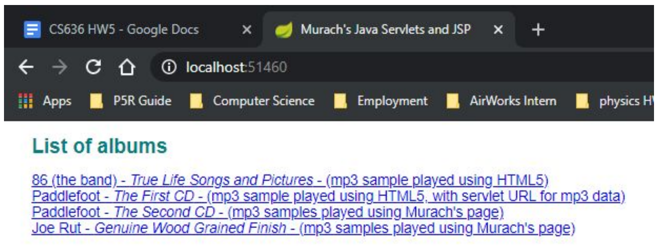
Run Lin
CS 636
HW5

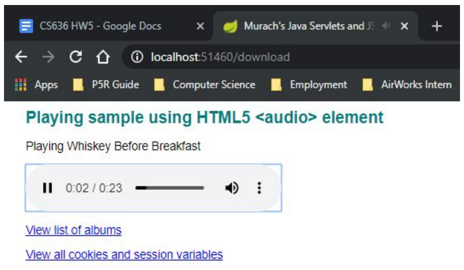
1.

a. No issue in running ch07downloadS



List of albums

86 (the band) - *True Life Songs and Pictures* - (mp3 sample played using HTML5)
Paddlefoot - *The First CD* - (mp3 sample played using HTML5, with servlet URL for mp3 data)
Paddlefoot - *The Second CD* - (mp3 samples played using Murach's page)
Joe Rut - *Genuine Wood Grained Finish* - (mp3 samples played using Murach's page)



Playing sample using HTML5 <audio> element

Playing Whiskey Before Breakfast

View list of albums
View all cookies and session variables

b. The session variables are set in DownloadServlet.java's showProduct() and registerUser() methods.

```
private String showProduct(HttpServletRequest request,
            HttpServletResponse response) {
        String productCode = request.getParameter("productCode");
        HttpSession session = request.getSession();
        String url = null;
        try {
            if (checkUser(request)) {
                // user is logged in, can see product page
                url = determineProductView(request, productCode);
            } else { // need to log in
                // remember chosen productCode for later
                session.setAttribute("productCode", productCode);
                url = "/register.jsp";
            }
        } catch (IOException e) {
            request.setAttribute("error", "error accessing user: " + e);
            url = "/error.jsp";
        }
        return url;
    }
```

```java
private String registerUser(HttpServletRequest request,
            HttpServletResponse response) {
      ...
        // store the User object as a session attribute
        HttpSession session = request.getSession();
        session.setAttribute("user", user);
      ...
    }
```

And are used in these methods below in DownloadServlet.java:

```java
private String logout(HttpServletRequest request,
            HttpServletResponse response) {
        request.getSession().invalidate();  // drop session
        return "/logout.jsp";
    }
private boolean checkUser(HttpServletRequest request) throws IOException {
     HttpSession session = request.getSession();
        User user = (User) session.getAttribute("user");
        return (user != null);
    }
```

