Author(s): Draganova, Chrisina
Article title: PHP and Wireless Web Development
Year: 2006
Citation: Draganova, C. (2006) 'PHP and Wireless Web Development', In: JICC 10, 10th Annual Conference of Java in the Computing Curriculum, London.

PHP & Wireless Web Development

Chrisina Draganova
Department of Computing, Communication Technology and Mathematics
London Metropolitan University
100 Minories, London EC3 1JY
c.draganova@londonmet.ac.uk

Abstract

In this paper we consider how teaching traditional web development with PHP can be extended to wireless web development. Wireless technology is a new exciting development in the computing field and, as such, should be addressed in the computing curriculum. This paper suggests a way of introducing topics on wireless web development using wireless technologies and the server side programming language PHP; it gives an overview of the main wireless standards and presents examples using these technologies in conjunction with PHP. The examples are aimed to be used in the "Internet Technologies" module for Foundation Degree students in Computing at London Metropolitan University.
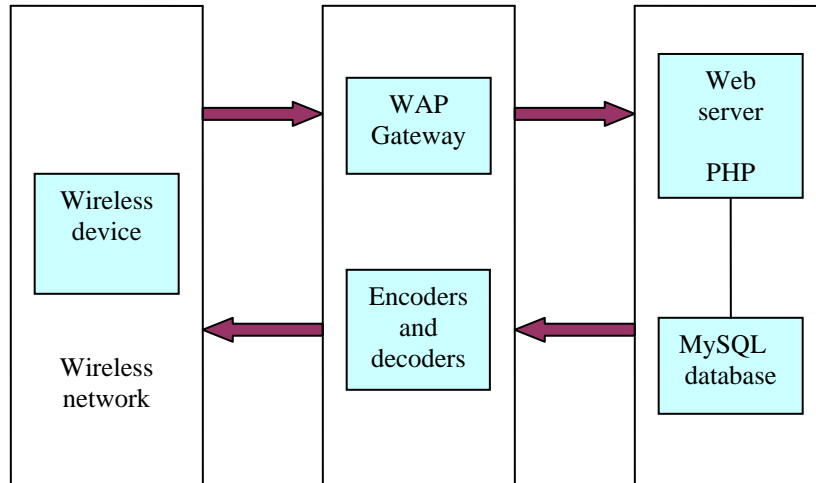
1. Introduction

Wireless technology is a significant development in the field of computing and it is used in all areas of our lives. Mobile devices such as personal digital assistants and mobile phones are part of our daily life communications, information and business transactions. Consequently the industry requires graduates to have knowledge and skills in developing applications for wireless devices and understanding of the specific usability issues related to these technologies.

This paper aims to share some ideas of how to introduce wireless web development with PHP and illustrates examples enabling students to understand the basic implementation issues. The paper is organised in four sections including the introduction. The next section gives an overview of the main wireless standards and their use in conjunction with PHP. The third section discusses issues related to teaching wireless web development and demonstrates some practical examples and the last section gives the conclusions.

2. Overview of wireless technologies

2.1 Wireless Application Protocol (WAP) is a protocol that is designed to allow communication of handheld devices such as PDAs or mobile phones on wireless networks and access to the Internet. WAP architecture includes different layers including application, session, transaction, security, transport and network layers providing application development, security and connectivity support. The communication between a WAP-enabled wireless device and a web server is very similar to that between a traditional web browser and a web server with one additional step that involves decoding and encoding messages sent to and from wireless devices via a WAP gateway. This step is required because the wireless Internet access requires the transferred data to be converted from text to binary format and encrypted. The WAP gateway is usually hosted by the telecommunications company providing the wireless phone service. A typical WAP transaction is illustrated in Figure 1. A detailed description of WAP can be found in [1], [2].

