SMU
SINGAPORE MANAGEMENT UNIVERSITY

School of
Information Systems

Khoo Khim Boon Ernest 01337679
Woon Quan Austin 01315332
Wong Hejian Javier 01340105
Sherman Lee 01333200
Leonard Tan 01368820
Tan Jiale Brennan 01346536

Background and Business Needs

Ascenda facilitates loyalty currency transfers between multiple loyalty programs. Without Ascenda's platform, building such capabilities is often time-consuming and complex. The loyalty program has to build individual integrations into each third party loyalty program and each integration is often quite different from each other. Some loyalty programs may be based on the REST architecture while others thrive on more traditional methods such as the SOAP protocol or file based transfer. Ascenda is required to collaborate with the banks with regards to the following functions: Provide the latest information about the loyalty programs, validate membership for a given loyalty program, accept and process accrual information on behalf of the banks and provide transaction details and status when being polled by the banks. Additionally, Ascenda is required to collaborate with the loyalty programs to perform transfer fulfillments as well.

Stakeholders

Stakeholder	**Stakeholder Description**	**Permissions**
Banks	Banks are Ascenda's primary customers that depend on Ascenda to carry out a multitude of functions which includes information gathering , validation of membership, processing of information and liaising with the different loyalty programs.	**Ascendas Middleware** - Read: Get latest available loyalty programmes - Write: Send information regarding transfer of points
Loyalty Programmes	Loyalty Programmes are the main vendors Ascenda works with to abstract out the complicated API integrations to other 3rd party loyalty programmes, perform seamless transfer and fulfil file retention policies in the correct file format.	**Ascendas Middleware** - Write: Send updated handback file to Ascenda's database to update transaction statuses - Read: Download accrual information
AWS	AWS provides Ascenda with the infrastructure and services to deploy and manage the applications. As an IT stakeholder, Ascenda is expected to use their services legally and perform all of the necessary security configuration and management tasks.	
Ascendas' Maintenance Team	Ascenda's Maintenance Team is responsible for the availability, scalability and security of the solution.	

Ascenda's Development Team	Ascenda's Development Team is responsible for the development of features that fulfill the business needs of the banks.	

Key Use Cases

Use Case Title - Retrieves loyalty programmes	
Use Case ID	1
Description	Ascendas API is invoked by the vendor to retrieve the necessary loyalty programmes information
Actors	Bank
Main Flow of events	Bank's client side calls Ascenda's API to retrieve all the subscribed loyalty programmes so that it may render it.
Alternative Flow of events	No subscribers. Error 404 / blank page.
Pre-conditions	Ascenda's API is up Ports are configured to accept data packets Have a registered API key to attach to header for authorization purposes
Post-conditions	Bank's Web Page displays all associated loyal programmes.

Use Case Title - Send loyalty programmes information	
Use Case ID	2
Description	Ascendas to expose endpoint for banks to call and retrieve latest loyalty programme information
Actors	Ascendas
Main Flow of events	Bank's client side calls Ascenda's API to retrieve all the subscribed loyalty programmes so that it may render it.
Alternative Flow of events	Client not authorized to call endpoint, 403 Forbidden error
Pre-conditions	Each bank should have a registered api key and attach this key to headers for authorization of access to API.
Post-conditions	Bank's Web Page displays all associated loyal programmes.

Use Case Title - Convert currencies to loyalty program points	
Use Case ID	3
Description	User initiates request to convert currencies associated with a loyalty program
Actors	User Bank
Main Flow of events	User enters amount to convert to loyalty points
Alternative Flow of events	None
Pre-conditions	Bank webpage is up User should have validated their membership
Post-conditions	Users should proceed for membership validation after.

Use Case Title - Validate loyalty program membership	
Use Case ID	4
Description	Upon receiving membership details, the Bank verifies validity of the membership via Ascenda.
Actors	Bank, Ascenda
Main Flow of events	Bank invokes Ascenda's API, Ascenda's API responses with with status code (200 or 404)
Alternative Flow of events	No alternative
Pre-conditions	Ascenda's API is up Ports are configured to accept data packets
Post-conditions	Validation Response which decides if the user can proceed to exchange currency

Use Case Title - Receive Accrual Information	
Use Case ID	5
Description	Bank sends accrual information to Ascenda via API endpoint
Actors	Bank, Ascenda
Main Flow of events	Bank sends requests including accrual information to

	Ascenda's endpoint, Ascenda replies with a unique system ID if accepted.
Alternative Flow of events	Bank sends requests including accrual information to Ascenda's endpoint, Asecnda rejects information due to incomplete or invalid data.
Pre-conditions	Ascenda's API is up Ports are configured to accept data packets
Post-conditions	Bank receives system ID for approved request

