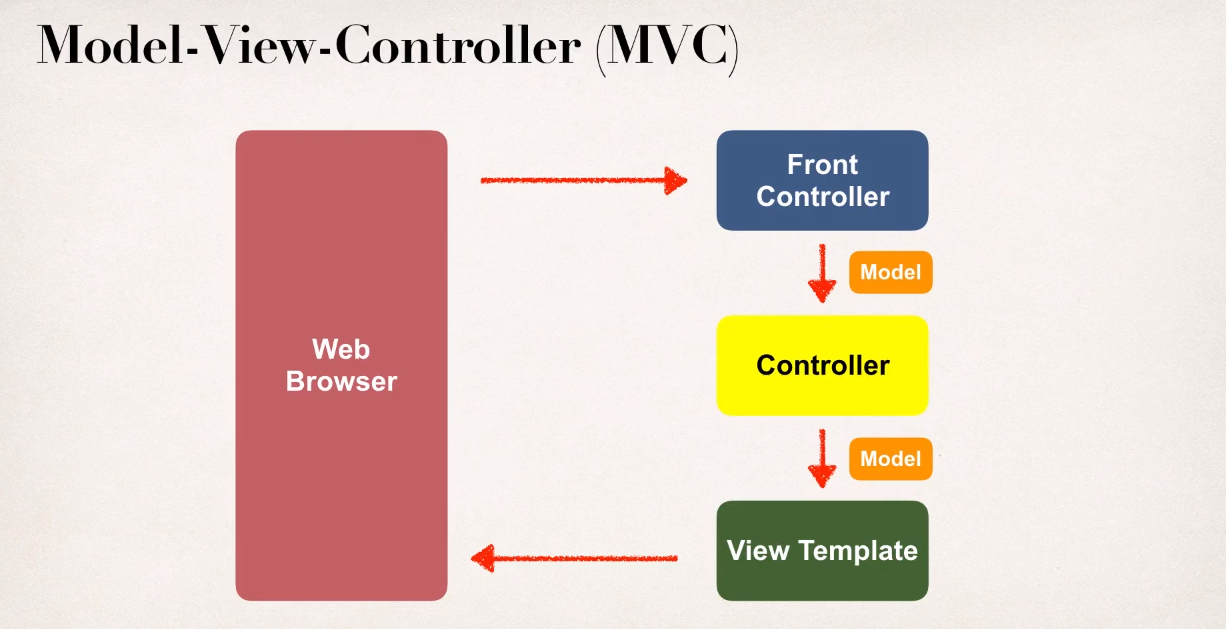
**Spring MVC**

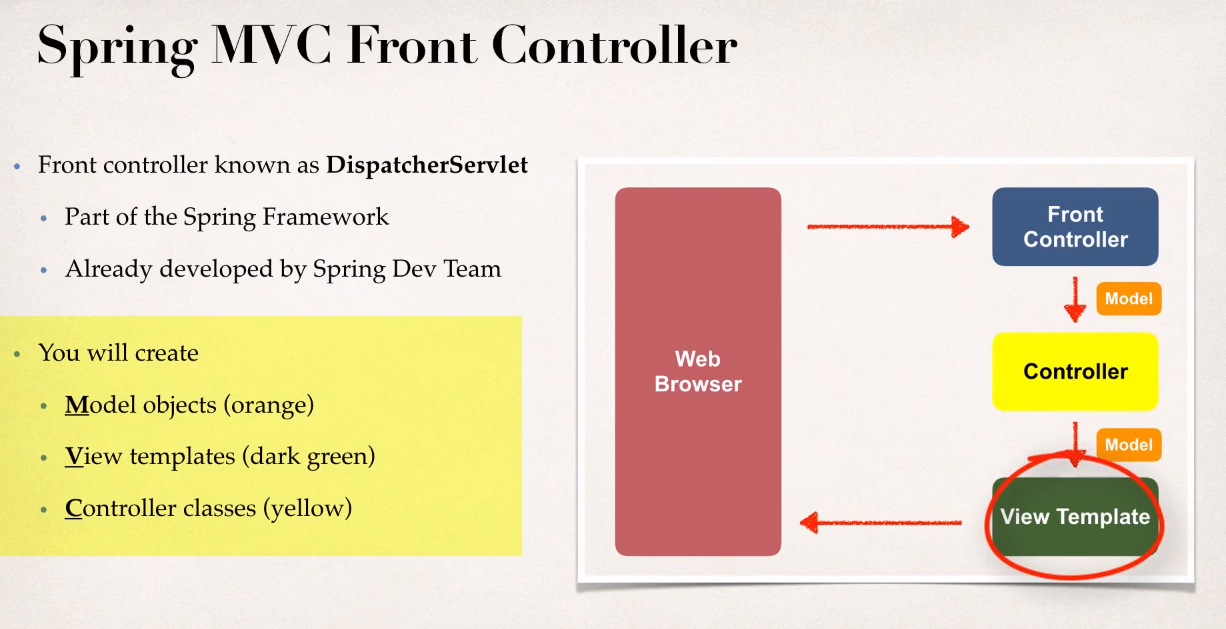
What is Spring MVC?

