AutoPets empowers pet owners with an enhanced IoT user experience.

Without interrupting legacy systems, AutoPets was able to modernize its server infrastructure and transform its customer-facing application experience.

Our customer

AutoPets is an innovative pet care company working to make pet care more enjoyable and convenient through the development of highly functional, IoT-connected pet products — such as its iconic Litter-Robot™ Connect.

The obstacles they faced

Facing issues related to onboarding new customers and struggling with unacceptably low mobile application speed and reliability, AutoPets needed an infrastructure overhaul before a critical new mobile app launch.

How we helped

AWS IoT Core, Amazon Kinesis, AWS Lambda, Amazon Elastic Compute Cloud (Amazon EC2), Amazon API Gateway, Amazon DynamoDB Cloud Infrastructure, Cloud Native Application Development.

What we achieved together

By leveraging the expertise of Onica (a Rackspace Technology company), AutoPets increased its product reliability and scalability, while lowering its operating costs.

auto·pets

Bringing pet care into the future with IoT

AutoPets is an innovative pet care company working to make pet care more enjoyable and convenient through the development of highly functional, connected pet products. Internationally known for the Litter-Robot Connect, an automatic self-cleaning litter box for cats, AutoPets continues to reimagine pet care as it develops creative IoT product solutions for its customers.

Improving app speed and reliability on AWS

While AutoPets leveraged Amazon Web Services (AWS) for Litter-Robot, the system didn't use any managed services such as AWS IoT Core. This meant the company wasn't getting the full benefits of a cloud-native solution. The product's software limitations were hindering its ability to scale, while also creating slow app speeds and unreliable performance.

Creating a better user experience

Onboarding new units to the system was difficult for users, with about 75% of customers failing on the first try. Action response times varied greatly. Sometimes it was nearly instantaneous, while other times it could take up to 45 seconds.

AutoPets had already completed its iOS application and was preparing to launch its Android application. However, due to the user experience and onboarding issues it faced with the iOS application, the team was nervous about adding new Android users to the current application backend infrastructure.

"We view Onica as our software technology partner. Onica is helping to develop our software technology and mobile products, which are used by pet owners around the world. Six months into the partnership and we're increasingly satisfied."

Jacob Zuppke
Executive Vice President,
AutoPets

Due to these reliability concerns and its deadlines for its new Android application, AutoPets determined that it needed to engage an experienced AWS integrator. AutoPets determined that it needed to engage an experienced AWS integrator and reached out to Onica.

"Launching the Android app compared to launching the iOS app was like night and day. With far fewer users at the time of our iOS launch, the iOS application fell flat twice, with both server outages and failures. By the time we launched the Android app, we had many more users coming online — 10,000+. It was much more successful thanks to the changes we made with Onica."

Jason Weihman – Research & Development Manager, AutoPets

Improving stability and cost savings on AWS

The Onica team came aboard and delivered in a tight timeline. The first priority was completing IoT firmware for the next-generation Litter-Robot, which needed to go through several hardware certifications. The second priority was a successful Android application launch. The goals were aggressive, including meeting a firmware deadline within 45 days from the start date.

The AutoPets team's number one goal was stabilizing its AWS infrastructure to ensure the Android app could launch without experiencing the same onboarding issues it experienced with its iOS launch. In addition, some updates to the hardware were necessary for the AWS IoT Core service to be used natively. This included designing the provisioning and onboarding process, implementing the communications firmware, building the cloud infrastructure for native use, updating the existing iOS application and updating the initial work on the Android application to ensure interfacing with the new device was successful. The results were instantaneous.

Accounting for legacy while planning for the future

The biggest challenge the AutoPets team faced was related to its legacy devices. Since the Litter-Robot is already in the customers' homes and can't be updated (there are no over-the-air-updates in v1.0), there were major challenges around what could be done to improve and update legacy hardware. This was described by the team as "changing the wheel of a car while it's going 60 mph." It was important that changes to the infrastructure were made incrementally without disrupting service to existing customers.

The existing infrastructure was built with Amazon EC2 instances that supported the legacy devices and, therefore, had to remain. However, the team was able to add automation to improve scalability, and modernize and streamline infrastructure for existing devices to create more efficient use of AWS services. This also improved costs by storing messages in Amazon DynamoDB and provisioning with Amazon API Gateway.

For everything built new, the focus was on managed services and serverless development. The process began with AWS

IoT Core and a rules engine for basic data ingestion. Amazon Kinesis streams were used to buffer data, and AWS Lambda was used to process data. As with the old infrastructure, data was stored in Amazon DynamoDB for long-term storage. Short of the requirements for legacy devices, there are no servers involved in the new process.

Greater reliability and scalability with lower costs

AutoPets' infrastructure changes have greatly improved reliability. With so many Litter-Robot units sending messages every few seconds, the previous infrastructure was unable to handle the rate of messaging. Using Amazon Kinesis for buffering allows batching of messages rather than individual processing. This prevents the kind of overload the infrastructure was previously facing. The native use of AWS IoT Core services has also greatly decreased server costs compared to the previous infrastructure setup, while allowing for greater scalability as new Litter-Robot units are released.