NANYANG TECHNOLOGICAL UNIVERSITY

SEMESTER 1 EXAMINATION 2019-2020

CI6206 – INTERNET PROGRAMMING

November/December 2019 Time Allowed: **3 hours**

INSTRUCTIONS

1. This paper contains **FOUR (4)** questions and comprises **TEN (10)** pages, and Appendices of **FIVE (5)** pages, comprising a total of **FIFTEEN (15)** pages.
2. Part One: Answer the **ONE (1) COMPULSORY** question. (40 marks)
3. Part Two: Answer **TWO (2)** of the **THREE (3)** questions. (60 marks)
4. This is a **CLOSED-BOOK** examination.

PART ONE: Answer the ONE (1) COMPULSORY question.

1. **Durian888** is a popular local durian stall that mainly sells a wide variety durians imported from Malaysia. Over the years, they have gradually gained the trust of many locals especially for their quality and price of durians. Recently, there has been a drop in revenue due to pressure from competitors, increasing high cost of rental and cost of labors. Durian888 wishes to embrace online as an extension of their physical store with an aim to increase sales and helps provide more visibility to potential customers. With an online presence, customers will be able to purchase durians online and get them delivered to their homes with just a few clicks over the computers or mobile devices.

   Durian888 engages your service to develop an 'Open source' Java-based web application that allows customers to purchase, perform payment, manage orders and request for delivery at their own convenience. The project team conducted an interview session with the operation team and managed to gather the following user requirements, and intended features of the system.

Note: Question No. 1 continues on Page 2

The following web site (Durian888.com) features are provided for the customers

- Registration page. Customer must first register an account in the website. The following are information that users must enter.

  - 'Email Address' as user id
  - 'Password' and 'Confirm Password'
  - Mobile number
  - Gender
  - Delivery Address
  - Agree to BOTH the following terms
    - "I confirm that I am above 18 years old"
    - "I agree to the Privacy Policy on the information I provide."
  - Verify valid mobile number by authenticating a 6-digits pin using OTP.
  - All fields are compulsory.

- Browse product catalog. This feature allows registered customers to view a catalog with a list of 4 types durians.

  - Browse by types of durians
    - Black Thorn
    - Mao Shan Wang
    - Green Bamboo
    - D24
  - Information on each type of durian includes
    - Price per kg (In SGD)
    - A photo of the durian. Clicking on the photo opens a 'view product detail' as a pop-up page.
    - 'Add to cart' button
    - Durian can be 'On Sale' (show original and discounted price)

- View product details. This feature allows registered customers to view a more detailed info of a particular product.

  - Display multiple photos (1 large and 4 small)
  - Detailed description of the product
  - 'Add to cart' button
  - Able to adjust quantity of product before adding to cart.
  - Durian that is 'On Sale' to show original and discounted price.

Note: Question No. 1 continues on Page 3

- Order delivery page. This feature allows registered customers to select their preferred delivery schedule.

  - Delivery must be done within a week. Customer may choose either Mon, Tues, Wed, Thu, Fri, Sat or Sun.
  - Select a time slot either 0900-1200 or 1400-1700 or 1700-2000
  - Display delivery address. Allow changes to the delivery address.
  - Option to receive SMS 30mins before arrival. Customer will be charged S$0.5. By default, option is not selected.
  - Option to vacuum pack all boxes. Customer will be charged a one-time cost of S$1. By default, this option is not selected.

(a) Design **TWO (2)** graphical user interfaces for the 'Registration" focusing on functionalities and user friendliness. State any assumptions clearly.

(12 marks)

(b) Design **ONE (1)** graphical user interface each for both the 'Browse product catalog" and "View product details" features focusing on functionalities and user friendliness. State any assumptions clearly.

(12 marks)

(c) Design **ONE (1)** graphical user interface for the 'Order delivery page" feature focusing on functionalities and user friendliness. State any assumptions clearly.

(6 marks)

(d) List and describe **TWO (2)** back-end administration features that will be useful to the owner of Durian888. State any assumptions clearly.

(6 marks)

(e) List and describe the server-side software required to successfully deploy the web application.

(4 marks)

PART TWO: Answer TWO (2) of the THREE (3) questions.

2. (a) The "HTTP" is the most common form of protocol used for client-server communication via the Internet. Describe with the help of code snippets and/or diagrams how a typical request-response works between a Client, Server and database. State any assumptions clearly.

