

5COM0085

Computer Science Development Exercise

5COM0086

Information Technology Development Exercise

Developing the D4 SAMS-P from the specimen application in VF1

L 29, Week 20, February 2012

- Technology perspective

**Your Lecture is
about to begin**

Prepare yourself

**Respect the
University
Student Code of
Conduct**

**Please switch
off your
mobile
phone**



**Please remove
headphones
& switch off
MP3 players**



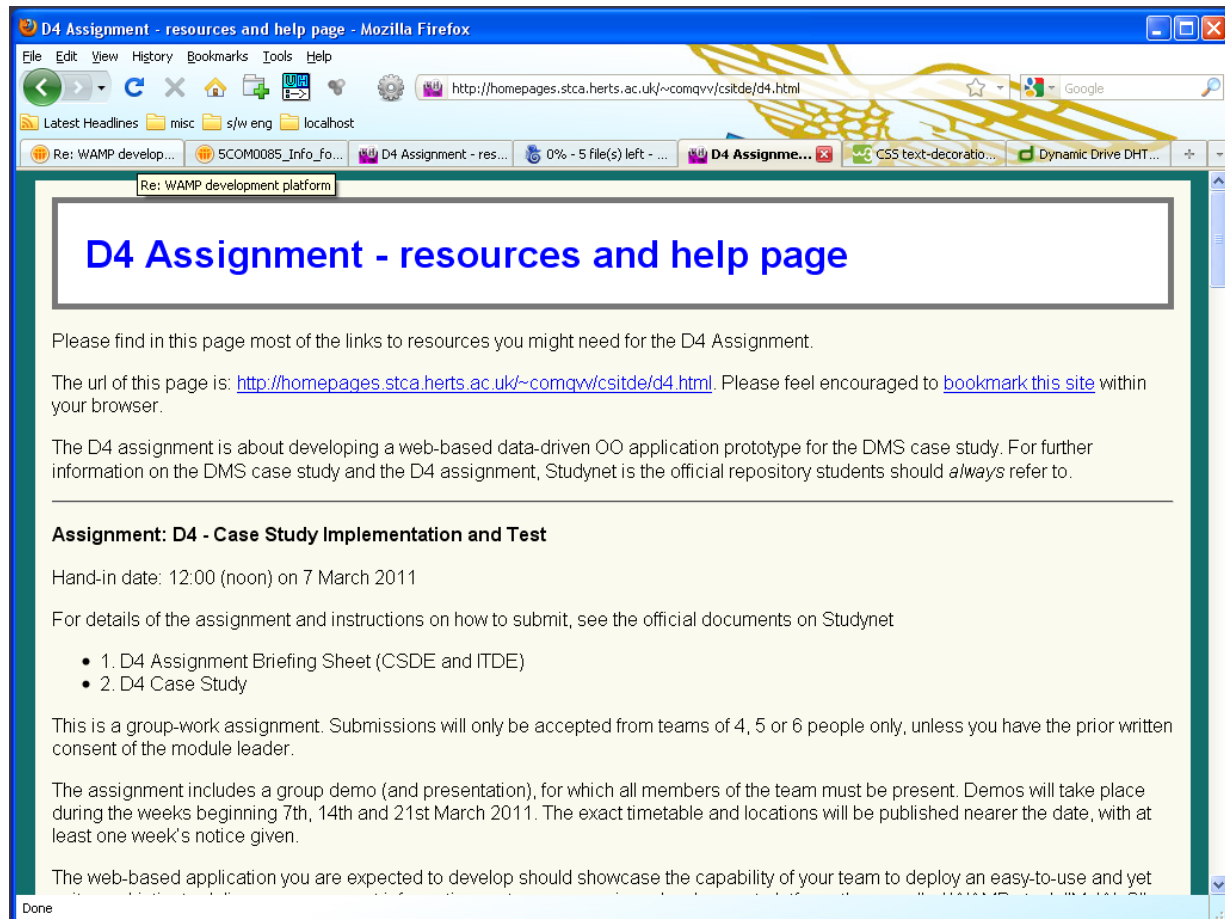
**Please be
quiet and be
prepared to
study**