Figure 1: WAP transaction

2.2 Wireless Markup Language (WML) is a markup language based on XML (eXtensible Markup Language) and is designed to be used on a variety of mobile devices. WML is similar to HTML, but the WML documents need to be well-formed (i.e. conform to the XML syntax rules). WML enables communication with wireless web browsers using the WAP protocol. The basic structure of an WML document includes three fundamental elements: a header indicating the version of XML and the XML document type definition (DTD), a root element called deck <wml> </wml>, and one or more elements- cards <card> </card> nested in the deck. Each card is displayed one at a time by a WML enabled browser. WML includes a number of tags which are similar to those in HTML such as paragraph tag <p> </p>, table tags <table> </table>, <tr> </tr>, <td> </td>, <input>, <select>, <a>, <img>, etc. Elements responding to events include <do>, <onevent> and <timer>. Figure 2 illustrates a simple WML document and how this document is displayed on a wireless device simulator. More information on WML is available in [1], [2].


```
< ?xml version= »1.0 » ?>
        < !DOCTYPE wml PUBLIC « -//WAPFORUM//DTD WML 1.2//EN »
« http ://www.wapforum.org/DTD/wml12.xml »>
        <!—begin deck ➔
<wml>
  <!—begin card ➔
        <card>
          <p>
            Hello World
          </p>
        </card>
</wml>
```

http://meglos.lgu.ac.uk/~draganov/wireless/hello.wml

Figure 2: hello.wml - source code and a display on a Nokia simulator

2.3 A Wireless Browser is a browser built to parse and display WML pages. A wireless web browser has many features that are typical for a conventional desktop web browser such as browsing through hyperlinks, rendering of text and images, secure transactions. However the content design for such a wireless browser is very different from the conventional one due to the limited display space of mobile devices and limited processing power. In order to improve the user experience of wireless devices the following guidelines are recommended in [4]: limiting graphical content, utilizing empty space, limiting or eliminating the use of widgets, enabling appropriate word-wrapping, placing non-essential links at bottom of page, displaying error messages in an obvious way.

2.4 PHP and WML. PHP is a scripting programming language, designed for Web development. It is used in combination with Apache's Web server, which runs almost 70% of sites on the Web. PHP provides support for different database systems, it has extensive built in functions for common tasks in web development, it is available on different OS platforms, and it is an open source product. The latest version is available free for download from [8]. PHP files may contain text, HTML tags and scripts and by default these are returned to the web browser as plain HTML. When developing a WML page that contains PHP script the file must include at the beginning the **header ("Content-type: text/vnd.wap.wml")** function in order to set the Content-type HTTP header to the correct form; by default this is set to text/html [3]. The rest of the generated content must be a valid WML document. We can use all the Figure 3 illustrates the source code for **hello.php** script which generates "Hello World" for display on a wireless browser.

```php
<?php
    // set Content type
    header("Content-type: text/vnd.wap.wml");
    echo("<?xml version=\"1.0\"?>");
    echo('<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD  WML 1.2//EN"'
    . '"http://www.wapforum.org/DTD/wml12.xml">');

    echo('<!—begin deck ➔'
    .'<wml>'
    .'<!—begin card ➔'
    .'<card>'
    .'<p>'
    .' Hello World'
    .'</p>'
    .'</card>'
    .'</wml>');
?>
```

Figure 3: hello.php – source code

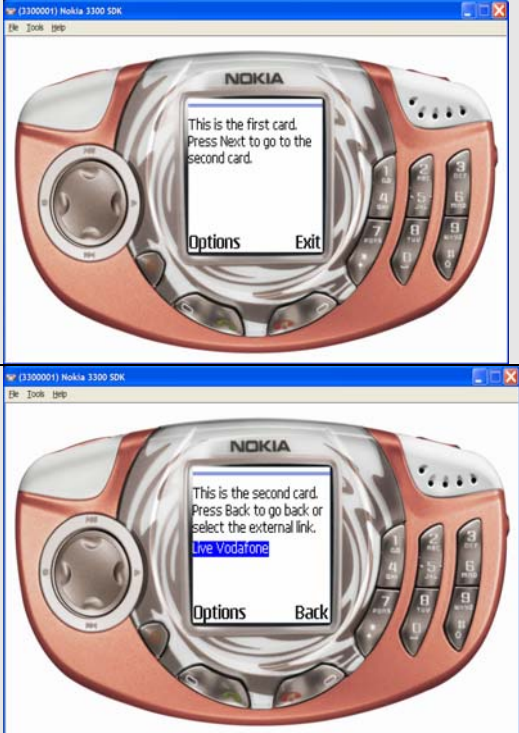## 3. Teaching wireless web application development using PHP

