DocumentCast Deployment Guide

(AWS Reference Architecture & Implementation Manual)

Contact:

📧 documentcast@globant.com
🌐 www.globant.com

Document Version	1.0
Release Date	November 2025
Author(s)	Ritesh Menon, Jaydeep Sheth
Confidentiality	Globant Confidential – For Authorized Use Only

Document Summary

This guide provides deployment, configuration, and maintenance instructions for **DocumentCast**. This AI-powered solution transforms static content (PDFs, manuals, and reports) into interactive, multilingual audio podcasts using AWS services such as ECS Fargate, Cognito, and Bedrock.

It includes:

- Solution architecture and security design.
- Step-by-step CloudFormation deployment instructions.
- Maintenance, cost, and support guidance.
- Compliance and operational best practices aligned with AWS Foundational Technical Review (FTR).

Disclaimer

Globant

Table of Contents

1. Introduction

DocumentCast is an AI-powered solution that converts static enterprise documents into natural, human-sounding audio content within minutes. It integrates advanced Optical Character Recognition (OCR) capable of processing over 1,000 pages, including charts and tables, with multilingual voice synthesis and coordinated multi-agent orchestration to deliver consistent tone, pacing, and brand personality.

This capability enables organizations to automatically repurpose whitepapers, training materials, and reports into scalable, brand-aligned podcasts for global audiences at a cost of under USD 1 per iteration.

1.1. Use Cases for the Solution

The AI Podcast Generator transforms static PDF documents into engaging, conversational audio podcasts. Key use cases include:

- **Content Repurposing:** Convert whitepapers, reports, and articles into an audio format.
- **Corporate Training & E-Learning:** Transform training manuals into on-the-go learning modules.
- **Accessibility:** Provide an audio version of written materials for visually impaired users.
- **Educational Material:** Convert academic papers and study guides into digestible podcast episodes.

Features:

- **Advanced OCR** that can process 1,000+ pages, including charts, tables, and images.
- **Voice cloning and novelty** for authentic, brand-aligned audio.
- **Multilingual support**, enabling content to reach global audiences.
- **Multi-agent orchestration**, which ensures content is not just read aloud but understood, scripted, and delivered with personality.
- Support to choose the right Large Language Models and Audio models.

And most importantly, every podcast iteration costs **less than $1**, making DocumentCast the most cost-effective way to turn static knowledge into a living, scalable audio channel.

1.2. Solution Overview

Customers deploy DocumentCast using a single AWS CloudFormation stack that automatically provisions all required infrastructure components. The template creates a multi-Availability Zone Virtual Private Cloud (VPC) that includes an Application Load Balancer (ALB), Amazon ECS on AWS Fargate, Amazon Cognito for user authentication, Amazon S3 for object storage, AWS Secrets Manager for credential management,

Amazon ElastiCache Serverless for caching, and Amazon CloudWatch for monitoring and alerting. The provided CloudFormation template provisions the following resources:

- **Networking:** A new Amazon VPC with public and private subnets across two Availability Zones, an Internet Gateway, and a NAT Gateway.
- **Load Balancing:** An Application Load Balancer (ALB) serving as the public entry point.
- **Authentication:** Amazon Cognito for user pool management and application authentication integrated with the ALB.
- **Compute:** Two AWS Fargate services within an Amazon ECS Cluster:
  - `Frontend Service`: Runs Nginx to serve the React user interface.
  - `API Service`: Runs the Python FastAPI backend application.
- **Storage:** An Amazon S3 bucket for all customer-uploaded documents and generated audio files.
- **Cache:** An Amazon ElastiCache Serverless (Valkey) cache for application state management.
- **Security & Configuration:**
  - AWS Secrets Manager for secure credential storage.
  - AWS IAM Roles with least-privilege permissions.
  - AWS Security Groups for granular network control.
  - AWS Certificate Manager (ACM) for provisioning SSL/TLS certificates.
- **Monitoring:** Amazon CloudWatch Log Groups and AWS X-Ray for tracing.

1.3. Solution Deployment

The standard deployment is configured for high availability across multiple Availability Zones within a single AWS Region. The CloudFormation template also supports optional single-Availability Zone deployments for development and testing, as well as multi-Region disaster recovery configurations through parameter settings.

1. Single-AZ (Development/Test): Simplified and cost-effective for internal validation.
2. Multi-AZ (Production – Default): High-availability configuration across two AZs.
3. Multi-Region (Optional DR): Supports cross-region replication via StackSets or automation pipelines.

Each model uses identical networking, IAM, and monitoring components, ensuring consistent compliance and operational posture.

1.3.1. Deployment Time

Typical end-to-end deployment completes in 15 – 25 minutes, depending on region and resource quotas:

Phase	Approx. Time	Description
Network provisioning	~10 min	Creates VPC, Subnets, Gateways
ECS service startup	~8 min	Registers task definitions, pulls images
Validation & DNS	~5 min	Confirms ALB health and domain CNAME

1.3.2. Supported AWS Regions

DocumentCast can be deployed in any AWS Region supporting ECS Fargate, ALB, S3, ElastiCache Serverless, Cognito, and Secrets Manager.
Validated production regions (Oct 2025):
us-east-1, us-west-2, eu-west-1, ap-southeast-1, ap-south-1.
Future updates will automatically validate region compatibility through CloudFormation Fn::GetAZs.

---

2. Prerequisites and Requirements

The solution requires an active AWS account with permissions to deploy AWS CloudFormation stacks and manage associated services, including Amazon ECS, networking resources, and AWS Identity and Access Management (IAM) roles. Deployment prerequisites include AWS CLI version 2, Docker Desktop, and Git for building and publishing container images to Amazon ECR.

A registered domain name and an AWS Certificate Manager (ACM) certificate are required to enable HTTPS access through the Application Load Balancer (ALB). All technical prerequisites and dependencies are documented to ensure the deployment executes successfully without manual intervention or configuration errors.

2.1. Technical Prerequisites

To deploy DocumentCast, customers need:

- An active AWS account with permissions to create VPC, ECS, ALB, Cognito, and Secrets Manager.
- An IAM user or role with AdministratorAccess (or equivalent custom policy).
- AWS CLI v2, Docker Desktop, and Git installed.

- A registered domain and an ACM-issued TLS certificate for HTTPS.

These prerequisites ensure an automated, end-to-end deployment without manual infrastructure setup.

2.2. Prerequisite Skills

Deployment engineers (from **Globant** or the **customer team**) should have practical experience with the AWS ecosystem and containerized application operations. A working knowledge of the following areas is required:

- Core AWS services include **VPC**, **IAM**, **ECS (Fargate)**, **S3**, **CloudFormation**, and **CloudWatch**.
- Building and pushing **Docker images** and integrating with **Amazon ECR**.
- Monitoring, troubleshooting, and scaling container workloads using **CloudWatch**, **ECS**, and related observability tools.
- Understanding of **security best practices** such as IAM least-privilege policies, KMS encryption, and Secrets Manager usage.

To ensure a consistent and reliable deployment experience, it is recommended that engineers hold one or more of the following AWS Certifications:

- **AWS Certified Solutions Architect – Associate**,
- **AWS Certified SysOps Administrator – Associate**, or **AWS Certified Developer – Associate**.
- **AWS Certified Security – Specialty** (Optional) **-** For advanced operations and compliance

These competencies and certifications enable teams to deploy and manage DocumentCast confidently and independently with minimal AWS support.

