

## Short Use Case Descriptions

After review of Project Vision's functional and NFRs, the following use cases have been identified.

### Actor Key

UAS = User Application Server

Dev = Developer

| Use Case Name                  | Precondition                                   | Actor | Need   | Do Something                   | Basic Intent/Goal  |
|--------------------------------|--|-------|--|--------------------------------|--|
| Join Cluster                   | Has information about the cluster              | UAS   | Join the cluster   | Join Cluster                   | Become a member of the cluster                               |
| Run Cluster                    | Join Cluster use case has run                  | UAS   | Participate in cluster actions                                       | Run Cluster                    | Start participating in the cluster                           |
| Receive Committed Entries      | N/A  | UAS   | Bring itself up to date  | Receive Committed Entries      | Can maintain consensus                                       |
| Append Entry                   | N/A  | UAS   | To add a message into consensus                                      | Append Entry                   | Message gets committed into the consensus                    |
| Receive Notice of Entry Commit | N/A  | UAS   | To know if the message they requested to be committed, was committed | Receive Notice of Entry Commit | So that they may update the running UAS state                |
| Receive Start Server           | The current node should be running a UAS       | UAS   | To be running a UAS when required                                    | Receive Start Server           | They may go ahead and start up their UAS to provide services |
| Receive Stop Server            | The current node should stop running their UAS | UAS   | To stop running a UAS when required                                  | Receive Stop Server            | They stop acting as a UAS and providing service              |

|                             |                                       |     |  |                             |   |
|-----------------------------|---------------------------------------|-----|--|-----------------------------|---|
| Respond with Server Fitness | UAS implementing own fitness function | UAS | Wants to respond with its fitness for being the UAS  | Respond with Server Fitness | The most fit node may be identified to run the UAS  |
| Read Entry Value            | N/A                                   | UAS | Read log entries that have been committed  | Read Entry Value            | Can reference the committed data  |
| Send Network Message        | Offloading node communication to UAS  | UAS | Send a message offloaded to it from the underlying consensus algorithm                       | Send Network Message        | The underlying consensus algorithm can still communicate without relying on it's own networking stack |
| Receive Network Message     | Offloading node communication to UAS  | UAS | Forward a message offloaded onto it's networking stack to the underlying consensus algorithm | Receive Network Message     | The underlying consensus algorithm can still communicate without relying on it's own networking stack |
| Confirm Identity            | N/A                                   | UAS | Confirm the identity a new node communicating for the first time                             | Confirm Identity            | Establish security through trust, and ensure they cannot be MITMd                                     |
| Read Logs                   | N/A                                   | Dev | Understand what the underlying consensus algorithm is doing, perhaps tracking a bug          | Read Logs                   | Understand what the underlying consensus algorithm is doing   |