**Unité d'Enseignement en Informatique**
**Année 2013-2014**

**Master M1 EFREI – ASI – BI**
**Devoir Écrit de ERP – First Session April 9, 2013**
(2h00 duration, no document allowed)

***Rule:*** *All answers must be written in English.*

**Exercise 1: (4 points)**

1) What do the letters ERP stand for?
2) In a company, what is the role of ERP software?
3) What do the letters EAI stand for?
4) What are the differences of architecture between the ERP and EAI approaches?
5) Give 2 advantages of ERP compared to EAI.
6) Give 1 advantage of EAI compared to ERP.

**Exercise 2: (4 points)**

Three software solutions for your company are presented as ERP by their vendors. An analysis of these three solutions, showing which applications are integrated (I) to the central core of the ERP and which are installable modules (M), is presented in the following table:

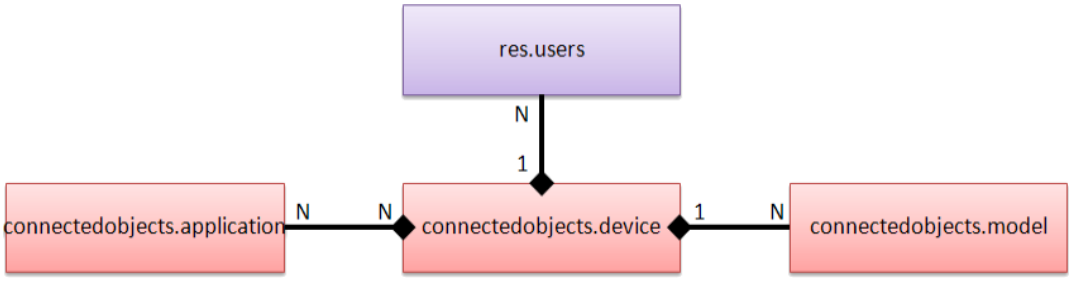
| Application | ERP1 | ERP2 | ERP3 |
|---|---|---|---|
| Sales | I | M | I |
| Purchasing | | M | I |
| Warehouse Management | I | M | I |
| CRM | | M | I |
| MRP | I | M | I |
| Accounting | | M | M |
| Human Resources | | M | M |
| CMMS | I | M | |

1) In what big kind of ERP category would you classify each of these ERP?

2) During a meeting of the IT department of your company, the question is about the installation of each of these ERP. You are asked to give your opinion. Could you give some advantages and disadvantages for each of these three solutions?

**Exercise 3: (12 points)**

Numerous connected objects have been spread inside a company. In order to help managing this large stock of devices, we want to manage the assigning of connected objects, and the applications which are installed, by developing a new OpenERP module. The data model of this new module is detailed below. A device is owned by a user, and a user can own several devices. Different models of devices exist. Each device belongs to a model, and several devices can be of the same model. Several applications can be installed on a device, and the same application can be installed on several devices.



The writing of source code of `connectedobject.model` has already begun. See the three files `connectedobject.py`, `connectedobject_data.xml` and `connectedobject_view.xml` below. In order to continue this development, you have to answer the following questions.

1) Create three new filters for `connectedobjects.model` in order to filter models with:

a) WiFi and 3G
b) WiFi only (neither 3G nor 4G)
c) 3G or 4G

2) Define the business object `connectedobjects.application` with 2 columns defining the name and the version of the software. Then, write the corresponding `tree` and `form` views.

3) Define the business object `connectedobjects.device` with 4 columns, defining particularly the three relations with the other business objects.

4) In `connectedobjects.model`, is it a good solution to have several booleans to describe the different possible connection types? Why? Would you change that for another solution? If so, then describe how?

5) What do webservices allow to perform? For OpenERP, cite 5 functions of the O.R.M. which are accessible through webservices. What protocol is used by OpenERP for its webservices? What is created by the script called `create.php` below?

6) Give the modifications of code to perform on the script `create.php` in order to create a new record of `connectedobjects.application` for a version of the navigator Safari.

**Documentation:**

a) Example of evaluation of a boolean value in a filter: `[('active','=',True)]`

b) The domain of a filter can also evaluate several conditions using Boolean operations with prefix notation (also known as Polish notation). For example, to filter whether the field `country_code` is the one of Belgium OR France would be done like this:

```
['|',('country_code','=','be'),('country_ code','=','fr')]
```

connectedobjects.py

```
from openerp.osv import osv
from openerp.osv import fields
from openerp.tools.translate import _
import time

listOS = [('win', 'Windows'), ('and', 'Android'), ('ios', 'iOS'), \
          ('blb', 'BlackBerry'), ('sym', ' Symbian,')]

class connectedobjects_model(osv.osv):
    """ The model of a connected objects """
    _name = "connectedobjects.model"
    _description = "The model of a connected objects"
    _columns = {
        'name': fields.char('Model of connected objects', size=64, required=True),
        'operating_system': fields.selection(listOS, 'Operating system', required=True),
        'bluetooth': fields.boolean('Bluetooth connection', required=True),
        'wifi': fields.boolean('WiFi connection', required=True),
        '3G': fields.boolean('3G connection', required=True),
        '4G': fields.boolean('4G connection', required=True),
    }
    _sql_constraints = [
        ('name', 'unique(name)', 'The name of a model must be unique')
    ]
    _order = 'name asc'

```

connectedobjects_data.xml

```xml
<?xml version="1.0"?>
<openerp>
    <data>
        <record model="connectedobjects.model" id="connectedobjects_model_GTN8000">
            <field name="name">Galaxy Note 10.1 (GT-N8000)</field>
            <field name="operating_system">and</field>
            <field name="bluetooth" eval="True"/>
            <field name="wifi" eval="True"/>
            <field name="3G" eval="False"/>
            <field name="4G" eval="False"/>
        </record>
        <record model="connectedobjects.model" id="connectedobjects_model_GTN8010">
            <field name="name">Galaxy Note 10.1 (GT-N8010)</field>
            <field name="operating_system">and</field>
            <field name="bluetooth" eval="True"/>
            <field name="wifi" eval="True"/>
            <field name="3G" eval="True"/>
            <field name="4G" eval="False"/>
        </record>
        <record id="base.user_demo" model="res.users">
            <field name="groups_id" eval="[(4,ref('base.group_user'))]"/>
        </record>
    </data>
</openerp>
```

connectedobjects_view.xml

```xml
<?xml version="1.0"?>
<openerp>
  <data>
    <!-- ConnectedObjects Model: Form View -->
    <record model="ir.ui.view" id="view_connectedobjects_model_form">
      <field name="name">connectedobjects.model.form</field>
      <field name="model">connectedobjects.model</field>
      <field name="arch" type="xml">
       <form string="Models of devices" version="7.0">
         <label for="name"/><field name="name"/>
         <label for="operating_system"/><field name="operating_system"/>
          <label for="bluetooth"/><field name="bluetooth"/>
          <label for="wifi"/><field name="wifi"/>
          <label for="3G"/><field name="3G"/>
          <label for="4G"/><field name="4G"/>
       </form>
      </field>
    </record>

    <!-- ConnectedObjects Model: Tree View -->
    <record model="ir.ui.view" id="view_connectedobjects_model_tree">
      <field name="name">connectedobjects.model.tree</field>
      <field name="model">connectedobjects.model</field>
      <field name="field_parent"></field>
      <field name="arch" type="xml">
       <tree string="Models of devices">
         <field name="name"/>
         <field name="operating_system"/>
          <field name="bluetooth"/>
          <field name="wifi"/>
          <field name="3G"/>
          <field name="4G"/>
       </tree>
      </field>
    </record>

    <!-- ConnectedObjects Model: Search Idea  -->
    <record model="ir.ui.view" id="view_connectedobjects_model_search">
      <field name="name">connectedobjects.model.search</field>
      <field name="model">connectedobjects.model</field>
      <field name="arch" type="xml">
        <search string="Models of devices">
          <field name="name" string="Models"/>
          <filter string="Windows OS" domain="[('operating_system','=', 'win')]" help="With
Windows OS"/>
          <filter string="Not Windows OS" domain="[('operating_system','!=', 'win')]"
help="Without Windows OS"/>
          <group expand="0" string="Group By...">
            <filter string="Operating System" help="Operating System"
context="{'group_by':'operating_system'}"/>
          </group>
```

```
49          </search>
50        </field>
51      </record>
52
53      <!-- ConnectedObjects Model: Action -->
54      <record model="ir.actions.act_window" id="action_connectedobjects_model">
55        <field name="name">Models</field>
56        <field name="res_model">connectedobjects.model</field>
57        <field name="view_type">form</field>
58        <field name="view_mode">tree,form</field>
59        <field name="search_view_id" ref="view_connectedobjects_model_search"/>
60      </record>
61
62      <!-- Top menu item -->
63      <menuitem name="Connected Objects" id="base.menu_connectedobjects_root" sequence="120"
    groups="base.group_user"/>
64
65      <!-- Menus sections -->
66      <menuitem name="Configuration" id="menu_connectedobjects_configuration"
    parent="base.menu_connectedobjects_root" sequence="2"/>
67
68      <!-- Menus items -->
69      <menuitem name="Models" id="menu_model" parent="menu_connectedobjects_configuration"
    action="action_connectedobjects_model" sequence="1"/>
70
71    </data>
72  </openerp>
```

