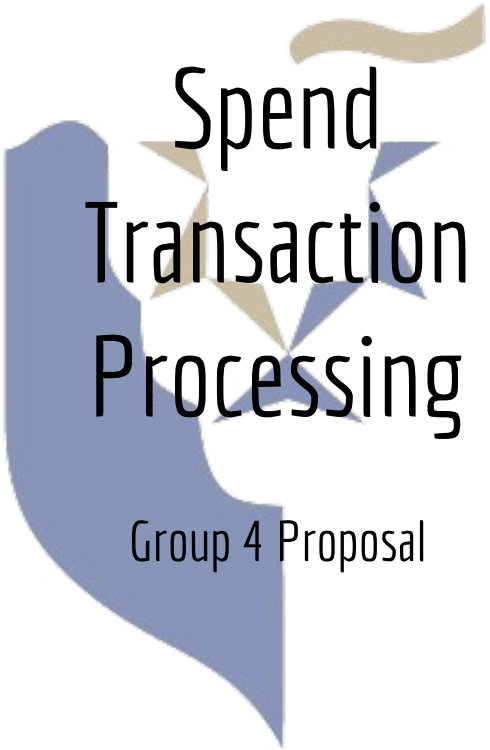


Spend Transaction Processing

Group 4 Proposal

Content

- Use Cases for Architectural Significant Requirements
- Solution View
- Quality Attributes
- Sequence Diagram
- Proposed Budgets (Production)

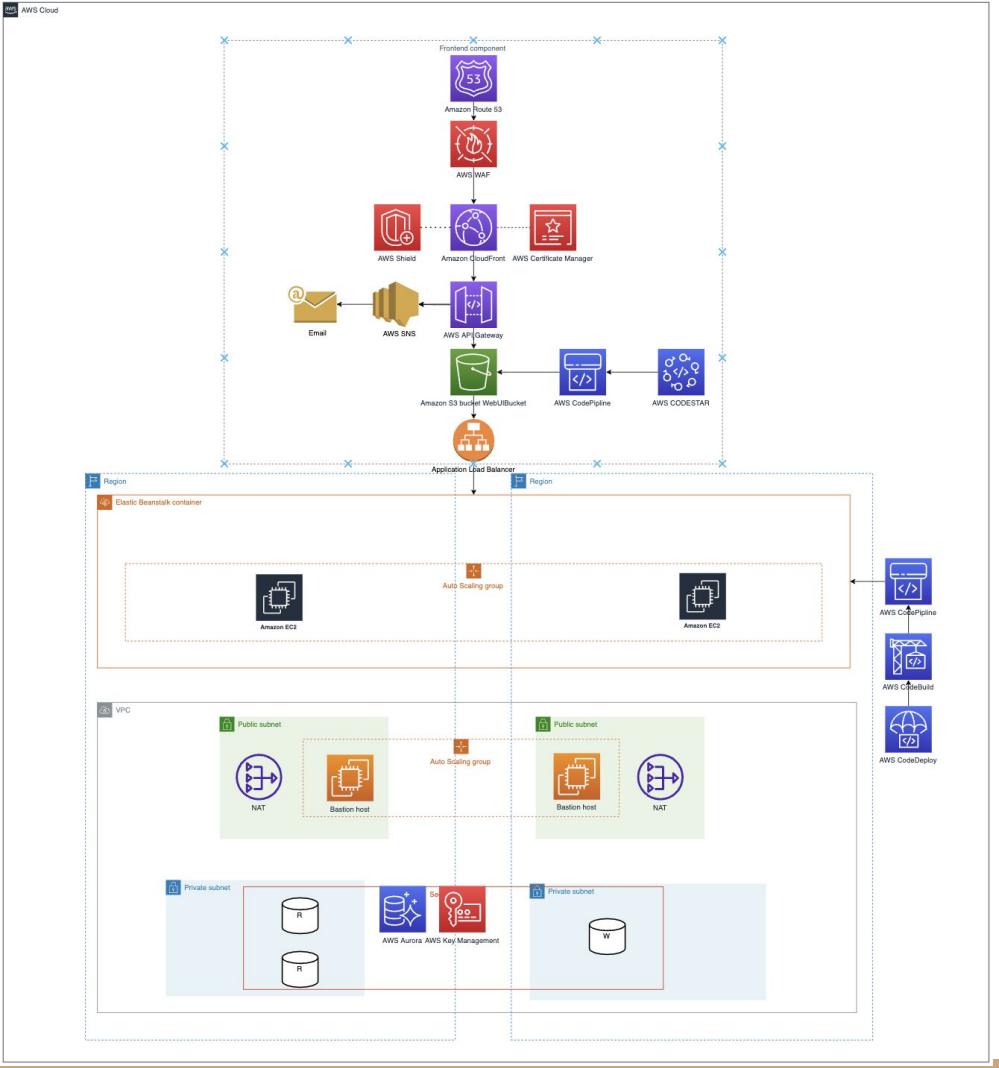
Use Cases

File Processing, Apply Exclusions & Benefits

- Upload Transaction Files (file transfer or API)
- Convert Currency
- Apply exclusions under a card program
- Convert transaction details to benefits
- Notify user of benefits
- View all transactions

Campaigns Management, Notify Users

- Create Promotional Campaigns
- Select Customers & Corresponding Transactions
- Notify user of benefits (email/ SMS)



SOLUTION VIEW

1. Frontend Components
2. Server Components
3. Database Components

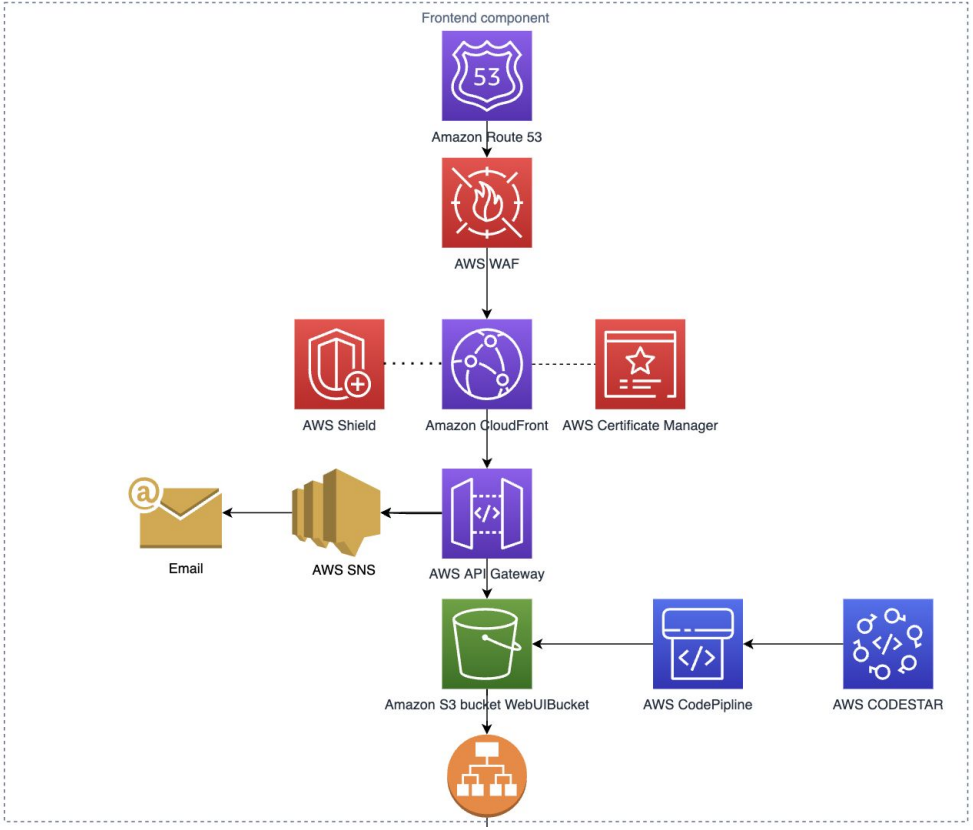
Monitoring:

Amazon CloudWatch

Costing:

AWS Budgets

AWS Cost Explorer



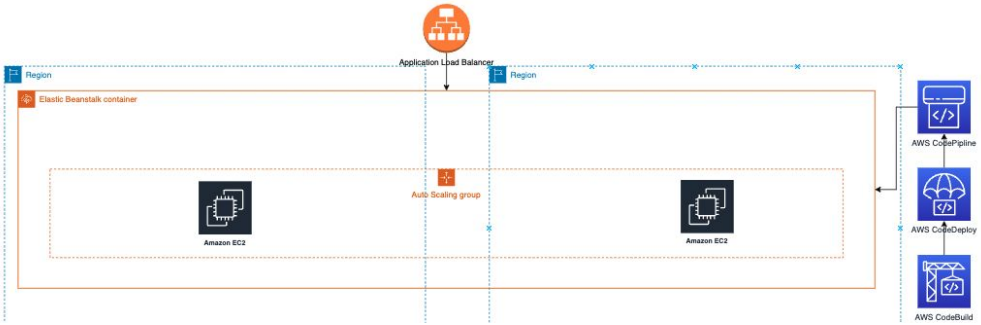
Frontend component

Static Web Pages are hosted on AWS S3 that scales at demand

AWS API Gateway: Manage RESTful API to expose HTTP endpoints & Notify Customer

AWS SNS for Email Notifications

Server component



EBS Deployed at regions:

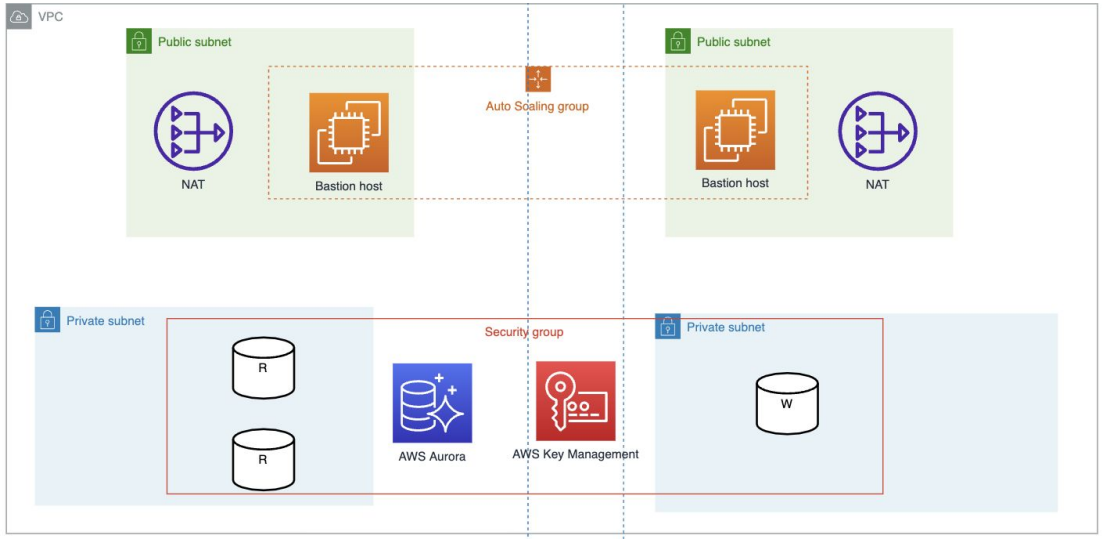
ap-northeast-2 (Seoul)
ap-southeast-1 (Singapore)

