

# Assignment 1 Report

## In-Memory Distributed Queue

Lorem Ipsum

## Persistent Distributed Queue

Lorem Ipsum

## Client Library (LogQueueSDK)

LogQueueSDK is a client library developed to provide an easy to use interface accessing the functionalities and engaging with the distributed queue server. Producers and Consumers can use this library to interact with the server while ignoring the underlying request mechanism. All necessary utility functions were implemented and classified into Producer and Consumer classes.

The library consists of a base class “**Client**” that implements the common functionalities between Producers and Consumers such as “**list\_topics**”. **Producer** class and **Consumer** class inherit from Client and implement their own required methods. A producer would instantiate an object of the Producer class using two parameters - *address of the broker* and *list of topics to register for* (none by default). Similarly, a consumer would instantiate an object of the Consumer class using the same parameters. The clients would then continue using these objects to access the functionalities of the server. This provides for an easy to use and intuitive interface with the server.

Producer and Consumer also have a member variable *topic\_id\_map* which is a dictionary of topics registered to and correspondingly allocated ID. Following is a brief summary of classes and their methods.

### **Client**

- *create\_topic : returns bool*
- *list\_topics : returns list*

### **Producer**

- *register : returns int (producer\_id)*
- *send : returns bool*
- *can\_send : returns bool*
- *stop : stops*

### **Consumer**

- *register : returns int (consumer\_id)*
- *dequeue : returns string*
- *size : returns int*
- *get\_next : iterates over all topics and dequeues the messages*
- *stop : stops*

## Tests and Results