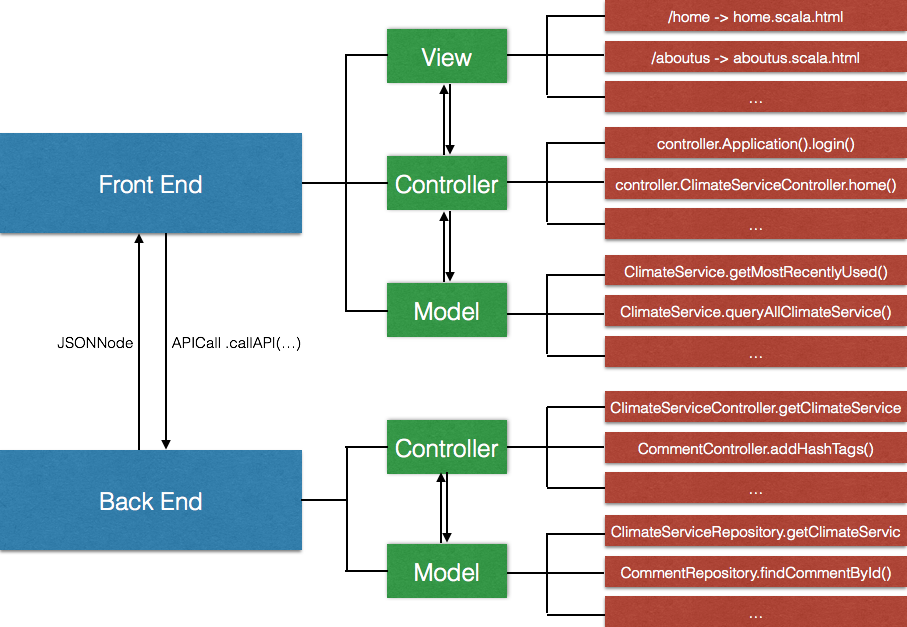
# **System Design**

Architecture Design



Our system is consisted of Front End and Back End, each of which has an independent Model View Controller (MVC) structure.

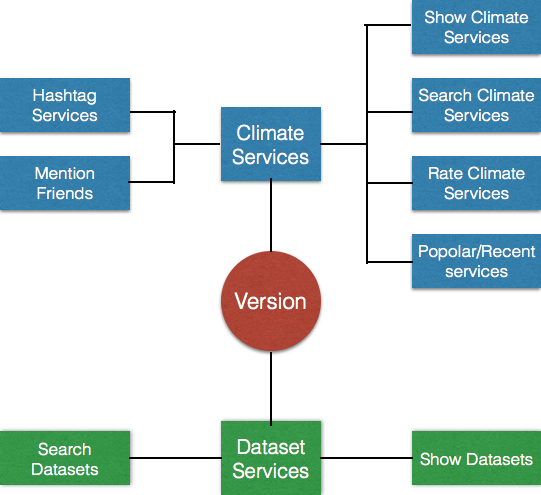
The front end view is displayed to the user, written in Scala. When the user requests a page, we route the request to its designated controller. The front end controller then calls the front end model in charge of handling the submitted request.

On the backend, we expose a RESTful API that can be invoked by the front end model. When the front end model invokes the backend API, a controller is triggered, which in turn calls the back end model. The back end model queries the underlying MySQL databases, and return the result to the back end controller.

When the backend receives the response, it then create a JSON object and passes it back to the front end. The JSON object is parsed by the front end model, sent back to the controller and the front end view. The result will then be displayed to the user.

The entire operation above happens within a fraction of milliseconds.

Feature Design



We are working with 2 major kinds of data, the climate services and the dataset services. All of our features are built on top of version control. Given that climate services and dataset services are constantly changing, it is important that we enable version-specific features to the users.

When the user selects a version, she is then exposed to many features. For dataset services, we have implemented features like showing all datasets, and searching datasets.

For climate services, the user will be able to list all climate services, search climate services using name or keywords, rate/grade climate services, and view popular/recent climate services. To make our site more interactive, we also implemented social features, including hashtagging climate services, and mentioning friends. A notification system has been implemented so that when a user is mentioned by friends, she will receive a notification, with which she can go directly into the comment to check the mentioning.