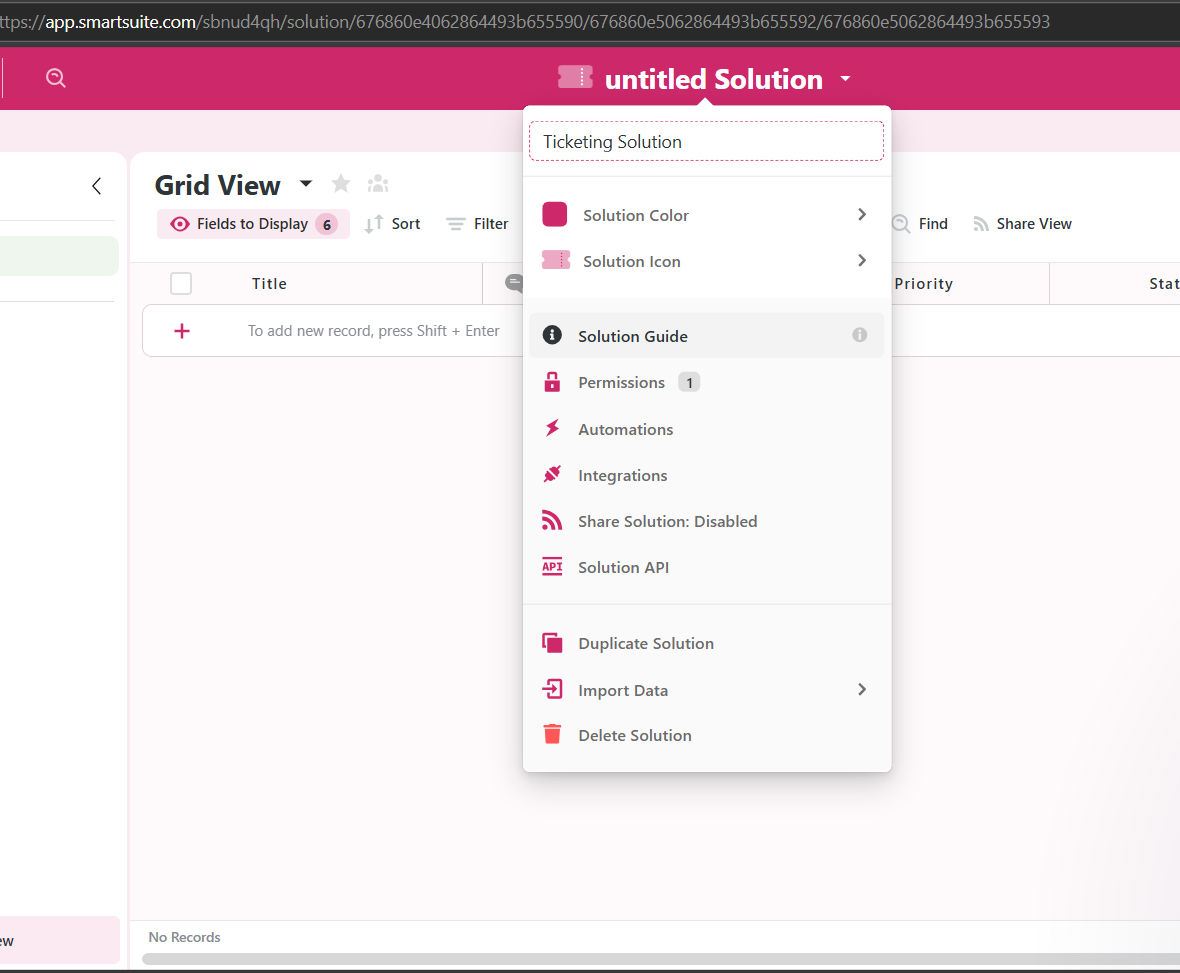
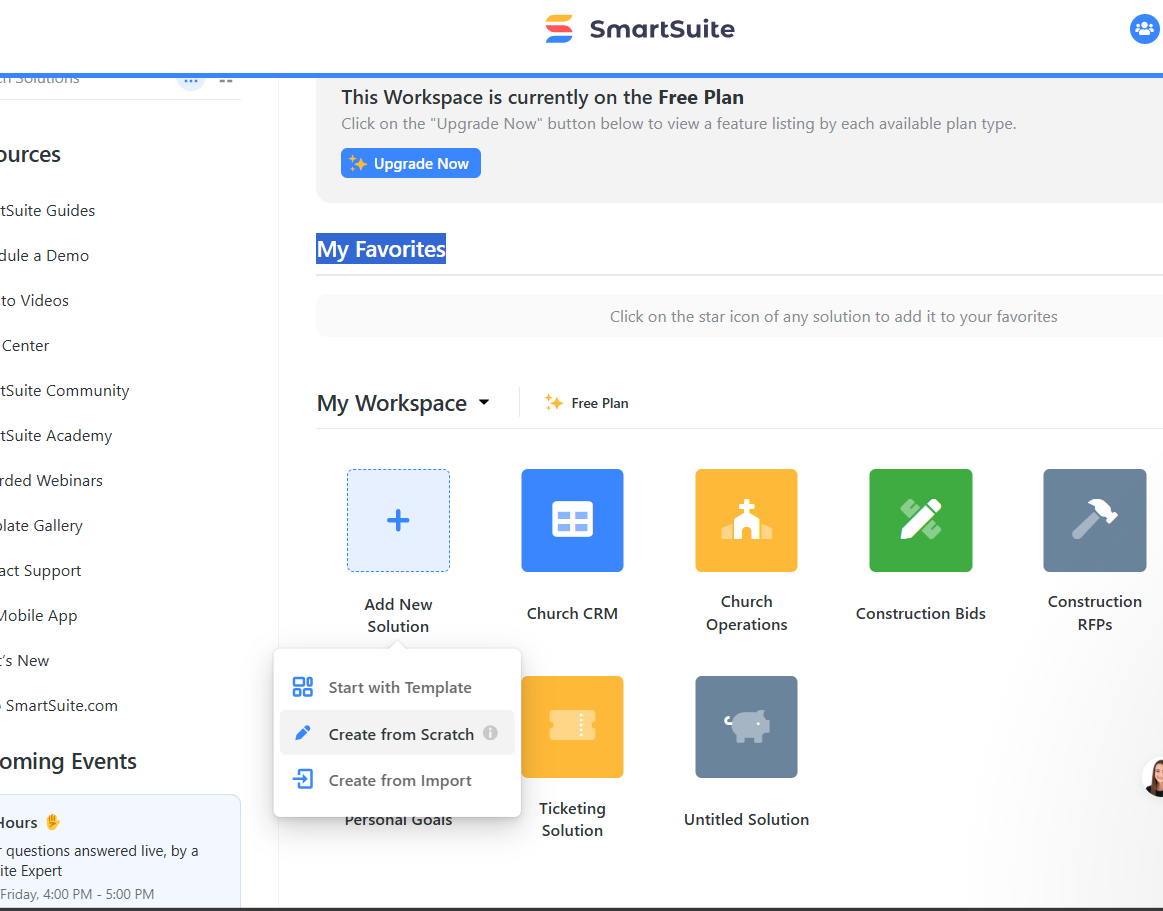
Ticketing System Tutorial

Section 1

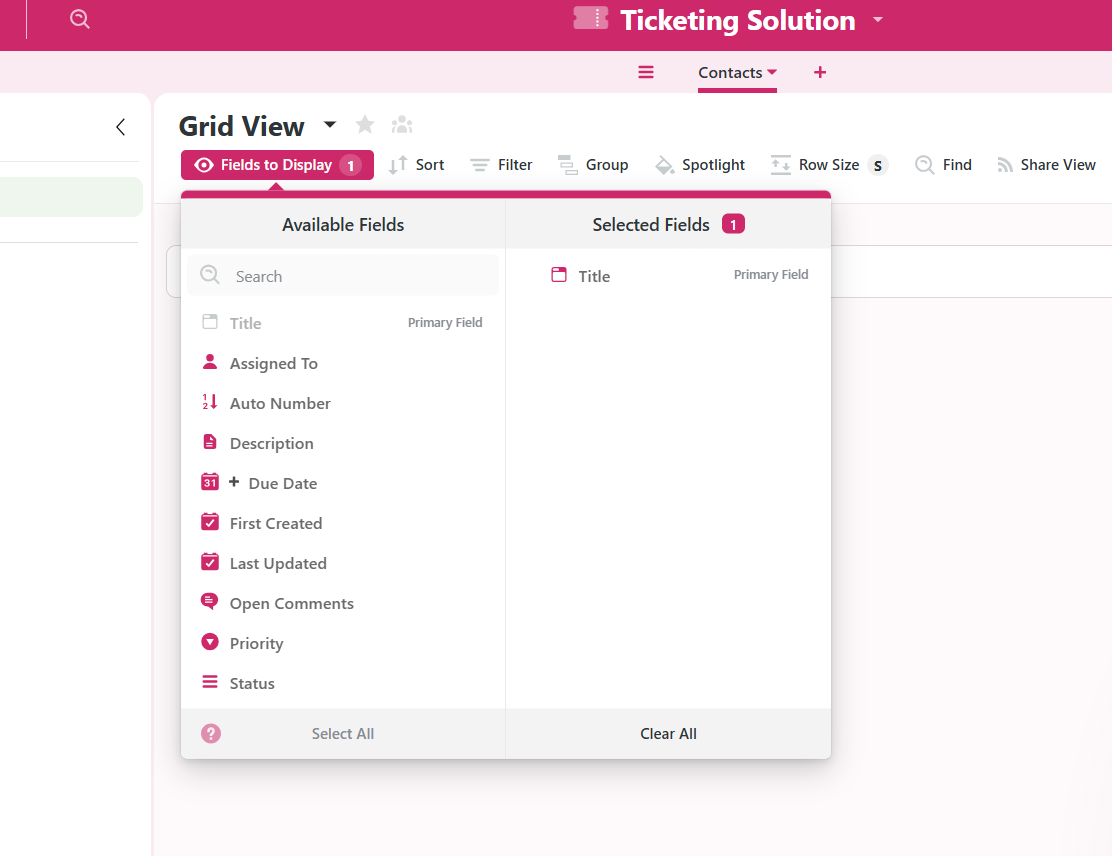
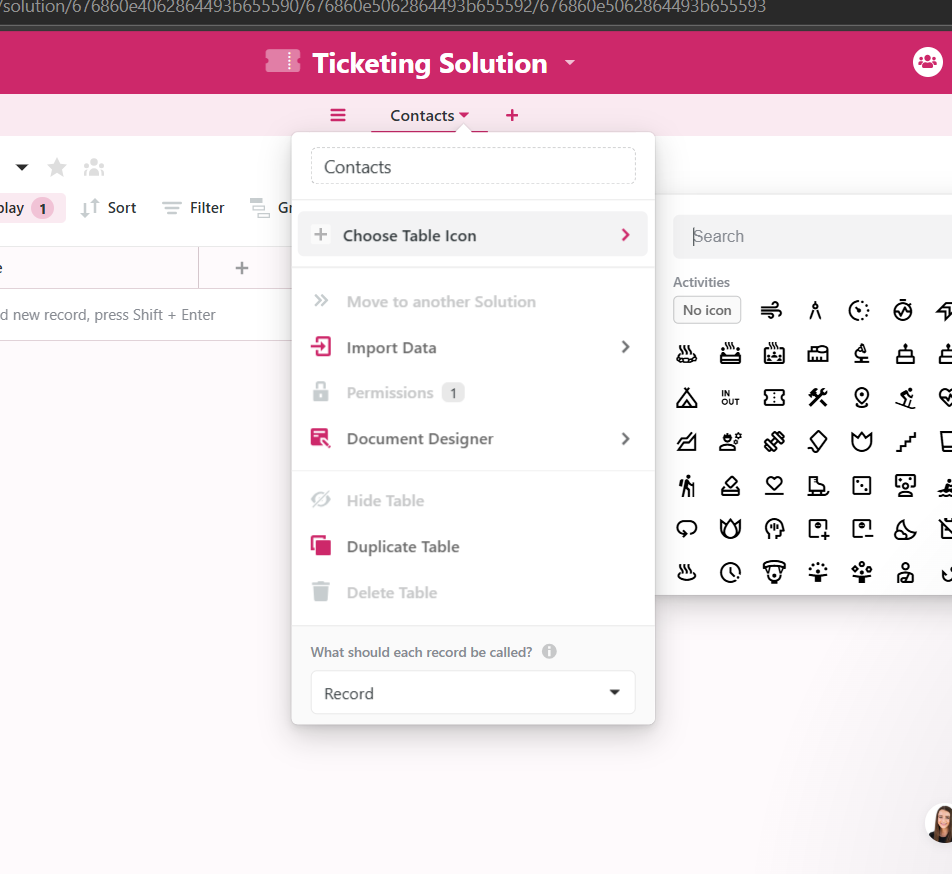
As some of you may know, ticketing systems are ideal when organizing and storing vast amounts of user data, whether it be client, employee, or both. SmartSuite offers a way to build an automated ticketing system from scratch. This will serve the purpose of better satisfying a customization need for employers. Having the ability to automate such a system can have positive effects on the network and may even increase the success of overall security.

