EXHIBIT KK

From:	Jesse Munitz-Alessio
Sent:	Thursday, October 20, 2011 11:57 PM
To:	discussions+128798383891472@xmail.facebook.com
Subject:	Re: Does Lombard need to go on a diet?

Jesse Munitz-Alessio - at 4:57pm
Phebe you guys are totally right to question the current structure - I don't want to hinder progress, just want to share insights.

A couple things - you mention wanting to break out issues into their own queues to improve accuracy - this is how lombard was originally set up, but I got so much pushback from people saying there were too many queues, that I decided to consolidate issues that we would be manually processing anyways into more generalized 'review' queues (while automated volume is bucketed into issue-specific queues). I think gauging accuracy by queue/CR usage will not be super valuable for this reason. Let me know if you want to discuss this in more detail, because it's important to differentiate different kinds of accuracy (eg. friendly fraud in the purchase problem queue is inaccurate, while dup charges/accidental/wrong amount are all intended to be in the purchase problem SAM queue).

Second - under 13 case - this is a sensitive flow because it potentially leading to a user being disabled, and because we are legally liable to review accounts that might be under 13. I think the case you describe is probably a rare edge case, but definitely worth looking at the data.

Last - making user experience more concise: My advice is to look into making the 'facts' way more dynamic in Lombard forms. Facts allow us to know things about users as they enter the CF, meaning we can show them customized flows depending on recent activity on their account. If we can surface the right question to ask a user who's experienced a particular problem, then we don't have to go through the trouble of asking them millions of clarifying questions (just need to have them confirm our guesses in the form). If we can get facts to a really advanced level, we can cut out a lot of the initial questions that our form uses to zero in on a particular purchase problem. There is always a risk in relying on user input to make decisions, so I think there is a certain value in repeating certain questions to reduce 'click-through syndrome', however if facts get to a super-advanced level, we could theoretically protect against users mis-classifying their issues.

Title: Does Lombard need to go on a diet?
Tags: payops, lombard
Sent To: payops@fb.com
by Michael Morrison
Likes: 2

At latest count our friend Lombard consisted of 22 individual contact forms and over 100 TPS ticket handlers that all feed into 130 seperate TPS queues. Yes, you read that correctly... 22 forms, 100 handlers and 130 queues. Lets say you see an area for improvement and you want to make a tweak to the form in order to gain more data from a user, or route them in a different way. Here is what you're up against:

1. Figure out which of the 22 forms your change needs to be made in

2. (hours later...) Make change
3. Realize this change just broke 14 other forms which had some sort of dependency
4. Realize that translations are now broken for said form

From the user's perspective, I'm sure there are more than a few people who get frustrated with the amount of questions (sometime duplicate) that we ask them throughout the form(s), just to let us know they ran into a problem. I was stuck in an infinite-loop of questions just today when testing Lombard (see video here: http://www.fburl.com/?key=2162134).

It feels like the form is this frankenstein beast that we've bolted together over the last 6 months or so. Maybe it's time to take a step back and ask... **is there an easier/more efficient way we can classify a user's problem?**

I propose that we re-think this beast from the ground up. I welcome your thoughts :-)

You can reply with a comment, or post it here:
https://www.intern.facebook.com/intern/discussions/?fbid=128798383891472
Reply with '!unsubscribe' to stop receiving replies to this discussion.

Full History

Paul Litvak - at 8:00am yesterday
This is interesting. Subscribe me to your informational pamphlet.

Phebe Huang - at 8:15am yesterday
For the past week, Brad and I have been working on mapping the Lombard flow on a very granular level to understand how our users are being directed and to gain a better understanding of how the form works. The problem exists in repetitive questions, similar forms that lead the user to answer the same questions but in actuality redirects them to a different queue, and unnecessary lengthiness. For example, a user that uses a credit card is lead to a series of questions that are relatively straightforward and is able to submit their ticket on one form, whereas a user that uses PayPal with the same issue is lead to an entirely different form. In the end, these two users are lead to the same TPS queue and are not treated differently. This makes us wonder - is it necessary to have two separate forms and several more handlers that do essentially the same thing?

Speaking on behalf of a team that regularly works these queues and hope to gradually automate some of our responses, it is painstakingly difficult to trace a problem we identify to the many handlers it comes from, to the multiple forms those handlers are attached to. For example, a CR that we commonly use is "Using Existing Facebook Credits". This simple CR, that essentially educates the user on how to use their Facebook Credits balance, takes up more than 5% of our volume. Hoping that we could automate this CR, Brad and I were curious to see which handler and form this CR was tied to. The results? Over 50 handlers and 19 different forms. The amount of time it would take to automate this one CR prompts us to wonder - Why are users with the same problem coming in from 19 different forms?

Speaking on behalf of a typical user, the form is repetitive, overly complex, and long. These are common complaints that we receive when reviewing our tickets. This makes us think - how many users give up while