

# SOEN 387

# WEB-BASED ENTERPRISE

# APPLICATIONS DESIGN

TUTORIAL – 10

Design Problems

## In class Exercise Exam Question

Create a small web app that receives anonymous complaints from external users.

- The complaint data must contain user's name, email address, contact number, and complain text with a possibility of a pdf attachment.
- In order to protect it against spammers, you implement a captcha service that displays a random image (i.e. via a servlet).
- The user then selects an appropriate answer from the dropdown list that can be verified by the server.
- In case of errors (i.e. incomplete data and/or wrong captcha answer), the user must re-enter the data.

## Design Approach



Design web app by providing the following items:



i) Web Pages / Servlets: List and specify dynamic web pages, servlets, as well static content.



ii) Class Diagram (Business Logic Layer): Specify entity classes, utility / helper classes, and other business classes.



iii) A Use-Case Scenario: Provide a sample scenario that the user enters the complaint by using the captcha servlet



iv) Beans: List and describe any bean that you may use to implement the above scenario.



v) Sequence Diagram: Draw the sequence diagram implementing the above scenario

# Complaint.jsp

UI comes here

```
<jsp:useBean id="captcha" class="model.Captcha" scope="session"></jsp:useBean>
<jsp:useBean id="complaint" class="model.Complaint" scope="request"></jsp:useBean>
<body>
<form method="post" action="ComplaintController" ENCTYPE="multipart/form-data">
  Name:<br>
  <input type="text" name="name" value="<%=complaint.setName(request.getParameter("name"))%>" >
  <br>
  Email:<br>
  <input type="text" name="email" value="<%=complaint.setEmail(request.getParameter("email"))%>"
  <br><br>
  Contact Information:<br>
  <input type="text" name="phone" value="<%=complaint.setPhone(request.getParameter("phone"))%>"
  <br><br>
  Complaint Text:<br>
  <textarea rows="10" cols="10">Please enter your complaint</textarea>
  Attachment: <INPUT TYPE="file" NAME="attachedfile" MAXLENGTH=50 >
  <input type="submit" value="Submit">
  <br> <br>
  <!-- Retrieve a random image from file src -->
  
  <br>
  Please enter fruit name in image :<br>
  <select id="fruit" name="fruit" value="<%=complaint.setFruit(request.getParameter("fruit"))%>"
  <option value="Apple">Apple</option>
  <option value="Mango">Mango</option>
  <option value="DragonFruit">DragonFruit</option>
  </select>
  <input type="submit" value="Submit"> <br> <br>
</form>
</body>
</html>
```

using the "img src", the captcha image is requested (via servlet).

then user enters data and captcha answer

# ComplaintController.java

```
public class ComplaintController extends HttpServlet {  
    public ComplaintController() {}  
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,  
        IOException {  
  
        String name= request.getParameter("name");  
        String email= request.getParameter("email");  
        String phone= request.getParameter("phone");  
        String complaint= request.getParameter("complaint");  
  
        String fruit=request.getParameter("fruit");  
        Complaint obj=new Complaint(name, email,phone, complaint, fruit);  
        //Complaint object created and data answer passed  
        request.setAttribute("complaint",obj);  
  
        //get captcha from session.  
        Captcha captchaObj =(Captcha)request.getSession().getAttribute("captcha");  
        RequestDispatcher rd;  
        if(captchaObj.validateCaptcha(obj.getFruit())) {  
            //image name is stored as actual answer  
            rd=request.getRequestDispatcher("/ComplaintReceived.jsp");  
            rd.forward(request, response);  
        }  
        else{  
            rd=request.getRequestDispatcher("/Complaint.jsp");  
            rd.forward(request, response);  
        }  
    }  
}
```

SOEN 387 WEB-BASED ENTERPRISE APPLICATIONS DESIGN

# Complaint.java

Insert your Code for Complaint class for business logic and saving complaint details

```
public class Complaint {
    private String name, email, phone,complaint,fruit;
    public Complaint(){}
    public Complaint(String name, String email, String phone, String
complaint, String fruit){
        this.name=name;
        this.phone=phone;
        this.email=email;
        this.complaint=complaint;
        this.fruit=fruit;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public String getPhone() {
        return phone;
    }
    public void setPhone(String phone) {
        this.phone = phone;
    }
    public String getcomplaint() {
        return complaint;
    }
    public void setcomplaint(String complaint) {
        this.complaint = complaint;
    }
    public String getFruit() {
        return fruit;
    }
    public void setFruit(String fruit) {
        this.fruit = fruit;
    }
}
```



