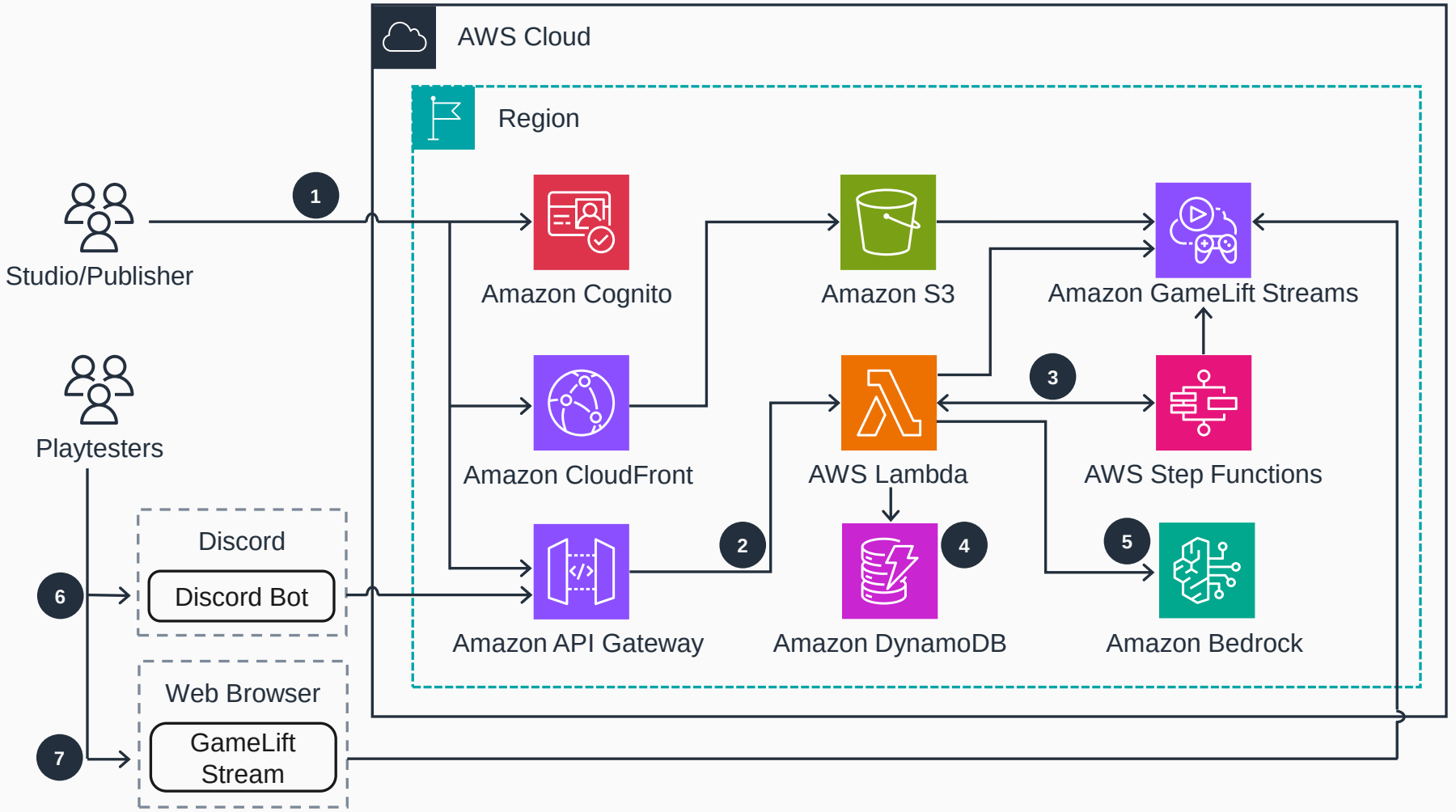
Guidance for Scalable Game Playtesting and QA on AWS

Scalable solution for playtesting games using GameLift Streams to pixel-stream and other backend components on AWS.



1. Users manage playtests through Single Page Application (SPA) hosted within **Amazon Simple Storage Service (Amazon S3)**. Management functions included (orchestrating playtest sessions, organizing playtesters, etc.) All authentication is handled by **Amazon Cognito**.
2. Managed **Amazon CloudFront** website utilizes **Amazon API Gateway** and **AWS Lambda** to call **Amazon GameLift Streams** to create new streaming client. Amazon S3 is used to store game builds for GameLift Streams.
3. **AWS Step Functions** are used to manage the various states of setting up new GameLift Streams.
4. Playtesting information (Observations, Observation Results, Playtesters, Playtest sessions, Studio, and Publisher) will be stored and retrieved from **Amazon DynamoDB**.
5. Users can leverage foundation models available through Amazon Bedrock to automatically summarize hundreds of playtester's feedback, derive sentiment and generate actionable insights from gameplay sessions
6. Play Testers interact with Discord via bot to receive playtesting URL's and to play test games. Players are able to self-register and initialize their playtest sessions without leaving discord.
7. Play Testers may receive playtesting URL outside of Discord where pixel streaming can be done within a browser.

Reviewed for technical accuracy September 24, 2025

AWS Reference Architecture