

On-Line Information Systems Projects in IPT



L.A.M.P.

By
Dave McGuinness
Urangan State High School

QSITE Conference 2007
IPT Strand

An Egyptian Ant’s Eye View 3

On-Line Information Systems Projects in IPT 3

Why XAMPPLite For Windows? 3

Why Not ASP/Access? 4

Setting Up the Software 4

 Apache and MySQL 4

 PHP - Server Side Scripting Language 4

 PHPMyAdmin – DBMS Web-Based Interface 4

 Script Editors 4

Learning the PHP Language 4

 Integrating PHP with MySQL 5

 MySQL_Query 6

 Making the Move 6

 Teacher as Non-Expert 6

 Scaffold the Task 7

 On-line Information System Project 7

 Conclusions 7

 Useful Texts 8

 Glossary 9

 Web Links 9

An Egyptian Ant's Eye View

“Web-based Programming is a bit like an ant building its hill on top of the great pyramid. It took a lot of guys a very long time to build the pyramid, but you don't have to know much about that to build a great ant-hill. A reasonable idea about the layout of your local block, a bit of organisational ability, some sand, and some hard work should be enough...”

McGoo, 2007

On-Line Information Systems Projects in IPT

Presenter: Dave McGuinness – Urangan State High School (mcgoo@eq.edu.au)

LAMP (Linux->Apache->MySQL->PHP) is an Information Systems model used widely on the Internet today. Students will quickly recognise that they are using applications based on this, or similar models, whenever they are on the net. LAMP's pervasiveness brings with it a context for its use in the IPT classroom. Students respond readily to any IT model which they perceive as being relevant. This has been my experience with on-line information systems in the classroom.

The use of the LAMP model does not require the IPT teacher to develop a whole new toolbox of skills, but rather some minor adaptations of some well-worn techniques developed on desktop IS systems. In fact, it is not at all necessary to get into Linux at all. If you have a Windows server, you can use WAMP instead, or if Internet Information Server is your flavour, "WIMP".

The LAMP model is in widespread actual use on commercial web sites and a good living is to be made in developing such systems. Joomla, an open source content management system developed in PHP/MySQL is recommended by EQ who have provided a Joomla-based template currently being used to build the Urangan SHS web site, <http://www.uranganshs.eq.edu.au>, providing a simple user interface to dynamically edit web content. Another implementation of PHP/MySQL in common use is the course management software, Moodle. A Moodle provides an excellent course delivery and assignment submission interface for IT classes.

The **XAMPP** project (<http://www.apachefriends.org>) provides resources for a range of operating systems. Thus the "X" instead of L or W. The extra P is for Perl if you want that sort of functionality. A very useful variation is called XamppLite that comes with both a Linux and a Windows version.. This is an integrated suite of tools that allows you to run a local web server off a USB memory stick.

This paper will concentrate on the XamppLite model because it allows teachers and students to set up everything on their windows workstation without the need to set up new operating systems. A school network may have an alternative web server model already set up, in which case the teacher will naturally want to use this, but as you will see, the process is much the same, whatever the model.

Why XAMPPLite For Windows?

The most compelling reason to use XAMPP apart from its free availability, is that it is small enough to fit on a 256MB memory stick and runs quite efficiently from there with nothing installed on the local workstation.. This means that keen students can set up a complete working Information System on their home computer which can be easily transferred from system to system, if they so wish.

In a number of ways, Windows-based desktop models (VB/Access or Delphi/Paradox), just do not work well for a significant number of our students. The OOP abstraction layer is just too abstract in some cases, and the object/properties/methods approach can disguise the real purpose of the exercise. It is possible in Delphi or VB/Access to set up a wonderful looking, fairly functional application, with little real understanding of what has been accomplished. PHP/MySQL takes you back to simpler days. Students get in and get dirty with the code and the SQL queries. Feedback from students is that they much prefer working in PHP than Delphi/VB. IPT Students who find the whole desktop IDE just a bit too confusing are very attracted to just cutting code in a simple text editor and then seeing what it looks like in the browser. A little self-reflection will tell some of us that this is the way we learnt to program, and that it was a fairly effective approach in its time. If students are saying that they really understand what they are doing when programming, it is real step forward.

PHP/MySQL provides a tight integration between program code and SQL, without wizards, which should result in a better SQL learning experience for IPT students.

Why Not ASP/Access?

This is a fine approach as well, but open source software is a great idea which we should promote. Not all students have access(sic) to the proprietary software of the MS model.

Setting Up the Software

XAMPP Lite basically just works out of the box. You can unzip it to a folder wherever you like. The default setup has no passwords which is fine for students as no security is really necessary at this level.

Apache and MySQL

In the root of the folder is a small application called xampp-control.exe. This allows you to manually start the Apache web server and MySQL database server.

Inside the xampp folder, you will find a \htdocs folder. This is the document root folder where you put your HTML documents. Open your browser and type in the address <http://localhost> and a default Xampp web page should come up.

MySQL provides some of its own software for managing MySQL from a Windows application rather than from its native command-line interface, but it is wise to stay away from all this and use PHPmyAdmin, an open source PHP-based interface to your MySQL engine. More later...

PHP - Server Side Scripting Language

PHP5 is installed as a module in Xampp. Click on the `phpinfo()` link from the Xampp home page to see the current settings.

PHPMyAdmin – DBMS Web-Based Interface

Now you have PHP working, a good way to test it is with PHPMyAdmin. As most internet hosting providers (including EQ) do not allow external applications to access their MySQL installations, a server-side application is needed to see the databases. An application written in PHP that runs off server is perfect for this and you will find that most hosting sites provide access to PHPmyAdmin as standard. Thus, it is useful for students to be familiar with this application, as they are very likely to encounter it if doing any web development.

PHPmyAdmin is open source and also comes pre-installed in Xampp Lite.

Script Editors

When doing a lot of coding in PHP/HTML it is handy to have purpose built editor. There are many available, but, of course, GPL or free software is preferable for student use. An editor that provides clear line-numbering and syntax-highlighting is essential for PHP as errors are often reported rather cryptically, but line numbers are always given.

ConTEXT is available for download from www.context.cx under a freeware licence. It provides a useful interface for editing PHP with good syntax highlighting and line numbering. It provides very useful HTML, CSS, XML and Javascript syntax highlighting as well. A major bonus with ConTEXT is its multiple document interface which allows you to have several documents open at once.

Notepad2 is a lightweight freeware text editor with syntax highlighting, but it does not have a multiple document interface, so its usefulness becomes limited in larger projects.

Learning the PHP Language

PHP has many constructs in common with Java, Javascript and C++, so anyone conversant in these will have little difficulty in adapting to PHP. Frankly, it is not a big jump from VB, VBscript, or even Pascal either.

Because PHP is designed to output to web browsers, many HTML constructs are used in developing good output. Students should have a basic grasp of HTML tags and CSS before moving in to PHP. Also, some things have to be managed on client-side, so some basic tutorials in Javascript are useful. The similarities in the syntax between Javascript and PHP are such that anything learned in Javascript will naturally carry over. Just get students to end each line in Javascript with a semi-colon. It is not mandatory in Javascript, but it is in PHP, so it is a good habit to start.

W3 Schools (<http://www.w3schools.com>) have some very useful, free, introductory tutorials in HTML, CSS, Javascript and PHP that students can do some homework on.

Once students are happily constructing and publishing web pages in HTML with Javascript, it is a good time to start a discussion of server-side vs. client-side scripting. PHP only works when the pages are run off an http server, so students have to know the difference. Initially, it will seem that there is none as Javascript will do pretty much everything that PHP can, but once you move on to accessing the database, the differences will be evident.

As a teacher some more heavy-weight references will be necessary. Most of us need a book to sit down and look at for some serious learning. *PHP and MySQL Web Development* published by SAMS is a little expensive, but very useful. The authors, Luke Welling and Laura Thomson, are Australians and the text is very readable. You should also download the PHP documentation as a .chm file. You will find yourself constantly referring to this because it is very useful to hyperlink all over the place when you are trying to figure out how to do something. The help file includes plenty of sample code to work from as well. A list of useful texts is included at the end of this document.

Integrating PHP with MySQL

PHP has all the necessary functions for MySQL access built-in so useful web pages are quite easy to build. Every PHP page that has any access to MySQL must include the `mysql_connect` function to connect to the database server and the `mysql_select_db` function to select a particular database. Now is a good time to introduce students to the PHP **include** statement.