Wireless and mobile computing is a major technology development and there are already a number of universities which offer such degree courses. For example, a new degree course on Mobile and Wireless Computing has started recently at London Metropolitan University. However in existing computing courses topics on wireless technologies are not always considered or offered as options. Undoubtedly there is an interest among the students in Computing to learn and experience developing not only traditional web sites but wirelesses web applications as well. For example this academic year we have a number of 3rd year computing students who choose to do their final year project namely in the area of wireless web development. Ideally we should offer these students an optional module that provides the necessary knowledge or at least consider the topic in existing web development modules.

In this section we suggest WML/PHP/MysQL examples which are to be used in the lab sessions for the "Internet Technologies" second year, second semester module for "Foundation Degree in Computing". The "Foundation Degree in Computing" is a vocational type of degree which aims to equip students with work-specific skills, relevant to a particular sector of industry [7]. In line with these aims the "Internet Technologies" module provides the students with knowledge and skills to develop Internet solutions for business requirements. Students studying this module have already passed a pre-requisite module in the first year "Website Design and Implementation" and they have: the skills of designing and developing a static web site, conforming to the current web design principles and standards; understanding of the structure and technology of the Internet; experience of using HTML, JavaScript, CSS and XML/XSL. In the first semester of their second year all of the students on the "Foundation Degree in Computing" have completed full time industrial placements in software development or other related areas. The majority of the students have worked during the industrial placements on web design and development projects and have extended further their knowledge on the subject.

In the Internet Technologies" module the students are introduced to dynamic web site development using PHP and MySQL. During lab sessions they gain experience with a range of PHP and MySQL examples demonstrating fundamental techniques such as manipulation of form variables and arrays, use of PHP functions, code re-use, creating database connections and queries. More advanced topics are considered in later session including: authentication, security, cookies, sessions and shopping cart application. For each topic the students are given suitable practical examples that they have to modify slightly, deploy and test. Already having the knowledge and experience of HTML, XML, PHP and MySQL students have sufficient background to be introduced to the main wireless technology standards and to the development of wireless web applications using PHP. During the lecture session students are presented with an overview of WAP architecture and WML. By demonstrating and comparing HTML and WML simple examples students are introduced to the development of a wireless web application. To give students basic skills of how they can further develop the content of their web sites so these become accessible through wireless browsers and get hands on experience we suggest some simple PHP, WML and MySQL related examples demonstrating the structure of a WML document, navigation, form submission and accessing a database [1], [3]. During the lab sessions the students are typically asked to try these examples and to modify some parts of them in order to add more functionality and solve small problems. To test the examples students could use a WAP simulator [6] or WAP enabled mobile phones. These examples are easily translated to any other server-side language (e.g. JSP) and could be re-used in similar Web application development modules.
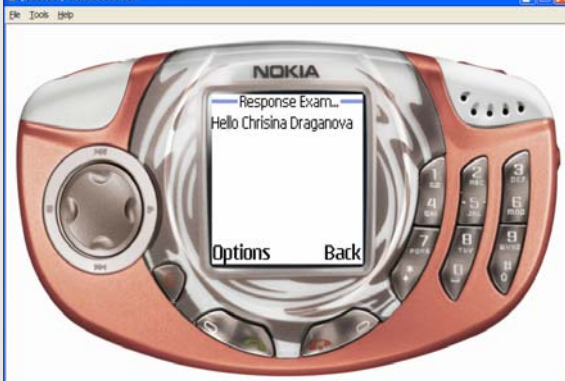
Introducing the topic of wireless web application development is relatively straight forward in this module given that the students have already quite extensive experience with the traditional web technologies. By doing so students are given the opportunity to gain insight in the rapidly expanding mobile technology area and reinforce the understanding of fundamental concepts in Internet computing such as data models, encapsulation, network protocols, distributed systems and usability issues.

Example 1 - Internal linking and external hyperlink This first example is a WML document which demonstrates how to do linking of two cards and an external link. The external link has the same syntax as HTML hyperlink.


```
<?xml version="1.0"?>
    <!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.xml">
<wml>
<!-- using internal and external links -->
<card id = "card1">
        <do type = "accept" label = "Next">  <!— internal link to card2➔
         <go href = "#card2" />
        </do>
        <p>
         This is the first card.
        Press Next to go to the second  card.
        </p>
</card>
<card id="card2">
        <p>
        This is the second card. Press Back to go back or select the external link.
        </p>
         <do type = "accept" label = "Back">  <!— internal link to card1➔
          <go href = "#card1" />
         </do>
         <p>
        <!—external link➔
        <a href="http://live.vodafone.com">Live Vodafone</a>    </p>
</card>
</wml>
```

http://meglos.lgu.ac.uk/~draganov/wireless/links.wml

Figure 4 WML internal and external links for navigation

Example 2 - **PHP, WML and passing parameters**. This example has two files: **request.php** and **response.php** which demonstrate a WML form with two parameters and rendering the parameters passed through the form, respectively.

```php
<?php // request.php
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version=\"1.0\"?>";
echo "<!DOCTYPE wml PUBLIC \"-//WAPFORUM//DTD WML 1.2//EN\"". "
\"http://www.wapforum.org/DTD/wml12.xml\">";
?>
<wml>
 <card id="request" title="Request Example">
  <p> <!—form➜
    <do type="accept">
     <go href="response.php" method="post">
       <postfield name="FirstName"
         value="$firstname"/>
       <postfield name="SecondName"
         value="$secondname"/>
     </go>
    </do>
First Name: <input title="First Name"  name="firstname"/> <br/>
Second Name: <input title="Second Name"  name="secondname"/> <br/>
   </p>
 </card>
</wml>
```

http://meglos.lgu.ac.uk/~draganov/wireless/request.php

```php
<?php // response.php
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version=\"1.0\"?>";
echo "<!DOCTYPE wml PUBLIC \"-//WAPFORUM//DTD WML 1.2//EN\"". "
\"http://www.wapforum.org/DTD/wml12.xml\">";
?>
<wml>
 <card id="response" title="Response Example">
<p>
 <?php  print 'Hello '.$FirstName.' '.$SecondName; //display parameters
 ?>
</p>
</card>
</wml>
```

http://meglos.lgu.ac.uk/~draganov/wireless/response.php

Figure 5 Passing and displaying parameters

Example 3 - **PHP, WML & MySQL** This example includes two files: **search.php** and **results.php**. In **search.php** a WML form is rendered which passes one parameter on submission to **results.php**. In **results.php** we have a connection to MySQL database, a query to a specific table based on the search parameter passed by **search.php** and finally rendering the results returned by the query.
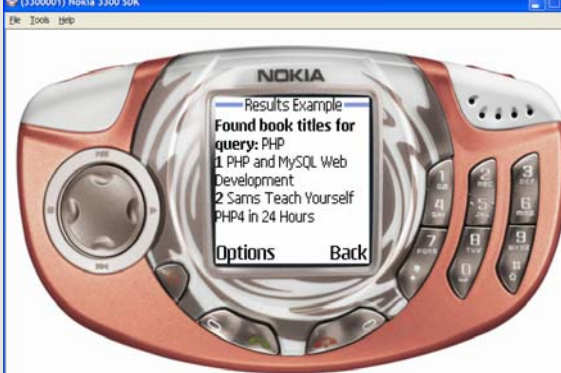
```
<?php
// search.php
header("Content-type: text/vnd.wap.wml");
echo "<?xml version=\"1.0\"?>";
echo "<!DOCTYPE wml PUBLIC \"-//WAPFORUM//DTD WML 1.2//EN\"". "
\"http://www.wapforum.org/DTD/wml12.xml\">";
?>
<wml>
 <card id="search" title="Search Example">
<p><do type="accept">      <!—form➔
<go href="results.php" method="post">
   <postfield name="BookTitle" value="$book">
 </go></do>
Search for a book title: <input title="Search" name="book"/> <br/> </p>
 </card>
</wml>
```

