Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Engineering Web Applications

Date: Thursday 21st January 2010
Time: 14.00 – 16.00

Please answer Question ONE from Section A and any TWO Questions from the THREE questions in Section B

The use of electronic calculators is NOT permitted
1. **Compulsory**

Answer each of the following questions concisely. **Each carries two marks.**

a) List the five main stages of Java development over the last decade particularly relevant to Web applications.

b) What are the three main stages in which a servlet handles a HTTP request and what distinguishes them fundamentally? How should these stages be reflected in the design of a Web application program text?

c) What is the role of the *init* function? How and when is it called exactly? What is the role of the *destroy* function? How and when is it called exactly?

d) Give at least two differences between the HTTP commands GET and POST. What are the functions called by the server in response to these commands respectively? What are the two parameters of these functions and their respective roles?

e) Describe the main features of a consistent HTML page, covering specifically:

i) The notion of an element and how elements textually relate to one another

ii) The notion and form of attributes

f) Define the notion of a *relative* URL. Explain how a relative URL is used by a Web browser and how this feature affects servlet programming.

g) Explain the problem of session tracking and indicate which feature of HTTP causes this problem. Describe the *principle common to the various methods of session tracking.* (Do not describe any specific method.)

h) What is an assertion and how should assertions be used in program development? What is a program invariant? Illustrate your answer by a brief example.

i) What are preconditions? How should they be derived from program invariants? Illustrate your answer by a brief example.

j) What is the *functionality condition* of the various Dictionary applications, and how is it expressed in the Java implementation involving `int n; String word[]; String definition;`?
Section B

Answer any TWO questions from this Section

2. a) Describe the design principles underpinning the variant Java implementation Dictionary1 of the Evolving Dictionary Model (EDM), as a general framework for Web applications. In particular:

i)     Describe the purposes of the four main categories A, B, C and D of classes involved.

ii)    Describe how the Web pages of the application are related to category B1 classes. Recall that each page indexed \( p \) has two associated classes. Name these two classes and explain their respective roles.

iii)   Explain how this design supports the scalability of the application and any other important criterion of good design for Web applications. (6 marks)

b)     Illustrate your answer to part (a) by explaining the role of the following (simplified) program components. In particular explain the difference between a command function and the corresponding basic Dictionary operation.

    // From class A0Dictionary1:
    C0Dictionary1 D = new C0Dictionary1(Max);
    // ...
    public void doGet(HttpServletRequest req, HttpServletResponse res)
    throws ServletException, IOException
    {
        String Page;                           // Page number
        Page = req.getParameter("Page");
        if (Page.equals("0"))
            B1Page0Proc.ProcReq(req, res);
        else if (Page.equals("1"))
            B1Page1Proc.ProcReq(req, res, D);
    }  // ...

    // From class B1Page1Proc:
    public static void ProcReq(HttpServletRequest req, HttpServletResponse res, C0Dictionary1 D)
    {
        String Com;                      // Command http parameter // . . .
        Com = req.getParameter("Com");
        if (Com.equals("Emp"))
            ComEmp(D, Com, req, res);
        else if (Com.equals("Ins"))
            ComIns(D, Com, req, res);
        // . . .
    }

(Question 2 continues on the following page)
(Question 2 continues from the previous page)

// From class C0Dictionary1:
public void Ins(String w, String d)
// Insert (w, d) into D
// Initial conditions
//   w is a non-empty string
//   d is a non-empty string
//   for all i with 0 <= i < n (!word[i].equals(w))

 c) Complete the following implementation of the function ComIns() consistently
   with part (b), using the function calls D.Ins(...), D.IsEntry(), Errs.IsMb() and
   Errs.NumErrs() etc. (Do not implement the basic function Ins().)

//...
Errs = new B0ErrorSet(B0ErrorSet.InsSet);
if (w == null || w.equals(""))
  Errs.Ins(B0ErrorSet.Ins_EmptyWord);

 d) i) Describe the paradigm Model View Controller (MVC). Briefly show how
 this model relates to the three-stage design principle derived from the
 HTTP request-response cycle.

 ii) Show that the generic Web application design illustrated by Dictionary1
 may be described as a simple realisation of MVC. In particular clearly
 relate the relevant aspects of Dictionary1 to those of MVC.
3. a) The text of a JSF page, index.jsp, is given below. Describe the effects of every aspect of each of the numbered lines in detail.

```html
<html>
  <%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>
  <%@ taglib uri="http://java.sun.com/jsf/html" prefix="h" %>
  <f:view>
    <head>
      <title>Simple JSF Application 00</title>
    </head>
    <body>
      <h:form>
        <h3>Welcome to your zeroth JavaServer Faces (JSF) application</h3>
        <h2>DING-DONG</h2>
        <table>
          <tr>
            <td>The bell did: </td>
            <td>
              <h:outputText value="#{bell.oldSound}"/>
            </td>
          </tr>
          <tr>
            <td>What will it do next? </td>
            <td>
              <h:inputText value="#{bell.sound}"/>
            </td>
          </tr>
        </table>
        <h:commandButton value="Proceed" action="anotherSound"/>
      </h:form>
    </body>
  </f:view>
</html>
```

(8 marks)

b) Explain the notions of a JSF component and the component tree associated with a JSF page. (6 marks)

c) Describe the JSF life cycle by a diagram and briefly explain each step with reference to your answer to (b). (6 marks)
4. a) Explain the notions of a *Java bean* and a *bean property*; the role of beans in developing a JSF application; and the basic rules which a bean must satisfy.  
(6 marks)

b) i) Explain the notion of a *configuration file* in a Web application, and the role of the file faces-config.xml in particular.

ii) There follows part of the text of faces-config.xml associated with the page of question 3(a). Explain each of its *elements* in detail.

```xml
  <navigation-rule>
    <from-view-id>/index.jsp</from-view-id>
    <navigation-case>
      <from-outcome>anotherSound</from-outcome>
      <to-view-id>/newsound.jsp</to-view-id>
    </navigation-case>
  </navigation-rule>
  <managed-bean>
    <managed-bean-name>bell</managed-bean-name>
    <managed-bean-class>com.corejsf.BellBean</managed-bean-class>
    <managed-bean-scope>session</managed-bean-scope>
  </managed-bean>
</faces-config>
```
(6 marks)

(Question 4 continues on the following page)
c) The following text consists of two classes from the QuizV1 Web application, simplified and jumbled up. Note that multiple occurrences of a line are only given once. Reconstruct these two classes.

```java
public class Problem {
    public class Problems {
        for (D.InitScan(), i = 0;
             return prblSet;
    } // class Problem
} // class Problems
D.GetNext(), i++) {
    private String question;
    int i = 0;
    public Problems() {D = new D0DictionaryDB(4, errs);}
    this.question = question;
    public Problem[] prblSet() {
        public Problem(String question) {
            private D0DictionaryDB D = null;
            Problem[] prblSet = new Problem[D.Num()];
            public String getQuestion() { return question; }
            private B0ErrorSet errs = null;
            prblSet[i] = new Problem(D.CurWord());}
    } // public Problem[] prblSet()
    D.NextEntry();
}
```

(8 marks)