2.3. Create a third-party application API Key

2.3.1. Creating Your Globant Enterprise AI

This solution optionally integrates with Globant Enterprise AI. To enable this functionality, you must first register for an account at www.globant.com/globant-enterprise-ai. Upon successful registration, you will be provided with the following required credentials, which should be used as parameters during deployment:

1. **GEAI_API_BASE** – Base URL for calling the GEAI core API service.
2. **SAIA_TTS_MODEL** – Specifies the Speech-AI voice synthesis model used for podcast generation.
3. **GEAI_API_KEY** – Authentication key granting access to GEAI API endpoints.
4. **SAIA_API_KEY** – API key used to access Speech-AI (SAIA) TTS and OCR services.
5. **GEAI_MODEL_NAME** – Defines which GEAI language model is invoked for text generation.

6. **SAIA_TTS_API_URL** – Endpoint URL for Speech-AI text-to-speech requests.
7. **SAIA_OCR_MODEL** – Identifies the OCR engine used for document text extraction.
8. **SAIA_API_ENDPOINT** – Base endpoint for all Speech-AI service interactions.

2.3.2. Other Keys:

1. **AWS_DEFAULT_REGION** – AWS Region where the stack and resources are deployed.
2. **MISTRAL_API_KEY** – API key used to authenticate with Mistral AI models for content synthesis. Since Mistral is not currently available through AWS Bedrock, this integration is managed as an external third-party service.
3. **JWT_SECRET_KEY** – Secret used to sign and verify JSON Web Tokens for authentication.
4. **MINIO_BUCKET_NAME** – Name of the MinIO (S3-compatible) bucket storing processed files.

2.2.3. Creating Your Mistral API Key:

1. Go to console.mistral.ai to sign up or log in.
2. Create a workspace for your projects if prompted.
3. Go to the "Billing" section to add payment information.
4. Navigate to the "API keys" page within your workspace.
5. Click the "Create new key" button to generate your key.
6. Optionally, give your key a name for easy identification.
7. Copy your new API key immediately and store it securely.

2.4. Environment Configuration

The CloudFormation stack provisions a **self-contained environment** with no dependency on existing infrastructure.
Key configurations:

- Dedicated VPC (10.0.0.0/16) with two public and two private subnets across two AZs.
- Public subnets host the ALB and NAT Gateway; private subnets host ECS tasks.
- DNS and TLS are handled via Route 53 and ACM.
- Resource tags (`App=DocumentCast`, `Environment`, `Owner`) applied for cost tracking.

---

3. Architecture

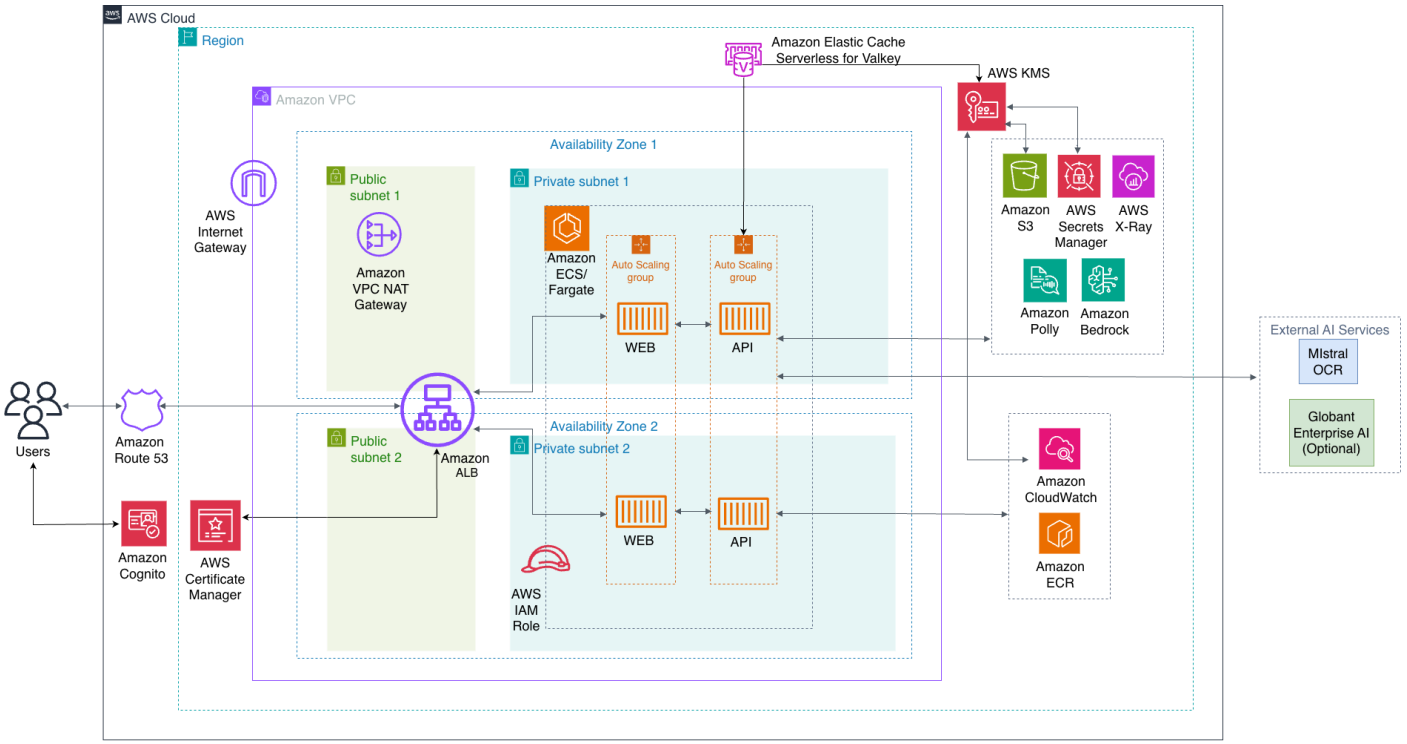
3.1. Architecture Diagram

The following diagram (Figure 1) illustrates DocumentCast's AWS architecture using official AWS icons.

Traffic enters through an HTTPS ALB secured by ACM, authenticates via Cognito, and routes to ECS Fargate tasks running the API and Frontend services in private subnets. Data flows to S3 for storage, ElastiCache for caching, and Secrets Manager for credentials.

The API service communicates with S3, ElastiCache, and Secrets Manager, while CloudWatch and X-Ray collect logs and traces.

This design ensures high availability, fault isolation, and end-to-end TLS encryption.



Data Flow:

1. An **Authorized User** accesses the application via a domain name.
2. The traffic is routed to the public **Application Load Balancer (ALB)** over HTTPS.
3. The ALB Listener has rules that integrate with **Amazon Cognito**. If the user is unauthenticated, they are redirected to the Cognito Hosted UI for login.

4. Upon successful authentication, the ALB forwards requests to the **Frontend** ECS service in the private subnets.
5. The Frontend service communicates with the **API** service. The ALB routes these requests, validating the user's session with Cognito before forwarding to the API service.
6. The API service interacts with AWS managed services (ElastiCache, S3, and Secrets Manager) and external AI services.
7. Outbound traffic to external APIs is routed through the **NAT Gateway**.

3.2. Network Architecture

- The VPC is designed for secure isolation and scalability:
- Two Availability Zones with public and private subnets.
- Security Groups restrict inbound/outbound traffic to necessary ports (443 only public).
- NAT Gateway enables outbound API access from private tasks.
- S3 Block Public Access is disabled to prevent accidental data exposure.

3.3. Integration Points

- **AWS Services:** The solution integrates with Amazon Cognito for authentication and other services like S3, ElastiCache, and Bedrock via the AWS SDK using secure IAM roles.
- **Third-Party APIs:** The application integrates with external AI services (e.g., Mistral and Globant Enterprise AI). Communication is conducted over HTTPS, with API keys managed by AWS Secrets Manager.

---

4. Security

4.1. IAM and Access Management

The solution does **not** require the use of the AWS account root user. All access is managed through two dedicated IAM Roles:

- **EcsTaskRole (`{ProjectName}-ecs-task-role`):** Grants the application itself permissions to interact with other AWS services.
- **EcsTaskExecutionRole (`{ProjectName}-ecs-task-execution-role`) :** Grants the ECS agent permissions to pull container images and retrieve secrets.

The `EcsTaskRole` is configured to follow the **principle of least privilege**. Permissions are scoped down to necessary actions on specific resources created by the stack, avoiding wildcards (`*`) on resources wherever possible.

The deployment defines two IAM roles with distinct purposes. The EcsTaskExecutionRole allows Amazon ECS to pull container images from Amazon ECR and retrieve credentials from AWS Secrets Manager during task initialization. The EcsTaskRole is used by the application at runtime to access authorized AWS services such as Amazon S3, Amazon ElastiCache, and Amazon CloudWatch in accordance with least-privilege permissions.

Refer to the “13.2. ECS Tasks Execution" and “13.3. ECS Tasks Role” for more details.

4.2. Secrets Management

All sensitive application and infrastructure configurations, including third-party API credentials, AWS service parameters, and application security keys, are securely managed within AWS Secrets Manager under a single resource named DocumentCastAppSecrets.

Secret Variable	Description
MISTRAL_API_KEY	API key used to authenticate with Mistral AI models for content synthesis. This integration is external to AWS Bedrock and requires its own subscription.

GEAI_API_KEY	Authentication key for Globant Enterprise AI (GEAI) model endpoints.
SAIA_API_KEY	API key for Speech-AI (SAIA) text-to-speech and OCR services.
GEAI_API_BASE	Base URL for GEAI API service calls.
GEAI_MODEL_NAME	Specifies which GEAI language model is used during content generation.
SAIA_TTS_MODEL	Speech-AI text-to-speech model used for generating podcast audio.
SAIA_TTS_API_URL	Endpoint URL for Speech-AI text-to-speech operations.
SAIA_OCR_MODEL	Model identifier for Speech-AI OCR processing.
SAIA_API_ENDPOINT	Root endpoint for Speech-AI service interactions.
AWS_DEFAULT_REGION	Region where the application stack is deployed, and AWS services are accessed.
USE_AWS_S3	Indicates whether AWS S3 is enabled as the storage backend.
AWS_BEDROCK_MODEL_ID	Specifies the Bedrock foundation model used by the application, if Bedrock integration is enabled.

4.3. Public Resources & Data Security Controls

The DocumentCast solution exposes only one public-facing component: the Application Load Balancer (ALB), which securely terminates HTTPS traffic using an ACM TLS certificate. All other resources are private and accessible only within the VPC.

The Amazon S3 bucket provisioned by the stack is configured with S3 Block Public Access, preventing unintended exposure of stored documents and generated audio files. Access to S3 objects is restricted to IAM roles used by the ECS tasks, adhering to the principle of least privilege. Similarly, ElastiCache Serverless, Secrets Manager, and all internal application components are deployed in private subnets and cannot be accessed from the public internet.

These controls ensure that no customer-sensitive data or internal system components are publicly exposed, and that only authorized AWS services within the deployment can access protected data.

4.4. Data Encryption

4.4.1 Customer Managed KMS Key (CMK)

The DocumentCast solution uses a single **AWS KMS Customer Managed Key** created during deployment. This CMK is used internally by AWS services provisioned by the CloudFormation stack to encrypt all persistent and sensitive data:

- **Amazon S3** – uploaded PDFs and generated audio files are encrypted using SSE-KMS
- **AWS Secrets Manager** – all application secrets stored under `DocumentCastAppSecrets`
- **Amazon ElastiCache Serverless (Valkey)** – cache data and automated snapshots
- **CloudWatch Logs** – application log streams
- **X-Ray trace data**, when applicable

IAM access to this CMK is restricted to the ECS task execution and task roles defined in the deployment, following least-privilege principles. The CMK **must be retained** for recovery operations because previously encrypted data cannot be decrypted if the key is deleted or replaced.

4.4.2 SSH Keys

The solution does **not** require SSH key pairs.
DocumentCast runs entirely on **AWS Fargate**, a serverless container environment where no EC2 instances are created, and no SSH access is required. Administrative access is performed using IAM roles, ECS Exec, CloudWatch Logs, and Secrets Manager, all of which provide audited, keyless access patterns.

Refer to the “13.4. KMS” for more details.

4.5. Sensitive Data Storage

Customer-sensitive data in the DocumentCast solution, including uploaded documents, generated audio files, user profile information, and authentication metadata, is securely stored in AWS-managed services designed for durability, isolation, and encryption.

- **Primary Data Storage:**
  All uploaded documents (e.g., PDFs) and generated podcast files (MP3s) are stored in a **private Amazon S3 bucket** created by the CloudFormation stack.

The bucket enforces **S3 Block Public Access**, and all objects are encrypted at rest using **AWS KMS (SSE-KMS)** with a dedicated customer-managed key.

- **Cache Data:**
  Transient application state (e.g., session tokens, temporary AI results) is stored in Amazon ElastiCache Serverless (Valkey). The ElastiCache cluster is deployed inside the application VPC using private subnets and security groups, ensuring all data remains on the internal AWS network and is inaccessible from the public internet
  These data are encrypted in transit and at rest, with automatic backup snapshots stored in AWS-managed infrastructure.
- **Secrets and Credentials:**
  All sensitive configuration data, such as API keys, JWT secrets, and model identifiers, is stored in **AWS Secrets Manager** under the centralized secret `DocumentCastAppSecrets`, encrypted using the same KMS key and accessed only by ECS tasks through least-privilege IAM roles.
- **User Authentication Data:**
  Amazon Cognito User Pools manage user credentials and session tokens. The application never stores or processes plaintext passwords.

No customer data is persisted on container instances, ephemeral storage, or local disks. Data are transmitted exclusively over **TLS 1.2+** connections.

4.6. IMDSv1

The solution runs on AWS Fargate, which does not use the traditional EC2 IMDS. The credential mechanism used by Fargate meets the security objectives of this requirement by default.

---

5. Cost

5.1. Billable Services

Deploying the DocumentCast solution incurs costs from the following **mandatory AWS services** that form the core infrastructure:

- **AWS Fargate** – For running the API and Frontend containerized services.
- **Application Load Balancer (ALB)** – For HTTPS traffic routing and Cognito authentication integration.
- **NAT Gateway** – For secure outbound internet access from private subnets.
- **Amazon S3** – For storing uploaded documents and generated audio files.
- **Amazon ElastiCache Serverless (Valkey)** – For caching and state management.
- **Amazon Cognito** – For user authentication and token management.
- **AWS Secrets Manager** – For storing application and integration secrets.
- **Amazon CloudWatch & AWS X-Ray** – For monitoring, logging, and distributed tracing beyond the free tier.

Additionally, customers can enable **AI integrations** that may introduce separate, usage-based costs:

- **Mistral AI** – For external content synthesis and OCR model inference.
- **Amazon Bedrock or Globant Enterprise AI (GEAI)** – For large language model (LLM) inference and speech synthesis.
- **Amazon Polly** or **Third-Party Voice Models (via GEAI)** – For advanced text-to-speech generation.

5.2. Cost Model and Licensing

DocumentCast is offered through AWS Marketplace as a **flat-rate monthly subscription**, priced at **$10 per subscription per month**, regardless of the number of users within the customer's organization. Globant does not charge per-user fees, per-request fees, or usage-based licensing for the DocumentCast application.

All AWS-managed services consumed by DocumentCast are billed directly through the customer's AWS account under AWS's standard pay-as-you-go model. Charges may include compute, storage, data transfer, caching, monitoring, and authentication services, such as:

- Amazon ECS Fargate
- Application Load Balancer (ALB)

- Amazon S3
- Amazon ElastiCache Serverless (Valkey)
- AWS Secrets Manager
- Amazon Cognito
- Amazon CloudWatch and AWS X-Ray

These AWS costs vary depending on real workload usage (number of documents, file sizes, user traffic, etc.).

In addition to AWS service charges, Customer will incur the cost for the following external API services such as:

- **Mistral AI** (Mandatory Service) – Billed separately through the Mistral platform (https://console.mistral.ai).
  Pricing is usage-based according to their public pricing page: https://mistral.ai/pricing#api-pricing.
  DocumentCast consumes this service using the customer-provided **MISTRAL_API_KEY**.
  Mistral operates outside AWS Bedrock and requires its own account.

- **Globant Enterprise AI (GEAI)** – Customers may choose to enable this service if they wish to use it. Charges may apply for LLM inference and SAIA text-to-speech. These services are billed per request according to GEAI's pricing model. The detailed pricing will be available upon contacting the GEAI team at the following link.
  Access requires customer-supplied credentials such as **GEAI_API_KEY**, **SAIA_API_KEY**, and model identifiers.

All related credentials (MISTRAL_API_KEY, GEAI_API_KEY, SAIA_API_KEY, etc.) are stored securely in **AWS Secrets Manager** under the consolidated secret **DocumentCastAppSecrets**, encrypted with the customer-managed KMS key, and retrieved by ECS tasks at runtime.

Example:
Mistral for OCR and GEAI for text-to-speech, the customer's AWS bill will include infrastructure costs (Fargate, S3, ALB, etc.).
Mistral and **GEAI** usage charges will appear separately under the customer's respective vendor accounts, as DocumentCast does not add any additional fees beyond the $10/month Marketplace subscription.

Customers are responsible for managing their own API keys and usage monitoring for third-party AI providers.

---

6. Sizing

6.1. Resource Provisioning

The CloudFormation template exposes parameters for task CPU and memory configuration, allowing customers to adjust resource allocations based on workload requirements. Recommended defaults are 512 CPU units and 1024 MiB memory for the API service, and 256 CPU units and 512 MiB memory for the frontend service. Customers can enable Amazon ECS Service Auto Scaling to handle variable traffic conditions.

The CloudFormation template includes parameters to customize resource allocation for the compute tasks:

- `ApiTaskCpu` & `ApiTaskMemory`
- `FrontendTaskCpu` & `FrontendTaskMemory`

---

7. Deployment Steps

7.1. Step-by-Step Instructions

Step 1: Prerequisites

Ensure all items from Section 2 (Page number 6) – Prerequisites and Requirements are completed, including domain registration, ACM certificate, and IAM permissions.

Step 2: Access the AWS Marketplace Listing

1. Navigate to the **AWS Marketplace** and search for **"DocumentCast – AI Podcast Generator."**
2. Choose **Continue to Subscribe** and accept the license terms.
3. Select **Continue to Configuration** and choose the appropriate AWS Region.
4. Under *Fulfillment Option*, select **CloudFormation Template**.
5. Click **Continue to Launch** → **Launch CloudFormation** to automatically open the CloudFormation console with the template pre-loaded and pre-configured image URIs.

Note: No image build or push steps are required the API and Frontend container images are securely hosted and maintained by Globant through AWS Marketplace.

Step 3: Deploy the CloudFormation Stack

1. In the CloudFormation console, confirm the template and parameters.
2. Review and, if necessary, provide values for the following required parameters:

CloudFormation Parameters Guide

Parameter Name	Description	Setting / Default Value	Example / Notes
ProjectName	A unique name for your project, used as a prefix for naming AWS resources.	documentcast	my-podcast-app
Environment	The deployment environment (e.g., dev, staging, prod), used for resource tagging.	dev	prod
Owner	The team or individual responsible for this	platform-team	marketing-dept

	deployment, used for tagging.		
AwsDefaultRegion	The AWS Region for deployment, passed to the application for SDK configuration.	(Required)	us-east-1, eu-west-1
VpcCidr	The overall IP address range for the new Virtual Private Cloud (VPC). eg.10.0.0.0/16	(Required)	Change only if this range conflicts with your existing network.
PublicSubnetCidr1	The IP range for the first public subnet (used for the ALB and NAT Gateway). eg. 10.0.1.0/24	(Required)	Must be within the VpcCidr range.
PublicSubnetCidr2	The IP range for the second public subnet in a different Availability Zone. eg. 10.0.2.0/24	(Required)	Must be within the VpcCidr range.
PrivateSubnetCidr1	The IP range for the first private subnet (used for application containers). eg. 10.0.3.0/24	(Required)	Must be within the VpcCidr range.
PrivateSubnetCidr2	The IP range for the second private subnet in a different Availability Zone. eg. 10.0.4.0/24	(Required)	Must be within the VpcCidr range.
ApiImage	The full ECR image URI for the backend API service.	(Required) Default is set by Marketplace	This parameter is pre-filled with the correct image URI
FrontendImage	The full ECR image URI for the frontend web service.	(Required) Default is set by Marketplace	This parameter is pre-filled with the correct image URI
ApiTaskCpu	CPU units for each API task (1024 = 1 vCPU).	512	Increase if your file size and number of users are higher.
ApiTaskMemory	Memory (in MiB) for each API task.	1024	Increase if your file size and number of users are higher.
FrontendTaskCpu	CPU units for each frontend task.	512	Defaults are typically sufficient.

