

IBM Schematics Case Study

Kathy Wang

The Process

Introduction

User Research

Prototyping

Testing

Prototyping Round 2

Introduction



Introduction

The Product

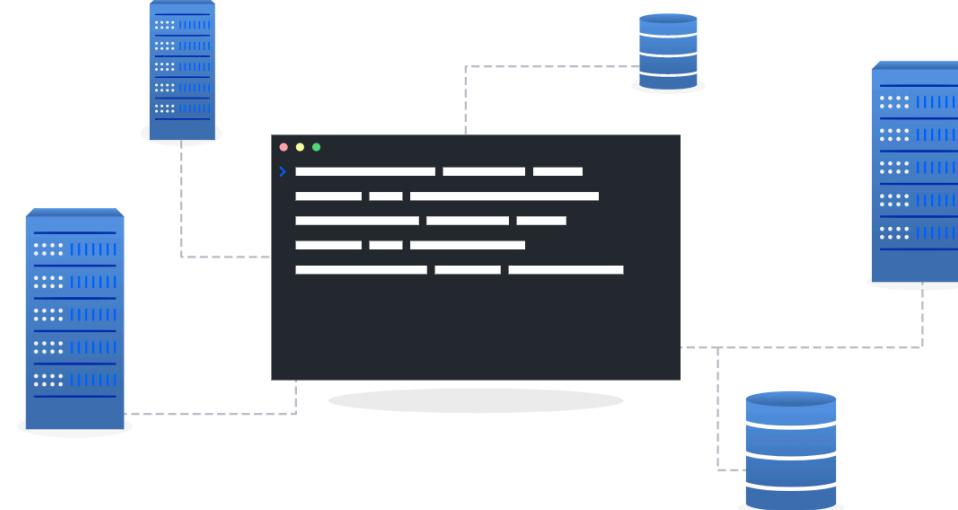
Schematics is an infrastructure as a service, enabling a DevOps engineer to deploy several resources (i.e. a Kubernetes cluster).



IBM Cloud Schematics

Enable Infrastructure as Code (IaC) and start automating the provisioning and management of your IBM Cloud resources across environments.

[Create a workspace](#)



Explore our capabilities

Terraform-as-a-Service
Use a high-level scripting language to model the IBM Cloud resources that you want and automate the setup across IBM Cloud environments. Instead of learning the API or CLI command to work with a specific resource, simply describe the desired state in a Terraform template and watch Schematics spin up the resources for you.

IaaS, PaaS, and more
Whether you want to work with IBM Cloud Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), or Function-as-a-Service (FaaS) resources, Schematics lets you define and configure all these resources so that you can concentrate on the apps that run in IBM Cloud.

Get started

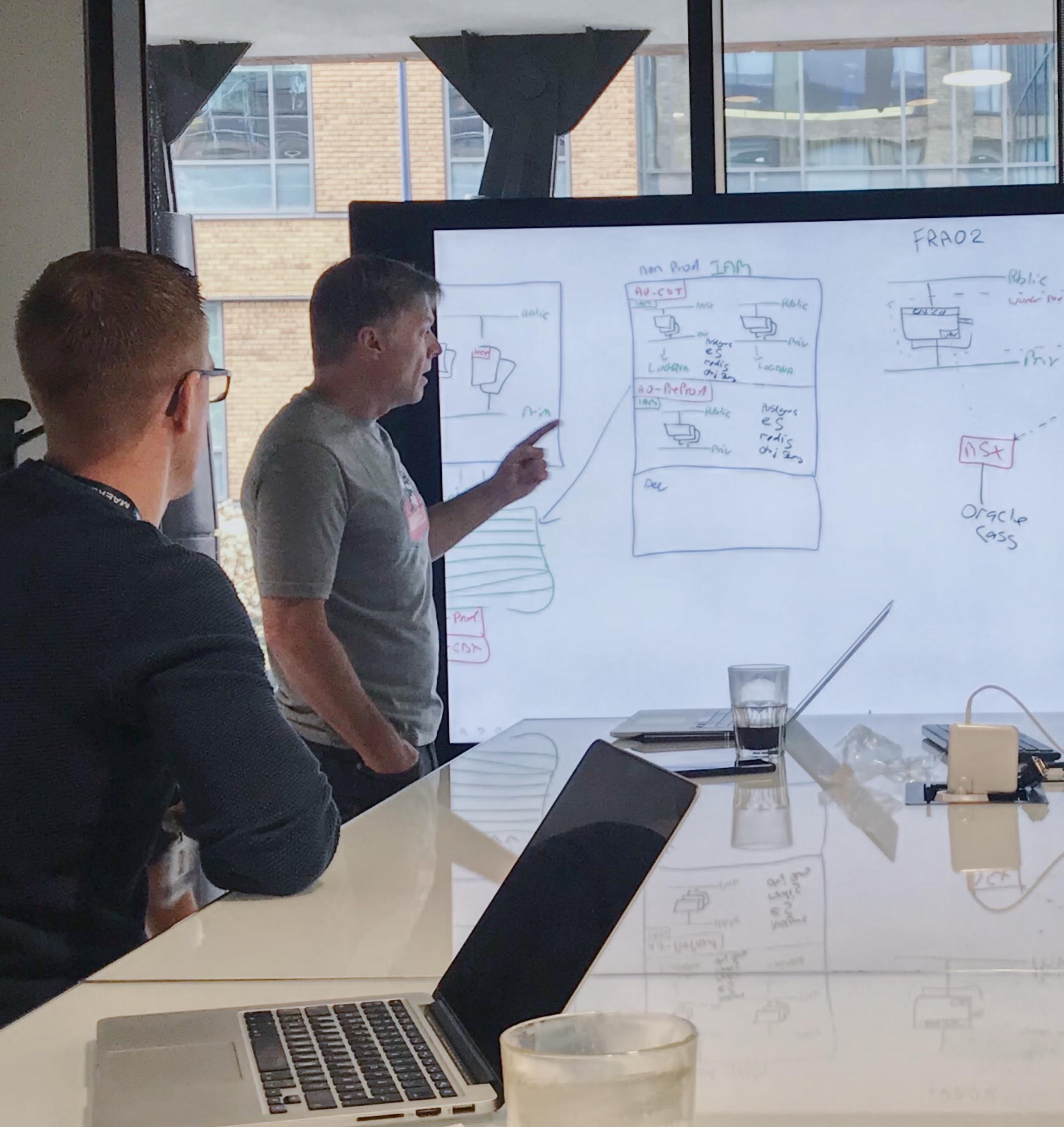
[Create a workspace](#)
Connect IBM Cloud Schematics to an existing GitHub or

[Explore the docs](#)
View the documentation to find more information about how

Introduction

Persona

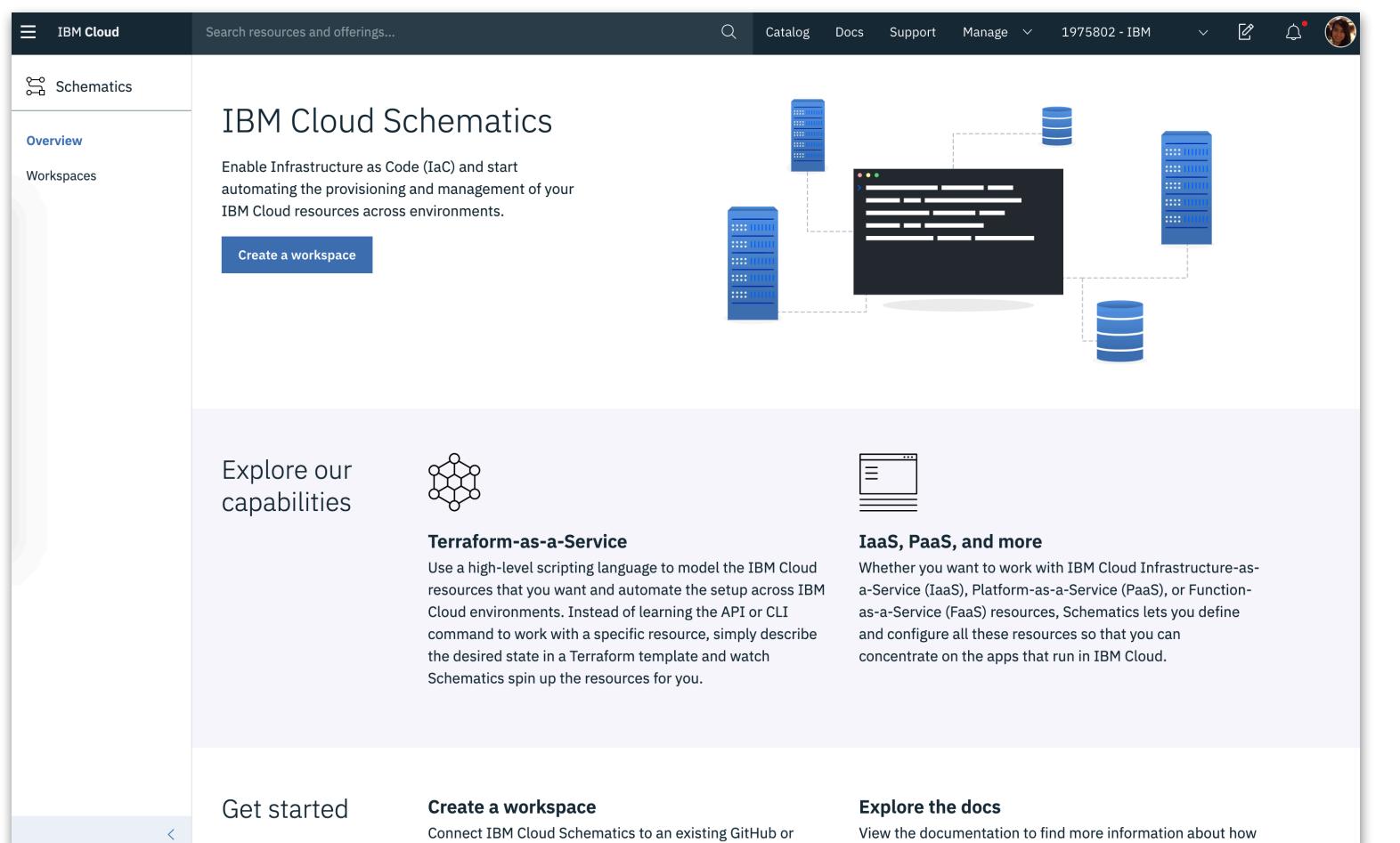
“RJ is an operations engineer with a holistic view of the network, infrastructure, applications, and services. He is responsible for the 24/7 operation of the environment including deployments, monitoring, automation, and troubleshooting. He often receives and responds to alerts in order to reduce and prevent disruptions in service.”



