

Protocol

Client Module

Because we chose to work in python, and the types and "out parameter" conventions differ, we describe the functions of our importable module here.

kv739_init(server) Takes a string indicating the server in the format "host:port" and initializes client code. Returns 0 on success or -1 on failure.

kv739_get(key) Takes a key string and requests the value for that key from the server. If the key is present, return a tuple (0, value). If not present, return tuple (1, <undefined>). If there is a failure, return tuple (-1, <undefined>).

kv739_put(key, value) Takes strings key and value and (1) requests the old value for key and (2) requests the server to store the given value for the key. Returns tuple (0, old_value) on success if there was an old value. Returns tuple (1, <undefined>) on success otherwise. Returns (-1, <undefined>) on failure.

kv739_delete(key) Takes a key string and (1) requests the old value for key and (2) requests the server delete its [key, value] mapping for the given key. Returns tuple (0, old_value) on success if there was a mapping. Returns tuple (1, <undefined>) on success if the key had no value on the server. Returns (-1, <undefined>) on failure.

Network Protocol

We defined our protocol using HTTP GET, PUT, and DELETE. The requests, responses, and semantics are defined below.

GET Request	
URL	<HOST>:<PORT>/?key=<KEY>
Headers	
Body	
Response	
Code	200 if exists, 404 if not, 500 if error
Body	JSON {value: <VALUE>, *errors: [<ERROR STRINGS>]}
Description	
Server gets the key to lookup from query string. If a value is found, 200 is returned and the value field of the returned JSON object is set. If a value is not found, 404 is returned. If problem(s) are encountered, 500 is returned and error strings may be returned in the JSON object.	

PUT Request	
URL	<HOST>:<PORT>/
Headers	Content-Type: application/x-www-form-urlencoded
Body	urlencoded(key=<KEY>&value=<VALUE>)
Response	
Code	200 if old value, 201 if newly created, 500 if error
Body	JSON {old_value: <VALUE>, *errors: [<ERROR STRINGS>]}
Description	
Server gets key and new value from the body of the request. If the key already has a mapping, 200 is returned and the old_value field of the JSON object is set. If not, 201 is returned. If problem(s) are encountered, 500 is returned and error strings may be returned in the JSON object.	

DELETE Request	
URL	<HOST>:<PORT>/
Headers	Content-Type: application/x-www-form-urlencoded
Body	urlencoded(key=<KEY>)
Response	
Code	200 if success, 500 if error
Body	JSON {old_value: <VALUE>, *errors: [<ERROR STRINGS>]}
Description	
Server gets key to delete from the body of the request. On success return 200, and if the key had a mapping, the old_value field of the JSON object is set. If problem(s) are encountered, 500 is returned and error strings may be returned in the JSON object.	