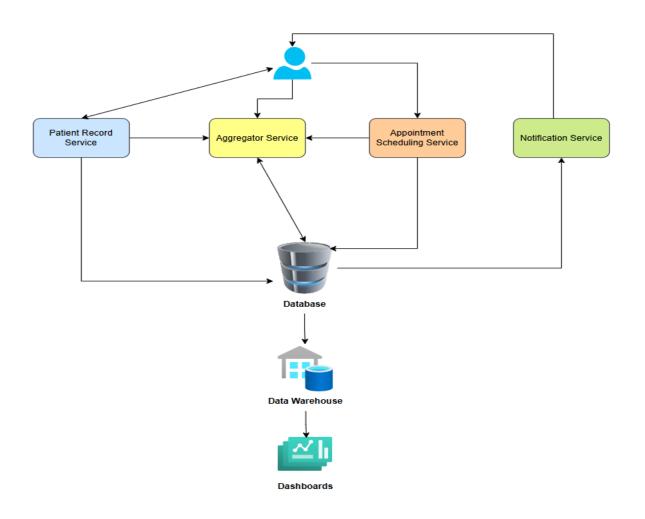
HealthSync Solution - Runbook

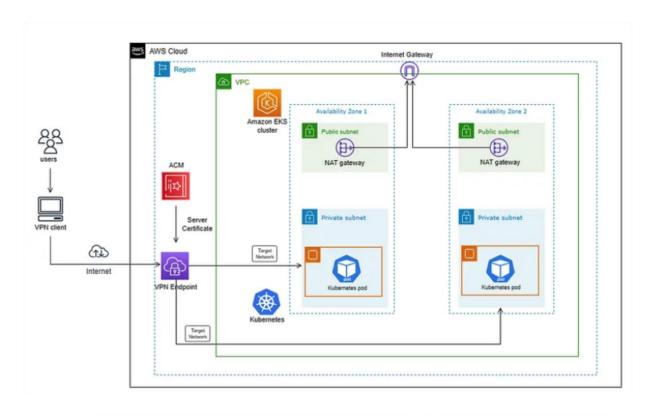
Runbook Name	HealthSync Solution	
Runbook Description	Patient Record Service, Appointment Scheduling	
	Service, Notification Service, and Aggregator	
	Service Included as micro services	
Owner	Hasal Chandrasiri	
Version	1.0	
Version Date	29/12/2024	

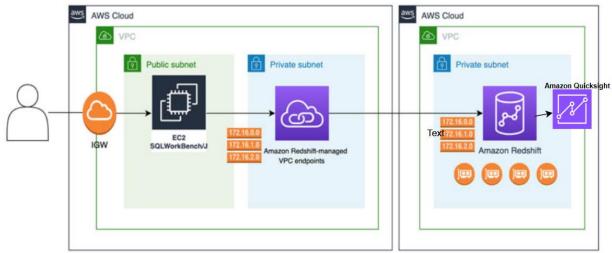
Architectures

Solution Architecture

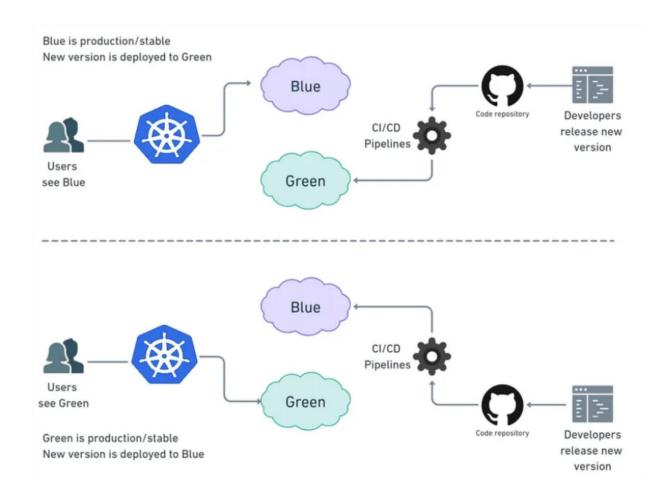


Deployment Architecture





Continuous Integration / Continuous Deployment Process



Support Contacts

Expertise Level	Team	Team lead	Contact Info
Developer	Hasal Chandrasiri	-	Hasal.20241029@iit.ac.lk
Product Owner	Hasal Chandrasiri	-	<u>Hasal.20241029@iit.ac.lk</u>

