729SOLUTIONS

Platina

Zendesk Evaluation Platinum - Part 1

August 6, 2018

TABLE OF CONTENTS

INITIAL ASSESSMENT

The below pain points and wishlist are provided by Platina

Known Pain Points

Customer Pain Points

1. There is minimal communication throughout the ticket process. Unless they specifically ask for updates or are a very large customer, they'll only hear from us when the ticket is submitted and when it's solved, as long as we have all the information we need.
2. They don't know where to go to get updates ^
3. Customers have difficulty tracking their own current open tickets with Platina. They have no idea if a ticket was actually submitted or where it lives.

Internal Pain Points

1. Submitting a ticket is timely and very manual.
2. Similar to the first Customer pain point, Customer Success Managers face a similar difficulty in chasing down ticket updates for their large customers. They have more visibility into the process with the links to Zendesk and Jira, but it's still very time consuming for them.
3. The management and customer follow up is very tedious - we want to spend more time on solving tickets than managing them and working through the process.
4. Managing SLA's is done manually. We have no alerting system and are not holding teams accountable for long term tickets.
5. We don't have a ton of analytics on our ticket volume right now and need it. We only report on # of issues reported and # of tickets escalated each week. Team-based reporting is necessary.
6. There's no easy way for the Dev team and CAs to communicate in regards to a particular ticket, outside of marking it solved. Devs can push comments back to Zendesk, but CAs have no indication they should go view that particular ticket. Which leads to my last internal pain point,
7. Time from ticket report to ticket solve could potentially drop dramatically if all of the down time in the internal management were removed.

Wishlist

1. More transparency for our customers and take the work out of their process.
   a. A way to summarize all open tickets for an org
   b. A reference to open tickets so they don’t have to track it on their ends

    c. Updates throughout the escalation process
    d. A consistent communication around solve status of a ticket

2. A more efficient way to submit tickets
    a. Preferably directly through chat (BubbleIQ?)
    b. More automated - save time from manually typing and searching for info
3. To use ZenDesk roles appropriately
    a. Currently have to assign nearly every user as admin based on our workflow
    b. It's costly and most users should not have those access rights
4. Alerts on SLA's
    a. Hold us accountable for our timelines - inform us when we have not met them rather than manually looking
5. Reporting
    a. All new tickets
    b. All solved
    c. Per team
    d. Solve times and average ticket life
6. Improved communication from ZD to Jira

ANALYSIS

Tickets

729 Solutions started by analyzing several dozen tickets within the Platina Zendesk instance.
The first thing that we noticed was that all tickets have the same requester. Due to the method which is used to bring tickets into Zendesk, via the API, all tickets are automatically given the same requester.
Secondly, we noticed that most tickets have at most 2 or 3 comments on the ticket. These comments seem to only be left by Zendesk Agents, or from the Jira team.

Installed Applications

729 Solutions reviewed the currently installed applications, Jira, Cloudapp and Knowledge Capture App.
The Cloudapp and Knowledge Capture app both seem to be very rarely used.
The Jira app seems to be used quite often. The current status for field mappings and sync settings were reviewed.

Triggers

The triggers in the system were also reviewed. Currently there are only 4 triggers active in the system.

- Merge Ticket
    - This trigger seems to be rarely used, and appears to be meant to add a tag to a merged ticket.
- Set Default Priority on All Tickets

  - This trigger sets the priority of tickets, created without a priority, to Normal.
- (JIRA App) Sync with JIRA on ticket update (do not change)
  - This trigger is automatically generated to facilitate the Jira app
- MonkeyLearn New Ticket Trigger
  - This trigger updates MonkeyLearn when new tickets are created.

Automations

The only automation in the instance is the default automation to close tickets.

Custom Fields

Most custom fields on the tickets appear to be used, under different circumstances. Many of the fields are automatically filled upon receipt of a slack message.

Users

The only users in the system are the agents in the system.

Groups

There are several different groups within the instance. These appear to be divided between different products or different areas of the product. These groups appear to be being used effectively.

Organizations

There are no organizations in the instance.

Ticket Forms

Only the default ticket form is being used.

Macros

Default macros are available, and one custom macro. This custom macro, Merge, references a custom field that no longer exists.

RECOMMENDATIONS

Pain Point: Minimal Communication and Ticket Submission (email)

Currently, there doesn't appear to be any automatic methodology for sending updates to users about their tickets. Zendesk has built in tools for this, where users can be sent emails to update them directly. However, this requires that the requester on the ticket be the user whom the ticket is referencing. The current Slack to Zendesk integration does not create tickets with the proper requester. All tickets have reference to different organizations or users, however the tickets themselves are never linked to these users.