Use Case Title - Process Accrual Information	
Use Case ID	6
Description	Ascenda processes accrual information from the bank and converts into a format applicable to the loyalty programmes
Actors	Ascenda
Main Flow of events	Ascenda retrieves information, transforms the information by aggregation or modification and subsequently exporting it as a file format.
Alternative Flow of events	No Alternatives
Pre-conditions	Ascenda's necessary processing mechanisms should be up.
Post-conditions	An accrual file generated should be ready for export

Use Case Title - Send Accrual Information	
Use Case ID	7
Description	Ascenda sends accrual file to loyalty program
Actors	Ascenda, 3rd party loyalty program
Main Flow of events	Asecnda sends accrual file to loyalty program via SFTP
Alternative Flow of events	No Alternatives
Pre-conditions	Loyalty Program's application must be able to accept the accrual file format Loyalty Program's application must be able to accept the

	accrual file via SFTP
Post-conditions	Upon sending an accrual file, Ascenda should expect to receive a handback file in the right format.

Use Case Title - Receive handback file	
Use Case ID	8
Description	Ascenda receives the handback file sent from loyalty program
Actors	Ascendas' 3rd party loyalty program
Main Flow of events	Loyalty Program sends handbackfile to Ascenda via SFTP
Alternative Flow of events	No Alternatives
Pre-conditions	Ascendas' application must be able to accept the accrual file format Ascendas' application must be able to accept the accrual file via SFTP
Post-conditions	Upon accepting the accrual information, Ascendas should check for any errors and update the DB and message broker Ascendas should update DB on outcomes of transactions from handback file

Use Case Title - Process handback file	
Use Case ID	9
Description	Ascenda processes the handback file sent from loyalty programs and updates the database and message broker of the status
Actors	Ascendas
Main Flow of events	Ascendas checks that there are no errors in format of the information provided by the loyalty program and updates the DB and message broker of the successful processing
Alternative Flow of events	There are format errors in the information provided by the loyalty program and Ascenda's provides an option to download the erroneous data in a .csv format

Pre-conditions	Ascenda's necessary processing mechanisms should be up.
Post-conditions	Ascenda's database and the message broker are updated of a success or failed processing

Use Case Title - Accepts banks Transaction enquiry	
Use Case ID	10
Description	Ascenda's application accepts the poll from the bank for a status on conversion status
Actors	Bank, Ascendas
Main Flow of events	Bank sends a real-time API call to Ascendas' application requesting for a status
Alternative Flow of events	No Alternative
Pre-conditions	Ascendas' application must be able to accept the banks API-call Bank has registered API key with Ascendas
Post-conditions	Ascenda's application begins processing the banks request

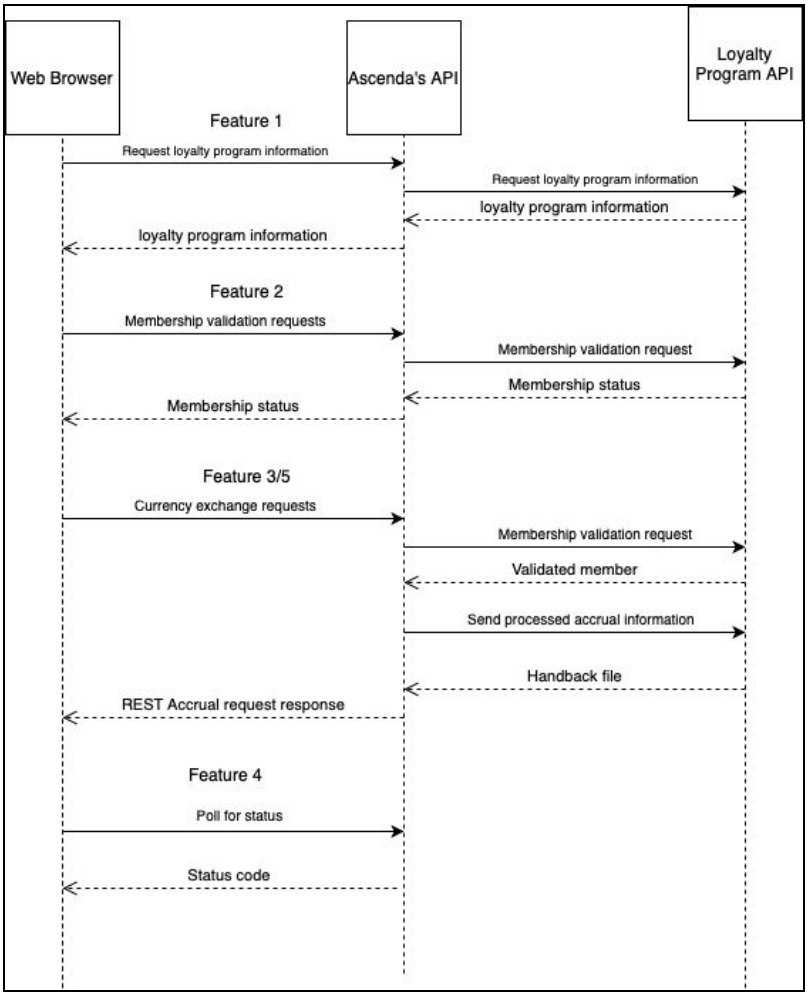
Use Case Title - Send transaction outcome information	
Use Case ID	11
Description	Ascenda's replies to the banks the status of the enquired transaction
Actors	Bank, Ascendas
Main Flow of events	Ascenda's application sends the status of the transaction via API-call to the bank's application
Alternative Flow of events	No Alternative
Pre-conditions	Bank's application must be able to accept the responses from Ascendas application via a protocol
Post-conditions	Bank's are updated on the status of the enquiry