Let us have a look!

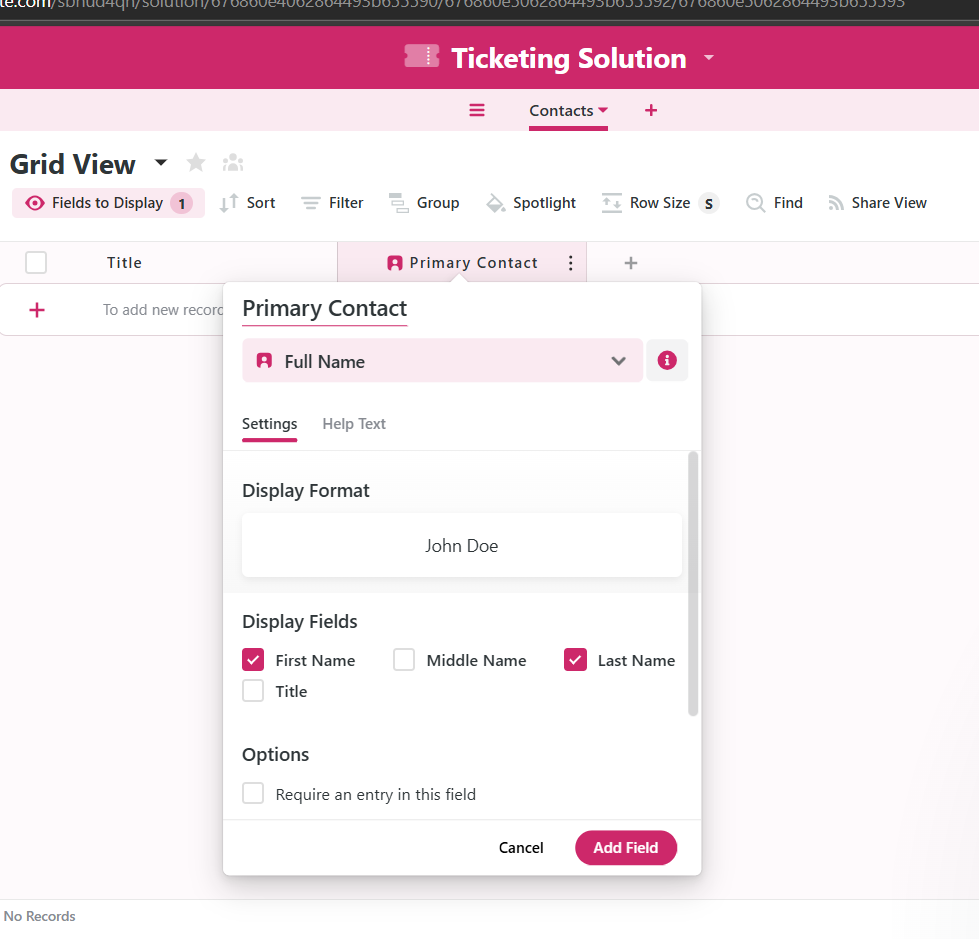
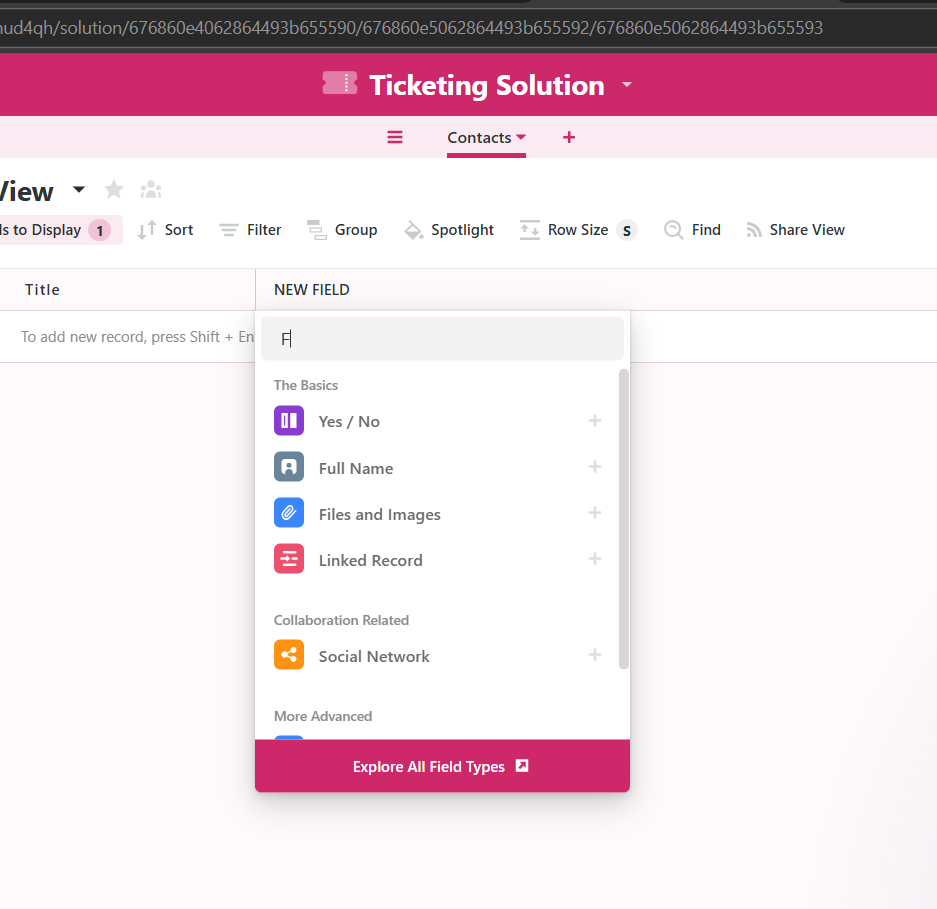
First, we will navigate over to SmartSuite homepage and from there we can “add new solution > create from scratch option.” We should rename our “untitled Solution” to something like “Ticketing Solution.” It is critical that we build out two separate apps, one will be for the people submitting the ticket and the other for the ticket itself. These are two separate data sets, which could be from an organization or an individual within an organization.



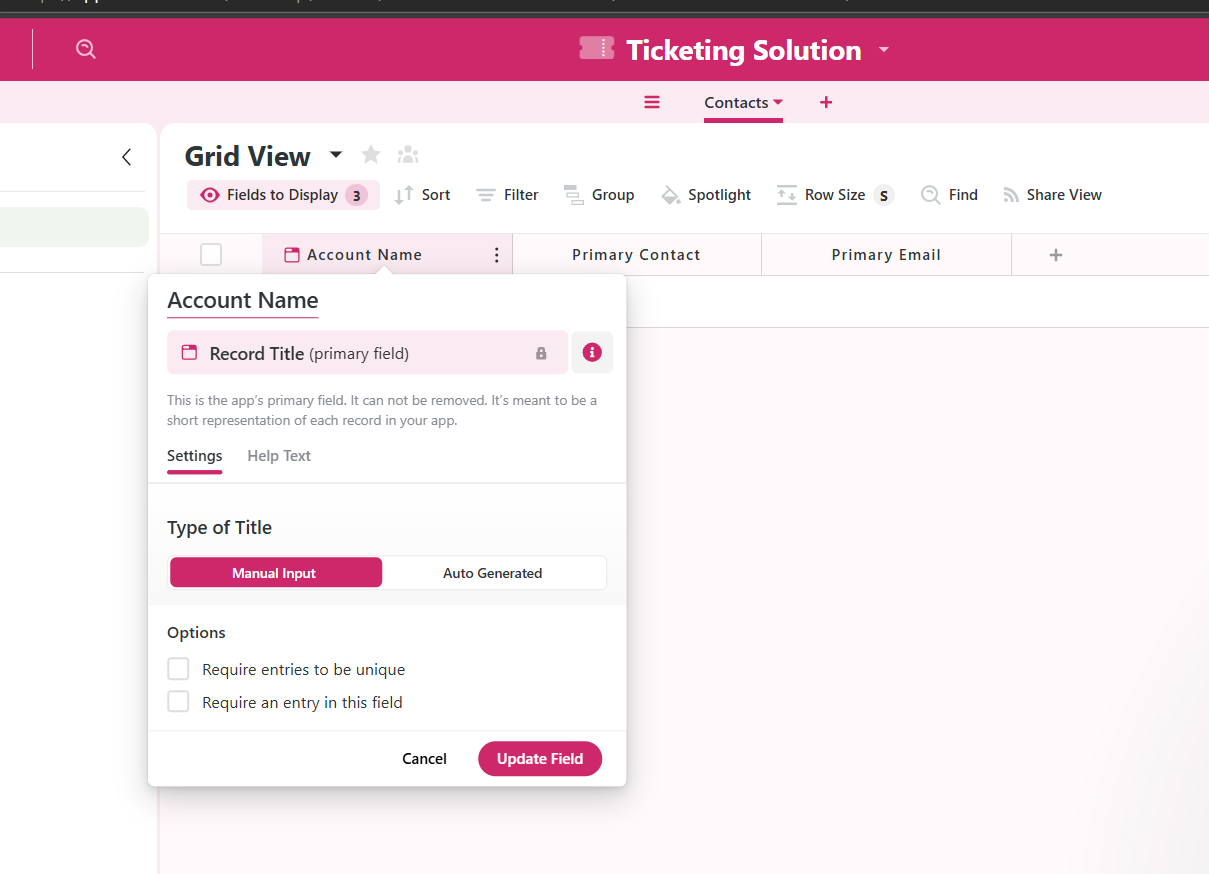
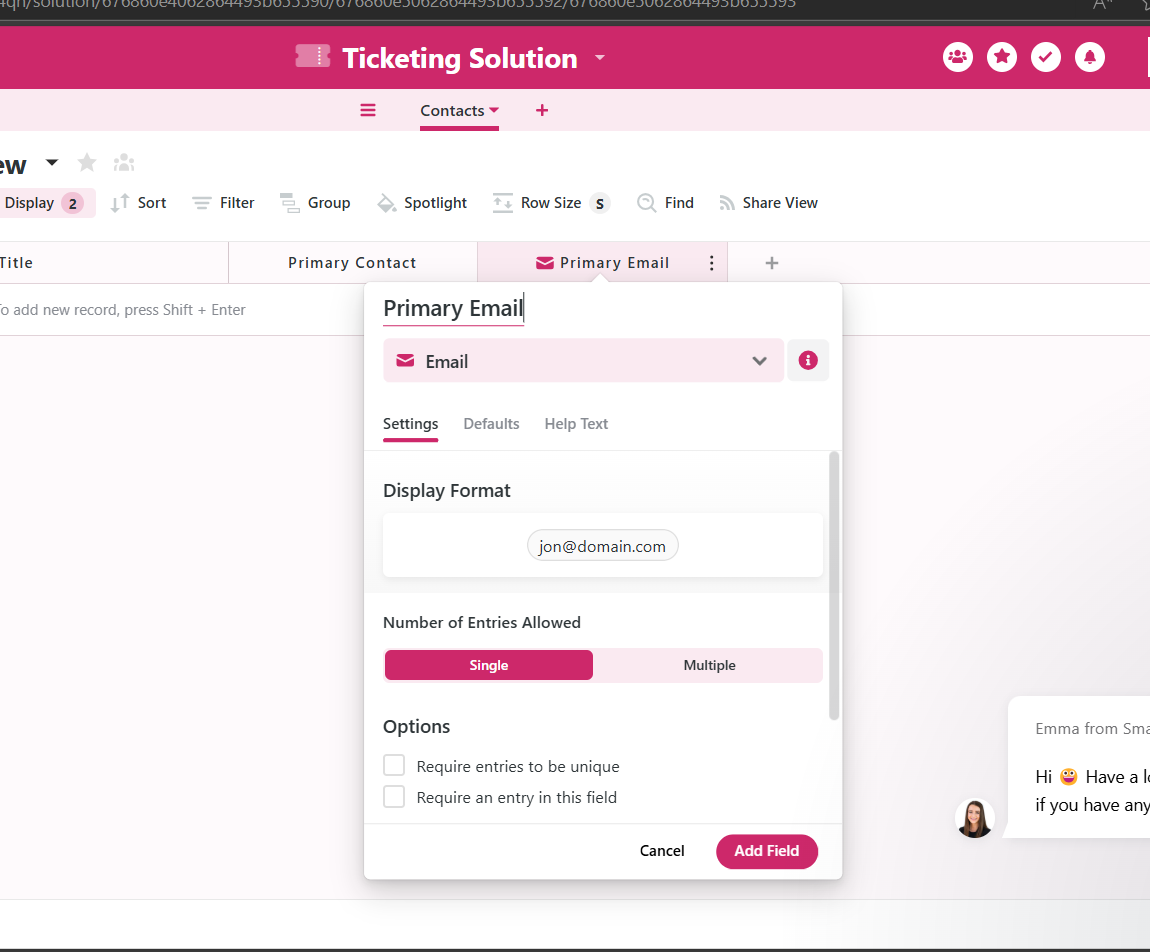
Moving on, we will need to label our “app1” app to something a little more Intune with this project. Let us select app1 and change its name to “contacts”. In this space we will store information about the entity submitting the ticket as well as trace that data throughout the solution. Go ahead and hide our “set up in the field display” > then clear all, this is not to delete them but hiding them from this view.