AWS CodePipeline to automate release pipelines

AWS AWSDeploy: automates software deployment

AWS CodeBuild to runs builds in preconfigured build

Database component



Bastion host in an Auto Scaling Group to allow access to Aurora DB Cluster in private Subnets

AWS Aurora replicas synchronously replicated across AZs

AWS Key Management Service to enable encryption at rest for Aurora DB Clusters

Quality Attributes

Performance

- Load balancers will equally distribute load to different servers available

- Ensure that there none of the servers will be overloaded, which can cause a reduction in performance and hence delays

Scalability

- AWS EC2 auto-scaling

- EC2 will detect a surge in traffic and auto scale up by adding EC2 instances. After traffic reduces, it will also auto scale down

- Aurora can scale up horizontally automatically to handle a much greater number of requests simultaneously. It will scale down subsequently on minimum usage

Availability

- Use of multiple servers (active-active or active-passive) and load balancer, which will detect that a request to a particular server is failing and handle the fault by routing requests to a secondary server

- Deployments are distributed across 2 availability zones

Maintanability

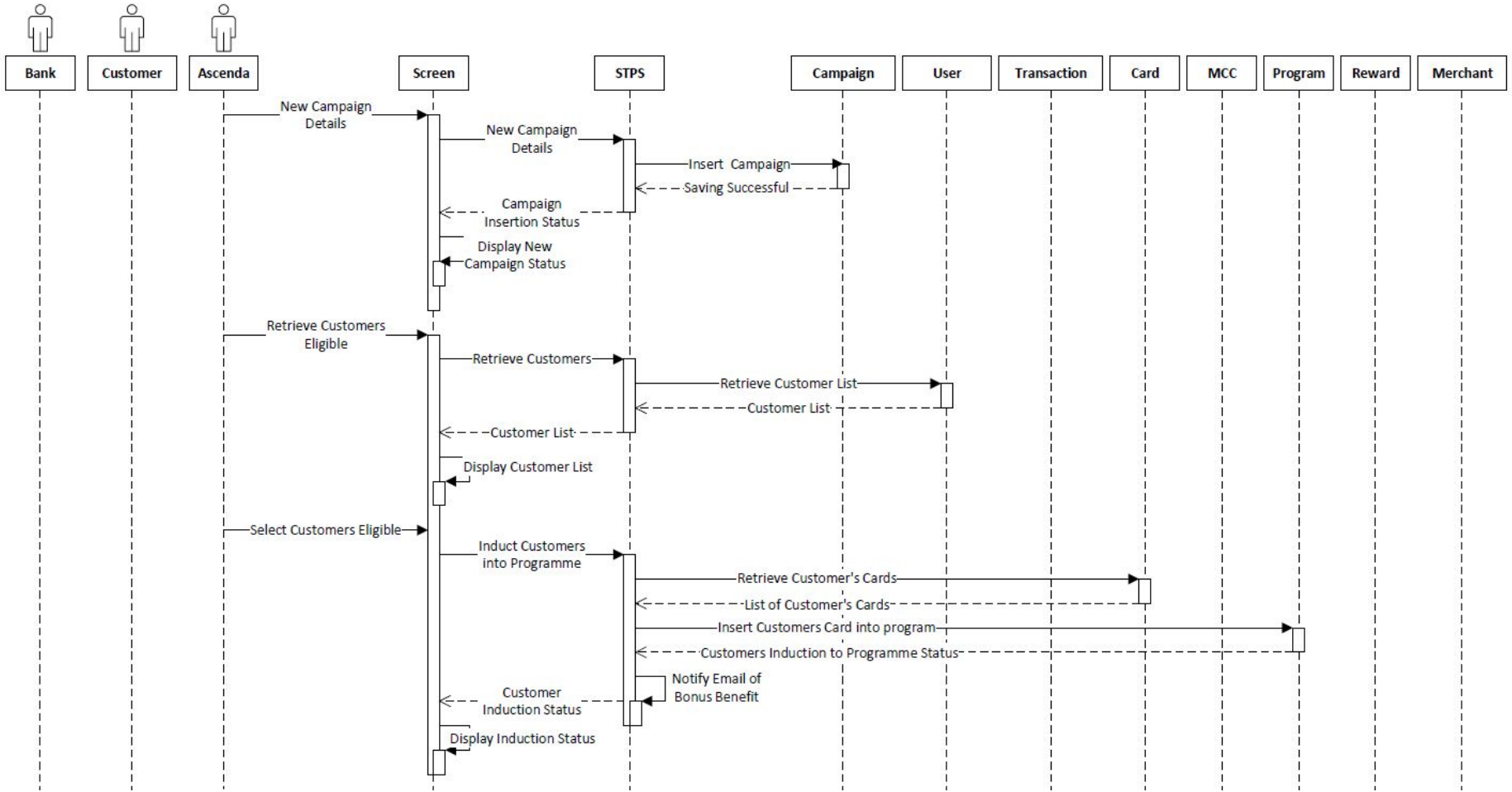
- Utilise AWS Codestar, AWS CodePipeline, AWS CodeBuild and AWS CodeDeploy for continuous deployments, resulting in little to no downtime during updates

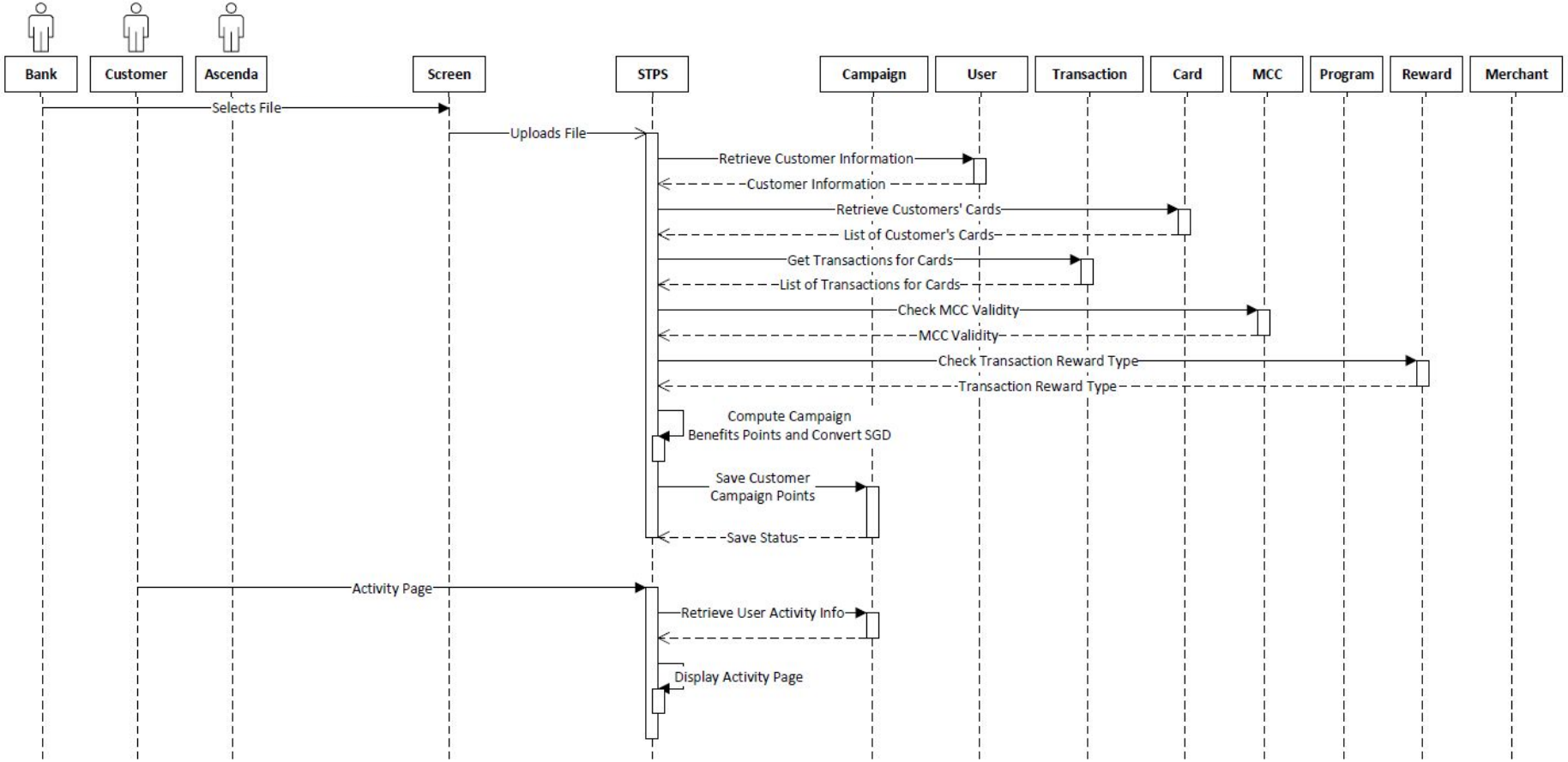
- Health checks are conducted using AWS ElasticBeanstalk

