

**NAPIER UNIVERSITY**  
**SCHOOL OF COMPUTING**

**Article I. UNDERGRADUATE PROJECT REVIEW REPORT**

**Section 1.01 1. Student details**

**Name:** Cedric LEO

**Matriculation no:** 00210536

**Programme:** Multimedia Systems      **Preferred e-mail address:** cedricleo@hotmail.com

**Dissertation title:**

**Date:**

**2. Project details**

**Title:** Evaluation of Dynamic Reconfigurable WWW-content Prototypes

**Scope and aims of the study** (Max. 150 words)

The aim of the project is to develop prototypes in dynamic content WWW pages using XML and database-systems.

The research will be to look for all technologies like XML, databases (MySQL, Microsoft Access) – in the same time, investigate database-driven -, Server Side Processing (Microsoft ASP, PHP) and the user side application (Flash with ActionScript and JavaScript). They have been proven to work but I have to compare all of them (it will be the research part to collect a lot of information about each of these), decide on the technologies to achieve my aim (which ones will give the best results?), then evaluate each by developing little prototypes of the implementation technologies (the interaction of the chosen ones and the advantages or the disadvantages that will discover) and finally elaborate a critical approach (i.e. during the elaboration of the project, some problems will appear, perhaps some of them will not be solved, so “will they be the next steps to perfect the prototypes and give them more reliability and increase their performances?” and so on).

**3. Progress to date** (Max. 5 sides of A4)

**a) Research undertaken**

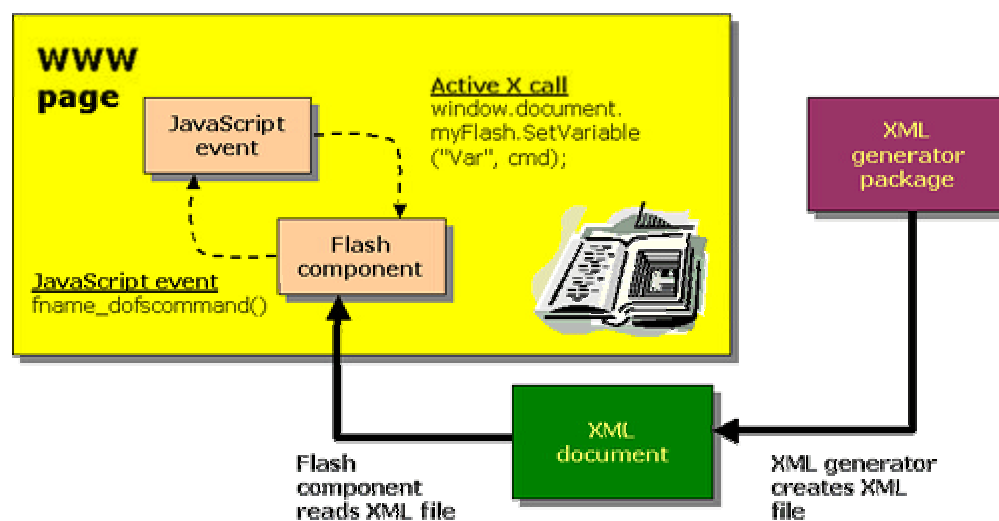
The first step of my research was to investigate the possibilities of XML and compare them to the ones of the well known MySQL database instead of Microsoft Access because it is not portable across systems and so a comparison won't be appropriate.

Their main possibilities are described as follows:

- XML allows updating of content without thinking of the presentation or future processing. It can generate automated multiple presentations, sorting and so on, on multiple medias. The parsing is explicated by specific tags, independent of the application that is used.
- MySQL allows accessing to data more quickly than files and you can extract easily data following some criterions. Moreover, it provides systems of integrated privileges.

