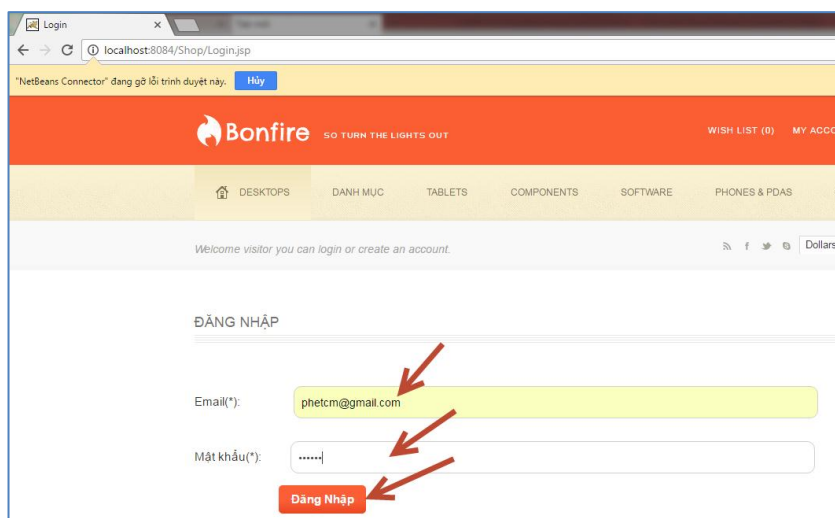


Lab 09:**XÂY DỰNG CHỨC NĂNG THANH TOÁN****A. MỤC TIÊU:**

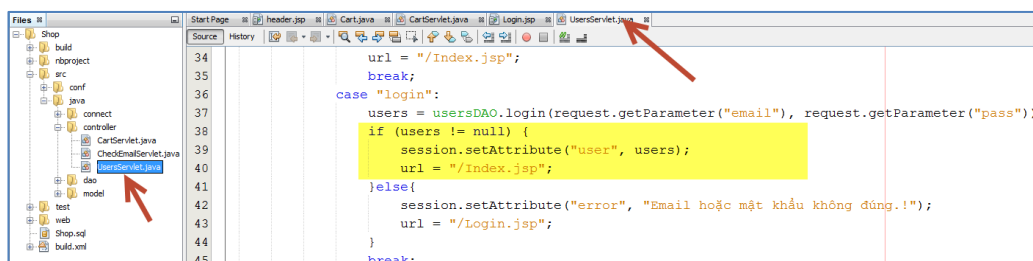
- ✓ Ở các Lab trước đã thực hiện các trang: Sản phẩm theo từng danh mục, Chi tiết sản phẩm, giỏ hàng, đăng ký, đăng nhập.
- ✓ Lab này sẽ thực hiện trang thanh toán đơn hàng sau khi người dùng chọn xong sản phẩm cần mua vào giỏ hàng.
- ✓ Khi người dùng nhấn nút thanh toán đơn hàng ở trang giỏ hàng thì sẽ kiểm tra người dùng đã đăng nhập chưa, nếu đã đăng nhập rồi thì sẽ cho người dùng xác nhận thanh toán, còn chưa thì chuyển qua trang đăng ký.
- ✓ Sau khi đăng ký sẽ chuyển qua trang đăng nhập.
- ✓ Mục đích Lab này là lưu được thông tin đơn hàng vào đơn hàng và tiết đơn hàng
- ✓ Sau đó người quản lý sẽ xem lại đơn hàng do khách hàng đặt và xử lý các chức năng như xác nhận hay hủy đơn hàng, ...

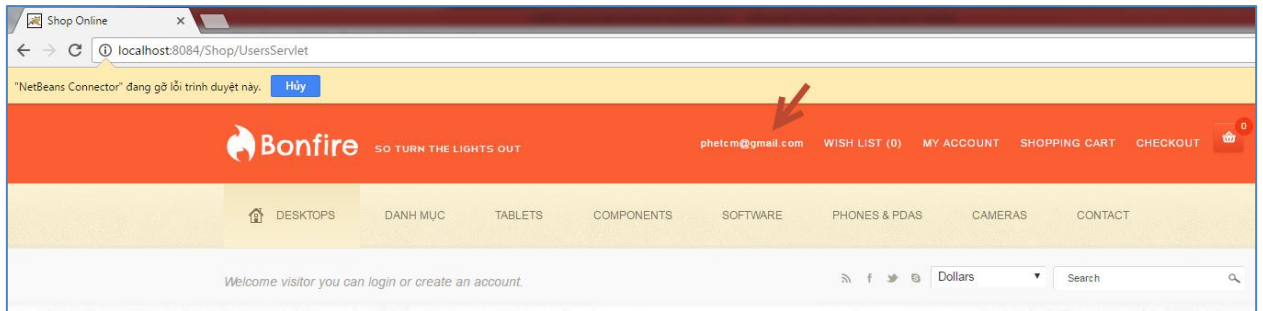
NỘI DUNG:

Chuẩn bị: Xem lại trang đăng nhập, Giỏ hàng



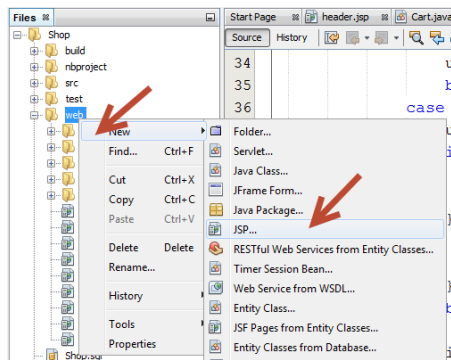
Khi đăng nhập thành công: Dùng biến session để lưu thông tin user đã đăng nhập và cho hiện thị email đăng nhập vào phần header của website:





Bước 1: Thiết kế trang thanh toán (checkout):