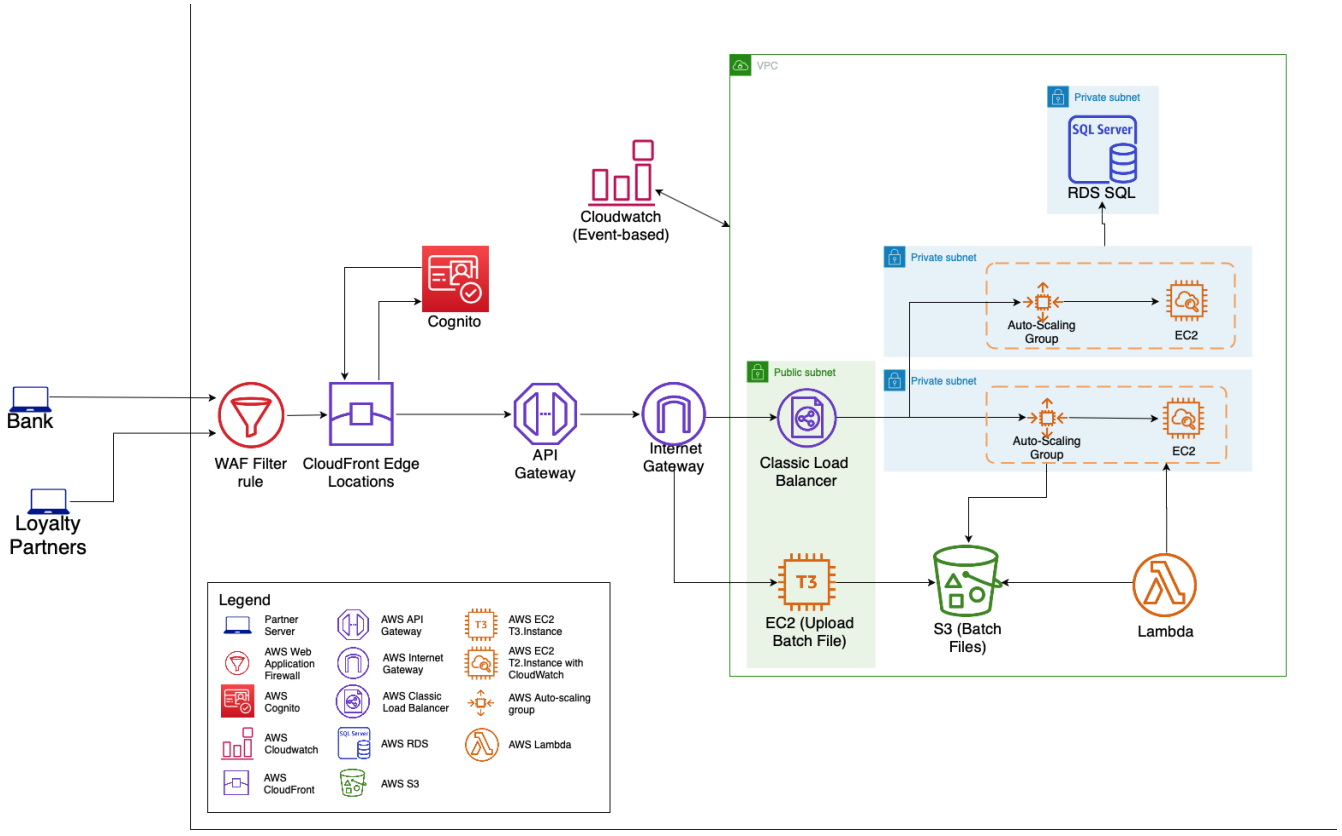
Proposed Services and cost

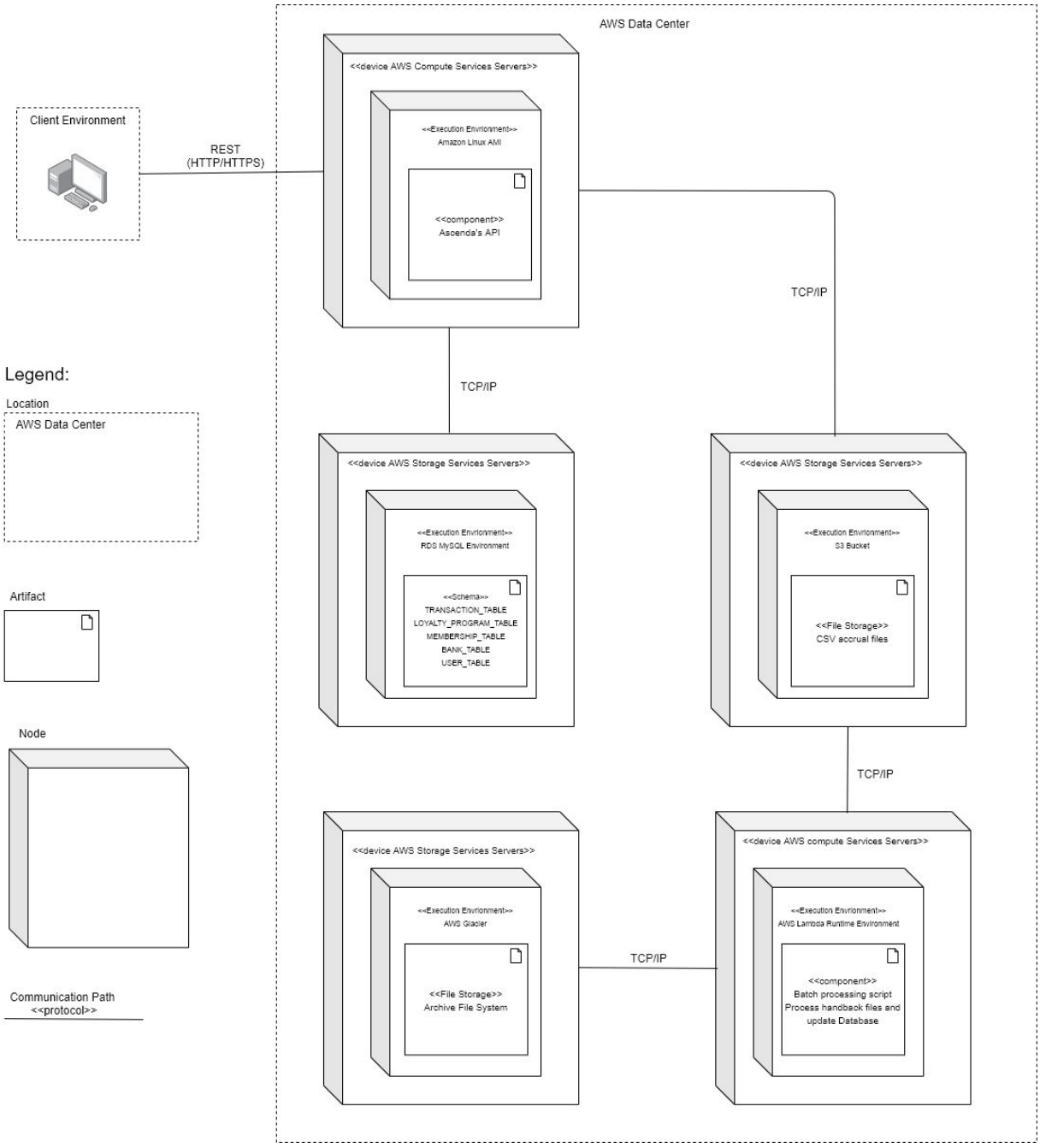
AWS Service	Description of the purpose and justification	Cost
S3	Storage of CSV accrual files generated by batch processing and CSV handback files from loyalty programs. **Justification:** We considered between S3, Elastic Block Store (EBS), Elastic File Store (EFS), and decided that S3 best suits our storage needs. ● Do not need high IOPS from EFS ● Did not go with EBS as we did not need high throughput for read and write operations ● S3 is best option for our use case case we only read and write once	Using an estimate of 10000 POST requests and 1GB of storage per month, the estimated cost is **USD0.25/month**
Elastic beanstalk (EBS)	Elastic Beanstalk is used to manage deployment and hosting of our backend API services, developed with Django. **Justification:** We considered between EC2 instances, Elastic container service (ECS), Elastic Kubernetes service(EKS), and EBS. ● Workload is not containerised, decided not to use ECS or EKS ● Managing individual EC2 instances individually has a lot of management overhead ● Chose EBS as there is no additional charge and it simplifies deployment and management	Using 3 instances of T2.micro under the EC2 instance savings plan, estimated cost is **USD 19.93/Month**
Load Balancer	Classic Load Balancer (CLB) allow for high availability and act as a reverse proxy	A CLB costs **USD 20.44/Month**
Lambda	We are using lambda functions for 2 use cases. First, to handle batch processing at the end of each day to convert the day's transactions to CSV, and send the accrual file to loyalty programs. Second, to process the handback files from loyalty partners and update the data using the backend. **Justification**:We considered EC2 instances or lambda functions. Our use case is event and schedule driven, hence lambda functions are more cost effective than EC2 ins	With 60 estimated invocations per month, 200ms per request and 500mb of memory allocation, the lambda functions cost **USD 0.00/Month**

API gateway	To act as another layer before any request hits the application layer. We are using AWS API gateway to handle authentication in conjunction with Cognito. This allows us to handle authentication outside of the backend application.	With 1 million requests per month, the API gateway costs **USD 4.25/Month**
Cognito	To be used with the API gateway to authenticate loyalty programs and banks. Cognito manages the user accounts that will have access to the backend. It is used with the API gateway to simplify authentication.	With estimated 30,000 Monthly Active Users, the cost will be **USD 0.00/Month**
WAF	An application firewall aimed at blocking common attack patterns. (e.g. SQL Injection, XSS)	1 Web ACL with 5 rules **USD 16/month**
CloudWatch	Application based monitoring is free for EC2 instances	Free tier for CloudWatch costs
RDS	MySQL RDS as the managed database for the application. Justification: Our choices for databases include managing a MySQL server or using RDS. RDS handles automatic snapshots and multi-AZ deployments. This removes the management overhead for ensuring availability and durability.	1 db.t3.micro instance costs **USD 17.45/month**.

Views

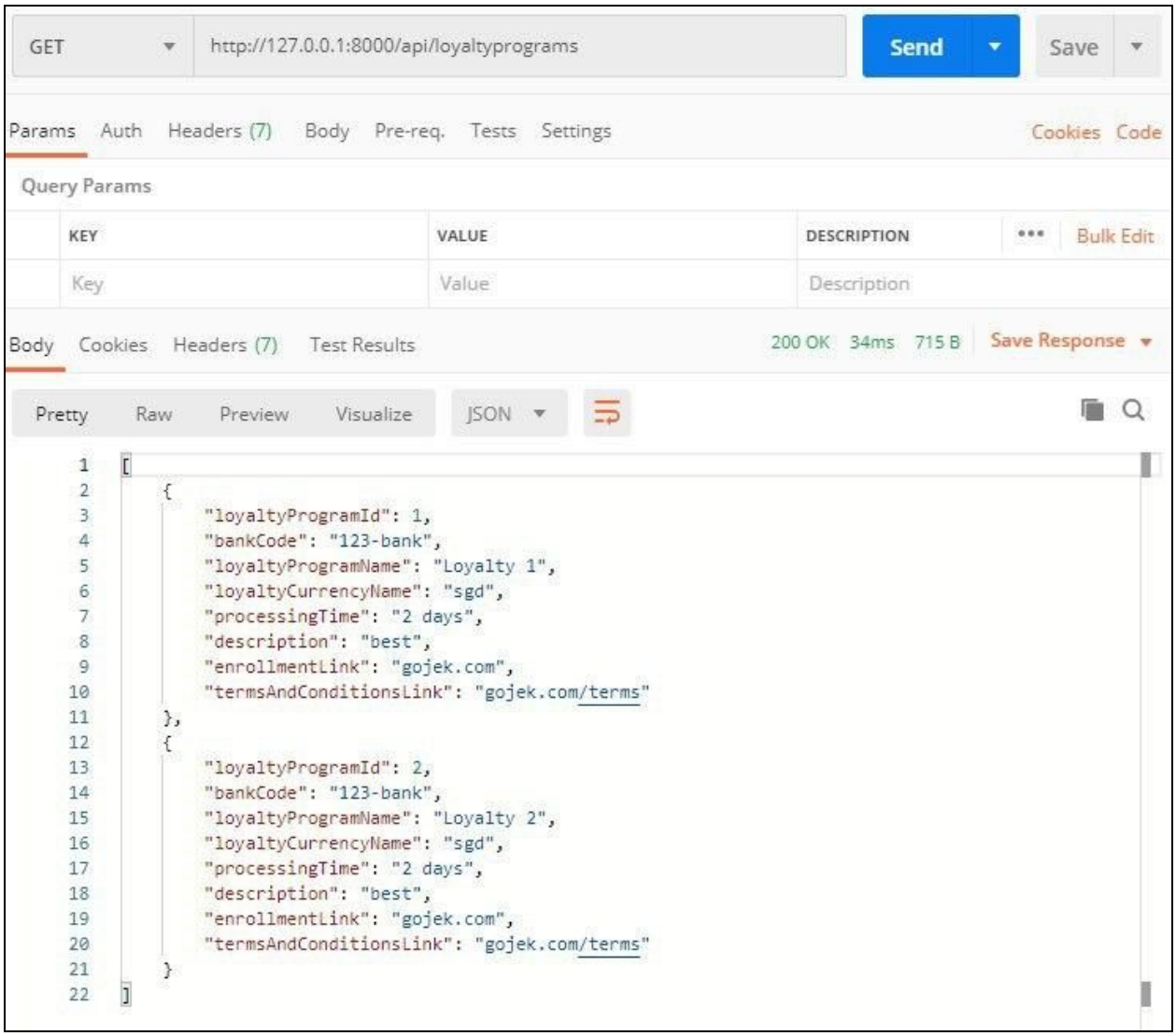

Web Browser
Ascenda's API
Loyalty Program API
Feature 1
Request loyalty program information
Request loyalty program information
loyalty program information
loyalty program information
Feature 2
Membership validation requests
Membership validation request
Membership status
Membership status
Feature 3/5
Currency exchange requests
Membership validation request
Validated member