(10 marks)

(b) Compare and contrast Java Servlets and JSP in the context of server side dynamic Web page generation. Use appropriate code example to illustrate your solution.

(10 marks)

(c) There is a server-side program (servlet) that allows users to pay for the final orders to Durian888.com (Refer to Question 1 for more information). The Servlet will verify if the credit card number entered is valid. Assuming all user data is stored in a database. Write Java Servlet(s) and/or JSP that meets the following specifications:

- In **Checkout.jsp**, registered user input the required Credit card number ,CCV number and Card date of expiry. The parameters values are sent to a servlet **PaymentServlet**. The parameters can be retrieved using the **getParameter** method and **getParameterValues** method.

- There is an existing method, **DBController.verifyCreditCard** (CreditCardNum, CCV, DateOfExpiry) that verifies if the user's credit card info exists in the database. It returns a boolean value:

    - True – if credit card info match in the database.
        - Servlet will redirect user to verifyOTP.jsp. If the OTP entered is correct, the system will redirect user via a servlet to Delivery.jsp.

    - False – if the credit card info is wrong.
        - Servlet will redirect user to Checkout.jsp again. System should then display the message : "Credit card information provided is not correct. Please try again."

    - Database Controller Method to verify OTP challenge :
        - DBController.verifyOTP(String HP, String OTP)
        - Return 'True' if OTP challenge is correct. Return 'False' if data entered is incorrect.

Note: Question No. 2 continues on Page 5

- You do not need to write the entire Servlet. Only the doGet method is required. Additional methods and fields may be created if necessary.
- Perform session tracking whenever necessary.
- State any assumptions clearly.

(10 marks)

3. (a) The code listed in Appendix Q3a is a sample quiz based on an XML data format. Convert the XML data to a JSON format.

(8 marks)

(b) List and describe **FOUR (4)** differences between XML and JSON.

(8 marks)

(c) Figure Q3c shows the "Search orders" page from an e-commerce website used by an administrator. A JSP page **SearchOrders.jsp** allows the administrator to search and display a list of orders according to the purchased date. After submitting a search request, an **AJAX request** is sent to a server-side program that returns a search results in **JSON or XML format**. If no results are being return, a message "No data is available" is displayed instead of the table.

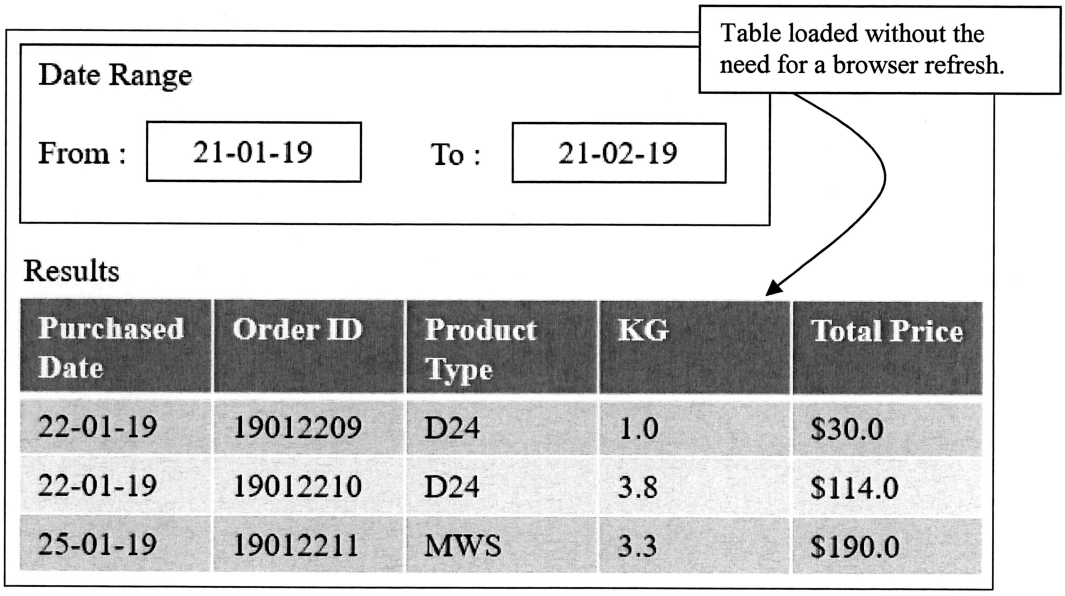

Figure Q3c. Search Orders by purchased date (SearchOrders.jsp)

Note: Question No. 3 continues on Page 6

Write an AJAX application that supports the functionality as described above that meets the following specifications:

- The content of the table shown in figure Q3c is loaded without requiring the screen to perform any refresh.
- Assume that the server-side program exists and its absolute path is **/SearchOrders**
- User enters the search string.
- The server-side application accepts one query string parameter named **searchString** as input field from **SearchOrders.jsp**.
- The server-side application will return data in XML or JSON format. Propose and design your own XML or JSON file and format.
- Provide sufficient comments in your code.
- A graphical user interface is shown in Figure Q3c.
- State any assumptions clearly.

