

LoginController.java

```
package com.cts.cit.controller;
import java.util.Date;
import javax.validation.Valid;
import javax.websocket.server.PathParam;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.i18n.LocaleContextHolder;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.ui.ModelMap;
import org.springframework.validation.BindingResult;
import org.springframework.validation.Validator;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;
import com.cts.cit.exception.ApplicationException;
import com.cts.cit.model.CCRepresentative;
import com.cts.cit.model.Customer;
import com.cts.cit.model.IssueDetail;
import com.cts.cit.service.CustomerIssueService;

@Controller
public class LoginController
{
    @Autowired
    CustomerIssueService custService;
    @Autowired
    public LoginController(CustomerIssueService service, Validator validator) {
        System.out.println("in parametrized constr of controller with validator");
        this.custService = service;
        this.validator = validator;
    }
    @Autowired
    private Validator validator;
    @RequestMapping(value = "/login1", method = RequestMethod.GET)
    public String showLoginPage(@ModelAttribute("customer") Customer customer, ModelMap map) {
        // IssueDetail issueDetail = new IssueDetail();
        // Type Your Code here
        //map.addAttribute("customer",new Customer());
        return "login";
    }
    @RequestMapping(value = "/login", method = RequestMethod.POST)
    public String showCustomerPage(@ModelAttribute("customer") Customer customer, ModelMap map,
        BindingResult result) throws ApplicationException{
        // Type Your Code here
        validator.validate(customer, result);
        if (result.hasErrors()) {
            return "login";
        } else {
            custService.addCustomerObject(customer);
            return "customerform";
        }
    }
    @GetMapping("/customerform")
    public String showCustomerForm() {
        return "customerform";
    }
    @RequestMapping(value = "/customerform", method = RequestMethod.POST)
    public ModelAndView showCustomerIssue(ModelMap model, @RequestParam String issueid, @RequestParam String option) {
        IssueDetail issueDetail = new IssueDetail();
        ModelAndView modelAndView = new ModelAndView();
        // Type Your Code here
        if(option.equals("ri")){
            int issueId= Integer.parseInt(custService.getMaxissueId().substring(1));
            System.out.println(issueId);
            issueId= issueId+1;
            String currentIssueId = "I"+issueId;
        }
    }
}
```

```

        model.addAttribute("custIssued",currentIssued );
        modelAndView.setViewName("raiseissue");
    }else if(option.equals("vi")) {
        modelAndView.setViewName("customerissue");
        IssueDetail issueobj = custService.getIssueDetailById(issueid);
        System.out.println(issueid);
        System.out.println(issueobj);
        if(issueobj!=null) {
            modelAndView.addObject("issueobj", issueobj);
        }else {
            throw new ApplicationException("No records found with that id");
        }
    }

    return modelAndView;
}

@RequestMapping(value = "/raiseissue", method = RequestMethod.POST)
public ModelAndView raiseIssue(ModelMap model, @RequestParam String issueReportDate, @RequestParam String custId,
    @RequestParam String category, @RequestParam String descr) {
    CCRRepresentative ccr = new CCRRepresentative();
    IssueDetail issueDetail = new IssueDetail();
    CCRRepresentative ccrToModify = new CCRRepresentative();
    ModelAndView modelAndView = new ModelAndView("issueraised");
    // Type Your Code here
    if(issueReportDate.trim().isEmpty() || custId.trim().isEmpty() || category.trim().isEmpty() || descr.trim().isEmpty() )
    {
        throw new ApplicationException("Please Enter All the Required Details");
    }
    // check if the entered date is valid or not
    if (!custService.validateDateOfReport(issueReportDate))
    {
        // if Date is not valid then we will raise a Exception
        throw new ApplicationException("Please Enter Valid Date(Format:yyyy-mm-dd)");
    }
    else
    {
        // setting up issueDetail object with values received from the form
        issueDetail.setIssueReportDate(CustomerIssueService.stringToDateConverter(issueReportDate));
        issueDetail.setCustId(custId);
        issueDetail.setCategory(category);
        issueDetail.setDescription(descr);
        // finding available CCR
        CCRRepresentative availableCCR = custService.getAvailableCCR();
        //We need to Increment the availableCCR's issueCount
        //Because we are assigning him a Issue to resolve
        availableCCR.setIssuesInProgress(availableCCR.getIssuesInProgress()+1);
        // setting the availableCCR's Id to issueDetail Object
        issueDetail.setCcRepId(availableCCR.getCcRepId());
        // Since we are not getting custIssued from the form we need to generate it again
        int issued = Integer.parseInt(custService.getMaxIssued().substring(1)); //return 9
        issued = issued + 1;// 10
        String currentIssued = "I"+ issued; // it will be I10
        issueDetail.setCustIssued(currentIssued);
        //also add issue id to availableCCR issueds
        //I1;I2;I3;I4
        availableCCR.setIssueds( availableCCR.getIssueds()+","+currentIssued);
        issueDetail.setIssueStatus("InProgress"); // setting the default status as InProgress
        // adding issue detail to our issuesList
        CustomerIssueService.issuesList.add(issueDetail);
        // adding that issueDetail to model
        model.addAttribute(issueDetail);
        modelAndView.addObject("issue", issueDetail);
        return modelAndView;
    }
}
}
}

```

```

ApplicationException.java
package com.cts.cit.exception;
public class ApplicationException extends RuntimeException {
    private static final long serialVersionUID = 1L;

```

```

    public ApplicationException() {
        super();
    }
    public ApplicationException(final String message) {
        super(message);
    }
}

```

ExceptionHandlerControllerAdvice.java

```

package com.cts.cit.exception;
import javax.servlet.http.HttpServletRequest;
import org.springframework.http.HttpStatus;
//import org.springframework.http.HttpStatus;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.ControllerAdvice;
import org.springframework.web.bind.annotation.ExceptionHandler;
import org.springframework.web.bind.annotation.ResponseStatus;
//import org.springframework.web.bind.annotation.ExceptionHandler;
//import org.springframework.web.bind.annotation.ResponseStatus;
import org.springframework.web.servlet.ModelAndView;
@ControllerAdvice
public class ExceptionHandlerControllerAdvice {
    @ExceptionHandler(ApplicationException.class)
    public String handleException(ApplicationException error, Model model) {
        model.addAttribute("msg", error.getMessage());
        return "error1";
    }
}

```

CCRepresentative.java

```

package com.cts.cit.model;
import org.springframework.stereotype.Component;

@Component
//CustomerCareRepresentative
public class CCRepresentative {
    String ccRId;
    String ccrName;
    String ccrAddress;
    String ccrPhone;
    String ccrEMail;
    int issuesInProgress;
    String issueIds;
    public CCRepresentative() {
        super();
    }
    public CCRepresentative(String ccRId, String ccrName, String ccrAddress, String ccrPhone, String ccrEMail,
        int issuesInProgress, String issueIds) {
        super();
        this.ccRId = ccRId;
        this.ccrName = ccrName;
        this.ccrAddress = ccrAddress;
        this.ccrPhone = ccrPhone;
        this.ccrEMail = ccrEMail;
        this.issuesInProgress = issuesInProgress;
        this.issueIds = issueIds;
    }
    public String getCcRId() {
        return ccRId;
    }
    public void setCcRId(String ccRId) {
        this.ccRId = ccRId;
    }
    public String getCcrName() {
        return ccrName;
    }
    public void setCcrName(String ccrName) {
        this.ccrName = ccrName;
    }
}

```

```

    public String getCcrAddress() {
        return ccrAddress;
    }
    public void setCcrAddress(String ccrAddress) {
        this.ccrAddress = ccrAddress;
    }
    public String getCcrPhone() {
        return ccrPhone;
    }
    public void setCcrPhone(String ccrPhone) {
        this.ccrPhone = ccrPhone;
    }
    public String getCcrEMail() {
        return ccrEMail;
    }
    public void setCcrEMail(String ccrEMail) {
        this.ccrEMail = ccrEMail;
    }
    public int getIssuesInProgress() {
        return issuesInProgress;
    }
    public void setIssuesInProgress(int issuesInProgress) {
        this.issuesInProgress = issuesInProgress;
    }
    public String getIssuelds() {
        return issuelds;
    }
    public void setIssuelds(String issuelds) {
        this.issuelds = issuelds;
    }
    @Override
    public String toString() {
        return "CCRepresentative [ccRId=" + ccRId + ", ccrName=" + ccrName + ", ccrAddress=" + ccrAddress
            + ", ccrPhone=" + ccrPhone + ", ccrEMail=" + ccrEMail + ", issuesInProgress=" + issuesInProgress
            + ", issuelds=" + issuelds + "]";
    }
}

```

Customer.java

```

package com.cts.cit.model;
import javax.validation.constraints.NotBlank;
import javax.validation.constraints.NotNull;
import javax.validation.constraints.Pattern;
import org.springframework.stereotype.Component;

@Component
public class Customer {
    @NotBlank(message="Customer Id is required")
    String custId;
    @NotBlank(message="Customer name is required")
    String custName;
    @NotBlank(message="Customer address is required")
    String custAddress;
    @Pattern(regexp="^[789][0-9]{9}",message="Mobile number should be 10 digits and starting with 7/8/9")
    String custPhone;
    @NotBlank(message="loginId is required")
    String loginId;
    @NotBlank(message="password is required")
    String password;
    public Customer() {
        super();
    }
    public Customer(String custId, String custName, String custAddress, String custPhone, String loginId,
        String password) {
        super();
        this.custId = custId;
        this.custName = custName;
        this.custAddress = custAddress;
        this.custPhone = custPhone;
    }
}

```

```

        this.loginId = loginId;
        this.password = password;
    }
    public String getCustId() {
        return custId;
    }
    public void setCustId(String custId) {
        this.custId = custId;
    }
    public String getCustName() {
        return custName;
    }
    public void setCustName(String custName) {
        this.custName = custName;
    }
    public String getCustAddress() {
        return custAddress;
    }
    public void setCustAddress(String custAddress) {
        this.custAddress = custAddress;
    }
    public String getCustPhone() {
        return custPhone;
    }
    public void setCustPhone(String custPhone) {
        this.custPhone = custPhone;
    }
    public String getLoginId() {
        return loginId;
    }
    public void setLoginId(String loginId) {
        this.loginId = loginId;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
    @Override
    public String toString() {
        return "Customer [custId=" + custId + ", custName=" + custName + ", custAddress=" + custAddress + ", custPhone="
            + custPhone + ", loginId=" + loginId + ", password=" + password + "];"
    }
}

```

IssueDetail.java

```

package com.cts.cit.model;
import java.util.Date;
public class IssueDetail {
    String custIssueId;
    Date issueReportDate;
    String custId;
    String category; //Complaint - Feedback - Suggestion
    String description;
    String issueStatus; //InProgress;Closed;Open
    String ccRepld; // Customer Care Representative Id
    public IssueDetail() {
        super();
    }
    public IssueDetail(String custIssueId, Date issueReportDate, String custId, String category, String description,
        String issueStatus, String ccRepld) {
        super();
        this.custIssueId = custIssueId;
        this.issueReportDate = issueReportDate;
        this.custId = custId;
        this.category = category;
        this.description = description;
        this.issueStatus = issueStatus;
    }
}

```

```

        this.ccRepld = ccRepld;
    }
    public String getCustIssued() {
        return custIssued;
    }
    public void setCustIssued(String custIssued) {
        this.custIssued = custIssued;
    }
    public Date getIssueReportDate() {
        return issueReportDate;
    }
    public void setIssueReportDate(Date issueReportDate) {
        this.issueReportDate = issueReportDate;
    }
    public String getCustId() {
        return custId;
    }
    public void setCustId(String custId) {
        this.custId = custId;
    }
    public String getCategory() {
        return category;
    }
    public void setCategory(String category) {
        this.category = category;
    }
    public String getDescription() {
        return description;
    }
    public void setDescription(String description) {
        this.description = description;
    }
    public String getIssueStatus() {
        return issueStatus;
    }
    public void setIssueStatus(String issueStatus) {
        this.issueStatus = issueStatus;
    }
    public String getCcRepld() {
        return ccRepld;
    }
    public void setCcRepld(String ccRepld) {
        this.ccRepld = ccRepld;
    }
    @Override
    public String toString() {
        return "IssueDetail [custIssued=" + custIssued + ", issueReportDate=" + issueReportDate + ", custId=" + custId
            + ", category=" + category + ", description=" + description + ", issueStatus=" + issueStatus
            + ", ccRepld=" + ccRepld + "];"
    }
}

```

CustomerIssueService.java

```

package com.cts.cit.service;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;
import java.util.NoSuchElementException;
import org.springframework.stereotype.Service;
import com.cts.cit.exception.ApplicationException;
import com.cts.cit.model.CCRepresentative;
import com.cts.cit.model.Customer;
import com.cts.cit.model.IssueDetail;

```

```

@Service
public class CustomerIssueService {

```

```

public static List<IssueDetail> issuesList = new ArrayList<IssueDetail>();
public static List<CCRRepresentative> ccrList = new ArrayList<CCRRepresentative>();
public static List<Customer> customerList = new ArrayList<Customer>();
public void addCustomerObject(Customer c) {
    // write code to add valid customer object to collection
    customerList.add(c);
}
static {

    ccrList.add(new CCRRepresentative("CCR1", "RakeshSingh", "RKPuram", "8273838833", "rak@gmail.com", 4,
        "I1;I2;I3;I4"));
    ccrList.add(
        new CCRRepresentative("CCR2", "RajeshSingh", "RJPuram", "8262568833", "raj@gmail.com", 3, "I5;I6;I7"));
    ccrList.add(new CCRRepresentative("CCR3", "UttamSingh", "RNPuram", "8273873913", "utt@gmail.com", 2, "I8;I9"));

    issuesList.add(new IssueDetail("I1", stringToDateConverter("2020-04-12"), "C001", "Complaint",
        "M/C Not Working", "InProgress", "CCR1"));
    issuesList.add(new IssueDetail("I2", stringToDateConverter("2020-03-13"), "C001", "Complaint",
        "Mixie Not Working", "InProgress", "CCR1"));
    issuesList.add(new IssueDetail("I3", stringToDateConverter("2020-02-14"), "C001", "Complaint",
        "Fridge Not Working", "InProgress", "CCR1"));
    issuesList.add(new IssueDetail("I4", stringToDateConverter("2020-01-15"), "C002", "Complaint", "TV Not Working",
        "InProgress", "CCR1"));
    issuesList.add(new IssueDetail("I5", stringToDateConverter("2020-01-16"), "C002", "Complaint",
        "Stove Not Working", "InProgress", "CCR2"));
    issuesList.add(new IssueDetail("I6", stringToDateConverter("2020-02-17"), "C002", "Complaint",
        "M/C Not Working", "InProgress", "CCR2"));
    issuesList.add(new IssueDetail("I7", stringToDateConverter("2020-03-17"), "C003", "Complaint",
        "Mixie Not Working", "InProgress", "CCR2"));
    issuesList.add(new IssueDetail("I8", stringToDateConverter("2020-04-18"), "C003", "Complaint",
        "Grinder Not Working", "InProgress", "CCR3"));
    issuesList.add(new IssueDetail("I9", stringToDateConverter("2020-05-19"), "C003", "Complaint",
        "Fridge Not Working", "InProgress", "CCR3"));

}
// Get the First CCR in the List who has least number of issues being handled by
// him/her
public CCRRepresentative getAvailableCCR() {
    CCRRepresentative ccrByMinIssues = null;
    // Type Your Code here
    List<Integer> list = new ArrayList<Integer>();
    for (CCRRepresentative obj : ccrList) {
        list.add(obj.getIssuesInProgress());
    }
    Collections.sort(list);
    int min = list.get(0);
    for (CCRRepresentative obj : ccrList) {
        if (min == obj.getIssuesInProgress())
            ccrByMinIssues = obj;
    }
    return ccrByMinIssues;
}

public CCRRepresentative getCCRById(String ccrId) {
    CCRRepresentative ccRep = new CCRRepresentative();
    // Type Your Code here
    for (CCRRepresentative ccr1 : ccrList) {
        if (ccr1.getCcRId().equalsIgnoreCase(ccrId)) {
            ccRep = ccr1;
        }
    }

    return ccRep;
}

public IssueDetail getIssueDetailById(String issueId) {
    IssueDetail issueDetail = new IssueDetail();

    // Type Your Code here

