

Lab 10

Introduction to Model-View-Control

Mục tiêu

- Áp dụng mô hình MVC vào phát triển các ứng dụng web
- Hiểu rõ các thành phần trong MVC và áp dụng thành thạo MVC vào các ứng dụng web.

Phần I Bài tập step by step

Bài 1.1

Phát triển ứng dụng web sau theo mô hình MVC.

Tạo database cho ứng dụng theo các lệnh sql sau:

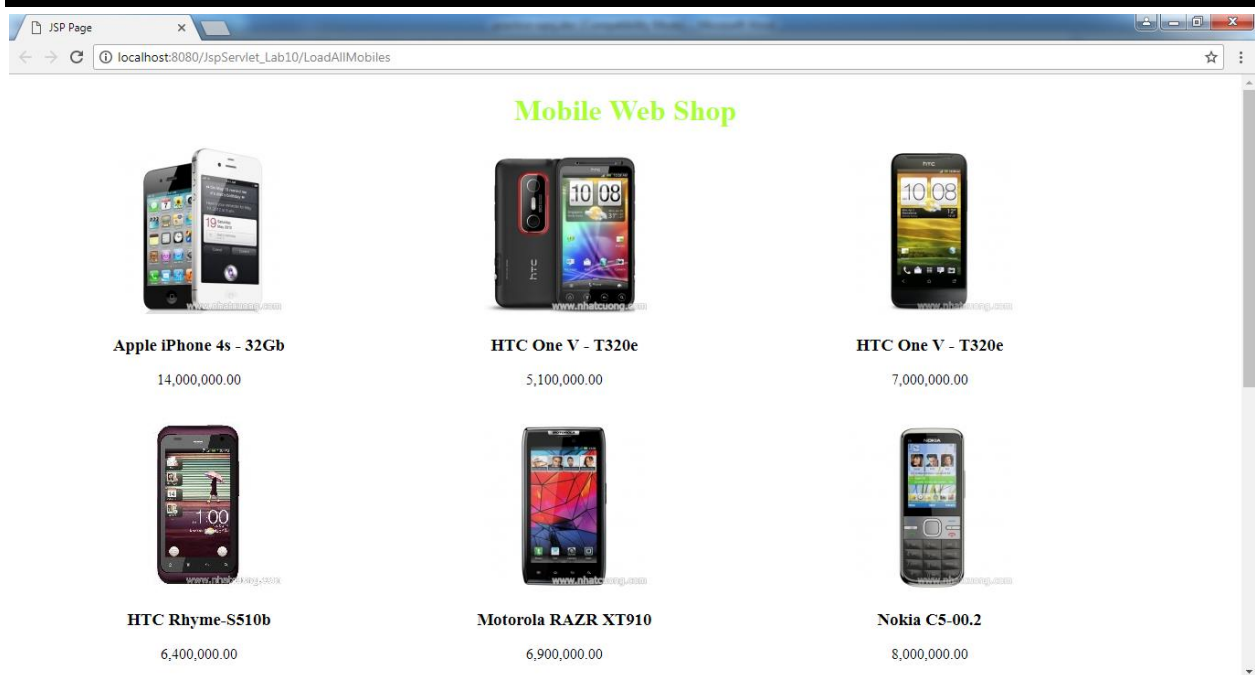
```
create database JspServlet_Lab10
go
use JspServlet_Lab10
go
create table tblMobile(mobileId int primary key identity, mobileName
nvarchar(100) not null, warranty nvarchar(200),
inOutStock bit, price float, accessories nvarchar(100), imgSrc
varchar(100))

insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'Apple iPhone 4s - 32Gb',N'03 tháng',1,1400000,N'Sạc, cáp, tai
nghe, hộp sách','image/w182h182_apple-iphone-4s-16gb-
1331181668.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'HTC One V - T320e',N'06 tháng',1,5100000,N'Pin, sạc, cáp, tai
nghe, hộp sách','image/w182h182_htc-evo-3d-1331180995.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'HTC One V - T320e',N'12 tháng',1,7000000,N'Sạc, cáp, tai
nghe, hộp sách','image/w182h182_htc-one-v-1334399457.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'HTC Rhyme-S510b',N'12 tháng',1,6400000,N'Pin, sạc, cáp, tai
nghe, hộp sách, thẻ 8GB đi kèm, Còn BH chính
```

```
hãng','image/wl82h182_htc-rhyme-s510b-1331113653.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'Motorola RAZR XT910',N'12 tháng',1,6900000,N'Pin, sạc, cáp,
tai nghe, sách hướng dẫn, hộp đựng','image/wl82h182_motorola-razr-
xt910-1331194891.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'Nokia C5-00.2',N'12 tháng',1,8000000,N'Sạc, cáp, tai nghe,
hộp sách','image/wl82h182_nokia-c5-00-2-1334289932.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'Nokia C5-06',N'12 tháng',0,3000000,N'Sạc, cáp, tai nghe, hộp
sách','image/wl82h182_nokia-c5-06-1334289945.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'Samsung Galaxy Note N7000',N'10 tháng',1,8900000,N'Pin, sạc,
cáp, tai nghe, hộp sách','image/wl82h182_samsung-galaxy-note-cty-
n7000-1331191686.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'Sony Xperia S-LT26i',N'6 tháng',1,9400000,N'Sạc, cáp, tai
nghe, hộp sách','image/wl82h182_sony-xperia-s-lt26i-1334226065.jpg')
insert into
tblMobile(mobileName,warranty,inOutStock,price,accessories,imgSrc)
values(N'Sony Xperia Sola - MT27i',N'6 tháng',0,5300000,N' Sạc, cáp,
tai nghe, hộp sách hướng dẫn','image/wl82h182_sony-xperia-sola-
1336725289.jpg')

select * from tblMobile
```

