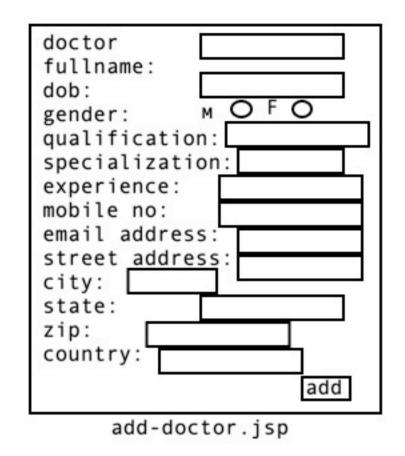
- hospital management system - doctors
- patients
- billing
- user management module



```
class AddDoctorServlet extends HttpServlet {
  public void service(HttpServletRequest httpReq,
HttpServletResponse httpResp) {
     String doctorName =
httpReq.getParameter("doctorName");
     String dob = httpReq.getParameter("dob");
     String gender =
httpReq.getParameter("gender");
     String qualification =
httpReq.getParameter("qualification");
     // read all the req parameter values that
user has submitted from jsp form page.
     DoctorForm form = new DoctorForm();
     form.setDoctorName(doctorName):
     form.setGender(gender);
     form.setDob(sdf.parse(dob));
     form.setQualification(qualification);
     form =
DoctorRequestWrapper.buildDoctorForm(httpReq);
     // populate all the data into form object
    doctorService.addDoctor(form);
```

class DoctorService { public int addDoctor (DoctorForm form) { // business logic

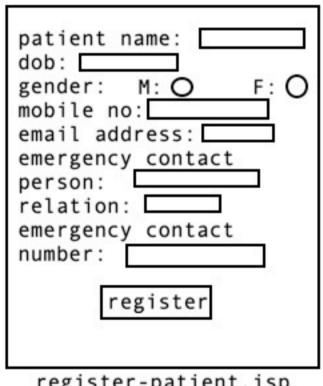
class DoctorDao {

DoctorForm

 String doctorName; - Date dob: - String gender; - String qualification; String specialization; - int experience; - String mobileNo; - String emailAddress; - String streetAddress; - String city; - String state; int zip; - String country;

Dont write business and persistence logic inside the Servlet class itself, because: 1. if we write business/persistence logic inside the servlet class, in future if we move to an different presentation-tier technology instead of servlets to an Struts or Spring Mvc Framework, then we need to remove Servlet class and replace with Action/Controller classes of the respective frameworks. When we remove the Servlet, along with presentation-tier logic in receiving the request, we loose business/persistence logic aswell and we need to rewrite the application from scratch.

2. If we write business/persistence logic inside the servlet we cannot reuse it in other parts of the application.



register-patient.jsp

class RegisterPatientServlet extends HttpServlet { public void service(httpReq, httpResp) { PatientForm form = new PatientForm(); String patientName = httpReq.getParameter("patientName"); String dob = httpReq.getParameter("dob"); // all the request parameter values and bind to form object form.setPatientName(patientName); form.setGender(gender); patientService.registerPatient(form);

class PatientService { int registerPatient(PatientForm form) {

PatientForm String patientName; Date dob; String gender; String mobileNo; String emailAddress: String emergencyContactPerson; String relation: String emergencyContactNo;

register-patient.jsp patient name: <input type="text" name="patientName"/> dob: <input type="text" name="dob"/> gender: <input type="text" name="gender"/>

> patientName: alex dob: 10/01/1999 gender: Male mobileNo: 9383984 emailAddress: alex@gmail.com emergencyContactPerson: bob emergenecyContactNo: 929384944 relation: Father

class DoctorRequestWrapper { DoctorForm buildDoctorForm(HttpServletRequest reg) { String doctorName = httpReq.getParameter("doctorName"); String dob = httpReq.getParameter("dob"); String gender = httpReq.getParameter("gender"); String qualification = httpReq.getParameter("qualification"); // read all the req parameter values that user has submitted from jsp form page. DoctorForm form = new DoctorForm(); form.setDoctorName(doctorName); form.setGender(gender); form.setDob(sdf.parse(dob)); form.setQualification(qualification); return form;