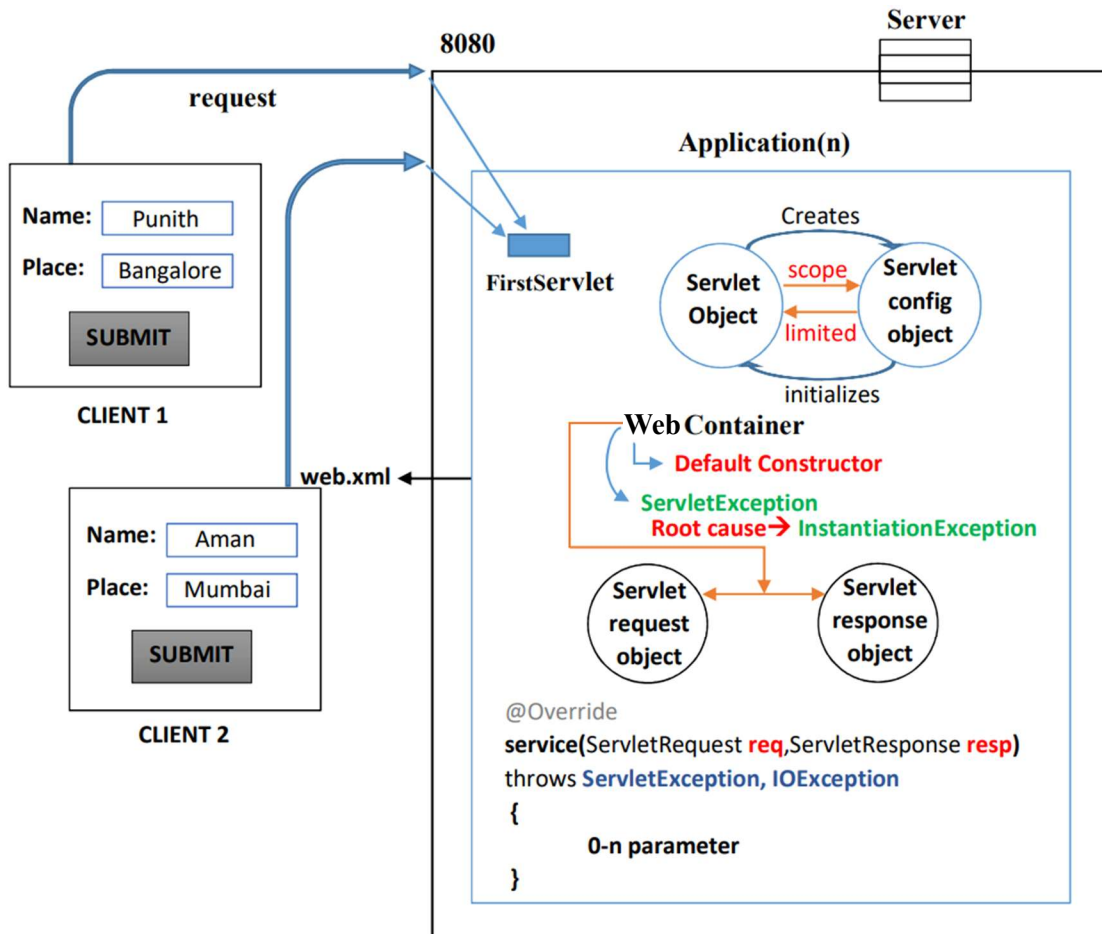


- If this phase fails, then Web container throws an Exception called as ServletException.



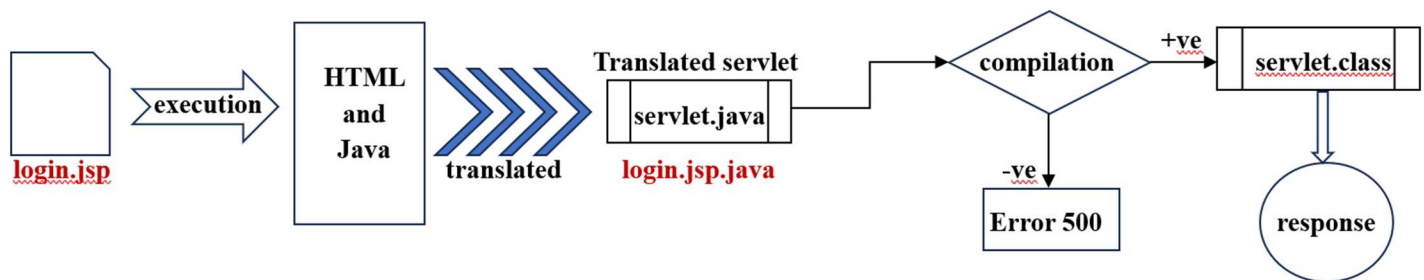
Destruction Phase

- In this phase, the `destroy()` is called by Web container to close all costly resources.
- `destroy()` is called by Web container only once.
- If this phase fails, then the performance of application decreases.

JSP [Java Server Page]

- Servlet & JSP both technologies are provided for server-side web applications development.
- In Servlet technology, a Java developer has to implement both application logic and presentation logic in a servlet class.
- To implement presentation logic, user can't write html tags directly in a servlet class. It must be wrapped into Java code only.
- It means, user must use PrintWriter class for writing html tags.
- Hence the complication occurs in servlet by writing both business logic and presentation logic.
- So, the alternate technology provided for separating application logic and presentation logic is JSP.
- In a JSP page, we can write html tags directly. So, implementing presentation logic becomes easy.
- The UI team can work on presentation logic and Java team can work on Application logic, so that both logics can be implemented parallelly.

JSP Architecture



- In a JSP page, we can write html tags and JSP tags.
- HTML tags are for applying the presentation and JSP tags are for generating dynamic content.
- Every JSP page will be translated into a servlet program, then it is compiled into a class then the web container creates an object for that class.
- The JSP page will be converted into a servlet program, when it receives very first request. For other requests, the existing servlet object in server, will process the request and sends the response to the client.

Tags in JSP

1. Scriptlet Tag:

This tag is used to write the java code in the JSP file.

```
<% //Java logic %>;  
<% Class ref_name=new Constructor() %>;
```

2. Expression Tag:

This tag is used to print the content of the Java code.

```
<% Class ref_name=new Constructor() %>;  
<% = ref_name %>; //expression tag
```

3. Import Attribute:

This attribute is present inside the page directive. It is used to import the java library packages.

```
<%@ page import="java.util.*" %>
```

4. Include directive:

This attribute is used to include the source code of another JSP file.

```
<%@ include file="nextPage.jsp" %>
```

Rules of JSP tags

- We cannot write one JSP tag inside another JSP tag.

```
<%  
    <% %>  
%>
```



- We cannot write HTML tags inside any JSP tags.

```
<%  
    <% %>  
%>
```




- We can write multiple JSP tags.

```
<% %>  
<%= %>  
<%= %>  
<% %>
```



- We can write JSP tags inside another JSP tags.

```
<h1> <% %> </h1>  
<td> <%= %> </td>
```



Implicit Objects in JSP

Implicit objects are a set of Java objects that the JSP Container makes available to developers on each page. These objects may be accessed as built-in variables via scripting elements and can also be accessed programmatically by JavaBeans and Servlets.

- **request:** This is the object of `HttpServletRequest` class associated with the request.
- **response:** This is the object of `HttpServletResponse` class associated with the response to the client.
- **session:** This is the object of `HttpSession` class associated with the request
- **out:** This is the `PrintWriter` object where methods like `print` and `println` help for displaying the content to the client.