Poem of the Day Specification

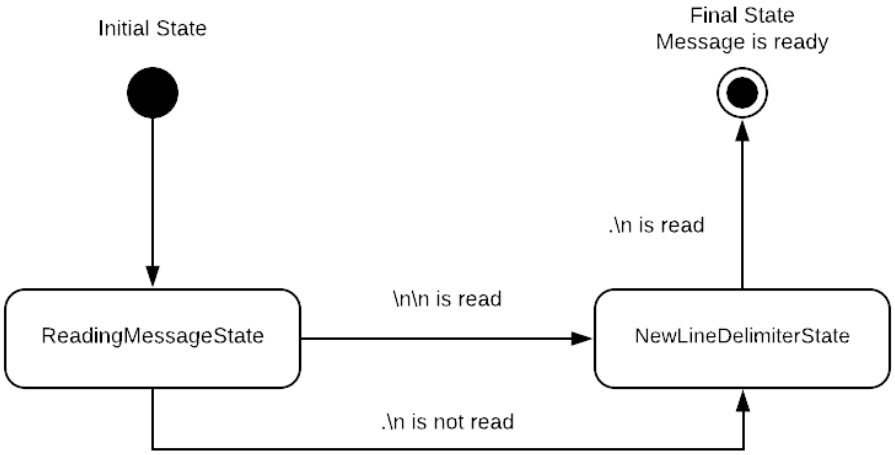
Prepared by: George Antonious (3364768)

Messaging Protocol

Messages between the server and and client are sent in their plain text form followed by two newline characters then by a period character followed by another new line. The **"`\n\n.\n`"** sequence indicates the end of the message. The following is an example of a poem selection message that can be sent to the Poem of the Day server using telnet:

```
2\n\n.\n
```

The messaging protocol will decode the above message as the number 2. This protocol utilizes state machine to handle parsing the raw message data. The state machine has two states one that reads line by line from the client socket to build the message and one to handle when the first section of the end of message sequence is read. When the first two line characters are read the state machine will enter the `NewLineDelimiterState` which will expect the `.\n` sequence to mark the end of the message. If this client sends something else during this state it will add the the new line characters read previously back to he raw message as it's not part of the end of message sequence. This state machine is outlined below:



Server-Client State Machine

When a client connects to the server the server starts a state machine to handle the client's request. This state machine involves the following states:

- `WelcomeState`: This state sends the user the available poems
- `WaitForPoemSelectionState`: This state waits for the user to select a poem
- `DeliverPoemState`: This state sends the selected poem to the user
- `ErrorState`: This state sends an error message to the user

This state machine is outlined in the following diagram: