

Capmation helps Park Industries gain efficiencies with automated remote updates.

Service Provided



Full Stack Development

About Park Industries

Park Industries is a machinery manufacturing company focused on designing, manufacturing, selling, and supporting industrial solutions for the stone industry. Established over 70 years ago, Park Industries brings its experience in customer service to provide products and services to the stone fabrication and custom manufacturing markets. It's currently sold over 18,000 machine solutions in North America.

Benefits

- Elimination of time-consuming manual processes involved in updating remote machines.
- Increased efficiency by enabling multiple machine updates at one time.
- Improved operations by creating communication with remote machines to monitor their status and detect issues.

Their need

Park Industries builds high-precision machines for stone and metal fabricators. With over 18,000 machines sold across the U.S. and Canada, the company needed a more efficient way to monitor the status of its remote machines and automate the process of updating the machine's software.

To update the remote machines, Park Industries had to contact each customer to validate the status and then have someone onsite update the software. This required administration time for both Park Industries and its customers.

Automating this process would not only reduce or eliminate administration time, it would also develop a critical line of communication between the remote machines and the company's main computer.

The solution required an instruction input from the main computer that leveraged the company's Azure SQL platform to implement SQL communication to pull files, write logs, and place the files back on the remote machines.

Developing integration between the main computer and the customers' infrastructures through previously developed APIs could come with challenges and risks, considering the application had to be compatible with different Windows Operating Systems.

Mike Rankin, Project Engineer at Park Industries, created a precise list of requirements and turned to the Capmation team to bring the project to completion.

Our solution

The project began with a discovery phase to provide the Capmation team with a clear understanding of the company's needs and projects requirements. From there, the team set up the DevOps environment and repositories in the Sprint 0 phase. The team was also given access to resources it would need to complete the project.

Sprint 1 consisted of dividing the work into three components: Database, API, and Windows Service. The team established connectivity with the API and imported the initial data for testing purposes. Next, the API was developed with the endpoints needed to communicate with the Database, and connection was confirmed. Lastly, information was gathered for the actions that would execute and create the connection between the Windows Service and the API.

At this point in every sprint, the Capmation team developed specific Proof of Concepts demos that would help significantly reduce the testing time down the road.

Sprint 2 focused on refining the implemented components of the previous sprint and began the integration needed to begin testing the download instructions. On Sprint 3, the first version of the application was released. This included delivering the working installer to the customer so they could start the testing process on a virtual machine that complied with corporate policies. Sprint 4 included additional testing and data encryption of all files that were used throughout the communication process between applications.

Sprints 5 and 6 were dedicated to completing project documentation and fine-tuning all the project's functionalities and security features. In addition to the written documentation delivered, the Capmation team provided a central repository for easier access to all information, which will give customers a better understanding of all the components used.

While the initial scope included only the distribution of update files, it was later modified to capture other types of data that would provide Park Industries with an accurate status of every machine in the field. This information would give the company important insights on its machines, enabling it to provide proactive maintenance and customer service.

“ We went from an individual taking an hour or more of their time to update a machine to literally just queuing up an update and the machine automatically performing that update. It’s significant.”

Mike Rankin

Project Engineer • Park Industries

Results

The success of the project means what once took countless hours to update remote machines can now be done with a push of a button.