http://meglos.lgu.ac.uk/~draganov/wireless/search.php

```
<?php  // results.php
header("Content-type: text/vnd.wap.wml");
echo "<?xml version=\"1.0\"?>";
echo "<!DOCTYPE wml PUBLIC \"-//WAPFORUM//DTD WML 1.2//EN\"". "
\"http://www.wapforum.org/DTD/wml12.xml\">";
?>
<wml>
 <card id="results" title="Results Example">
<p>
<?php
$link = mysql_pconnect("localhost","204user","204pass") or exit("Could not
connect"); // mysql connection
mysql_select_db("qf204");
$query = "select title from books where title like '%$BookTitle%'";
$result=mysql_query($query) or die("Query failed"); //execute query
$num_results = mysql_num_rows($result);
if ($num_results < 1) echo "No data found";
else{ echo '<b>Found book titles for query: </b> '.$BookTitle.'   <br />';
$i=1;
while ($row = mysql_fetch_array($result)){ echo '<b>'.$i++.' </b>'.$row['title'].'
<br />'; // render results
}  }?></p>
 </card>
</wml>
```

http://meglos.lgu.ac.uk/~draganov/wireless/results.php

Figure 6 Search in MySQL database and display the results

Note: In order for the Apache server to make the correct interpretation of the MIME type extensions related to WML we add the following MIME types: AddType application/x-httpd-php .wml; AddType application/x-httpd-php3 .wml; AddType text/vnd.wap.wml .wml ; AddType image/vnd.wap.wbmp .wbmp into an **.htaccess** file and place this file into the directory where the WML pages reside [3]. Modifying the **httpd.conf** file if security permissions allow also could do this.

4. Conclusions

In this paper we have given short overview of WAP, WML and the use of WML with PHP. We have presented examples how to do navigation, form submission and accessing a database using WML, PHP and MySQL. This topic is included in a second year module "Internet Technologies" for the Foundation Degree in Computing at London Metropolitan University. The module is currently running and an evaluation will be made at the end of this semester.

One of the main principles in designing a computing curriculum emphasised in [5] states that there must be an ongoing review process of the curriculum as a response to the rapid evolution of computer science and the constant changes in technology. Following this principle and making our programmes up to date is particularly important in order to attract students and improve the recruitment of computing undergraduates. Introducing wire-

less technologies is a way of maintaining the relevance of web development modules and providing students with skills demanded in industry.

5. References

[1] H. M. Deitel, P. J. Deitel, T. R. Nieto, K. Steinbuhler, Wireless Internet & Mobile Business, How to Program, Prentice Hall, 2002

[2] Dr Gordon W. Skelton, Wireless Application Development, Thomson Course Technology, 2003

[3] Building WML Applications Using PHP, http://www.zend.com/zend/tut/wap.php, W.J. Gilmore, 2000

[4] Designing Web content for mobile browsers UCD Recommendations for optimizing information presentation on mobile devices http://www-128.ibm.com/developerworks/library/wi-web/, James Cotton, Patrick Commarford, 2005

[5] Final Report of the Joint ACM/IEEE-CS Task Force on Computing Curricula 2001 for Computer Science http://www.computer.org/education/cc2001/final/index.htm, 2001

[6] Forum Nokia – Developer Resources, http://www.forum.nokia.com/main.html, 2005

[7] Foundation Degrees in the UK http://www.foundationdegree.org.uk/, 2005

[8] PHP http://www.php.net, 2005