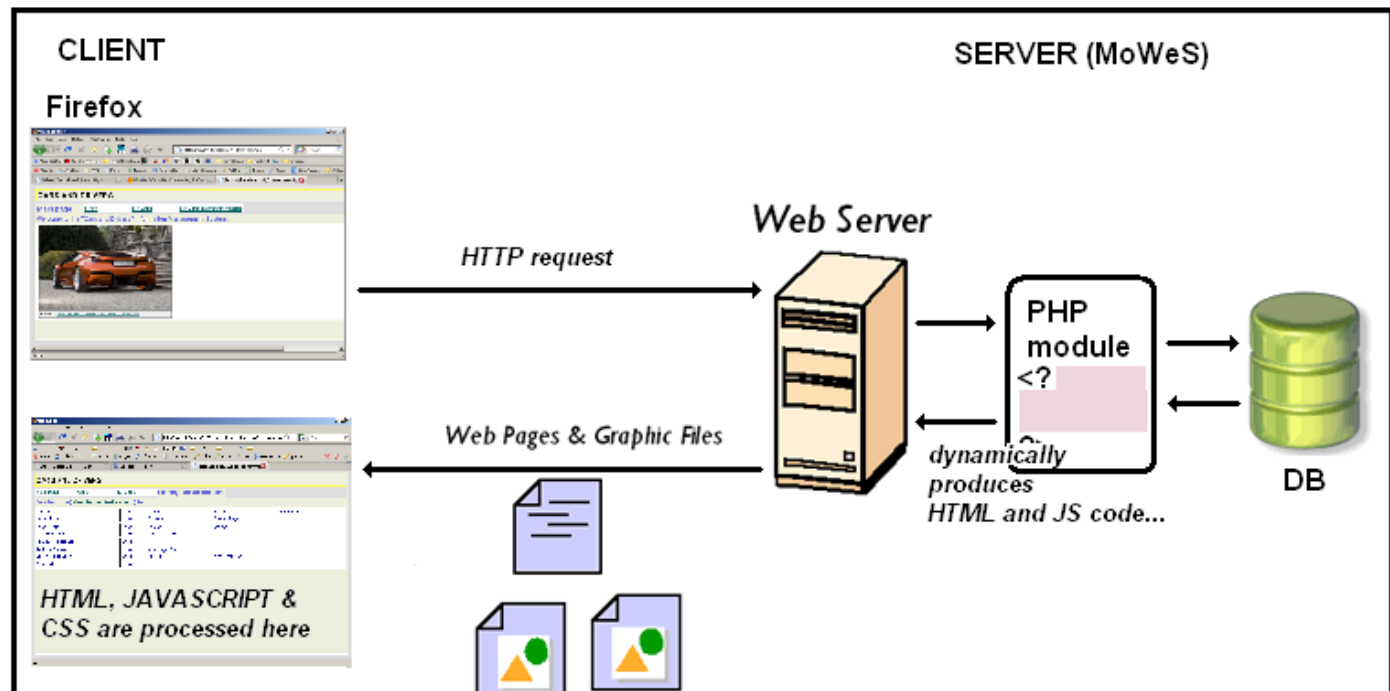
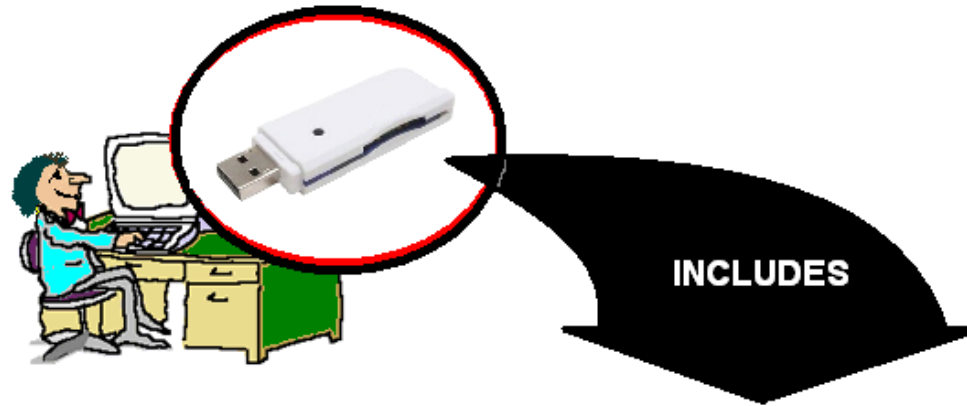
D4 Assignment - resources and help page

Available at:

<http://homepages.feis.herts.ac.uk/~comqv/csitde/d4.html>



An excursion into the MoWeS platform



Resources (to be used during and after these lectures)

Ensure you have available within your MoWeS platform the specimen application develop by using VF1. If not, you can download it from:

<http://homepages.feis.herts.ac.uk/~comqv/vf1>

Copy and unzip the file **vf1_application.zip** within the folder **www**, and call it with the name of your application (e.g., “application”), as in **C:\wos_csitde\www\application**.

Within the vf1_application folder

Four folders and several files to be edited

- In the vf1_application root folder:
 - index.php,
- In the folder "include":
 - vf1-1024x768.css,
 - vf1.js,
 - many other files providing the feel & look of the application (images, calendar's css, etc.)
- In the folder "model":
 - model.php
- In the folder "view":
 - main_page_customization.php

Launch the MoWeS platform, open Firefox and go to...

http://localhost:8080/vf1_application/

VF1 - Case Study: Airplane Ticket Booking Management System - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://localhost:8080/vf1_application/

Latest Headlines misc s/w eng localhost


Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source Options

Inbox - Outlook We... My Homepage D4 Assignment - res... VF1 - Case Study Miscellaneous localhost:8080 / loc... http://ho...w0910.sql homepages - FireFTP

