Microservices Based Application on Azure Cloud for Enhanced Patient Experience

Customer Overview

The customer delivers innovative technology solutions to meet the diverse needs of their clients. Using a consultative and collaborative approach they architect efficient and affordable solutions to enhance productivity, security, and profitability for their clients.

The Business Context

In partnership with Cisco the customer produces a range of networking solutions that includes signage solutions. These signage solutions run on a media player (Alto box) attached to a screen. Applications compatible with the Alto box enable a wider reach through installations of hardware bundled with the software, wherever required. The customer wanted to develop one such software-based signage solution for the healthcare domain.

The current monolithic, on-premises solution had limited accessibility, responsiveness, scalability, and incurred high infrastructure costs. Although it was suitable for areas like public lobbies and waiting rooms, building a patient-centric solution considering their bedside needs was important. Hence, the requirement was to develop a microservices based application hosted on cloud that would run on the Alto box. Bundling the software with the Alto box would also lead to increased hardware sales and SLA contracts to support both hardware and software, resulting in better business opportunities and increased revenue.

Type of Service Provided

Application Development

Technologies Used

Web Applications, Web Frameworks, Containerization, Azure Cloud

The Solution

The GS Lab | GAVS team of experts created an application deployable on Azure cloud with backward compatibility for on-premises deployment. This deployment flexibility makes it suitable for hospitals of all sizes. Integration with Azure services like Active Directory (AD), Postgres Service, and Key Vault drive increased application availability and durability. Containerized application microservices allow them to be deployed over any cloud service provider with minimal changes. Additionally, a new patient app with rich UX and compatibility with the Alto box hardware was developed to address poor responsiveness in the legacy application and as a feel-good factor.

To find out how GS Lab | GAVS can help your organization, please visit
www.gavstech.com

Challenges

The challenges with the existing solution:

- Accessible only to hospital inpatients and not to outpatients or those discharged

- Single tenant per application deployment since there was no multitenancy

- High infrastructure costs due to on-premises solution

- Not scalable due to monolithic architecture

- Could only run on legacy Geniatech devices using older JavaScript versions; not with SDKs compatible with the Alto box

- Not responsive and did not support 2K or 4K screen resolutions

Solution Highlights

- Deployable on-premises (supporting Debian and RHEL) and on Azure cloud

- Option for cost-effective single VM deployment or more robust Kubernetes cluster

- Simplified infra setup, application deployment and upgrade with RPM and Debian packages

- Horizontal scalability through microservices-based architecture

- Multi-tenancy support

- Strengthened CI/CD processes

- Remote application access for patients with Azure AD

- Industry grade UX inputs and comparative analysis for new patient app design

- Fully revamped design of video library app screen based on Netflix and Amazon Prime

- Enhanced picture quality on patient app through 4K resolution support

- Analytics on patient, staff activities for hospital use

Solution Impact

- ~50% savings on infrastructure costs for cloud deployment as compared to on-prem deployment (when cloud solution caters to 10+ tenants)

- Increased staff productivity and efficiency through streamlined digital workflows and software-based automation

- Higher HCAHPS scores for better Medicare/Medicaid reimbursements

- Enhanced patient satisfaction and loyalty

- Improved clinical outcomes and quality-of-care

- 10K+ app downloads from Google Play Store

Microservices Based Application on Azure Cloud for Enhanced Patient Experience

Solution Details

Core Features

- **Microservices – The monolithic java application was logically broken down into multiple microservices followed by the containerization of those services.**

  ○ Created multiple Maven modules to represent each microservice. Refactoring of the code shared by all services was added in common modules.

  ○ Introduced distributed caching using Hazelcast to enable access to shared data from all microservices.

  ○ Inter service communication was done with REST APIs.

  ○ Introduced additional pipeline jobs to build microservices.

- **Multitenancy - Used Hibernate to achieve multitenancy to separate tenant specific data.**

  ○ Implemented schema-based database multitenancy option provided by Hibernate ORM.

  ○ Modified entire application cache design to store tenant specific cache data.

  ○ Added tenant parameter in the analytics system to generate tenant specific analytical reports.

- **Azure Integration - Integrated with Azure services like AD, Key Vault, and Postgres and migrated the infrastructure to AKS service.**

  ○ A separate tenant specific Azure AD was created to onboard patient and staff. Created Azure app registration to configure authentication, role assignment, tenant branding, etc. User authentication and authorization with Azure AD is with OAuth 2.0 and Java SDK.

  ○ Provided option to integrate with Azure Postgres Service to store tenant data on cloud, and in turn avail benefits of data encryption, backup, and replication from the cloud managed service.

  ○ Integrated with Azure Key Vault to store certificates, secrets, and encryption keys securely.

  ○ Added Kubernetes layer to run the application on Azure Kubernetes Service. In the process, added Azure Application Gateway Ingress configuration and Azure Disk and Storage accounts to store persistent data. Implemented Helm charts for each service to seamlessly deploy services on AKS.

- **New Patient App – A new patient app was created considering the limitations of the existing patient app.**

  ○ Created React based patient app to address design concerns like model, view, and controller separation.

  ○ Used Material UI as a component library to get out of the box responsiveness capability. This allowed the application to run on 4K displays.

  ○ UX and accessibility were taken into consideration while designing each screen.