FrontendTaskMemory	Memory (in MiB) for each frontend task.	1024	Defaults are typically sufficient.
AcmCertificateArn	The full ARN of your SSL certificate from AWS Certificate Manager (ACM).	(Required)	arn:aws:acm:us-east-1:123456789012:certificate/xxxx-xxxx-xxxx Important: Must be in the same region as the deployment.
DomainName	The custom domain name to access the application.	(Required)	documentcast.mycompany.com
UserPoolDomain	A globally unique prefix for your Cognito-hosted login page URL.	(Required)	documentcast-app
HostedZoneId	The Route 53 Hosted Zone ID. If provided, a DNS A-record will be created automatically.	(Optional)	If blank, you must manually create a DNS record.
S3BucketName	A globally unique name for the S3 bucket. Must be all lowercase.	(Required)	
ExpirationInDays	Days after which user files will be automatically deleted from the S3 bucket.	365	Supports data retention policies (e.g., for GDPR).
MistralApiKey	Your API key for the Mistral service.	(Required)	Do not use personal or trial keys in a production
SaiaApiKey	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise subscription
GeaiApiKey	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise subscription
SaiaTtsApiUrl	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise subscription
SaiaApiEndpoint	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise subscription
SaiaOcrModel	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise

			subscription
GeaiModelName	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise subscription
GeaiApiBase	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise subscription
SaiaTtsModel	Globant Enterprise API Keys & Secrets	(Optional)	Required only if using a Globant enterprise subscription
AwsBedrockModelId	The specific model ID for the Bedrock foundation model you intend to use.	(Required)	amazon.nova-pro-v1:0
UseAwsLlm	Set to true to enable the use of Amazon Bedrock for language model processing.	true	If false, the application relies on Globant enterprise models.
ApiMinCapacity	The minimum number of tasks the API service will maintain.	2	More than 2 for high availability
ApiMaxCapacity	The maximum number of tasks the API service can scale out to under heavy load.	10	
ApiTargetCpuUtilization	The average CPU utilization (%) that will trigger a scaling event.	70%	
FrontendMinCapacity	The minimum number of tasks the frontend service will maintain.	2	
FrontendMaxCapacity	The maximum number of tasks the frontend service can scale out to.	10	
FrontendTargetCpuUtilization	The average CPU utilization (%) that will trigger a scaling event for the frontend.	70%	
AllowedCidrForAlb	The IP range allowed to access the Application Load Balancer.	0.0.0.0/0	0.0.0.0/0 allows all public internet traffic.

CreateEcsServiceLinkedRole	Set to false only if the AWSServiceRoleForECS IAM role already exists.	true	For a first-time ECS deployment in an account, this must be true.

3. Acknowledge the creation of IAM roles and permissions when prompted.
4. Choose **Create stack** to begin deployment.
5. Wait until the stack status shows **CREATE_COMPLETE**.
6. Retrieve the **Application URL** or **ALB DNS Name** from the *Outputs* tab.
7. Access the application via the published domain to verify successful deployment.

Step 4: Access the Application

Navigate to the **Outputs** tab of the CloudFormation stack and find the `AlbDnsName`. You will need to configure a CNAME record in your DNS provider to point your domain to this ALB DNS name.

7.2. Testing and Troubleshooting

7.2.1. End-to-End Deployment Validation

This guide provides a streamlined process to verify that your AI Podcast Generator is fully operational.
Prerequisites: An active user in the Cognito User Pool and a sample PDF document (1-3 pages).

Step 1: Authenticate and Upload

1. Navigate to your application's URL. You will be redirected to a login page.
2. Log in with your user credentials.
3. On the main dashboard, upload your sample PDF document.
   (Download a sample PDF from here; however, this solution accepts all kinds of PDF layouts, Images, and tables within a 5MB file size.)
4. Click "Next: Configure & Create" once the file appears in the list.

Step 2: Configure and Initiate

1. In the configuration modal, set the following options:
   - Script Type: `Dialogue (1 Guest)`
   - Content Detail: `Short (Concise Summary)`
   - Voices: Assign a different voice to the Host and the Guest.
   - Require Script Review: Ensure this is checked `on`.
2. Click "Start Generation".

Step 3: Monitor and Review

1. Observe the real-time status updates on the dashboard card as it processes the document.

2. When the status changes to "Action Required: Review Script", click the "Review Script" button.
3. Review the generated script and click "Approve and Generate Audio".

Step 4: Verify and Download

1. The card will update as it generates and merges the audio, finally showing "Completed Successfully".
2. Click "Preview Audio" to listen to the generated podcast directly in your browser.
3. Use the menu button (three dots) to "Download Podcast" and "Download Script".

Validation Complete: If all steps were completed without error and the artifacts are accessible, your deployment is fully validated.

7.2.2. Troubleshooting Guidelines

If you encounter an issue, use the following guide to diagnose the root cause. Your primary resource for detailed error messages is the **Amazon CloudWatch Log Group** for the `api` service.

Symptom	Potential Cause(s)	Recommended Solution(s)
Cannot access the application URL (e.g., 503 error, timeout).	ECS services are not healthy or have failed to start.	1. Navigate to **EC2 -> Target Groups** and check the health of your service targets. 2. If unhealthy, go to the **ECS Cluster** and review logs for any stopped tasks to identify the startup error.
File upload fails with an error message.	1. The file is not a valid PDF. 2. The application's IAM role lacks permission to write to the S3 bucket.	1. Ensure the file is a valid `.pdf`. 2. In the **IAM Console**, verify that the EcsTaskRole has s3:PutObject permissions on the provisioned S3 bucket.
Workflow fails on "Agent-Vision" (OCR) or "Agent-Narrator" (Scripting) steps.	An API key for a third-party or AWS AI service is incorrect or invalid.	1. Navigate to **AWS Secrets Manager** and verify that all API keys (MISTRAL_API_KEY, etc.) are correct. 2. If using AWS Bedrock, ensure the EcsTaskRole has bedrock:InvokeModel permissions. 3. Check **CloudWatch** logs for "401 Unauthorized" or "Authentication Error" messages.

Workflow fails on "Agent-Voice" (Audio Generation) step.	The API key for the Text-to-Speech (TTS) service is incorrect.	1. In **AWS Secrets Manager**, verify the API keys for your configured TTS provider (SAIA or AWS Polly). 2. If using AWS Polly, ensure the EcsTaskRole has polly:SynthesizeSpeech permissions.
Workflow completes, but the final MP3 audio is silent.	An error occurred during the audio merging step.	This is an uncommon condition. Review the full logs for the specific run in the API service's **CloudWatch Log Group** and search for errors related to "pydub" or "ffmpeg".
Login page shows an error or redirects incorrectly.	The Callback URLs in the Cognito App Client are misconfigured.	1. Navigate to the **Cognito Console** and select your User Pool. 2. Under "App integration", ensure the "Allowed callback URLs" match your application's domain exactly.
Cloudformation Stack fails with `ROLLBACK_COMPLETE`	Invalid Parameter Values Resource already exists ECS Service Stabilization Failure Missing IAM Permissions	In the CloudFormation console, select your stack and go to the "Events" tab. This will show a log of the resources being created. The resource with a `CREATE_FAILED` or `UPDATE_FAILED` status is the cause of the failure. The "Status reason" column will provide a detailed error message.

8. Health Check

The DocumentCast deployment includes several built-in health checks and monitoring features through AWS CloudWatch, ECS, and the Application Load Balancer (ALB).

This section explains what to watch, where to look, and how to interpret the results.

