Topic 31: Distributed Systems (CDN + Storage) Design

Team Members

Yani Jin (yanijin@usc.edu)

Xinyi Cai (caixinyi@usc.edu)

Ryan Afranji ([afranji@usc.edu](mailto:afranji@usc.edu))

Database?

Web server

Followed this https://www.youtube.com/watch?v=ivstVNUKAW8 video series to set up the web server. Teaches how to use express generate the template of a web server and all the associated files. So followed its directions to generate the template and modify it to support the cassandra-driver. Next, I modified the index.js and index.jade page to make the queries needed for our own database, package those results into a data structure and then send those to the index.jade file I modified to display our website the way we wanted it to and populated it with values from that data structure.

index.js can be found under Web Server/project/routes/index.js

index.jade can be found under Web Server/project/views/index.jade

Learned for to make a list in jade and handle for loops from this site:

https://naltatis.github.io/jade-syntax-docs/

Learned cassandra-driver and cql commands from these sites :

https://github.com/datastax/nodejs-driver

https://docs.datastax.com/en/cql-oss/3.3/cql/cql\_reference/cqlSelect.html

Load balancer

Reference:

1. <https://www.nginx.com/blog/choosing-nginx-plus-load-balancing-techniques/>
2. <https://www.nginx.com/resources/glossary/round-robin-load-balancing/>

Implementation:

The load\_balancer.conf file is the nginx configuration file of load balancer. Here we define the listening port, the backend servers and their public ip addresses and the load balancing algorithms. The load balancer is currently running on a local virtual machine. We use Ngrok to provide a public ip address for the load balancer.

Proxy cache