Include allows any PHP script to include the contents of another, eg

```
<?php
include("connect.php");
?>
```

The contents of the *connect.php* file might be:

```
<?php
    // connect to mysql server
    $db = mysql_connect ("localhost", "student", "password") or die("Cannot connect: ".mysql_error());
    //select the database
    $connect_db = mysql_connect_db("mydb") or die("Cannot select db: ".mysql_error());
?>
```

This construct allows just the include statement to be placed at the beginning of every file that uses the database without including all the password information in every file. Imagine the work involved if there was a password change and every file had to be updated individually. This approach also allows the teacher to de-emphasise the need for students to use passwords to access a school server. They can be just given the URL of the *connect.php* file without needing to look into it. A line to include to a secret *connect.php* will give their scripts access to the database. Eg,

```
<?php
    include(http://www.server.com/teacherssecretfolder/connect.php);
    // define a query
    $Query = "Select * from Students";
    // run query
    $Result = mysql_query($Query);
    // set $iRows = number of result rows
    $iRows = mysql_num_rows($Result);
    //
?>
```

To demonstrate a range of possible code structures using PHP/MySQL, an SQL query application is provided as an example. This example could be developed by students as a learning experience to prepare them for developing their own code in a major project.

MySQL_Query

This application uses a series of PHP files to allow access to selected databases to run SQL queries. An installation of this code can be found on the Urangan SHS website at:

http://www.uranganshs.eq.edu.au/home/dmcgu11/mysql_query

It gives access to a limited range of databases found in the textbooks of Kevin Savage and Offer and Thompson. It is useful for students who use these texts to run some queries from home. All SQL constructs suggested by the IPT syllabus seem to work fine.

Teachers with access to a school web server running MySQL 4.1 or above could use this code to allow students to run all SQL queries including sub-queries as well as insert, update and delete statements. PHPmyAdmin allows the manager to allocate any level of privileges on individual databases on a server to individual users if so desired. A range of possibilities for providing useful learning experiences exist within this framework.

At present MySQL_Query consists of the following files:

index.php	Main Page
header.php	Provides header for all pages
footer.php	Provides footer for all pages
db_select.php	Provides list of available databases for selection
db_connect.php	Connects to selected database
my_query.php	Main query entry page. Also allows viewing of tables
result.php	Displays result of queries
help.php	Basic help pop-up
view_table.php	Displays tables for viewing
result_table.php	Include file for result.php
result_html.php	Include file for result.php, creates output for copying
view_html.php	Displays output for copying
query.css	Cascading Style Sheets file for all pages.

MySQL_Query is open-source and the full source code, plus data-dumps of the some databases are included in mysql_query.zip, available for download from:

http://www.uranganshs.eq.edu.au/home/dmcgu11/mysql_query.zip

Making the Move

Most IPT teachers have a fair bit of history invested in one programming environment or another. Moving on to something new requires teachers to commit to the significant workload involved in mastering a whole new language, so why do it? IPT students deserve exposure to a range of ideas. Presenting the whole course in one language across IIS and SSE is probably letting them down to some extent.

Teacher as Non-Expert

Try looking at the problem from a different angle. Do not try to master the new environment. Get a few basics together, enough to be a reasonably effective facilitator, and give the task of learning the environment to the students. **Refuse** to be the expert. They will be better off in the end. Make it your practice, as a teacher of junior and senior IT classes, **not** to master any software you require your students to use, rather become a **master of finding answers**, searching help files and web-based tutorials and leading students to where they can find answers for themselves. Let them take the responsibility for their own learning. If you do happen to know a significant amount, keep it quiet.

A good approach is to gather together a set of clear and student-accessible tutorials that students can use as needed. W3Schools.com (<http://www.w3schools.com>) provides a very accessible and useful resource for finding very helpful code samples. php_manual_en.chm is a very useful compiled help manual that can be downloaded from http://www.php.net/get/php_manual_en.chm/from/a/mirror.

Start students off with some sample code that includes sufficient examples to get them 75% of the way through the project and then challenge them to make up the difference. In this way you will have an assessment item that **does** allow clear discrimination between students at all levels.

This is not something that should be sprung on students in Year 12. Self-directed learning needs to be developed as a skill from as early as possible, if students are going to recognise their innate ability to make on their own.

