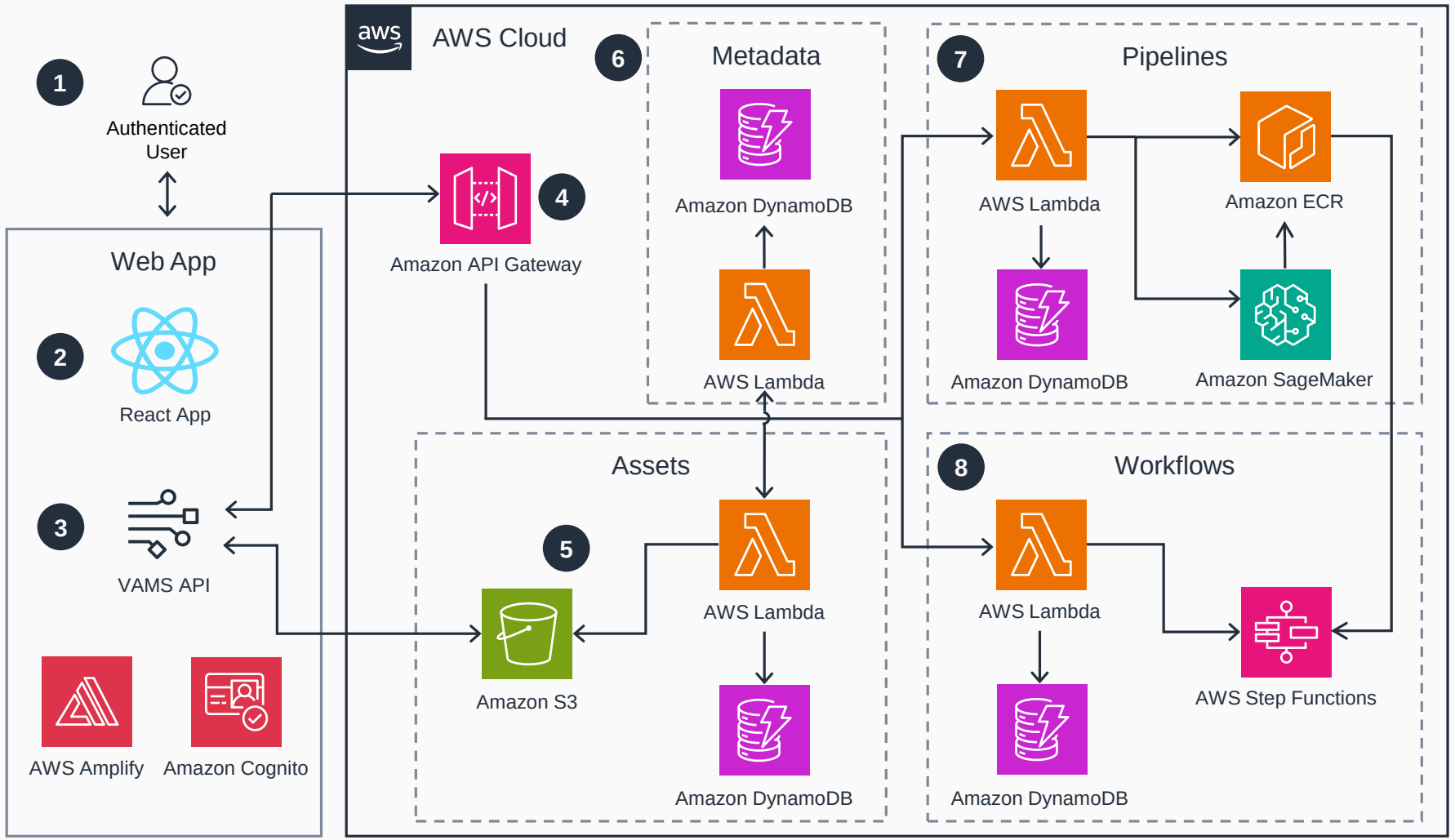
Guidance for Visual Asset Management System on AWS

This architecture diagram shows the standard system architecture for a VAMS deployment. AWS Cloud Development Kit (AWS CDK) is used to deploy a React-based web app for asset management and a serverless backend for storing and processing spatial data.



1. **Amazon Cognito user pools** handle user authentication, and **AWS Amplify** interfaces with the VAMS API.
2. Using the React.js-based content management system, upload assets to **Amazon Simple Storage Service (Amazon S3)**, apply metadata to the assets, process the assets using custom workflows and pipelines that the user creates, and generate download links for the assets.
3. The VAMS API uses the **Amplify** JavaScript library to handle client-side requests and results and route asset uploads and downloads from **Amazon S3**.
4. **Amazon API Gateway** routes server-side payloads and results, interfacing with **AWS Lambda** functions for handling assets, metadata, pipelines, and workflows.
5. The **Lambda** asset function delivers assets to **Amazon S3**, which stores and versions the data and stores asset data in **Amazon DynamoDB**.
6. **Lambda** data functions store asset metadata in a metadata **DynamoDB** table, allowing for extensible and customizable schema management.
7. **Lambda** functions deliver data to **Amazon SageMaker**, which creates containerized asset processing pipelines. **Amazon Elastic Container Registry (Amazon ECR)** stores the containerized pipelines while **DynamoDB** stores the pipeline data. Schedule pipelines into workflows using the React.js content management interface.
8. **Lambda** workflow functions direct workflow data to **AWS Step Functions** to orchestrate workflows and store them in **DynamoDB**. **Step Functions** automate the complex chaining of pipeline workloads by allowing the user to compose custom-ordered steps within the web app from their previously-defined pipelines. For example, a user could have a pipeline for file-type conversion, another for level of detail (LOD) generation, and a third for asset decimation, which they could order together into a single workflow.

Reviewed for technical accuracy October 19, 2023

AWS Reference Architecture