VF1 - Case Study: Airplane Ticket Booking Management System

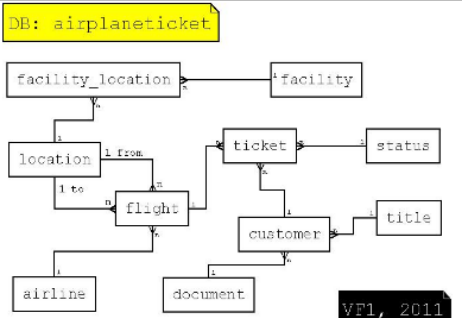
Main page [airline](#) [location](#) [flight](#) [ticket](#) [customer](#) [facility](#)

Welcome to the VF1 - Case Study: Airplane Ticket Booking Management System



From: FlightX © 2010

DB: airplaneticket



VF1, 2011

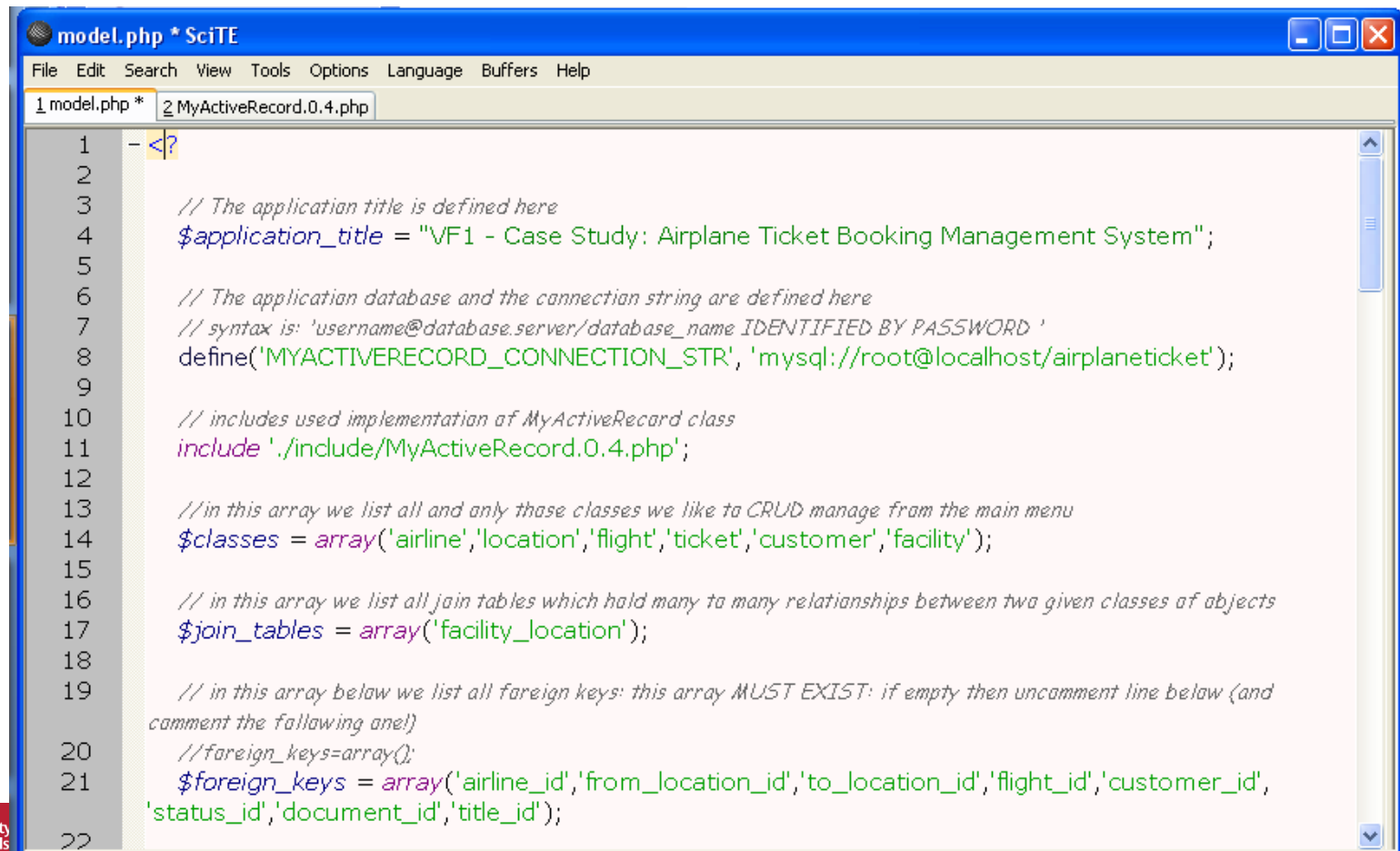
ERD developed using DIA Diagram Editor

This program has been developed by using VF1, possibly the simplest open source free PHP development framework in the world! :) You can redistribute VF1 and/or modify it under the terms of the [GNU General Public License](#) as published by the Free Software Foundation. VF1 is distributed in the hope that it will be useful, but comes WITHOUT ANY WARRANTY. See the GNU General Public License for more details.

