

## Capstone Final Report PGPCC-July-19 Team 11

*Capstone Final report showcases the final implementation of the project. Challenges you faced in the implementation and your learnings from this whole exercise. The write up must adhere to the guidelines and should include the following.*

### Version History:

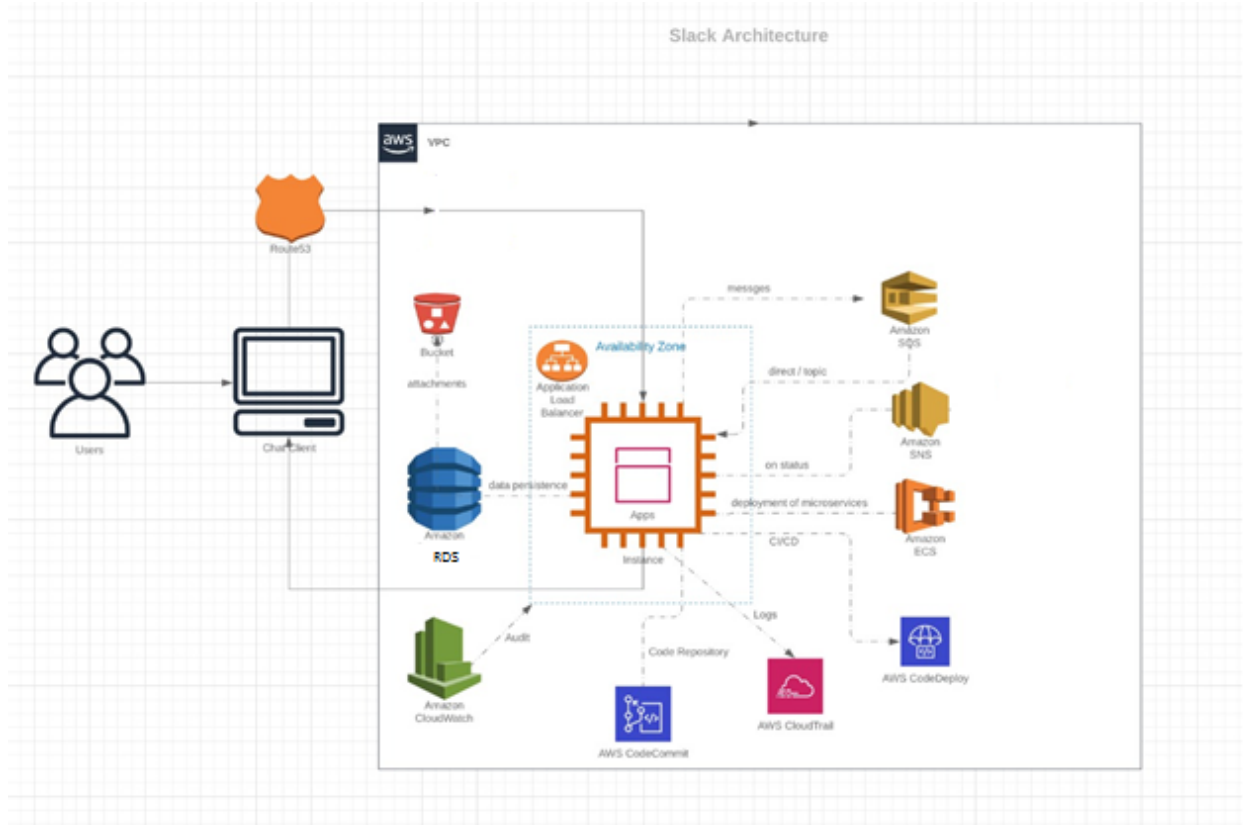
<b>Version</b>	<b>Date</b>	<b>Author</b>	<b>Key Changes</b>
1.0	18/12/2019	Rahul, Rex, Nagendra, Prakhar	<ul style="list-style-type: none"> <li>Fully searchable message history</li> <li>DevOps integration set</li> <li>File sharing</li> </ul>

### Document References:

<b>#</b>	<b>Title</b>	<b>Link</b>
1	Application server api	<a href="https://developers.mattermost.com/contribute/server/rest-api/">https://developers.mattermost.com/contribute/server/rest-api/</a>
2	Matter most docker file	<a href="https://github.com/mattermost/mattermost-docker">https://github.com/mattermost/mattermost-docker</a>

## Architecture:

- Instead of Amazon Dynamo DB, Amazon RDS is been used.
- Amazon API Gateway was not necessary, covered DNS details in Route53
- Amazon Lambda is been removed from architecture



## Implementation:

Screenshots (with brief description) of actual implementation of solution based on final architecture and its complete end to end deployment.

- Amazon EC2 instances for basic compute tasks
- Elastic Load Balancer to balance the workloads of EC2 instances
- Amazon S3 for Users file uploads and static data storage
- Amazon RDS for User data storage
- VPC to control security groups and firewall rules
- IAM to control user authentication and roles
- Route 53 for DNS management
- Amazon Cloud watch for monitoring all the services
- AWS CodeCommit for maintaining repository of Codebase

- *Amazon ECS that runs and scales containerized application*
- *Amazon Cloud trial for monitoring logs related to EC2 instances*
- *SNS for pushing notifications*

## Business Challenges & Technical Challenges:

*Configuring and integrating various AWS services with MatterMost was time consuming*

## Effort & Cost Impact (Actual vs Planned):

*As the numbers of users is less, we have gone with free tier except for ECS.*

*As MatterMost is implemented in GoLang and MySQL we had to use RDS MySQL instead of Dynamo DB. We did not use Code deploy.*

## Learnings from Capstone:

*Implementation of an entire software from scratch*

*Was able to work on various stacks like Development, Networking and Monitoring of the application*

*Cost sensitive*

## Team Roles and Responsibilities:

Clearly mention team's role & responsibilities in the below format.

Team Member Name	Contribution
Rex Roy	Architecture Design and implementation MatterMost application in the EC2 instance. Created and Integrated Amazon S3 bucket with the application Created and Integrated RDS - MySQL with the application Implemented Cloud Trail Implemented Elastic Container Services
Rahul	Worked on finalizing the collaboration tool Created VPC and subnets Created SNS and Lambda
Nagendra	Implemented Code commit Code maintenance
Prakhar	Implemented Route53