- Pipeline will identify errors detected in newly deployed updates and they will not be pushed to production

Security

- Ensure passwords are hashed before storing in the database
- Separate different functionalities of the application to smaller microservices with different authentication details
- AWS WAF blocks web exploits such as cross site scripting, SQL injections
- SSL/TLS certifications allow for secure network communications


Bank
Customer
Ascenda
Screen
STPS
Campaign
User
Transaction
Card
MCC
Program
Reward
Merchant
New Campaign Details
New Campaign Details
Insert Campaign
Saving Successful
Campaign Insertion Status
Display New Campaign Status
Retrieve Customers Eligible
Retrieve Customers
Retrieve Customer List
Customer List
Customer List
Display Customer List
Select Customers Eligible
Induct Customers into Programme
Retrieve Customer's Cards
List of Customer's Cards
Insert Customers Card into program
Customers Induction to Programme Status
Notify Email of Bonus Benefit
Customer Induction Status
Display Induction Status


Bank
Customer
Ascenda
Screen
STPS
Campaign
User
Transaction
Card
MCC
Program
Reward
Merchant
Selects File
Uploads File
Retrieve Customer Information
Customer Information
Retrieve Customers' Cards
List of Customer's Cards
Get Transactions for Cards
List of Transactions for Cards
Check MCC Validity
MCC Validity
Check Transaction Reward Type
Transaction Reward Type
Compute Campaign
Benefits Points and Convert SGD
Save Customer
Campaign Points
Save Status
Activity Page
Retrieve User Activity Info
Display Activity Page