You are to use the code listed in Appendix Q3c. Comments are provided as a usage guide. Do not reproduce the code in your answers unless there are modifications. State any assumptions made.

(14 marks)

4. (a) Figure Q4a-1 shows a HTML page with a sentence "Check out our special price today!".

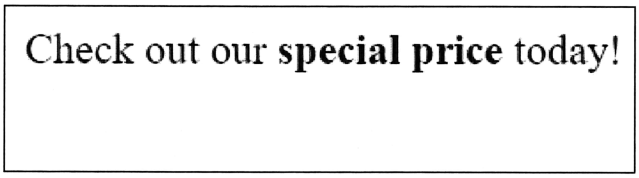

Figure Q4a-1. Before moving mouse over "special price".

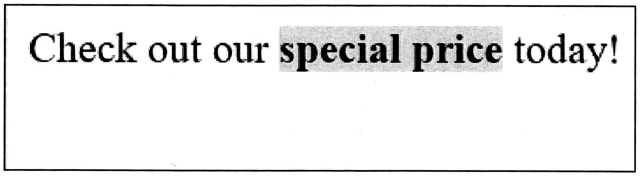

Figure Q4a-2. After moving mouse over "special price".

Note: Question No. 4 continues on Page 7

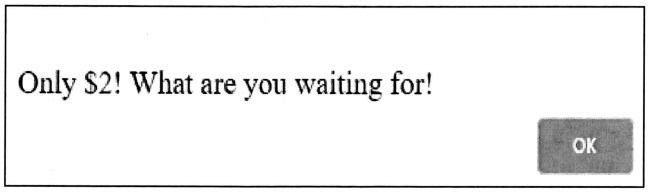

Figure Q4a-3. A pop-up message.

- Moving the mouse above the word "special price" will have 2 effects:
  - It will cause the background color of the word "special price" to turn Yellow as shown in figure Q4a-2.
  - A pop-up message displaying "Only $2! What are you waiting for!" as shown in Figure Q4a-3.

Write jQuery code that supports the functionality described. Part of the code is provided in Appendix Q4a.

(10 marks)

(b) Figure Q4b-1 shows a HTML page with 2 panels labelled as "PANEL 1" and "PANEL 2".

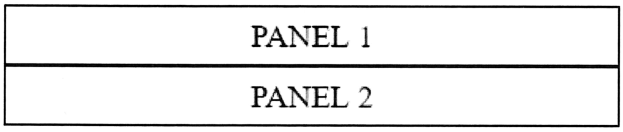

Figure Q4b-1. Before clicking on any of the panels.

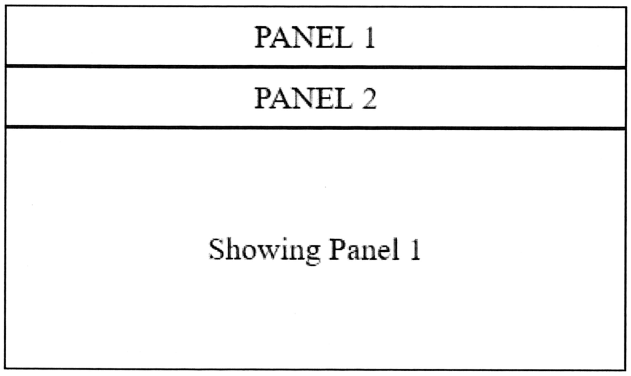

Figure Q4b-2. After clicking on "Panel 1".

Note: Question No. 4 continues on Page 8

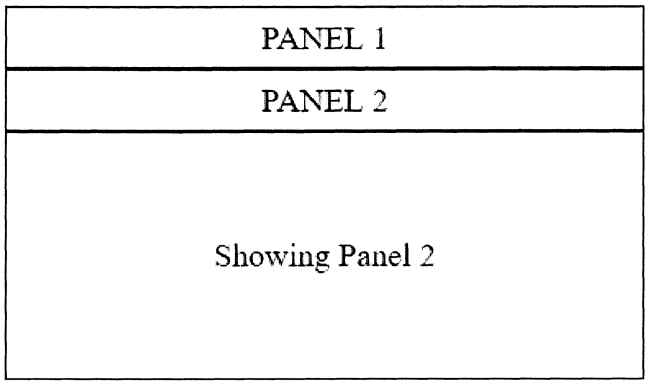

Figure Q4b-3. After clicking on "Panel 2".

- As shown in Figure Q4b-2, clicking on "Panel 1" will cause another panel to slide out, display the text "Showing Panel 1". Clicking on "Panel 1" again will cause the panel to slide back in.
- Similarly as shown in Figure Q4b-3, clicking on "Panel 2" will cause another panel to slide out, display the text "Showing Panel 2". Clicking on "Panel 2" again will cause the panel to slide back in.
- CSS styles for the panels
  - CSS style for all 4 panels
    - padding: 5px;
    - text-align: center;
    - border: solid 1px;
  - CSS style for panel with text "Showing Panel 1" and "Showing Panel 2"
    - padding: 50px;
    - display: none;

Write jQuery code that supports the functionality described. Part of the code is provided in Appendix Q4b.

(10 marks)

Note: Question No. 4 continues on Page 9

(c) Figure Q4c-1 shows a HTML page with a two-section jQuery accordion UI namely "ADMINISTRATION" and "SEND MESSAGE".

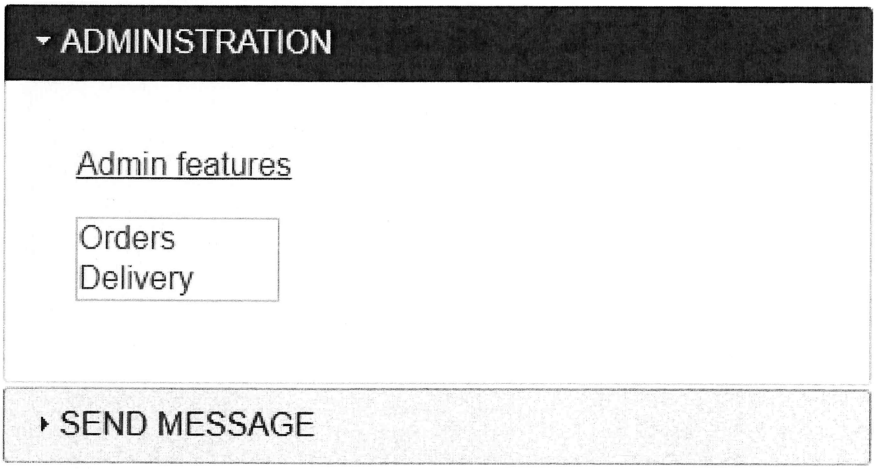

Figure Q4c-1. After clicking on "ADMINISTRATION"

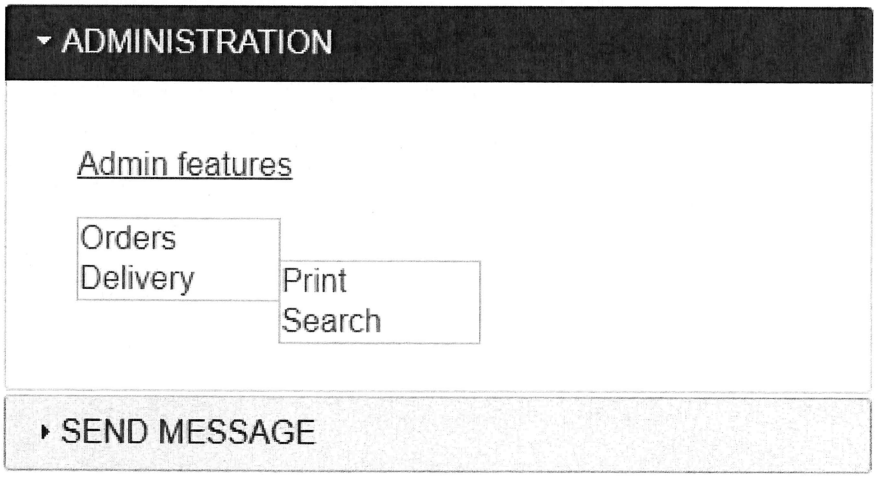

Figure Q4c-2. After clicking on "Delivery" menu item

Note: Question No. 4 continues on Page 10

- As shown in Figure Q4c-1, clicking on the "ADMINISTRATION" section displays a jQuery menu with 2 items namely "Orders" and "Delivery".
- As shown in Figure Q4c-2, clicking on either the "Orders" or "Delivery" menu item, a sub-menu with items labelled as "Print" and "Search" will be displayed.
- SEND MESSAGE section only displays a text "Message info"

Write jQuery code that supports the functionality described. Part of the code is provided in Appendix Q4b.