Process

	Step Instruction	Execution location	Run environments	Run conditions	Run instructions	Documentation
	s					
1	Configure AWS CLI	Local Machine	Local Machine CLI		aws configure	https://docs.aws.ama zon.com/cli/latest/us erguide/cli-chap- configure.html
2	Setup ektcl	Local Machine	Local Machine CLI	Run as administrato r	choco install –y eksctl	
3	Install Kubernetes CLI	Local Machine	Local Machine CLI	Run as administrato r	choco install -y kubernetes-cli	
4	Create EKS cluster	AWS	AWS Configured CLI in local machine		eksctl create clustername HealthSyncCluste rregion us-east- 1nodes 2 node-type t3.small version 1.30	
5	Create RDS for microservices	AWS EKS Cluster VPC	AWS	MySQL RDS on same VPC as the EKS Cluster	After creating the RDS, create the healthsync database and create tables required by each service	
6	Update connection details for patient record service	GitHub	Local Machine		Clone the repository, change the connection string, commit and merge to main branch using a pull request	https://github.com/ HasalChandrasiri/ Patient-Record- Service.git
7	Update connection details for notification service	GitHub	Local Machine		Clone the repository, change the connection string, commit and merge to main branch using a pull request	https://github.com/ HasalChandrasiri/ Notification- Service.git
8	Update connection details for appointment scheduling service	GitHub	Local Machine		Clone the repository, change the connection string, commit and merge to main	https://github.com/ HasalChandrasiri/ Appointment- Scheduling- Service.git

				branch using a	
				pull request	
9	Update	GitHub	Local Machine	Clone the	https://github.com/
	connection			repository, change	HasalChandrasiri/
	details for			the connection	Aggregator-
	aggregation			string, commit	Service.git
	service			and merge to main	
				branch using a	
1.0	~			pull request	
10	Setup	AWS	AWS	Set the	
	Environmenta		Configured CLI	DB_HOST,	
	1 variables for		in local	DB_NAME,	
	secrets		machine	DB_USER and	
				DB_PASSWORD	
1.1	G . 4	C'II 1	T 1 M1.*	in the secrets	
11	Setup Environmenta	GitHub	Local Machine	Set the KUBECONFIG	
	l variables for			in the secrets	
	KUBECONF			in the secrets	
	IG				
12	Deploy	AWS	AWS	kubectl apply -f	
12	service of	21,000	Configured	patient-record-	
	patient record		CLI in local	service-	
	service			blue/deployment	
	561 (166		machine	.yaml	
				kubectl apply -f	
				patient-record-	
				service-	
				green/deployme	
				nt.yaml	
13	Deploy	AWS	AWS	kubectl apply -f	
13	service of	AWS		notification-	
	notification		Configured	service-	
	service		CLI in local		
	Scrvice		machine	blue/deployment .yaml	
				kubectl apply -f	
				notification-	
				service-	
				green/deployme	
				nt.yaml	
14	Deploy	AWS	AWS	kubectl apply -f	
1-7	service of	11115	Configured	appointment-	
	appointment		CLI in local	scheduling-	
	scheduling			service-	
	service		machine	blue/deployment	
				.yaml	
				kubectl apply -f	
				appointment-	
				scheduling-	
				scheduling-	

			T T		
				service-	
				green/deployme	
				nt.yaml	
15	Deploy	AWS	AWS	kubectl apply -f	
	service of		Configured	aggregator-	
	aggregator		CLI in local	service-	
	service		machine	blue/deployment	
			macinic	.yaml	
				kubectl apply -f	
				aggregator-	
				service-	
				green/deployme	
				nt.yaml	
16	To monitor	AWS	AWS	kubectl port-	
	get the details		Configured CLI	forward -n	
	from		in local	monitoring	
	Prometheus		machine	svc/prometheus-	
				kube-prometheus-	
				prometheus 9090	
				http://localhost:90	
				<u>90</u>	
17	Monitor	AWS	AWS	kubectl port-	
	Health of		Configured CLI	forward -n	
	EKS cluster		in local	monitoring	
			machine	svc/prometheus-	
				grafana 3000:80	
				http://localhost:3	
				000	
18	i		· · · · · · · · · · · · · · · · · · ·		
10	Access	AWS	AWS		https://us-east-
		AWS			https://us-east- 1.quicksight.aws.am
	analytics data	AWS	AWS Quicksight		1.quicksight.aws.am
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse
	analytics data	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa-
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be-
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa-
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be- 9926ad20cfc2
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be- 9926ad20cfc2 https://us-east-
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be- 9926ad20cfc2 https://us-east- 1.quicksight.aws.am
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be- 9926ad20cfc2 https://us-east- 1.quicksight.aws.am azon.com/sn/analyse
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be- 9926ad20cfc2 https://us-east- 1.quicksight.aws.am azon.com/sn/analyse s/515c5e1d-d7a3-
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be- 9926ad20cfc2 https://us-east- 1.quicksight.aws.am azon.com/sn/analyse s/515c5e1d-d7a3- 41f0-a4af-
	analytics data using	AWS			1.quicksight.aws.am azon.com/sn/analyse s/06a61382-abfa- 4936-a9be- 9926ad20cfc2 https://us-east- 1.quicksight.aws.am azon.com/sn/analyse s/515c5e1d-d7a3-