- Tạo mới trang JSP:



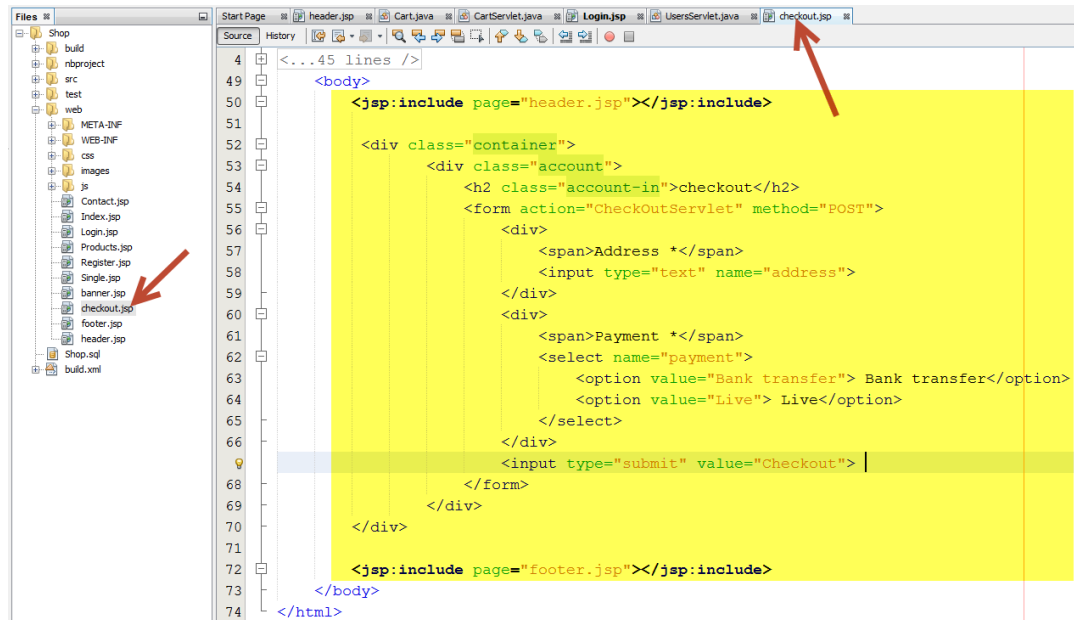
- Sao chép code từ trang login.jsp cho trang checkout.jsp, cập nhật code cho phần thân trang như sau:

```
<body>
<%
    Users users = (Users) session.getAttribute("user");
    if (users == null) {
        response.sendRedirect("/shop/login.jsp");
    }
%>
<jsp:include page="header.jsp"></jsp:include>
<div class="container">
    <div class="account">
        <h2 class="account-in">checkout</h2>
        <form action="CheckOutServlet" method="POST">
            <div>
                <span>Address *</span>
                <input type="text" name="address">
            </div>
            <div>
                <span>Payment *</span>
                <select name="payment">
                    <option value="Bank transfer"> Bank transfer</option>
                    <option value="Live"> Live</option>
                </select>
            </div>
        </form>
    </div>
</div>
```

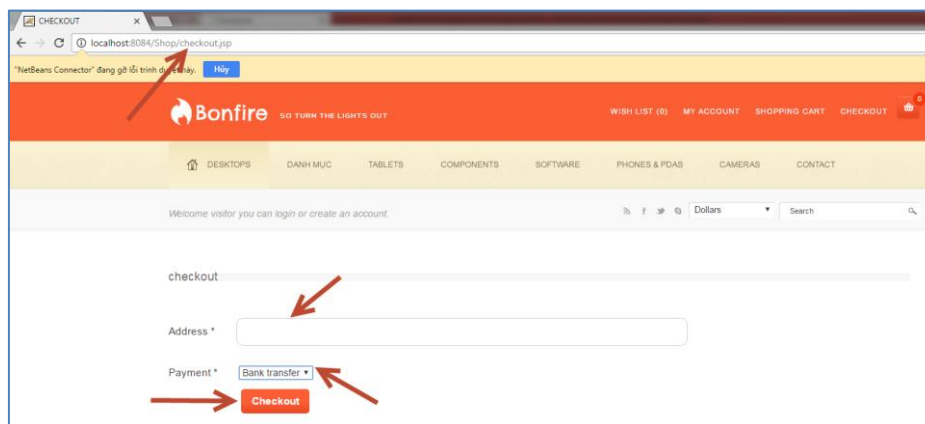
```

</div>
<input type="submit" value="Checkout">
</form>
</div>
</div>
<jsp:include page="footer.jsp"></jsp:include>
</body>

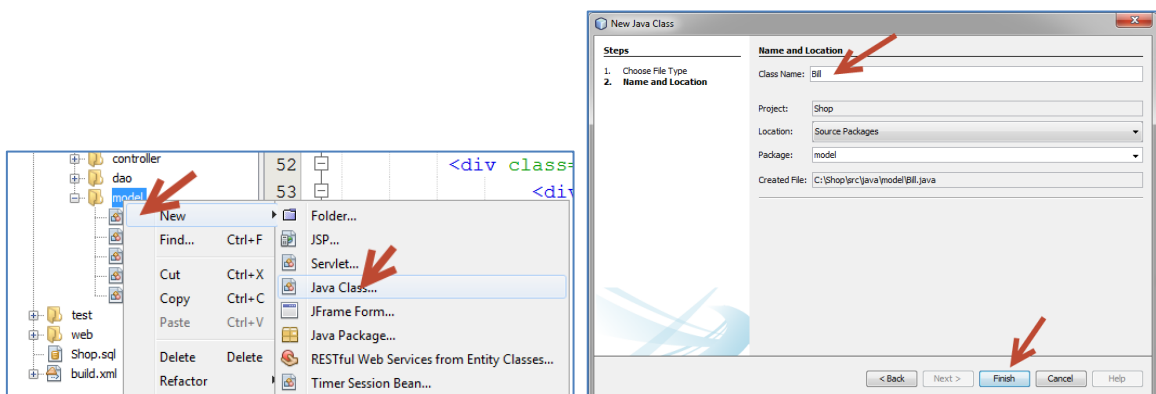
```