Trang **home.jsp** có nội dung như sau:



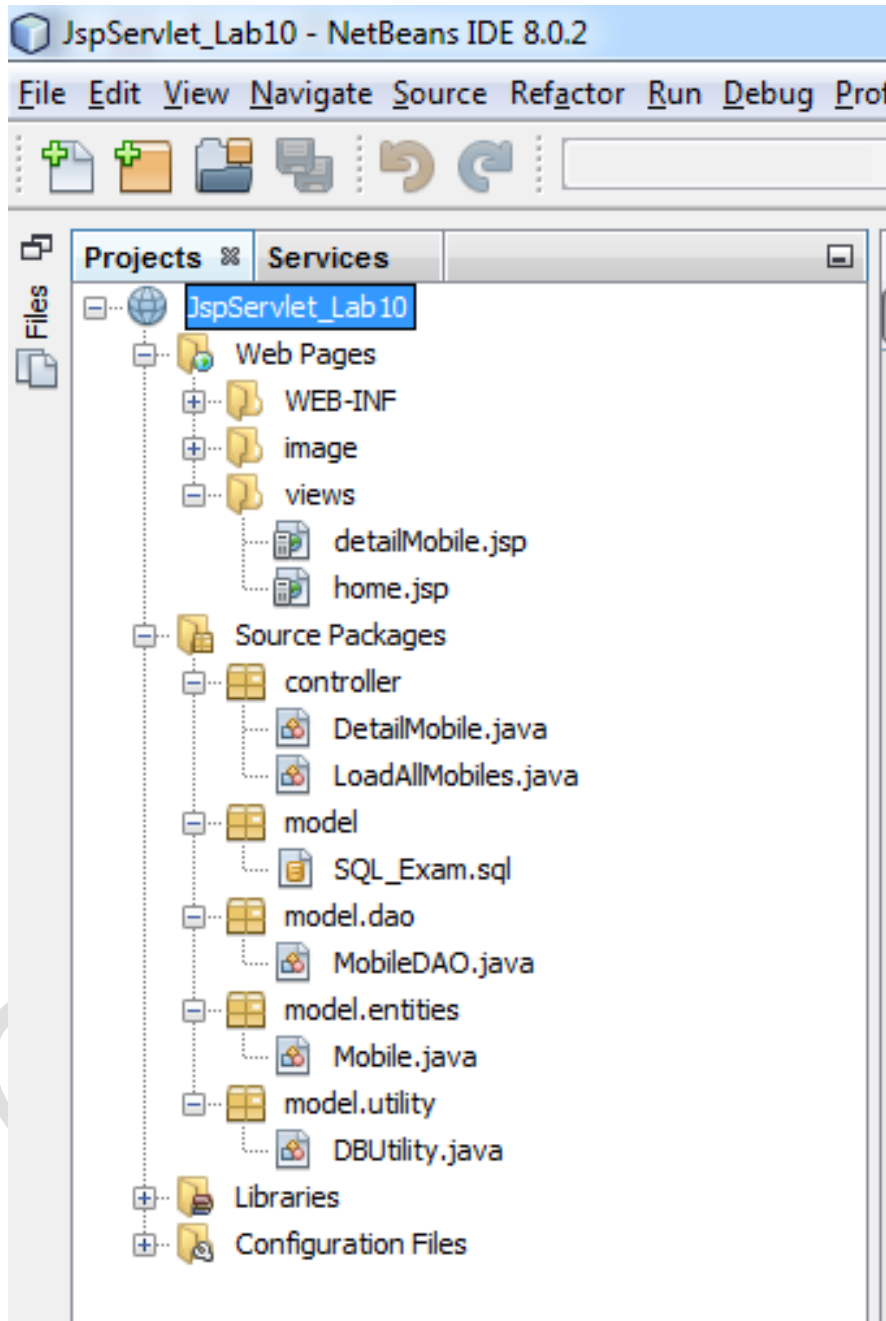
Khi Click vào hình ảnh hoặc tên của sản phẩm sẽ chuyển sang trang mobileDetail.jsp như sau:



Tạo liên kết để quay về trang home.jsp

STEP 1: Tạo project và add thư viện tương tác database

- New Project -> Java Web -> Web Application -> Đặt tên project "JspServlet_Lab10" -> Next -> Next -> Finish.
- Add gói thư viện sqljdbc4.jar vào Libraries.
- Cấu trúc của Project khi ứng dụng mô hình MVC sẽ như sau:



STEP 2: Tạo class đóng/mở kết nối, class entity tương ứng với bảng, class DAO có các hàm tương tác database (Thành phần Model).

Khi tạo database chúng ta có các câu lệnh sql trong file .sql hoặc database được backup thành file .bak, chúng ta sẽ copy các file đó vào trong thư mục model của ứng dụng.

Tạo class đóng/mở kết nối: Right click Source Packages -> New -> Java Class -> Đặt tên "DBUtility", package "model.utility".

Viết hàm đóng mở kết nối:

```
package model.utility;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

/**
 *
 * @author HAITHANH
 */
public class DBUtility {
    public static Connection openConnection(){
        Connection con = null;
        try {
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
            con =
DriverManager.getConnection("jdbc:sqlserver://localhost:1433;databaseName=JspServlet_Lab10","
sa","1234$");
        } catch (ClassNotFoundException ex) {
            Logger.getLogger(DBUtility.class.getName()).log(Level.SEVERE, null, ex);
        } catch (SQLException ex) {
            Logger.getLogger(DBUtility.class.getName()).log(Level.SEVERE, null, ex);
        }
        return con;
    }

    public static void closeAll(Connection con, PreparedStatement pstmt, ResultSet rs){
        if(con!=null)
            try {
                con.close();
            } catch (SQLException ex) {
                Logger.getLogger(DBUtility.class.getName()).log(Level.SEVERE, null, ex);
            }
        if(pstmt!=null)
            try {
                pstmt.close();
            } catch (SQLException ex) {
                Logger.getLogger(DBUtility.class.getName()).log(Level.SEVERE, null, ex);
            }
    }
}
```

```
        if(rs!=null)
            try {
                rs.close();
            } catch (SQLException ex) {
                Logger.getLogger(DBUtility.class.getName()).log(Level.SEVERE, null, ex);
            }
        }
    }
}
```

Tạo class Mobile tương ứng các cột của bảng: Right click Source Packages -> New -> Java Class.. -> Đặt tên file “Mobile”, package “model.entities”

```
package model.entities;

/**
 *
 * @author HAITHANH
 */
public class Mobile {
    private int mobileId;
    private String mobileName;
    private String warranty;
    private boolean inOutStock;
    private float price;
    private String accessories;
    private String imgSrc;

    public Mobile() {
    }

    public Mobile(int mobileId, String mobileName, String warranty, boolean inOutStock,
float price, String accessories, String imgSrc) {
        this.mobileId = mobileId;
        this.mobileName = mobileName;
        this.warranty = warranty;
        this.inOutStock = inOutStock;
        this.price = price;
        this.accessories = accessories;
        this.imgSrc = imgSrc;
    }

    public int getMobileId() {
        return mobileId;
    }

    public void setMobileId(int mobileId) {
        this.mobileId = mobileId;
    }
}
```

```
}

public String getMobileName() {
    return mobileName;
}

public void setMobileName(String mobileName) {
    this.mobileName = mobileName;
}

public String getWarranty() {
    return warranty;
}

public void setWarranty(String warranty) {
    this.warranty = warranty;
}

public boolean isInOutStock() {
    return inOutStock;
}

public void setInOutStock(boolean inOutStock) {
    this.inOutStock = inOutStock;
}

public float getPrice() {
    return price;
}

public void setPrice(float price) {
    this.price = price;
}

public String getAccessories() {
    return accessories;
}

public void setAccessories(String accessories) {
    this.accessories = accessories;
}

public String getImgSrc() {
```

```
        return imgSrc;
    }

    public void setImgSrc(String imgSrc) {
        this.imgSrc = imgSrc;
    }
}
```