# CaptchaDisplay.java

```
public class CaptchaDisplay extends HttpServlet {  
  
    protected void doGet(HttpServletRequest request, HttpServletResponse  
        response) throws ServletException, IOException {  
  
        Captcha obj=(Captcha)(request.getSession().getAttribute("captcha"));  
        //get servlet context from session bean  
        obj.selectRandomCaptcha();  
        //Here we will be displaying the image  
        response.setContentType(obj.getMimeType());  
        ServletOutputStream out = response.getOutputStream();  
        obj.getImageData(out);  
        out.close();  
  
    }  
  
}
```

# Captcha.java

Insert your Code for Captcha class for business logic and validating the captcha

```
public class Captcha {

    String imgName;
    String mime;
    public String getMimeType(){
        return mime;
    }
    public boolean validateCaptcha(String actualAnswer){
        if(imgName.contains(actualAnswer))
            return true;
        else return false;
    }

    public void selectRandomCaptcha() { //select random
        File file = new File("./Path to the files comes here");
        //file location has 3 images, so retrieve them in an arraylist
        ArrayList<String> ImgFiles = new ArrayList<>();
        for (final File fileEntry : file.listFiles()) {
            ImgFiles.add(fileEntry.getName());
        }
        Random rand = new Random();
        imgName = ImgFiles.get(rand.nextInt(ImgFiles.size()));
        //set mime type
        mime="image/jpeg";
    }
}
```

```
public void
getImageData(ServletOutputStream out){
    try {
        FileInputStream fin = new
        FileInputStream(imgName);
        BufferedInputStream bin = new
        BufferedInputStream(fin);
        BufferedOutputStream bout = new
        BufferedOutputStream(out);

        int ch = 0;
        ;
        while ((ch = bin.read()) != -1) {
            bout.write(ch);
        }

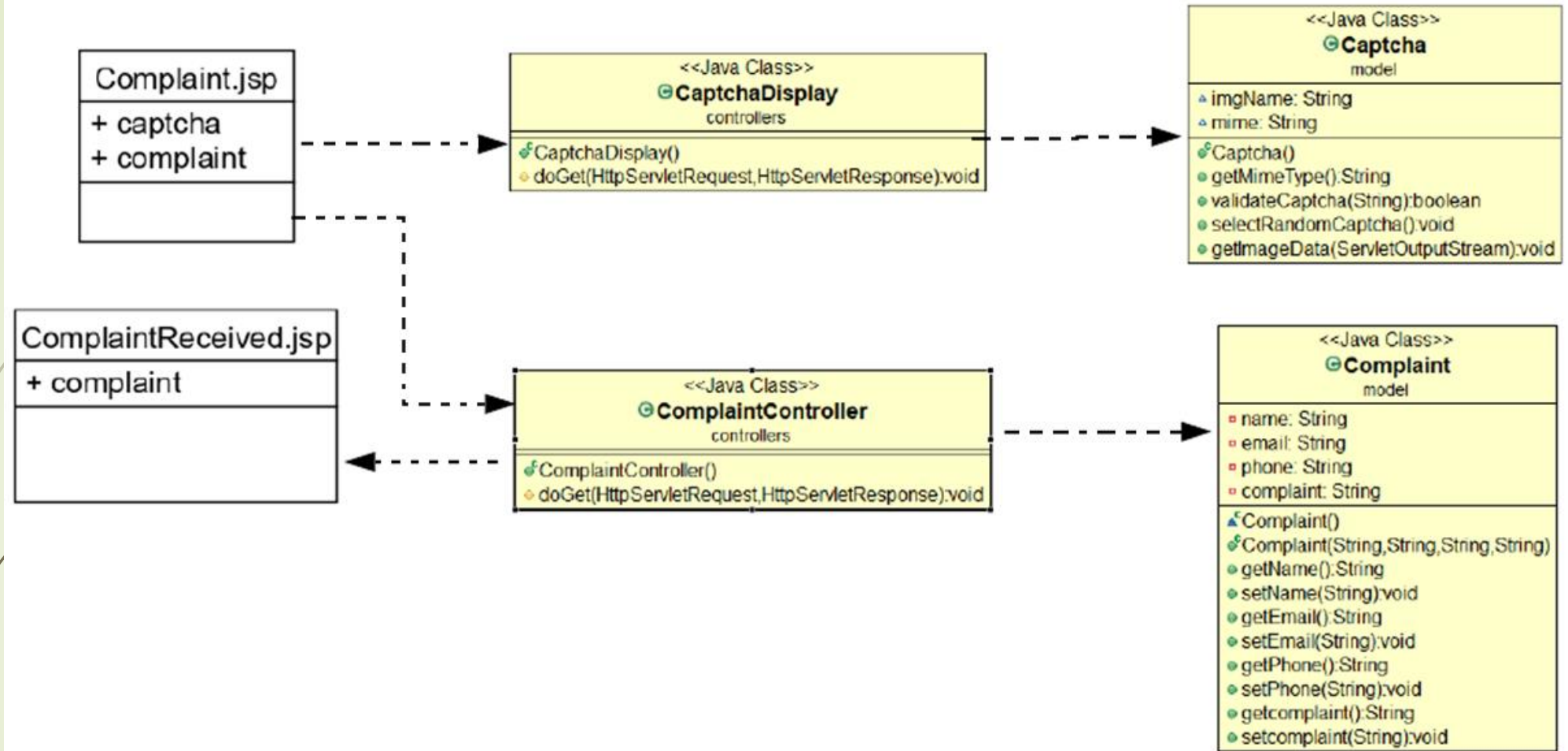
        bin.close();
        fin.close();
        bout.close();
    }
    catch (Exception e){
        System.out.println("Error is
        caught for "+e.getMessage());
    }
    finally {

    }
}
```



# ComplaintReceived.java

```
<jsp:useBean id= "complaint " class= "model.Complaint"
scope="request">
<%
    String complaintData=complaint.getcomplaint();
    out.print("Thanks! Your complaint is successfully captured");
%>
</jsp:useBean>
```

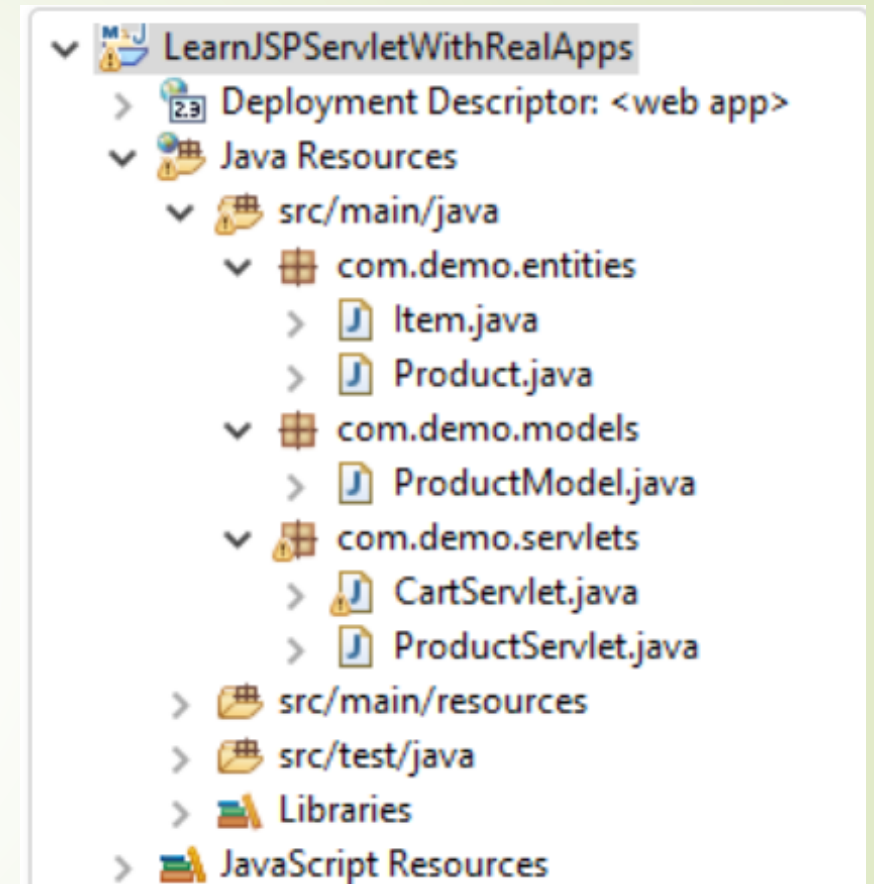


# Class Diagram

# Practise Exercise

Create a small web app for shopping cart :

- We need ProductPage and CartPage where ProductPage will display the products that can select and add it in your CartPage.
- We need entities as Product Entity, Item Entity.
- We need ProductServlet and CartServlet.



[Refer: http://learningprogramming.net/java/jsp-servlet/shopping-cart-with-session-in-jsp-servlet/](http://learningprogramming.net/java/jsp-servlet/shopping-cart-with-session-in-jsp-servlet/)

# To DO

- Create a Class Diagram
- Create a Sequence Diagram

# References

- <https://www.roseindia.net/servlets/web-application.shtml>
- [http://people.cs.ksu.edu/~reshma/798\\_ClassDiagram.htm](http://people.cs.ksu.edu/~reshma/798_ClassDiagram.htm)
- <https://o7planning.org/en/10285/create-a-simple-java-web-application-using-servlet-jsp-and-jdbc#a851015>
- <https://www.websequencediagrams.com/>