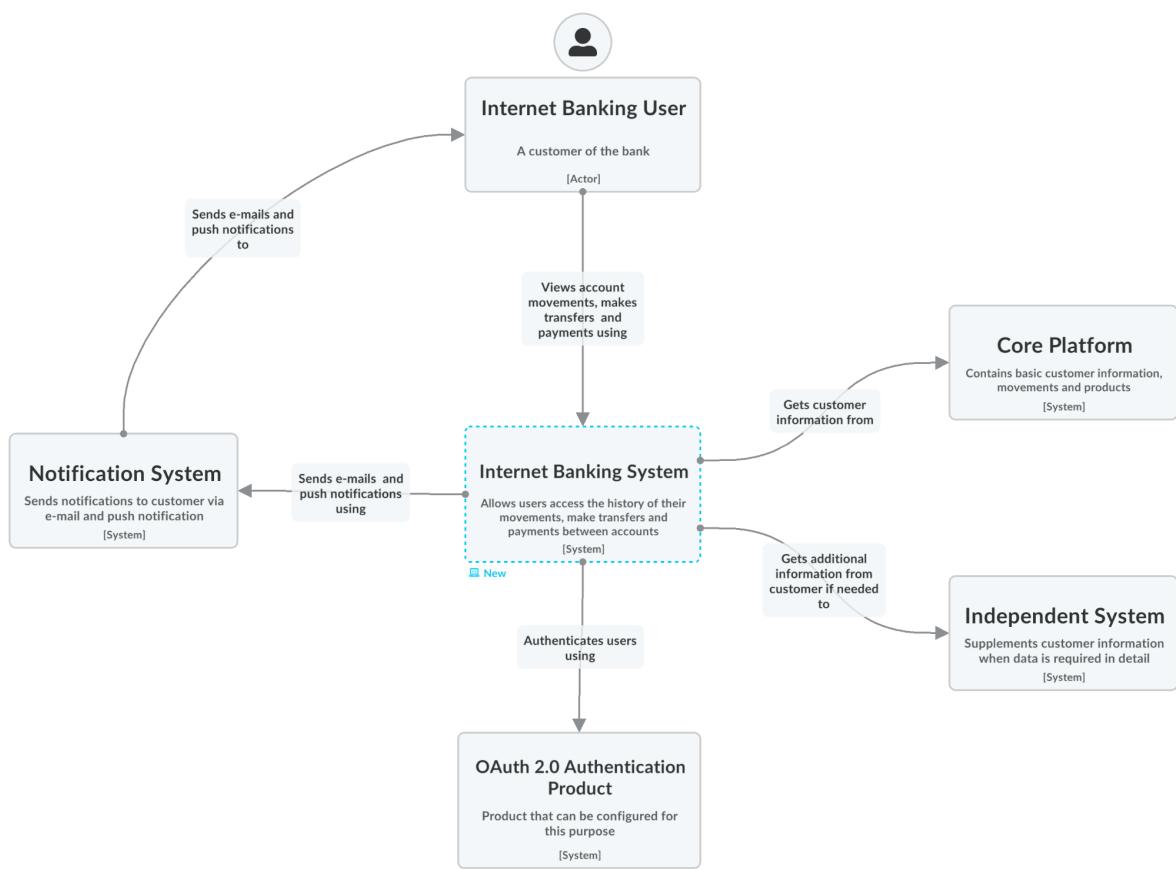


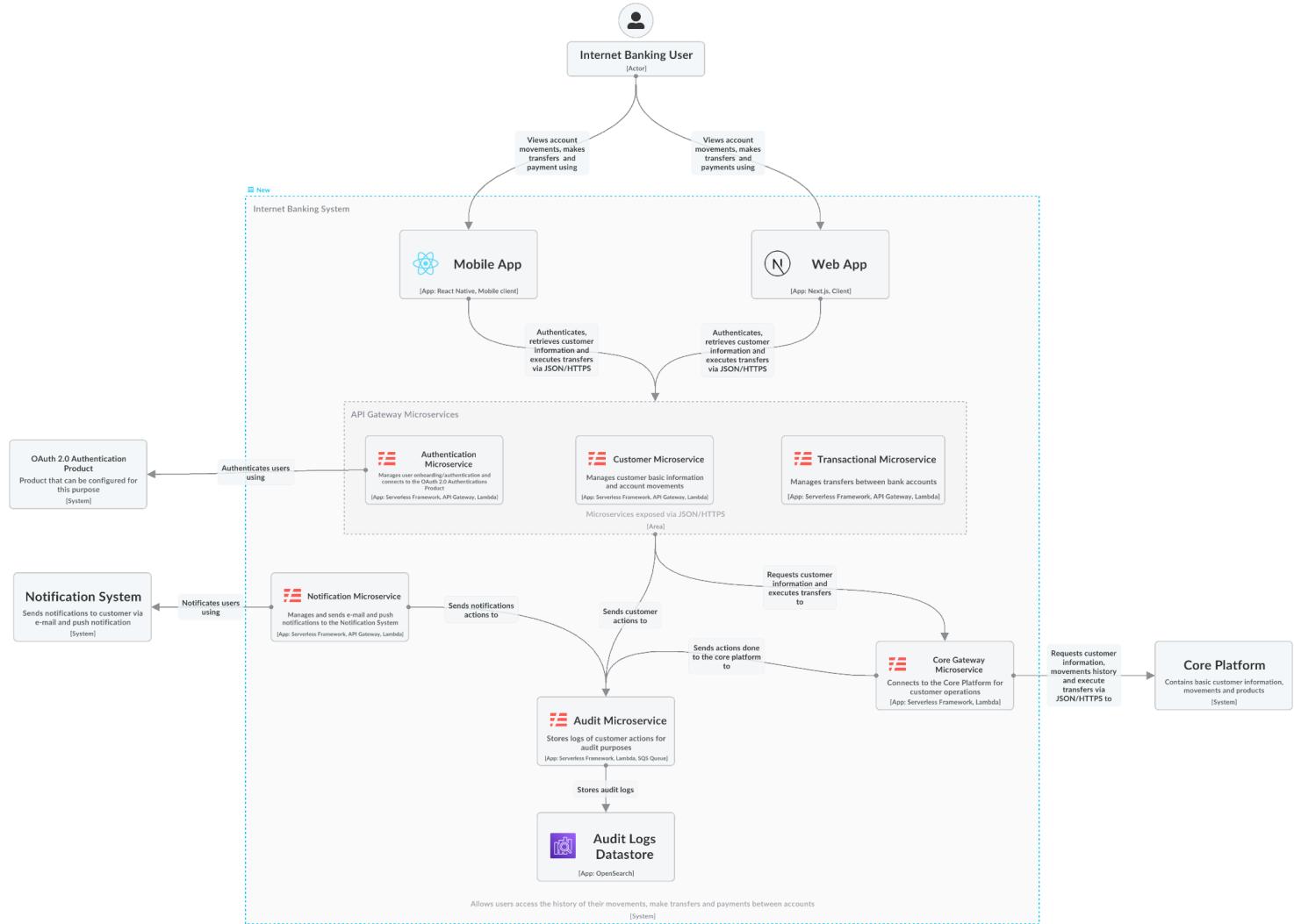
ANSWER ESTEBAN ARISTIZABAL BP (26 NOV 2023)

1. System Context Diagram



2. Application Diagrams

a. Internet Banking System

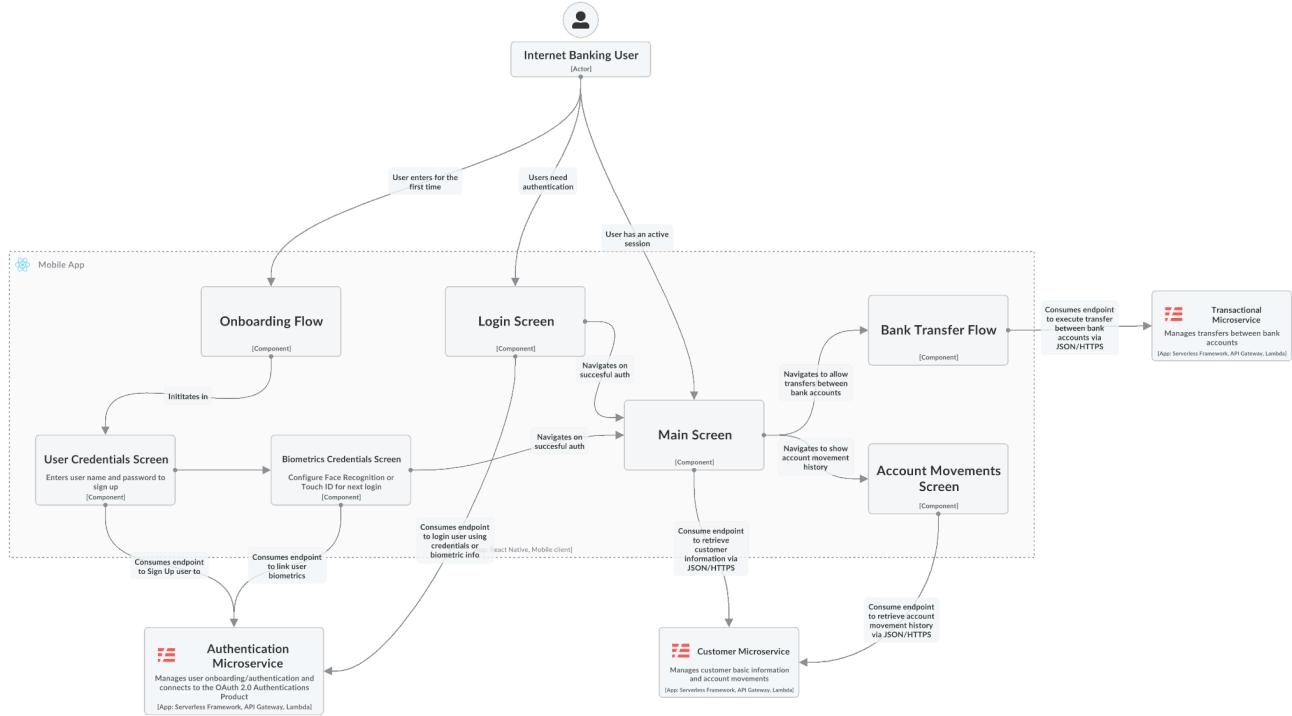


Notes:

