Validic

by Former Executive Director of Product Management

Remote Patient Monitoring

Details

Review Date	12/05/2023
Purchase Date	Q1'21
Implementation Time	3 months
Product Still in Use	Yes
Purchase Amount	~250k/year (per user per month)
Intent to Renew	100%
Review Source	Elion

Product Rating

Product Overall	4.0
Use Case Fit	4.5
Ease of Use	4.0
API	5.0
Integrations	N/A
Support	4.5
Value	3.5

About the Reviewer

Purchasing Team

Implementation Team

Product Oversight

Reviewer Organization

Commercial Health Plan

Reviewer Tech Stack

N/A

Other Products Considered

N/A

Summary

- **Product Usage: Validic was used for collecting and standardizing data from various health devices like Apple Watch, FitBit, and blood pressure cuffs, thereby helping the user keep track of their health data more efficiently.**
- **Strengths: Validic provides excellent device coverage, good uptime, a very clear and robust API, and robust technical support.**
- **Weaknesses: Due to Validic's model, when a device includes a new sensor or data, clients have to wait for Validic to update their system causing delays; additionally, the service can be relatively expensive.**
- **Overall Judgment: Validic is good at what they do, providing support, solid uptime, and expansive device coverage, making it a worthy investment in contrast to building a similar solution internally.**

So today we're chatting about Validic and how it's used at your company. Before we jump into that, could you give a brief overview of the company and your role there?

I was leading development for one of the wellness products our company had. The product was responsible for giving people sleep, mindfulness, exercise, and nutrition advice, with the end goal of hopefully keeping people out of the emergency room, which is really costly. Validic was already in use by one of the products at our company, and we weren't using it yet, but were evaluating whether to use it or to build a solution ourselves. As we were looking to expand the devices we supported, we ultimately decided to work with Validic rather than build it ourselves.

What was the need that drove you to look for a product like Validic?

So Validic is most relevant for pulling in a lot of different kinds of connected device data. If you only want Apple Watch data, you don't need Validic. You can just pull directly from the Apple Health app. But if you want to support Apple Watch and Fitbit and four different brands of blood pressure cuffs and glucometers, that's when a product like this is useful. Validic standardizes the format for data from all these different sources so your engineering team only has to work with one kind of data, instead of a different format for every product you're importing it from. There's an upfront benefit in not needing to do this normalization ourselves, but there's also an ongoing benefit, because Validic also connects to all these different manufacturers' data feeds and adds new device support as new products get developed.

Doing this in-house might have looked cheaper since we wouldn't be paying Validic, but it's not like the work is free to do: you have to pay engineers to set it up and monitor and maintain it over time.

What requirements were you looking for in a product like this?

When we were looking, there weren't really any other established competitors: it was just Validic. So for us it was really a build versus buy decision. First we identified the universe of devices we knew we wanted to support over time. The broader that list was, the more likely we'd need a solution like this. There were enough different devices that we wanted to support, all with their own data structures, that we decided it made sense to buy.

We were also looking at uptime and device throughput time. For an end user, if they're looking at their wearable and then at your app, it's pretty important that those two be very closely in sync. Otherwise, the user is going to perceive that your app is broken, when it might just not have fully updated yet. It looks bad on us if they have high latency or a lot of downtime: to the user, it looks like our systems are broken, because no one would know the problem is actually on Validic's end.

Are there any competitors to Validic now that you can compare them to?

I've seen a company called Rook; they seem to be doing almost exactly the same thing as Validic is doing. They probably realized there's a big market for this and not a lot of competitors.

What was the business value of the data you were trying to acquire?

It was foundational to the product. It was really centered around getting people to exercise more, so pulling in their wearable data was about as foundational as it gets. At my current company, we're thinking about doing the same

thing, but we also want to know what's happening with their blood sugar or blood pressure. For a healthcare company providing care in a value-based care world, it really matters to know what's happening to your patient in between visits.

How was the sales and onboarding process with Validic?

They were very polished. We had very good support. The people they assigned to us knew what they were doing; we didn't get the B team! It could be because we were a very big account for them, but in general, they seem like they have their act together.

How was the implementation process?

It was not very complex. They have very good API documentation and good technical support to answer your questions and help you find solutions. They're readily available and got back to us quickly.

How did Validic incorporate into your workflows?

The overall thing Validic does is get a user's data from the device vendor, to your back end, in the same structured format as all the other data, and from there you can do whatever you want. They get the user's authorization to access their data, similar to how Mint does it with your bank accounts. When they normalize the data, they make sure it all shows up for us in the same format. So for example, say Apple Watch stores distance walked in kilometers, whereas Fitbit stores it in miles, and the fields are labeled differently in their API. So Validic converts the data into the same units and sends it to us under the same label. You don't have to do anything special to work with it.

Are there any additional features or services that Validic offered that you didn't use?

They've recently expanded into trying to provide more intelligence on top of the data. I think as they look to the future, they probably saw that more competitors are going to be able to offer normalized data. So they were starting to look at, for example, alerting on which users have fallen off their baselines or which have out of range blood pressures. We weren't using them for that, but that's a new area for them.

How would you characterize the relative strengths and weaknesses of Validic's service?

Their device coverage is very strong. They've been around for a long time, so they've had a lot of time to build up the library of devices that they support. Newer entrants into the market will start with Apple, Fitbit, and Garmin, but after that there's a long tail of other kinds of devices that you'd have to add over time. And Validic has already had time to build out that long tail.

They're also pretty good in terms of uptime and stability and performance. The latency was a little longer than I would have liked, but it was acceptable for users. We didn't have any use cases around immediately alerting a user if there was a bad reading or an emergent situation, which would have been irresponsible, set us up for a lawsuit, and turned the latency from an annoying problem to an actual health risk.

They're relatively expensive compared to their competitors, partly because their competitors have to undercut them to win business. But I think Validic is kind of the Salesforce of what they do, where a lot of companies have only heard of Validic offering this solution, so they can set the price they want and people will realize it's still cheaper than building it themselves.

The one weakness I see, which is less their fault and more that of the model itself, is that if any devices add a new sensor or a new kind of data they're able to pass through, we won't get access to it right away in our back end. We have to wait for Validic to update their system. When Fitbit added the ability to share Heart Rate Variability, we had to wait a quarter for Validic to update their system so we could have access to it. If we had our own engineers, we could have told them to drop everything and work on this and get us access immediately. But Validic has a lot of priorities that they're balancing, so you can't necessarily expect that any upstream changes will be incorporated right away.

Apart from latency, did you see any actual issues such as errors in the readings, downtime, etc.?

There was occasional downtime like with any cloud-based software, but it wasn't a major issue. We had a cluster of a couple incidents once, but Validic apologized profusely and fixed it, and it didn't happen again.

How was the developer experience with the API and documentation?

It was very good. Their API documentation is very clear, it's robust. It's clear that this is not their first rodeo. We were able to figure out what we needed to integrate the data and pull it into our app very easily.

Looking back, do you think you made the right assessment moving forward with Validic?

I do, yes. Whether or not we'll choose them again will really come down to how they compare against the new entrants in the market on price and device coverage, but they're certainly in the running for the next time I need to make this decision.

Where do you think are the key areas for growth that would make Validic or a product like it exceptional to a buyer?

They all seem to be covering the wearable fitness tracker devices pretty well. That market is very concentrated among a few big players. It's the clinical devices that I think will differentiate one from the other.

Do you have any advice for someone going through this selection process?

Don't underestimate how hard it is to do it yourself, and how much it will cost in the long run!