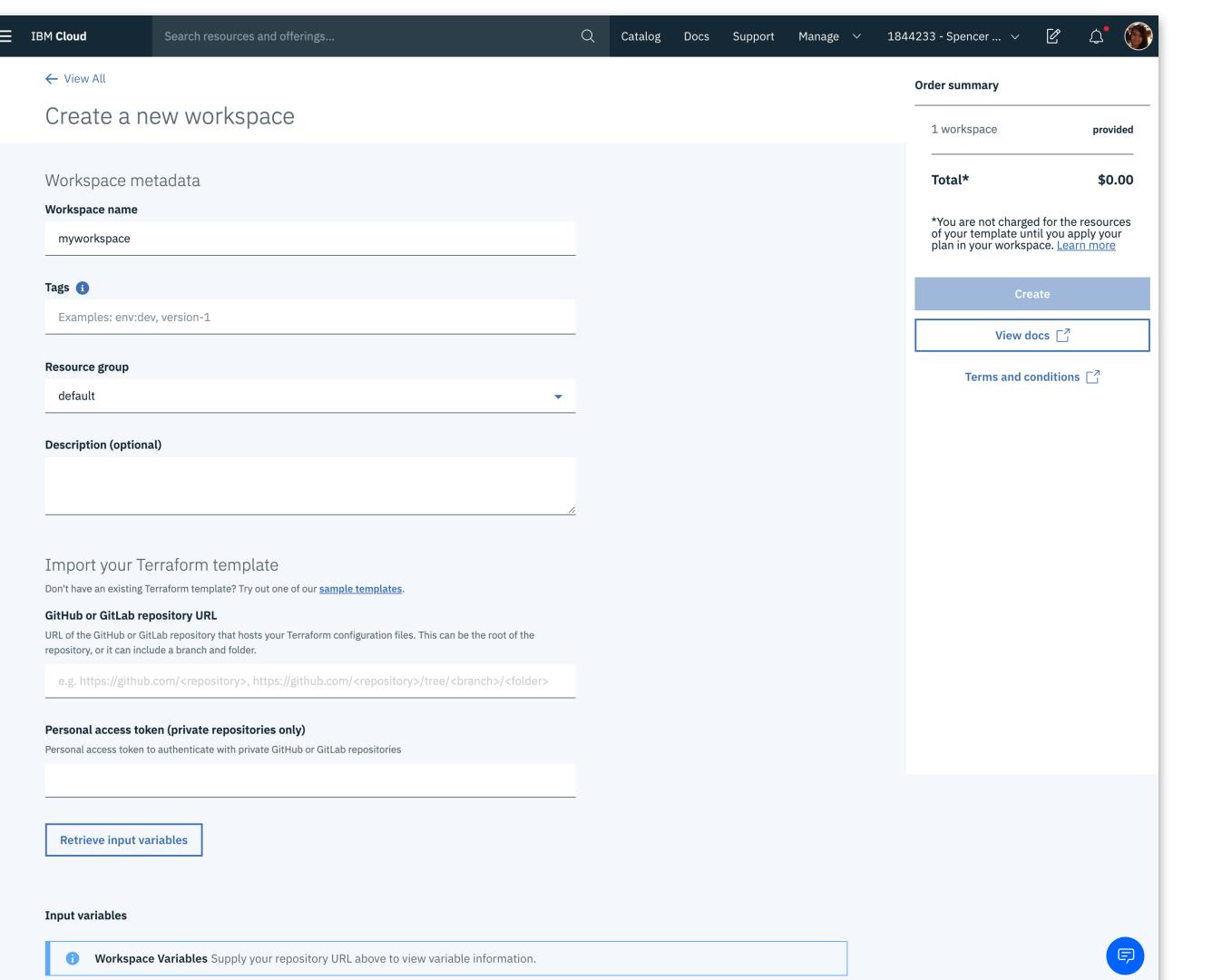
IBM

As-is

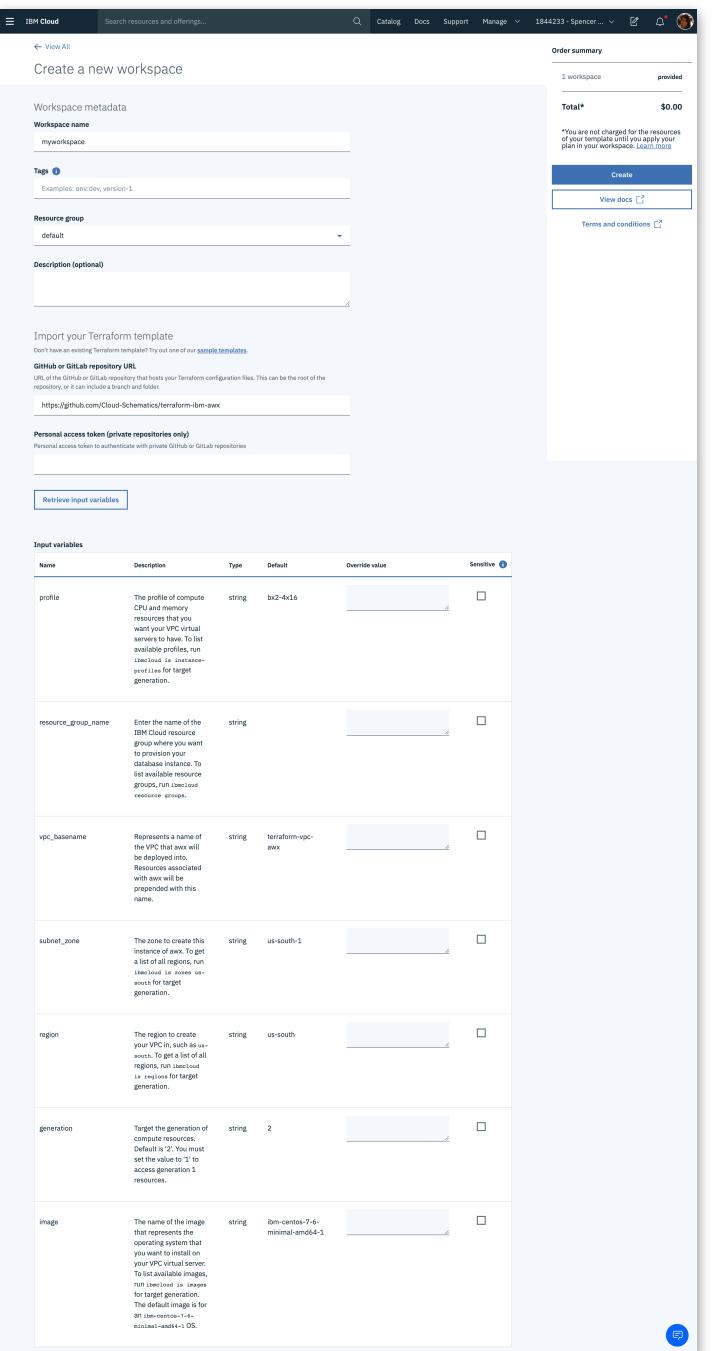
This is the current flow.



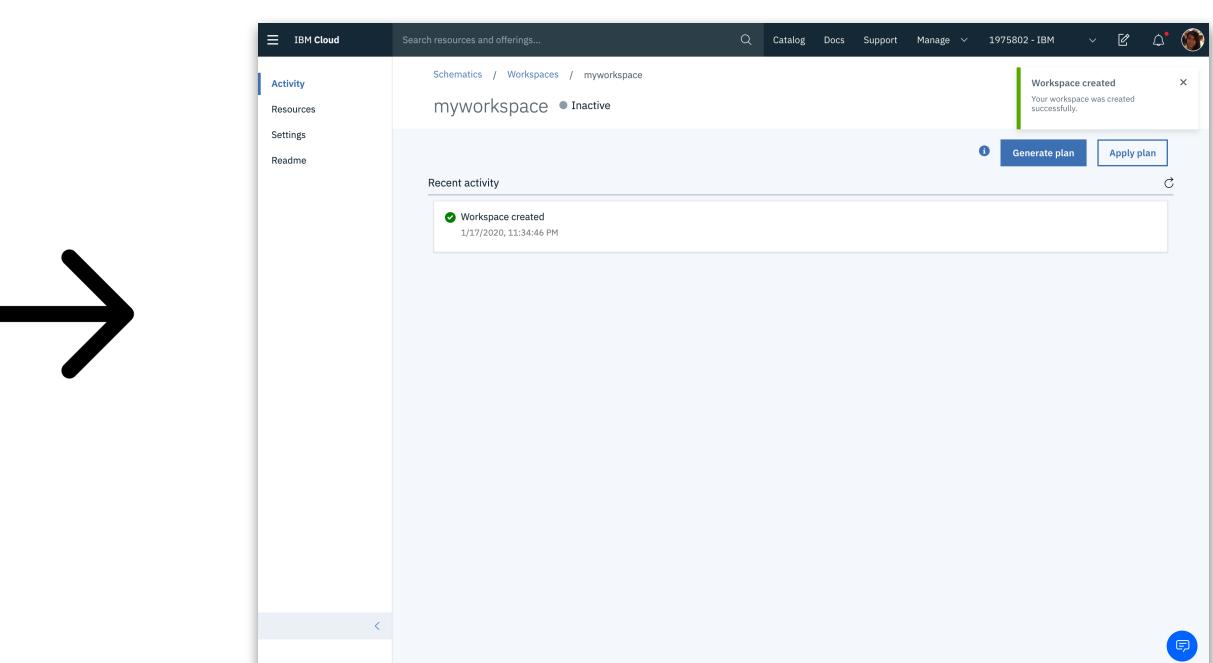
The screenshot shows the IBM Cloud Schematics interface. At the top, there's a navigation bar with 'IBM Cloud', 'Search resources and offerings...', 'Catalog', 'Docs', 'Support', 'Manage', and a user profile. Below the navigation is a section titled 'IBM Cloud Schematics' with a sub-section 'Overview'. A large blue button labeled 'Create a workspace' is prominently displayed. To the right of this button is a diagram illustrating the infrastructure-as-code concept, showing a central computer monitor connected to various cloud services like databases and storage. Below the diagram, there are sections for 'Explore our capabilities' (Terraform-as-a-Service, IaaS, PaaS, and more) and 'Get started' (Create a workspace, Connect IBM Cloud Schematics to an existing GitHub or GitLab repository, Explore the docs). A large black arrow points from this screen to the next one.



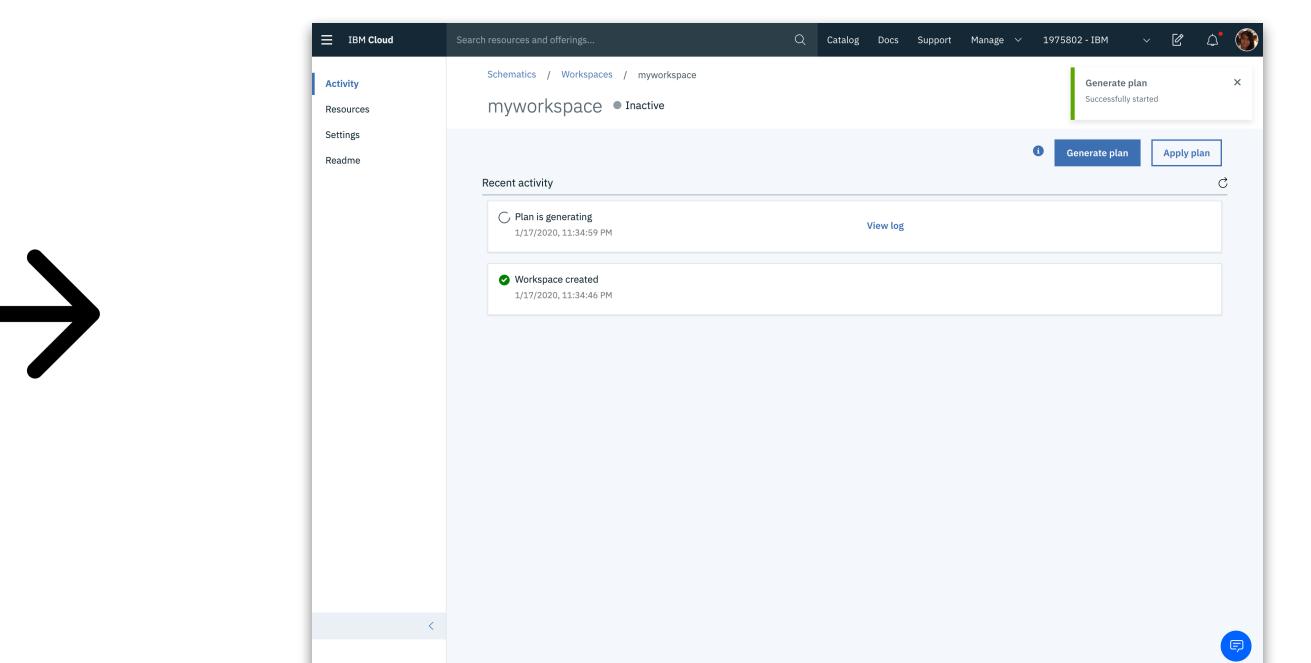
The screenshot shows the 'Create a new workspace' form. It includes fields for 'Workspace name' (set to 'myworkspace'), 'Tags' (examples: env:dev, version-1), 'Resource group' (set to 'default'), and a 'Description (optional)' field. On the right, there's a 'Create' button and a 'View docs' link. Below the form is a section for 'Import your Terraform template' with fields for 'GitHub or GitLab repository URL' (example: https://github.com/ibm-schematics/ibm-schematics.git) and 'Personal access token (private repositories only)'. A 'Retrieve input variables' button is also present. A large black arrow points from this screen to the next one.



The screenshot shows the 'Create a new workspace' form with a focus on the 'Input variables' section. This section contains numerous variables with their descriptions and default values. Variables include 'profile' (ibm-eks), 'resource_group_name' (ibm-schematics), 'vpc_subnetname' (vpc-subnet-0), 'subnet_iprange' (192.168.1.0/24), 'region' (us-south-1), 'generation' (2), and 'image' (ibm-cosmos-7.5-primary-node-1). A large black arrow points from this screen to the next one.



The screenshot shows the 'Workspaces' page. It lists a single workspace named 'myworkspace' with the status 'Inactive'. A message indicates that the workspace was created successfully. Below this, there's a 'Recent activity' section showing a log entry for 'Workspace created' on 1/17/2020 at 11:34:46 PM. A large black arrow points from this screen to the next one.



The screenshot shows the 'Workspaces' page for the 'myworkspace' workspace. The status is now 'Inactive' with a note that the workspace was created successfully. A message indicates that a plan is generating. Below this, there's a 'Recent activity' section showing a log entry for 'Plan is generating' on 1/17/2020 at 11:34:59 PM. A large black arrow points from this screen to the next one.

A photograph of a person sitting at a desk, viewed from the side. They are wearing a grey hoodie and are looking down at a laptop screen. Their right hand holds a silver pen over an open notebook, and their left hand rests on the laptop's trackpad. A white mug sits on the desk to the left, and a small potted plant is visible in the bottom left corner.

User Research

User Research

Interviews

4 usability sessions

1. DevOps Engineer, Toyota Connected
2. SRE, brightwheel
3. STSM, IBM
4. Executive IT Specialist, IBM

Interviewee Background

LinkedIn

Intro (5 min)

How's it going? We're with the IBM Cloud team here in Austin, TX. We have a new feature that we're hoping to do some usability testing.

Before we begin, do you mind if we record?

1. What's your role? And can you tell us a little bit about what you do?
2. I see that you have been working as a ___ for X time. Is that correct?
3. I see that you listed Terraform as a tool that you use, how are you using it today?

Tasks (20 min)

We're going to give you a task related to Terraform. As you are doing it, tell us what you are thinking out loud. There is no right or wrong answer here and we are not testing you on your skills but we are more looking to smooth out the feature before it goes to Generally Availability. If you have a question, like if something is not clear, definitely voice it in the moment and we will come back to it at the end.

1. Task #1: Here's some TerraForm starters. Deploy one of these to IBM Cloud.
 - a. [Starter link](#)
 - b. Things to Validate (Observe and Probe):
 - i. Time (?)
 - ii. Task completion rate
2. Dig into workspace creation, generate plan, apply plan. What if we generated on create workspace?

Notes

Post-Assessment (5 min)

1. Satisfaction rating
2. Was there anything you found particularly enjoyable about your experience?
3. Was there anything you found particularly challenging?
4. Is there anything you would change?

User Research

Synthesis

We took the important data points, and then gathered them into one document.

We found out that users were really annoyed with the fact that variables were not retrieved for them.