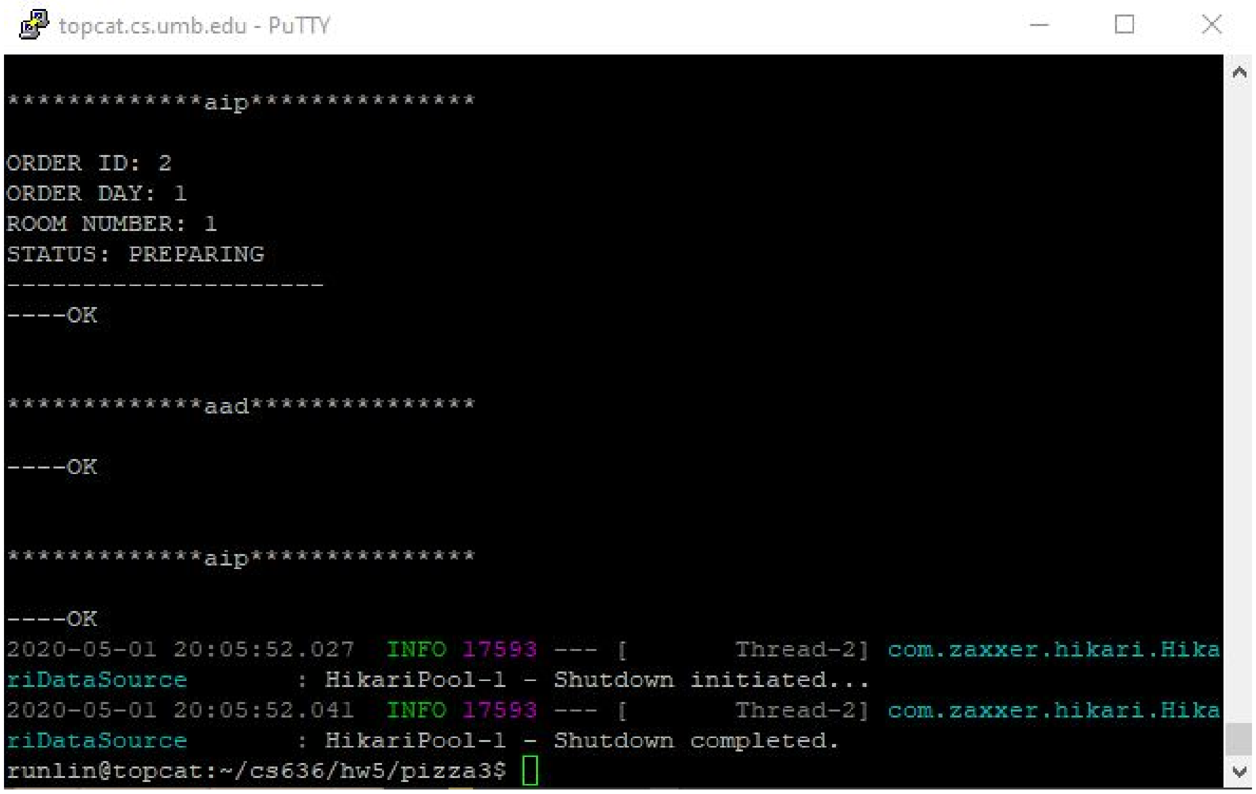
Session variables are seen throughout the whole program session until the program ends, but request variables are garbage collected when the request is fulfilled, so session variables are longer lived, so it’s better if we use session variables.

c. registerUser()   uses request.setAttribute for errors: goes to error.jsp.
doGet()   uses request.setAttribute for errors: goes to error.jsp.
showProduct()   uses request.setAttribute for errors: goes to error.jsp.
determineProductView() uses request.setAttribute for title and src: goes to universal.jsp.
determineProductView() also uses it for productCode: goes to _download.jsp.

d.

2.

a. Running command: "runJarByProfile.sh oracle SystemTest", success!



```
*************aip***************

ORDER ID: 2
ORDER DAY: 1
ROOM NUMBER: 1
STATUS: PREPARING
---------------------
----OK


*************aad***************

----OK


*************aip***************

----OK
2020-05-01 20:05:52.027  INFO 17593 --- [       Thread-2] com.zaxxer.hikari.Hika
riDataSource        : HikariPool-1 - Shutdown initiated...
2020-05-01 20:05:52.041  INFO 17593 --- [       Thread-2] com.zaxxer.hikari.Hika
riDataSource        : HikariPool-1 - Shutdown completed.
runlin@topcat:~/cs636/hw5/pizza3$
```

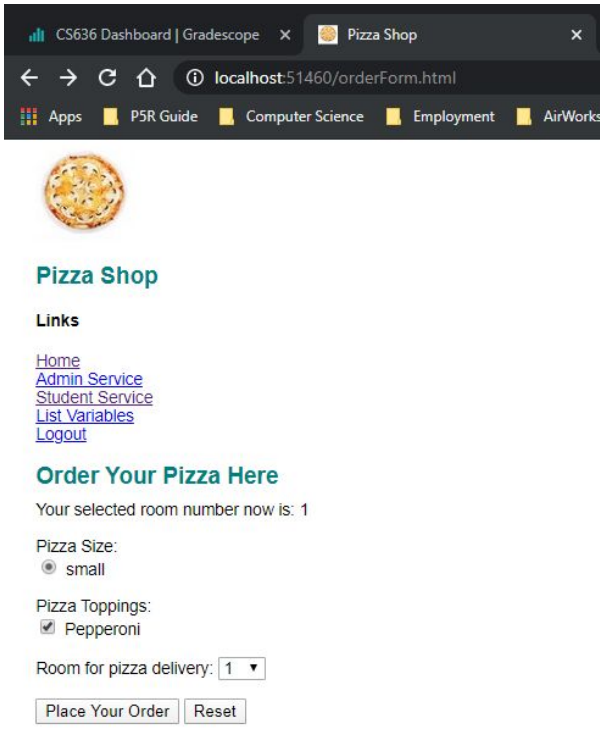
b. First ran into a problem with port 9002, but after changing application.properties to my own port, success!

