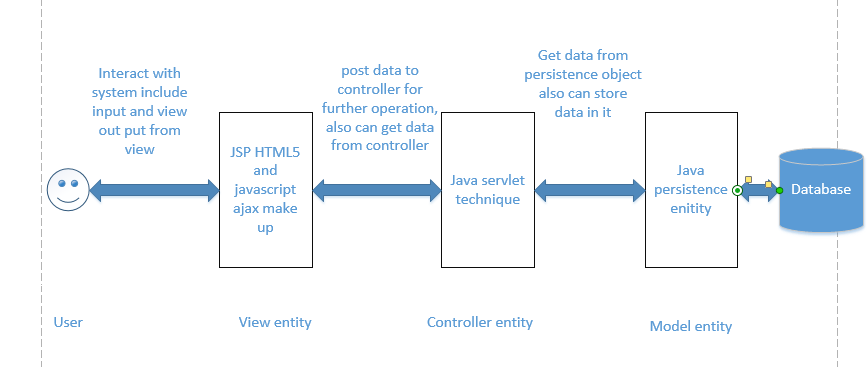
The architecture we are using is MVC pattern, it the most common pattern which everyone use when deploy web application.



View side:

We use HTML5 makeup which provide interaction with like date input and also it can output like table and list data and button for user to add delete or modify the data in the database. But for advance feature of view, we should provide a user friendly interface for user to use, which provide more aesthetic and usability. So I choose to use bootstrap framework which boost the quality of layout and also I use ajax to be a neutral entity to communicate with controller, the advantage is to ensure every time user submit date, this will use HTTP protocol get and post method, ajax will prevent refreshing the whole pages, which very impact the user mood.

So the view entity are JSP HTML pages, and js files with ajax technique

Controller side:

I use java servlet technique as controller, controller side will get post or get data from view side, do some calculation or communicate with persistence object, and store this data in the database, vice verse , java servlet can also get data from persistence object and show this data to view side, about advantage of using java servlet class as controller is mainly because each servlet class is a use case which make business logic very clear to understand, and it is multi-threading safe which will support multiple clients to use concurrently.

So the controller entities are java servlet classes

Modal side:

I choose to use JPA which JAVA persistence architecture as modal entity class, the advantage of using JPA is mainly because this technique can do as ORM which object related mapping, first this is object oriented design principle, second it provides automation of mapping between database and persistence class, each attribute of class is an column of table in database, and also provide some assistance function like set and get, so when retrieve data from database it will extract meta-data information and convert into persistence entity class, and control side can just new instance of these class and extract data

So the model entities are java persistence classes