I will proceed by entering the information relevant to this account. So, I will call it “Primary Contact,” we will use a full name field. Let us keep “first and last name” checked and then click add field.

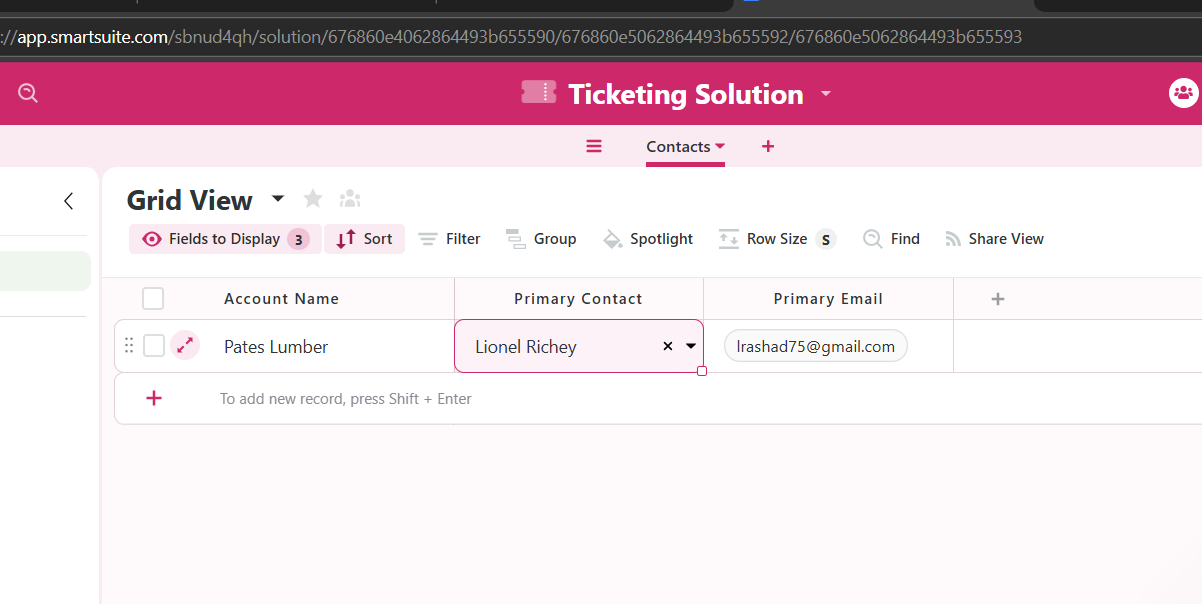


We will also create a more direct means of contact in the next field. We will go about the same setup as with the primary contact. Add field > search “email” > select email. This is where we will store the account holders’ email info. The theme of the information entered here should remain consistent with the account or contact submitting the ticket. Lastly, we will head over to the title tab and rename it to the actual account’s name or number. We then select “manage input > uncheck requirements to be unique > uncheck require an entry in this “> click update field.



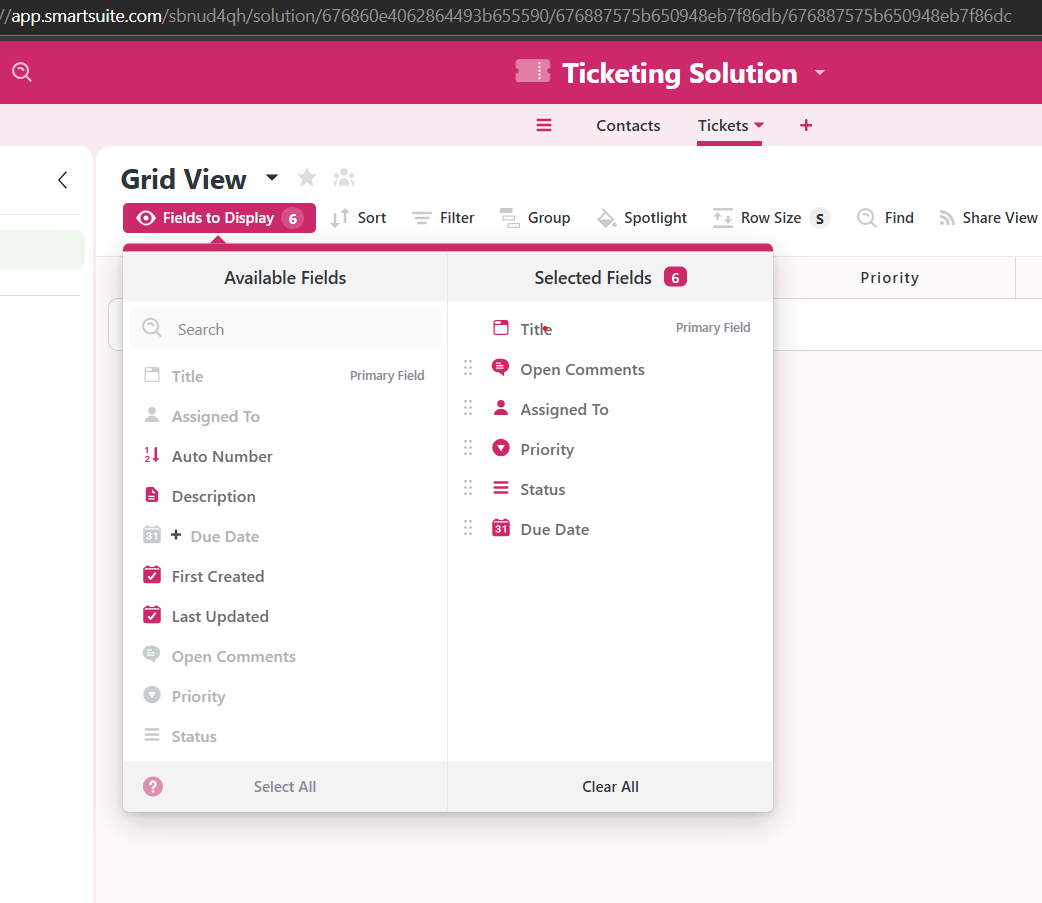
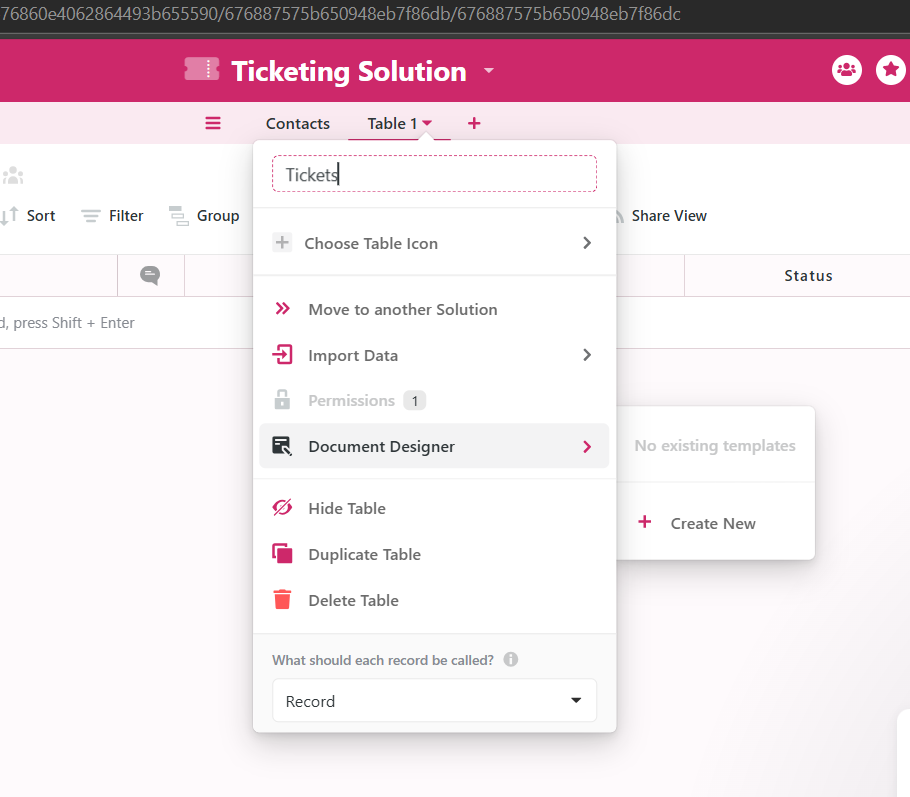
Once all that is saved, we will open our first account. Say I have an account named “**Pates Lumber**,” I, myself will serve as the primary contact. I will enter “**Lionel Richey**.” Save. To complete this section I will add my email address at [**lrashad75@gmail.com**](mailto:lrashad75@gmail.com) **(not my email lol!)**.

