Homework 02

The homework requirements are explained in Requirements file

- Homework 02
  - Technologies (Azure)
    * Cosmos DB
    * Azure API App
    * Event Hub
    * Azure Functions
    * Web PubSub
    * Client/Web Apps
  - Description (Requirements facing)
    * Components enumeration
    * Stateful vs Stateless
    * Functions as a Service (FaaS)
    * Websockets
  - System description
    * Data Sources
    * Event Processor
    * Websockets / Broadcasting to users in real time
    * Get/Post data
    * CI/CD
    * Stack
  - Architecture Diagram
  - System Analysis

Technologies (Azure)

A range of products called Azure Stack extends Azure services and capabilities to the environment of your choice, including datacenters, edge locations, and remote offices. Create and roll out edge and hybrid computing apps, and run them uniformly across geographic borders.

Cosmos DB

Cosmos DB is a Microsoft Azure multi-model, globally distributed NoSQL database service. It supports a range of data structures, including document, key-value, graph, and column-family data, and is built to manage large-scale, globally dispersed applications.

For mission-critical applications, Cosmos DB offers a highly scalable and highly available database service with low latency and quick throughput. Developers may select the ideal balance of consistency and availability for their applications thanks to its support for ACID transactions and availability of numerous consistency levels.

Many applications, including IoT, gaming, e-commerce, and web and mobile

applications, can be employed with Cosmos DB. Many APIs, including SQL, MongoDB, Cassandra, Azure Tables, and Gremlin are supported (for graph data). This makes it simple for developers to use the API that best suits their application's requirements or with which they are most comfortable.

Azure API App

A cloud-based service called Azure API App is offered by Microsoft Azure and allows developers to easily create, deploy, and manage RESTful APIs. It offers a fully managed hosting environment for developing and running APIs and is a component of the Azure App Service platform.

Developers can create APIs using a range of programming languages and frameworks, including as.NET, Java, Node.js, and Python, with Azure API App. Additionally, they may benefit from attributes like automatic scaling, integrated authentication and authorisation, and compatibility with Azure services like Azure Active Directory and Azure Functions.

Moreover, Azure API App offers API monitoring and debugging tools, as well as analytics and reporting features to assist developers in comprehending API usage and performance. Additionally, it allows integration with well-known development tools like Microsoft Studio, GitHub, and Jenkins, as well as continuous deployment.

In conclusion, Azure API App offers a complete platform for creating, deploying, and maintaining APIs in a secure and scalable way.

Event Hub

A cloud-based service offered by Microsoft Azure, Azure Event Hub, enables the real-time streaming of massive amounts of data from diverse sources into the cloud. It is perfect for situations like IoT telemetry, application logging, and analytics since it is built to handle large volume, low latency, and highly scalable event streams.

Developers may gather, analyse, and store huge volumes of data from sources like applications, devices, and sensors using Azure Event Hub. Depending on the settings, the service can process millions of events per second and can keep track of occurrences for days, weeks, or months.

Partitioning, event ordering, and durable storage are just a few of the features that Azure Event Hub offers to developers to assist them in managing their event streams. In order to make it simple to handle and analyze event data in real-time, it also enables integration with other Azure services including Azure Stream Analytics, Azure Functions, and Azure Data Lake Storage.

Overall, Azure Event Hub offers a highly scalable and trustworthy platform for obtaining, handling, and performing real-time analysis on massive amounts of event data from diverse sources.

Azure Functions

Developers can create and execute event-driven, serverless applications in the cloud using Azure Functions, a serverless computing service provided by Microsoft Azure. Without having to worry about managing the supporting infrastructure, it enables developers to concentrate on building code for their application logic.

Developers can use a number of programming languages, such as C#, Java, JavaScript, Python, and PowerShell, with Azure Functions. A range of triggers are also available to them, including timers, HTTP requests, and messaging services like Azure Service Bus and Event Hubs. They can then design functions that run in response to particular events or requests.

Automatic scaling, pay-per-use pricing, and connection with other Azure services like Azure Storage, Azure Cosmos DB, and Azure Event Grid are just a few of the capabilities that Azure Functions offers to assist developers in managing their serverless applications.

