### Final Project:

Submission Directions

1. All of your project code should be available on a publicly available repo such as GitHub. It must be publicly available.  I should be able to pull the code, and your distribution must have a build/make process that is easy to follow
2. show a demo of your solution.  The video should be available online over the Internet. should not be longer than 5 min in length and should have audio describing what you are doing as you demo your solution.
3. submit a single file to blackboard that includes your name and links to the video and source code repository.  You can also put any additional "readme" information that I will need to pull your code, build it, and test it.  The single file should also have a few views (between one and four) describing the architecture of your solution and a short description explaining the views that you provide.

**Project Topic:**

Assignment notification system: Developing a system that allows professors to add assignments and deadlines to the calendar. Professor can then send email to students containing the assignment information.

Group Members:

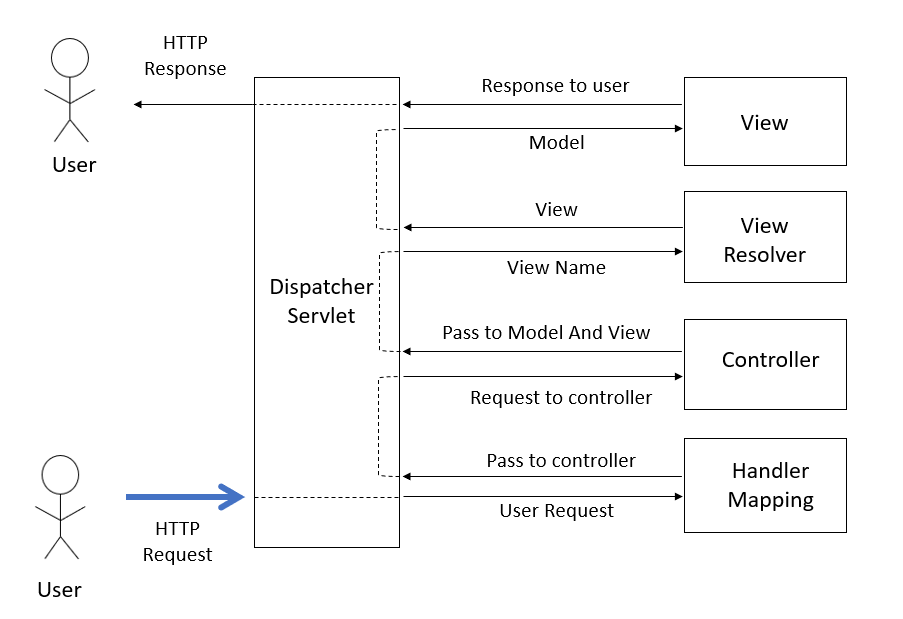
1. Anubha Barve (University ID: 14323474)
2. Mruga Shah (University ID: 14307770)

Architecture:

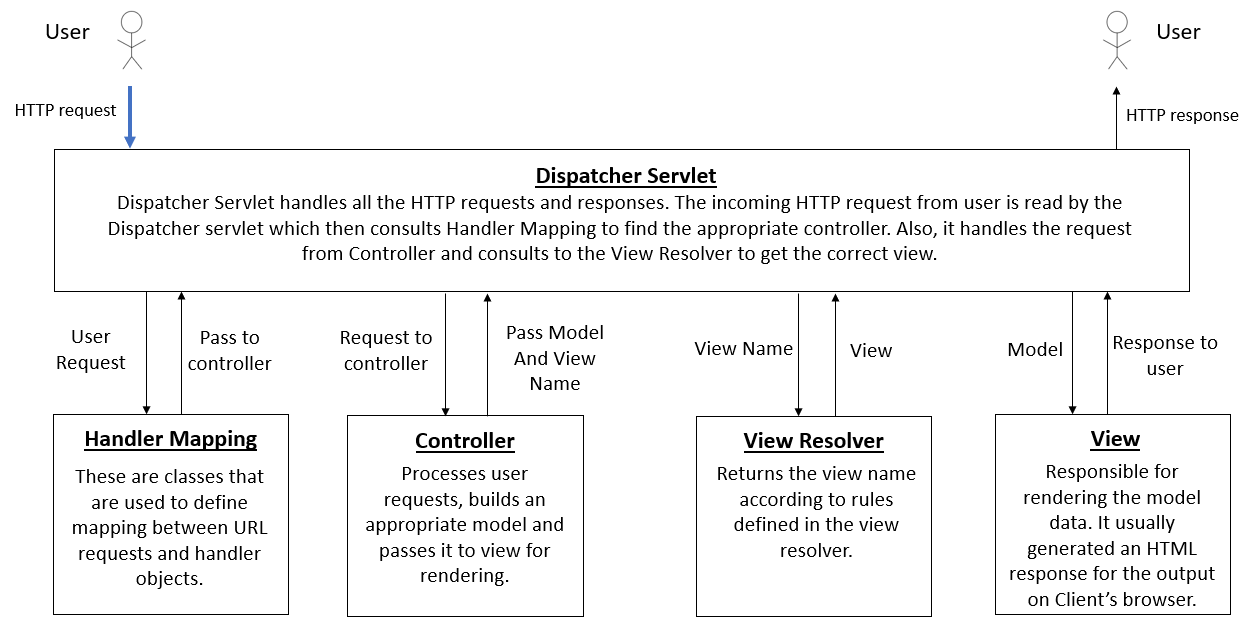
In building our website, we have used the Spring Web MVC framework.The MVC(Model-View-Controller) pattern separates the input logic, business logic and the UI logic while providing loose coupling between them.

The main component of the framework Dispatcher Servlet handles all the requests and responses. When a request from the user comes in, the Dispatcher Servlet first consults the Handler Mapping to call the appropriate Controller.According to the request, the Controller calls the appropriate service methods. Next, the service method sets the model data based on defined business logic and returns view name to the DispatcherServlet. The DispatcherServlet takes help from ViewResolver and pick-ups the defined view for the request.Once the view is finalized, The DispatcherServlet passes the model data to the view which is sent as an HTTP response back to user and rendered on the browser.

Overview Diagram



Detailed Diagram



Below is the diagram, where we explain the architecture diagram of our website. The steps followed are:

Step1:

The http request from the user (when the website is opened) goes to the home controller which processes the request and directs the request to Data Access Object (Dao).

Step2:

The Dao then directs the request to the appropriate model. The model sends the data which goes back to the home controller via Dao.

Step3:

The home controller then sends the request to get the view from the views(.jsp files) where all the model data is rendered on the respective view. Next, the response is sent back to the home controller that displays it on the Users screen.

Diagram including components of website we built

