Elion

Buyer Review Transcript

Product Reviewed	Customer.io
Integrations Reviewed	Segment, Mailgun, Twilio
Other Products Mentioned	Leanplum, CleverTap, Iterable, Braze, Mailchimp
Date	August 15, 2023
Expert Role	Chief Product Officer
Specialty	Clinical Follow-Up Technology
Payment Model	SaaS

We're discussing Customer.io and how you use it. Can you give a brief overview of your organization and your role there?

I'm one of the co-founders and the chief product officer of a company that focuses on identifying and tracking follow-ups in clinical notes.

With respect to Customer.io, can you give a little context about when you purchased the product and how long you've been using it?

We have been using it for over two years now. We're on a HIPAA-compliant version of their service because we exchange some levels of PHI, although it is limited. We mainly use it to communicate with patients about the need to follow up on a visit. We extract the follow-up recommendation made by the physician, which usually involves recommendations like coming back in six months for further imaging to evaluate an abnormality. We use Customer.io to send SMS and email reminders to these patients and stakeholders, urging them to take that action.

To provide some context for those who may not be very familiar with Customer.io, could you give a brief overview of what it is? Additionally, are you utilizing all of their capabilities, or are there specific areas that you are mainly focused on?

We think of it as somewhat like a CRM or marketing automation technology. Essentially, we provide it with user profiles and product-related events, which enables us to trigger other communications accordingly. These communications can take the form of SMS, push notifications, emails, or other types of messaging. Additionally, we receive reporting and analytics to measure the success or failure of these communication efforts.

What aspects of Customer.io have worked well for you?

They have an efficient process for message orchestration, including custom logic related to delivery and sequencing. They offer a lot of flexibility in setting up these campaigns, as they call them.

In our case, we determine what triggers a communication campaign, who should be included, and what actions to take once they meet the criteria to enter. For instance, we send three consecutive messages, alternating between SMS and email, based on whether we have a mobile phone number or email address, so that we can cover various communication channels.

Is there anything that doesn't work as well?

From our point of view, we haven't fully integrated with them yet. They seem to provide various options for sending data to them, and we faced challenges because most of the events we wanted to track were back-end related, rather than being focused on the user interface, which is where it seems like they started. So at first, we struggled with automating the tracking of back-end events instead of front-end UI events.

So, did you eventually manage to work through the problems and get it to work the way you needed it?

Initially, we started off with a different type of data integration with them. It was a more manual process, requiring a bit of extra effort. However, we're now transitioning toward a back-end integration approach, which I believe has been enhanced by their Python SDK.

Are there any features or capabilities that you aren't using?

We don't use push notifications at all. Our main communication channels are SMS and email. I believe push notifications are a fundamental aspect of their platform, but since we don't have a mobile app for our users, we don't make use of them. However, we do utilize a few data integration features, such as reporting webhooks for SMS and email, which have proven to be helpful, as well as a regularly scheduled data dump to a storage bucket. The platform also offers more advanced data warehouse integrations that we have yet to explore and fully utilize.

What integrations do you currently utilize with them, and were these out-of-the-box integrations with other solutions, or did you have to build them yourselves?

We integrate with Segment. This allows us to pass some data to them from our own application. For users who are using our web-based application, we automatically send them events. Otherwise, we manually upload CSV files on a regular basis to send them the data. Regarding how they send us data back, we have created a webhook API that ingests event data about SMS and email, which we then process and feed back into our own application.

Is there anything else to say about their integrations?

The good thing about them initially was that they already had a lot of email providers provisioned. For example, they already had Mailgun integrated, so we didn't need to go through that whole process ourselves. We did need to set up our own Twilio integration, but that was pretty straightforward and built into the product, so it was easy to connect our accounts. On the content personalization side, we make use of their liquid language. It allows us to use conditional statements and insert dynamic values like first names or phone numbers into the body of our SMS or email. We use it extensively to get better traction with the messages we send.

Can you elaborate on what you liked about the orchestrator, custom logic, and flexibility in the campaigns?

I really like how they have different sections for adding steps in a campaign. For example, you can add a message as one step and then add conditional logic as a different step, but it's all within the same campaign workflow creator.

One feature I find useful is the time-delays piece. That said, things can get a bit confusing when you're trying to specify nonspecific or relative dates, like "15 days ago" or "from today," and it can be tricky to figure out which option is the right one for your desired timing. On the other hand, using specific dates is fine and fairly easy to do if I just want to delay a message for a set number of weeks. That's pretty straightforward. But when it comes to creating an automated process for every person to enter the campaign with a relative date, such as within six months, it becomes a bit more complex.

In terms of the procurement process, who did you evaluate them against, and how did you ultimately decide to move forward with Customer.io?

I evaluated a few different vendors, including one where I used to work, so I was already quite familiar with the industry. In the beginning, the key factors we looked at were native SMS and email support, self-serviceability, and cost. As a startup, we needed something affordable but still capable, and this vendor fit the bill. They offered reasonable pricing for the messaging transaction volume they provided. Their pricing wasn't based on features, but rather transaction volume, which helped us to manage cost.

Who did you compare them against?