Scaffold the Task

Once in Year 12, a major project that runs for eight or nine weeks is a major undertaking for anyone. Weaker students, who trip up in the early stages, have no chance of recovering in the long run. To give everyone a fair chance, a good approach is to break up the project into significant sections, then after each section is submitted by students, give them all a level playing field for the next section. Below is a short description of how this might be done for an on-line information system major project. Possible assessment task and criteria sheets for the project are included as an appendix to this paper.

On-line Information System Project

Stage	Tasks for Completion	Continuing	Follow Up
1	Timeline, Problem Definition, Solution Specification, Elementary Sentences,	Journal, Documentation	Provide Elementary Sentences
2	CSD, ONF, Data Dictionary	Journal, Documentation	Provide Data Dictionary
3	Create and Populate Database	Journal, Documentation	Provide Database
4	Implementation of Web Site in PHP/HTML, Testing, Evaluation.	Journal, Documentation	Provide Marking

A project scaffolded in this manner will give students the best chance to show what they **can** do rather than what they **cannot**. A further major benefit of this approach is that students who struggle early are more likely to persist in later stages where they feel they still have a chance. A major benefit to teachers is that the staged submission provides effective drafts of student progress on which to base assessment if in the end they fail to complete. When everything is due right at the end, the most challenged students are least likely to complete, leaving teachers with the embarrassment of nil evidence on which to base exit levels.

Conclusions

Switching from the tried and true desktop database model of information system to a client-server, internet-based model provides teachers and students with both significant challenges and significant benefits. On the whole, the latter far outweigh the former. Given the task of “choosing your own topic” on a desktop database, a minority of IPT students will be able to come up with a worthwhile concept. Let them come up with an idea for a web database and most of them will be enthused.

Generally, I believe that “Choose Your Own Adventure Projects” are fine for the able students, but hopeless for the less able, and very difficult for the teacher. After years of hoeing that row, I now have everyone on the same topic where I can steer the ship and provide the scaffolding with some success. It is a much less hit-and-miss approach with a significant increase in student success and satisfaction. Students are much happier showing what they can do, rather than proving what that they cannot do.

Useful Texts

PHP and MySQL Web Development, Luke Welling and Laura Thomson, SAMS Publishing.

spring into PHP, Steven Holzner, AddisonWesley.

spring into HTML and CSS, Molly E. Holzschlag, Addison-Wesley

Javascript: A Programmer's Companion from the Basics through DHTML, CSS, and DOM, Stefan Koch, Wiley.

DHTML Utopia: Modern Web Design Using Javascript and DOM, Stuart Langridge, Sitepoint.

HTML Utopia: Designing Without Tables Using CSS, Dan Shafer, Sitepoint.

Simple Program Design, Lesley Anne Robertson , Nelson

Information and Intelligent Systems, Kevin Savage, www.edit.net.au

Glossary

LAMP	Linux -> Apache -> MySQL -> PHP
WAMP	Windows -> Apache -> MySQL -> PHP
WIMP	Windows -> IIS -> MySQL -> PHP
HTTP	HyperText Transfer Protocol
Web Server	Any application that provides web pages via HTTP
MySQL	MySQL Database Server (<i>Open Source</i>)
Apache	Apache Web Server (<i>Open Source</i>)
IIS	Internet Information Server (<i>Microsoft</i>)
PHP	PHP Hypertext Pre-Processor (<i>Open Source</i>)
PHPmyAdmin	A PHP interface to the MySQL DBMS (<i>Open Source</i>)
ConTEXT	A syntax-highlighted text editor for PHP/CSS/HTML (<i>Open Source</i>)
Mambo	A PHP/MySQL based Content Management System
Moodle	A PHP/MySQL based Course Management System

Web Links

McGoo Software	http://www.mcgoo.com.au
McGoo's MySQL_Query	http://www.eq.edu.au/dmcgu11/mysql_query
Apache	http://www.apache.org
Apache Friends	http://www.apachefriends.org
MySQL	http://dev.mysql.com
PHP	http://www.php.net
Context	http://www.context.cx
NoteTab Lite	http://www.notetab.com
Moodle	http://www.moodle.org
Mambo	http://www.mamboserver.com