- Notification System will be an external platform in charge of sending notifications to clients by email and push notifications to their mobile device
- For monitoring and error detection purposes "Rollbar" is the tool recommended for implementing in the microservices

3. Component Diagrams

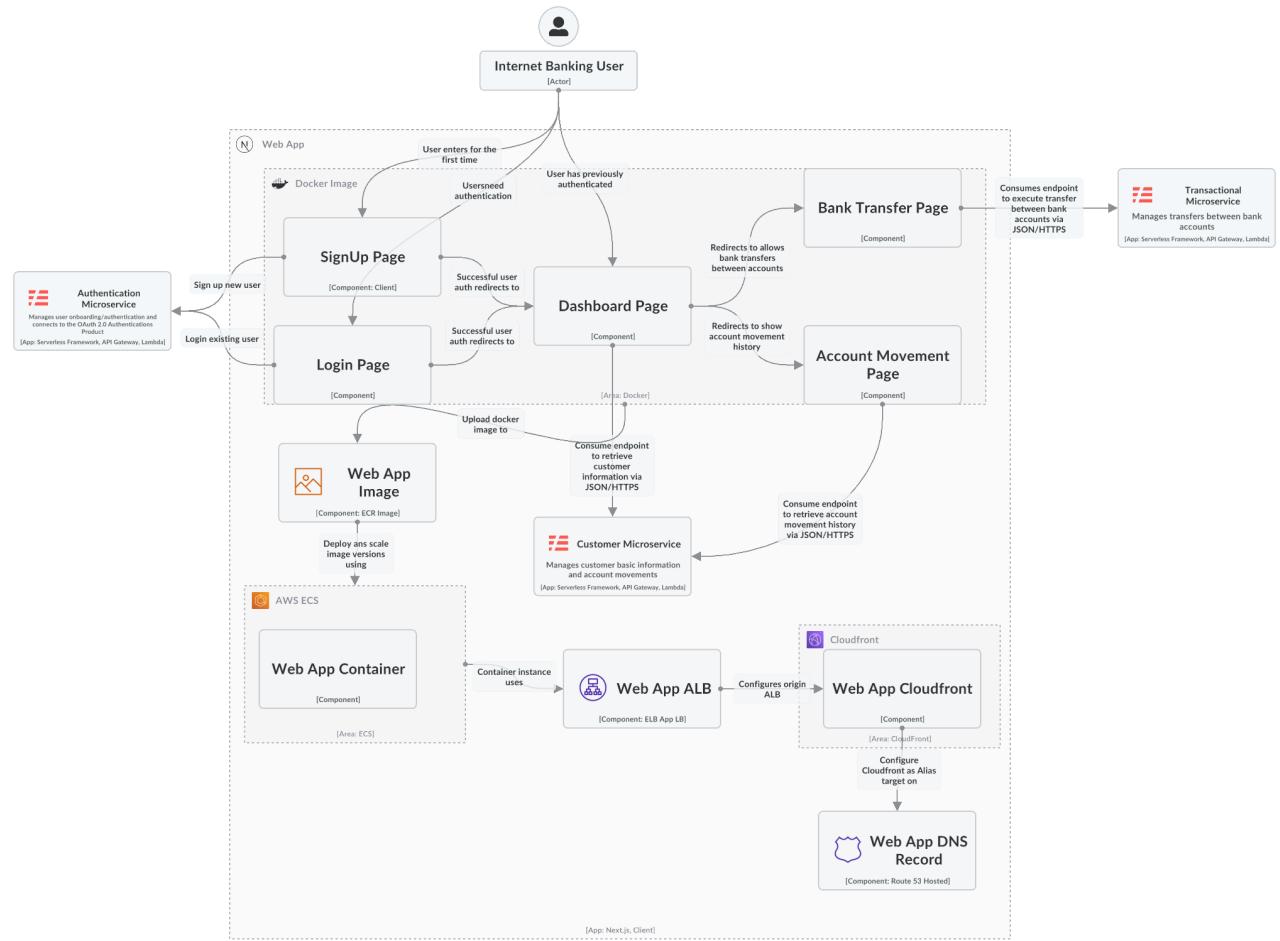
a. Mobile App



Notes:

- React Native was the chosen tool due to:
 - Maturity in the market
 - Vast ecosystem of libraries and modules for banking purposes
 - Similarity in building web application for smooth development
- Flutter was the discarded tool due to:
 - May have platform limitations access for banking purposes
 - Learning curve for new developers in adopting a new language
 - Smaller community support
- The mobile app will allow the user to login via touch id and face recognition biometrics

