

Building ToDo app hibernate validator

Add default description

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

Now we will try to restrict non empty description/value with minimum of 10 characters validating in server side. It can be done by using

Starting with Boot 2.3, we also need to explicitly add the spring-boot-starter-validation dependency

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-validation</artifactId>
</dependency>
```

Todo model

```
@Getter @Setter
@NoArgsConstructor
@AllArgsConstructor
@ToString
public class Todo {

    private int id;
    private String user;
    @Size(min = 10, message = "Enter atleast 10 characters")
    private String desc;
    private Date targetDate;
    private boolean isDone;

    public Todo(String user, String desc, Date targetDate, boolean
isDone) {
        super();
        this.user = user;
        this.desc = desc;
        this.targetDate = targetDate;
        this.isDone = isDone;
    }
}
```

When we add a todo data with desc field being less than 10 characters it won't work because we added a validation on bean not on Controller. To enable validation on controller we need to add @Valid annotation.

```
@RequestMapping(value = "/add-todo", method = RequestMethod.POST)
public String addTodo(ModelMap modelMap, @Valid Todo todo) {
    todoService.addTodo(new Todo((String) modelMap.get("name"),
```

```

    todo.getDesc(), new Date(), false));
    return "redirect:/list-todos";
}

```

Now it works but we need to handle the error (It gives runtime error it should not happen it should throw the error)

How do we know the validation is success or not ? By using BindingResult (It will have result which is success or failure)

```

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

```

Now when you try it. It works and throw error at server side but the client side will not get to know. To enable this we need to use JSTL error

```

<%@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"/>
                    <form:errors path = "desc" cssClass =
"text-warning"/>
                </fieldset>
                <button type = "submit" class = "btn btn-
success">Add</button>
            </form:form>

        </div>

        <script src="webjars/jquery/3.5.1/jquery.min.js"></
script>
        <script src="webjars/bootstrap/4.5.3/js/
bootstrap.min.js"></script>

```

```
        </body>  
</html>
```

Now it works properly by showing error message at client side
There are few more JSR Validation explore it.