Building ToDo app hibernate validator

As a Developer you should never be dependent on Client side validation. You must have to do server side validation.
We will use JSR validation (Java Specification Requests)
The JSR 303 and JSR 349 defines specification for the Bean Validation API (version 1.0 and 1.1, respectively), and Hibernate Validator is the reference implementation.
org.hibernate:hibernate-validator

hibernate-validator is something like if you have a bean and a property of a bean should have six character then it can be used as server side validation

Implementing server side validation
Command Bean or Form Backing Bean
What would you do if you want to send multiple data from form. Using multiple request param is not a good way so we must use command bean.
Once we setup command bean we can add validations on them.
Once we add the validations on bean we can use them in the controller
If their are any validation error then you can show it in the view

```
@Controller
@SessionAttributes("name")
public class TodoController {

    @Autowired
    private TodoService todoService;

    @RequestMapping(value = "/list-todos", method =
RequestMethod.GET)
    public String listTodo(ModelMap modelMap) {
        String name = (String) modelMap.get("name");
        modelMap.put("todos", todoService.listTodo(name));
        return "list-todos";
    }
```

```java
    @RequestMapping(value = "/add-todo", method =
RequestMethod.GET)
    public String showTodoPage(ModelMap modelMap) {
        modelMap.addAttribute("todo", new Todo(0, (String)
modelMap.get("name"), "", new Date(), false));
        return "todo";
    }

    @RequestMapping(value = "/add-todo", method =
RequestMethod.POST)
    public String addTodo(ModelMap modelMap, Todo todo) {
        todoService.addTodo(new Todo((String)
modelMap.get("name"), todo.getDesc(), new Date(), false));
        return "redirect:/list-todos";
    }

    @RequestMapping(value = "/delete-todo", method =
RequestMethod.GET)
    public String deleteTodo(@RequestParam int id) {
        todoService.deleteTodo(id);
        return "redirect:/list-todos";
    }
}
```

todo.jsp

```jsp
<%@taglib uri="http://www.springframework.org/tags/form"
prefix="form"%>
<html>
    <head>
        <title>First web application</title>
        <link href="webjars/bootstrap/4.5.3/css/
bootstrap.min.css" rel="stylesheet">
    </head>

    <body>
        <div class = "container">
            Add Todo page for ${name}

            <form:form method="post" modelAttribute="todo">
                <fieldset class = "form-group">
                    <form:label path = "desc">Description</
form:label>
                    <form:input path = "desc" type = "text"
class = "form-control" required="required"/>
                </fieldset>
                <button type = "submit" class = "btn btn-
success">Add</button>
            </form:form>

        </div>

        <script src="webjars/jquery/3.5.1/jquery.min.js"></
script>
```

```html
            <script src="webjars/bootstrap/4.5.3/js/
bootstrap.min.js"></script>
    </body>
</html>
```