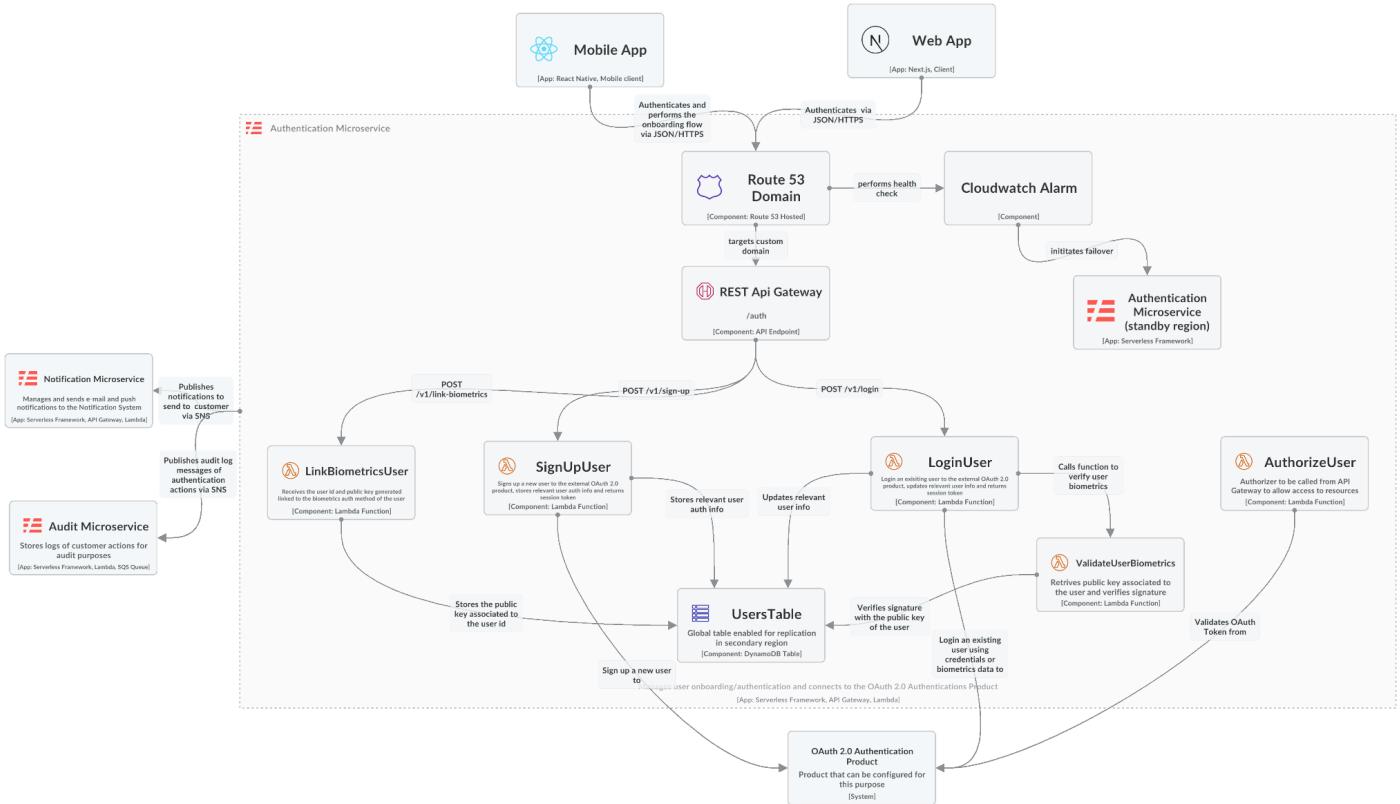
b. Web App



Notes:

- NextJS was the chosen tool due to:
 - Built in support for server side rendering
 - Offers enhanced user experience in terms of performance and usability
 - Improved security and SEO
- ViteJS was the discarded tool due to:
 - Not offers a BFF solution
 - Smaller community support
 - Uses modern JavaScript features that are not yet supported by all browsers
- Web app will be deployed and managed with AWS ECS for scaling purposes, no Ec2 instances to manage and increased CI/CD benefits