Participants

2 external (DevOps Engineer at Toyota Connected, SRE at [brightwheel](#))

2 internal (STSM, Executive IT Specialist)

Satisfaction

- 3 initially bc confused at prompt; actual task solid 4, only issue is logs and feedback
- 4, close to 5 but 5 is perfect so he said 4, it's simple, it works, it might be missing some things
- 3.6, it's ok, it's not terrible, i would want to see how hard it is to pull my terraform into this, modules, output variables

Task completion (follow up)

- no, got stuck at GitHub URL and second time at input variables
- partial, plan generate started but no input variables
- partial, plan generate started but no input variables
- yes

Issues to open

1. Input Variable Defaults: i'm logged into my account, why haven't you retrieved those variables for me? api key, username
 - a. external
 - b. (internal)
 - c. (internal)
 - d. - external
2. GitHub Repo URL: difficulty getting the right formatting for [github](#) URL, 3+ tries, 2+ minutes (Note: may be an easy change to the readme: deploy to [ibm](#) cloud vs deploy locally)
 - a. - external
 - b. - external
3. Log Navigation: difficulty viewing most recent logs and navigating around in general, wished for floating buttons to make this easier
 - a. (external)
 - b. (internal)
4. Rename Workspace: would like option to rename a workspace
 - a. (internal)
5. ReadMe Appearance: readme looks broken
 - a. (internal)

Issues already in progress

1. Workspace Create Feedback: to show system is processing / in progress (spinner or logs), to show whether activity was success or failure
 - a. (external)
 - b. (internal)
 - c. (internal)
2. Terminology confusion: confused if frozen/frozen and locked/unlocked are referring to the same thing
 - a. (internal)
3. Terraform actions: clicking buttons lead to error while workspace is creating

Feature requests

- Output variables (2 / 3)
- Download Pure Terraform: ability to download pure terraform. google lets you generate terraform state from pure infrastructure, they create template and state file for you (1 / 3)

Content review

- Inconsistent Wording: he clicked on sample templates in UI but the [github](#) repo says examples.
- Template Wording: passed over app sample template bc he says it implies he needs an app ready to go -
- Sample Naming: didn't match expectations -
- GitHub Repo URL: difficulty getting the right formatting for [github](#) URL, 3+ tries, 2+ minutes (Note: may be an easy change to the readme: deploy to [ibm](#) cloud vs deploy locally)

User Research

Requirement

Thus, we opened an issue regarding the defaults for input variables.

User testing: defaults for input variables #228

Open Gabriela-Moreno-Cesar opened this issue on Sep 16, 2019 · 4 comments

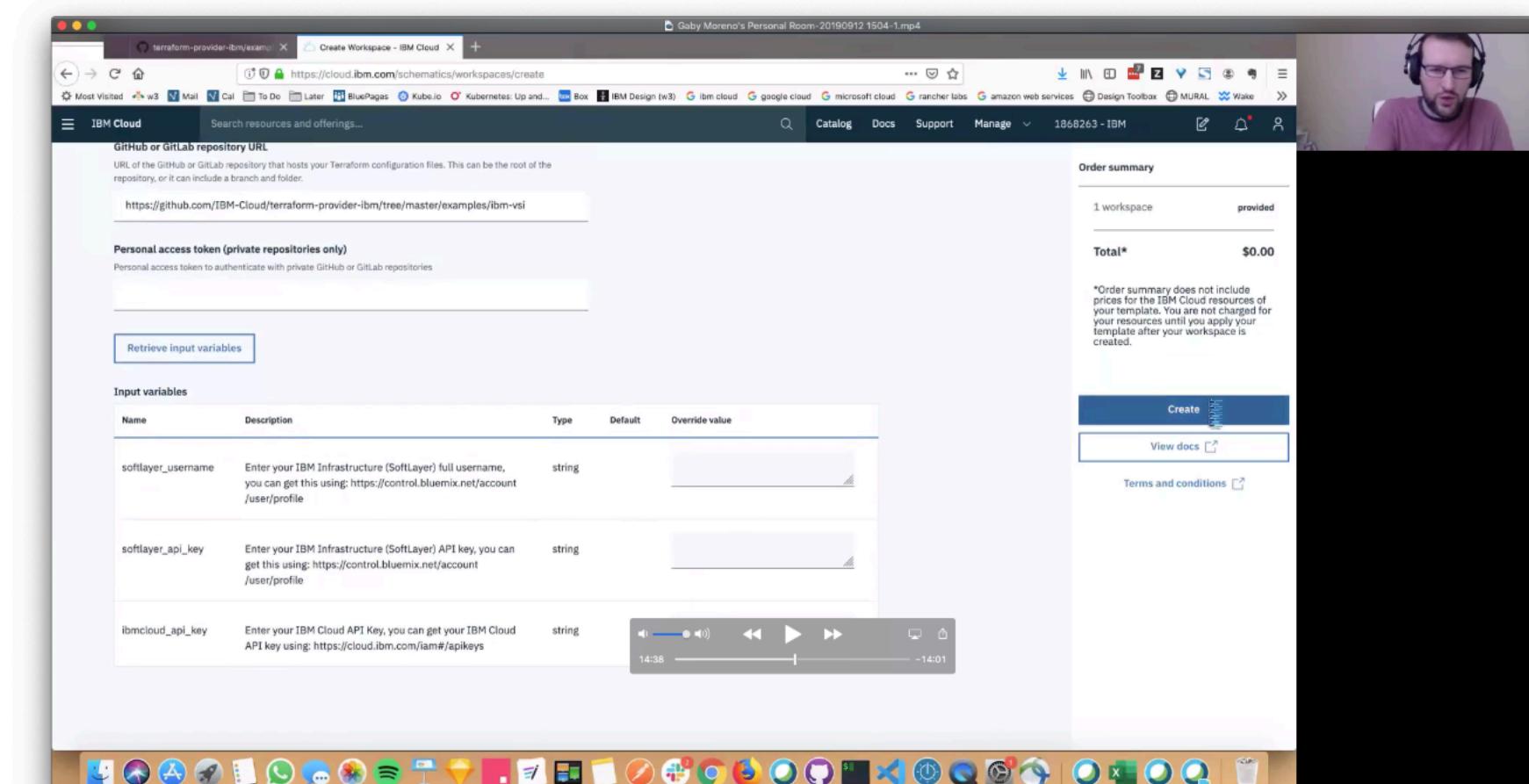
Gabriela-Moren... commented on Sep 16, 2019 • edited

This is from our usability sessions on Schematics that we ran last week.

4/4 users wondered if we might be able to provide more help with inputting variables that IBM Cloud knows about you, ex: softlayer_username, softlayer_api_key, ibmcloud_api_key.

Of the 4 tests we ran, only 1 person took the time to go dig up the variables. The other 3 created the workspace, hit plan, and stopped there with the task.

Some of the samples that they tried loading in from the github repo didn't have any descriptions for the variables either and they mentioned that more context would be nice.



User testing: defaults for input variables has no dependencies

+ add dependency

Pipelines

S Schematics Personal Workspace New Issues

Show 2 more...

Manage this Issue's pipeline per Workspace. Change its pipeline to show progress in other Workspaces. See related Workspaces

Assignees

janardhanreddy

Labels

P2

Projects

None yet

Milestone

No milestone

Estimate

No estimate yet

Releases

Not inside a Release

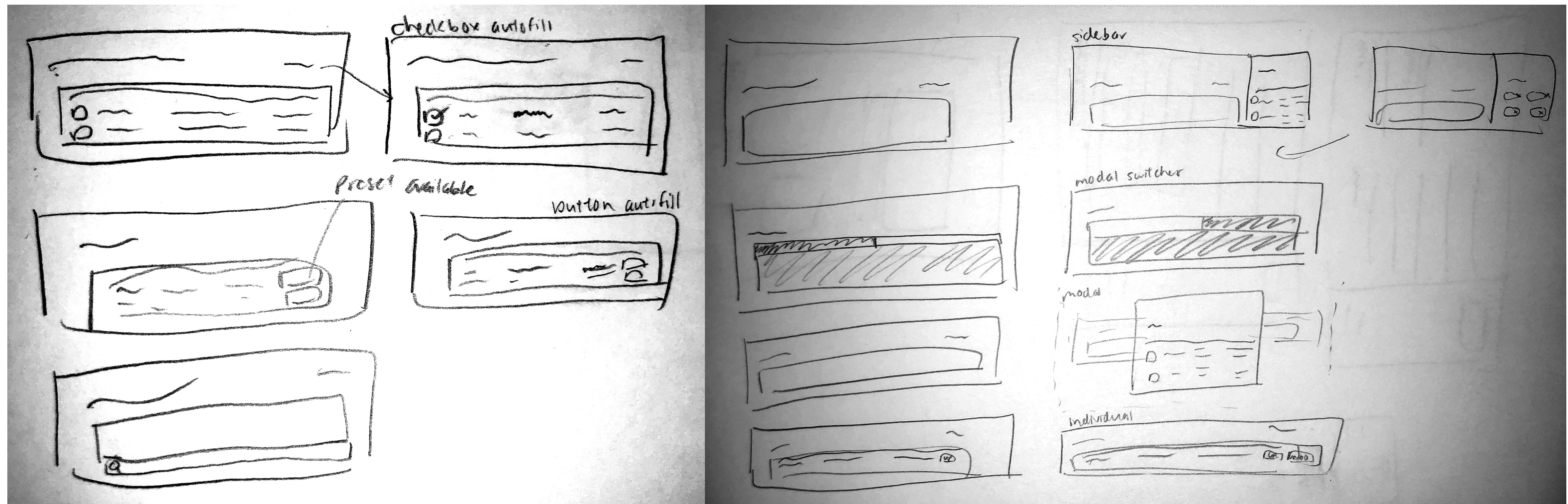
Define

Schematics users want to fill out the values for variables so that they can generate and apply plan, but they can't do this, because they get annoyed having to fill out the values, and thus, don't fill them out.