* Framework for building web applications in Java
* Based on **Model-View-Controller** design pattern
* Leverages features of the Core Spring Framework(IoC, DI)



Spring MVC Benefits

* The Spring way of building web app UIs in Java
* Leverage a set of reusable UI components
* Help manage application state for web requests
* Process form data: validation, conversion etc
* Flexible configuration for the view layer



**Controller:**

* Code created by developer
* Contains your business logic
  + Handle the request
  + Store/retrieve data (db, web service ...)
  + Place data in model
* Send to appropriate view template

**Model:**

* Model:contains your data
* Store/retrieve data via backend systems
  + database,web service,etc ...
  + Use a Spring bean if you like
* Place your data in the model
  + Data can be any Java object/collection

**View Template:**

* Spring MVC is flexible
  + Supports many view templates
* Most common is JSP + JST

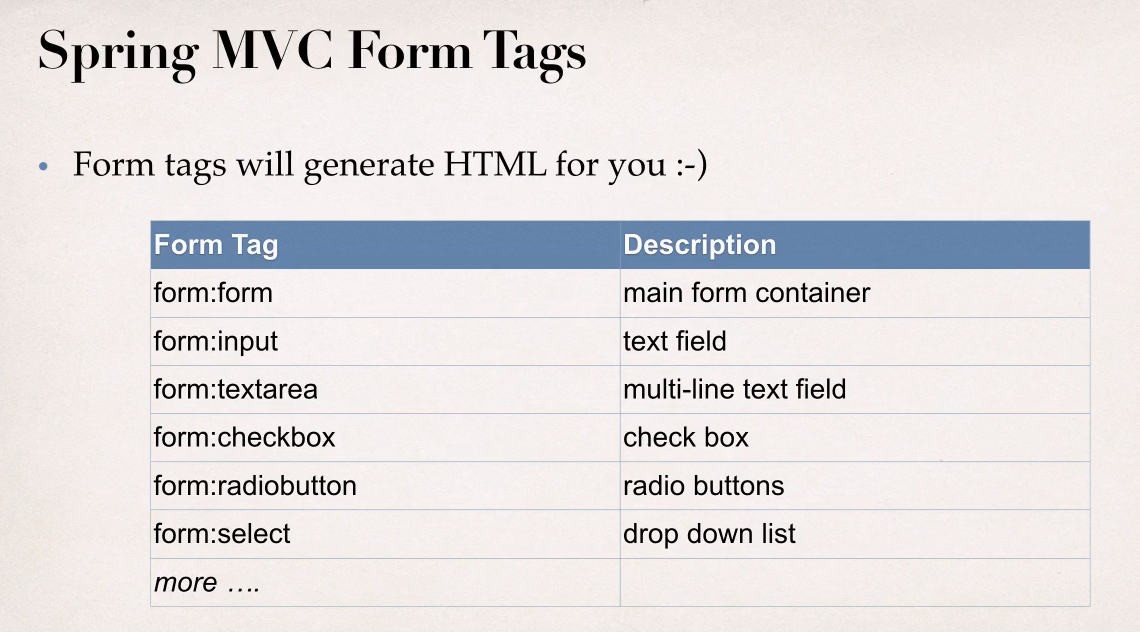
JSP: Java Server Pages

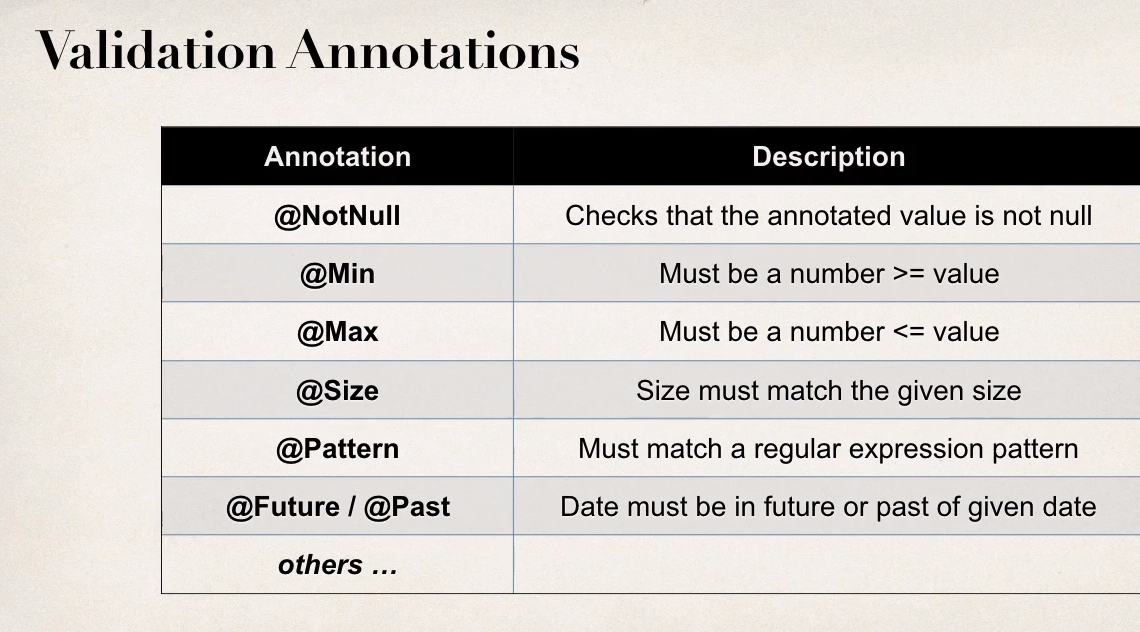
JSTL:JSP Standard Tag Library

* Developer creates a page
  + Displays data

**Development Process:**

1. Create Controller class
2. Define Controller method
3. Add Request Mapping to Controller method
4. Return View Name
5. Develop View Page





**@InitBinder:**

* @InitBinder annotation works as a pre-processor
* It will pre-process each web request to our controller
* Method annotated with @InitBinder is executed

**Custom Annotation:**

* group: It allows the specification of validation groups, to which this constraint belongs.
* message: This will return the text message in case constraints get violated.
* payload: It is used by the client of the validation API to add the custom payload. It is not used by the API itself.