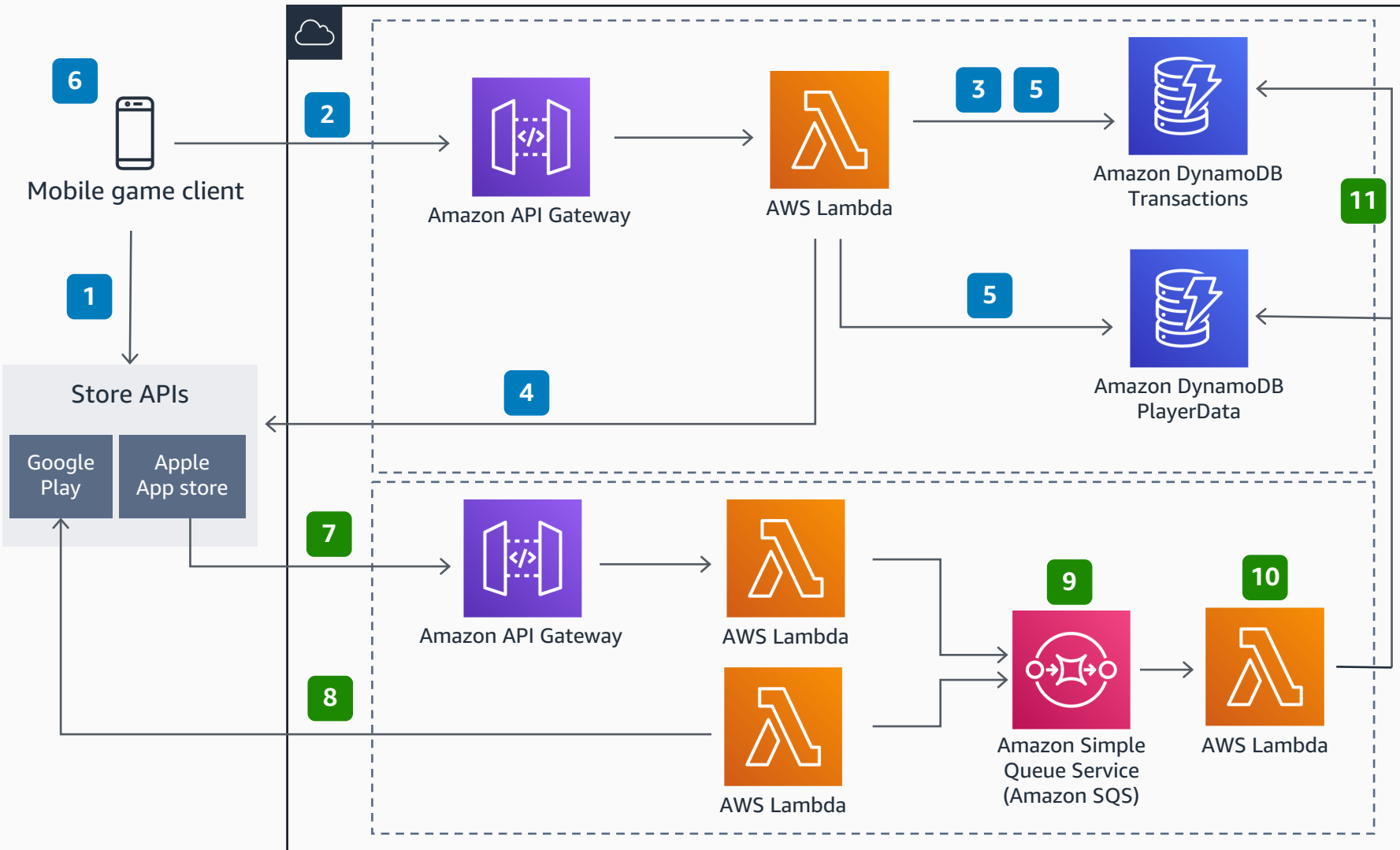


Mobile In-App Purchase Validation on AWS

An architecture for validating in-app purchases and managing refunds on AWS with a serverless backend



- 1 The mobile game client makes a purchase using the OS-specific SDK and stores the receipt locally.
- 2 The game client makes an API request to **Amazon API Gateway** to validate receipt and receive purchased items
- 3 The **AWS Lambda** function checks the "Transactions" table in **DynamoDB** to validate that the receipt is not yet used.
- 4 The **Lambda** function validates the receipt through the Google Play Store or Apple App Store API.
- 5 Once validated, the **Lambda** function adds items to the "PlayerData" table in **DynamoDB**, and adds the receipt to the "Transactions" table in **DynamoDB** with any additional metadata.
- 6 The client receives a success message. The client locally synchronizes the purchased items data, and deletes the local receipt copy.
- 7 **API Gateway** receives notifications on refunded transactions on iOS.
- 8 A scheduled **Lambda** function queries refunded transactions on Android.
- 9 Refunded transactions are pushed to an **Amazon SQS** queue by the **Lambda** functions.
- 10 A **Lambda** function handles the refund, including any additional actions like closing the player account.
- 11 A **Lambda** function updates the **DynamoDB** tables to remove items and mark the transaction refunded.