Prototyping

Prototyping

Low-Fidelity Explorations

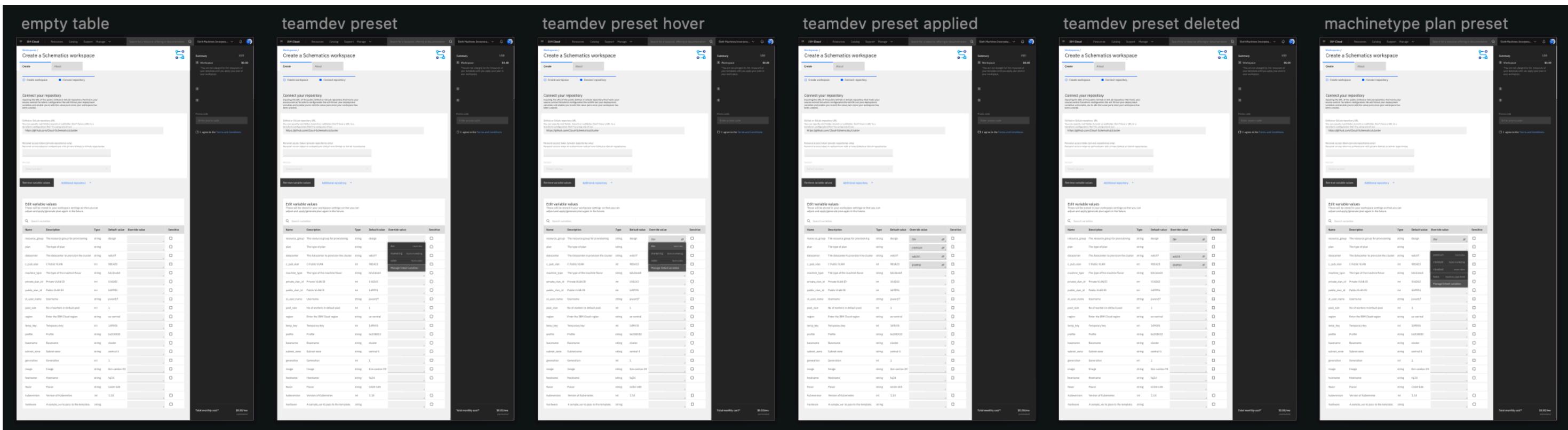


Prototyping

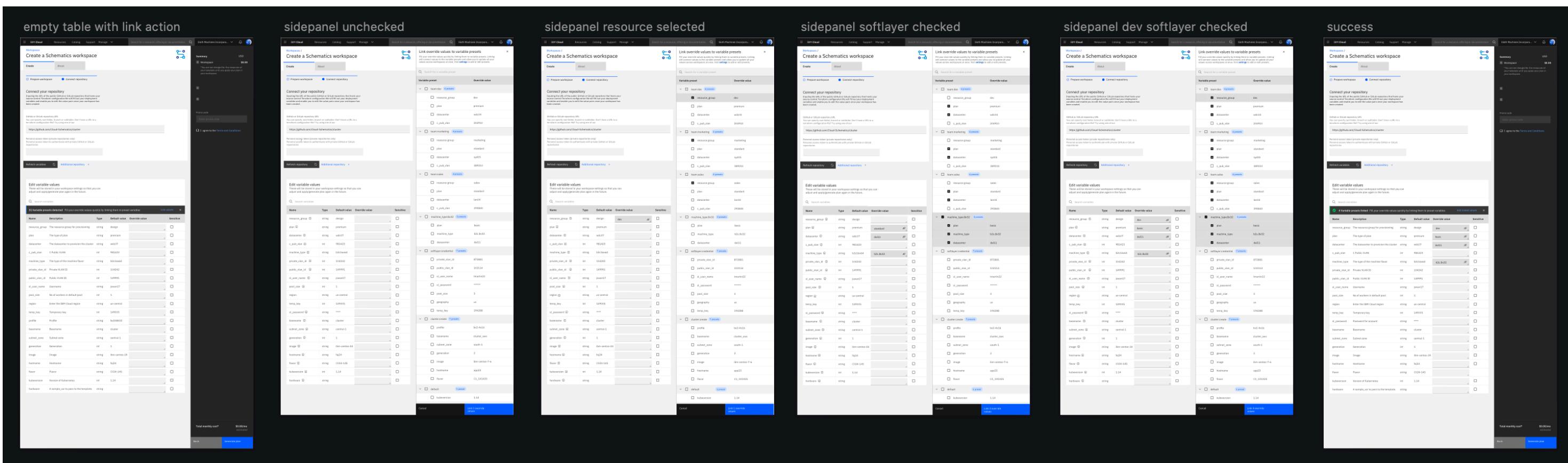
High-Fidelity Prototyping

We ended up with two concepts: autofill and side panel.

Autofill



Side panel



Discussion Point: Reveal System vs. Provide Convenience

The advantages of the autofill were discoverability and convenience, while the advantages of the side panel were visibility and understanding. The question here was, do we need to reveal the system in order for it to be useful (similar to Google Chrome's autofill feature)? We decided to test this out.

The image displays two screenshots side-by-side. On the left is a screenshot of a travel booking form on Google. It shows a 'PERSONAL DETAILS' section where 'First name' is set to 'Kathy'. A dropdown menu is open over this field, showing suggestions: 'Kathy', 'Kathy T Wang', and 'Manage...'. On the right is a screenshot of the Google Chrome settings page under 'Passwords'. It lists several saved password entries, each with a website, username, and password (represented by dots). The entries include 'dl.acm.org', 'login.activebuilding.com', 'aetna.com', 'amazon.com', 'passport.amazon.work', and 'americanexpress.com'. Both screenshots illustrate how the system provides convenience through autofill or password storage.

A photograph showing a person's hands resting on a light-colored wooden desk. In the background, a laptop is open, displaying a dark screen. The overall lighting is warm and focused on the hands.

Testing

Testing

Research plan

We created a research plan, ensuring that we captured our existing assumptions, so that we could test them.

Motivation

When creating a workspace, users of schematics are shown an option to fill in [a certain area of tags/metadata]. Past testing has shown that people do not want to and sometimes will not fill in these tags. However, filling this is could be very useful for them in the long run. (why) We want to make this understandable for users and easy for them to complete.

General research goal

The goal of the research is to decide between two design options.

The options aim to solve the problem that users *are annoyed about filling out a specific section of Schematics*.

We want to know which design solves that problem better and helps them to complete this stage of the flow.

In scope: We don't know if users understand these presets and can use them in a way that makes sense.

Out of scope: exploration into other ways in journey to avoid filling this section out.

Research Questions:

1. Which design leads to less experience of annoyance?
2. With which design do we see more of the intended behavior (what user was "supposed" to do)?
 - a. Which design leads to a greater fill-in rate of adding variable values?
 - b. Are they filling in the "correct" variable values? -- *what's correct*
3. Do users feel confident in what they're doing?
 - a. With which design do we see more confidence?

Methods:

In person user testing

- Usability test of two designs (half start with design A, half start with design B)
 - Task completion (binary), count of tags added
 - Answers question #2.
 - Self-reported measures
 - SEQ, Simple/Cumbersome to answer question #1
 - Confidence on task completion to answer question #3