8.1. Application Health Monitoring

Component	Where to Monitor	Key Metrics / Checks	Expected Normal Values / Notes
Application Load Balancer (ALB)	AWS Console → *EC2 → Target Groups → Health checks*	• **HealthyHostCount** • **UnHealthyHostCount** • **HTTPCode_Target_5XX_Count**	• Healthy = all targets show “healthy.” • Unhealthy = 0. • 5XX errors should remain near 0; consistent spikes indicate backend/API issues.
ECS Services (API & Frontend)	AWS Console → *ECS → Clusters → Services → Metrics tab*	• **CPUUtilization** • **MemoryUtilization**	• Average CPU < 70% under normal traffic (target autoscaling threshold). • Memory steady under 75%. Sustained > 80% may trigger scaling.
Amazon CloudWatch Dashboard	AWS Console → *CloudWatch → Dashboards → {ProjectName}-Application-Dashboard*	• ALB RequestCount • 4XX/5XX Error Rate • ALB Target Health • ECS CPU & Memory graphs • Cache Performance • Cognito: User Sign-In Activity • Bedrock: Token Usage	• RequestCount increases with traffic. • 4XX ≤ 5% of requests. • 5XX near 0. • API and Frontend CPU curves are stable.
ElastiCache Serverless (Valkey)	*CloudWatch → Metrics → AWS/ElastiCache*	• CPUUtilization • MemoryUsage	• CPU < 60% typical.

			• MemoryUsage < 70%. Sudden jumps may indicate traffic spikes.
Amazon Cognito	*CloudWatch → Metrics → AWS/Cognito*	• SignInSuccesses • UserAuthenticationFailures	• Sign-in failures < 5% of total sign-ins. • Any sudden rise may mean misconfigured redirect URIs or user lockouts.
Application Logs	*CloudWatch → Log Groups → /ecs/{ProjectName}-api* and */ecs/{ProjectName}-frontend*	• Log entries for errors/exceptions.	• Only INFO or DEBUG logs expected during normal use. Frequent ERROR logs indicate service failure or missing secrets.

8.2. Tracing and Distributed Monitoring

- **AWS X-Ray** is integrated with the API container through the ADOT collector.
  Access it via *CloudWatch → Service Map → X-Ray traces*.
  - Normal traces show sub-second latency for internal ECS calls.
  - Requests consistently over **1.5 seconds** indicate API latency or external model timeout.

8.3. Automated Health Checks

- The ALB automatically performs `/api/health` probes on each ECS task every **30 seconds**.
  - "Healthy" means the endpoint returns HTTP 200.
  - Two consecutive failed checks mark a target as unhealthy and trigger replacement.
- ECS automatically replaces failed tasks to maintain the desired count defined in the stack parameters (`ApiMinCapacity`, `FrontendMinCapacity`).

8.4. Alerting and Anomalies

You can configure **CloudWatch Alarms** to alert the operations team if:

- 5XX errors > 10 in 5 minutes
- ECS CPUUtilization > 80% for 10 minutes
- ElastiCache MemoryUsage > 80%
- Cognito Sign-in failures > 10% of total attempts

Alerts can send notifications to an **SNS topic** or email distribution list.

8.5. Summary of Normal KPI Ranges

Metric	Normal Range / Target
ALB HealthyHostCount	100% healthy
API CPUUtilization	40–70%
Frontend CPUUtilization	30–60%
MemoryUtilization	≤ 75%
4XX Error Rate	≤ 5%
5XX Error Rate	≈ 0%
Cache CPUUtilization	≤ 60%
Cognito Auth Failures	≤ 5%

9. Backup and Recovery

9.1. Data Store

Purpose: Stores uploaded PDF documents, generated MP3 podcast files, and derived content.

Configuration:

- Versioning enabled (`VersioningConfiguration: Status: Enabled`)
- Server-Side Encryption with KMS (`SSE-KMS` using a dedicated `KMSKey`)
- Lifecycle rules for automatic expiration (`ExpirationInDays` parameter)
- CORS and Block Public Access configured

Backup Strategy:

- S3 Versioning automatically maintains historical copies of deleted or overwritten objects.
- Optionally, customers can enable Cross-Region Replication (CRR) for disaster recovery.
- Backup verification can be done via the AWS Management Console → *S3* → *Versions tab*.

Recovery Steps:

1. Go to Amazon S3 Console → your application bucket.
2. Enable “Show versions.”
3. Select the previous object version and choose Restore or Download.
4. For mass restore, use the AWS CLI:

   `aws s3api list-object-versions --bucket <bucket-name>`

   then `aws s3api copy-object` for the desired version.

9.1.1. Amazon ElastiCache Serverless (Valkey) – Ephemeral Cache

Purpose: Stores transient runtime data (session tokens, state tracking, temporary AI responses).

Configuration:

- Managed by AWS; encryption in transit and at rest enabled.
- Snapshots are automatically managed by AWS for point-in-time recovery.

Backup Strategy:

- ElastiCache Serverless automatically creates daily encrypted snapshots.
- Customers can export snapshots to S3 if needed for offline retention.

Recovery Steps:

1. Navigate to ElastiCache Console → Snapshots.
2. Select the latest snapshot and choose Restore Cache Cluster.
3. Update the ECS service configuration to point to the new cache endpoint.

9.1.2. AWS Secrets Manager – Configuration Data Store

Purpose: Stores API keys, JWT secrets, and integration parameters (`DocumentCastAppSecrets`).

Configuration:

- Encrypted using the same KMS CMK (`KMSKey`).

Backup Strategy:

- Secrets Manager provides versioning for each secret value update.
- Backup by exporting the JSON value to a secure S3 bucket using:

```
aws secretsmanager get-secret-value --secret-id DocumentCastAppSecrets > backup.json
```

Recovery Steps:

1. In AWS Secrets Manager Console, open the secret.
2. Choose Versions → select a previous version → click Restore.
3. For manual restore, re-import from the saved JSON file via CLI:

```
aws secretsmanager put-secret-value --secret-id DocumentCastAppSecrets --secret-string file://backup.json
```

4. Redeploy ECS service to load the restored secrets.

9.1.3. Amazon Cognito – User Data Store

Purpose: Stores user authentication records, tokens, and identities.

Configuration:

- Managed service; encrypted by default.

Backup Strategy:

- Amzon Cognito automatically stores user data in a managed directory.
- For long-term retention, use the Cognito User Export feature or AWS CLI:

```
aws cognito-idp list-users --user-pool-id <UserPoolId> > users_backup.json
```

Recovery Steps:

1. In case of user data corruption or migration, recreate a user pool.
2. Import users from the backup JSON file using Cognito APIs or custom Lambda scripts.

9.2. Recovery Objectives (RPO and RTO)

DocumentCast relies on AWS-managed backup capabilities for recovery, and customers can align these to their own **Recovery Point Objective (RPO)** and **Recovery Time Objective (RTO)** requirements by adjusting the AWS services that store their data.

Because DocumentCast runs entirely on serverless and managed services, **RPO and RTO are determined by the backup frequency and retention policies applied to each AWS data store**, as shown below:

9.2.1 Configuring RPO

Customers can control how much data they may lose in the event of a disaster by adjusting the backup or versioning settings of AWS services:

- **Amazon S3 (Uploaded PDFs & MP3s)**
  RPO is determined by S3 Object Versioning. Customers may reduce RPO by enabling:
  - Cross-Region Replication (CRR)
  - Additional lifecycle rules to retain more object versions
- **Secrets Manager**
  RPO depends on how frequently customers export or rotate secret versions.
- **ElastiCache Serverless**
  RPO is based on AWS-managed snapshots. Customers may optionally export snapshots more frequently to reduce RPO.
- **Cognito Users**
  RPO corresponds to how frequently user exports are performed.