Done

Open the model.php file with a code editor

You might want to open also the MyActiveRecord.0.4.php file for consulting purposes: be sure you are not editing it!



```
1  - <?
2
3  // The application title is defined here
4  $application_title = "VF1 - Case Study: Airplane Ticket Booking Management System";
5
6  // The application database and the connection string are defined here
7  // syntax is: 'username@database.server/database_name IDENTIFIED BY PASSWORD '
8  define('MYACTIVERECORD_CONNECTION_STR', 'mysql://root@localhost/airplaneticket');
9
10 // includes used implementation of MyActiveRecord class
11 include './include/MyActiveRecord.0.4.php';
12
13 // in this array we list all and only those classes we like to CRUD manage from the main menu
14 $classes = array('airline','location','flight','ticket','customer','facility');
15
16 // in this array we list all join tables which hold many to many relationships between two given classes of objects
17 $join_tables = array('facility_location');
18
19 // in this array below we list all foreign keys: this array MUST EXIST: if empty then uncomment line below (and
    comment the following one!)
20 //foreign_keys=array();
21 $foreign_keys = array('airline_id','from_location_id','to_location_id','flight_id','customer_id',
    'status_id','document_id','title_id');
22
```


The VF1 *model.php* file (1)

Main elements:

... *The application title is defined here*

```
$application_title = "VF1 - Case Study: Airplane Ticket  
Booking Management System";
```

... *The application database and the connection string are defined as follows:*

```
'username@database.server/database_name IDENTIFIED BY  
PASSWORD '
```

```
define('MYACTIVERECORD_CONNECTION_STR',  
'mysql://root@localhost/airplaneticket');
```

... *includes used implementation of MyActiveRecord class*

```
include './include/MyActiveRecord.0.4.php';
```

The VF1 *model.php* file (2)

... in this array we list all and only those classes we like to CRUD manage from the main menu

```
$classes =  
array('airline','location','flight','ticket','customer','facility');
```

... in this array we list all join tables which hold many to many relationships between two given classes of objects

```
$join_tables = array('facility_location');
```

... in this array below we list all foreign keys: this array MUST EXIST: if empty then uncomment line below (and comment the following one!)

```
//foreign_keys=array();
```

```
$foreign_keys =  
array('airline_id','from_location_id','to_location_id','flight_i  
d','customer_id','status_id','document_id','title_id');
```

The VF1 *model.php* file (3)

... relationships between entities/classes are named below: if no name has been given to a certain relationship, the bare foreign key would be displayed

function

name_child_relationship(\$class_name,\$foreign_key)

{

*if (\$class_name == 'customer' && \$foreign_key ==
'title_id') {*

return " title ";

}

*... else if (\$class_name == 'flight' && \$foreign_key ==
'from_location_id') {*

return " from ";

}

The VF1 *model.php* file (4)

// classes are defined below as extensions of MyActiveRecord class

```
class airline extends MyActiveRecord
{
}
```

When using MyActiverecord, an example of OO code would be as follows:

```
$airline = MyActiveRecord::FindBySql('airline', 'SELECT * FROM airline
WHERE id > 1 ORDER BY name');
```

```
From: class MyActiveRecord {
function FindBySql( $strClass, $strSQL, $strIndexBy='id' ) {} }
```

Returns an array of objects of class strClass mapped from SQL query strSQL

```
$airline = MyActiveRecord::FindBySql('airline', 'SELECT *
FROM airline ORDER BY name');
foreach ( $airline as $this_company ) print $this_company->first_name;
```

The VF1 *index.php* file

```
<? include "model/model.php"; ?>
```

```
<html>
```

```
<link rel="shortcut icon" href="include/images/vf1_icon_big.ico" />
```

```
<link id="style_link" rel=StyleSheet href="include/vf1-1024x768.css"  
      type="text/css" media=all />
```

```
<link id="style_link" rel=StyleSheet href="include/calendar.css" type="text/css"  
      media=all />
```

```
<script src="include/vf1.js"></script>
```

```
<script type="text/javascript" src="include/calendar.js"></script>
```

```
<title><? echo $application_title; ?></title>
```

```
<? ...
```

```
    include "controller/controller.php";
```

```
?>
```

The VF1 *controller.php* file

Provides the main menu options (“Cars”, “Drivers”, “Drivings”...)

```
$here = isset($_REQUEST['here']) ? $_REQUEST['here'] : "";
```

Assess status related to the CRUD operations for each “\$here”
 (“update”, “confirm_update”, etc...)

```
$mode = isset($_REQUEST['mode']) ? $_REQUEST['mode'] : "";
```

Hence creating context-sensitive menu (and submenus)