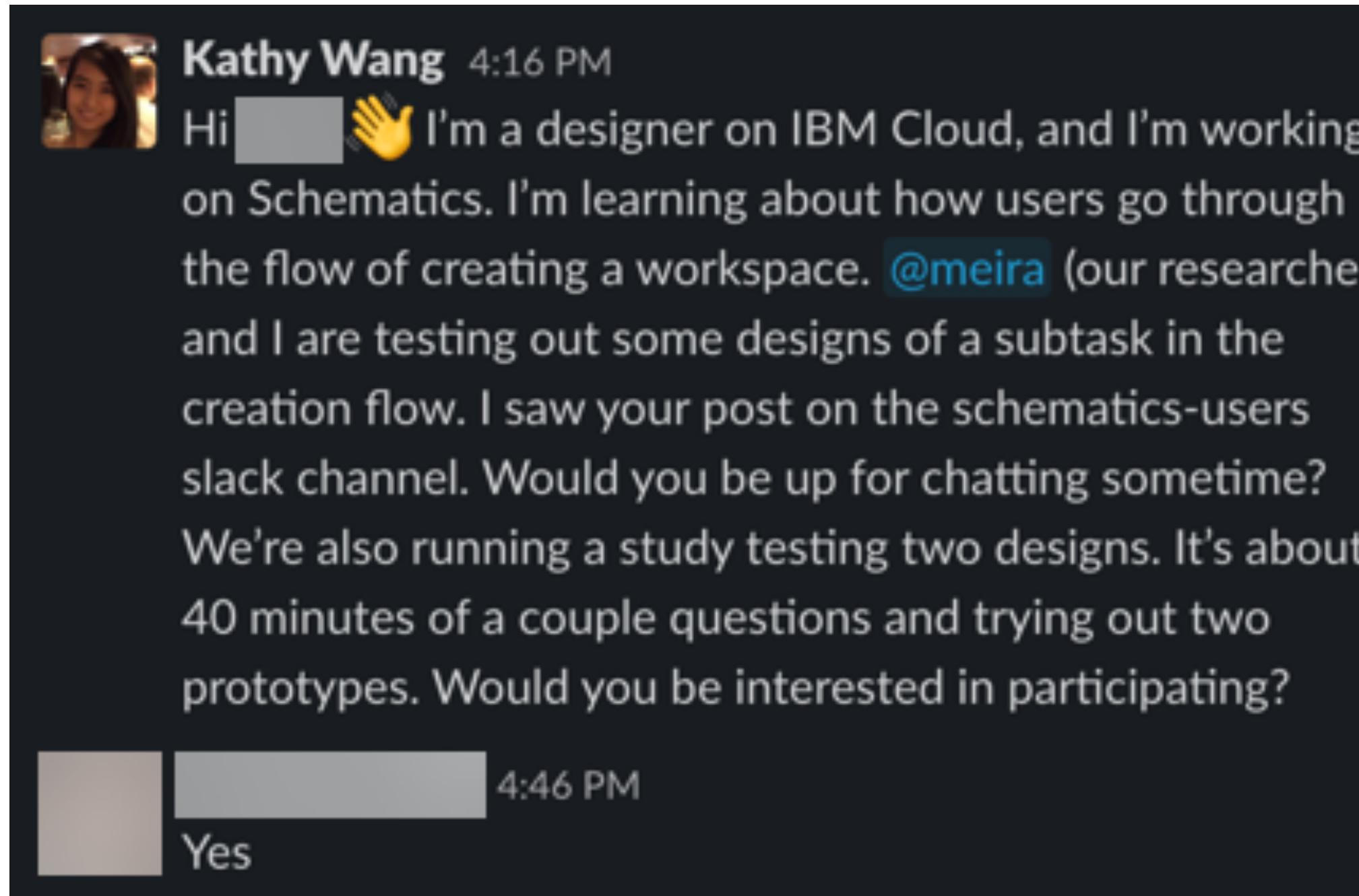
Post-test user interviews on context of text

- Gather type of employee (sales, marketing, dev, design)
- Need to gather any other details that may impact how they populate variable values
- Ask additional questions to answer above research questions

Testing

Recruit participants

We recruited participants via Slack, and I kept track of this through a spreadsheet.



Order	Name	Slack	Profile	Location	Contacted	Follow up	Scheduled	First test	Interview Notes	Recording	Roles
1	[REDACTED]	[REDACTED]	Bluepages LinkedIn	Durham, NC	<input checked="" type="checkbox"/>	N/A	1/27 Monday 10:15-11am	A	Link	Recording Password: DvUe7wJU	Facilitator: Meira Notetaker: Kathy Other(s): Spencer
2	[REDACTED]	We know him	Bluepages LinkedIn	Austin, TX	<input checked="" type="checkbox"/>	N/A	1/27 Monday 11:15am-12pm	A	Link	Recording Password: tR5A9Sx3	Facilitator: Meira Notetaker: Kathy Other(s): Spencer
3	[REDACTED]	[REDACTED]	Bluepages LinkedIn	Austin, TX	<input checked="" type="checkbox"/>	N/A	1/27 Monday 1:15-2pm	B	Link	Recording Password: AdWZ5vQ3	Facilitator: Meira Notetaker: Kathy Other(s): Spencer
4	[REDACTED]	[REDACTED]	Bluepages LinkedIn	Durham, NC	<input checked="" type="checkbox"/>	Followed up 1/27	1/29 Wednesday 12-12:45pm	B	Link	Recording Password: 8zPWdwKf	Facilitator: Meira Notetaker: Kathy Other(s): Spencer
5	[REDACTED]	[REDACTED]	Bluepages LinkedIn	Cambridge, MA	<input checked="" type="checkbox"/>	N/A	1/29 Wednesday 1:15-2pm	A	Link	Recording Password: gRE5FAM4	Facilitator: Spencer Notetaker: Kathy Other(s): N/A
6	[REDACTED]	[REDACTED]	Bluepages LinkedIn	Portland, Oregon	<input checked="" type="checkbox"/>	N/A	1/29 Wednesday 4-4:45pm	B	Link	Recording Password: JrtV3YFp	Facilitator: Meira Notetaker: Kathy Other(s): Spencer
7	[REDACTED]	[REDACTED]	Bluepages LinkedIn	Arlington, VA	<input checked="" type="checkbox"/>	N/A	1/29 Thursday 1-1:45pm	A	Link	Recording Password: wGkwjPb3	Facilitator: Spencer Notetaker: Kathy Other(s): N/A
8	[REDACTED]	[REDACTED]	Bluepages LinkedIn	Austin, TX Building 45	<input checked="" type="checkbox"/>	N/A	2/1 Friday 3:30-4:15pm	B	Link	Recording Password: rXAdu3fJ	Facilitator: Spencer Notetaker: Kathy Other(s): N/A

Testing

Facilitating user sessions

We then went on to facilitate eight user testing sessions.

Introduction (5 min)

- How's it going? We're with the IBM Cloud team here in Austin, TX [Introduce each person on the call]. We have a new feature with which we're hoping to do some usability testing.
- Before we begin, do you mind if we record?
- How long have you used terraform or infrastructure as code?
- After creating a workspace, what are the major tasks you have to do?
- For stuff that you're doing in prod, what's a normal number of variables in a workspace?
- And are you mostly filling those out manually?

Prototype A (15min)

Set up scenario

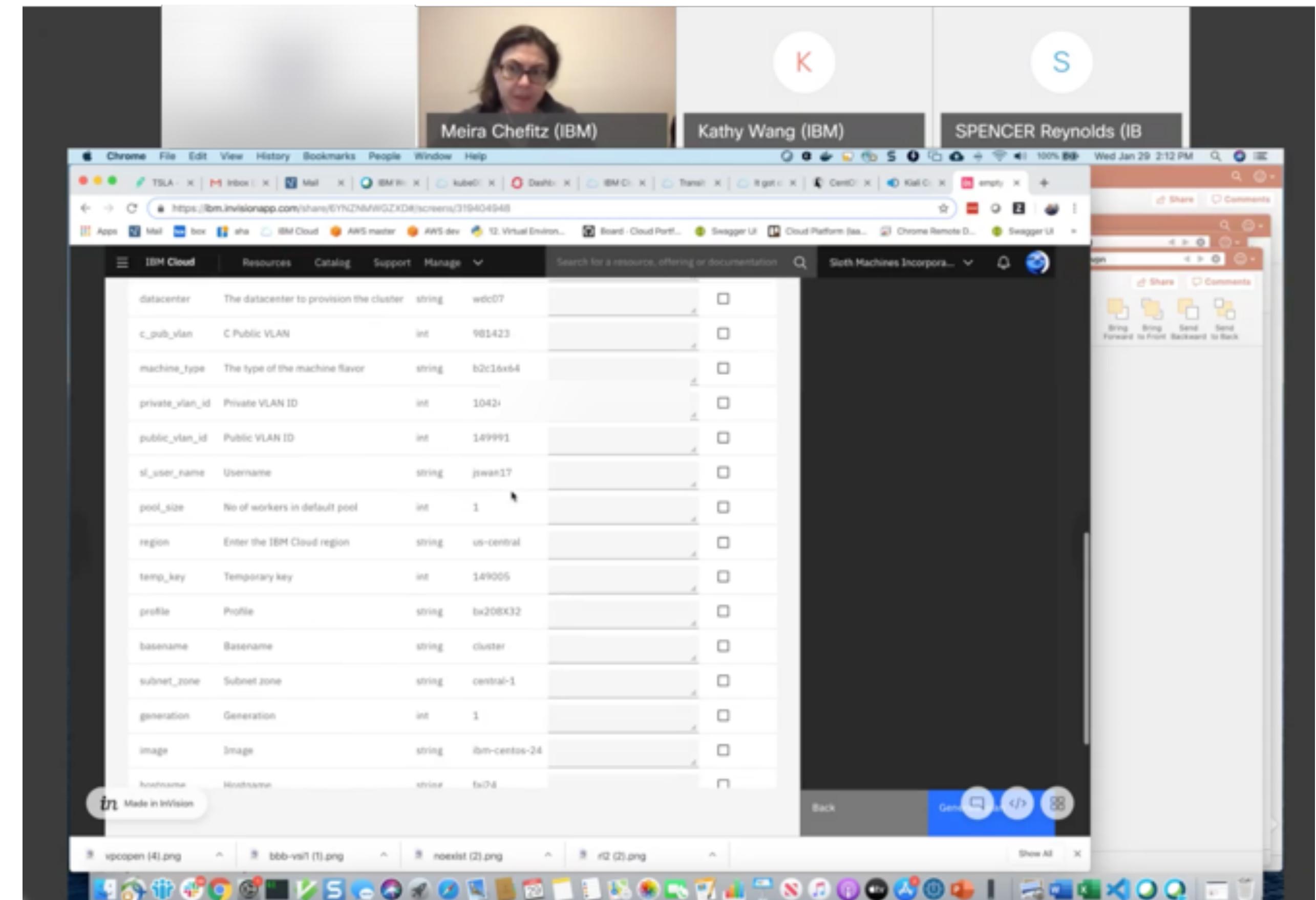
- Send link to autofill prototype.
 - https://ibm.invisionapp.com/share/NDNZNISTR3W#/319401610_schematics_Prepares_Workspace
- Could you share your screen?
- Read out message, and Slack it.
 - You're using Schematics to deploy your infrastructure as code for a **dev environment**, and your manager is telling you to use the override values he put in the Schematics tool.
 - It needs an **8x32 machine type** (32 gigabyte of ram machine type),
 - the **datacenter is in Dallas**, and
 - it's on a **basic plan**
- Based on your understanding of this task and service, on a scale for 1 to 7, with 1 being extremely difficult and 7 being extremely easy, how difficult or easy do you expect this task to be?