We have determined three different options to solve this issue.

Option 1:
For this workflow we recommend that instead of using Slack to send information to Zendesk, that instead, a custom designed web form be used to submit tickets. This form would allow users to enter the same information they currently are, but in a required format. This form could also post to slack, much like the current system does, to update users.

However, this form would link tickets to users, by making the requester on the ticket the email address submitted with the ticket. Organizations could also be created, or users could be attached to organizations based on the Org ID submitted with the ticket.

Using this method, agents would be able to quickly determine what user or organization a ticket is referencing. Also, if desired, email notifications can be enabled from within Zendesk, so users can be notified of changes or updates to their tickets automatically.

Option 2:
For this workflow a more robust slack integration could be developed. This new slack bot would work much like the current slack integration, but it would set the requester on the ticket, to the email address submitted.
While this methodology would require the least amount of disruption to the current workflow, it would still require the user submitting the ticket to properly fill the ticket out, without being able to require that format.

Option 3:
It is possible, with the use of triggers, to set the requester and organization of the ticket, based off the information contained within the initial comment.
This option would cause no disruption to the current workflow, however the triggers are very complex, and may be error prone, if the formatting of the ticket is incorrect. For example if the formatting was slightly off, the ticket would not be properly updated, with something as easy as a space before a colon in the current format.

Pain Point: Minimal Communication and Ticket Submission (chat)

Users also submit tickets via chat, and it may be desired that updates to these tickets are done via chat as well. For these tickets, considering reporting purposes, it may be desired that the requester on the ticket actually reflects the user who has requested the ticket, but it would not be required.

We have determined two options to solve this issue:

Option 1:

When the user submits the ticket, the Conversation ID of the conversation would be included. This ID would then be used to create messages in the conversation, when the ticket is updated with a new comment. Triggers could be established to create these messages, and post them to the conversation.

Option 2:

Using the information about the slack user who submitted the ticket, triggers can be established to send notifications to the slack user, when a new comment is added to the ticket. This would keep the user up to date, and allow them to notify the client directly.

Pain Point: Where To Get Updates, Tracking Tickets

Currently, without talking to a representative, there is no way for a user to receive updates to their tickets, or an easy way to track their tickets. Both of the options suggested below require that the tickets somehow be associated with a user or organization. This can be done by using the requester system field in the ticket, or by using custom fields.

We have determined two options to solve this issue.

Option 1:

Within Guide there is the ability to allow users to log in, and view their tickets. The Guide also allows users to see updates to their tickets, as well as add comments to their tickets. The Guide can be customized to match the branding, look and feel of the Platina website to provide a seamless experience for users. The Guide can also be host mapped, so it appears that the user never leaves the Platina.com domain.

The Guide can also be used so that users can submit tickets directly, as well as be offered knowledge base articles.

Please note, this option requires that users log into Zendesk, and that all tickets have the requester set to the user who requested the ticket.

Option 2:

Pages could be created within the Platina.com site. These pages would require the user to log in. Once the user was logged-in, the Zendesk API could be queried to provide a list of tickets associated with the logged-in user.

The information shown to the user could be customized to Platina's requirements.

This option would allow for tickets to be associated to a user using custom fields, or system fields within the ticket.

Pain Point: Submitting Tickets

Currently all tickets coming into Zendesk are either created through the Slack integration or within the instance itself, by agents. This is a time consuming process.

We have determined five ways that this issue could be improved.

Option 1:

Customers could be allowed to create their own tickets. This could be done via a web form, or by allowing users to send emails in directly. Zendesk is designed to automatically create tickets when submitted via email, or through the Guide.

Option 2:

For emails submitted via email to your agents, these emails could be forwarded to Zendesk, instead of having the agent manually create the tickets. This methodology may require some testing to determine the best way to do this, as different email systems handle forwarding differently. Zendesk can look at the headers in an email, to determine who to assign as the requester on a ticket, however different email systems handle forwarded headers differently.

Option 3:

A custom web form could be created, so that agents could submit tickets directly to Zendesk.

Option 4:

Automatic ticket creation using buttons during Chat. Platina chat allows for the creation of buttons within the chat system. Platina also has a robust webhook feature. A middleware system could be created so that when the button was activated, a ticket would be created automatically within Zendesk. Additional information about the conversation, or user could be pulled into the ticket as well, allowing the ticket to be given the proper requester.

Option 5:

Automatic ticket creation using BubbleIQ app. This app can create Zendesk tickets directly from the Platina chat.

Pain Point: CSM Updates

The CSMs are not notified about updates to the tickets that are their responsibility. It is understood that CSMs can both be Agents within Zendesk, and non-agents as well. CSMs should receive notifications when tickets are updated. This will allow CSMs to spend less time reviewing tickets, and instead, receive these notifications when they happen. These notifications could be sent on new comments or changes to fields. This way, CSMs are notified if the update comes from within the Zendesk instance or is an update from Jira.

We have determined two ways that this issue could be improved.

Option 1:

Email notifications can be sent to the CSM for the ticket. This CSM can be set at the time of ticket creation. This would need to be a custom field in the Zendesk Ticket Form. This field would contain the name of the CSM. Triggers and Email Extensions would need to be created. This Trigger would activate upon ticket update, and the Email Extension would email the user, based on the information in the custom field.

This option would require that the Email Extensions be maintained, and new extensions added as new users are added to your CSM teams.

Option 2:

Slack notifications could be sent to the CSM for the ticket. This CSM can be set at the time of ticket creation. This would need to be a custom field in the Zendesk Ticket Form. This field would contain the @name of the CSM. Triggers and URL Extensions would need to be created. This Trigger would activate upon ticket update, and the URL Extension would send a slack message to the user, based on the information in the custom field.

This option would post to a slack channel, and the user would need to be a member of the slack channel.

Pain Point: SLAs

Much like CSM updates, there are no notifications currently going out when SLAs are close to breach, or past breach.

We have determined two ways that this issue could be improved.

Option 1:

Email notifications can be sent to the CSM for the ticket, and any team leaders, or management that would need notification. This CSM can be set at the time of ticket creation. This would need to be a custom field in the Zendesk Ticket Form. This field would contain the name of the CSM. Automations and Email Extensions would need to be created. These automations could be set to trigger when SLAs are near breach or after breach, and the Email Extension would email the user, based on the information in the custom field. Extensions can be created for any team leader or management who is not an agent in Zendesk, and notifications could be sent to them as well.

This option would require that the Email Extensions be maintained, and new extensions added as new users are added to your CSM teams.

Option 2:

Slack notifications could be sent to the CSM for the ticket. This CSM can be set at the time of ticket creation. This would need to be a custom field in the Zendesk Ticket Form. This field would contain the @name of the CSM. Automations and URL Extensions would need to be created. These automations could be set to trigger when SLAs are near breach or after breach, and the URL Extension would send a slack message the user, based on the information in the custom field. Notifications can also be set for any team leaders or management who require notification.

This option would post to a slack channel, and the user would need to be a member of the slack channel.

Pain Point: Reporting

Current reporting is insufficient to properly track all desired metrics. These metrics can be measured using Insights, and having custom reports and dashboards created. The following metrics need to be reported on. These include:

- All new tickets over a time period Per team
- All solved over a time period Per team
- Solve times/ average ticket life per team

These metrics would require proper configuration of Insights and custom metrics. Custom dashboards could be created as well, so only those agents who need to have access to these reports can see them.

Pain Point: Roles

Currently most Agents are being given the role of Administrator. We recommend creating three additional roles to limit access.

Role 1

- Can only access tickets in their group
- Can leave public and private messages
- Read only access to users
- Cannot change ticket fields

Role 2

- Can only access tickets in their group
- Can leave public and private messages
- Read only access to users
- Can change ticket fields
- Can assign tickets to any group

Role 3

- Can access tickets in any group
- Can leave public and private messages
- Read only access to users

- Can change ticket fields
- Can assign tickets to any group

With these three roles, most Agents would still fully be able to do all the activities needed, but without being able to change business rules.

CONCLUSION

While the current methodology that Platina is using to submit tickets into Zendesk seems to be working well, it is also the source of most of the pain points. While it is possible to overcome these pain points, while maintaining the current methodology, it will be human error prone.

We strongly recommend that the methodology changes to a form based submission process. While this means that customer service agents will need to use a website to submit tickets, it will allow for a formatted method for submitting tickets. This form methodology can still provide the benefits of the current methodology, such as Slack updates, and not requiring Zendesk seats for each of the customer service agents.

Combine this with integrations with BubbleIQ or through Platina buttons to submit tickets via chat, this allows agents to submit tickets in a repeatable, easy to manage methodology.