Additional Features

- Seamless integration with hospital EMR systems

- Password and PIN based authentication for patient app

- Video call feature with doctors and family in integration with vendors like Cisco Meeting Server and Virtual Care Room

- Video Library screen design inspirations from Netflix and Amazon Prime

- Improved TV Guide screen for better experience

- Integration with RTLS system to show staff movement and care team in patient app

- Option to customize patient app with choice of themes

- Support for English and Spanish languages in the app

Microservices Based Application on Azure Cloud for Enhanced Patient Experience

- Integration with HashiCorp Vault for Encryption as a Service to encrypt Patient Health Information (PHI)

- Monitoring of services using Azure Application Insights

- Multiple analytics reports for the hospital to understand app usage patterns

- Application compatibility with Android and iOS tablets

- Automated builds on every merge completion

Value-Adds

- Reiterated to the customer about the importance of investment in UX for the new application. Successfully delivered the Video Library screen developed from ground up with UX considerations and delighted the customer with the end result.

- DevOps was entirely handled by the GS Lab | GAVS team, showcasing our expertise by migrating existing services into Docker and Kubernetes. Although there was a huge learning curve, the team was able to successfully set up a very streamlined DevOps process that can be easily maintained and extended as required.

- The team periodically suggested process related improvements especially in Agile workflows like conducting regular sprint retros, time tracking in JIRA, etc. Standards set by ISO 13485 audit made this rigorous and helped process adherence.

- Having worked with the customer for almost a decade, the GS Lab | GAVS team has not only understood the product completely but has built a strong relationship, ensuring high CSAT scores each year. The customer acknowledges the team's reliability and product, domain, and technology expertise.

- Although the team has worked extensively on backend projects during the initial years, the customer entrusted the team with UI and DevOps work after being convinced about expertise in those areas as well.

Solution Architecture
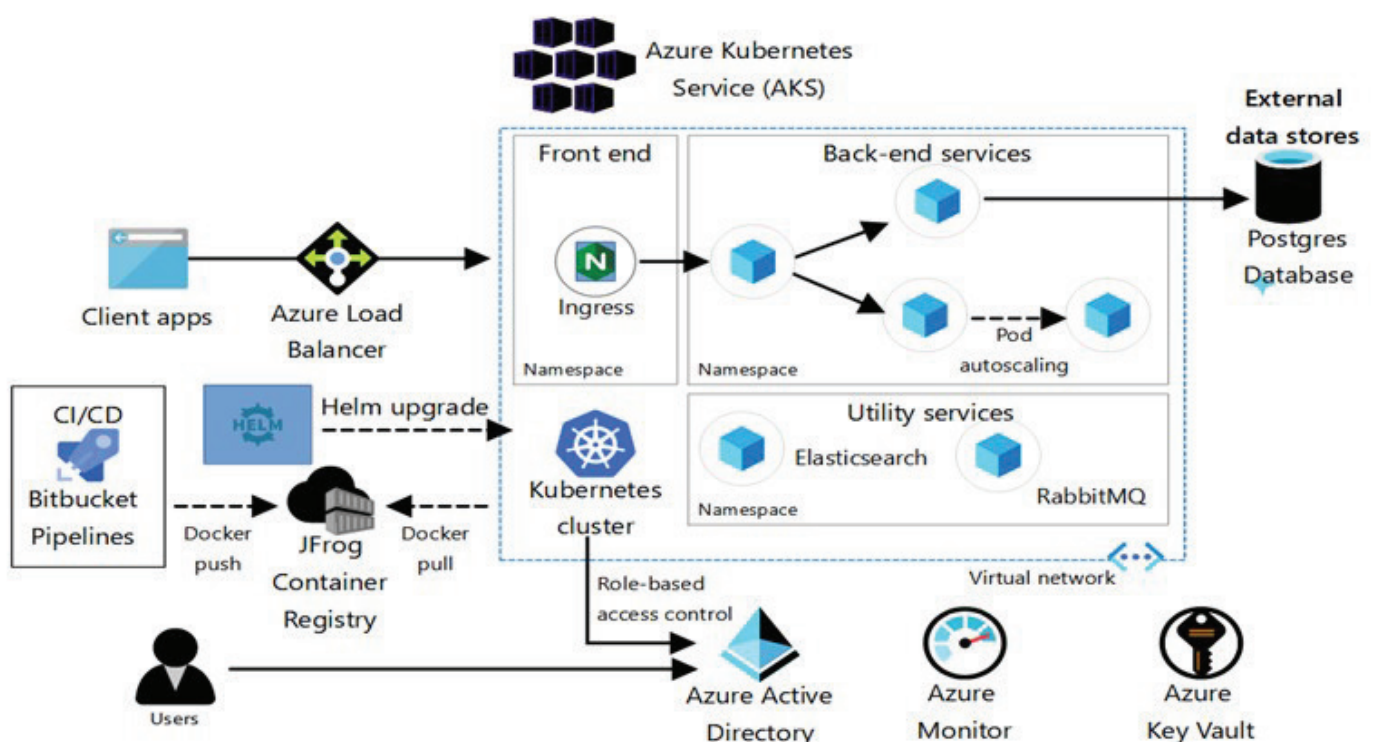
Microservices Based Application on Azure Cloud for Enhanced Patient Experience

Solution at Work