Running command: "runJarByProfile.sh oracle web", success!

Ordering a Pizza:

- Goto "localhost:51460/welcome.html"
- Click "Student Service"
- Click "Order a pizza now!"
- Choose desired size "small" or topping "Pepperoni"
- Choose "Room for pizza delivery"
- Click "Place Your Order"

URL: http://localhost:51460/studentWelcome.html



Pizza Shop

Links

Home
Admin Service
Student Service
List Variables
Logout

Order Your Pizza Here

Your selected room number now is: 1

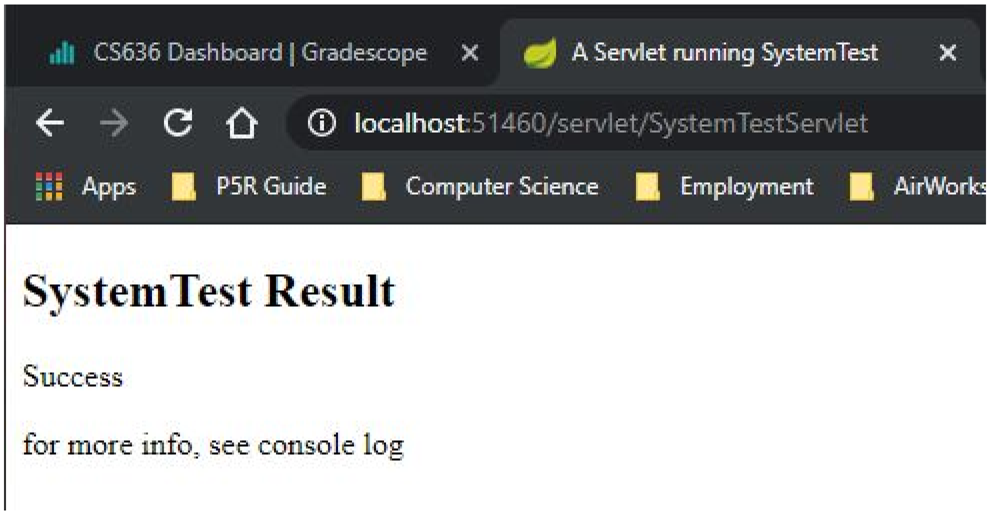
Pizza Size:
● small

Pizza Toppings:
✓ Pepperoni

Room for pizza delivery: 1

Place Your Order | Reset

c. URL: localhost:51460/servlet/SystemTestServlet, success!


CS636 Dashboard | Gradescope
A Servlet running SystemTest
localhost:51460/servlet/SystemTestServlet
Apps
P5R Guide
Computer Science
Employment
AirWorks

SystemTest Result

Success

for more info, see console log

3.

- Pizza3 uses Spring Boot, and some similarity with pizza2 using Object-relational Mapping:
  import org.springframework.beans.factory.annotation.Autowired;
  import org.springframework.stereotype.Service;
- In AdminDAO.java:
  import org.springframework.stereotype.Repository;
  Using @repository for the class AdminDAO
  Connection to service the sql instead of another DAO object for connection.
- In DbDAO.java new changes:
  import org.springframework.beans.factory.annotation.Autowired;
  import org.springframework.stereotype.Repository;
  import javax.sql.DataSource;
  @Repository for class DbDAO.
  Connection connection = dataSource.getConnection();
- Similar uses for Connection connection in MenuDAO.java and PizzaOrderDAO.java

In all the above, transactions are also implemented, for connections, it is used for spring to manage our bean, the DAO objects, where pizza1 needs to create new DAO() objects to service the connection. Transactions are used to make sure the execution of the an sql string is safe and contains no errors, else we need to rollback.

4. The session variable “student” is set in StudentController.java:

```
StudentBean student = (StudentBean) request.getSession().getAttribute("student");
        if (student == null)
            student = new StudentBean();
        if (roomNo != null)
            student.setRoomNo(roomNo); // set newly obtained roomNo
        if (student.getRoomNo() > 0)
            roomNo = student.getRoomNo(); // just set or older setting
 here → request.getSession().setAttribute("student", student);
```

It’s inside of studentWelcome.html method displayWelcome():

```
@RequestMapping("studentWelcome.html")
    public String displayWelcome(Model model, @RequestParam(value = "room",
required = false) String chosenRoom,
            HttpServletRequest request) throws ServletException {
        // At studentWelcome, the user gets a StudentBean.
        // If it's not there for subsequent pages, hand the request to
        // studentWelcome. Having the bean is like being "logged in".
        // boolean hasBean = (session.getAttribute("student") != null);
        Integer roomNo = null;
      ...
      }
```

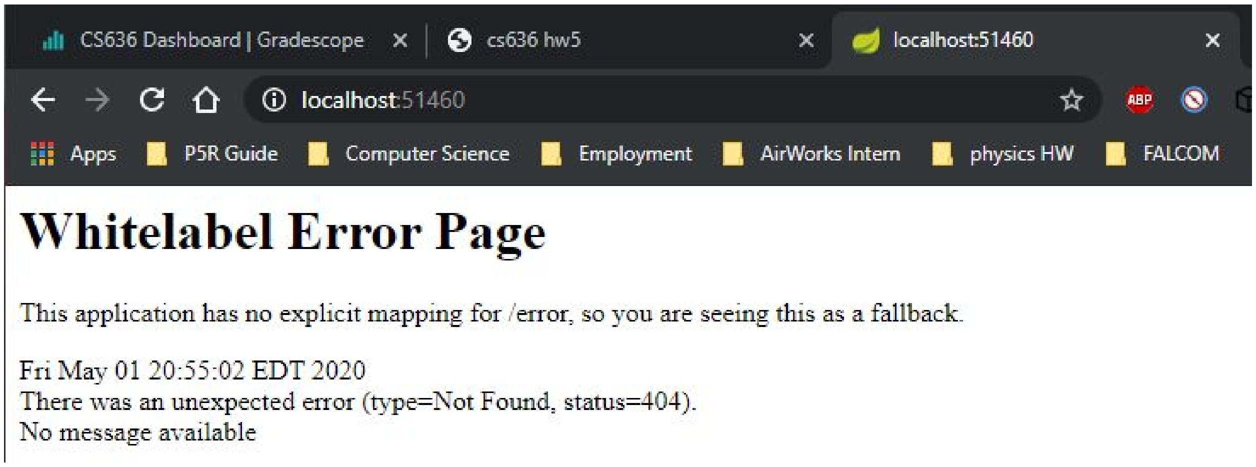
Initially the roomNo is set to null in displayWelcome, but then when the student order a pizza, the “student” session variable’s room number is updated in orderPizza() method:

```
@RequestMapping("orderPizza.html")
    public String orderPizza(Model model, @RequestParam(value = "size", required
= false) String chosenSize,
            @RequestParam(value = "room", required = false) String chosenRoom,
            @RequestParam(required = false) Set<String> toppings,
HttpServletRequest request) throws ServletException {
StudentBean student = (StudentBean) request.getSession().getAttribute("student");
    if (student == null)
        return "forward:studentWelcome.html"; // get started right
    Integer roomNo = student.getRoomNo();
    if (chosenRoom != null) {
      roomNo = Integer.parseInt(chosenRoom);
      student.setRoomNo(roomNo);
    }
}
```

5.

a.

Initially it fails:



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Fri May 01 20:55:02 EDT 2020
There was an unexpected error (type=Not Found, status=404).
No message available

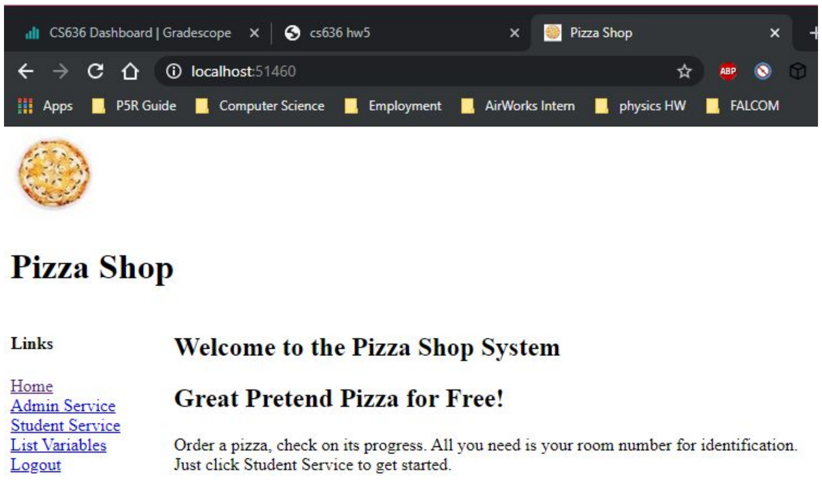
Then I edited the code in StudenController.java:

```
@RequestMapping("welcome.html")
    public String welcome(Model model) {
        return "welcome";
    }

@RequestMapping("index.html")
    public String index(Model model) {
        return "welcome";
    }

@RequestMapping("")
    public String empty(Model model) {
        return "welcome";
    }
```

Success for “localhost:51460” !



Pizza Shop

Links

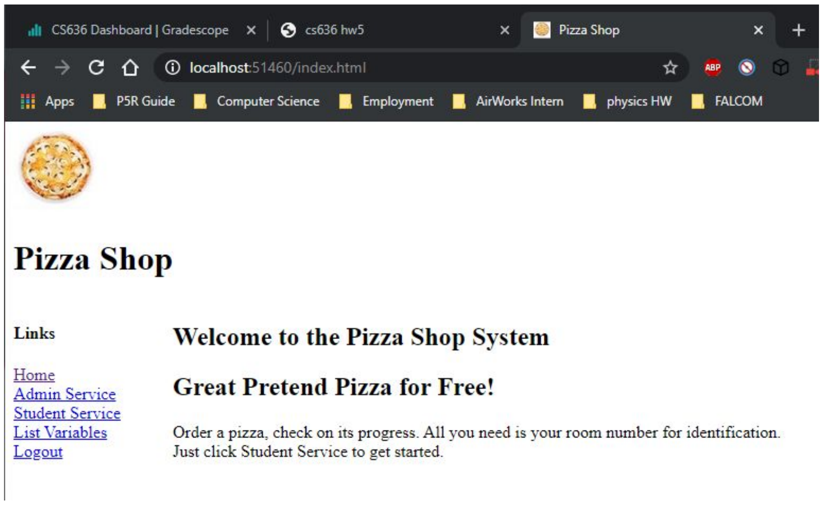
Home
Admin Service
Student Service
List Variables
Logout

Welcome to the Pizza Shop System

Great Pretend Pizza for Free!

Order a pizza, check on its progress. All you need is your room number for identification. Just click Student Service to get started.

Success for “localhost:51460/index.html” !