I was familiar with Leanplum, and they were acquired by CleverTap, so I considered evaluating them too. I also compared them against Iterable. I didn't delve too deeply into Braze or Appboy, but I did have them on my radar as other competitors to compare against.

You mentioned the main factors you considered in your evaluation, including native SMS and email support, self-serviceability, and cost. What were the one or two primary reasons that made you choose Customer.io?

I forgot to mention the fourth factor, which is that we handle protected health information (PHI). It was crucial for the company to be able and willing to sign a business associate agreement (BAA) with us. This agreement allows us to use our messaging system while ensuring HIPAA compliance. This was a basic requirement before considering any further partnerships.

Based on my previous experiences with Braze, I knew they would sign a BAA, but they had some limitations on the information they could receive. It was almost like they were telling you they were HIPAA compliant, but not entirely HIPAA compliant. I don't think Iterable signed BAAs at that time, so they were immediately ruled out. However, they might have changed their policies since then. As for Leanplum, I knew they didn't sign BAAs either, but I was familiar with their capabilities. Overall, from a maintenance perspective, Customer.io was the easiest to set up.

From a pricing perspective, you mentioned they were more volume based, as opposed to feature based. Can you elaborate on that?

They provided us with a generous number of built-in messages and profiles that exceeded our needs. I believe we were given a million messages per month included in the price, along with nearly 200,000 unique profiles. We were cautious about this and made sure not to exceed these limits.

In addition, since we were handling various instances for our clients, such as radiology and hospital groups, it was important to keep their environments separate to avoid cross-contaminating user data. And we didn't want to reach the upper limit of their volume pricing, so that we could use the service for all our clients without worrying too much about cost scaling when adding more clients.

Was this a product-led sales process, or did you work with a sales team to purchase the product?

It was a product-led process.

How was the implementation and setup experience?

Regarding how we transmit data to them, initially the process was quite simple, and the onboarding was largely self-service. They also provided helpful documentation in case we had any questions. Support was available if we needed it, but we didn't really have many major questions. I found the documentation to be fairly clear, outlining the steps for implementation.

How long did it take you to get up and running?

I would say it took no more than a week. Maybe not in real time, but in terms of actual time. I'm not really sure how much time I actually spent doing it, but we signed up and got it live within a week.

Have you had to use their support since you've been working with them? If so, what has that experience been like?

We have two support contacts and a dedicated CSM who we can contact for escalations. We usually reach out to their support team through chat or email. They're not always available, but usually get back to you pretty quickly if not.

When we were just getting started, they were responsive to hopping on calls for technical implementation questions. They usually respond quickly with either a solution or letting us know that it's on their roadmap and they will escalate it. I would say we have been reasonably satisfied with the support we've received from them.

Is there anything else we haven't touched on regarding the product that's worth mentioning?

We use some of their broadcast features for sending out notifications to a large group of people at once, so we don't have to use a platform like Mailchimp. That's another benefit we've found. Additionally, the separation of different workspaces is helpful as mentioned earlier. Their user permissions also offer a lot of flexibility in terms of what users can do on the platform, which is quite useful too.

Let's wrap up with a few final summary questions. What is it that you like most about the product?

What I really like the most is that there are already a ton of features and functionalities built in. This means that I can use them in a very flexible way and don't have to come up with creative ways to make things work. That's definitely the best thing about it, in my opinion.

What would you say you dislike most about the product?

I'm currently working on improving our analytics reporting methods. Since we don't have a UI-driven workflow, attribution towards a success metric is always tough for us. I'm trying to figure out the most effective way to track the success of a message, but I haven't quite gotten that from them yet.

Looking back at the purchase decision, do you feel like you made the right decision for you?

I think we made the right decision back then, and it's still proving to be true as time goes on. Considering the cost, functionality, and ability for us to achieve things we wouldn't have to build ourselves, it made a lot of sense for us to go with them. Buying gave us a lot of capabilities out of the box so we didn't have to spend time and resources building from scratch.

Do you anticipate continuing to use the product for the next 18-24 months?

Yes, I do. Once we reach the stage where scaling is our priority, we'll probably start pushing the boundaries in various ways. We definitely expect to continue to use it as we grow for now. We're also trying to build out more reporting hooks into our own product for it, so we're hoping they natively support some of those things going forward.

You mentioned there might be a few areas you might push up against its boundaries. Can you elaborate?

In the healthcare industry, there's still a big reliance on fax messages and we're getting pushed to support faxes on our end. So it would be great if they could support sending faxes through a webhook. So we're basically looking for a third-party tool that can integrate with a fax sender or fax API sender, which we can then incorporate into our system.

Is there anything that you know now about Customer.io that you wish you had known when you were making the buying decision?

Regarding messaging channels setup, it would have been helpful to know if we could go beyond their default options. There's always a balance between relying on them to provide most of the features and having the flexibility to go beyond that. They do offer some flexibility, but I haven't explored it yet. So having more visibility into that would be useful. But overall, we have used them extensively, and it has been good for us so far.

Do you have any advice for someone who's going through the process of selecting this type of product?

I would suggest making use of free trials. See if you can come up with some specific use cases that you want to accomplish, and then figure out which vendors provide you with the most support or flexibility to achieve those goals without having to customize too many things.