# Results of Stress Testing

The test Framework is made to support the same message passing that is done by the system to create and vote on polls but without using GUIs to generate the messages. The framework supports a voter object that sends messages the same way a voter client does but with no GUI. The voter also supports randomly selecting an option from the given options.

Results: after running several load tests the system is not dropping any votes. When 100 votes are created then all 100 votes are recorded.

The table below shows the response time of the admin when sending the pause or end commands. The values in the table is how many votes out of 100 where recorded before the poll was paused or ended.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | |  | | |
| Lag in PollingSystem During AminClient Operations | | | | | |
|  | Number transmitted | | | | |
| Trial# | Paused | | | End | |
| 1 | 7 | | | 10 | |
| 2 | 10 | | | 38 | |
| 3 | 30 | | | 5 | |
| 4 | 15 | | | 37 | |
| 5 | 26 | | | 17 | |
|  | |  |  | |

As we can see from the graph above, there was a variable latency in the PollingSystem. What specifically happened is that with 100 votes occurring simultaneously (within the limitations of the lab computers), the system was slowed down enough that we were able to pause and end the poll before votes were recorded. However, without pausing or ending the poll, all votes were recorded.