- Tạo class MobileDAO để cài đặt các hàm tương tác database: Right click Source Packages -> New -> Java Class -> Đặt tên file "MobileDAO", package "model.dao":

```
package model.dao;

import model.entities.Mobile;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import model.utility.DBUtility;

/**
 *
 * @author HAITHANH
 */
public class MobileDAO {

    //Hàm lấy hết thông tin của bảng tblMobile

    public List<Mobile> getAllMobiles() {
        List<Mobile> list = new ArrayList<>();

        Connection con;
        PreparedStatement pstmt = null;
        ResultSet rs = null;

        con = DBUtility.openConnection();
        try {
            pstmt = con.prepareStatement("select * from tblMobile");
            rs = pstmt.executeQuery();
            while (rs.next()) {
```



```
        Mobile m = new Mobile();
        m.setMobileId(rs.getInt("mobileId"));
        m.setMobileName(rs.getString("mobileName"));
        m.setWarranty(rs.getString("warranty"));
        m.setInOutStock(rs.getBoolean("inOutStock"));
        m.setPrice(rs.getFloat("price"));
        m.setAccessories(rs.getString("accessories"));
        m.setImgSrc(rs.getString("imgSrc"));
        list.add(m);
    }
} catch (SQLException ex) {
    Logger.getLogger(MobileDAO.class.getName()).log(Level.SEVERE, null, ex);
} finally {
    DBUtility.closeAll(con, pstmt, rs);
}
return list;
}
```

STEP 3: Tạo Servlet để thực hiện chức năng Load All (Thành phần Controller)

- Right click Source Packages -> New -> Servlet -> Đặt tên file "LoadAllMobiles", package "controller" -> Next -> Finish.
- Viết lệnh xử lý trong hàm "processRequest()":

```
protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");

    List<Mobile> allMobiles = new MobileDAO().getAllMobiles();
    request.setAttribute("listM", allMobiles);
    request.getRequestDispatcher("views/home.jsp").forward(request, response);
}
```

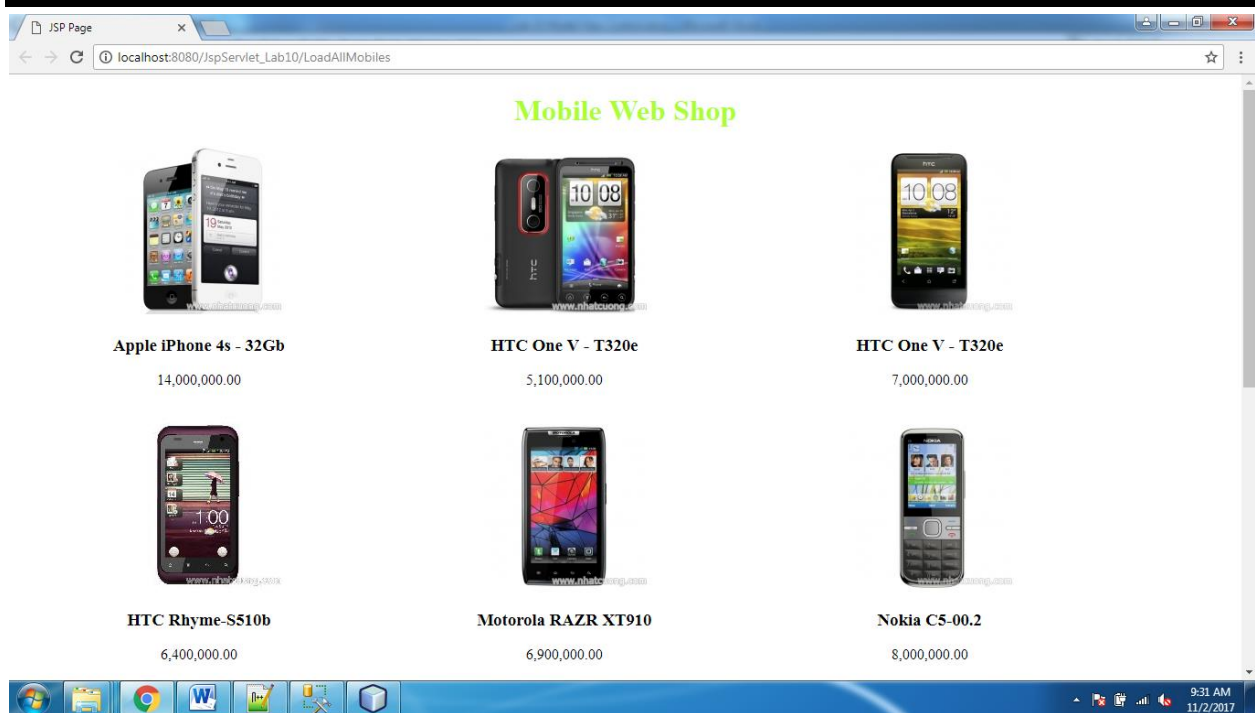
STEP 4: Tạo trang home.jsp, sử dụng thẻ để trình bày nội dung như yêu cầu, cấu hình để servlet LoadAllMobiles chạy trước và chạy thử ứng dụng (Thành phần View)

- Tạo trang home.jsp: Right click Web Pages -> New -> JSP -> Đặt tên file "home", folder "views" -> Finish.
- Sử dụng thẻ <div> để hiển thị thông tin của mỗi mobile:

```
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
<!-- Định nghĩa vùng nội dung hiển thị cho mỗi Mobile -->
<style>
.mobile{
width: 30%;
height: 300px;
float: left;
}
</style>
</head>
<body>
<center>
<h1 style="color: greenyellow">Mobile Web Shop</h1>
<c:forEach items="${listM}" var="m">
<div class="mobile">

<h3>${m.mobileName}</h3>
<fmt:formatNumber currencySymbol="" type="currency" value="${m.price}"/>
</div>
</c:forEach>
</center>
</body>
</html>
```

- Copy thư mục "image" chứa các ảnh của mobile vào Web Pages.
- Cấu hình để servlet LoadAllMobiles chạy trước: Right click project -> Properties -> Run -> Gõ vào Relative URL: /loadAllMobiles
- Deploy và run project.



STEP 5: Tạo chức năng detail mobile

- Trong trang home.jsp tạo liên kết từ ảnh và tên của mobile

```
<a href="DetailMobile?mId=${m.mobileId}">
  
  <h3>${m.mobileName}</h3>
</a>
```

- Tạo Servlet **DetailMobile** trong package **controller**
- Viết hàm trong **MobileDAO** để lấy về thông tin mobile theo mobileId

//Hàm lấy thông tin của Mobile theo mobileId

```
public Mobile getMobileByMobileId(int modId) {
    Mobile m = null;

    Connection con;
    PreparedStatement pstmt = null;
    ResultSet rs = null;

    con = DBUtility.openConnection();
    try {
        pstmt = con.prepareStatement("select * from tblMobile where mobileId=?");
        pstmt.setInt(1, modId);
        rs = pstmt.executeQuery();
        if (rs.next()) {
```

```

        m = new Mobile();
        m.setMobileId(rs.getInt("mobileId"));
        m.setMobileName(rs.getString("mobileName"));
        m.setWarranty(rs.getString("warranty"));
        m.setInOutStock(rs.getBoolean("inOutStock"));
        m.setPrice(rs.getFloat("price"));
        m.setAccessories(rs.getString("accessories"));
        m.setImgSrc(rs.getString("imgSrc"));
    }
} catch (SQLException ex) {
    Logger.getLogger(MobileDAO.class.getName()).log(Level.SEVERE, null, ex);
} finally {
    DBUtility.closeAll(con, pstmt, rs);
}
return m;
}

```

- Quay lại servlet **DetailMobile** viết lệnh xử lý trong hàm **processRequest()**:

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    int modId = Integer.parseInt(request.getParameter("mId"));
    Mobile mobile = new MobileDAO().getMobileByMobileId(modId);
    request.setAttribute("mobile", mobile);
    request.getRequestDispatcher("views/detailMobile.jsp").forward(request, response);
}

```

STEP 6: Tạo trang detailMobile.jsp trong thư mục views

- Right click views -> New -> JSP -> Đặt tên file "detailMobile" -> Finish

```

<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<center>
<h1 style="color: greenyellow">Mobile Web Shop</h1>
<table>
<tr>
<td>

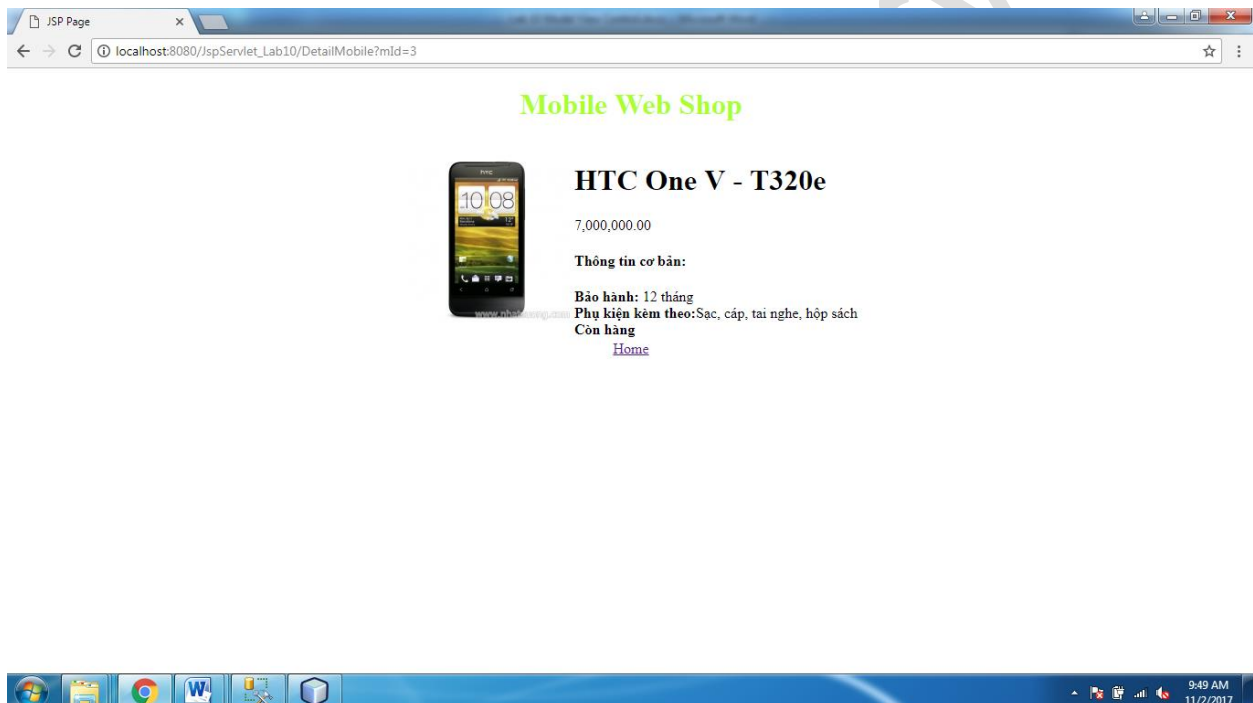
```

```

<td style="vertical-align: top">
  <h1>${m.mobileName}</h1>
  <fmt:formatNumber currencySymbol="" type="currency" value="${m.price}"/><br/>
  <h4>Thông tin cơ bản:</h4>
  <b>Bảo hành:</b> ${m.warranty}<br/>
  <b>Phụ kiện kèm theo:</b> ${m.accessories}<br/>
  <b>${m.inOutStock?"Còn hàng":"Hết hàng"}</b>
</td>
</tr>
</table>
<a href="LoadAllMobiles">Home</a>
</center>
</body>
</html>

```

- Chạy lại project chúng ta được kết quả:



PHẦN 2: BÀI TẬP TỰ LÀM

Áp dụng mô hình MVC vào thực hiện ứng dụng web bất kỳ.

HẾT