

JOEL SHAPIRO

Creating a Business Rule for Remote Maintenance

SnackAttack is a restaurant delivery service that allows users to order food delivery from their mobile devices. Competing with companies like DoorDash, UberEats, Zomato, and GrubHub, SnackAttack's premise is simple – users open the SnackAttack app, order food from their favorite restaurant, and wait for delivery. When the system works, it is simple and seamless for the customer.

However, operations can be much more complicated for the restaurant. For instance, when a customer places an order, that information gets sent to a tablet device that SnackAttack provides to the restaurant. The information from a customer's order appears on the tablet, which provides a visual and audio notification that an order has come in. Then, the order must be manually re-entered by a restaurant employee into the restaurant's own point-of-sale (POS) system. Clearly, SnackAttack wants to ensure that the tablet works well. If it doesn't, then the order may never be seen and thus never enter into the restaurant's POS ordering queue.



SnackAttack executives know full well that not every restaurant loves mobile food delivery services. In theory, these services increase a restaurant's volume, but the cost of the delivery service is significant, yielding a much lower profit margin for the restaurant, or a significantly larger cost to the consumer, both of which can be problematic.

SnackAttack therefore tries to keep its costs of service, including the capital outlay for tablets, as low as possible. SnackAttack's tablets are relatively inexpensive, and have a tendency to need rebooting and even repair occasionally. The IT team at SnackAttack is well aware of this problem, but believes that quick diagnosis and repair of broken tablets

is more cost-effective than investing in higher-quality tablets (which also tend to stop working, just not as frequently as the cheaper ones).

Recently, the Director of Analytics at SnackAttack suggested that, rather than wait for a tablet to stop working, the client-service team could benefit from a simple diagnostic tool added to each tablet. The tool could feed a “live” predictive model that would provide real-time likelihood of tablet failure. Then, if the tablet was likely to stop working, SnackAttack’s IT team could proactively engage in remote maintenance, rebooting and ensuring the tablet is working well before it malfunctions.

You have been asked by SnackAttack’s VP of Client Services to work with the business team to generate a precise criterion (or, business rule) that will tell the IT team when to do remote rebooting and maintenance. Here, assume that you are able to build an accurate predictive model that tells you the likelihood (between 0-100%) that any given tablet will fail within the next 12 hour period.

Consider the following issues / questions when thinking about how to build a business rule:

1. A business rule should clearly and unambiguously tell the IT team when to engage in remote rebooting and maintenance.
2. What information should go into defining a rule? That is, what are the elements / variables in your business rule that need to be fleshed out and/or estimated?
3. How might you gather the information / variables you laid out in #2, above.

In case you are struggling with how to approach this task, perhaps begin by asking yourself about a specific example, such as “should the IT team conduct remote repair for a tablet that is (for instance) 31% likely to fail in the next 12 hours?” The answer, of course, is “it depends.” Your goal is to map out the precise nature of that dependency.