App modernization - Whiteboard design session student guide

November 2023

The names of manufacturers, products, or URLs are provided for informational purposes only and Microsoft makes no representations and warranties, either expressed, implied, or statutory, regarding these manufacturers or the use of the products with any Microsoft technologies. The inclusion of a manufacturer or product does not imply endorsement of Microsoft of the manufacturer or product. Links may be provided to third party sites. Such sites are not under the control of Microsoft and Microsoft is not responsible for the contents of any linked site or any link contained in a linked site, or any changes or updates to such sites. Microsoft is not responsible for webcasting or any other form of transmission received from any linked site. Microsoft is providing these links to you only as a convenience, and the inclusion of any link does not imply endorsement of Microsoft of the site or the products contained therein.

Updates November-2023, Christian Nagel

Contents

App modernization whiteboard design session student guide

Abstract and learning objectives

In this whiteboard design session, you work with a group to analyze and design a solution for moving legacy on-premises applications and infrastructure to cloud services. As part of the modernization effort, you will discuss modern concepts such as Serverless.

At the end of this workshop, your ability to design and implement a modernization plan for organizations looking to move services from on-premises to the cloud will be improved.

Step 1: Review the customer case study

Outcome

Analyze your customer's needs.

Timeframe: 15 minutes

Directions: With all participants in the session, the facilitator/SME presents an overview of the customer case study along with technical tips.

1. Meet your team members and trainer.

2. Read all directions for steps 1-3 in the student guide.

3. As a team, review the following customer case study.

Customer situation

Parts Unlimited is an online auto parts store. Founded in Spokane, WA, in 2008, they are providing both genuine OEM and aftermarket parts for cars, sport utility vehicles, vans, and trucks, including new and remanufactured complex parts, maintenance items, and accessories. Its mission is to make buying vehicle replacement parts easy for consumers and professionals.

In its first year, Parts Unlimited opened fourteen stores in three states: Washington, Idaho, and Oregon. The company established a 12,000 square foot distribution center to serve customers in the area. In two years, annual sales volume rose to $11 million. Parts Unlimited saw its business come from 40% professional and 60% do-it-yourself (DIY) omnichannel customers. With the situation surrounding COVID-19, the company saw a surge in online orders from professional and DIY customers. Unfortunately, their infrastructure and their team were not ready for the spike in e-commerce transactions.

Parts Unlimited has a hosted web application on its internal infrastructure and using a Windows Server, Internet Information Services (IIS), and Microsoft SQL Server to host the solution. All servers are located in an onsite server room in their distribution center. The office for their IT staff is just across the server room. Their IT staff includes a hardware technician, a system/network administrator, two software engineers. Their technical support is outsourced to a third-party service provider in Indonesia.

The web server that hosts the e-commerce website is recently updated to Windows Server 2019 Datacenter. The e-commerce application is tailor-made by a software development company that is now out of business. The application is built on .NET Core 2.2.207 that hit the end of life on December 23, 2019. Their team tried to change the version number in a configuration file to a recent version, but it crashed the website. They rolled back and left it as it is. They understand that they have to migrate to a more recent version of .NET Core, but they do not have the resources internally to make it happen. "We have to fix our scaling problems first. Then we can think of updating to something new," says Casey Jensen, Parts Unlimited's CEO. The source code left from the vendor has many solution files and codes that the team does not know if they are used. They open the primary solution file named PartsUnlimited.sln and deploy from Visual Studio into a folder on the server.

The team at Parts Unlimited is terrified to touch anything on the servers as long as it works. When they have to introduce new functionality or a bug fix, they schedule overnight deployments at 2 AM. This strategy has worked well so far, but it is not ideal. When a fix is ready, the team has to schedule deployment for 2 AM and wait. The scheduling of overnight shifts puts too much stress on the team and increases turnover in IT staff.

The SQL database used by Parts Unlimited e-commerce site is deployed on a separate server that has been there since the company was founded. It is a SQL Server 2008 R2 SP3 deployed on a Windows Server 2008 R2 SP1. The e-commerce application incurs ongoing maintenance costs in hardware, operating system updates, and licensing fees. These maintenance costs make Microsoft Azure App Service an attractive alternative. Their team is looking to migrate Microsoft ASP.NET applications and any SQL Server database to Azure App Service and Azure SQL Database. However, they are worried that their application might not be supported because of its .NET Core version being at the end of life. They wonder if they can move to the cloud now and migrate their application later or if the old version will be a show stopper.

Parts Unlimited has plans to increase its marketing investment, currently on hold because of scaling issues. The company is stuck and cannot grow without increasing its infrastructure footprint. Allegra wants to finalize their cloud vs. on-premises decision based on the current migration effort's success. CFO Jára Cimrman says, "We have to drive and scale our e-Commerce presence forward while controlling costs."

The engineering team is worried about their order processing subsystem. Currently, they have a strongly coupled order processing system that runs synchronously during checkout. When moved to the cloud, they do not want to be worried about their order processing system's scalability. They are looking for a modern approach with the least migration effort possible. They want to keep the changes and their investment into the current code base at a minimum.

Finally, Parts Unlimited is looking to invest in DevOps practices to decrease human error in deployments. They are looking for options to have a staging environment to test functionality before shipping to production. However, their team does not have any experience in building CI/CD pipelines. They are not sure if this goal is achievable in the short term, and they do not want it to hold up their migration to the cloud.

Customer needs

1. Parts Unlimited wants to assess its current environment to see if it can move its e-commerce site to the cloud as it is.

2. Parts Unlimited wants to move to the cloud and be able to scale its e-commerce solution.

3. They want to migrate their SQL Server 2017 database to a fully-managed SQL database in Azure.

4. Parts Unlimited wants to find a solution for their testing in production problem. They want to be able to test functionality before pushing it to their servers.

5. They want to minimize human errors in deployments. They heard about DevOps practices and that publishing from developer machines is not ideal.

6. Parts Unlimited is looking to separate its order processing subsystem and scale it independently to accommodate a large number of orders.

Customer objections

1. Our developers were not able to migrate our .NET Core 2.2 application to a more recent version. Should we expect a steep upgrade path with every new version?

2. When a .NET Core version is EoL (End-of-life), does that mean we cannot host our solution in Azure?

3. We hear a lot about Kubernetes. What is the difference between App Services and Kubernetes?

4. We have plans to scale to Mexico and Brazil. Anything we should be worried about while moving to Azure?

Step 2: Design a proof of concept solution

Outcome

Design a solution and prepare to present the solution to the target customer audience in a 15-minute chalk-talk format.

Timeframe: 45 minutes

Business needs

Directions: With your team, answer the following questions and be prepared to present your solution to others:

1. Who will you present this solution to? Who is your target customer audience? Who are the decision makers?

2. What customer business needs do you need to address with your solution?

Design

Directions: With your team, respond to the following questions:

High-level architecture

1. Without getting into the details, which you address below, diagram your initial vision for handling the top-level requirements for the e-commerce site, database, DevOps, and order processing.

Assessment and Migration

1. Where should Parts Unlimited start its assessment and migration journey? Is there a single place to start and find all the tools and services?

2. What tools would you recommend Parts Unlimited use to assess and migrate its web application? How would you use these?

3. What tools would you recommend Parts Unlimited use to migrate its database? How would you use these? Be specific.

4. What Azure service would host the website?

5. Would there be any problems with the .NET Core version being EoL (End-of-Life)?

Modernization

1. What Azure service would provide a scalable, serverless solution for order processing that handles unexpected spikes and can be implemented with the least amount of code change required in the web application?

2. Parts Unlimited wants to create PDF invoices. What would be a serverless way of implementing it?

3. How can Parts Unlimited monitor its application performance in the cloud?

4. How can Parts Unlimited make sure their SQL Database in the cloud always has the right amount of resources to accommodate unexpected spikes in load?

DevOps

1. Parts Unlimited wants to test new functionality and bugfixes before shipping to production. What would you suggest?

2. Parts Unlimited team is familiar with GitHub. How would you suggest them to set up a CI/CD pipeline?

Prepare

Directions: As a team:

1. Identify any customer needs that are not addressed with the proposed solution.

2. Identify the benefits of your solution.

3. Determine how you will respond to the customer's objections.

Prepare a 5-minute chalk-talk style presentation to the customer.

Step 3: Present the solution

Outcome

Present a solution to the target customer audience in a 5-minute chalk-talk format.

Wrap-up

Discuss the proposed solution with the larger group.

Additional references

Description	Links
Azure SQL Database serverless	https://learn.microsoft.com/en-us/azure/azure-sql/database/serverless-tier-overview
The .NET Upgrade Assistant	https://learn.microsoft.com/en-us/dotnet/core/porting/upgrade-assistant-overview
Choosing Azure compute platforms for container-based applications	https://learn.microsoft.com/en-us/dotnet/architecture/modernize-with-azure-containers/modernize-existing-apps-to-cloud-optimized/choosing-azure-compute-options-for-container-based-applications
Azure Monitor Application Insights documentation	https://learn.microsoft.com/en-us/azure/azure-monitor/azure-monitor-app-hub
Azure Functions hosting options	https://learn.microsoft.com/en-us/azure/azure-functions/functions-scale
Azure Migrate documentation	https://learn.microsoft.com/en-us/azure/migrate/
GitHub Actions documentation	https://docs.github.com/en/actions