Send processed accrual information
Handback file
REST Accrual request response
Feature 4
Poll for status
Status code


VPC
Private subnet
SQL Server
RDS SQL
Cloudwatch (Event-based)
Cognito
Private subnet
Auto-Scaling Group
EC2
Public subnet
Private subnet
Bank
Loyalty Partners
WAF Filter rule
CloudFront Edge Locations
API Gateway
Internet Gateway
Classic Load Balancer
Auto-Scaling Group
EC2
T3
EC2 (Upload Batch File)
S3 (Batch Files)
Lambda
Legend
Partner Server
AWS Web Application Firewall
AWS Cognito
AWS Cloudwatch
AWS CloudFront
AWS API Gateway
AWS Internet Gateway
AWS Classic Load Balancer
AWS RDS
AWS S3
AWS EC2 T3.Instance
AWS EC2 T2.Instance with CloudWatch
AWS Auto-scaling group
AWS Lambda

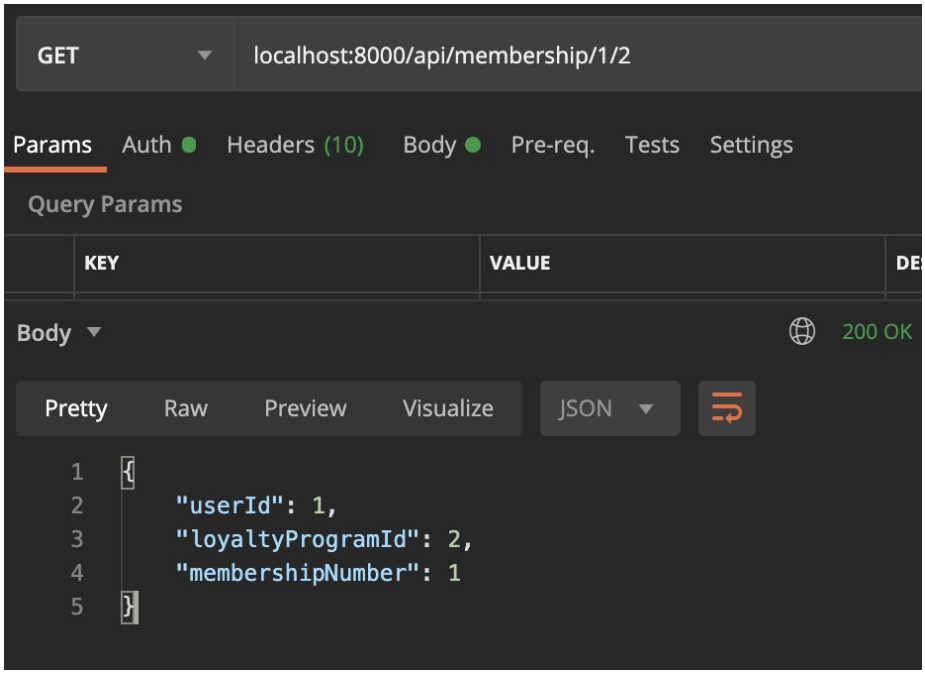

AWS Data Center
Client Environment
REST
(HTTP/HTTPS)
<<device AWS Compute Services Servers>>
<<Execution Environment>>
Amazon Linux AMI
<<component>>
Asoenda's API
TCP/IP
TCP/IP
Legend:
Location
AWS Data Center
Artifact
Node
Communication Path
<<protocol>>
<<device AWS Storage Services Servers>>
<<Execution Environment>>
RDS MySQL Environment
<<Schema>>
TRANSACTION_TABLE
LOYALTY_PROGRAM_TABLE
MEMBERSHIP_TABLE
BANK_TABLE
USER_TABLE
<<device AWS Storage Services Servers>>
<<Execution Environment>>
S3 Bucket
<<File Storage>>
CSV accrual files
TCP/IP
<<device AWS Storage Services Servers>>
<<Execution Environment>>
AWS Glacier
<<File Storage>>
Archive File System
TCP/IP
<<device AWS compute Services Servers>>
<<Execution Environment>>
AWS Lambda Runtime Environment
<<component>>
Batch processing script
Process handback files and
update Database

