



A High AI-Q[™]
Company



Inclusive Peer-to-Peer Mental Health Support Platform

Creating a secure, audio-only safe space for female-identifying and non-binary individuals using AWS serverless architecture and real-time communication tools.

Overview

We engineered a secure, peer-led mental health application tailored for female-identifying and non-binary individuals. The solution utilizes an AWS serverless architecture (Amplify, Lambda, and AppSync) to provide high scalability with minimal operational overhead. Featuring real-time, audio-only support groups powered by the Agora SDK, the app provides a private platform for Gen Z users to share experiences and attend peer-led discussions in a moderated, exclusive digital environment.

100%

Serverless
Backend

Zero

Server
Maintenance

Scalable

Auto scaling
audio

- **Cost-Optimized Infrastructure:** Reduced operational expenditure significantly by adopting AWS Lambda and Cognito, ensuring the client only pays for active user sessions.
- **Data-Driven Agility:** Implemented GraphQL via AWS AppSync, empowering the mobile frontend to dynamically request specific data sets for a faster, more responsive user experience.



Client Profile

Based in the United States, the client is a mission-driven technology company. They are dedicated to building digital "safe havens" that foster community support and mental wellness, specifically targeting Gen Z who seek peer-to-peer connection over traditional clinical interventions.

Challenges: Safety and Scalability in Social Audio

The client identified a gap in the market for a moderated, exclusive audio platform that could handle sudden traffic spikes without compromising user privacy.

- **Privacy & Safety:** Standard audio room apps lacked the exclusivity needed to protect female-identifying and non-binary users from external harassment.
- **Infrastructure Costs:** Traditional EC2-based hosting would have required 24/7 billing and manual scaling, which was financially unsustainable for a growing startup.
- **Complex User Management:** Building a custom, secure authentication and OTP verification system from scratch would have significantly delayed the time-to-market.
- **Mobile Data Efficiency:** Traditional REST APIs often resulted in over-fetching of data, slowing down the mobile experience for users on limited bandwidth.

QBurst Solution: Serverless Cloud Architecture

QBurst proposed a "Serverless-First" architecture using AWS Amplify to bridge the native Swift mobile app with backend cloud services. This approach ensured that the application remains lean, secure, and highly responsive.

Real-Time Audio Integration: Leveraged the Agora SDK to facilitate high-quality, audio-only support groups, ensuring low latency and high concurrency for peer-led sessions.

Automated Identity Management: Integrated AWS Cognito for secure user onboarding, including built-in support for OTP-based verification and encrypted user management.

Dynamic Data Layer: Implemented AWS AppSync (GraphQL) to allow the mobile client to precisely define data requirements, reducing payload sizes and improving app performance.

Compute on Demand: Used AWS Lambda for custom tasks and scheduled jobs, ensuring the system scales automatically during peak usage hours without manual intervention.

Secure Storage: Utilized Amazon S3 for multimedia content and DynamoDB as a highly available, NoSQL data store for user thoughts and experiences.

Key Features and Technical Highlights

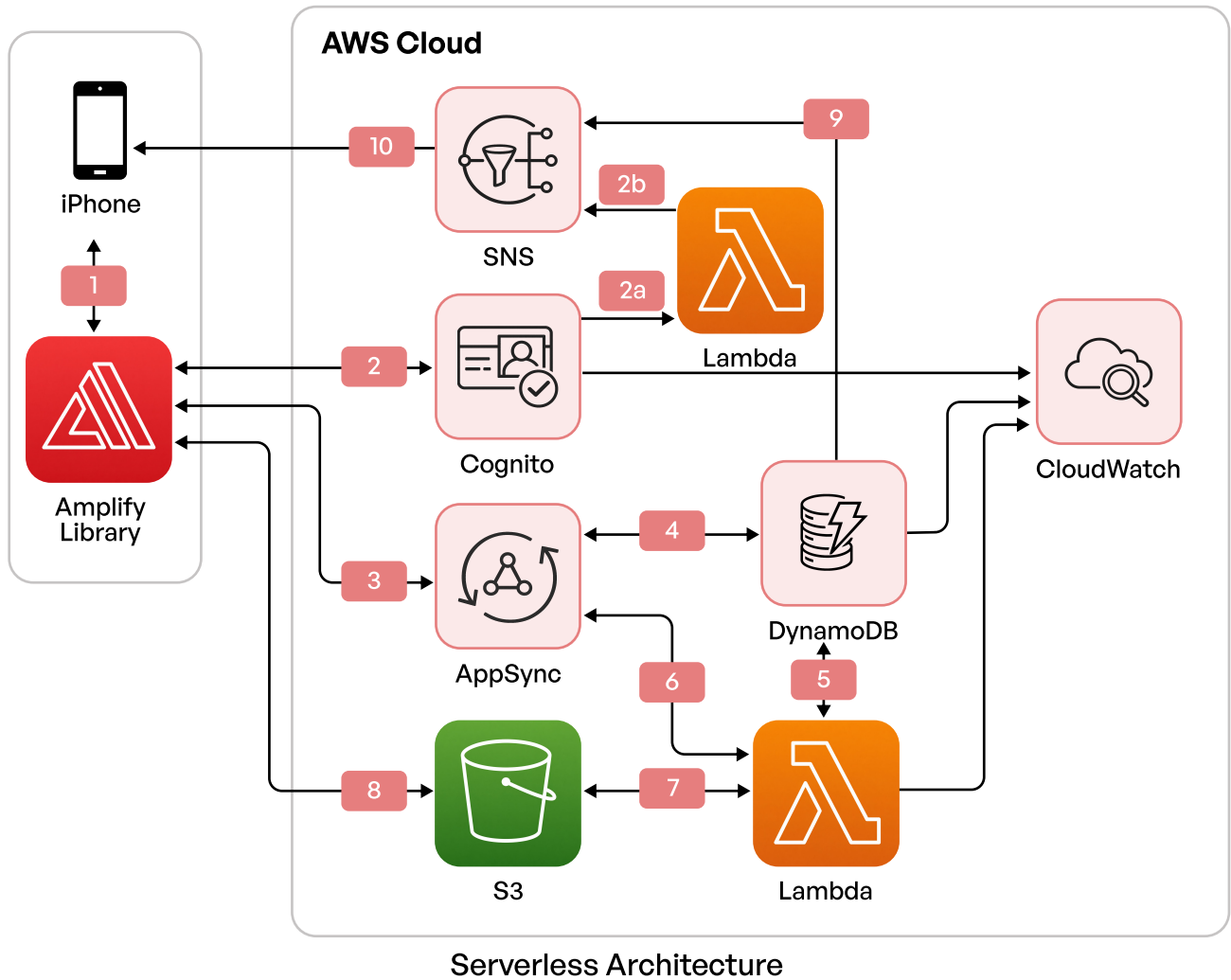
The platform is designed to be a high-engagement, low-friction community tool:

- **Peer-Led Audio Rooms:** Users can host or join audio-only discussions on various life topics in a safe, moderated environment.
- **Thought Sharing Feed:** A social platform for users to post personal experiences and thoughts in a non-judgmental space.
- **Secure OTP Verification:** Built-in mobile verification to ensure the integrity and exclusivity of the community.
- **CloudWatch Monitoring:** Integrated AWS CloudWatch for real-time monitoring of application health and Lambda execution logs.
- **Native Swift UI:** Developed using Swift and XCode to provide a fluid, high-performance experience on iOS devices.

Implementation Approach

By choosing a serverless model, our team focused on building community-centric features rather than managing infrastructure. The use of AWS Amplify allowed for rapid prototyping and deployment, while GraphQL provided the frontend flexibility needed to iterate on the user interface based on early feedback from the Gen Z community.

High Level Architecture Diagram



Impact

- **Significant Cost Savings:** The shift from EC2 to Lambda and Cognito eliminated round-the-clock billing, aligning costs directly with user growth.
- **High Trust & Safety:** The robust built-in security features of Cognito have provided users with the confidence to share sensitive experiences.
- **Limitless Scalability:** The auto-scaling nature of AWS serverless services ensures the app can handle viral growth or large community events without performance degradation.
- **Agile Evolution:** The GraphQL-based data layer allows the client to add new features and data fields with minimal backend reconfiguration.