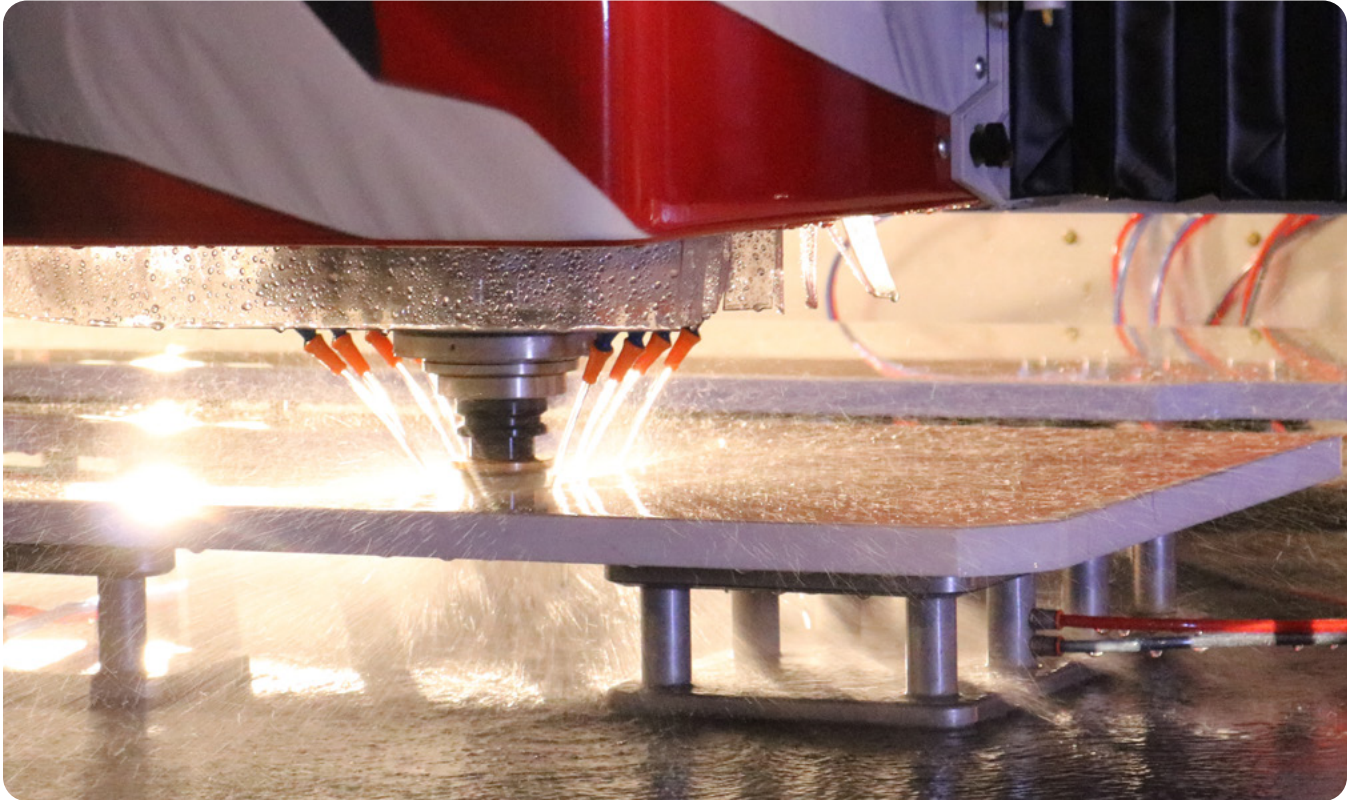
“We went from an individual taking an hour or more of their time to update a machine to literally just queuing up an update and the machine automatically performing that update. It’s significant, and it allows us the ability to update multiple machines at once,” said Rankin.

He says the project was successful, in part, because of strong collaboration with the Capmation team.

“Communication was extremely good. We developed a very open rapport very quickly, and they did not hesitate to ask me when they were uncertain about something. We had a daily stand up and open communication via Slack, so there was constant communication. Even a 60-second conversation can make all the difference in the world” he explained.

In addition to effective communication, Rankin said having a clear set of requirements kept the team on course.

“They took seriously the requirements we laid out before them, and they strove to fulfill those requirements as written,” he concluded.



“ They took seriously the requirements we laid out before them, and they strove to fulfill those requirements as written.”

Mike Rankin
Project Engineer • Park Industries

Length of Project

Six sprints over the course of three months.

Technology Used

Cloud – Azure Portal

- Blob Storage
- Azure SQL
- Park IQ APIs

Windows Service

- SQL Lite DB

Development Language

- C#
- JSON instructions

CI/CD

- Azure DevOps