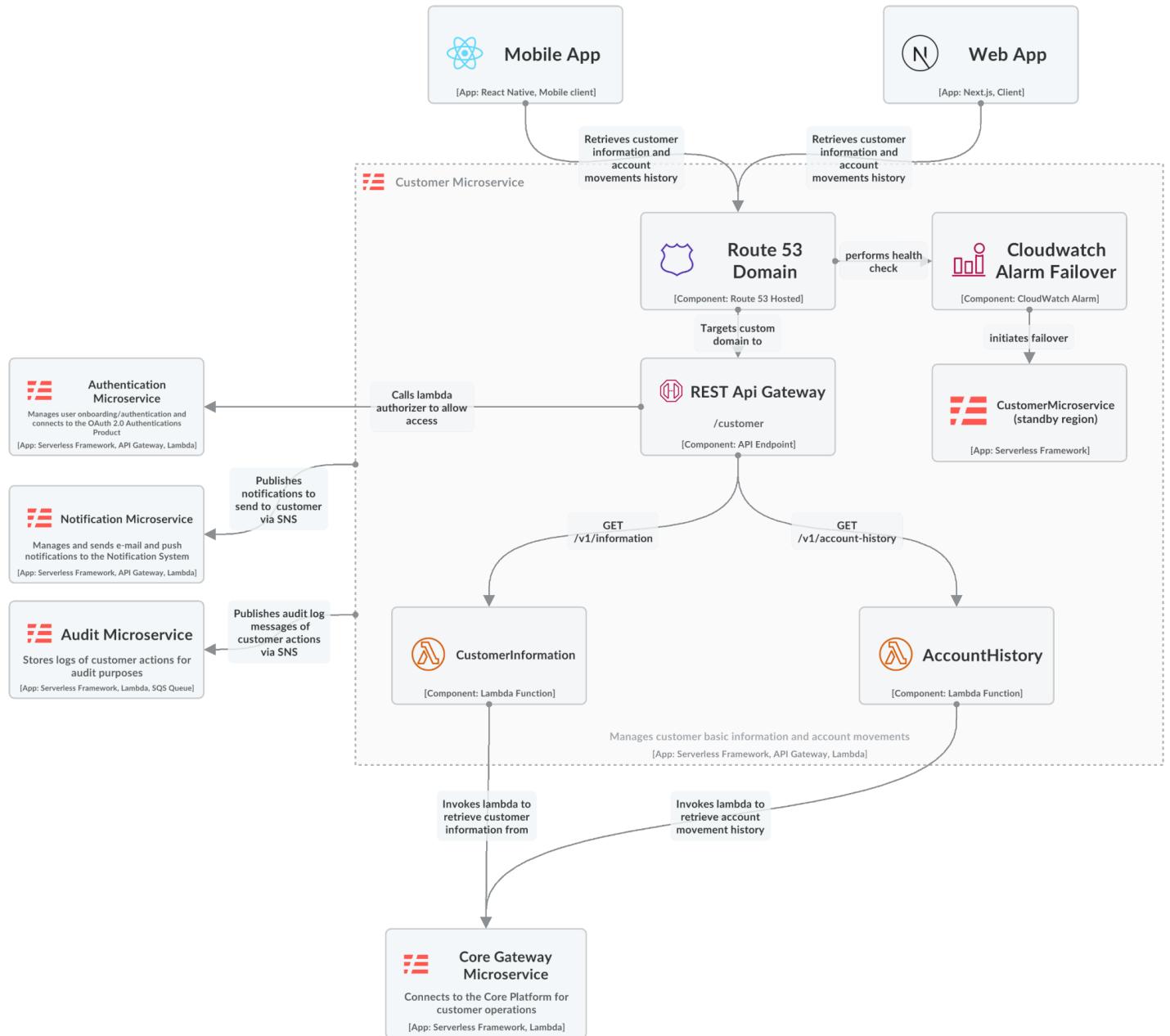
c. Authentication Microservice



Notes:

- The mobile app will allow the user to login via touch id and face recognition biometrics
- The standard for user authentication will be OAuth 2.0 due to:
 - Increased Access Control
 - Seamless Integration With Third-Party Applications
 - Easy Revocation Of Access
 - Exchange The Authorization Code For An Access Token
- The biometrics authentication will follow the "biometric key pair approach", first a key will be generated in the mobile app, and the public key will be stored in the backend solution for user validation
- Microservice will be deployed in standby region for (DR) purposes
- Node js is chosen as the programming language for the backend