- determines main menu

```
include "../view/menu.php";
```

- determines sub_menu

```
include "../view/sub_menu.php";
```

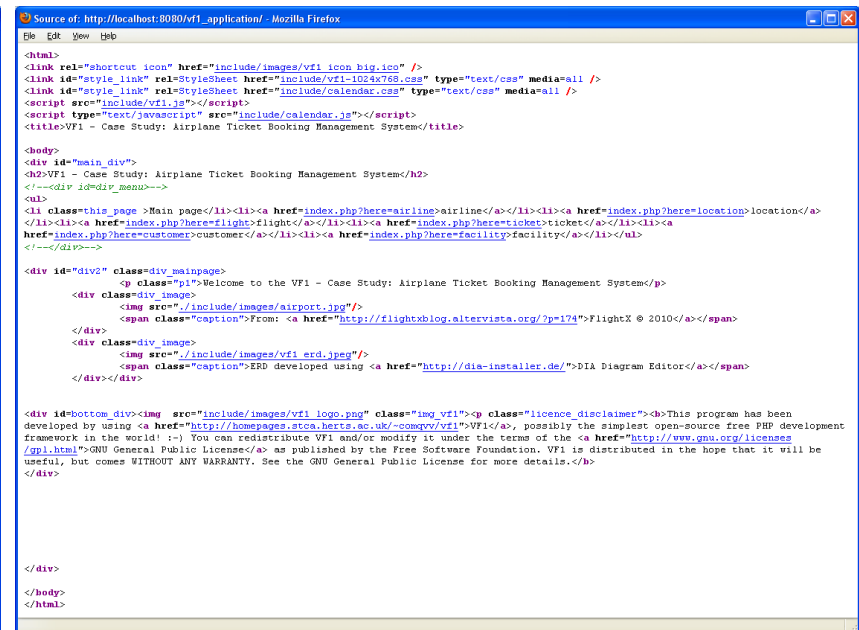
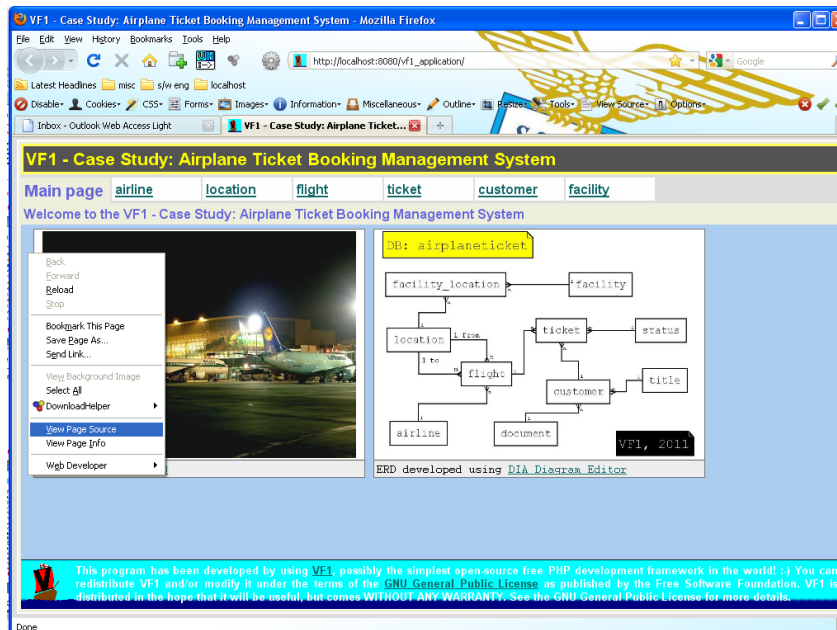
determines content in the main page depending on \$here and \$mode

```
include "../view/main_page.php";
```

The VF1 *index.php* HTML STRUCTURE (1)

It is possible to retrieve and analyse the HTML structure of the index.php web-page by:

- Viewing the page source (right button, choose option)
- Identifying all relevant HTML <div> elements



The VF1 *index.php* HTML STRUCTURE (2)

Understanding the HTML structure is due for several views of VF1-based applications, namely:

- the “Main page” (see previous slide, Figures 1 and 2)
- “General Listing” (see below, Figure 3)
- “CRUD-manage” (i.e., those views including search, create or update forms: see below, Figure 4)