```

```

        for (IssueDetail issue : issuesList) {
            if (issue.getCustIssued().equalsIgnoreCase(issueId)) {
                issueDetail = issue;
            }
        }

        return issueDetail;
    }

    public String getMaxIssued() {
        String maxIssued = "1";
        //Type Your Code here
        int max = 0;
        for (IssueDetail issue : issuesList) {
            int issued = Integer.parseInt(issue.getCustIssued().substring(1));
            if(issued>max)
                max = issued;
        }
        maxIssued = maxIssued + max;
        return maxIssued; // 19
    }

    public boolean validateDateOfReport(String dateOfRepo) {
        boolean flag = false;
        // Type Your Code here
        if(stringToDateConverter(dateOfRepo)!=null)
            flag = true;
        else
            flag = false;

        return flag;
    }

    // NOT A BUSINESS FUNCTION BUT UTILITY FUNCTION
    public static java.util.Date stringToDateConverter(String stringDate) {
        SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd");
        dateFormat.setLenient(false);
        try {
            return dateFormat.parse(stringDate);
        } catch (ParseException pe) {
            return null;
        }
    }
}

```

CustomerIssueTrackerApp.java

```

package com.cts.cit;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import com.cts.cit.skeleton.SkeletonValidator;
@SpringBootApplication
public class CustomerIssueTrackerApp {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        SpringApplication.run(CustomerIssueTrackerApp.class, args);
        SkeletonValidator skeletonValidator = new SkeletonValidator();
    }
}

```

CustomerForm.jsp

```

<%@ taglib prefix="spring" uri="http://www.springframework.org/tags"%>
<%@ taglib uri="http://www.springframework.org/tags/form" prefix="form"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Customer's Page</title>

```



```

</head>
<body style='margin-left: 30px'>
    <h2 id="heading3">Welcome To Customer Page</h2>
    <!-- TYPE YOUR CODE HERE -->
    <form name="form" modelAttribute="customer" action="customerform" method=post >
        <table>
            <tr>
                <td>Please Select Your Option: </td>
                <td><input type="radio" id="ri" name="option" value="ri" />Raise Issue</td>
                <td><input type="radio" id="vi" name="option" value="vi" />View Issue Status</td>
            </tr>
            <tr>
                <td>Enter Id of Issue You Wish to See</td>
                <td><input type="text" id="issueid" name="issueid" /></td>
            </tr>
            <tr>
                <td><input type="submit" id="sub1" name="submit" value="Submit" /></td>
            </tr>
        </table>
    </form>
</body>
</html>

```

CustomerIssue.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Customer Issue Details</title>
</head>
<body>
<h3 id="cust">Customer Issue Details</h3>
    <!-- TYPE YOUR CODE HERE -->
    <table border="1">
        <tr>
            <th>Issue Id</th>
            <th>Issue Report Date</th>
            <th>Customer Id</th>
            <th>Category</th>
            <th>Description</th>
            <th>Status</th>
            <th>CC Rep Id</th>
        </tr>
        <tr>
            <td>${issueobj.getCustIssueId()}</td>
            <td>${issueobj.getIssueReportDate()}</td>
            <td>${issueobj.getCustId()}</td>
            <td>${issueobj.getCategory()}</td>
            <td>${issueobj.getDescription()}</td>
            <td>${issueobj.getIssueStatus()}</td>
            <td>${issueobj.getCcRepId()}</td>
        </tr>
    </table>
    <br />
    <a href="login1" id="login">Back to Login Page</a> <br />
    <a href="customerform" id="customerform">Back To Customer's Page</a>

</body>
</html>

```

Error1.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>

```

```

</head>
<body>
    <h3 id="errPg">Error Page!!!</h3>
    <!-- TYPE YOUR CODE HERE -->
    ${msg }<br />

<br/>
    <a href="login1" id="login">Back to Login Page</a><br />
    <a href="customerform" id="fback">Back To Customer's Page</a>

</body>
</html>

```

```

IssueRaised.jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Following Issue Is Raised</title>
</head>
<body>
    <h3 id="issueR">Following ISsue Is Raised....</h3>
    <!-- TYPE YOUR CODE HERE -->
    <table border="1">
        <tr>
            <th>Issue Id</th>
            <th>Issue Report Date</th>
            <th>Customer Id</th>
            <th>Category</th>
            <th>Description</th>
            <th>Status</th>
            <th>CC Rep Id</th>
        </tr>
        <tr>
            <td>${issue.getCustIssuedId()}</td>
            <td>${issue.getIssueReportDate()}</td>
            <td>${issue.getCustId()}</td>
            <td>${issue.getCategory()}</td>
            <td>${issue.getDescription()}</td>
            <td>${issue.getIssueStatus()}</td>
            <td>${issue.getCcRepld()}</td>
        </tr>
    </table>
    <br />
    <a href="login1" id="back">Back to Login Page</a><br />
    <a href="customerform" id="fback">Back To Customer's Page</a>

</body>
</html>