Overall, Azure Functions offers a scalable and adaptable framework for developing and deploying serverless, event-driven applications. It frees developers from having to worry about overseeing the underlying infrastructure so they can concentrate on building code for their application logic.

Web PubSub

A cloud-based service called Azure Web PubSub allows for real-time websocket communication between client and server applications. It is perfect for use cases like real-time chat, gaming, and live broadcasting since it is built to handle high-frequency, low-latency messaging scenarios.

Developers can establish and manage hubs, which are conduits for messaging between client and server applications, with Azure Web PubSub. Websockets can be used by clients to connect to a hub and subscribe to one or more of the hub's channels to receive messages.

To assist developers in managing their messaging scenarios, Azure Web PubSub offers a variety of features, such as the ability to broadcast messages to all connected clients, authorise and authenticate connections, and scale the number of service instances up or down in response to demand.

As messages are received by the service, Azure Web PubSub can connect to other Azure services, such as Azure Functions and Azure Logic Apps, to launch serverless processes or workflows.

All things considered, Azure Web PubSub offers a highly scalable and dependable platform for websocket-based real-time communication between client and server applications. Developers may create real-time communications scenarios with it rapidly and without having to worry about the supporting infrastructure.

Client/Web Apps

With the help of Azure Web Apps, developers can quickly and easily create, deploy, and scale web apps in the cloud. Web application developers may concentrate on creating their apps rather than managing the underlying infrastructure because to the completely managed hosting environment it offers.

Developers can create web applications using Azure Web Apps using a range of programming languages and frameworks, such as.NET, Java, Node.js, PHP, and Python. Additionally, they may benefit from functions like built-in load balancing, intelligent scaling, and connection with Azure services like Azure SQL Database and Azure Storage.

In order to help developers understand the usage and performance of their apps, Azure Web Apps offers tools for monitoring and debugging web applications, as well as analytics and reporting capabilities. Additionally, it allows integration with well-known development tools like Microsoft Studio, GitHub, and Jenkins, as well as continuous deployment.

Generally speaking, Azure Web Apps offers a complete platform for creating, deploying, and managing online apps in a safe and scalable way. It is the best option for businesses of all sizes since it enables developers to concentrate on creating their apps rather than managing the supporting infrastructure.

Description (Requirements facing)

A stock ticker is a report of the price for certain securities, updated continuously throughout the trading session by the various stock exchanges.

Components enumeration

Azure Cosmos DB, Azure Event Hub, Azure API App, Azure Functions, Azure Web PubSub, and Azure Web Apps.

Stateful vs Stateless

Stateful services, such as Azure Cosmos DB and Azure Event Hub, keep track of the current state of the data they store or process across time.

Several data models can be used with Azure Cosmos DB, a NoSQL database service that also offers capabilities like worldwide distribution, automatic indexing, and various consistency levels. It enables for data retrieval using a variety of query methods and stores and organizes data in an organized fashion.

A streaming platform called Azure Event Hub is made to manage enormous amounts of data, like real-time event streams from IoT devices. It has capabilities including event ordering, segmentation, and durable storage, enabling it to preserve the state of the data it processes over time.

On the other side, stateless services, such as Azure API App, Azure Functions, Azure Web PubSub, and Azure Web Apps, do not preserve any state of the data they handle. They don't retain or manage data over time; instead, they run code in response to events or requests.

Functions as a Service (FaaS)

By default, functions as a service (FaaS) like Azure Functions are stateless, but depending on the use case, they can also be built to be stateful.

For instance, a FaaS application may be created to keep state in a database or other external storage service. As a result, the function can keep state across function calls by accessing and updating the state as required.

To benefit from serverless computing's scalability advantages, it is often advised to build FaaS applications as stateless. The provider can quickly create new instances of the function to manage an increase in load by keeping the functions stateless, freeing them from the burden of handling state across many instances.

Websockets

Real-time messaging situations are made possible by Microsoft Azure's cloud-based Azure Web PubSub service. Azure Web PubSub uses WebSockets as one of its main protocols to offer real-time bi-directional communication between clients and the server.

A permanent connection is created between the client and the server when a client connects to Azure Web PubSub using WebSockets. The server can push messages to the client in real-time since this connection is kept open during the client's session.

