

ECE 4564 Assignment 2 Report

System Syntax

Client Operation:

The client can be run in the following manner:

```
client.py -proc <Processor IP Addr> -action <Action> -book <Book Info>
```

The IP address following the -proc flag specifies the IP address for the intermediary between the client and the storage server, known as the processor. The client and the processor will communicate via RabbitMQ Remote Procedure Call. The argument following the -action flag can be either ADD, BUY, SELL, DELETE, COUNT, or LIST. This determines what operation will be performed in the database on the next piece of information passed to the program, the Book Info.

Processor Operation:

The processor can be run in the following manner:

```
processor.py -storage <Storage Bluetooth Mac Addr> -p <Storage Port Number> -z  
<Socket Size>
```

As described above, the processor acts as an intermediate step between the client and the storage server, Therefore it requires the Bluetooth Mac address and port number of the storage, as the processor and storage will communicate via Bluetooth. The Mac address follows the -storage flag, and the storage port number follows the -p flag. Finally, a socket size can be designated following the -z flag according to the needs of the program.

Storage Operation:

The storage can be run in the following manner:

```
storage.py -p <Port Number> -b <Backlog> -z <Socket Size>
```

Following the -p flag, the port number to listen for communication from the processor can be specified. No Mac address argument is needed here, because the processor will initiate the communication between the two. A backlog size can be specified following the -b flag, and the socket size can be specified following the -z flag.

Libraries

Client:

Pika - Python library for communication using RabbitMQ.

Processor:

Pybluez - Library communicating with Bluetooth using sockets.

Server:

Pymongo - Library for interacting with a MongoDB instance. In this case its functions were encapsulated in the MongoDB class.

In addition to the libraries listed above, Python built-in libraries such as time, threading, json, and sys were used.

Contributions

Michael Pocta: Set up threading on Storage.py to run the bluetooth communication and LED blink at the same time. Created LED class in LED.py file representing varieties in the database. Set up the bluetooth communication for the Storage.py and handled the communication with the processor.

Kish Jadhav: Set up the rabbitMQ for sending data between client and processor. Created an argument parser, client class and processor class. Set up bluetooth communication for the processor.