create.php

```
 1  <?php
 2  require_once('XML/RPC.php'); // Include PEAR library for XML-RPC
 3  require_once('login.inc.php'); // Include login function for connexion
 4  require_once('vars.inc.php'); // Define connexion variables
 5
 6  $uid = login ($HOST, $PORT, $DB, $USER, $PASS, $DEBUG) ;
 7
 8  $client = new XML_RPC_Client('/xmlrpc/object', "http://$HOST:$PORT");
 9  $client->setDebug($DEBUG);
10
11  $structVal = array( // Values to give to the fields of the new record
12    'name' => new XML_RPC_Value('iPad Air 4G', 'string'),
13    'operating_system' => new XML_RPC_Value('ios', 'string'),
14    'bluetooth' => new XML_RPC_Value(true, 'boolean'),
15    'wifi' => new XML_RPC_Value(true, 'boolean'),
16    '3G' => new XML_RPC_Value(false, 'boolean'),
17    '4G' => new XML_RPC_Value(true, 'boolean'),
18  );
19
20  $msg = new XML_RPC_Message('execute');
21  $msg->addParam(new XML_RPC_Value($DB, 'string'));
22  $msg->addParam(new XML_RPC_Value($uid, 'int'));
23  $msg->addParam(new XML_RPC_Value($PASS, 'string'));
24  $msg->addParam(new XML_RPC_Value('connectedobjects.model', 'string')); // Name of the relation
25  $msg->addParam(new XML_RPC_Value('create', 'string'));        // The ORM method
26  $msg->addParam(new XML_RPC_Value($structVal, 'struct'));      // Values of the new record
27
28  $resp = $client->send($msg);
29
30  if (!$resp)
31    die('Communication error: ' . $client->errstr);
32  else if ($resp->faultCode()) { // FaultCode seams to be never set by OpenERP 7.0
33    echo 'Fault Code: '.$resp->faultCode()."\r\n";
34    echo 'Fault Reason: '.$resp->faultString()."\r\n";
35    exit(1);
36  }
37
38  $val = $resp->value() ;
39
40  if ($val->kindOf() == 'struct') {
41    $struct = XML_RPC_decode($val) ;
42    if (isset($struct['faultCode'])) // Trick to catch fault code
43      echo '<p>FAILURE RESPONSE: '.$struct['faultCode']."</p>\r\n" ;
44  }
45  else { // kindOf == 'scalar'
46    // Show id of the new record
47    $id = $val->scalarval() ;
48    echo "<p>New element created (id:$id)</p>\r\n";
49  }
50  ?>
```