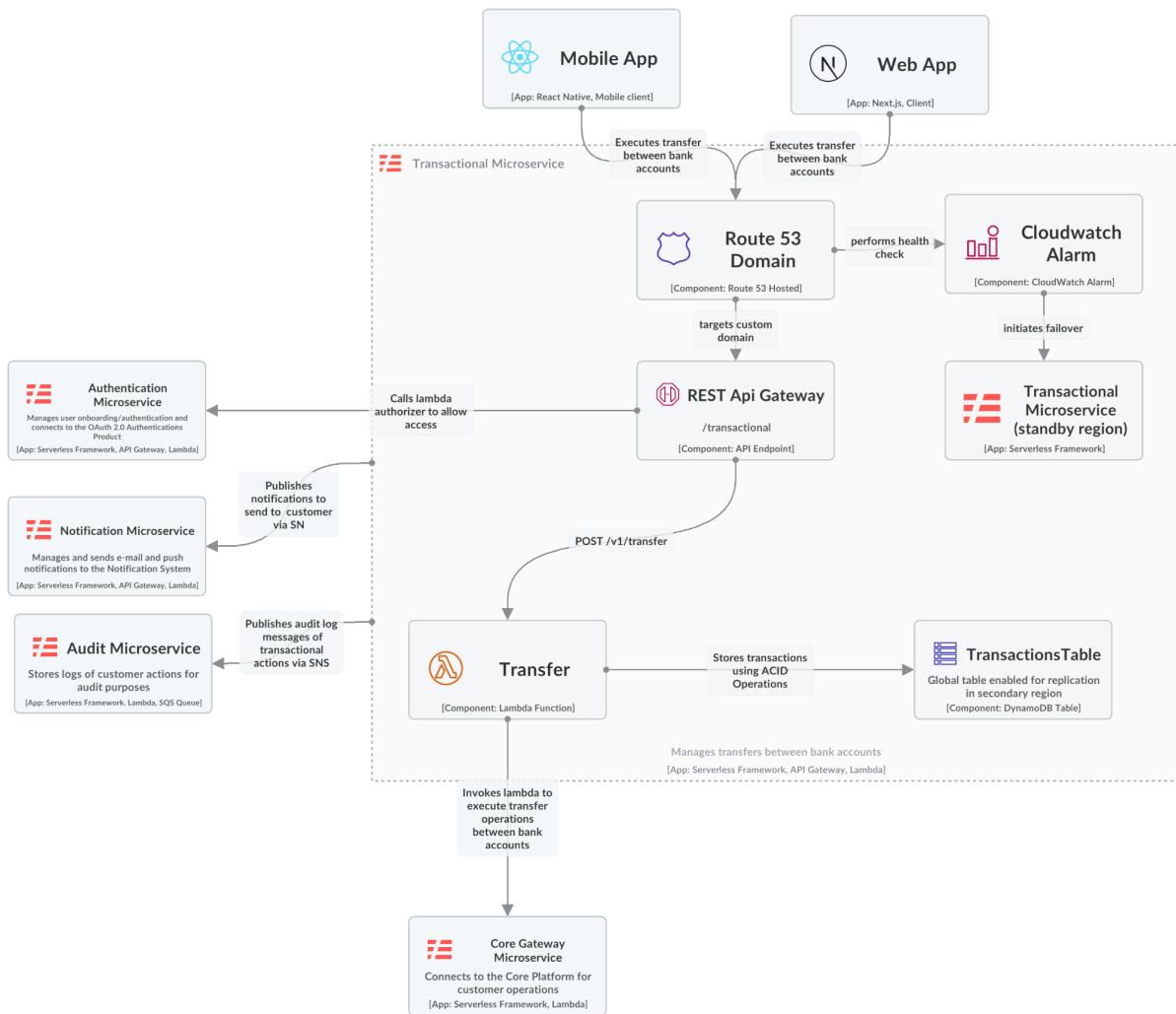
d. Customer Microservice



Notes:

- Microservice will be deployed in standby region for (DR) purposes
- Node js is chosen as the programming language for the backend

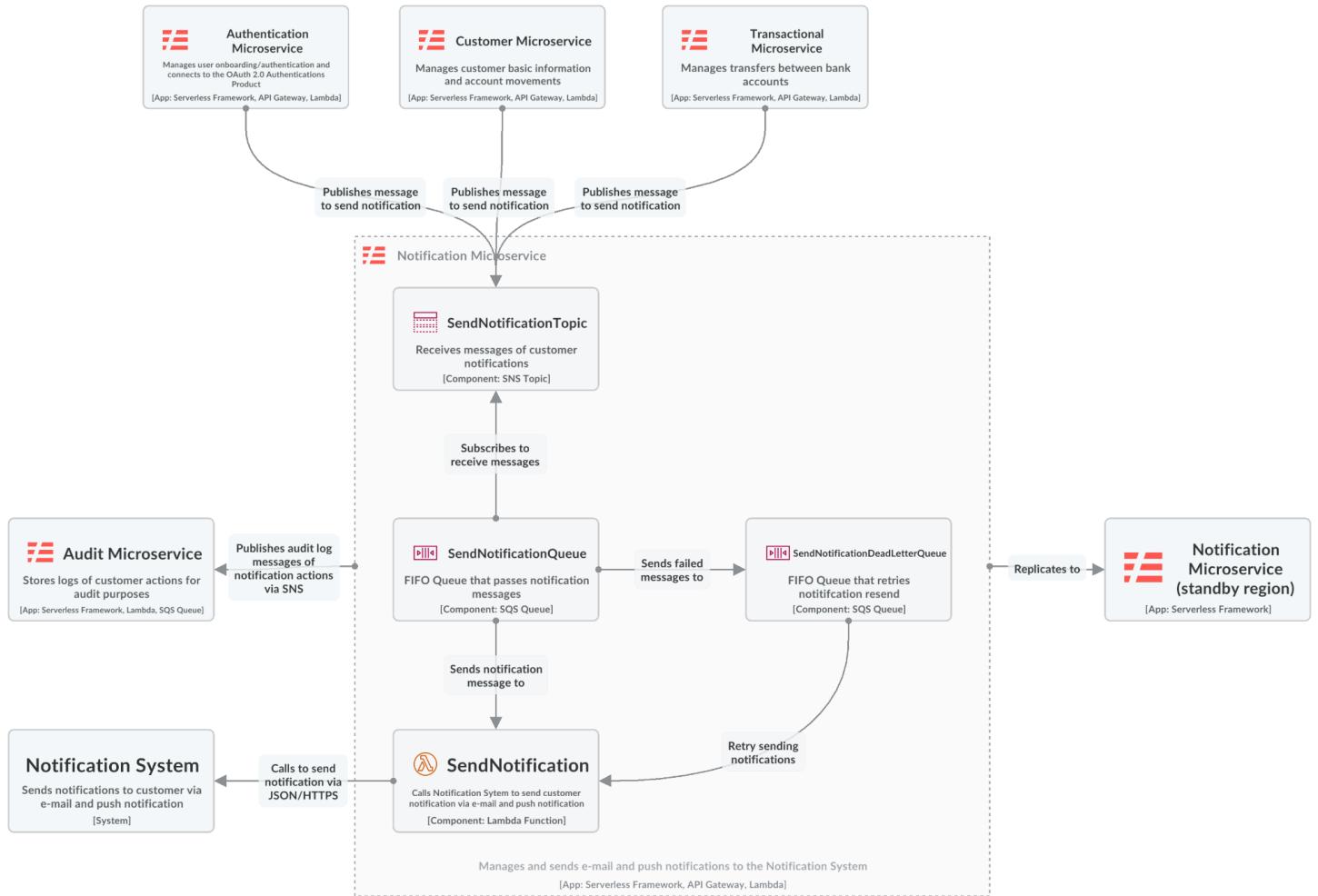
e. Transactional Microservice



Notes:

- Microservice will be deployed in standby region for (DR) purposes
- Node js is chosen as the programming language for the backend

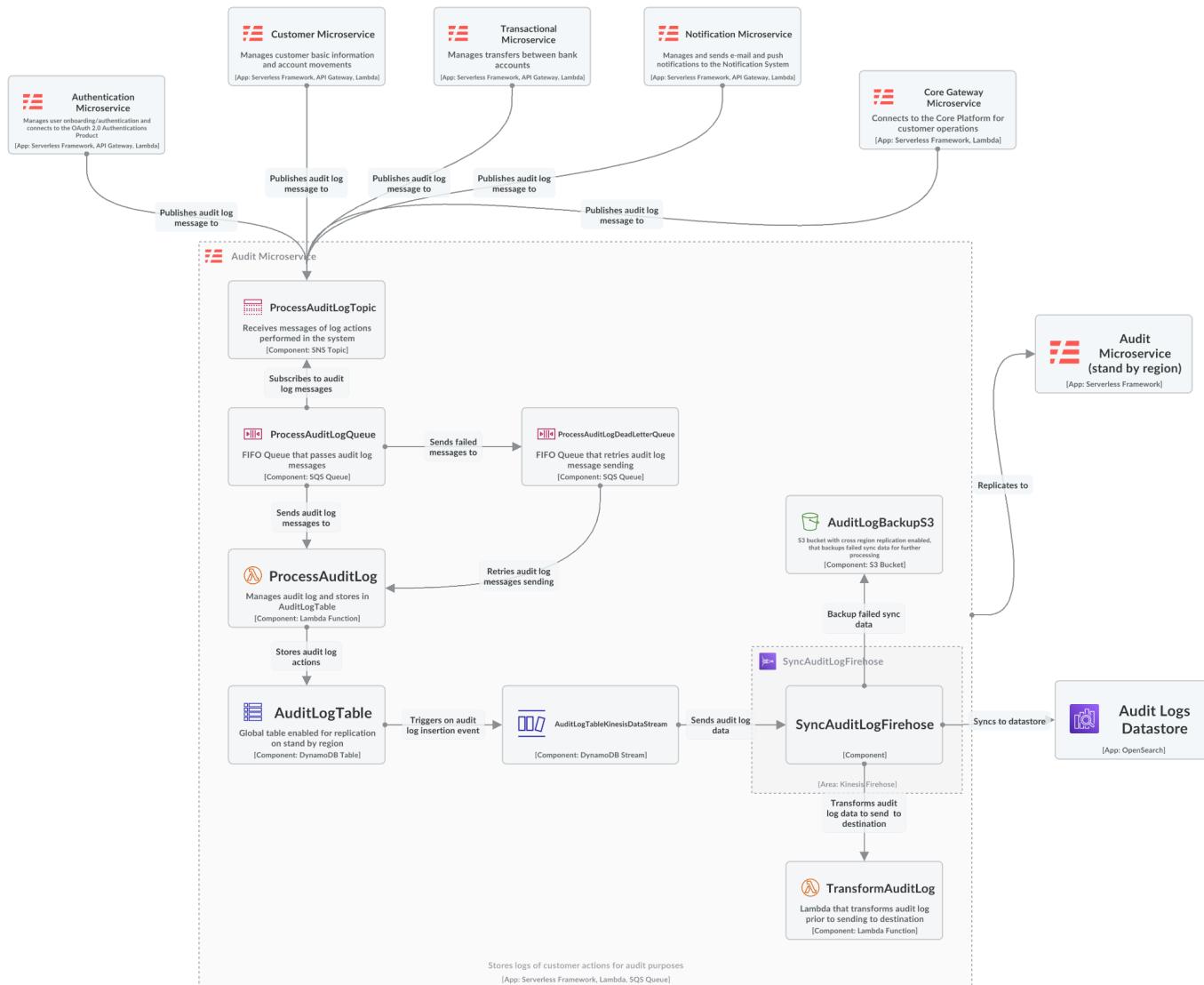
f. Notification Microservice



Notes:

- Microservice will be deployed in standby region for (DR) purposes
- Node js is chosen as the programming language for the backend

