**FRONT CONTROLLER DESIGN PATTERN**

[**Definition**](https://www.google.com/search?hl=en&sxsrf=ALiCzsZJAe661oN3fSniZlg2RZb4rwgnmA:1670213941377&q=definition&spell=1&sa=X&ved=2ahUKEwiDv7Llz-H7AhUH0YUKHWX7ByYQBSgAegQICBAB)**:**

The front controller design pattern means that **all requests that come for a resource in an application will be handled by a single handler and then dispatched to the appropriate handler for that type of request**. The front controller may use other helpers to achieve the dispatching mechanism.

**How It Works:**

This handler can do the authentication/ authorization/ logging or tracking of request and then pass the requests to corresponding handlers. Following are the entities of this type of design pattern.

* **Front Controller** - Single handler for all kinds of requests coming to the application (either web based/ desktop based).
* **Dispatcher** - Front Controller may use a dispatcher object which can dispatch the request to corresponding specific handler.
* **View** - Views are the object for which the requests are made.

**Use:**

* When you want to control the page flow and navigation.
* When you want to access and manage the data model.
* When you want to handle the business processing.

**Advantages:**

* It reduces the duplication of code in JSP pages, especially in those cases where several resources require the same processing.
* It maintains and controls a web application more effectively.
* A web application of two-tier architecture, the recommended approach is front controller to deal with user requests.
* One advantage of using a Front Controller is its testability.

**DisAdvantages:**

* It is not possible to scale an application using a front controller.
* We have a perfectly good front controller its called a web browser.
* Each http request is unique and separate and should be treated as such.
* If you break a web application into small modules that are loosely coupled its easier to test the unit/module (your not testing the architecture as well as the controller for example) .

**UML Diagram:**

