Sr. No	Activity	Sub-task	Man Days	Owner	SatrtDate	End Date	Acceptance Criteria	Remarks
1	Evaluation and implementation of Load balancer and API Gateway.	- Implemenation of Metal-LB for allocating IP Addres in case of bare metal clusters.	2	Darshil	21June	2August	**Exposed URLs in CSP** 10.151.33.100:30580/som/index.zul - SOM 10.151.33.100:31088/nbreq/execute - NB Request Service 10.151.33.100:<port>/<subURI> - Activemq portal, kiali, kibana, grafana etc dashboards	Need to test
		- Integrate metal lb with ambassdor	1					
		- Deploy Ambassador for below functions:					**Features required**	
		- API Gateway.	1					
		- Ingress Controller.	1				- K8s Aware Gateway should be Kubernetes aware i.e. it should be part of K8s cluster and should communicate	
		- Service discovery.	1					Yes
		- Integration with application.	1				- Load Balancing - Support for Round Robin Algorithm Gateway should support Load Balancing of incoming request in Round Robin fashion.	Need to test
							- support for HTTP and HTTPS (including SSL Termination Proxy) Gateway should accept request on HTTP or HTTPS protocol. It should also have capability to work SOM Portal should be able to be accessed on HTTPS protocol.	
							- Routing - Support for Host, Header and Path (URI) based routing rules Gateway should be able to route the request to SOM or NBRequest MS based on URI.	
							- Resiliency - Support for Retry, Timeout, Rate Limiting (Overload protection) and Circuit Breaking Gateway should retry for a request URL for preconfigured amount of time. Gateway should timeout a request after preconfigured amount of time. Gateway should not allow concurrent request after preconfigured threshold is reached. Gateway should be able to stop request to a particular URL when a preconfigured condition is met.	
							- Observability - Record metrics on Prometheus (via Istio integration if possible) Gateway should be able to integrate with Istio if possible and share the metrics to Prometheus via Is If integration with Istio is not possible then it should be able to send the metrics to Istio's Prometheus	
							- Distributed Tracing - Record tracing on Jaeger Gateway should be part of the HTTP request tracing shown in Jaeger.	
							- Scaling - Support for Auto Scaling of Deployment in K8s Auto Scaling of Gateway should be possible based on CPU and RAM metrics.	
							- Hardware Requirment - Not more then 2 vCPU and 4 GB RAM required per deployment	
							- Centralized Logging - Support for ELK Gateway logs should be stored in common platform ELK.	
							- Packaging - Support for HELM Chart Deployment of Gateway should be possible using HELM Chart.	
		- Testing and verification.	1					Features availabe need to be tested
2	Integration of Platform components with Grafana and Prometheus for metrics and KPI mgmt	- Installation of prometheus and grafana.	1	Zaid	21June	1 Aug		
		- Grafana dashboards for below components: K8S, Container/POD CPU & Memory, API Gateway, ActiveMQ, Hazelcast, Tomcat JVM, Sprint boot	5				Springboot covered by Safvan	
		- Alerts to be configured in case of pod failure/restart or autoscaling., -Resource Utilization - CPU, RAM, Thread Monitoring, JVM etc.,Alert on utilization violation, load monitoring	2					
		- Fetch and prepare dashboards for application specific metrics (i.e. TPS)	5				To be covered by CSP Team	
		-Services / Component Availability	2					
		-Services Communication	2					
		-Pod failure (service wise) alert	1					
		-Pod scale (service wise) alert	2					
		-Overload protection - Queue level restriction and REST level restriction report	2					
		-Response / Request-time monitoring	1				To be covered by CSP Team	
		-Order-wise Status Count-Frequency, drill down L1 & L2	2				To be covered by CSP Team	
		-SLA Analysis - Order Type-wise - SLA violation alert	2				To be covered by CSP Team	
		-Batch processing	2				To be covered by CSP Team	
		- Deliverable creation	2					
		- Testing and verification	1					
		-Queue (Message count) Monitoring-Pendency Alert						
		-DB Connectivity Pool						
		-Long Running queries						
		-Queue / Bridge Availability						
		-DB calls count (S/I/U/D)						
3	Monitoring with ELK	- Centralised logging for application with ELK	2	Shivani	3July	17-July		
		- Dashboards for visualising application specific details.	2	Shivani				
		- Gather k8s cluster logs.	2	Shivani				
		- Custom filebeat module for csp with docker image						
		- Performance tunning of ES cluster						
		- ELK Cluster setup						
		- Helm Charts						
		- Index Purging						
		- Installation Document						
4	Postgres Containerisation	- Deploy postgres on k8s cluster	1	Shivani	21June	19July		
		- Master and replica architecture using statefulsets	2					
		- Failover handling : If master/primary instance fails then replica to be promoted as primary	3					
		- Loadbalancing between primary and standby pods.	3					
		- R/W queries to be loadbalanced	2					
		- Deliverables	4					
		- Testing and verification	2					
5	Overall deployment through Helm charts	Prepare helm charts for all the deliverables	3	Darshil and Za	10July	13July		
6	Understanding sessions required	CSP product understanding	2	CSP team				
		Prometheus and Grafana	1	Chintan				
		ISTIO	1	Vijay/Safvan				
		Microservices	2	Vijay/Safvan				
		API Gateway-Throttling	1	Vijay				
		PostGres	1	CSP team				

Sr. No	Technology	Primary Owner(s)
1	Kubernetes	Darshil, Chintan, Shivani, Zaid,Ramakrishna
2	Prometheus & Grafana	Zaid,Ramakrishna
3	ELK	Zaid,Ramakrishna
4	Istio	Chintan, Darshil, Shivani
5	CentOs8	Chintan
6	OpenStack	Chintan,Ramakrishna
7	Gradle	Shivani, Darshil