- Kết quả:



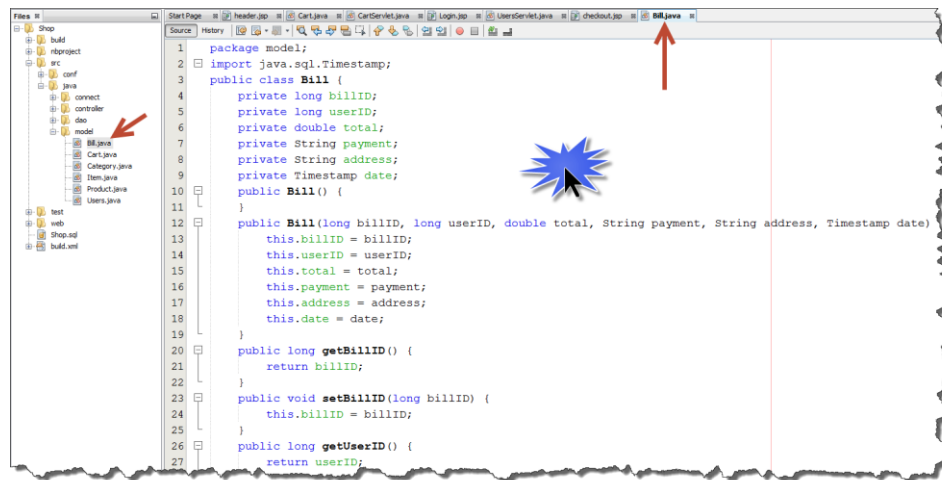
Bước 2: Tạo lớp Hóa đơn, chi tiết hóa đơn để lưu thông tin



- ✓ Tạo và code cho lớp Bill các thuộc tính sau:

```
package model;
import java.sql.Timestamp;
public class Bill {
    private long billID;
    private long userID;
    private double total;
    private String payment;
    private String address;
    private Timestamp date;
}
```

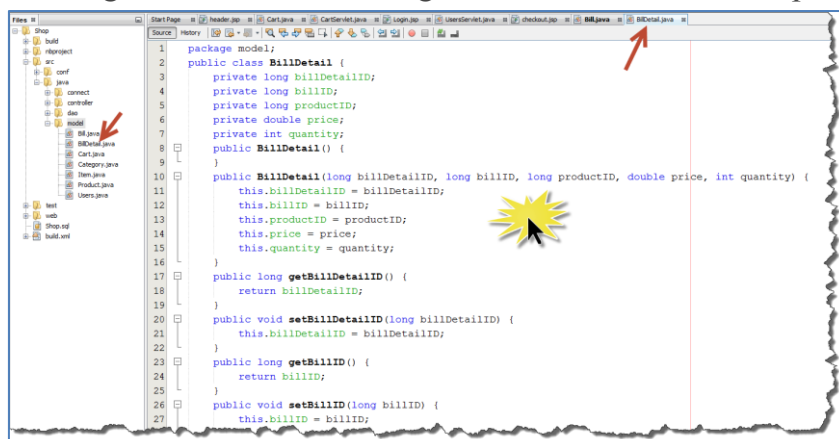
Bổ sung các constructor không tham số, có tham số, các phương thức set, get.



- ✓ Tạo và code cho lớp BillDetail các thuộc tính sau:

```
package model;
public class BillDetail {
    private long billDetailID;
    private long billID;
    private long productID;
    private double price;
    private int quantity;
}
```

Bổ sung các constructor không tham số, có tham số, các phương thức set, get:



Bước 3: Tạo các Table (bill và billdetail) và nhập liệu cho CSDL

-- Table structure for bill

```
DROP TABLE IF EXISTS `bill`;
```

```
CREATE TABLE `bill` (
```

```
`bill_id` bigint(20) NOT NULL,
```

```
`user_id` bigint(20) default NULL,
```

```
`total` double default NULL,
```

```
`payment` varchar(255) collate utf8_unicode_ci default NULL,
```

```
`address` longtext collate utf8_unicode_ci,
```

```
`date` timestamp NULL default NULL on update CURRENT_TIMESTAMP,
```

```
PRIMARY KEY (`bill_id`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

-- Records of bill

```
INSERT INTO `bill` VALUES ('1463295350519', '1463056081952', '13', 'Live', '123 - Q1 - TPHCM', '2016-05-15 13:55:50');
```

bill_id	user_id	total	payment	address	date
1463295350519	1463056081952	13	Live	123 - Q1 - TPHCM	2016-05-15 13:55:50

-- Table structure for bill_detail

```
DROP TABLE IF EXISTS `bill_detail`;
```

```
CREATE TABLE `bill_detail` (
```

```
`bill_detail_id` bigint(20) NOT NULL auto_increment,
```

```
`bill_id` bigint(20) default NULL,
```

```
`product_id` bigint(20) default NULL,
```

```
`price` double default NULL,
```

```
`quantity` int(11) default NULL,
```

```
PRIMARY KEY (`bill_detail_id`)
```

```
) ENGINE=MyISAM AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

-- Records of bill_detail

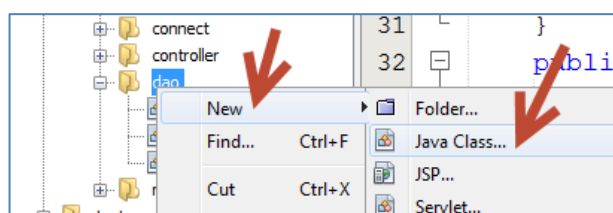
```
INSERT INTO `bill_detail` VALUES ('4', '1463295350519', '1', '2', '1');
```

```
INSERT INTO `bill_detail` VALUES ('5', '1463295350519', '2', '3', '1');
```

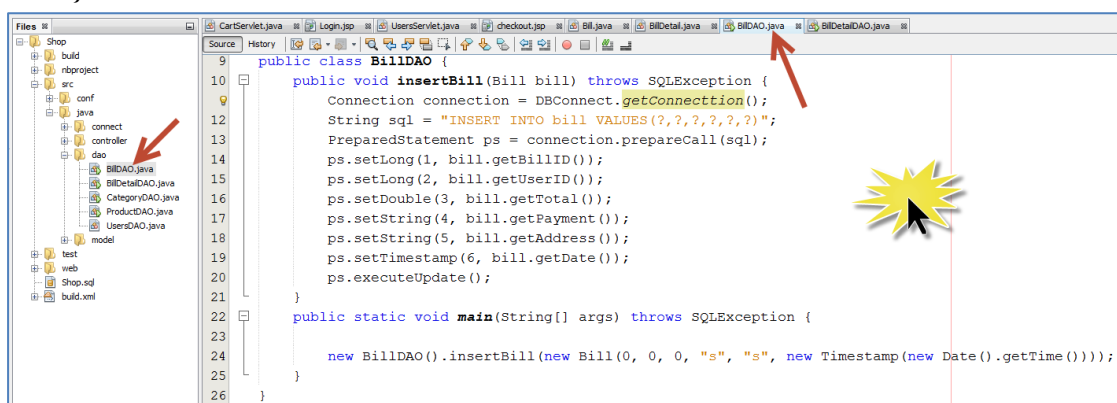
```
INSERT INTO `bill_detail` VALUES ('6', '1463295350519', '3', '4', '2');
```

bill_detail_id	bill_id	product_id	price	quantity
4	1463295350519	1	2	1
5	1463295350519	2	3	1
6	1463295350519	3	4	2

Bước 4: Tạo lớp Bill trong package dao để xử lý thêm Hóa đơn:



```
package dao;
import connect.DBConnect;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.sql.Timestamp;
import java.util.Date;
import model.Bill;
public class BillDAO {
    public void insertBill(Bill bill) throws SQLException {
        Connection connection = DBConnect.getConnection();
        String sql = "INSERT INTO bill VALUES(?,?,?,?,?,?)";
        PreparedStatement ps = connection.prepareStatement(sql);
        ps.setLong(1, bill.getBillID());
        ps.setLong(2, bill.getUserID());
        ps.setDouble(3, bill.getTotal());
        ps.setString(4, bill.getPayment());
        ps.setString(5, bill.getAddress());
        ps.setTimestamp(6, bill.getDate());
        ps.executeUpdate();
    }
    public static void main(String[] args) throws SQLException {
        new BillDAO().insertBill(new Bill(0, 0, 0, "s", "s", new Timestamp(new
        Date().getTime())));
    }
}
```



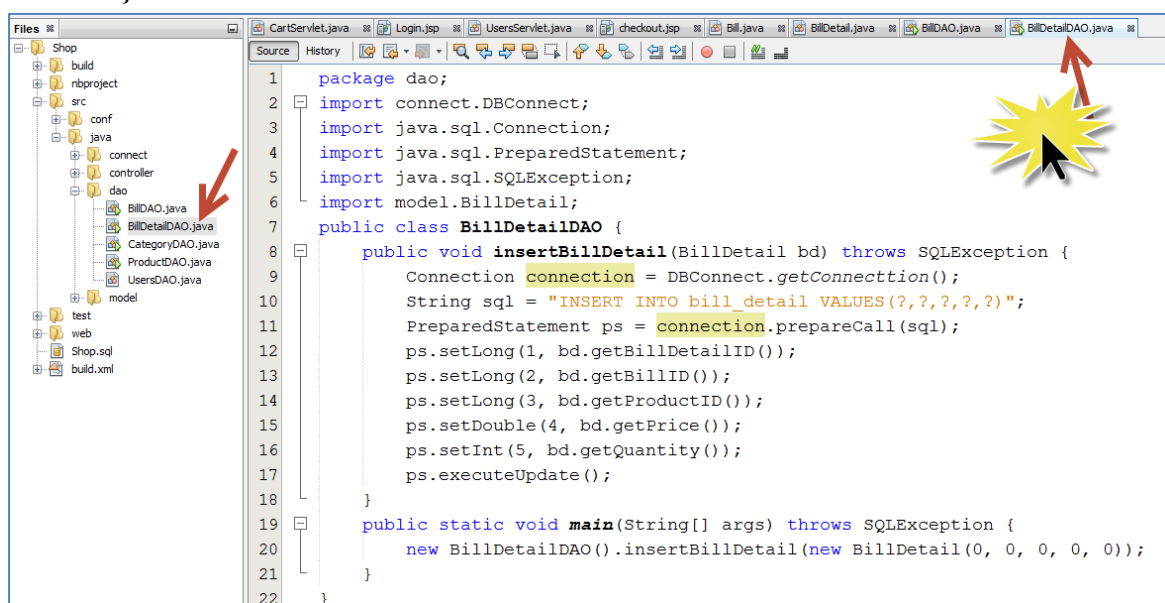
Thực thi kiểm tra:

bill_id	user_id	total	payment	address	date
1463295350519	1463056081952	13	Live	123 - Q1 - TPHCM	2016-05-15 13:55:50
0	0	0	s	s	2016-10-08 15:09:13

Tạo lớp BillDetailDAO trong package dao để xử lý thêm chi tiết hóa đơn:

```
package dao;
import connect.DBConnect;
```

