**Ms Davies Rachel**

**CLDV6212**

**POE PART 2**

**ST10259834**

1. Integrating Functions to build a robust application architecture

Web App Link: https://st10259834part2.azurewebsites.net

Function App Link: https://st10259834func.azurewebsites.net

GitHub Repo Link: <https://github.com/IIEWFL/cldv6212-part-2-ST10259834-Aaryan-Makan>

**Postman: Blob Success:**

A black rectangular object with white text

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**File share upload:**

A black rectangle with white text

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Enqueue order command + update order:**

A black rectangular object with white text

AI-generated content may be incorrect.

A screenshot of a computer

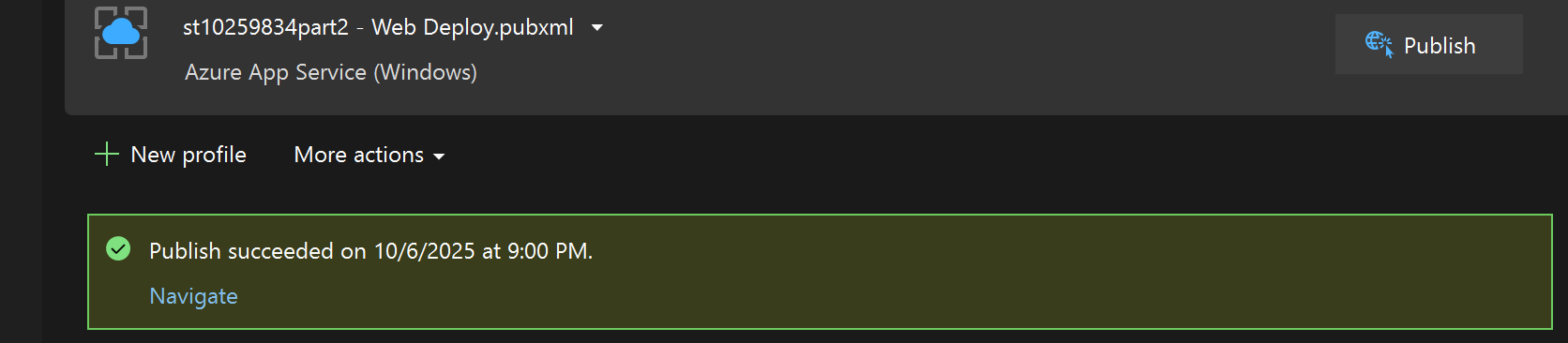
AI-generated content may be incorrect.

**Publish Function App:**

A screenshot of a computer

AI-generated content may be incorrect.

**Publish web app:**



**Search feature:**

A screenshot of a computer

AI-generated content may be incorrect.

1. Using services for improving the customer service
2. Azure event hubs

**Description:**

Azure Event Hubs is a real-time data streaming service designed to capture and process millions of events per second. It’s often described as a “front door” for event data, because it can ingest continuous streams of information from applications, sensors, or user actions, and then make those events immediately available for analytics or processing. In a retail scenario, every customer action—like placing an order, checking inventory, or updating account details—can be turned into an event that Event Hubs processes instantly.

**Mechanism:**  
When a customer performs an action in the ABC Retail app, such as creating an order, a small message describing the event (for example, *“OrderCreated”* or *“OrderUpdated”*) is sent to Azure Event Hubs. Event Hubs then streams this event to consumers, such as Azure Stream Analytics or a reporting dashboard. These connected services can process the data in real time, identify trends, or trigger alerts. For instance, if there’s a sudden increase in orders for a product, Event Hubs ensures that this information is instantly visible to administrators or linked systems without slowing down the app.

**How it adds value to users:**  
The biggest advantage for customers is **real-time responsiveness**. They receive immediate feedback when they place an order—such as instant confirmation messages or updated stock levels—because the system processes data as it arrives. Event Hubs also improves reliability by managing spikes in traffic smoothly, ensuring that the experience remains consistent even during peak shopping times. Beyond this, businesses can use the live event data to anticipate customer needs, personalize recommendations, and quickly detect problems. In other words, Event Hubs helps make the app feel faster, smarter, and more attentive to each customer’s actions.

1. Azure service bus

**Description:**  
Azure Service Bus is a message-brokering service that allows different parts of an application to communicate reliably, even if they run at different times or on separate systems. It ensures that messages are delivered in order, only once, and without being lost—making it ideal for handling business-critical operations such as order processing or inventory updates.

**Mechanism:**  
In the ABC Retail app, when a user submits an order, the system doesn’t process it immediately within the same request. Instead, it sends a message to a Service Bus queue. This message contains the details of the order—such as the product, quantity, and customer ID—and waits in the queue until a background service or Azure Function retrieves it. That function then processes the order: updating the database, adjusting inventory, or sending a confirmation email. This asynchronous model means the user doesn’t have to wait for all these steps to complete before seeing a response.

**How it adds value to users:**  
For the end user, this approach translates into a **smoother and faster experience**. The app responds immediately, even while complex operations continue in the background. Because Service Bus guarantees message delivery, users can trust that their orders are never lost, duplicated, or delayed by system errors. It also allows the application to scale effortlessly—if many users submit orders at once, messages are safely queued and processed one by one without overloading the system. Overall, Service Bus enhances both the reliability and efficiency of the app, which in turn builds user confidence and satisfaction.

Together, Azure Event Hubs and Service Bus bring intelligence and resilience to the ABC Retail system. Event Hubs ensures that customer actions generate instant, real-time insights, while Service Bus guarantees that important transactions are processed reliably and efficiently behind the scenes. By combining real-time data streaming with dependable message delivery, the application can offer a more responsive, consistent, and trustworthy experience for customers. Qualities that define a modern, cloud-based retail platform.

**Reference**

Microsoft Docs (2024) *What is Azure Event Hubs?* Available at: https://learn.microsoft.com/en-us/azure/event-hubs/event-hubs-about (Accessed: 6 October 2025).

Microsoft Docs (2024) *Event Hubs – Features and Architecture*. Available at: https://learn.microsoft.com/en-us/azure/event-hubs/event-hubs-features (Accessed: 6 October 2025).

Microsoft Docs (2024) *What is Azure Service Bus?* Available at: https://learn.microsoft.com/en-us/azure/service-bus-messaging/service-bus-messaging-overview (Accessed: 6 October 2025).

Microsoft Docs (2024) *Service Bus Queues – Concepts*. Available at: https://learn.microsoft.com/en-us/azure/service-bus-messaging/service-bus-queues-topics-subscriptions (Accessed: 6 October 2025).

Azure Architecture Center (2024) *Event-Driven Architecture Patterns*. Available at: https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/event-driven (Accessed: 6 October 2025).