Pizza Shop

Links

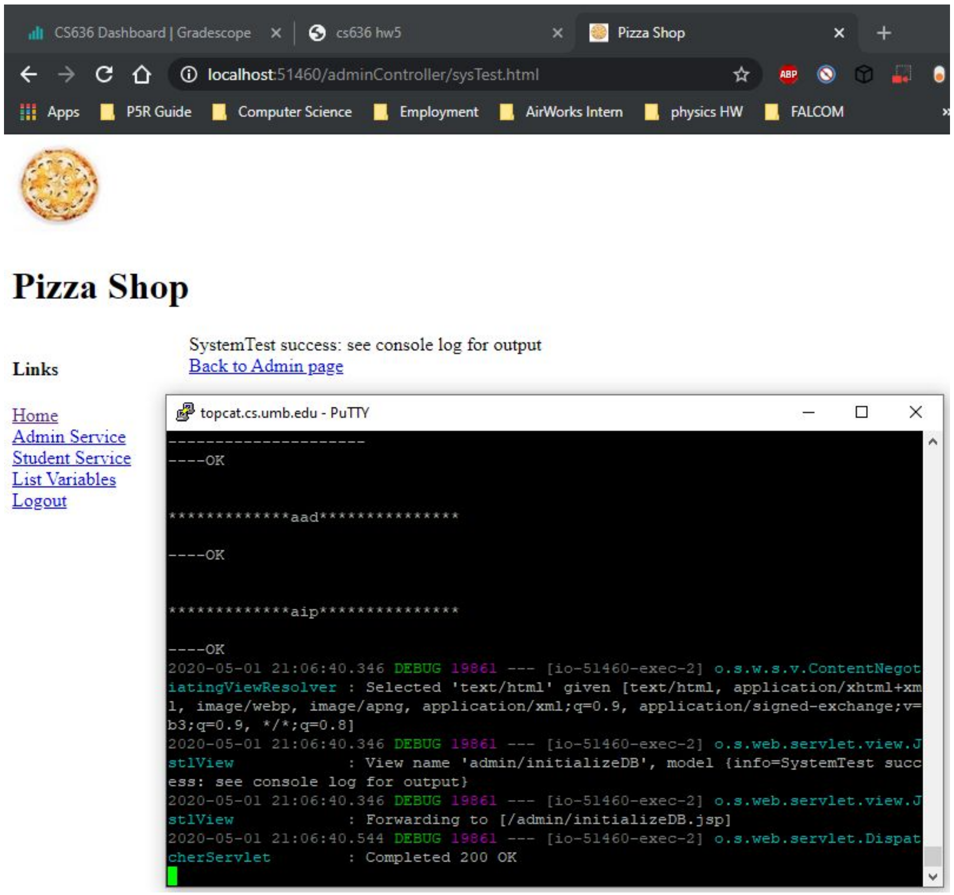
Home
Admin Service
Student Service
List Variables
Logout

Welcome to the Pizza Shop System

Great Pretend Pizza for Free!

Order a pizza, check on its progress. All you need is your room number for identification. Just click Student Service to get started.

b. Goto: “http://localhost:51460/adminController/sysTest.html”, success!



Pizza Shop

Links

Home
Admin Service
Student Service
List Variables
Logout

SystemTest success: see console log for output
Back to Admin page

topcat.cs.umb.edu - PuTTY

```
---------------------
----OK


*************aad***************

----OK


*************aip***************

----OK
2020-05-01 21:06:40.346 DEBUG 19861 --- [io-51460-exec-2] o.s.w.s.v.ContentNegot
iatingViewResolver : Selected 'text/html' given [text/html, application/xhtml+xm
l, image/webp, image/apng, application/xml;q=0.9, application/signed-exchange;v=
b3;q=0.9, */*;q=0.8]
2020-05-01 21:06:40.346 DEBUG 19861 --- [io-51460-exec-2] o.s.web.servlet.view.J
stlView            : View name 'admin/initializeDB', model {info=SystemTest succ
ess: see console log for output}
2020-05-01 21:06:40.346 DEBUG 19861 --- [io-51460-exec-2] o.s.web.servlet.view.J
stlView            : Forwarding to [/admin/initializeDB.jsp]
2020-05-01 21:06:40.544 DEBUG 19861 --- [io-51460-exec-2] o.s.web.servlet.Dispat
cherServlet        : Completed 200 OK
```

And after looking at initializeDB.jsp, it’s only a href link to adminWelcome.html, linking me back to the admin page, a very misleading name for this purpose.

c. In StudentController.java:
    - The comments tells us what forward does, if either size or topping is not available, forward back to /orderForm.html to redo the form
    - If chosenToppings are also not selected, redo the form as well from /orderForm.html
    - However if none of those are problems, proceed to forward to studentWelcome.html to start correctly.