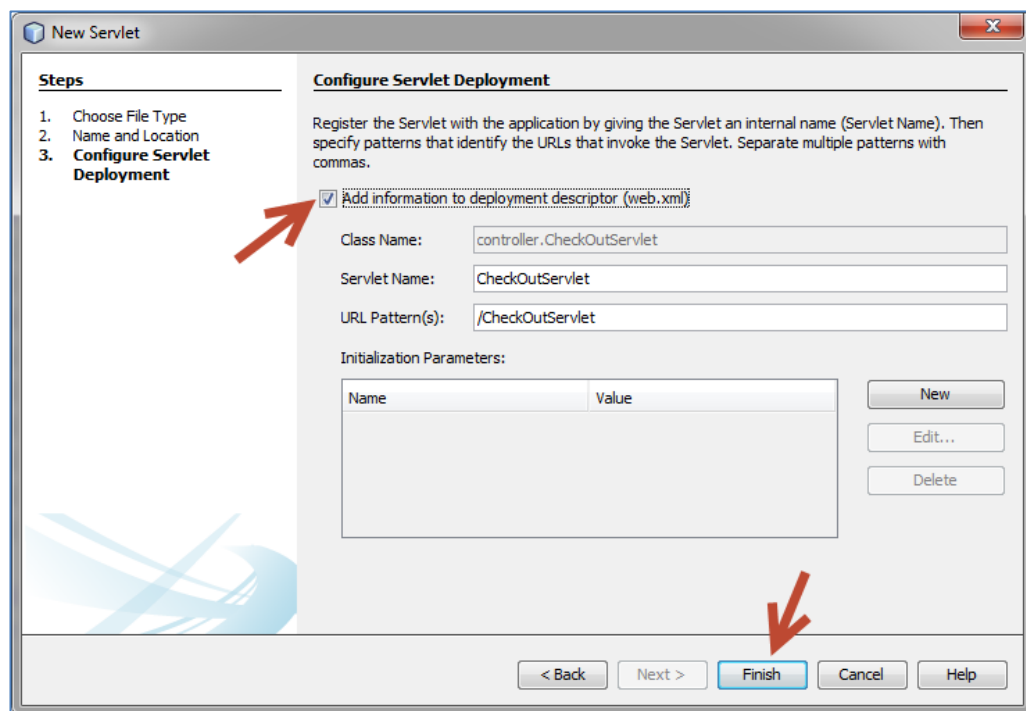
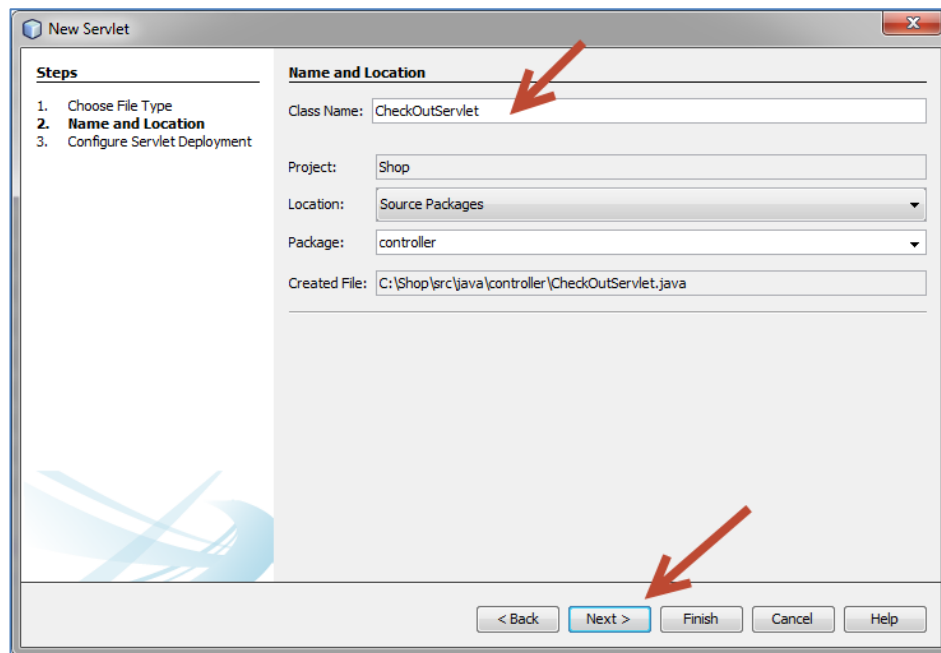
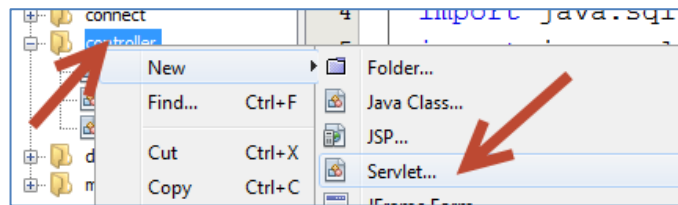
```
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import model.BillDetail;
public class BillDetailDAO {
    public void insertBillDetail(BillDetail bd) throws SQLException {
        Connection connection = DBConnect.getConnection();
        String sql = "INSERT INTO bill_detail VALUES(?,?,?,?,?)";
        PreparedStatement ps = connection.prepareStatement(sql);
        ps.setLong(1, bd.getBillDetailID());
        ps.setLong(2, bd.getBillID());
        ps.setLong(3, bd.getProductID());
        ps.setDouble(4, bd.getPrice());
        ps.setInt(5, bd.getQuantity());
        ps.executeUpdate();
    }
    public static void main(String[] args) throws SQLException {
        new BillDetailDAO().insertBillDetail(new BillDetail(0, 0, 0, 0, 0));
    }
}
```



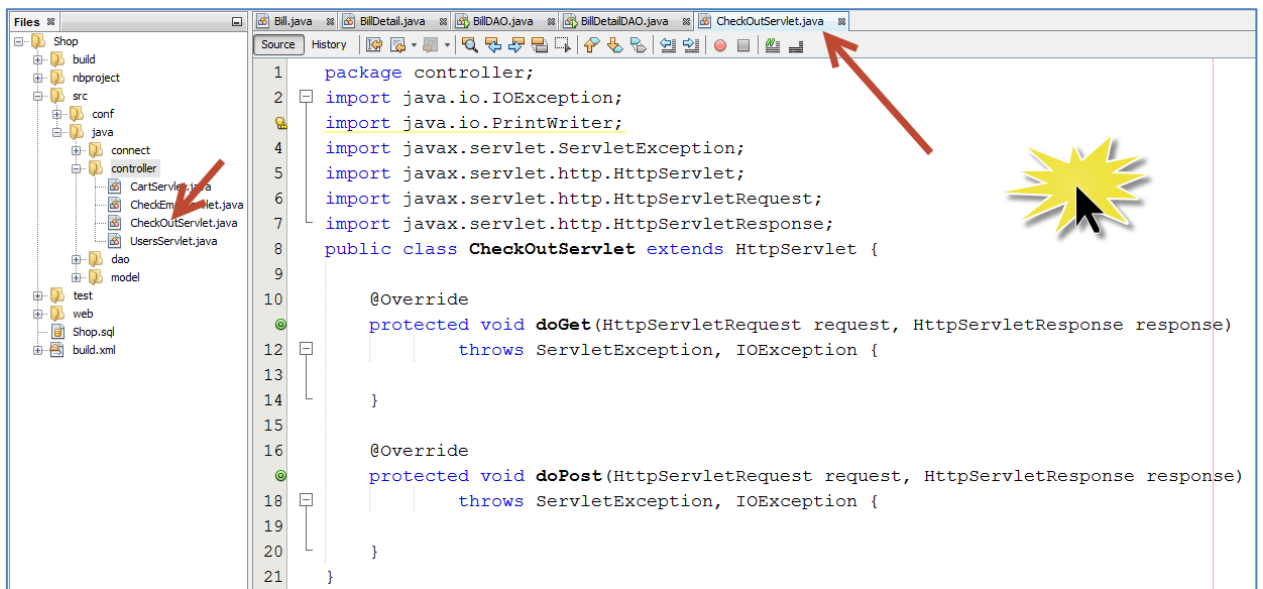
Thực thi kiểm tra:

bill_detail_id	bill_id	product_id	price	quantity
4	1463295350519	1	2	1
5	1463295350519	2	3	1
6	1463295350519	3	4	2
7	0	0	0	0

Bước 5: Tạo Servlet xử lý trang checkout.



Xóa các code hiện tại còn lại như sau:



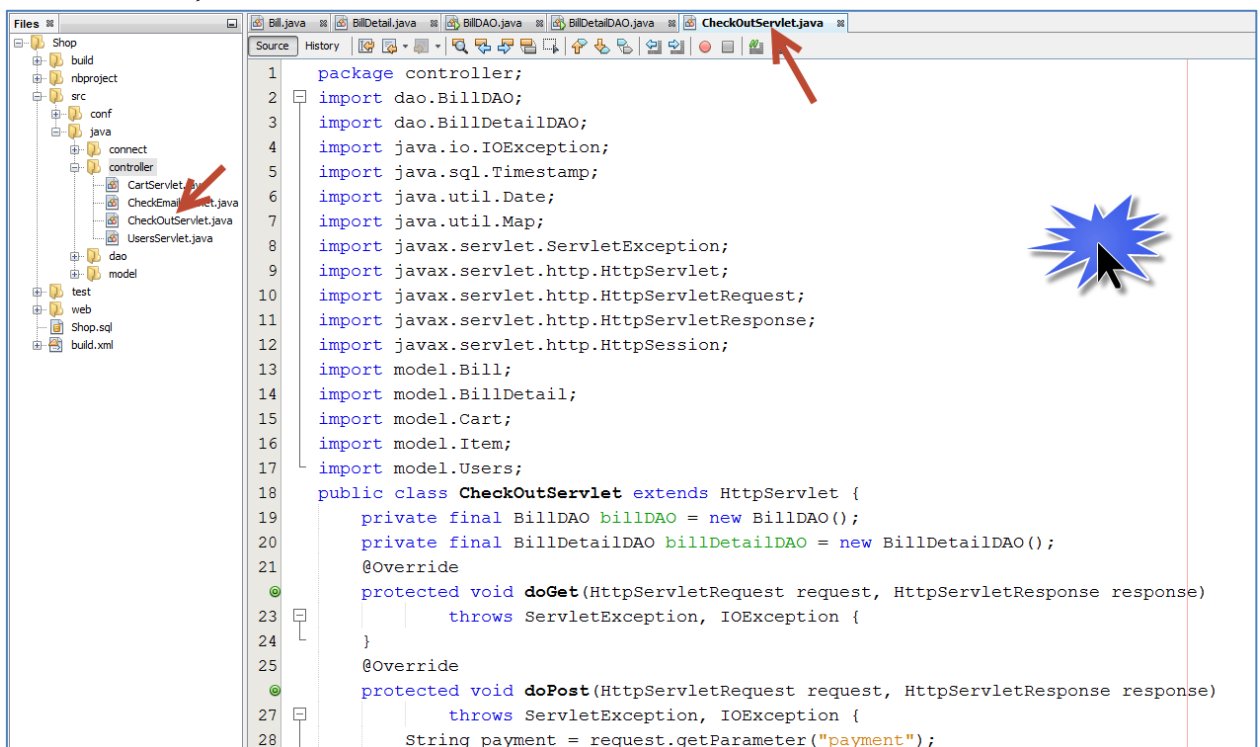
Cập nhật thêm code như sau:

```
package controller;
import dao.BillDAO;
import dao.BillDetailDAO;
import java.io.IOException;
import java.sql.Timestamp;
import java.util.Date;
import java.util.Map;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import model.Bill;
import model.BillDetail;
import model.Cart;
import model.Item;
import model.Users;
public class CheckOutServlet extends HttpServlet {
    private final BillDAO billDAO = new BillDAO();
    private final BillDetailDAO billDetailDAO = new BillDetailDAO();
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    }
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        String payment = request.getParameter("payment");
        String address = request.getParameter("address");
        HttpSession session = request.getSession();
        Cart cart = (Cart) session.getAttribute("cart");
        Users users = (Users) session.getAttribute("user");
```

```

try {
    long ID = new Date().getTime();
    Bill bill = new Bill();
    bill.setBillID(ID);
    bill.setAddress(address);
    bill.setPayment(payment);
    bill.setUserID(users.getUserID());
    bill.setDate(new Timestamp(new Date().getTime()));
    bill.setTotal(cart.total());
    billDAO.insertBill(bill);
    for (Map.Entry<Long, Item> list : cart.getCartItems().entrySet()) {
        billDetailDAO.insertBillDetail(new BillDetail(0, ID,
            list.getValue().getProduct().getProductID(),
            list.getValue().getProduct().getProductPrice(),
            list.getValue().getQuantity()));
    }
    cart = new Cart();
    session.setAttribute("cart", cart);
} catch (Exception e) {
}
response.sendRedirect("/shop/index.jsp");
}
}