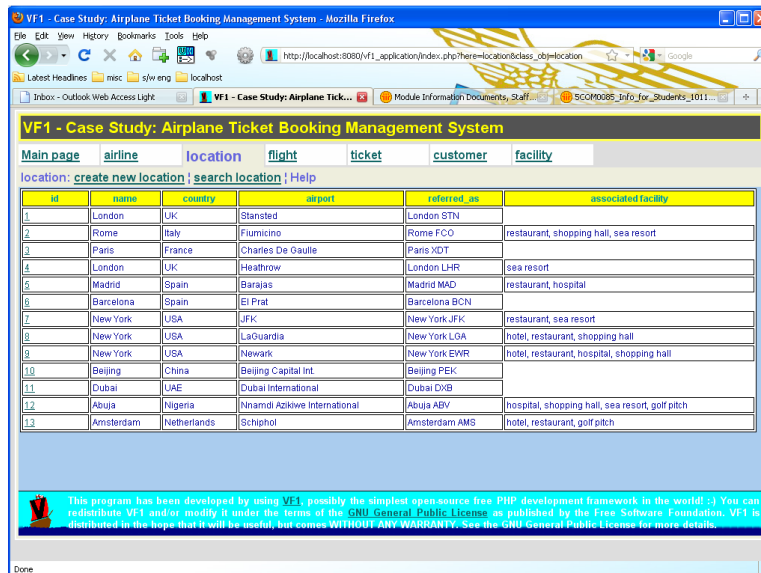


Fig. 3

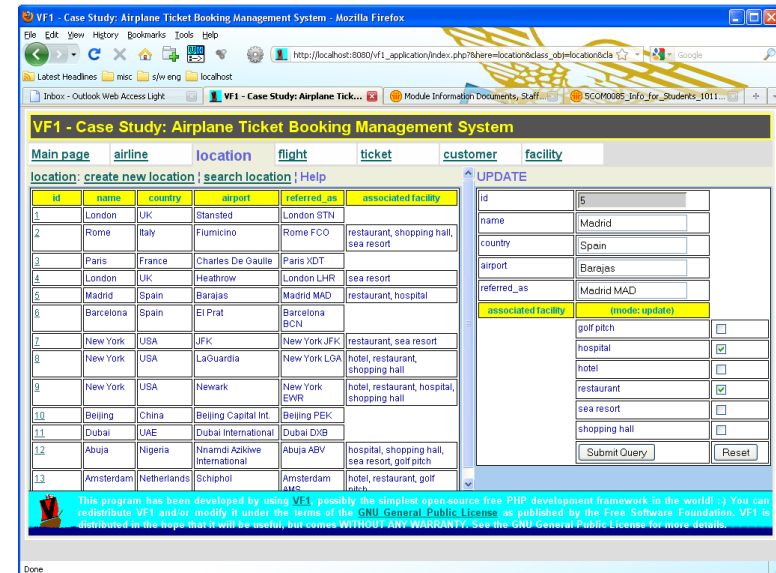


Fig. 4

The VF1 *index.php* HTML STRUCTURE (3)

Let's consider the CRUD-manage view (see previous slide, Figure 4). Its HTML source code is retrieved by right-clicking on the web-page (see Figure 5).

Save it on an editable file.
Once saved...



```
<html>
<link rel="shortcut icon" href="include/images/vf1 icon big.ico" />
<link id="style_link" rel="StyleSheet" href="include/vf1-1024x768.css" type="text/css" media=all />
<link id="style_link" rel="StyleSheet" href="include/calendar.css" type="text/css" media=all />
<script src="include/vf1.js"></script>
<script type="text/javascript" src="include/calendar.js"></script>
<title>VF1 - Case Study: Airplane Ticket Booking Management System</title>
|
<body>
<div id="main_div">
<h2>VF1 - Case Study: Airplane Ticket Booking Management System</h2>
<!--<div id=div_menu-->
<ul>
<li><a href=index.php>Main page</a></li><li><a href=index.php?here=airline>airline</a></li><li class=this_page>location</li><li><a href=index.php?here=flight>flight</a></li><li><a href=index.php?here=ticket>ticket</a></li><li><a href=index.php?here=customer>customer</a></li><li><a href=index.php?here=facility>facility</a></li></ul>
<!--</div-->

<div id="div2" class=div_mainpage>

<p class='p1'><a href='index.php?here=location&class_obj=location'>location</a>: <a href='index.php?here=location&mode=create&class_obj=location'>create new location</a> | <a href='index.php?here=location&mode=search&class_obj=location'>search location</a> | Help</p>

<table class=table1>
<tr>
<th><td>name</td><td>country</td><td>airport</td><td>referred_as</td><td>associated facility</td><tr><td><a href=javascript:update_obj('index.php','location',1)>1</a><td>London</td><td>UK</td><td>Stansted</td><td>London STN</td><td><a href=javascript:update_obj('index.php','location',2)>2</a><td>Rome</td><td>Italy</td><td>Fiumicino</td><td>Rome FCO</td><td>restaurant, shopping hall, sea resort</td><td><a href=javascript:update_obj('index.php','location',3)>3</a><td>Paris</td><td>France</td><td>Charles De Gaulle</td><td>Paris XDT</td><td><a href=javascript:update_obj('index.php','location',4)>4</a><td>London</td><td>UK</td><td>Heathrow</td><td>London LHR</td><td>sea resort</td><td><a href=javascript:update_obj('index.php','location',5)>5</a><td>Madrid</td><td>Spain</td><td>Barajas</td><td>Madrid MAD</td><td>restaurant, hospital</td><td><a href=javascript:update_obj('index.php','location',6)>6</a><td>Barcelona</td><td>Spain</td><td>El Prat</td><td>Barcelona BCN</td><td><a href=javascript:update_obj('index.php','location',7)>7</a><td>New York</td><td>USA</td><td>JFK</td><td>New York JFK</td><td>restaurant, sea resort</td><td><a href=javascript:update_obj('index.php','location',8)>8</a><td>New York</td><td>USA</td><td>LaGuardia</td><td>New York LGA</td><td>hotel, restaurant, shopping hall</td><td><a href=javascript:update_obj('index.php','location',9)>9</a><td>New York</td><td>USA</td><td>Newark</td><td>New York EUR</td><td>hotel, restaurant, hospital, shopping hall</td><td><a href=javascript:update_obj('index.php','location',10)>10</a><td>Beijing</td><td>China</td><td>Beijing Capital Int</td><td>Beijing PKX</td><td><a href=javascript:update_obj('index.php','location',11)>11</a><td>Dubai</td><td>UAE</td><td>Dubai International</td><td>Dubai DXB</td><td><a href=javascript:update_obj('index.php','location',12)>12</a><td>Abuja</td><td>Nigeria</td><td>Nnamdi Azikiwe International</td><td>Abuja ABV</td><td>hospital, shopping hall, sea resort, golf pitch</td><td><a href=javascript:update_obj('index.php','location',13)>13</a><td>Amsterdam</td><td>Netherlands</td><td>Schiphol</td><td>Amsterdam AMS</td><td>hotel, restaurant, golf pitch </td></tr>
</div>

<div id="div_right"><script>document.getElementById('div2').style.width = '60%';</script><p class=p1>UPDATE</p><table class=table1><form id=update_location' action=index.php?here=location&mode=confirm_update&class_obj=location&class_obj_id=5 method=post><tr><td>id</td><input type=text name=input_id' value=5' readonly=true><tr><td>name</td><input type=text name=input_name' value=Madrid><tr><td>country</td><input type=text name=input_country' value=Spain><tr><td>airport</td><input type=text name=input_airport' value=Barajas><tr><td>referred_as</td><input type=text name=input_referred_as' value=Madrid MAD><tr><td>associated facility (mode: update)</td><td><td>golf pitch </td><input type=checkbox id=jt_input_facility_0' name=jt_input_facility_0' value=6><tr><td><td>hospital </td><input type=checkbox id=jt_input_facility_1' name=jt_input_facility_1' value=3> checked=true <tr><td><td>hotel </td><input type=checkbox id=jt_input_facility_2' name=jt_input_facility_2' value=1><tr><td><td>restaurant </td><input type=checkbox id=jt_input_facility_3' name=jt_input_facility_3' value=2> checked=true <tr><td><td>sea resort </td><input type=checkbox id=jt_input_facility_4' name=jt_input_facility_4' value=5><tr><td><td>shopping hall </td><input type=checkbox id=jt_input_facility_5' name=jt_input_facility_5' value=4><tr><td><td><input type=hidden name=jt_name' value=facility_location><input type=hidden name=jt_class' value=facility><tr><td><td><input type=submit><td><input type=reset></table></form></div>

Line 8, Col 1
```