The core concept that I have demonstrated here is how to store contact or account information.

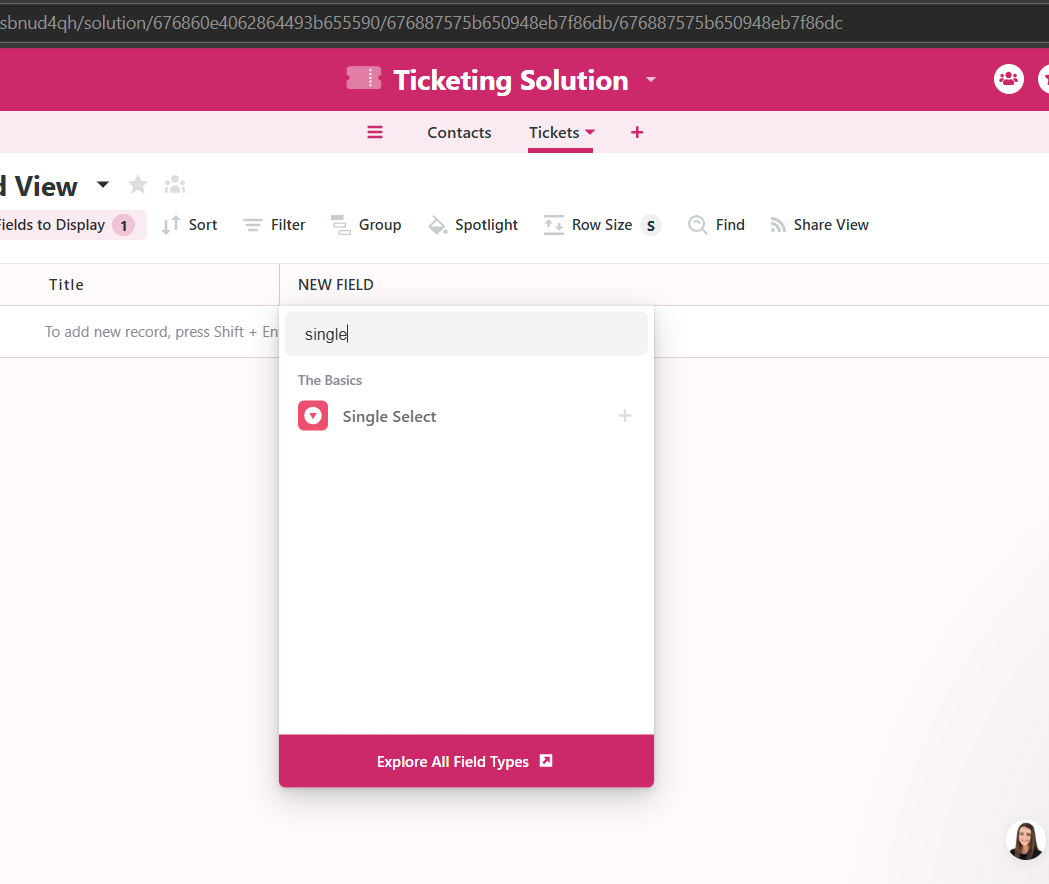
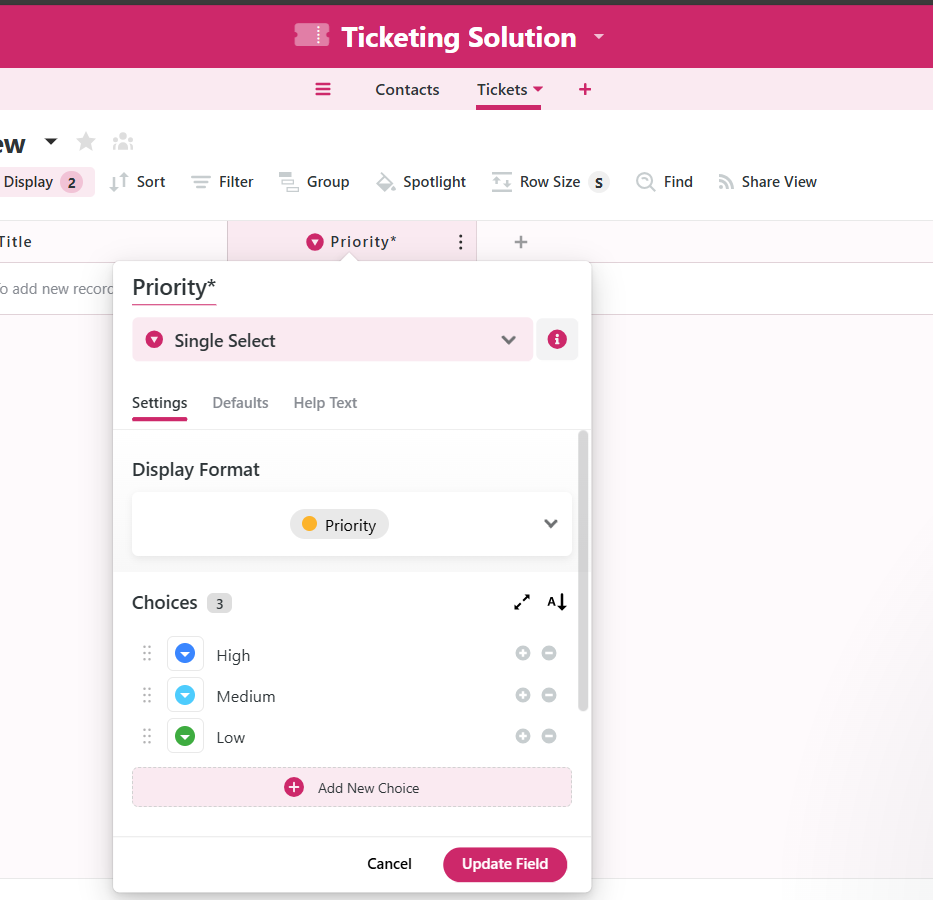


Section 2

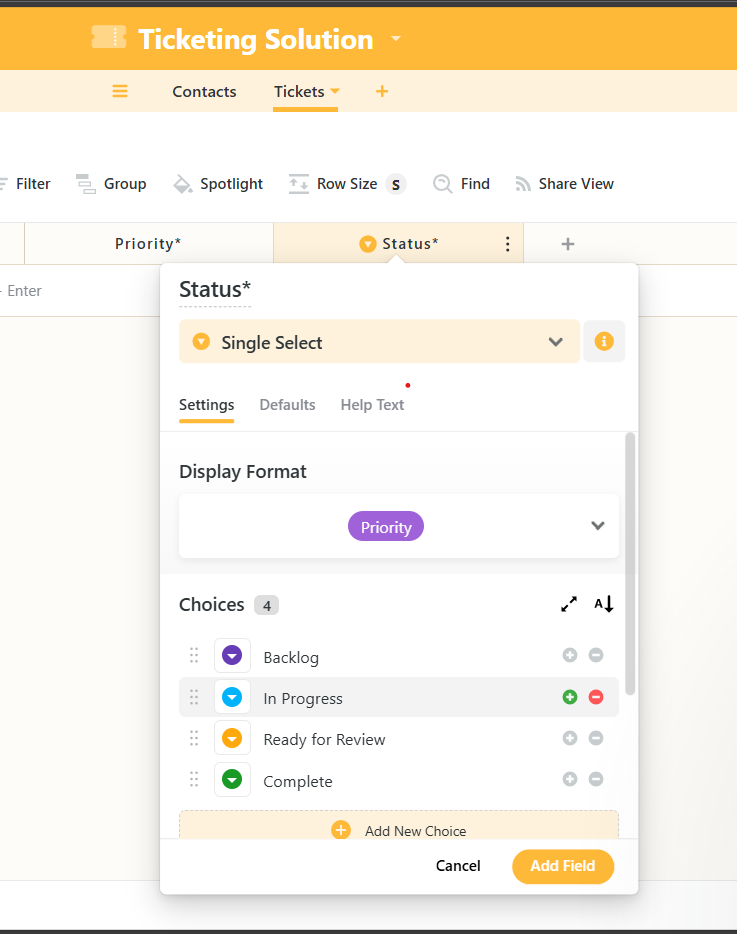
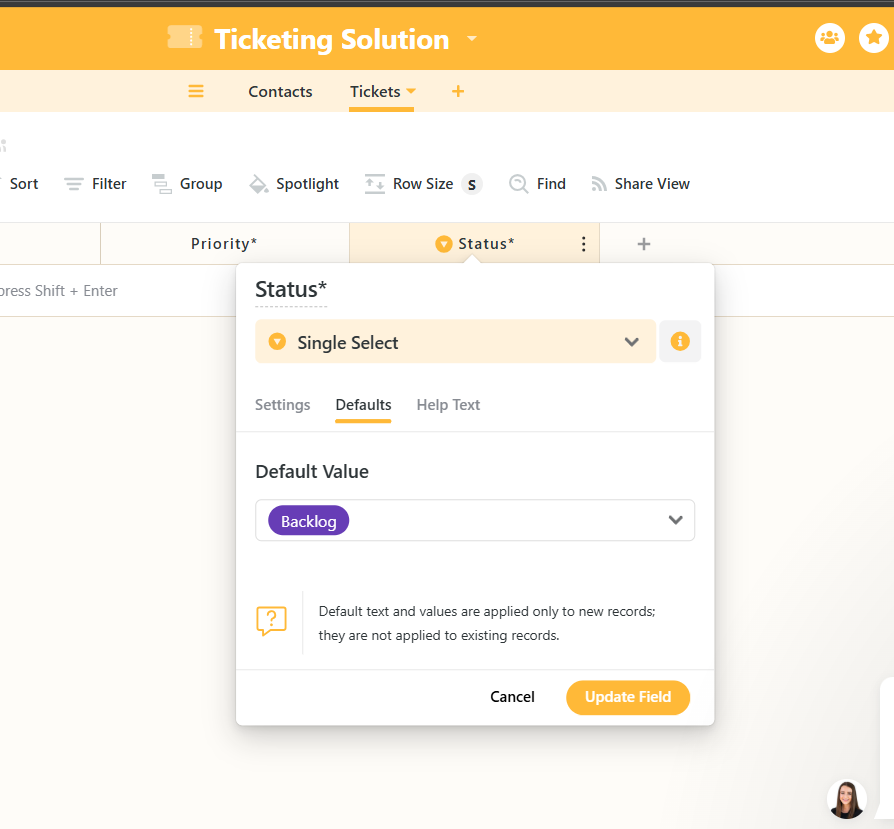
Much like setting up the first app and renaming it, we will continue by doing the same for the second app. Beside the contact app, lets click the plus sign to add our “ticket” app. By preference I like to go ahead and immediately clear the display field.



