

Module objectives

By the end of this module, you will be able to:

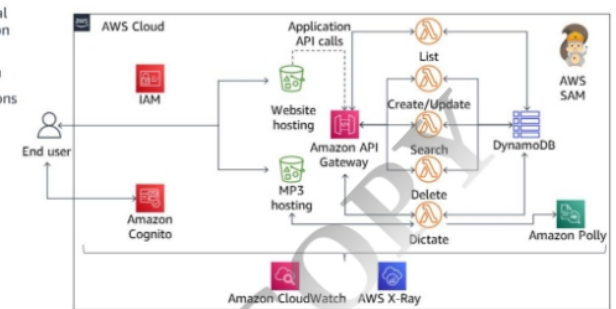
- Describe the architecture of the application you are going to build during this course
- List AWS services needed to build your web application
- Determine how to store, manage, and host your web application



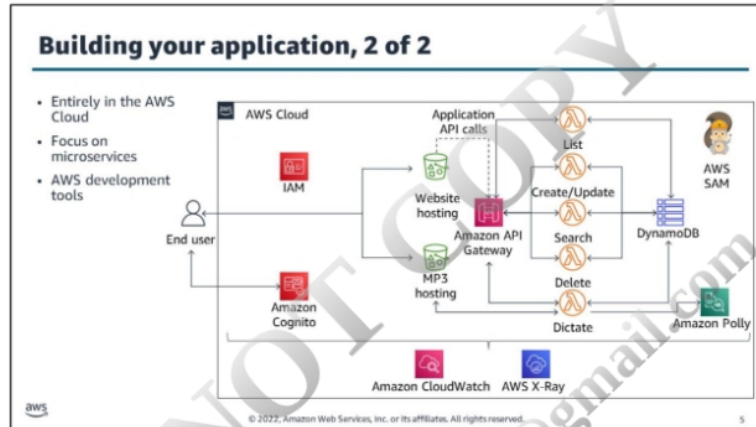
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Building your application, 1 of 2

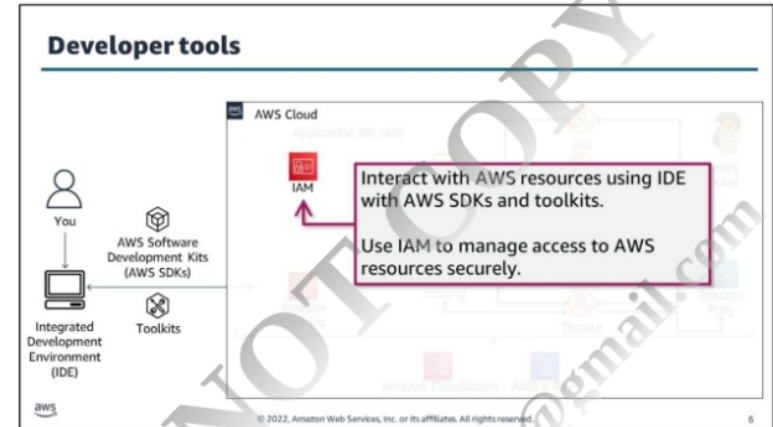
- Fully functional web application
- User authentication
- CRUD operations



This fully functional web application allows authorized users (using Amazon Cognito) to add custom notes that are converted to speech using Amazon Polly. Users are authenticated to a web portal where they can search, list, update, and delete audio notes. To build the text-to-speech application, you will need to understand how to connect to the AWS environment. Next, you will examine a high-level view of the architecture that illustrates how the AWS services interoperate.



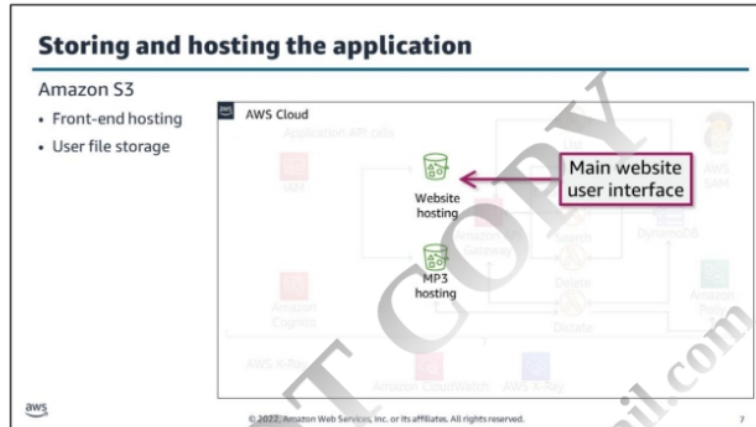
You are building a cloud-native application. This application uses serverless applications such as Amazon API Gateway, AWS Lambda, and Amazon DynamoDB. When you build your application, you will not rely solely on the AWS Management Console. You will learn to use AWS development tools such as AWS Software Development Kits (SDKs). As you build the application, you will take advantage of additional AWS development tools.