Complete task

-

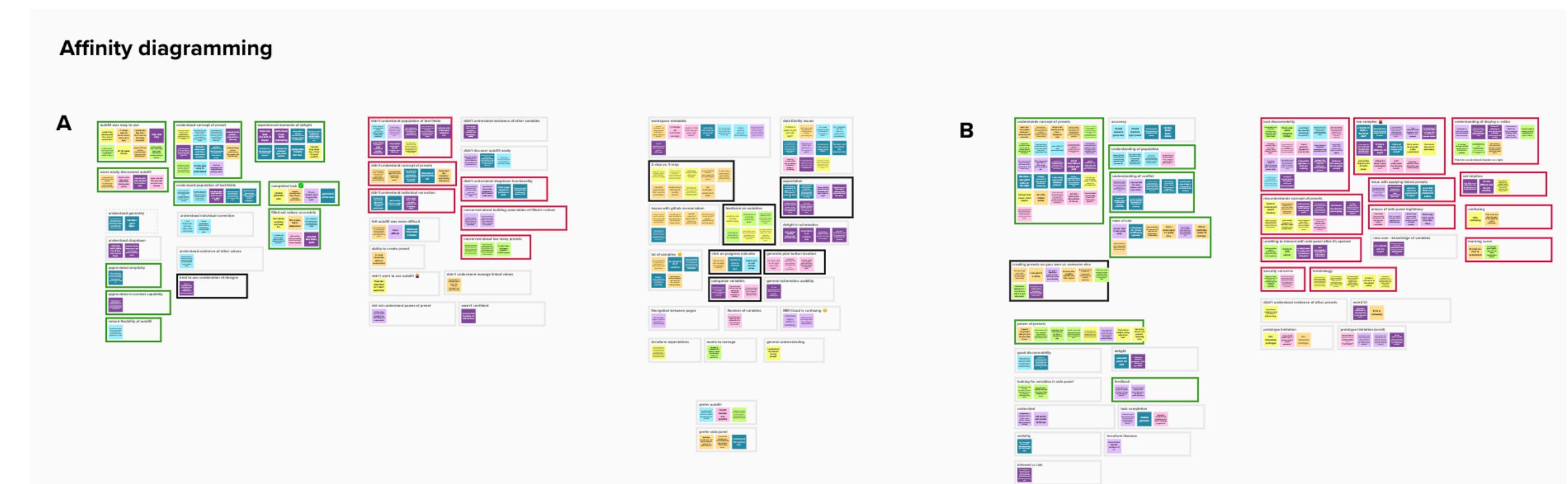
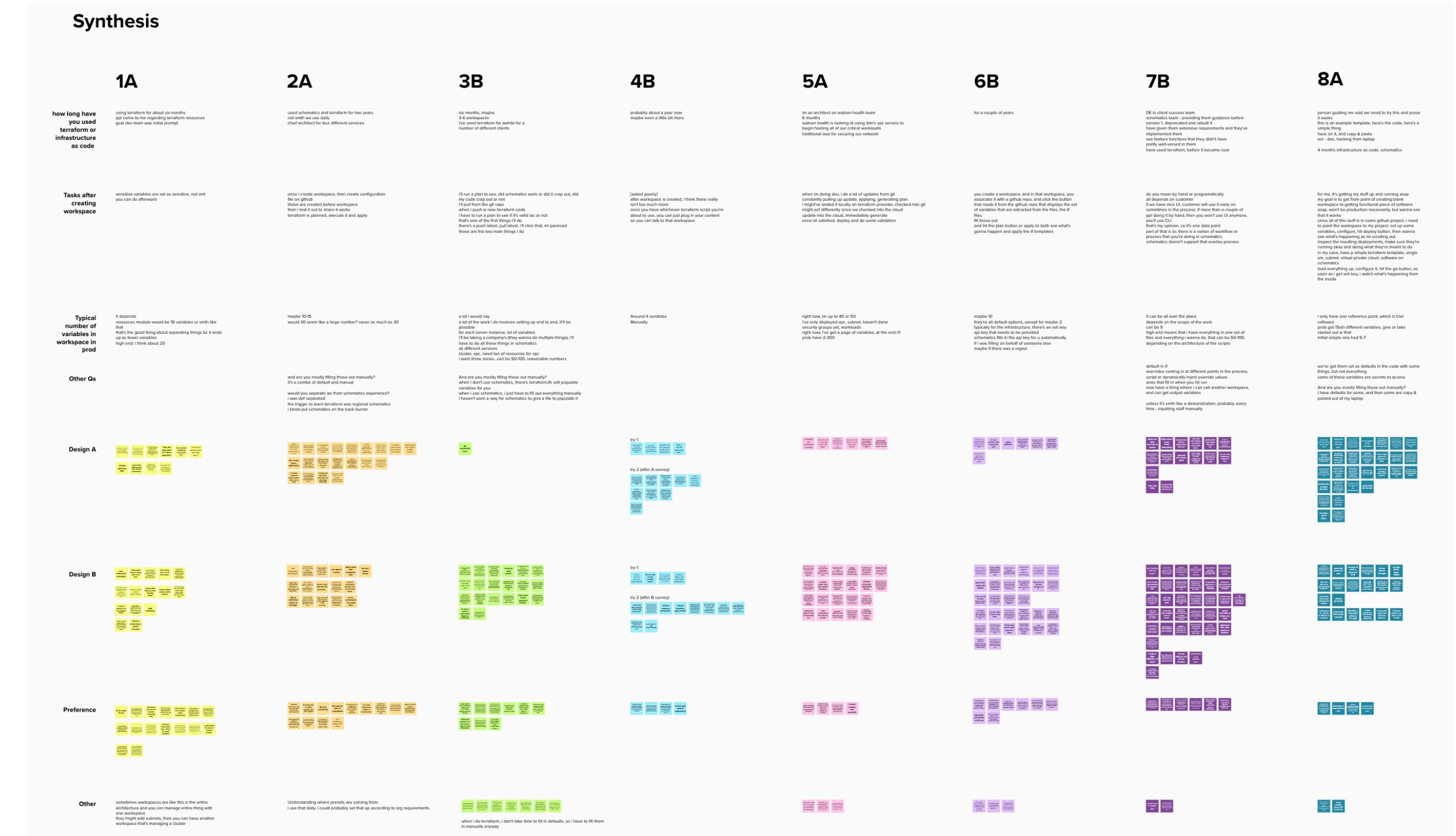
Reflect

- Based on that experience, on a scale of 1 to 7 with 1 being extremely difficulty and 7 being extremely easy, how difficult or easy was that task?
 - Why do you say that?
- Send link to survey
 - <https://www.surveygizmo.com/s3/5424974/Schematics-Test-Design-A>



Testing Synthesis

I compiled all of the interview notes into a Mural, where I could synthesize them.



Testing

Findings

From the synthesis, we found the following pros and cons.

