**Proof of Technology**

**Summary:** This is documentation provides readers with a general overview of our project along with a simple working demo

**Prerequisite:** {SEE: Skupper\_Namespace\_Connection}

Ensure the following function as expected: MetalLB on the Minikube cluster.

**Log**

| Change Made | Signature |
| --- | --- |
| Created Initial Docuement | AG, SP |
|  |  |
|  |  |
|  |  |

Code/Process Block 1

Light description of step

| > Block to include code or commands |
| --- |

Code/Process Block 2

Light description of step

| > Block to include code or commands |
| --- |

Code/Process Block 3

Light description of step

| > Block to include code or commands |
| --- |

Run the following commands in sequential order

Git Clone

| > git clone git@github.com:GemaSoftware/lowCarbonBlockchain.git |
| --- |

Change Directory

| > cd lowCarbonBlockchain/rolesetup |
| --- |

Apply serviceaccount.yaml first to the namespace

| > kubectl apply -f serviceacc.yaml --namespace ns1  > kubectl apply -f serviceacc.yaml --namespace ns2 |
| --- |

Then apply the role binder to both namespaces

| > kubectl apply -f rolebinder.yaml --namespace ns1  > kubectl apply -f rolebinder.yaml --namespace ns2 |
| --- |

Change Directory

| > cd ../backEndListener |
| --- |

Apply

| > kubectl apply -f backend-listener-1.yaml --namespace ns1  > kubectl apply -f backend-listener-2.yaml --namespace ns2 |
| --- |

Change Directory

| > cd ../frontEndListener |
| --- |

Apply

| > kubectl create deployment frontend --image gemajlia/basicwebcontainer:latest --namespace ns1  > kubectl create deployment frontend --image gemajlia/basicwebcontainer:latest --namespace ns2 |
| --- |

Expose port 8080

| > kubectl expose deployment/frontend --port 8080 --target-port 80 --type LoadBalancer --namespace ns1 |
| --- |

Expose the web port

| > kubectl expose deployment/frontend --port 8080 --target-port 80 --type LoadBalancer --namespace ns1 |
| --- |

Ensure that the frontend service is listening properly

You will see an external IP connected to the frontend with port 8080

| > kubectl get services --namespace ns1 |
| --- |

Using the External IP shown in the above command. In our case, it is 172.19.100.4. Open your browser to http://<external-ip>:8080

| You will see something like this    Enter the number of heads to get in a row as the input  Once clicking submit, you will receive an answer from one of the other namespaces' backend giving you the total number of flips it took to get X amount of heads in a row |
| --- |