```

```

Login.jsp
<%@taglib uri="http://www.springframework.org/tags" prefix="spring"%>
<%@taglib uri="http://www.springframework.org/tags/form" prefix="form"%>
<%@ page contentType="text/html; charset=UTF-8"%>
<%@ page session="false"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<style type="text/css">
.loginform {
    align-content: center;
}
.error
{
    color: red;
}

```

```

}
table
{
    margin:0 auto;
}
</style>
<title>Welcome To Login Page</title>
</head>
<body>
    <div
        style="text-align: right; padding: 5px; margin: 5px 0px; background: #ccc;"
    </div>
    <!-- TYPE YOUR CODE HERE -->
    <h2 id="logid">Welcome To Home Page</h2>
    <form:form name="form" action="/login" method="post" modelAttribute="customer" >
        <table>
            <tr>
                <td><form:label path="custId">Customer Id:</form:label></td>
                <td><form:input path="custId" id="custId" name="custId" /></td>
                <td><form:errors path="custId" cssClass="error" ></form:errors> </td>
            </tr>
            <tr>
                <td><form:label path="custName">Customer Name:</form:label></td>
                <td><form:input path="custName" id="custName" name="custName" /></td>
                <td><form:errors path="custName" cssClass="error" ></form:errors> </td>
            </tr>
            <tr>
                <td><form:label path="custAddress">Customer Address:</form:label></td>
                <td><form:input path="custAddress" id="custAddress" name="custAddress" /></td>
                <td><form:errors path="custAddress" cssClass="error" ></form:errors> </td>
            </tr>
            <tr>
                <td><form:label path="custPhone">Customer Phone:</form:label></td>
                <td><form:input path="custPhone" id="custPhone" name="custPhone" /></td>
                <td><form:errors path="custPhone" cssClass="error" ></form:errors> </td>
            </tr>
            <tr>
                <td><form:label path="loginId">Create new login Id:</form:label></td>
                <td><form:input path="loginId" id="loginId" name="loginId" /></td>
                <td><form:errors path="loginId" cssClass="error" ></form:errors> </td>
            </tr>
            <tr>
                <td><form:label path="password">Create new Password:</form:label></td>
                <td><form:password path="password" id="password" name="password" /></td>
                <td><form:errors path="password" cssClass="error" ></form:errors> </td>
            </tr>
            <tr>
                <td><input type="submit" value="submit" id="submit" name="submit" ></td>
                <td><input type="reset" value="Clear" id="Clear" name="Clear" ></td>
            </tr>
        </table>
    </form:form>
</body>
</html>

```

RiseIssue.jsp

```

<%@taglib uri="http://www.springframework.org/tags" prefix="spring"%>
<%@ page contentType="text/html; charset=UTF-8"%>
<%@ page session="false"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<style type="text/css">
.raiseissue {
    align-content: center;
}
table

```

```

{
    margin:0 auto;
}
</style>
<title>RAISE ISSUE</title>
</head>
<body>

    <!-- TYPE YOUR CODE HERE -->

    <form action="/raiseissue" method="post">
        <table>
            <tr>
                <td><b>Issued Id:</b></td>
                <td><input type="text" name="custIssueId" value="{custIssueId}" ></td>
            </tr>
            <tr>
                <td><b>Issued Report Date:</b>Format:yyyy-mm-dd</td>
                <td><input type="text" name="issueReportDate" ></td>
            </tr>
            <tr>
                <td><b>Customer Id:</b></td>
                <td><input type="text" name="custId" ></td>
            </tr>
            <tr>
                <td><b>Category:</b></td>
                <td><input type="text" name="category" ></td>
            </tr>
            <tr>
                <td><b>Description:</b></td>
                <td><textarea rows="5" cols="20" name="descr" ></textarea> </td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Submit" name="Submit" ></td>
            </tr>
        </table>
    </form>

</body>
</html>

```