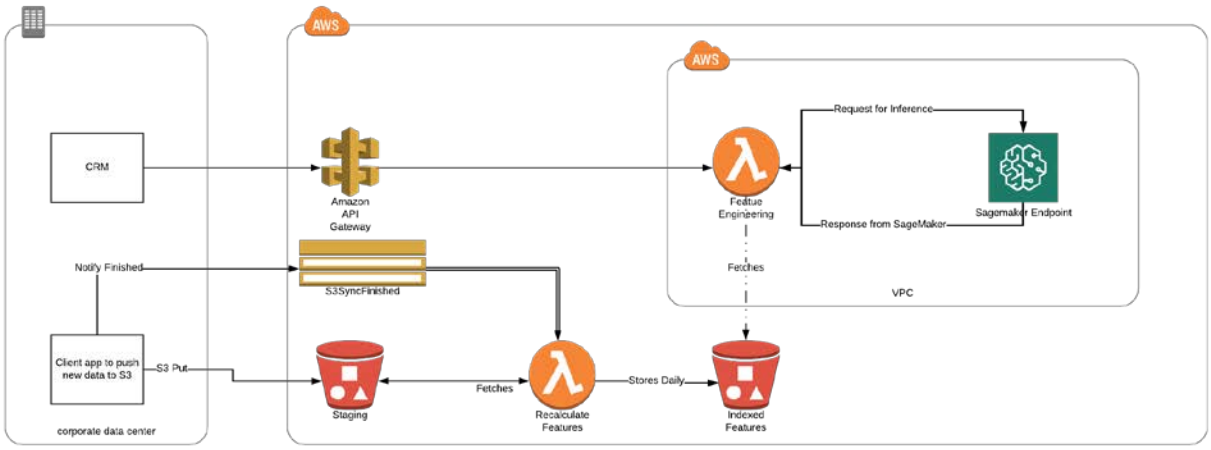
The Challenge:
A Fortune 500 company specializing in the trade of electronic components is utilizing a custom-built CRM system to manage its trade processes. With tens of thousands of new requirements they receive every week, the buyers have little information regarding which requirements that they should work on, causing opportunity losses and potential revenue drain.

- Many requirements are forgotten due to volume of requests or shortages in the market.
- A solution to prioritize the requirements existed, but it didn't account for many important factors such as number of active vendors, cross-product sales, current product trends, and quantity requested, rendering it unrealistic, inaccurate, hence often being ignored.
- The client's data science team had tried to improve the scoring solution and built out on-prem sandboxes, but the servers were limited in their ability to scale with more memory-intensive algorithms and data scientists were confined to using a small set of data.

PREDICTif Solutions was brought in to leverage machine learning to help improve the requirement scoring system, increasing gross profit. The project was dubbed "Requirements Prioritization v2".

The Solution:
Working alongside the client's data science and IT teams, our solution architects executed a 3-month plan to develop an improved requirement scoring model and integrate it with CRM to provide a more precise prioritization for the buyers. We recommended using a serverless AWS machine learning stack that includes SageMaker as well as API Gateway, Lambda and SNS for notifications. The below picture illustrates a high-level architecture design.



SageMaker is an all-in-one machine learning environment. With it, we were able to provide a sandbox to code, train, and test various models utilizing Jupyter Notebooks and then deploy trained models as API endpoints that CRM integrated with. SageMaker was selected to simplify the process of productionizing ML workloads. It provides a dockerized environment and powerful API's to deploy trained models as a microservices architecture which can then be accessed via other AWS services. To reduce time, we also leveraged the hundreds of AWS-provided algorithms to develop our new requirement scoring algorithm.

AWS offers a very rich set of API services, which have made it very easy to integrate with CRM. Every service in AWS exposes an API. SageMaker has an intuitive UI exposed through the AWS Console, but it was the management API that really excited us. Without having to incorporate any third-party tools, we were able to start/stop, schedule, and promote training jobs in a continuous integration fashion to automatically roll out newer models.

Challenges Addressed

- **Model Retraining** - while the process of building a new model is relatively simple, model retraining for comparing performance, approving changes, and testing with live data poses several challenges. We incorporated a SageMaker feature called production variants to simplify the process. A variant allows the user to deploy multiple models to the same endpoint and declare what percentage of the traffic will go to each model. This way, a new model can be tested until confidence is high enough to remove the old one.
- **Parameter Tuning** - One of the most time-consuming phases of developing ML projects is hyper-parameter tuning: the art of tweaking the configuration parameters that control model training. SageMaker hyper-parameter tuning jobs helped by allowing us to choose a performance metric to maximize. After each job was finished, 20 independent training jobs had been run, each using the output of the last to enrich and further optimize our performance metric. The SageMaker Console provided an easy-to-use comparison tool, enabling us to quickly identify the right hyper-parameter combinations.
- **Cost Management -** After determining the target algorithm would be an AWS provided XGBoost algorithm, the DS team wanted to try using 1-hot encoding, wherein every possible product was converted to a feature. This made the final feature set very wide (~700,000) columns, which was much better for CPU performance but needed more memory for each training iteration and endpoint call. A Cloudwatch event was created on a schedule to start/stop notebooks and endpoints during non-work hours, thereby cutting development and testing costs by 60%.
- **Security -** Since a fully-deployed ML cloud solution was brand new at this client, there was skepticism and concern around utilizing cloud, particularly regarding security. AWS shared responsibility model simplifies the project teams task list
    - API Keys were used for communication between on-prem and API gateway
    - S3 transactions are all SSL encrypted by default, and we enabled encryption at rest in the S3 bucket.
    - All information transferred between services within an AWS account is secure and monitored.

The Results

PREDICTif Solutions has been innovating exciting solutions for our clients for over a decade now, so it was satisfying in using some of most cutting-edge technology that AWS offers to breath new intelligence into an older, deterministic CRM system for this client, that has resulted in an increase of profit margin by over **30%**, after just the first phase of this project.

- **Productivity Increase –** the productivity of the data science team has increased by over **200%** with the AWS machine learning technology. This project would have taken at least 6 months to deliver if we had done it using on-prem sandboxes
- **Development Cost Reduction** – A TCO analysis revealed a solution of this size would have cost **3x** to host on prem and taken many months to procure. By using AWS, the DS team began building models from day 1. A notebook instance was created in minutes, rather than weeks of waiting for servers to be racked, or clusters to be configured. With SageMaker's 1-click deploy methodology, architects could integrate developed models much faster because we weren't waiting on DS team to finalize the feature set
- **Solution Portability** – The ML model exists in a docker container, which can easily be ported to another cloud provider or even on prem if the client wanted

The client's executive leadership wanted to pick something with a low up-front cost, and minimal impact to business process until the value could be proved to in front of a larger audience. We were able to deliver on this mandate and provide an impactful solution to move the business at the speed that the market demands.