For tickets we will set up a few fields that could help systems better manage and allow quicker response times. “Add new field > single select > rename Priority\*.” For increased accuracy when addressing a ticket, we can add three variations of priority high, medium, and low.

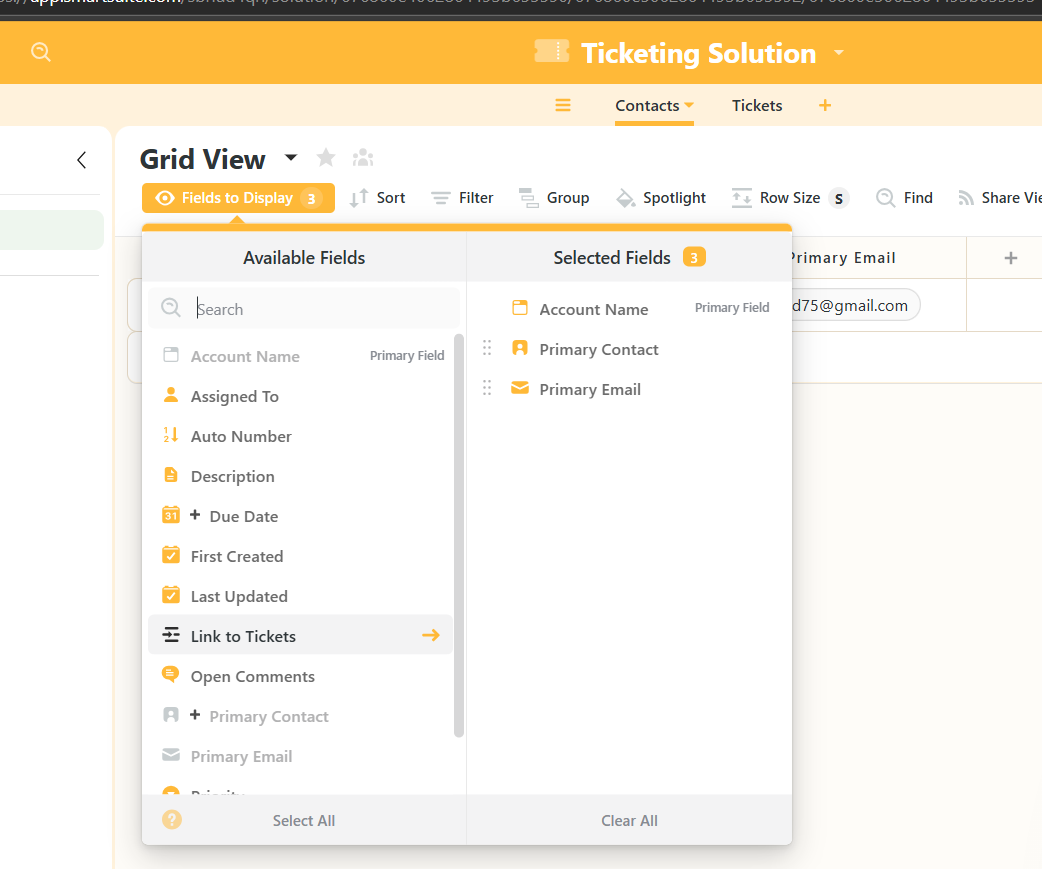
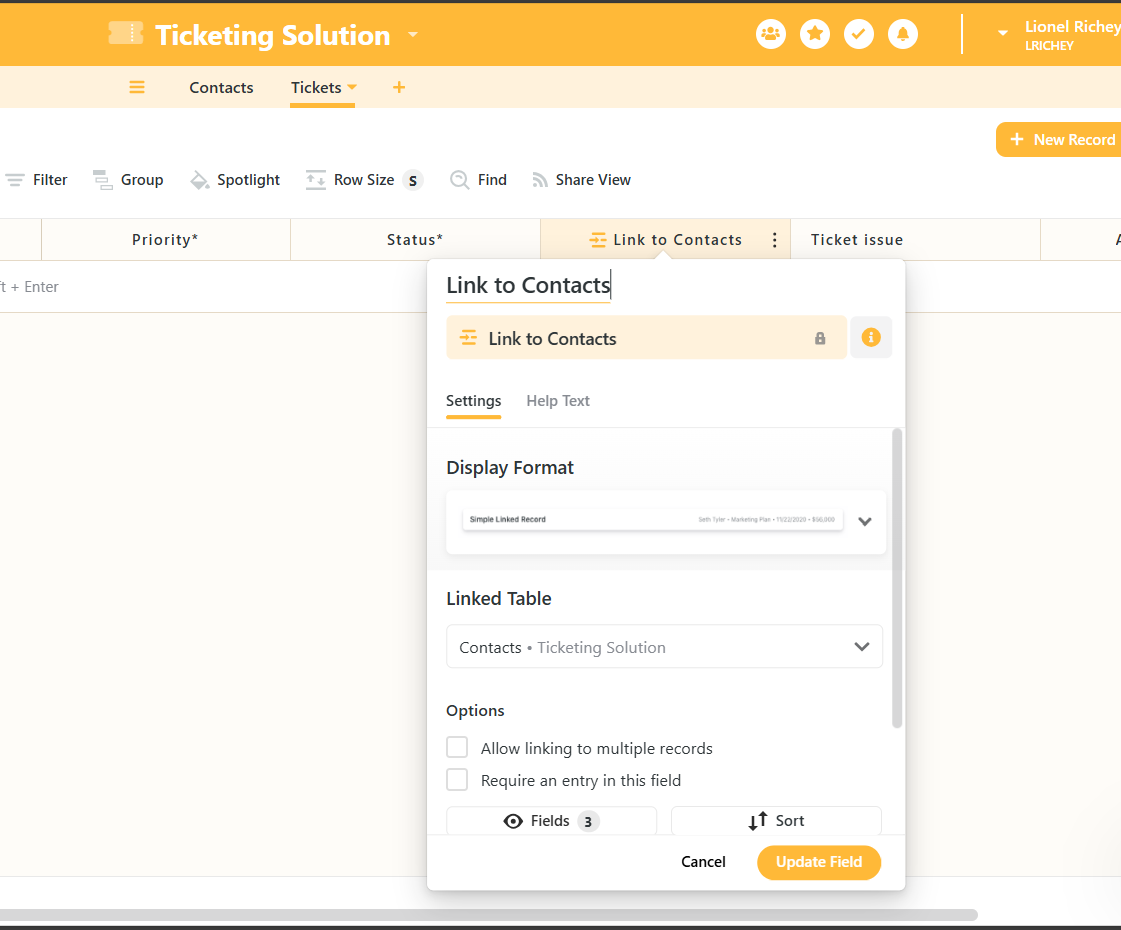
You can even change the color themes if you wish. What if we wanted to create a status that tracks a specific ticket? I’m going to add my second field to the ticket. “add field > single select > rename Status\*”, settings around mid page show several default statuses. These can be edited to say whatever you like, the clor can be changed, as well as number of statuses shown as option. Under the default tab ensure that when a ticket comes through it has a Backlog status. This notifies the team that they need to start working with it. Continue by updating and adding field.

Section 3

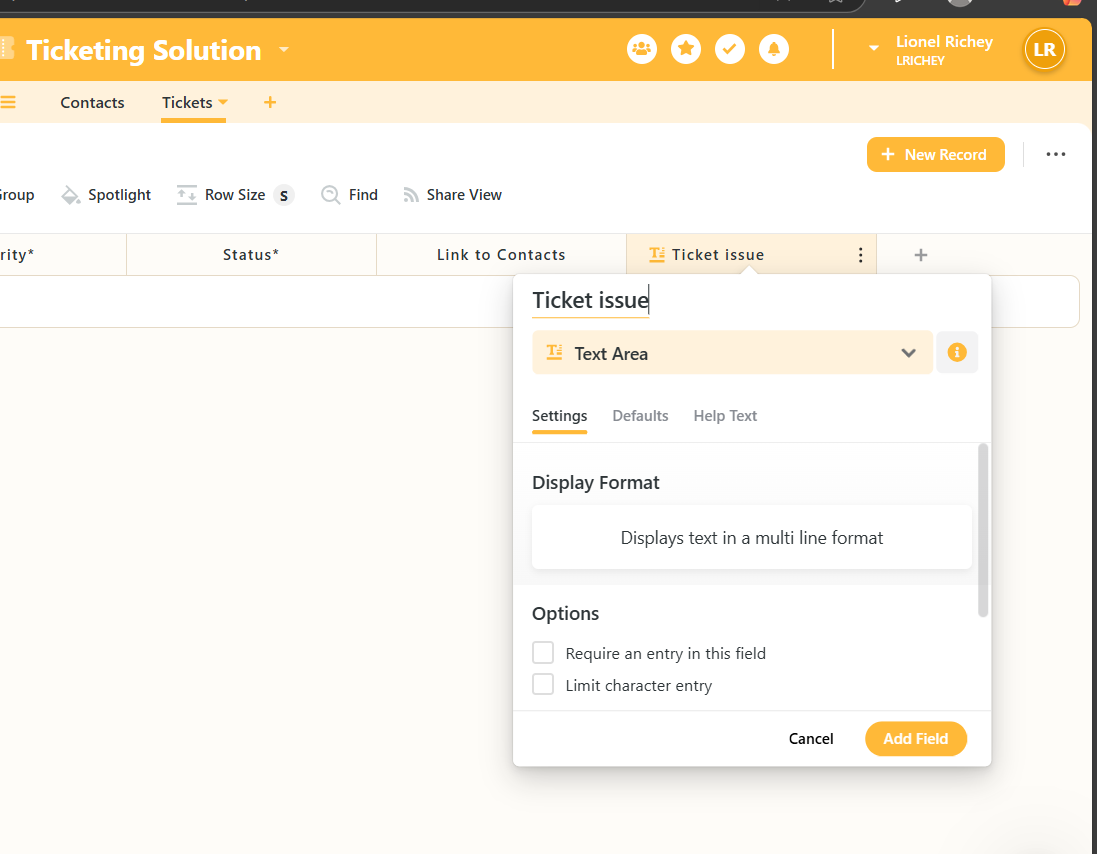
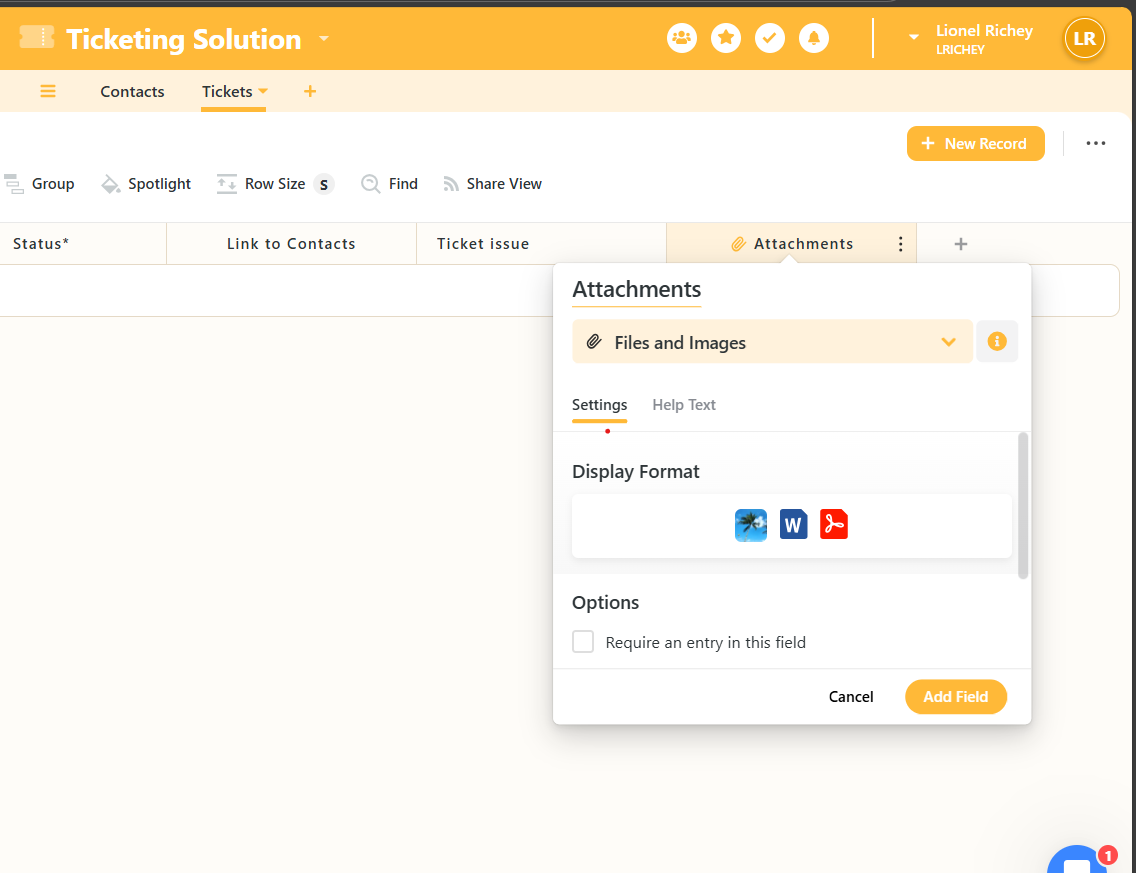
Build a linked relationship from your tickets to your conatcts. “Add a new field > Linked record option > under linked tables” verify the correct solution is chosen (Contact – Ticketing Solution)

Proceed with cancellation of allowing links to multiple records. You should want each ticket to have only one link to your contacts. Click add field. We’ve added the link to contacts, to verify go to contacts and fields to display, you’ll find a link to tickets that wasn’t there prior to us creating it. We’ve successfully built the opposite side of the link. so, we learned that if a contact links to a ticket, a corresponding ticket must also share a link with that contact.

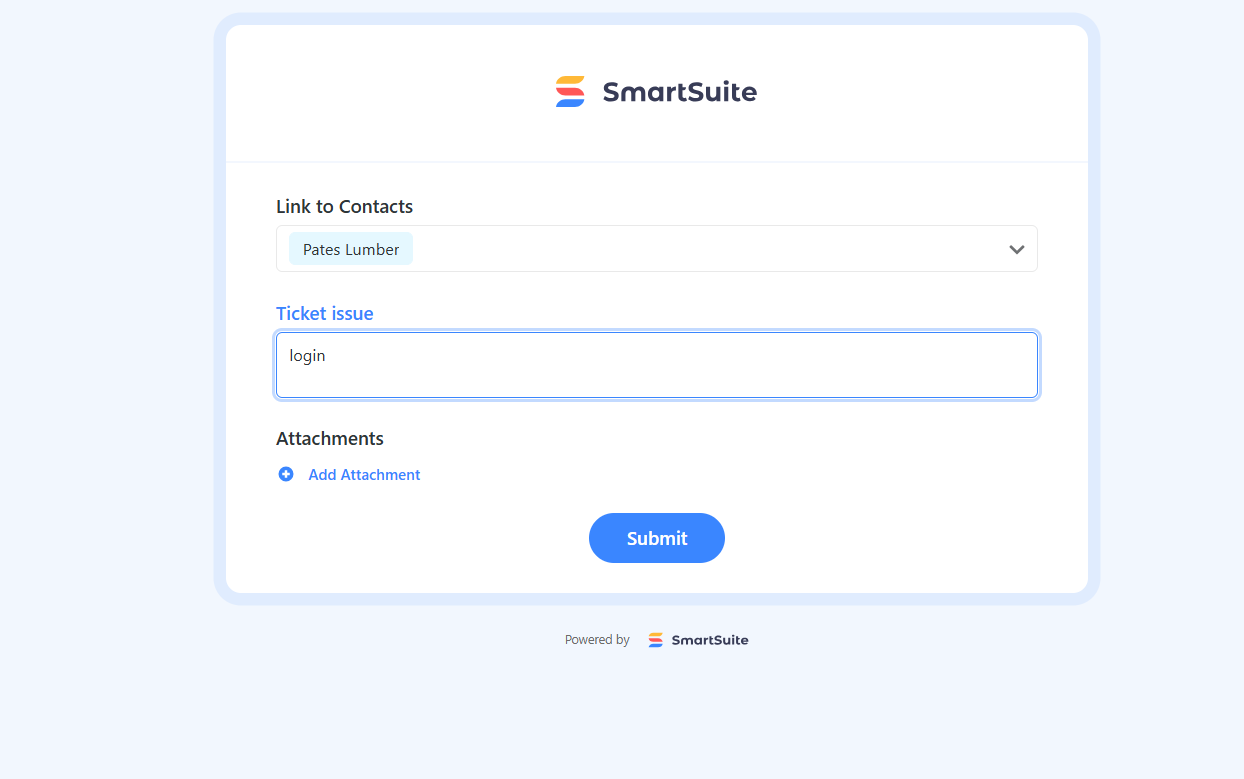
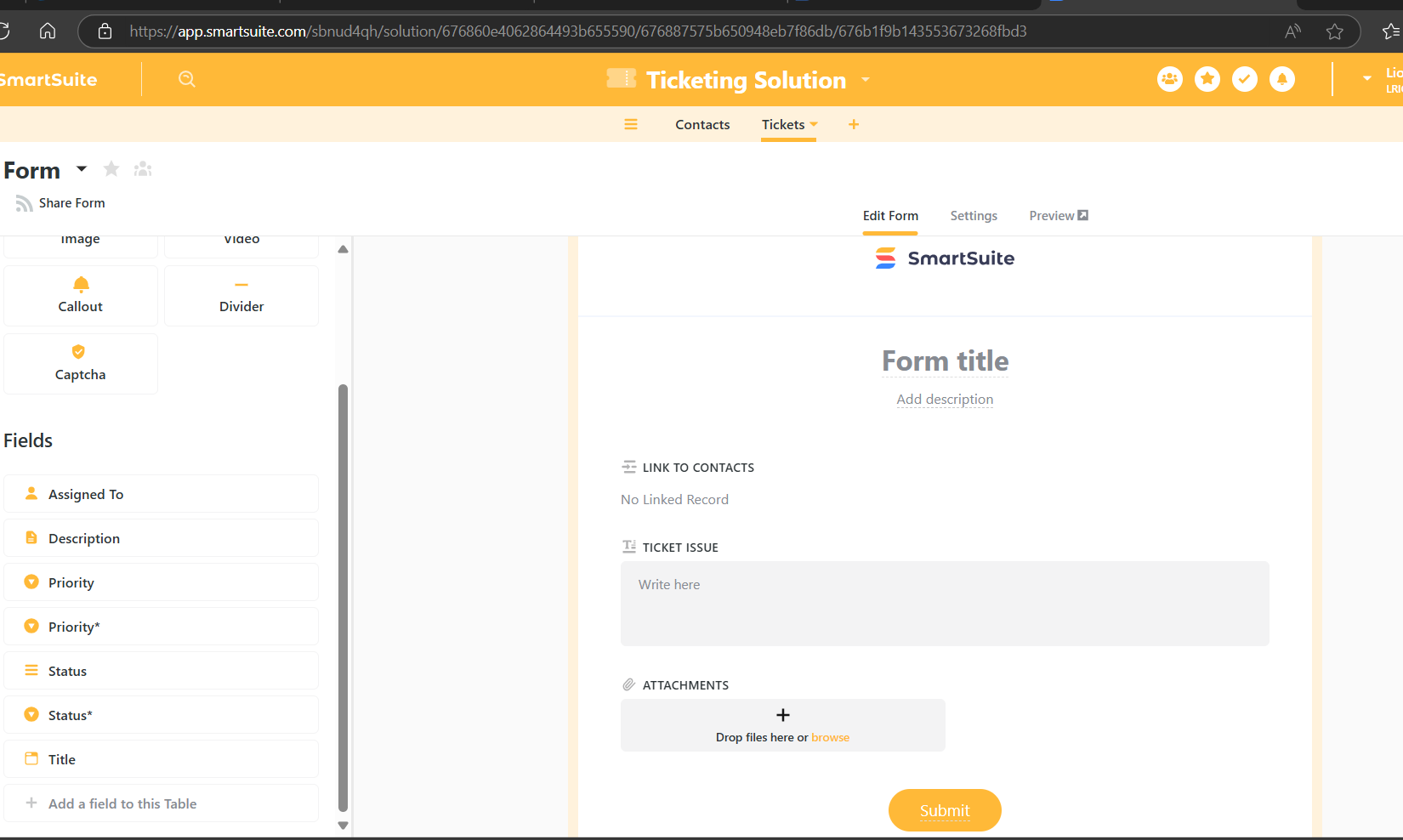


Next we should consider the information we want on the ticket itself. When someone submits a ticket we need to categorize that and have them fill out some additional information. “add new field > text area field >”. A text area field- is a large block of text used to collect several sentences from clients pertaining to technical issues they are currently experiencing. Let’s rename this field to “Ticket issues”. Add field.

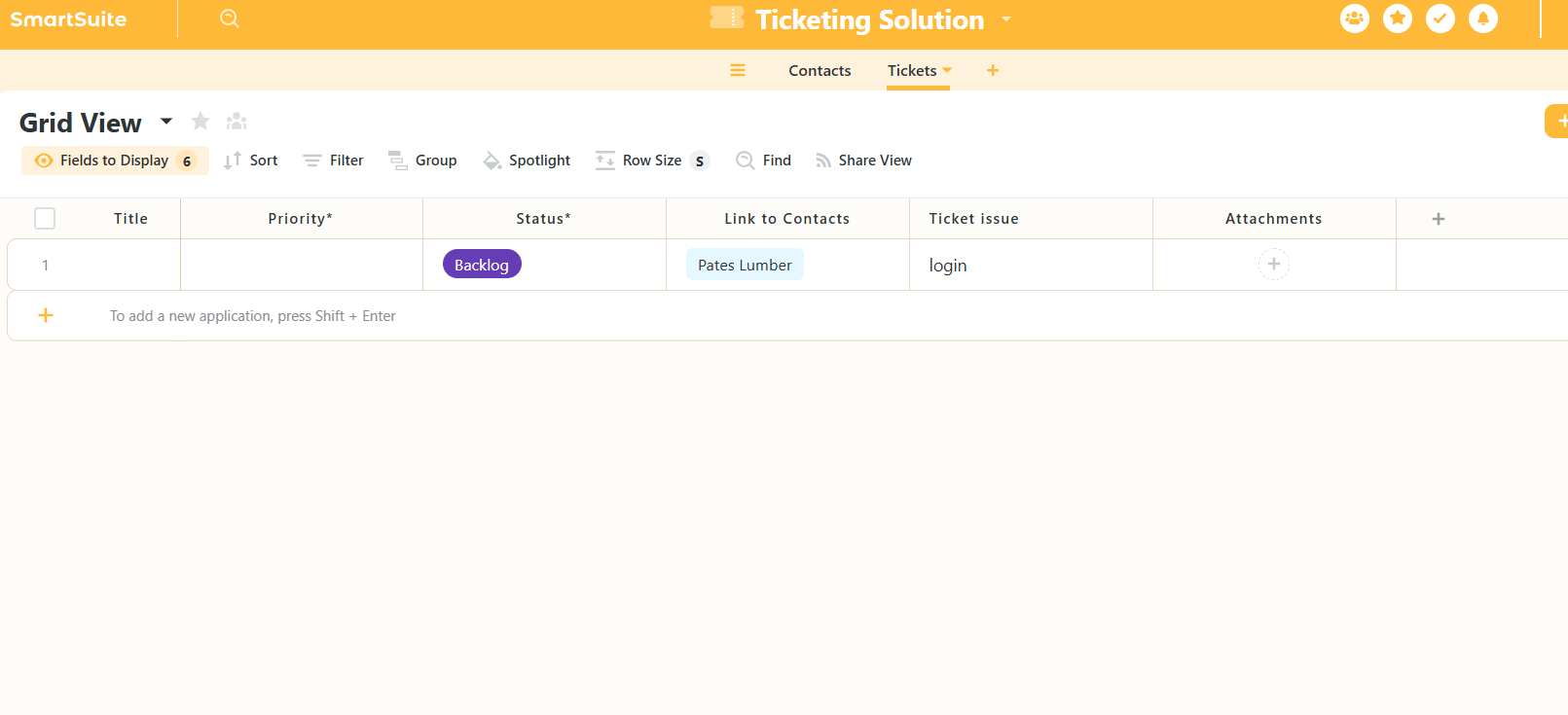
If clients perfer to send screenshots of an issue it wouldn’t be a problem. We’re likely to already have an attachment section set. Add field > search for “files & images” > rename “Attachments” > update and add.

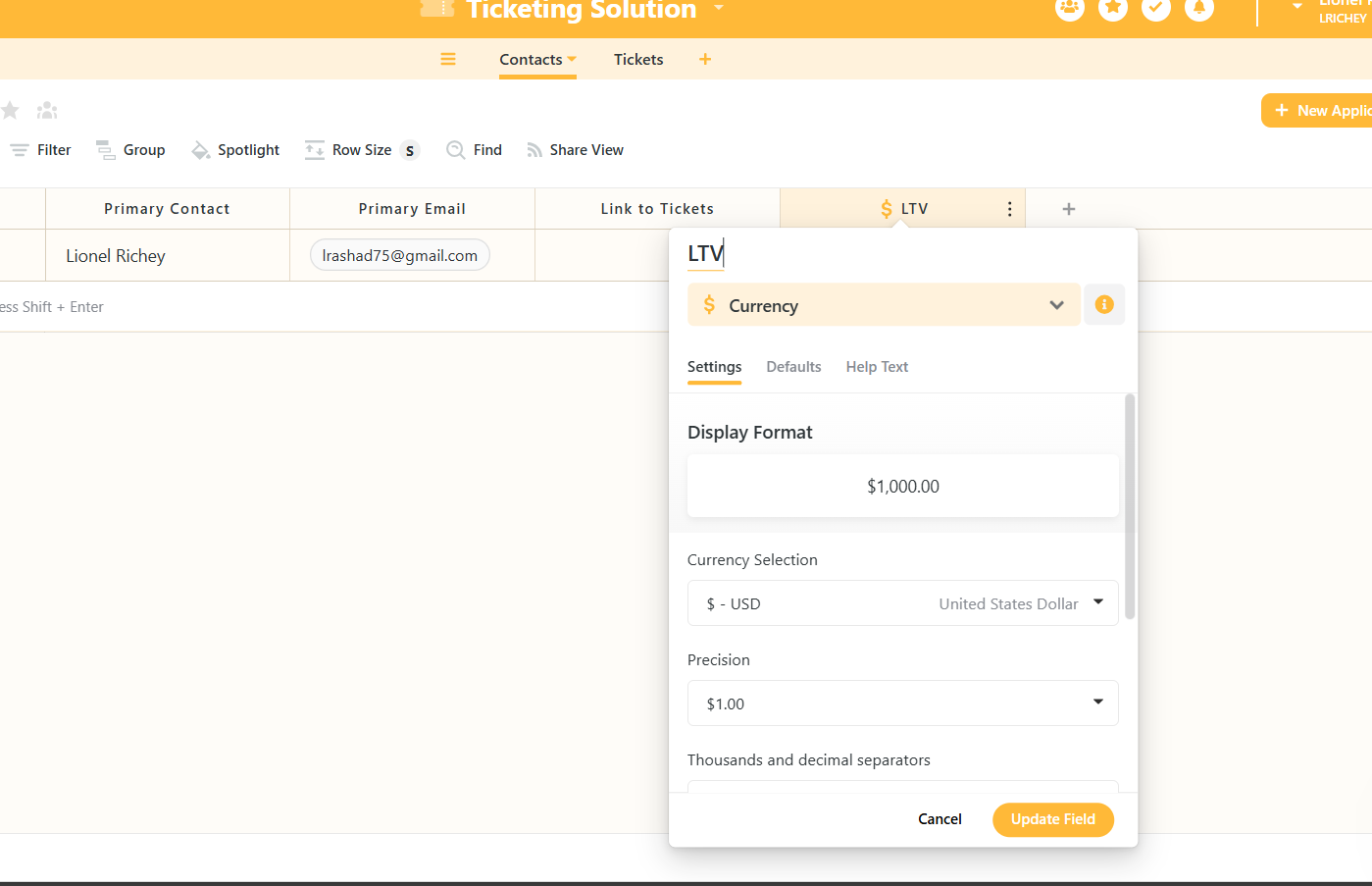
This next feature is pretty cold lol. Click on “Grid view(top left)” > create new view > Find form > make selection create view. This tool allows you a preview of what is seen by the person submitting the ticket. The primary fields that we ideally want to present in the ticket are contacts, ticket issues, attachments. Once we have the form tailored to our liking, lets copy the link from the sharable link section and send that to accounts we want to be able to submit tickets into our system.



Looking back at the ticket app, we see that the ticket was accpeted at this point all our team has to do I track it through the process. From this screen, we can see that all requested info was delivered status is back logged and no assignment of priority yet, that will be for our team to update as we work towards completion of the ticket.

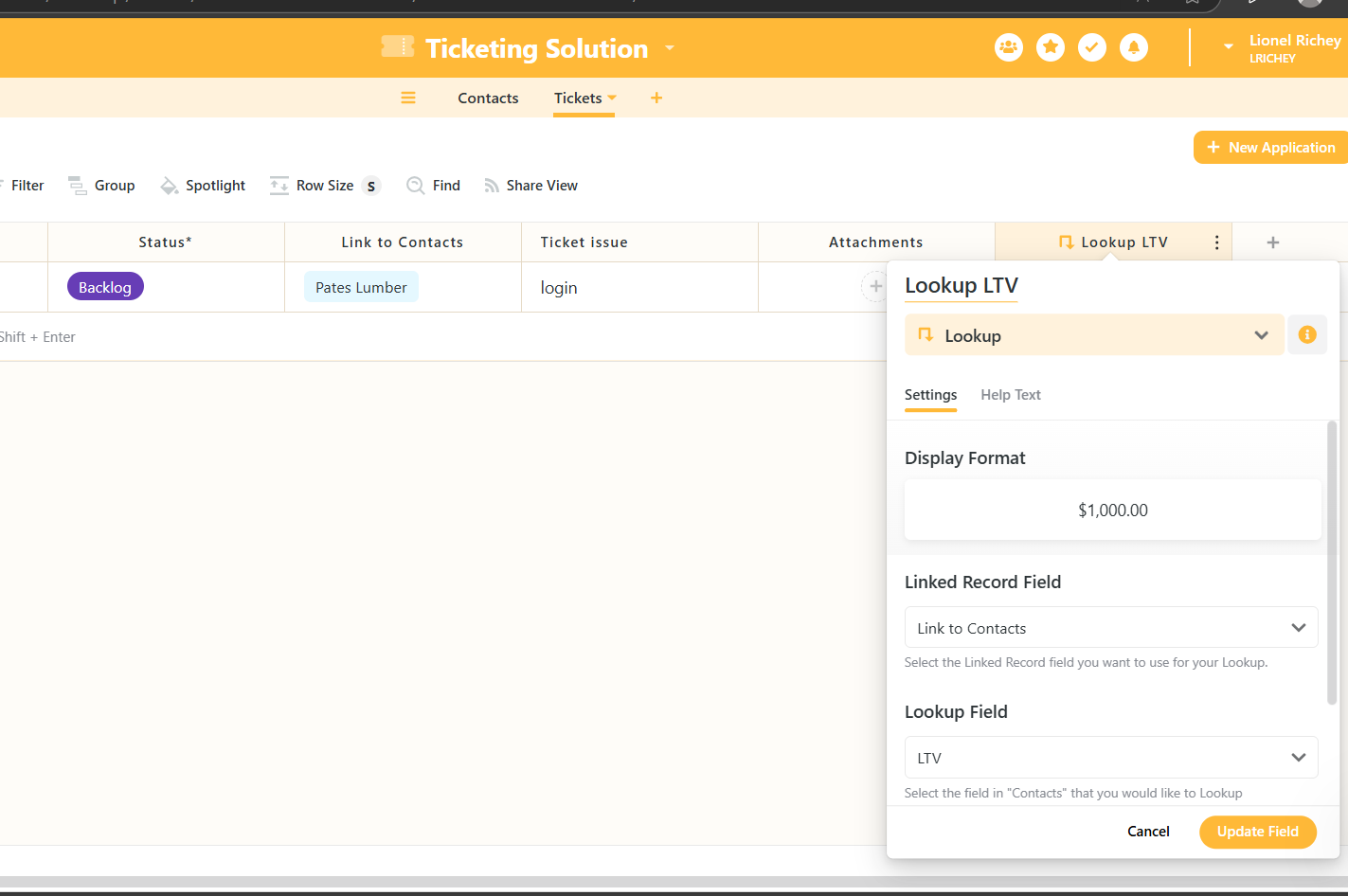


Some organizations like to categorize ticket urgency by amount of money the client brings to the table. So after going back to our contact’s tab, in our next field we will have the option to add a “ Lifetime Value” (LTV). Add field > currency field type > rename LTV >, for orginazations that tend to handle tickets by monetary contribution, this feature will quickly help yo sort through vast amounts of accounts.