```



```

28 String payment = request.getParameter("payment");
29 String address = request.getParameter("address");
30 HttpSession session = request.getSession();
31 Cart cart = (Cart) session.getAttribute("cart");
32 Users users = (Users) session.getAttribute("user");
33 try {
34     long ID = new Date().getTime();
35     Bill bill = new Bill();
36     bill.setBillID(ID);
37     bill.setAddress(address);
38     bill.setPayment(payment);
39     bill.setUserID(users.getUserID());
40     bill.setDate(new Timestamp(new Date().getTime()));
41     bill.setTotal(cart.total());
42     billDAO.insertBill(bill);
43     for (Map.Entry<Long, Item> list : cart.getCartItems().entrySet()) {
44         billDetailDAO.insertBillDetail(new BillDetail(0, ID,
45             list.getValue().getProduct().getProductID(),
46             list.getValue().getProduct().getProductPrice(),
47             list.getValue().getQuantity()));
48     }
49     cart = new Cart();
50     session.setAttribute("cart", cart);
51 } catch (Exception e) {
52 }
53 response.sendRedirect("/Shop/Index.jsp");
54 }
55 }
    
```

Cập nhật địa chỉ trang checkout trên header:

```

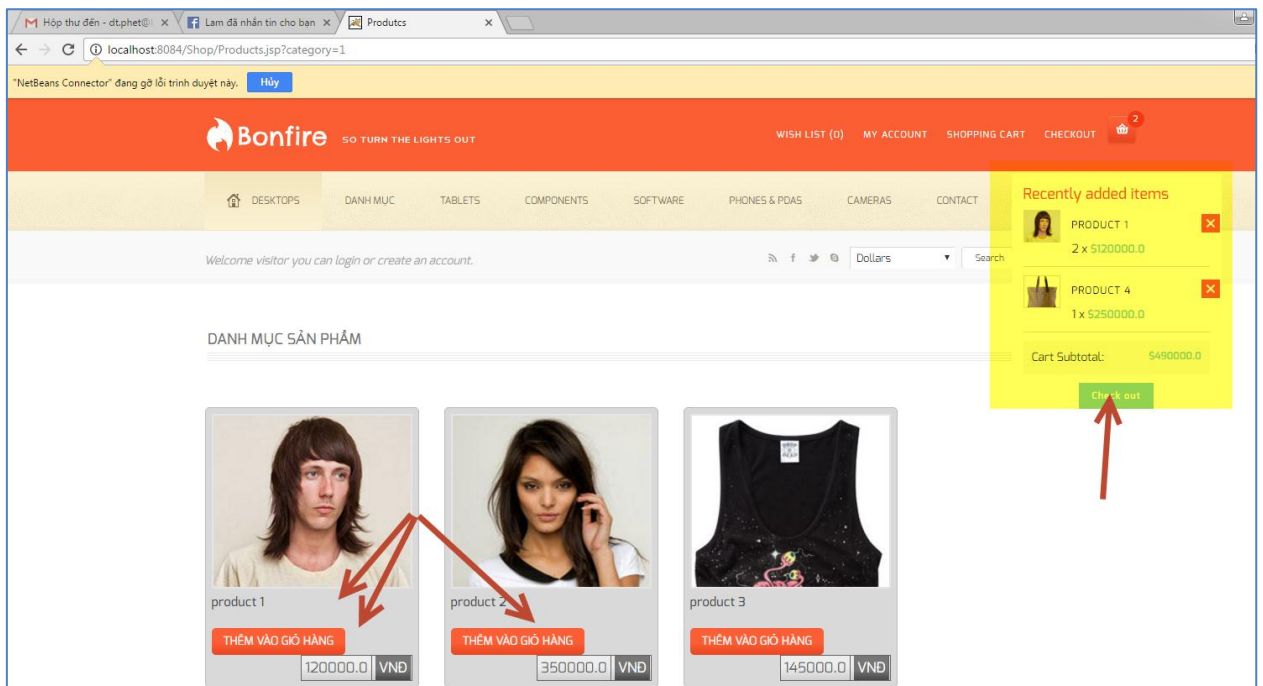
76 <div class="total_left">Cart Subtotal: </div>
77 <div class="total_right">${cart.total()}</div>
78 <div class="clearfix"> </div>
79 </div>
80 <div class="login_buttons">
81 <div class="check_button"><a href="checkout.jsp">Check out</a></div>
82 <div class="clearfix"></div>
83 </div>
84 <div class="clearfix"></div>
85 <ul>
    
```

Bổ sung kiểm tra: Khi click vào check out nếu chưa đăng nhập thì yêu cầu đăng nhập:
cập nhật code cho checkout.jsp:

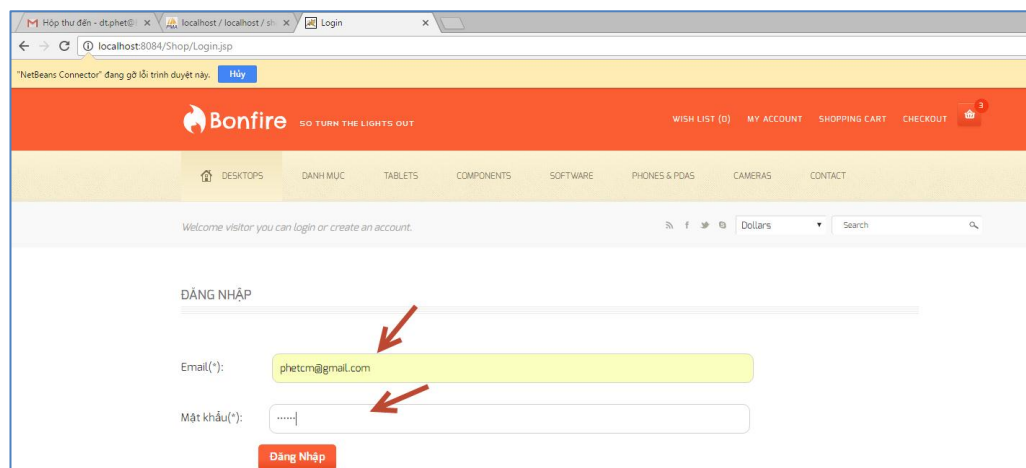
```

1 <%@page import="model.Users"%>
2 <%@page contentType="text/html" pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <...45 lines />
50 <body>
51 <%
52     Users users = (Users) session.getAttribute("user");
53     if (users == null) {
54         response.sendRedirect("/shop/login.jsp");
55     }
56 %>
57 <jsp:include page="header.jsp"></jsp:include>
58 <div class="container">
59
60
    
```

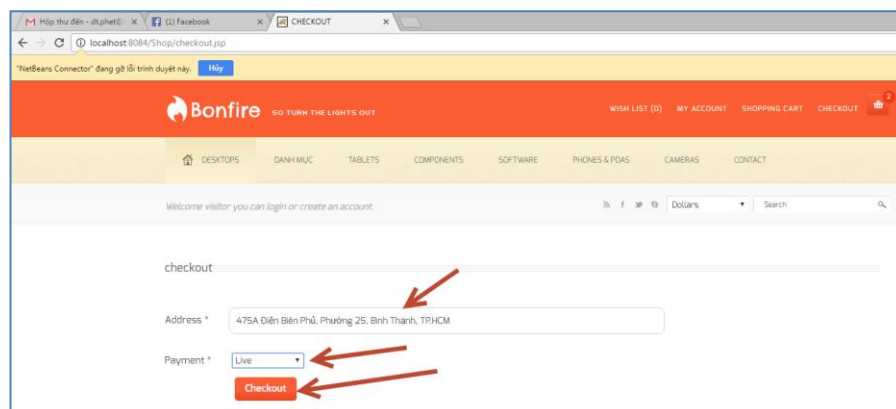
Thực thi: Chọn vài sản phẩm thêm vào giỏ hàng, click vào Check out



Nếu chưa đăng nhập sẽ chuyển sang trang Login.jsp:



Điền thông tin, click Check out:

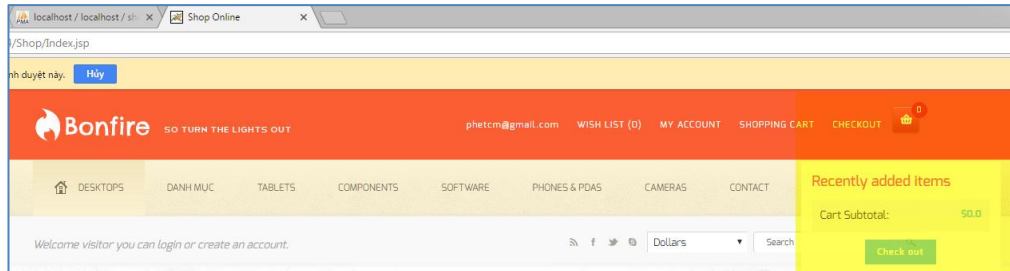


Kết quả : sẽ lưu vào CSDL bảng Bill và Bill_Detail như sau:

bill_id	user_id	total	payment	address	date
1463295350519	1463056081952	13	Live	123 - Q1 - TP.HCM	2016-05-15 13:55:50
0	0	0	s	s	2016-10-08 15:09:13
1475930298250	0	840000	Live	175A Ấlả»n BiẢ»n Phả»p.25, BẢ»nh ThẢ»nh, TP.HC...	2016-10-08 19:38:18

bill_detail_id	bill_id	product_id	price	quantity
4	1463295350519	1	2	1
5	1463295350519	2	3	1
6	1463295350519	3	4	2
7	0	0	0	0
8	1475930298250	1	120000	2
9	1475930298250	2	350000	1
10	1475930298250	4	250000	1

Giỏ hàng về trạng thái null:



-----Hết Lab 09-----