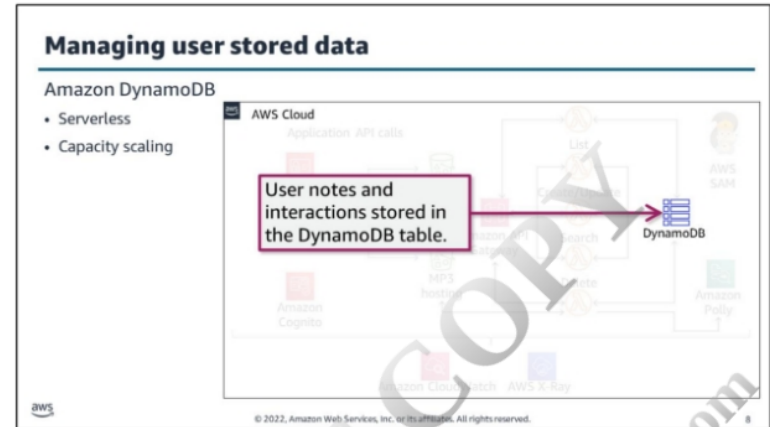
You can use several ways to build an application on AWS. The AWS Management Console supports all the functionality you need to gain access and manage your application. As a developer, you might prefer to interact with the AWS Cloud through a specific coding language or tool.

You will learn how to set up an Integrated Development Environment (IDE) so that it is configured with the right AWS Identity and Access Management (IAM) permissions. Using IAM, you can securely manage access to AWS resources. By setting up these environment configurations, you can call AWS services through the AWS Software Development Kits (AWS SDK) and toolkits. Developers can use these tools to build applications without using the AWS Management Console interface.

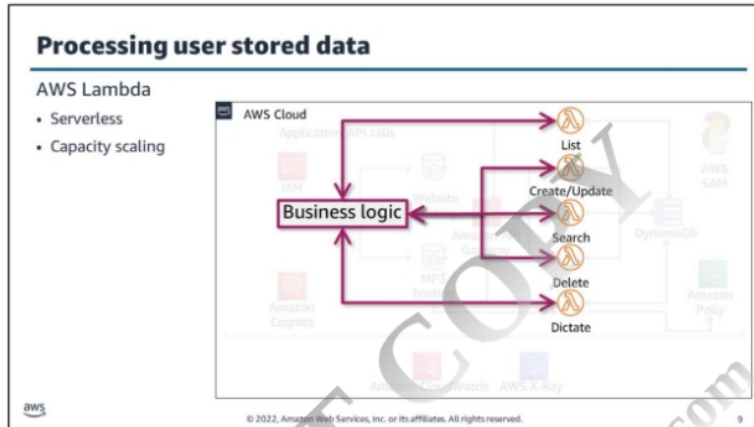
You will also learn about IAM account permissions and configuring necessary security profiles in the application development lifecycle.



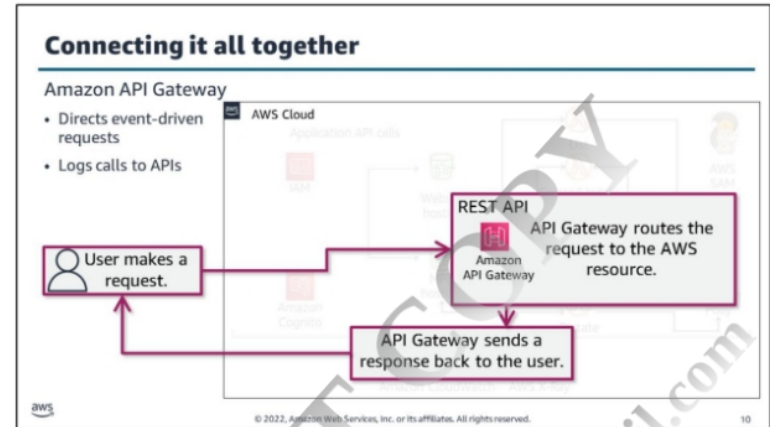
Amazon Polly stores the application's front end and the user's MP3 files in Amazon Simple Storage Service (Amazon S3). Although AWS offers other viable storage solutions, Amazon S3 provides a streamlined solution that satisfies two major needs: hosting and file storage. The application's storage solution also integrates well with AWS compute solutions for create, read, update, and delete (CRUD) operations.



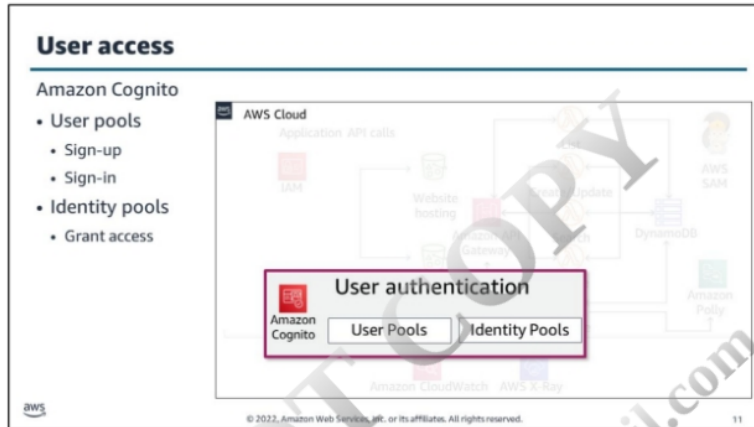
As your users add, read, update, and delete their notes, your application needs a database to store these interactions. Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. You can use this scalability to meet your application's fluctuating capacity needs. DynamoDB offers a serverless solution that does not require you to install or maintain any software.



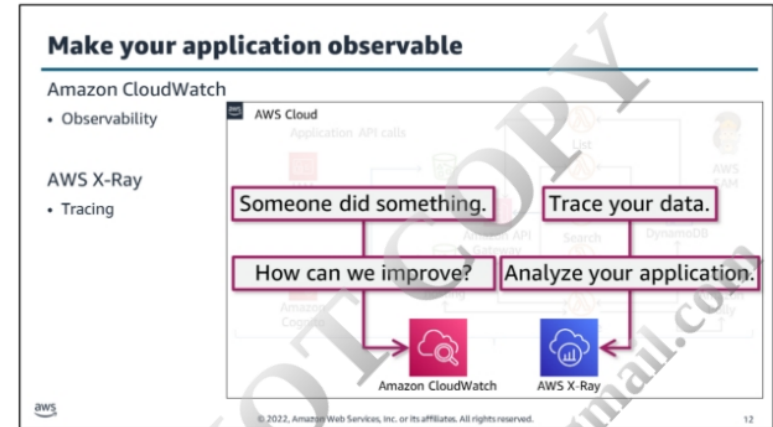
Although an Amazon Elastic Compute Cloud (Amazon EC2) instance will solve the application's compute needs, it requires continual monitoring and maintenance. To streamline application development while keeping costs down, you will create multiple AWS Lambda functions to power the application's CRUD operations. These Lambda functions offer a workload-aware scaling solution that can be configured and maintained in your development environment.



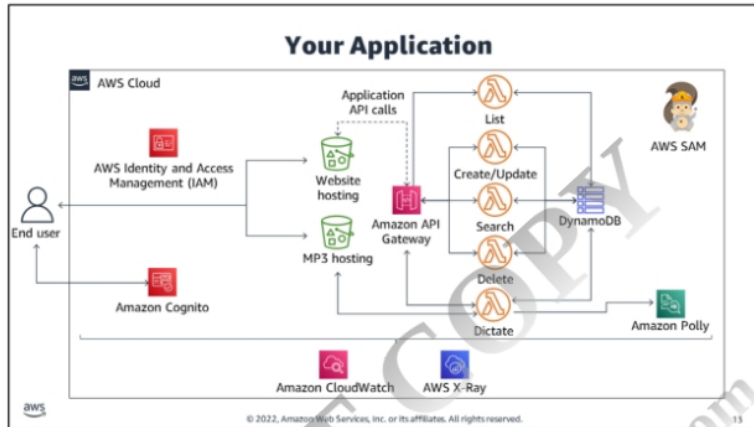
Amazon API Gateway connects all the application's services together. API Gateway is used to direct the event-driven requests to and from your users to your compute, database, and storage services. API Gateway simplifies the process to create, publish, maintain, monitor, and secure an API that suits your application's specific tasks.



Your users need secure authentication and authorization to the application. The two main components of Amazon Cognito are user pools and identity pools. *User pools* are user directories that create built-in sign-up and sign-in options for your application users. Use *identity pools* to grant user access to other AWS services. You can use identity pools and user pools separately or together. You will learn about tools, such as Amazon Cognito, to help streamline access control for users, focusing on **user pools** for sign-up and sign-in.



An application requires monitoring and troubleshooting services to help correct any issues that might interfere with user interactions or future development updates. You must be able to monitor and troubleshoot your application, and Amazon CloudWatch provides you with actionable insights. The application will also employ AWS X-Ray to assist with analyzing and debugging your distributed application.



In this module, you were introduced to the application architecture and individual components. You will continue to follow this architecture diagram throughout the course as you proceed through the application development process. Next, you will begin your development journey with a conversation about developer tools necessary to build this application.

