Catalog Server:

Catalog server exposes two operations, query and update, which have corresponding root URLs /query and /update. The arguments that are needed to perform an operation are passed along in URL. For example, to list all books that have topic of distributed systems, a request would be made to /query/distributed\_system and to search item 2, a request would be made to /query/2. A json object is returned as the query response. Examples of the json responses are shown below,

/query/distributed\_system

{"items":{"How to get a good grade in 677 in 20 minutes a day":1,"RPCs for Dummies”:2}}

/query/3

{"Xen and the Art of Surviving Graduate School”:{"COST":150.0,"QUANTITY":13}}

Update service works in a similar manner. The service supports three operations on two fields. Three operations are “increase”, “decrease” and “set”. Two fields are “cost” and “quantity. The update service only checks for validity of the operations and fields to update. It doesn’t check if a request would result in an illegal update such as decrement a 0, which could occur when queries and updates are interleaved.

In the cases where quantity or/and cost need to be retrieved or updated, a lock specific to that row of data has to be obtained first. This makes sure the data would be in consistent states and it prevents the aforementioned illegal update from happening.

Frontend:

Frontend has three operations search(topic), look(item\_number) and buy(item\_number). Each operation redirect the client request to the responsible server and formats the json response in a readable way before it returns a response that contains string data to the client.