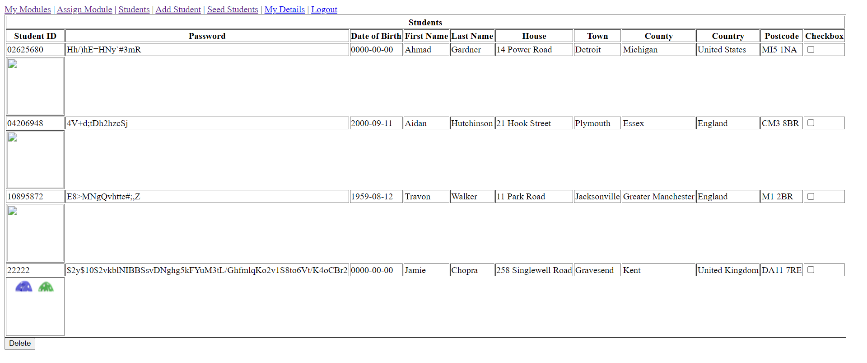
A picture containing text

Description automatically generated

Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedApplication, table

Description automatically generated with medium confidence

## Text Description automatically generated(savestudent.php)

## Text Description automatically generatedPrevents SQL injections via user input, as mysql\_real\_escape\_string() filters special characters by inputting random characters between them, so if a user attempts to manipulate a SQL query, the code is made ineligible by random characters.

## (getjpg.php)

## Text Description automatically generated (delete.php)

1. A.

Node.js as discussed by CHANIOTIS, I.K., KYRIAKOU, K.D. and TSELIKAS, N.D. (2015) is a server-sided development platform to deploy JavaScript code. It is used when developing high-scale backend services that will undergo a high intensity processing of user requests, for example commercial web and mobile applications which include running tasks of storage of data to databases.

Node.js allows for the scaling of JavaScript web applications as they can handle larger amounts of HTTP requests via placing HTTP requests from all users into an EventQueue which is then processed by an EventLoop. The EventLoop identifies and processes the HTTP requests in order of when the events were applied, this in turn reduces the processing power intensity requirements of a web application as each instance of the web application being ran does not require a new thread on the web server, instead all processed are ran on one individual thread (single-threaded).

PHP as discussed O’Reilly Online Learning (2022) is a server-sided scripting programming language mainly utilised for the development of web applications. PHP provides the back-end code for the input, output and storage of HTML data processed on a web server. It allows for dynamic implementations to a website such as temporary storage of user input data. PHP then can access MySQL (database software) where it is able to store its temporary data permanently. PHP code can then be utilised to bind HTML features to MySQL commands to access, retrieve, manipulate and store data to the MySQL database via a web browser application.

B.

SQL injections as discussed by Yunus, M.A.M, et al. (2018) is the” execution of malicious SQL queries on the database server”. These malicious queries include the extraction, manipulation and deletion of sensitive data from an SQL database and therefore pose a threat on the confidentiality responsibility proposed in GDPR regulations.

SQL injections are deployed by attempting to manipulate code strings in the input fields within the front-end access of a database e.g the login field to a web application, without sufficient SQL injection prevention mechanism an attacker can control the SQL database in every form possible from retrieving sensitive data (SELECT 1=1 FROM insertDatabse) to deleting all the data stored in the database (DELETE FROM insertDatabase WHERE 1=1).

SQL injections can be prevented against by the implementation of user input validation and the parameterization of SQL queries, which uses placeholders for the ‘VALUES’ when making changes to an SQL database which are only executed when intended, this prevents attackers from being able to randomly execute malicious SQL code.

Another threat to a system handling personal data is as discussed by S. T. Zargar, J. Joshi and D. Tipper (2013) is Denial of Service, as under GDPR regulations availability requirement, a user must always have sufficient access to their personal data. Denial of Service is an attack in which a server (in this case web or SQL server) is overloaded with the flooding of network traffic causing the system to shut down. This can be prevented by the implementation of intrusion detection systems which identifies unusual trends in the movement of traffic on a network (web server) and generates alerts based on these unusual trends allowing for security measures to be implemented.

## References

CHANIOTIS, I.K., KYRIAKOU, K.D. and TSELIKAS, N.D., 2015. Is Node.js a viable option for building modern web applications? A performance evaluation study. *Computing.Archives for Informatics and Numerical Computation,***97**(10), pp. 1023-1044. doi: 10.1007/s00607-014-0394-9

O’Reilly Online Learning. 2022. *Learning PHP & MySQL, 2nd Edition*. [online] Available at: <https://www.oreilly.com/library/view/learning-php/9780596514013/ch01.html> (Accessed 17 May 2022)

Yunus, M.A.M, et al. 2018. Review of SQL Injections: Problems and Preventions*. JOIV: International Journal on Informatics Visualization* Vol 2. No, 3-2 pp. 215-219 doi: <https://doi.org/10.30630/joiv.2.3-2.144>

S. T. Zargar, J. Joshi and D. Tipper. 2013. A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks, *in IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2046-2069 doi: 10.1109/SURV.2013.031413.00127