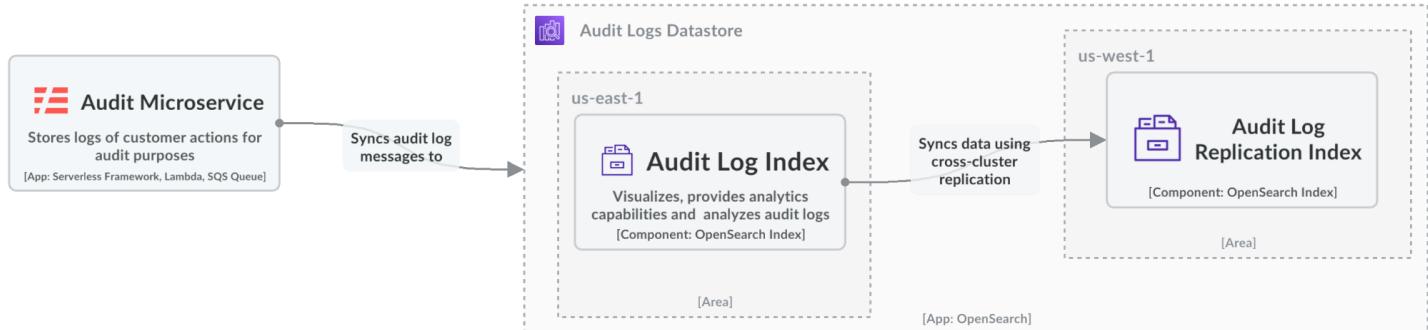
g. Audit Microservice



Notes:

- Microservice will be deployed in standby region for (DR) purposes
- Node js is chosen as the programming language for the backend

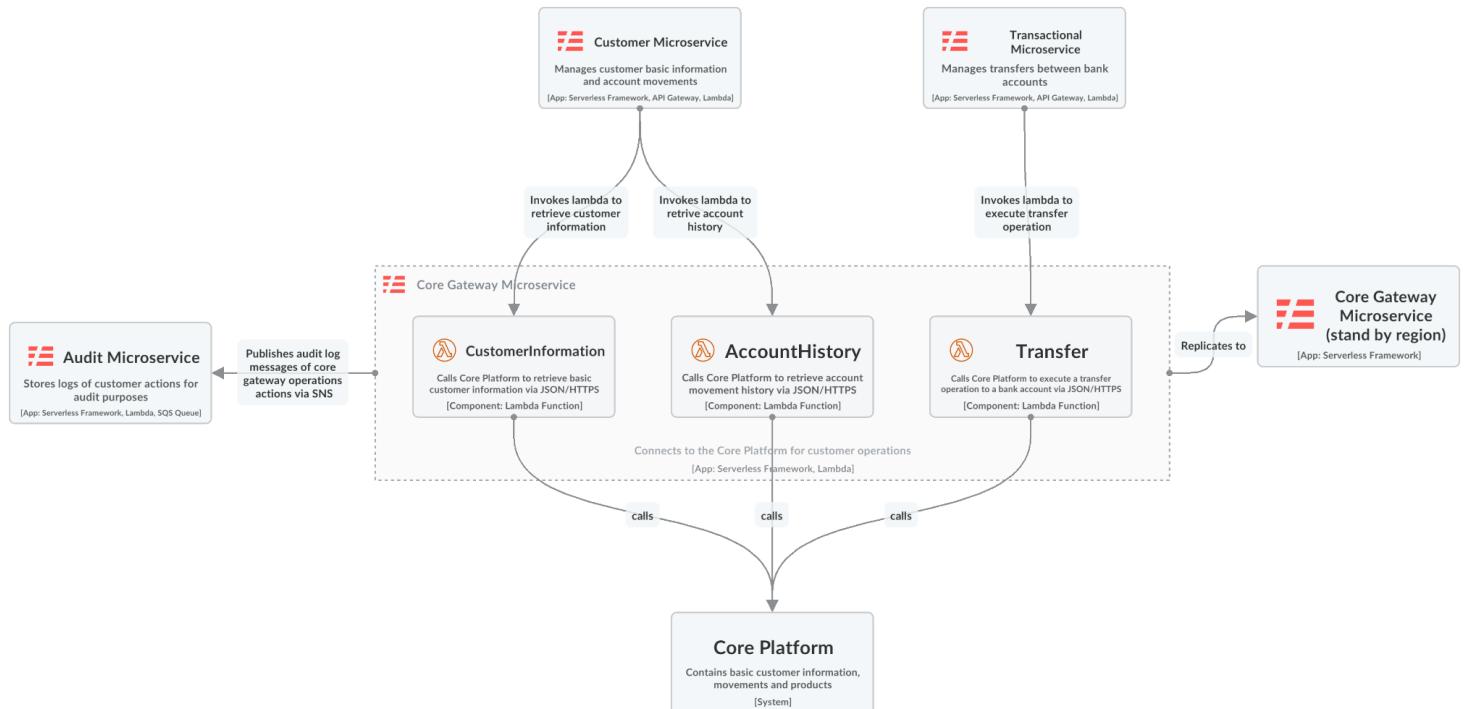
h. Audit Logs Datastore



Notes:

- Opensearch cross-cluster replication will be enabled for (DR) and (HA) options

i. Core Gateway Microservice



Notes:

- Microservice will be deployed in standby region for (DR) purposes
- Node js is chosen as the programming language for the backend