**Component: Orchestration Layer**

**Module Name: Service Broker**

**Author: Tyler Benson**

**Reviewers: Mariam Topchyan, Mandy Lin, Gabrielle Cornelius, Even Pham**

|  |  |  |
| --- | --- | --- |
| Inputs | Process | Outputs |
| Service Code (args[1]) | Looks up service code in service.txt to find the file path to call the correct module and passes ParamList to that module. | Return value from service module called |
| ParamList (args[0]) |  |
|  |  |

**Inputs:**

Service Code which Service Broker uses to look up which module to call.

ParamList is a string of useful information delimited by commas that subsequent modules will interpret and use.

**Process Description:**

Service broker looks up the service code in service.txt to find a corresponding file path that it will use to call the desired module. It will pass ParamList to that module.

If service broker does not find the service in Service.txt, it calls Error module and passes “703” as the paramlist in args[0].

**Outputs**

ParamList as args[0]

**APIs/Objects:**

**Pseudocode:**

Read the contents of Service.txt into an ArrayList.

Loop through the ArrayList and if any String contains the ServiceCode in args[1], save the file path to a local string and update the flag to indicate the service was found.

If service was found, create a new process using that filepath and pass along args[0].

Read in the child process input stream and print it to return it to whichever module called Service Broker.

If service is not found, call Error module and pass it “703” in args[0].

If Service.txt is not found, call Error module and pass it “404” in args[0].