COMP90015 Distributed Systems
Assignment 1: Multi-threaded Dictionary Server

ZHAOFENG QIU 1101584
The University of Melbourne

September 5, 2019

1 Problem Context

In this project, a multi-threaded dictionary client-server system is designed and implemnted using a client-server architechture. The server of the system allows concurrent clients to modify or query the remote dictionary by using thread-per-request architecture in the server. The system provite a reliable communication between the server and clients by using TCP protocol. Also, JSON is used in the system for providing a message exchange protocol between clients and server. As for failure handling, errors including I/O errors, Network Connection error and parameters errors, are properly manageed on both the server and the client side. Also, illegal request(such as adding a word that is already in the dictionary, removing or querying a none-exit word and adding a word without meaning) from clients can be detected and handled by the server properly. Moreover, the server can handle concurrent requests at the same time and edit the data in a corect way.

2 System Components

2.1 Server

The server is implemented using thread-per-request architecture and can detect and handle multiple failure. For every new request sent by clients, the server would create a new thread to handle it and give it a response. Since the server is using multi-thread technology, it can handle requests from different client concurrently. However, when it comes to data modification, because the relative functions provided by dictionary controller are synchronized, there will be no ambiguity on modifying the data in the dictionary. According to specific contexts, the server would handle requests in different ways and send back different states(which indicate the specific implementation) and feedbacks to clients. For better exchanging messages with clients, every response would be packed in JSON format before sending back to clients. In addition, the server provides a GUI for logging operations and showing information about the server.

2.2 Client

The client is mainly composed of two parts: the Client's Controller and the Client's GUI. It can detect and handle failures such as problem of Network communication problems and Incorrect parameter input problems. The Client's Controller part is responsible for handling user's operational requirements captured by the GUI. For each ADD, DELETE, QUERY operations given by user, the controller would establish a new TCP connection with the server and would close it when the controller get the response of it . In this case, the client would not check whether the server is running before processing the three operations given above. The GUI implemented with Swing can provide visual interfaces to user. In addition to satisfying the user's query and modification operations, it also prompts the user how to use the client, such as asking them to enter a word before clicking ADD button or REMOVE button, which can improve user's experience.

3