For WebSocket messages, Azure Web PubSub supports both JSON and MessagePack encoding. When the message is brief or compatibility with other WebSocket clients is required, JSON encoding—the default—is employed. When dealing with huge amounts of data or when bandwidth efficiency is a problem, MessagePack encoding, a more compact binary format, might be employed.

In addition, Azure Web PubSub offers capabilities like message filtering, message broadcasting, and authentication and authorisation that may be combined with WebSockets to enable a range of real-time messaging scenarios.

System description

The system gets data about stocks (ticks) while sending them to the users and also storing them.

Data Sources

The Azure Function generates dummy data that is sent to Event Hub. The client can also add data through the client app using the API.

Event Processor

The Azure function gets triggered by new events from the Event Hub. It processes them adding them to the database and publishing them to Web PubSub.

Websockets / Broadcasting to users in real time

The Web PubSub service broadcasts the data received from Azure Function to all its subscribers via websockets in real time.

Get/Post data

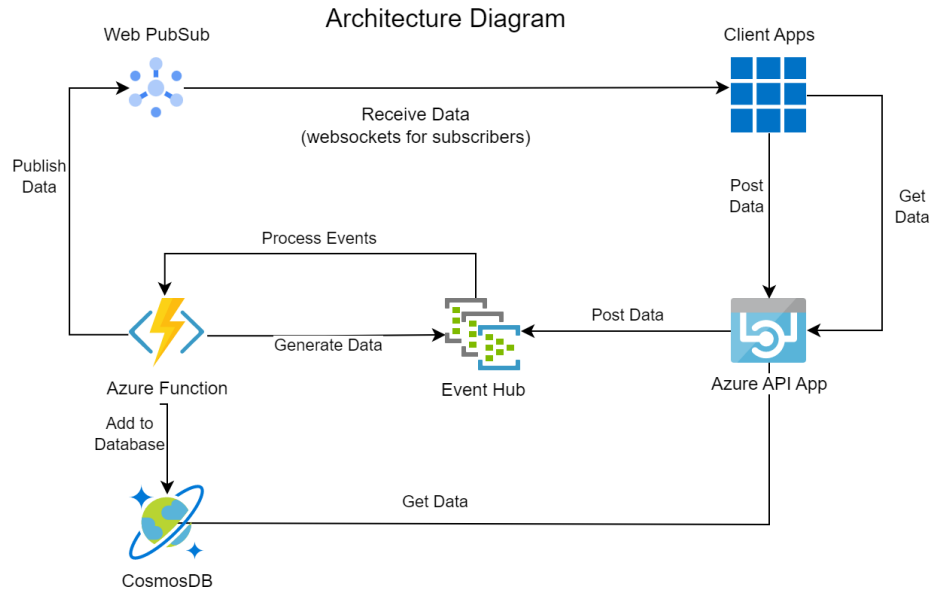Via the API you can either query all the data available or add new entries via client app.

CI/CD

All the code written here is deployed via Github Actions to the Azure Services.

Stack

Everything is written in .NET with several snippets of Javascript (requests, websockets, real-time table rows addition).

Architecture Diagram

Architecture Diagram

System Analysis

It is crucial to take into account a number of variables while examining the performance of a system architecture, including scalability, dependability, latency, throughput, and cost-effectiveness.

Many Azure components that are tuned for high performance, low latency, and dependability are included in the system architecture as mentioned. A distributed database with numerous consistency levels to fit application needs, Azure Cosmos DB provides high availability and low latency. Azure Event Hub is a streaming platform with low-latency event delivery, compatibility for numerous protocols, and the ability to manage massive volumes of data in real-time. With capabilities like caching and auto-scaling, Azure API App develops and exposes APIs that can manage large request rates. A serverless computation service that can handle large numbers of requests, Azure Functions reacts to events. A messaging service for real-time communication, Azure Web PubSub, can manage large numbers of connections and provides low-latency messaging. With Azure Web Apps, web applications may be hosted and run in the cloud while handling huge traffic volumes, auto-scaling, and load balancing. Each component has been tailored for low latency, high performance, and dependability, and managed services are available to cut down on operational expenses. The architecture is well-designed for a high-performing system, but as the system develops, monitoring and optimization are required.