Fig. 5

The VF1 *index.php* HTML STRUCTURE (4)

```
<html>
...
<body>
<div id="main_div">
  <h2>VF1 - Case Study: Airplane Ticket Booking Management System</h2>
  <!-- Main menu --><ul><li><a href=index.php>Main page</a></li><li><a
href=index.php?here=airline>airline</a></li>...</ul> <!-- end Main menu -->

  <!-- div containing whatever is below the main menu on the left side --><div id="div2" class=div_mainpage>
    <!-- CRUD submenu --> <p class='p1'><a href='index.php?here=location&class_obj=location'>location</a>:
    <a href='index.php?here=location&mode=create&class_obj=location'>create new location</a> | ... | Help</p>
    <!-- end CRUD submenu -->
    <!-- table containing listing of records --><table class=table1><tr><th>id<th>name<th>country<th>airport
    <th>referred_as<th>associated facility<tr><td><a href=javascript:update_obj('index.php','location',1);>1
    </a> <td>London<td>UK<td>Stansted<td>London STN<td>...</table>
    </div> <!--end div containing whatever is below the main menu on the left side -->

  <!-- div containing whatever is below the main menu on the right side --><div id='div_right'>
    <script>document.getElementById('div2').style.width='60%';</script><p class=p1>UPDATE</p>
    <table class=table1><form id='update_location'
    action=index.php?here=location&mode=confirm_update&class_obj=location&class_obj_id=5
    method=post><tr><td>id<td><input type=text name='input_id' value='5'
    readonly=true>...</table></form>
    </div> <!-- end div containing whatever is below the main menu on the right side -->

  <!-- div containing disclaimer in the bottom side --><div id=bottom_div>...</div> <!-- end div disclaimer -->

</div> <!-- end div id="main_div"-->
</body>
</html>
```

Fig. 6

The VF1 *index.php* HTML STRUCTURE (5)

VF1 - Case Study: Airplane Ticket Booking Management System - Mozilla Firefox

location: [create new location](#) | [search location](#) | [Help](#)

ID	City	Country	Airport	Facilities
1	London	UK	Heathrow	London LHR
2	Madrid	Spain	Barajas	Madrid MAD
3	Paris	France	Charles de Gaulle	Paris CDG
4	Barcelona	Spain	El Prat	Barcelona BCN
5	New York	USA	JFK	New York JFK
6	New York	USA	LGA	New York LGA
7	New York	USA	Newark	New York EWR
8	Beijing	China	Beijing Capital Int'l	Beijing PEK
9	Dubai	UAE	Dubai International	Dubai DXB
10	Abuja	Nigeria	Nnamdi Azikiwe International	Abuja ABV
11	Amsterdam	Netherlands	Schiphol	Amsterdam AMS