In addition, to develop my prototypes, I needed to integrate a database to a server side script for access to the database (some books from the library were helpful). First, I have chosen PHP Hypertext Pre-Processor because I have already used it and a lot of resources are available on the Internet. My second choice is focused on Active Server Pages (ASP) for testing the Microsoft technology and comparing it with a free server side script like PHP. Moreover, Practical Extraction and Report Languages will be investigated because is a very powerful server side script and is very portable and support a wide range of databases. Finally, if I have the time to implement a prototype with Java Server Pages, I will make it because JSP is as powerful as PERL and it is a good opportunity to test the integration of a database in these scripts. The disadvantages of JSP are there are few resources and I don't know if there is a provider that accepts this server side script.

Moreover, authoring software like Flash can develop dynamic websites and a recent opinion poll on 900 people about the favourite multimedia technology has shown that 38% of them favour *Macromedia Flash* and 15.5% of them prefer DHTML and *JavaScript*. Moreover the authoring software can read XML file. That's why an investigation should be done with the help of JavaScript allowing the user the handling of some data from the flash application. This following figure from Bill Buchanan's website shows you that I explained previously.



Finally, Flash can interact with PHP. But does that interaction give Flash more possibilities for handling of data?

From all these information, I have deduced several prototypes which should be implemented:

- ☞ Prototype number 1: Flash, PHP & XML (an authoring software and a server side scripting language...is it a good solution for dynamic content?)
- ☞ Prototype number 2: PHP with XML (to use objects and classes for parsing).
- ☞ Prototype number 3: Flash, XML & JavaScript (Flash creates dynamic applications, but is there the same kind of interaction Flash and XML like there is between PHP and XML?)

- ⌘ Prototype number 4: PHP & MySQL (to compare the possibilities between the database systems and structured files).
- ⌘ Prototype number 5: ASP & XML (what does Microsoft provide for parsing in comparison with its competitor PHP?)
- ⌘ Prototype number 6: PERL / XML (they are both portable across systems so it should be inevitable to implement such a prototype).

If I have time and I can find a provider that accepts JSP, the following prototype will be implemented:

- ⌘ Prototype number 7: JSP & XML (is it necessary to compare the classes and objects of PHP with the ones of JSP? I think so).

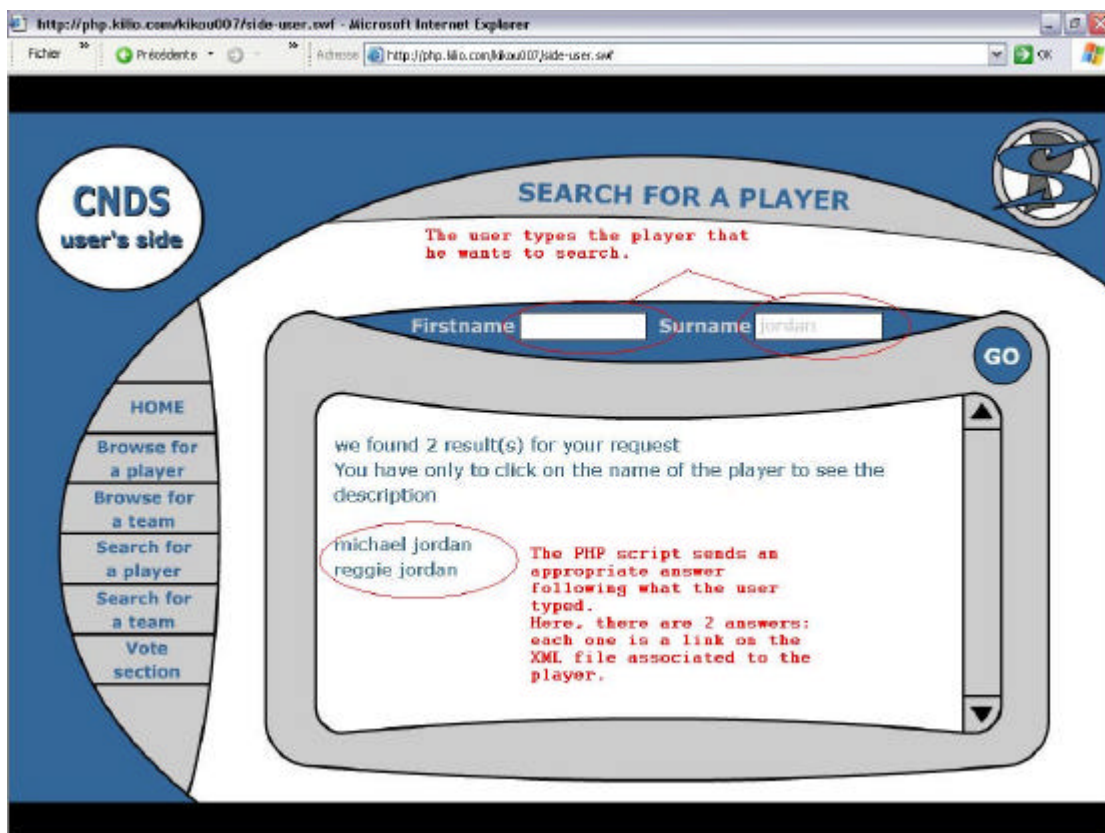
At the end, we will have 3 kinds of comparisons:

- ⌘ Between MySQL / XML (database systems / structured files)
- ⌘ Between the parsing of Flash and PHP (authoring software and server side script)
- ⌘ Between different server side scripting languages (PHP, ASP, PERL and maybe JSP)

### b) Practical work undertaken (e.g., requirements capture, design, implementation)

The aim of my CNDS project was to develop a prototype that could read XML files and display their content in WWW content: so I have already implemented the prototype number 1. There are still some bugs that will be fixed later following my Gantt chart.

The prototype 1 is divided in 2 parts: server's side and user's side, you can see a picture of the client's side as follows:



In addition of reading XML file, the user's side as the server side can modify XML files via PHP scripts, here there are a couple of pictures about the system of vote in the user's part. The first one shows what the user selects (appendix 1) and in the second one, we can see the effect of the user's answer in the XML file (appendix 2).

Moreover, several small prototypes have been already implemented testing different possibilities of development:

- Flash read XML that is one of the main study of the prototype 3
- I have already started to test some JavaScript functions with Flash.
- The reading and the editing of PHP on XML files following 2 ways, one of which is the use of oriented objects and classes. Useful for the prototype number 2.

#### **4. References** (Must be cited in 3a)

2 books to study Flash and ActionScript:

"Flash 5 Interactivity and Scripting" by Nigel Chapman, Wiley

"Foundation ActionScript" by Sham Bhangal, friends of ED

3 books to investigate the interaction Flash/server side scripting languages:

"Server-Side Flash - Scripts, Databases, and Dynamic Development" by William B.Sanders & Mark Winstanley, Hungry Minds

"Flash 5 magic with ActionScript" by J. Scott Hamlin & David J. Emberton, New Riders

"Flash 5 Dynamic Content Studio" by Philippe Archontakis, Friends of ED edition

A book about server side programming and XML:

"XML Processing with Perl, Python, and PHP" by Martin C. Brown, Transcend Technique

A book about server side programming and a database system:

"PHP&MySQL" by Luke Welling & Laura Thomson, by CampusPress

A table to compare the server-side scripting languages:

<http://www.ebiz-intellect.com/page.cfm?onumber=36>

Several pages describing server-side scripting languages:

<http://hotwired.lycos.com/webmonkey/99/46/index1a.html?tw=programming>

#### **5. Self-assessment of progress** (Max 150 words)

Following the Gantt chart, I am in late. Indeed, I took a lot of time to implement the first prototype (Flash, PHP & XML). This one works very well, except for few bugs on the control of text typing in the admin part. But I used that time to prepare myself in the implementation of 2 other prototypes that are number 2 (PHP/XML) and 3 (Flash/XML/Java Script). Moreover, I have already investigated a new way to read XML files with PHP and I have started to implement the second prototype using oriented objects and classes. The third one is only the set of several programs that I have developed during the implementation of the first one (read XML files only following 2 ways).

In addition, if I don't have enough time to terminate my investigation, I will develop only 5 prototypes instead of 6.

#### **6. Gantt chart**

Please attach a Gantt chart showing actual against estimated progress to date, and planned activities to hand in.