9.2.2. Configuring RTO

Customers can influence how quickly the system can be recovered by:

- Keeping the **original KMS key** active to avoid decryption failures
- Ensuring CloudFormation parameters are well-documented so the stack can be redeployed quickly
- Using S3 Versioning and Secrets Manager Versioning to restore data immediately without manual recreation
- Maintaining optional ElastiCache snapshot exports for faster cache rehydration

Because DocumentCast is fully serverless, compute resources (ECS Fargate tasks) recover within minutes once the underlying data stores are restored. **The primary factor affecting RTO is access to encrypted data via the original KMS key.**

10. Routine Maintenance

10.1. Rotating Credentials:

Phase 1: Update the Stack Parameter

This step updates the key's value in AWS Secrets Manager.

1. Navigate to CloudFormation and select your application's stack.
2. Click Update and choose Use current template.
3. In the Parameters section, replace the old MistralApiKey with your new key.
4. Proceed through the wizard and click Update stack.

The secret is now updated, but the application has not yet loaded it.

Phase 2: Redeploy the ECS Service

This step forces the application to load the new key.

1. Navigate to Amazon ECS and select your cluster.
2. Go to the Services tab, select the api service, and click Update.
3. Check the Force new deployment box.
4. Click Update.

Result: ECS will replace the old application tasks with new ones that automatically pull and use the updated secret upon startup.

10.2. Software Patches:

10.2.1. This zero-downtime process updates your running API and Frontend services with new container images.

1. Navigate to CloudFormation and select your application's stack.
2. Click Update and choose Use current template.
3. In the Parameters section, locate ApiImage and FrontendImage.
4. Paste the new image URIs into the corresponding parameter fields.
5. Proceed through the wizard and click Update stack.

Result: CloudFormation will perform a rolling update on your ECS services. It will create new task definitions with the new image URIs and deploy new tasks. Once the new tasks are healthy, it will gracefully stop the old ones, ensuring a zero-downtime deployment.

10.2.2 License Management:

DocumentCast uses AWS-managed services and open standards that do not require separate AWS License Manager configuration. Optional integrations with third-party providers such as Mistral AI or Globant Enterprise AI (GEAI) use API key–based subscriptions managed directly by the customer through the provider's platform. AWS License Manager is not required for this solution, as no commercial software licenses are deployed or tracked through AWS.

10.3. AWS Service Limits:

DocumentCast relies on AWS-managed services such as Amazon Bedrock, Amazon Polly, ECS Fargate, ALB, and ElastiCache. If usage approaches or exceeds service limits, AWS automatically exposes signals through CloudWatch metrics, logs, ECS task status, and CloudFormation events. Customers can use these indicators to understand when quota thresholds are being reached and take corrective action through the AWS Service Quotas console.

10.3.1. Amazon Bedrock

If your application's usage exceeds the default Tokens-per-Minute (TPM) or Requests-per-Minute (RPM) limits for the selected model, you must purchase dedicated capacity.
Purchase Provisioned Throughput.

1. Navigate to the Amazon Bedrock console.
2. Click Provisioned Throughput in the bottom-left menu.
3. Click Purchase Provisioned Throughput and select the model used by the application.
4. Define the number of Model Units required to meet your capacity needs.
5. After purchase, update your application's code to use the ARN of your new Provisioned Throughput when calling the Bedrock service.

10.3.2. Amazon Polly

The application may experience ThrottlingException errors from Amazon Polly if the default request rate (Transactions Per Second) is exceeded.
Request a service quota increase.

1. Navigate to the Service Quotas console.
2. Select Amazon Polly.
3. Find and request an increase for the SynthesizeSpeech operations per second (TPS) quota.

10.3.3. Detecting Service Limit Exceedance

AWS surfaces quota-related conditions automatically through:

- **ECS task failures or scaling errors** (e.g., insufficient CPU/Memory).
- **CloudFormation events** indicating quota or capacity issues.
- **CloudWatch metrics** showing rising 4XX/5XX errors, throttling patterns, or sustained high CPU/memory.
- **CloudWatch Logs** containing messages such as *Rate exceeded* or *ThrottlingException*.
- The **AWS Service Quotas console**, where customers can view usage relative to limits and request increases.

These AWS-native indicators allow customers to monitor health and capacity and take proactive action without requiring additional notifications inside the DocumentCast interface.

10.4. License Management

Globant generates revenue from DocumentCast through two channels:

10.4.1. AWS Marketplace Subscription

DocumentCast is offered on AWS Marketplace as a **flat-rate subscription priced at $10 per month** per deployment.
This subscription fee is paid to Globant through AWS Marketplace and represents the primary revenue source for the solution itself.
The subscription is **not usage-based and not per user**, a single $10 monthly subscription enables unlimited internal use of the application.

10.4.2. Optional Globant Enterprise AI (GEAI) Integration

If customers optionally choose to use **Globant Enterprise AI (GEAI)** services for LLM inference or text-to-speech, these calls are billed separately through the GEAI platform.
This usage generates an **additional revenue stream for Globant**, independent of the AWS Marketplace subscription.

Customers only incur GEAI charges if they decide to activate GEAI features and supply the necessary API keys (**GEAI_API_KEY**, **SAIA_API_KEY**, etc.).

11. Emergency Maintenance

11.1. Fault Conditions

The troubleshooting guide helps customers fix common problems that might happen after deployment. It explains what to check, where to look in the AWS console, and what actions to take.
Examples include:

- **ECS service fails to start** – check CloudFormation Events or ECS task logs in CloudWatch to find the error and redeploy the service.
- **ALB shows unhealthy targets** – confirm the application container is running and that the health-check path `/api/health` is responding.
- **S3 access errors** – verify the bucket exists and that IAM permissions include `s3:GetObject` and `kms:Decrypt`.
- **Login issues** – check that Cognito domain names and redirect URLs are correct.
  Each issue lists a short cause and simple steps to resolve it using the AWS Console.

11.2. Software Recovery

The recovery guide explains how to bring the system back to a working state if something serious happens such as an AWS account problem, region outage, or accidental deletion.

It includes step-by-step instructions for restoring all critical parts of the DocumentCast solution:

1. **Amazon S3** – recover uploaded PDFs and audio files by restoring earlier object versions (S3 versioning is enabled).
2. **Secrets Manager** – restore old API keys or configuration values by selecting a previous version or re-importing the secret JSON.
3. **ElastiCache Serverless (Valkey)** – restore from the most recent AWS-managed snapshot.
4. **Cognito** – if needed, recreate the user pool and import users from backup.
5. **KMS key** – reuse the same KMS key that was created during the original deployment to regain access to encrypted data.
6. **CloudFormation stack** – redeploy the stack using the same parameters and existing KMS key ARN, then verify the application URL.

Note: To successfully restore the DocumentCast environment, you must retain and reuse the original KMS key created by the stack. Without this key, encrypted data such as S3 files, Secrets Manager values, and cache data cannot be recovered.

12. Support and SLAs

Globant provides application-level support for the DocumentCast solution with a standard 24-hour response time. Customers can reach the support team at documentcast@globant.com or through the Globant Enterprise AI Help Center:
🔗 https://www.globant.com/globant-enterprise-ai/help-center

12.1 Support Ownership

- **Globant** provides support for the DocumentCast application itself, including issues related to deployment, configuration, API errors, GEAI/Mistral integrations, and runtime behavior.
- **Customers** are responsible for managing their AWS account and AWS infrastructure components (ECS, S3, IAM, ALB, ElastiCache, CloudFormation, etc.).