UPDATE

associated facility (mode: update)

golf pitch ☐

hospital ☒

hotel ☐

restaurant ☒

sea resort ☐

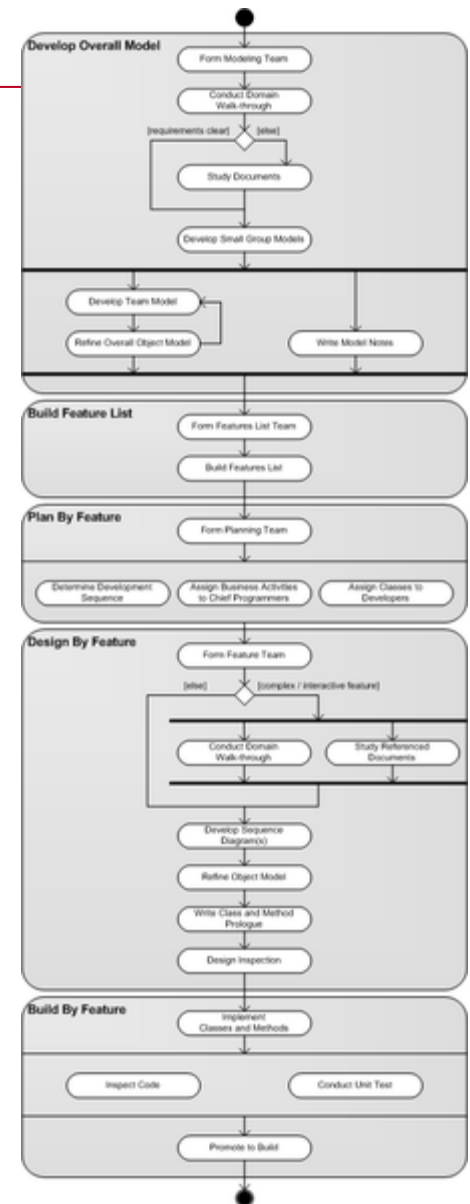
shopping hall ☐

This program has been developed by using VF1, possibly the simplest and most powerful web development tool in the world. You can redistribute VF1 and/or modify it under the terms of the GNU General Public License, which is distributed in the hope that it will be useful, but comes WITHOUT ANY WARRANTY.

Fig. 7

In class code walkthrough...

- We'll get through the code *live* and see how it is possible to develop a simple and yet usable application
- We'll organise the user interface space by means of the *\$here* variable (a sort of “Feature Driven Development”)
- (from wikipedia) “FDD is an iterative and incremental software development process. It is one of a number of Agile methods for developing software and forms part of the Agile Alliance. FDD blends a number of industry-recognized best practices into a cohesive whole. These practices are all driven from a client-valued functionality (feature) perspective. Its main purpose is to deliver tangible, working software repeatedly in a timely manner.”



FOCUS on vf1.js

It includes javascript functions, each of which calls HTML document objects.

```
function update_obj(file_name,class_name,obj_id)
{
    /* alert("CIAO"); */
    document.location =
    file_name+"?"+"&here="+class_name+"&class_obj="+class
    _name+"&class_obj_id="+obj_id+"&mode=update";
}

function change_obj(obj_name)
{
    /* alert(obj_name); */
    var selobj =
    document.getElementById('select_'+obj_name);
    document.getElementById('input_'+obj_name).value =
    selobj.options[selobj.selectedIndex].value;
}
```

From VF1 to SAMS-P (1)

Have a walkthrough along VF1

- Focus on the suggested HCI layout, but you are reminded to change it: main actions on understanding the existing .css file and create a completely different one for your SAMS-P application
- Focus on the structure of the .php files: main actions are to understand and possibly reproduce and customise them within your SAMS-P application
- Focus on the used functions: main actions on understanding, using and customising them within your SAMS-P .js file

From VF1 to SAMS-P (2)

General actions:

- Understand how `MyActiveRecord.0.4.php` has been used in VF1
- Consider how the data model for your SAMS-P DB has been designed and manage the classes on your php application file accordingly
 - Take into account VF1 constraints on how to design and name tables within your VF1 application
 - Instructions on how to import it are available on this and next week's tutorials (PHPMYADMIN & CLASSMAKER2)

From VF1 to SAMS-P (3)

Further general actions:

- Create a folder hosting your “SAMS-P” application within your MoWeS “www” folder, e.g.
`U:/wos_csitde/www/D4-CW/`
- All files for your D4 application should be stored into the above mentioned “D4-CW” folder
- Keep record of the development process, and manage it in a professional way (in your demo/presentation, your team will be assessed against this as well)

MAIN RESOURCES

- D4 Assignment - resources and help page
<http://homepages.feis.herts.ac.uk/~comqv/csitde/d4.html>
- Vf1: <http://homepages.feis.herts.ac.uk/~comqv/vf1/>
- Jake Grimley, MyActiveRecord.0.4.php
<http://freshmeat.net/projects/myactiverecord/>
- Wikipedia (06 Dec 2009):
http://en.wikipedia.org/wiki/Feature_Driven_Development
- All other lectures AND tutorials!
- Class discussions and news on StudyNet.