Git URL: <https://github.ccs.neu.edu/yihtian/CS6650/tree/master/Assignment2>

Design:

On the server side, there are majorly 3 parts: servlets, data access layer, and database connector.

Connector part provides information of database and thus create new connection to database.

For database part, 2 schemas are designed. LiftRide schema is the database to record information such as skierId, dayId and etc., Stat schema is the database to record respond time for each request. SkierDAO provide 2 functions, first allow new data from a request to be inserted into database as a query, the second allows client to get vertical information from existed data by providing a skierId and a dayId. StatsDAO also provides 2 functions that both allow client to write in respond time information and check the mean and max response time at runtime.

For servlets, SkierServlet doPost method accepts requests from client which sends the information of a lift ride, then insert info from request into database LiftRide through SkierDao then send back response. After sending back response, record the total response to process 1 request and insert that information into Stat database by using StatDao. SkierServlet doGet method accepts skierId and dayId from request and using these 2 things to lookup vertical information in existed LiftRide data by calling the get function in SkierDao.

For StatServlet, only doGet method is designed, when client sends a request to get information, the servlet look through Stat table and get the mean and max for current data and send back this information as a APIStats.

On the client side, the design is similar to assignment1. In a single thread, the thread will send required number of post requests, but at this time, if the thread is running in the third phase will be checked, if true then the thread will send out additional get request. Third phase information will be an input as Boolean. Then each phase creaetes required number of threads, and all phases will be executed in main function.

Results:

For single server:

A close up of text on a black background

Description automatically generatedA close up of text on a black background

Description automatically generated

A close up of text on a black background

Description automatically generatedA close up of text on a black background

Description automatically generated

For load balanced server:

A close up of text on a black background

Description automatically generatedA close up of text on a black background

Description automatically generated

A close up of text on a black background

Description automatically generated

A close up of text on a black background

Description automatically generated

Runtime Statistics:

Called get for 3 different times during run time, and got following information.

A close up of a logo

Description automatically generated

A close up of a device

Description automatically generated

A close up of a logo

Description automatically generated

Charts for single server:

Chart for load balanced server: