Product Details Portal

Algorithm:

\* Start the web application.

\* Define a class with fields for product ID, name, description, and price.

\* Create a Servlet class that extends HttpServlet.

\* Override the doPost method in the Servlet to handle the form submission.

\* Inside the doPost method:

Retrieve the form data from the HttpServletRequest object.

Parse the product ID, name, description, and price from the form data.

Validate the mandatory fields (name and description).

If the entered fields are valid:

▪ Create a new Product object with the parsed data.

▪ Store the product object in the HttpSession.

▪ Redirect to the productDisplay.jsp page.

o If the entered fields are not valid:

▪ Set an error message as a request attribute.

▪ Forward the request to the jsp page to display the error message.

\* Create a productDetails.jsp file with a form to enter product details.

o Form should include input fields as product ID, name, description, price.

o Name and description fields should be marked as required.

o Display the error message (if any) using JSP tags.

o Submit the form to the ProductServlet using the POST method.

\* Create a productDisplay.jsp file to display the product details.

o Retrieve the product object from the HttpSession.

o Display the product ID, name, description, and price using JSP tags.

\*Configure the web.xml file to map the Servlet to the appropriate URL pattern

("/productDetails").

\* Deploy the web application to a servlet container (e.g., Tomcat).

\*Access the productDetails.jsp page through the web application URL.

\*Enter the product details in the form and submit it.

\*If the fields are valid, the productDisplay.jsp page will be displayed with the entered

product details.

\*If the fields are not valid, the productDetails.jsp page will be displayed with an error message