(10 marks)

END OF PAPER

APPENDIX Q3a

```
<?xml version="1.0" encoding="UTF-8" ?>
<quiz>
<sport>
    <q1>
        <question>When was the first Common Weath Games held?</question>
        <options>1930</options>
        <options>1934</options>
        <options>1938</options>
        <options>1948</options>
        <answer>1930</answer>
    </q1>

    <q2>
        <question>The term Butterfly Stroke is referred to in which sport?</question>
        <options>Wrestling</options>
        <options>Volleyball</options>
        <options>Tennis</options>
        <options>Swimming</options>
        <answer>Swimming</answer>
    </q2>

</sport>

<maths>
    <q1>
        <question>5 + 7 = ?</question>
        <options>10</options>
        <options>11</options>
        <options>12</options>
        <options>13</options>
        <answer>12</answer>
    </q1>
</maths>

</quiz>
```

APPENDIX Q3c

```
var request=null;
var data=null;
var READY_STATE_COMPLETE=4;


// This is the entry point for communicating with the
// server-side application. The function accepts the URL // of the application. Since the
GET method is used, any
// parameters must be included as part of the URL.

function sendRequest(url) {
    request = createRequest();
    if (request == null) {
        return;
    }
    request.open("GET", url, true);
    request.send(null);
    request.onreadystatechange = processRequestChange;
}

function createRequest() {

    var req = null;

    if (XMLHttpRequest) {
        req = new XMLHttpRequest();
    }
    else if (ActiveXObject ) {
        req = new ActiveXObject("Microsoft.XMLHTTP");
    }
    else {
        req = null;
    }
    return req;
}

// This function invokes a renderData() function
// when the server-side application has finished
// sending its XML data to the XMLHttpRequest object.
// You will have to implement the renderData() function
// to display the data on the Web page.
```

Note: Appendix Q3c continues on Page iii

```
function processRequestChange() {

    if(request == null) {
        return;
    }

    if(request.readyState == READY_STATE_COMPLETE) {
        data = request.responseXML;
        renderData(data);
    }
}
```

APPENDIX Q4a

```
<html>
<head>
<title>jQuery Q4a</title>
<link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css">
<script src="http://code.jquery.com/jquery-3.4.1.js"> </script>

        … (Add additional codes here)

</head>
<body>

Check out our special price today!

        … HTML : Make necessary changes to the HTML


</body>
</html>
```

APPENDIX Q4b

```
<html>
<head>
<title>jQuery Q4b</title>
<link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css">
<script src="http://code.jquery.com/jquery-3.4.1.js"> </script>

        … (Add additional codes here)

</head>
<body>

<div>PANEL 1</div>
<div>PANEL 2</div>
Showing Panel 1
Showing Panel 2

        … HTML : Make necessary changes to the HTML


</body>
</html>
```

APPENDIX Q4c

```
<html>
<head>
<title>jQuery Q4c</title>
<link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css">
<link rel="stylesheet" href="/resources/demos/style.css">
<script src="https://code.jquery.com/jquery-1.12.4.js"></script>
 <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
```

… (Add additional codes here)

```
<style>
  .ui-menu { width: 100px; }
</style>

</head>
<body>

<h3>ADMINISTRATION</h3>
<h3>SEND MESSAGE</h3>
<p>Message info.</p>
```

… HTML : Make necessary changes to the HTML

```
</body>
</html>
```

END OF APPENDICES

CI6206 INTERNET PROGRAMMING

Please read the following instructions carefully:

1. **Please do not turn over the question paper until you are told to do so. Disciplinary action may be taken against you if you do so.**

2. You are not allowed to leave the examination hall unless accompanied by an invigilator. You may raise your hand if you need to communicate with the invigilator.

3. Please write your Matriculation Number on the front of the answer book.

4. Please indicate clearly in the answer book (at the appropriate place) if you are continuing the answer to a question elsewhere in the book.