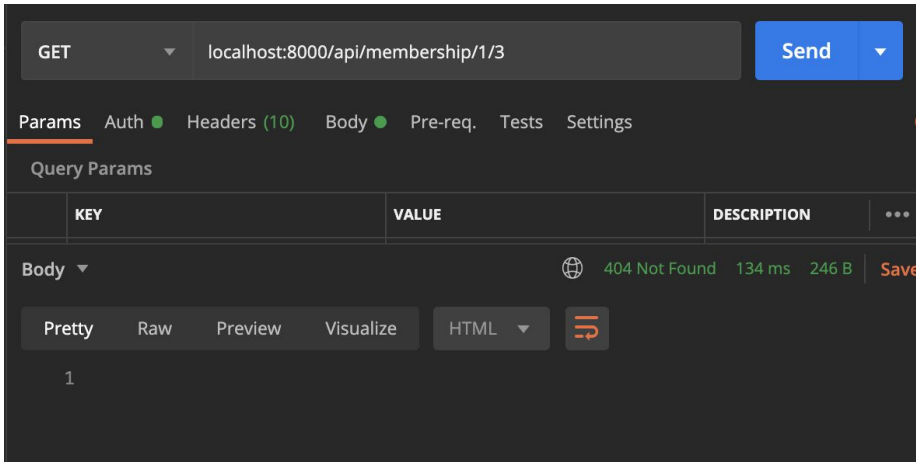
Appendix

Feature 1

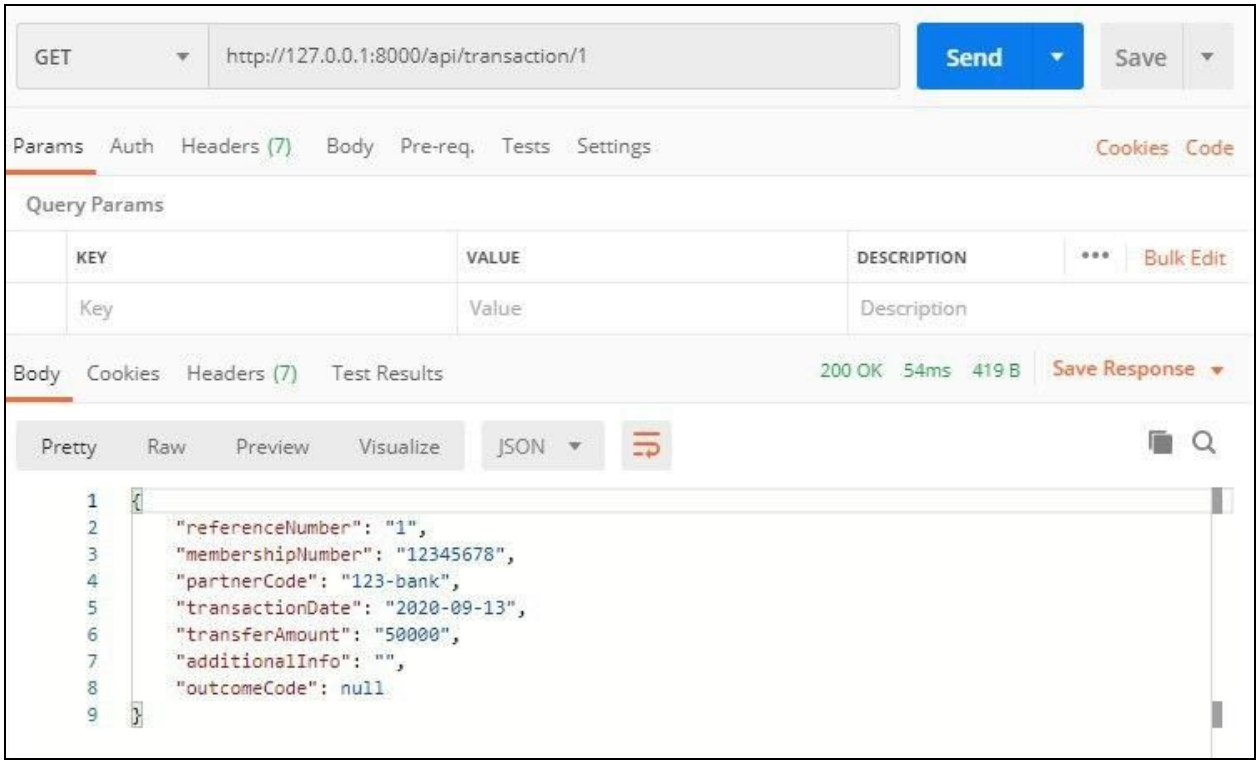


Feature 2

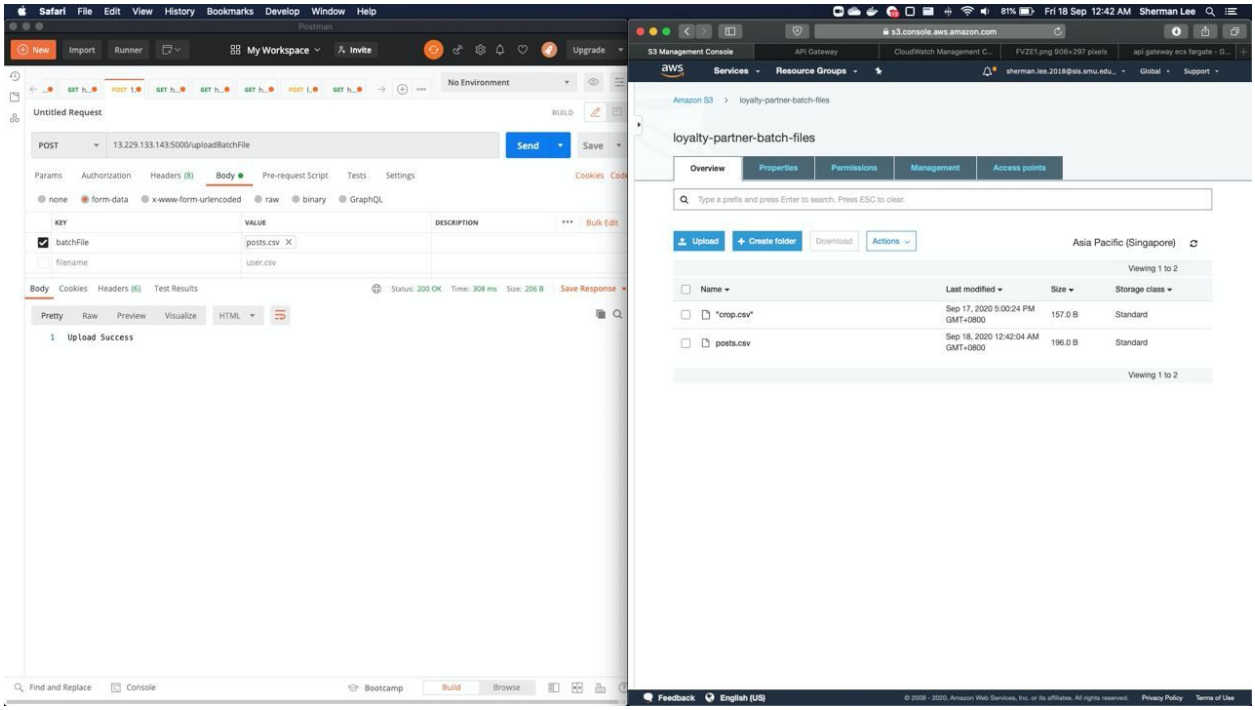




Feature 4



Feature 5


Safari File Edit View History Bookmarks Develop Window Help
Fri 18 Sep 12:42 AM Sherman Lee
Postman
New Import Runner My Workspace Invite Upgrade
No Environment
Untitled Request
POST 13.229.133.143:5000/uploadBatchFile
Send Save
Params Authorization Headers (8) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL
KEY VALUE DESCRIPTION Bulk Edit
batchFile posts.csv
filename user.csv
Body Cookies Headers (6) Test Results Status: 200 OK Time: 308 ms Size: 206 B Save Response
Pretty Raw Preview Visualize HTML
1 Upload Success
Find and Replace Console Bootcamp Build Browse
s3.console.aws.amazon.com
S3 Management Console API Gateway CloudWatch Management C... FVZE1.png 906×297 pixels api gateway ecs fargate - G...
aws Services Resource Groups sherman.lee.2018@sis.smu.edu... Global Support
Amazon S3 > loyalty-partner-batch-files
loyalty-partner-batch-files
Overview Properties Permissions Management Access points
Type a prefix and press Enter to search. Press ESC to clear.
Upload Create folder Download Actions Asia Pacific (Singapore)
Viewing 1 to 2
Name Last modified Size Storage class
"crop.csv" Sep 17, 2020 5:00:24 PM GMT+0800 157.0 B Standard
posts.csv Sep 18, 2020 12:42:04 AM GMT+0800 196.0 B Standard
Viewing 1 to 2
Feedback English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use