Budget

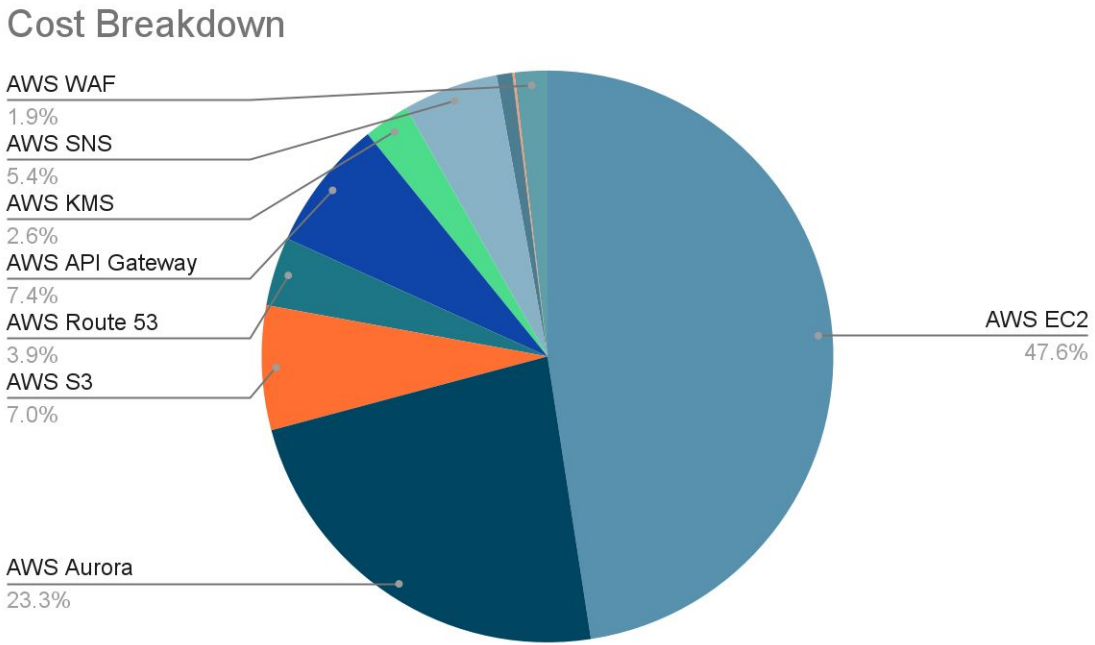
Service	Description	Cost
AWS EC2 T3.micro with VPC and ALB	EC2 for 1 AZ in 2 different regions	~ 128.41 USD/month
AWS Aurora MySQL	db.t3.medium with 20 GB database storage and backup storage	~ 62.81 USD/month
AWS S3	S3 Standard 20GB/month, S3 Glacier 5TB/month	~ 18.99 USD/month
Amazon CloudFront	Global CDN to deliver website and API call at high speed with low latency	1TB Free Tier for data transfer to internet and 10 million http/https requests
AWS Route 53	Cloud DNS web service, 1 hosted zone	~ 10.50 USD/month
AWS Web Application Firewall (WAF)	Number of Web Access Control Lists (Web ACLs) utilized (1 per month)	~ 5 USD/ month

Budget

Service	Description	Cost
AWS API Gateway	30 million API call per month, including free tier	~ 20 USD/month
AWS Key Management	Securely Generate and Manage AWS Encryption Keys	~ 7 USD/ month
AWS SNS	Notification service with estimated 30 million request/month	~ 14.50 USD/month
AWS CodeBuild	Estimated 12 builds per month Estimated 10mins build duration	~ 2.40 USD/month
AWS CodeDeploy	Number of on-premise instances (4), Number of deployments (4 per month)	~ 0.32 USD/ month

Budget

Total Cost: ~ 268.92 USD/ month



Thank you

Group 4

SMU
SINGAPORE MANAGEMENT UNIVERSITY

1. Instructions – Proposal Presentation

Assuming you are the architect, and you are proposing your architecture design of the solution. Your audience includes the management, development, and operation teams. You understand that this presentation is meant to solicit inputs from the sponsor and your final solution can still differ.

1. Keep your presentation to **10 minutes including Q&A**.
2. Explain your business need. Keep this less than 2 mins.
3. Spend most of the time explaining your technical solution. Justify your decisions as much as you can. Your objective is to solicit as many inputs from the sponsors as possible to finalize your design for implementation.
4. Please submit your presentation slides via GitHub together with your proposal document. Create a **presentation** folder in your GitHub project repository, name your presentation deliverable document as **Project-Presentation-<GitHub-Team-Name>** and place your presentation deliverable in it.
5. Late submission or failure to comply with the instructions will be penalized.
6. Presentation slots will be announced separately.

2. Grading Criteria - Proposal Presentation

- Timely and complete submission with conformance to the instructions.
- The extent to the quality of presentation including clear descriptions, explanations, and justifications.