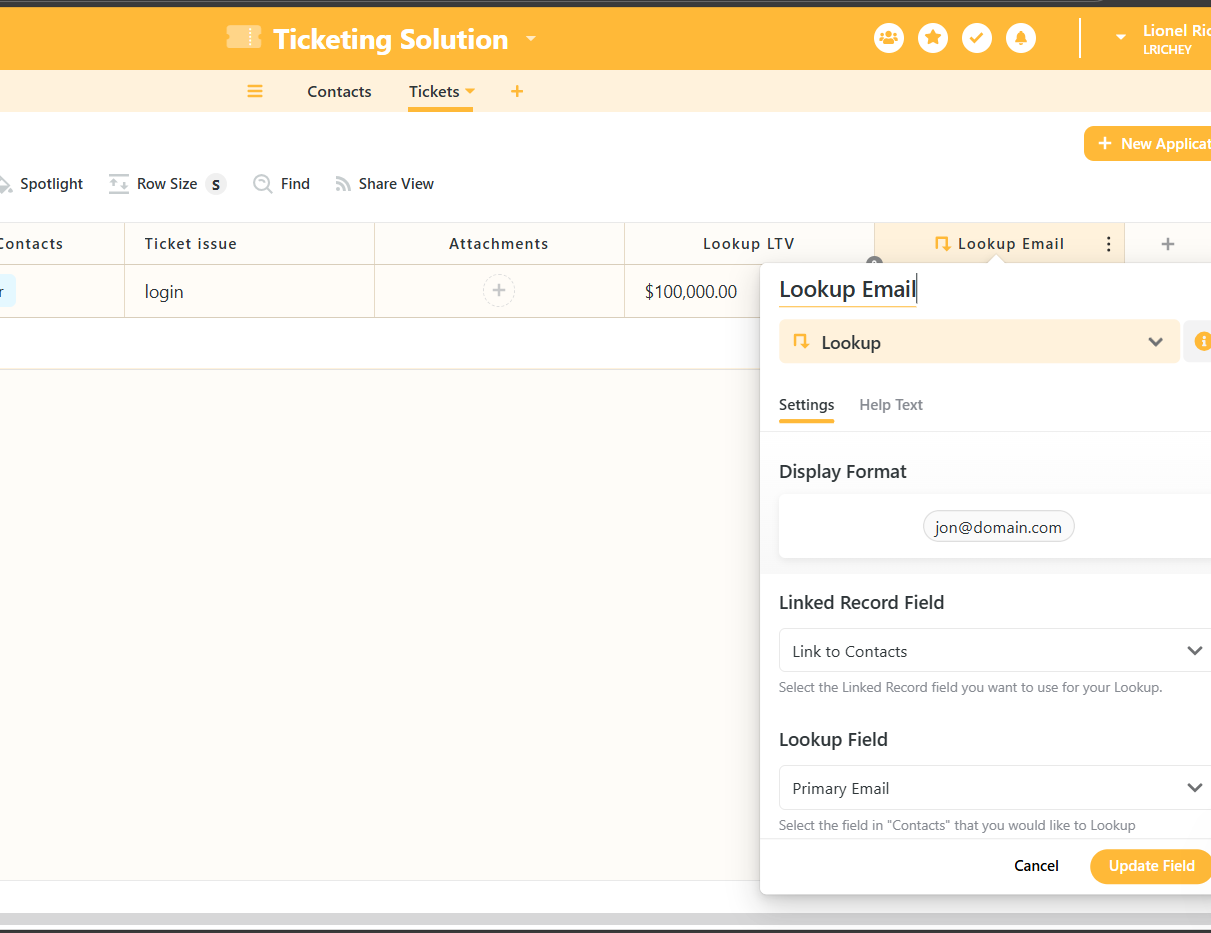
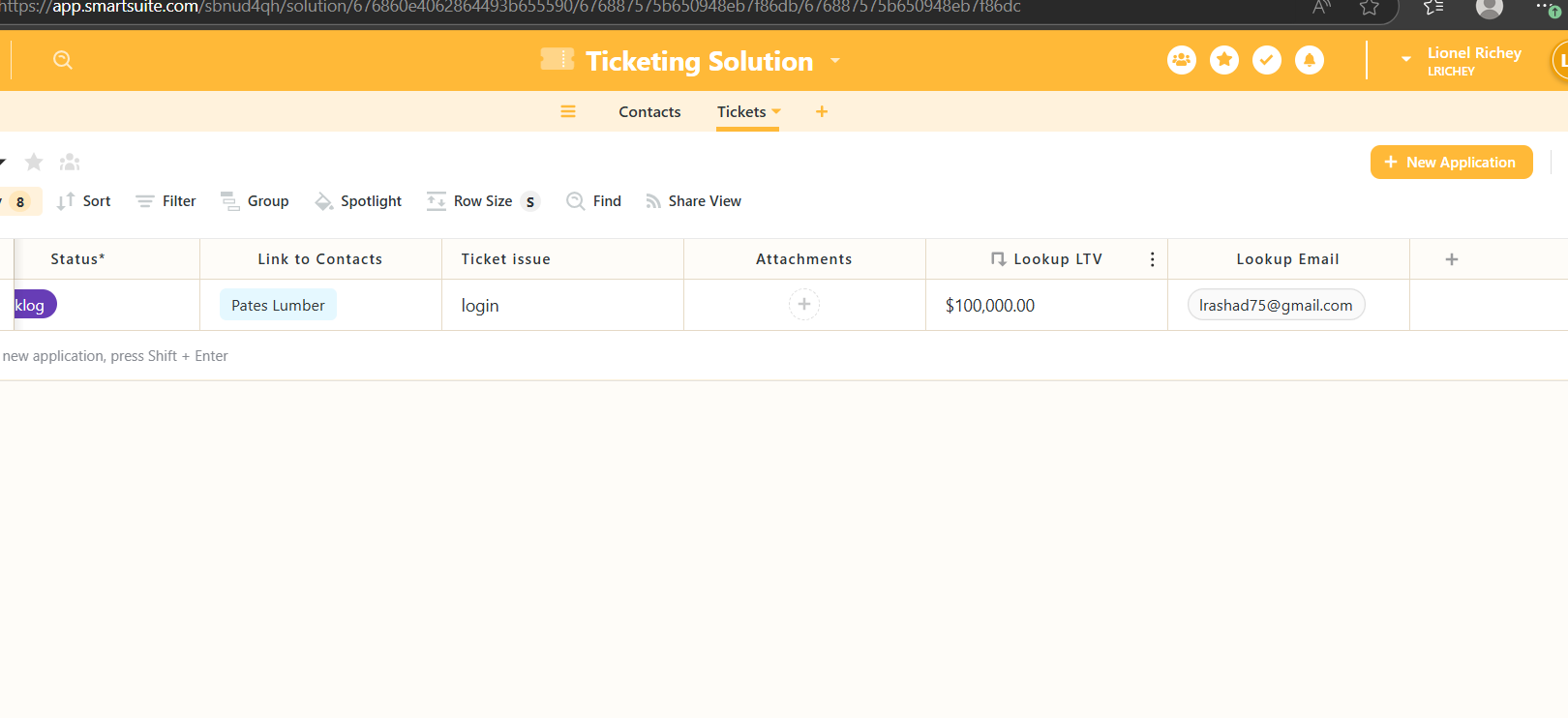


Section 4

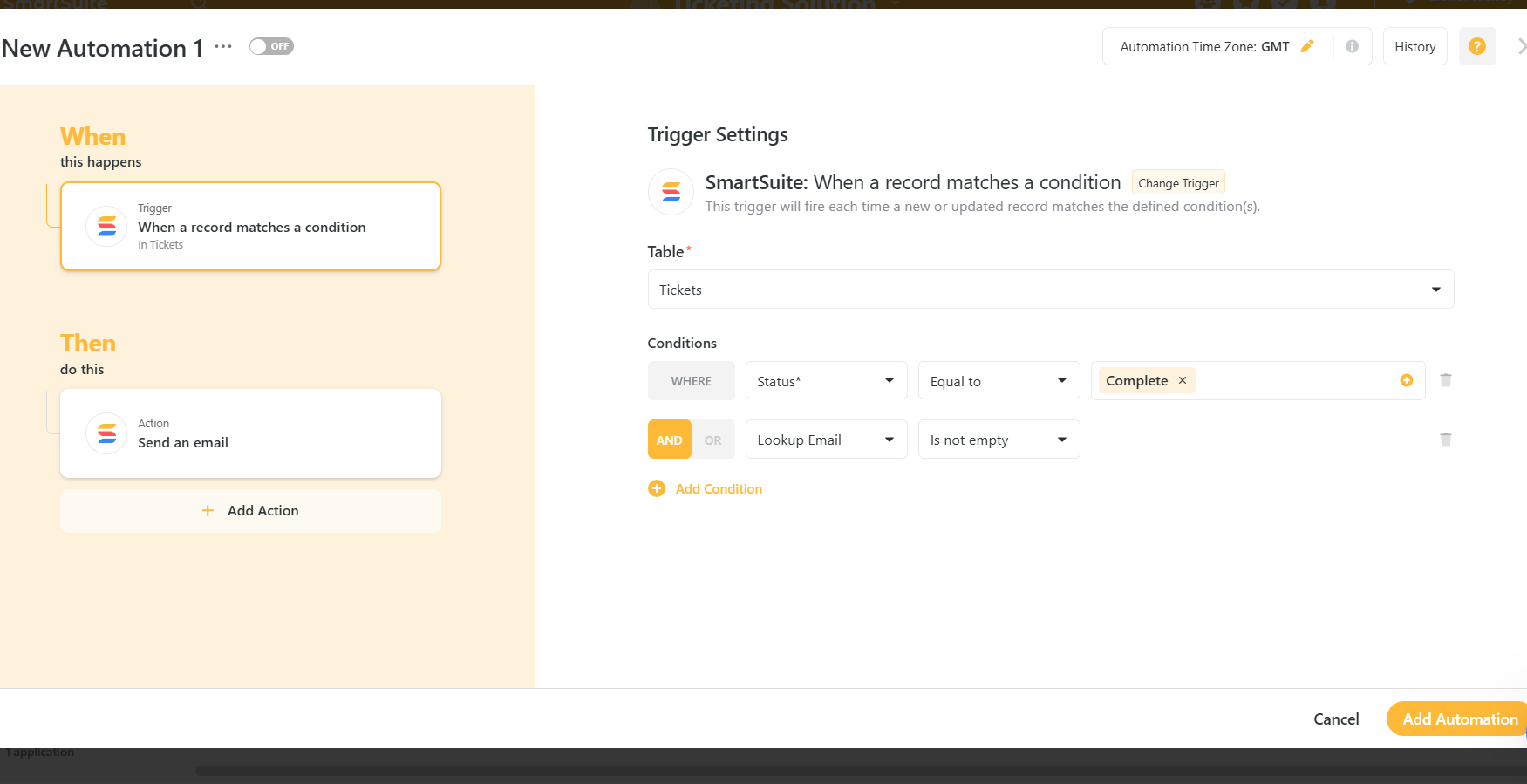
By having this knowledge on hand we could set values and have our ticketing system assign priorties automatically. In order to automate this process we will need to add a field. Add new field > lookup > lookup LTV. What we want, is if a new ticket comes in and it’s LTV is 75.000.00 with the high being 100.000.00, our lower end being 20.000.00 we could tell it any number over 50,000 should automatically be labeled High.



Finally, the most important piece of this process would be building automation around the tickes themselves. As the tickets move throughout the process we are able to automatically configure SmartSuite to notify our clients of a completion status. Let’s add a field. Add new field > lookup > lookup email now we have all the info we need included on the ticket. So that when the ticket reaches our desired status we can send it to whosever email is set for that particular account.

Once we have finalized those entries, we can begin to build the automation. Click and view, Ticket Solutions > Automation > when a record matches conditions > priority field, Status\* + to complete and ensure that the lookup email field is not empty.



Tho the left side of the screen we will see the then option. Click add an action. For convenience I’ve chosen to send from SmartSuite, although there are several choices, some other popular ones include gmail and outlook. In the action email we can notify the recipent with a short message stating “Your ticket is now Complete, thank you for submitting through SmartSuites”.

I hope this presentation will be able to help someone pick up basi, but neseccary help desk ticketing system skills, which may advance their careers beyond their wildest imiginations.