12.2 AWS Support Requirements

A **Basic AWS Support plan** is sufficient to deploy and operate DocumentCast.
Globant recommends **Business or Enterprise Support** for production workloads to receive 24×7 AWS assistance for:

- CloudFormation provisioning
- ECS task failures
- VPC, DNS, and load balancer issues
- IAM permission troubleshooting
- Service Quota increases

AWS Support covers the underlying platform; Globant Support covers the DocumentCast application layer.

12.3 Globant Support SLAs

Support Category	Description	SLA / Response Target	Availability
General Support Requests	Deployment help, configuration questions, and non-urgent issues.	Response within **24 hours**.	Monday–Friday (Business Hours)

Production-Impacting Issues	Errors affecting core DocumentCast features (e.g., podcast generation failures, API timeouts).	Response within **24 hours**.	24×5 (Weekdays)
Critical Service Outage	DocumentCast is completely unavailable in the customer's AWS account.	Response within **24 hours**.	24×5 (Weekdays)
GEAI / SAIA Integration Issues	Problems with Globant Enterprise AI service keys, SAIA TTS, or GEAI LLM usage.	Response within **24 hours**.	Monday–Friday
Feature Requests	Enhancements or non-urgent product suggestions.	Reviewed within **5 business days**.	Monday–Friday
Support Channels	documentcast@globant.com, Globant Enterprise AI Help Center.	Submit tickets anytime.	24×7 ticket submission

13. Appendices

13.1. Cloud Formation Execution Policy

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudFormationPermissions",
            "Effect": "Allow",
            "Action": "cloudformation:*",
            "Resource": "*"
        },
        {
            "Sid": "IAMPermissions",
            "Effect": "Allow",
            "Action": [
                "iam:CreateRole",
                "iam:DeleteRole",
                "iam:GetRole",
                "iam:TagRole",
                "iam:UntagRole",
                "iam:PutRolePolicy",
                "iam:DeleteRolePolicy",
                "iam:CreateServiceLinkedRole",
                "iam:DeleteServiceLinkedRole"
            ],
            "Resource": [
                "arn:aws:iam::*:role/*-EcsTaskExecutionRole-*",
                "arn:aws:iam::*:role/*-EcsTaskRole-*",
                "arn:aws:iam::*:role/*-ApiAutoScalingRole-*",
                "arn:aws:iam::*:role/*-FrontendAutoScalingRole-*",

"arn:aws:iam::*:role/aws-service-role/ecs.amazonaws.com/AWSServiceRoleForECS"
            ]
        },
        {
            "Sid": "PassECSRoles",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/*-EcsTaskExecutionRole-*",
                "arn:aws:iam::*:role/*-EcsTaskRole-*"
```

```
            ],
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ecs-tasks.amazonaws.com"
                }
            }
        },
        {
            "Sid": "PassAutoScalingRoles",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/*-ApiAutoScalingRole-*",
                "arn:aws:iam::*:role/*-FrontendAutoScalingRole-*"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "application-autoscaling.amazonaws.com"
                }
            }
        },
        {
            "Sid": "NetworkingPermissions",
            "Effect": "Allow",
            "Action": "ec2:*",
            "Resource": "*"
        },
        {
            "Sid": "ECSandServiceDiscoveryPermissions",
            "Effect": "Allow",
            "Action": [
                "ecs:*",
                "servicediscovery:*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "LoadBalancingAndScalingPermissions",
            "Effect": "Allow",
            "Action": [
                "elasticloadbalancing:*",
                "application-autoscaling:*"
            ],
```

```
            "Resource": "*"
        },
        {
            "Sid": "StorageAuthAndDnsPermissions",
            "Effect": "Allow",
            "Action": [
                "s3:*",
                "cognito-idp:*",
                "route53:*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SecretsCachingAndCryptoPermissions",
            "Effect": "Allow",
            "Action": [
                "kms:*",
                "elasticache:*",
                "secretsmanager:*",
                "ssm:*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "LoggingAndCertificatePermissions",
            "Effect": "Allow",
            "Action": [
                "logs:*",
                "cloudwatch:*",
                "acm:DescribeCertificate"
            ],
            "Resource": "*"
        }
    ]
}
```

13.2. ECS Tasks Execution

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EcsTaskExecutionPolicy",
            "Effect": "Allow",
            "Action": [
                "ecr:GetAuthorizationToken",
                "ecr:BatchCheckLayerAvailability",
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchGetImage",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ParameterStoreAccess",
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameter",
                "ssm:GetParameters"
            ],
            "Resource": "arn:aws:ssm:*:*:parameter/*/adot-config"
        },
        {
            "Sid": "SecretsManagerAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource":
"arn:aws:secretsmanager:<Region>:<Account_ID>:secret:<Secret_Name>"
        }
    ]
}
```

13.3. ECS Tasks Role

JavaScript

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "BedrockAccess",
            "Effect": "Allow",
            "Action": [
                "bedrock:InvokeModel"
            ],
            "Resource":
"arn:aws:bedrock:<Region>::foundation-model/<Bedrock_Model>"
        },
        {
            "Sid": "ElastiCacheAccess",
            "Effect": "Allow",
            "Action": [
                "elasticache:DescribeCacheClusters"
            ],
            "Resource":
"arn:aws:elasticache:<Region>:<Account_ID>:serverlesscache/<ElastiCache_Cluster_Nam
e>"
        },
        {
            "Sid": "KMSAccess",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:DescribeKey",
                "kms:GenerateDataKey"
            ],
            "Resource": "arn:aws:kms:<Region>:<Account_ID>:key/<KMS_Key_ID>"
        },
        {
            "Sid": "XRayAccess",
            "Effect": "Allow",
            "Action": [
                "xray:PutTraceSegments",
                "xray:PutTelemetryRecords"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CloudWatchLogsAccess",
```

```
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": [

"arn:aws:logs:<Region>:<Account_ID>:log-group:/ecs/<api_LogGroup_name>:*",

"arn:aws:logs:<Region>:<Account_ID>:log-group:/ecs/<frontend_LogGroup_name>:*"
            ]
        },
        {
            "Sid": "CloudWatchMetricsAccess",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "PollyAccess",
            "Effect": "Allow",
            "Action": [
                "polly:SynthesizeSpeech",
                "polly:DescribeVoices"
            ],
            "Resource": "*"
        },
        {
            "Sid": "S3ObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject",
                "s3:DeleteObject"
            ],
            "Resource": "arn:aws:s3:::<S3_Bucket_Name>/*"
        },
        {
            "Sid": "S3BucketAccess",
            "Effect": "Allow",
            "Action": [
```

```
                "s3:ListBucket"
            ],
            "Resource": "arn:aws:s3:::<S3_Bucket_Name>"
        },
        {
            "Sid": "SecretsManagerAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource":
"arn:aws:secretsmanager:<Region>:<Account_ID>:secret:<Secret_Name>"
        }
    ]
}
```

13.4. KMS

JavaScript

```
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "Enable IAM User Permissions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::<Account ID>:root"
      },
      "Action": "kms:*",
      "Resource": "*"
    },
    {
      "Sid": "Allow CloudWatch Logs",
      "Effect": "Allow",
      "Principal": {
        "Service": "logs.us-east-1.amazonaws.com"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
```

```
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
      ],
      "Resource": "*"
    }
  ]
}
```