Autofill

Pros

1. High discoverability: users easily discovered autofill
 2. High ease: autofill was easy to use
 3. High accuracy: users filled out values accurately
- Note: someone tried to use combination of designs

Cons

1. Low understanding: didn't understand the concept of presets
2. Low understanding: didn't expect or understand the population of other text fields
3. Low Understanding: didn't understand individual correction

Side panel

Pros

1. High understanding: understood concept of presets
 2. High understanding: understood conflict scenario
 3. High ease: felt it was easy to use - once they understood it / got past learning curve
- Note: people assumed you could create presets on your own, didn't assume someone else created them

Cons

1. Low discoverability: users never knew to click on the inline notification
 2. High complexity: users thought it was too complex (not intuitive, confusing, learning curve)- some didn't even want to interact with it even after it was opened
 3. Low trust: security concerns - unsure of side panel legitimacy
- Note: display v. editor

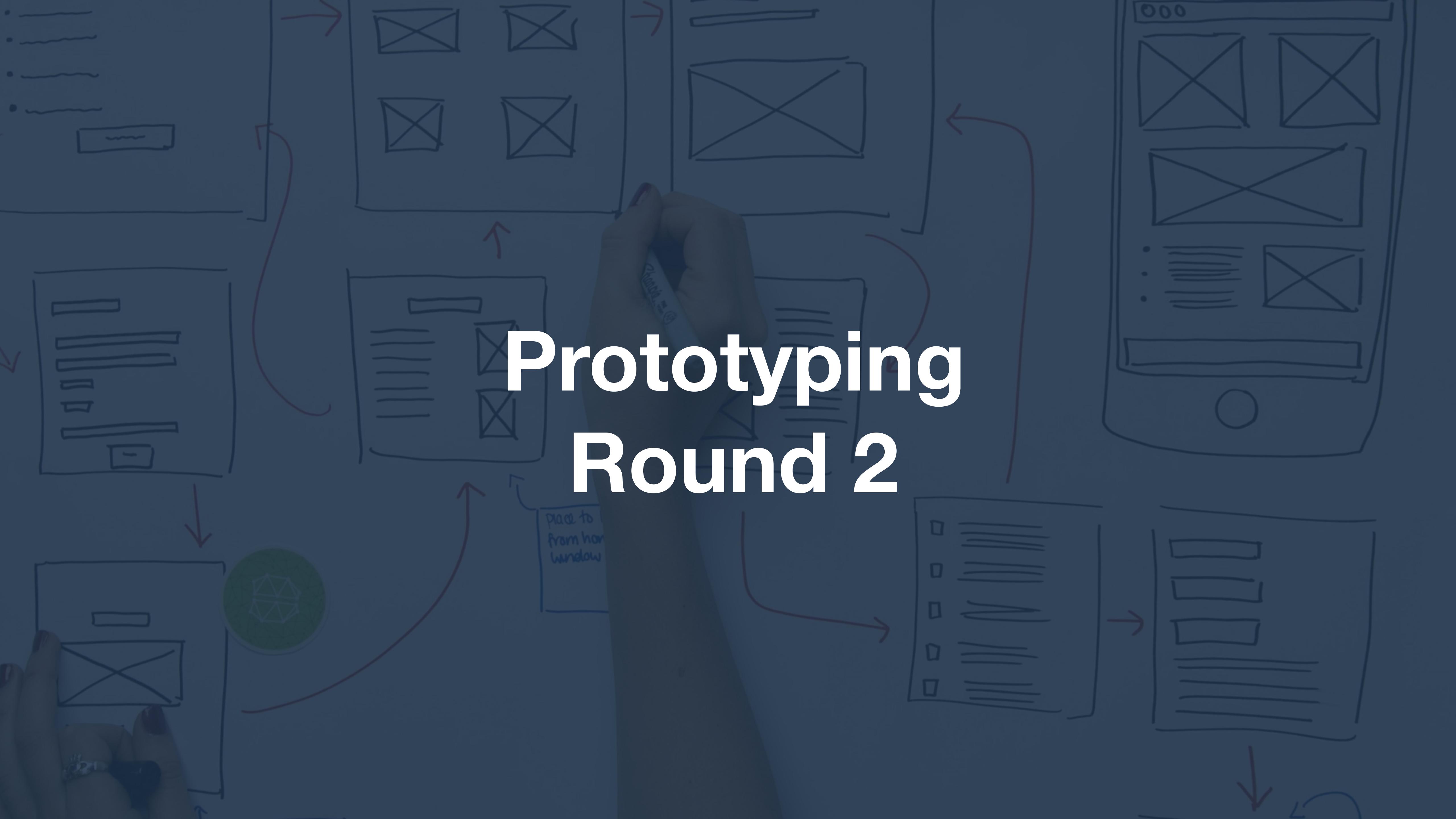
Findings

We found that the side panel is high in understanding and low in discoverability and ease of use. In contrast, autofill is high in discoverability and ease of use, but low in understanding.

Thus, there seems to be a trade-off between understanding and discoverability / ease of use.

How might we design for both understanding and discoverability / ease of use?
Or do we value one over the other?

Prototyping Round 2



Prototyping

Checklist

I created a checklist, in which I detailed our proposed design solutions.

Big issues

- could just have it filled out / pre-populated (solves the issue of user having to interact w presets in the first place)...we should get rid of defaults then... (mb not??)
 - and then do it one by one (sai)
- Make sure autofill suggestions don't cover up text input field (enable users to expect behavior of other text fields being populated)
 - could make suggestions smaller
 - or could give understanding via suggestions itself
- could buff it up, like credit card (enable understanding of the concept of presets)
 - have name of variables
 - +4 other values
 - modal: link other values
 - or side panel (could keep going to work on the page)
- advanced settings

Smaller issues

- categorize them (understand what fields have dropdown functionality, build association of filled in values)
- only show first three for autofill (allow users to juggle amongst a lot of presets)
- have broken link?? to let them know that this can be linked (...understand which fields have dropdown functionality?)
 - or just say "enter or select a namespace"
- create flow
 - design for it
- create preset flow: few things (metadata) are possibly missing)

General Schematics

- Carbon: Figure out whether or not the user can click on the progress indications (post on slack)
- Cloud Pal: The button should be at bottom. Take this up with CloudPAL (mock up proposal)
 - also, buttons on order summary will be deactivated until the right time (not ideal...)
 - OR just get rid of order summary buttons... (move button from order summary to bottom)
- Design: Give the user some **feedback** on whether it's mandatory to fill out variable, or if they correctly filled it out.
 - Make sure you present preset general schematics insights (just feedback tbh)

Designing for understanding

1. Understand the concept of
presets ➔ Buff up the
autofill

2. Understand that this preset
will fill the other values out
➔ Ensure that the autofill
suggestions don't cover up
the value text fields

The screenshot shows the 'Variables' section of the IBM Cloud Workspaces interface. The 'Variables' tab is selected. A dropdown menu is open over the 'resource_group' row, listing suggestions like 'dev', 'marketing', 'sales', and 'Manage saved variable values'. Other variables listed include 'kubeversion', 'plan', 'datacenter', etc.

Name	Description	Type	Value	Sensitive
kubeversion	Version of Kubernetes	int	Enter a kubeversion	<input type="checkbox"/>
resource_group	The resource group for provisioning	string	Enter or select a resource	<input type="checkbox"/>
plan	The type of plan	string	@ dev resource_group:dev plan:premium datacenter:wdc04	<input type="checkbox"/>
datacenter	The datacenter to provision the cluster	string	@ marketing plan:standard data center:syd05 + 3 other values	<input type="checkbox"/>
c_pub_vlan	C Public VLAN	int	@ sales plan:standard data center:lon04 + 2 other values	<input type="checkbox"/>
machine_type	The type of the machine flavor	string	@ Manage saved variable values	<input type="checkbox"/>
private_vlan_id	Private VLAN ID	int	jswan17	<input type="checkbox"/>
public_vlan_id	Public VLAN ID	int	1	<input type="checkbox"/>
sl_user_name	Username	string	us-central	<input type="checkbox"/>
pool_size	No of workers in default pool	int	149005	<input type="checkbox"/>
region	Enter the IBM Cloud region	string	bx208x32	<input type="checkbox"/>
temp_key	Temporary key	int	cluster	<input type="checkbox"/>
profile	Profile	string		<input type="checkbox"/>
basename	Basename	string		<input type="checkbox"/>

Prototyping

Designing for other issues

1. Build association of filled in values → Put presets all together in one spot
2. Understand which fields have dropdown functionality → Use combobox component
3. Juggle amongst a lot of presets → Only show three presets at a time

The screenshots illustrate the 'Variables' section of the IBM Cloud Workspaces interface. On the left, a list of variables is shown with dropdown menus for certain value fields. On the right, a dropdown menu is open, displaying a list of preset values categorized by environment (dev, marketing, sales) and a 'Manage saved variable values' option.

Name	Description	Type	Value	Sensitive
kubeversion	Version of Kubernetes	int	Enter a kubeversion	<input type="checkbox"/>
resource_group	The resource group for provisioning	string	Enter or select a resource	<input type="checkbox"/>
plan	The type of plan	string	Enter or select a plan	<input type="checkbox"/>
datacenter	The datacenter to provision the cluster	string	Enter or select a datacenter	<input type="checkbox"/>
c_pub_vlan	C Public VLAN	int	981423	<input type="checkbox"/>
machine_type	The type of the machine flavor	string	b2c16x4	<input type="checkbox"/>
private_vlan_id	Private VLAN ID	int	104242	<input type="checkbox"/>
public_vlan_id	Public VLAN ID	int	149991	<input type="checkbox"/>
sl_user_name	Username	string	jswan17	<input type="checkbox"/>
pool_size	No of workers in default pool	int	1	<input type="checkbox"/>
region	Enter the IBM Cloud region	string	us-central	<input type="checkbox"/>
temp_key	Temporary key	int	149005	<input type="checkbox"/>
profile	Profile	string	bx208x32	<input type="checkbox"/>
basename	Basename	string	cluster	<input type="checkbox"/>

Prototyping Create flow

The screenshots illustrate the process of creating a new workspace in IBM Cloud. The workspace is named 'myworkspace01234'.

Screenshot 1: Initial Workspace Creation Page

- Variables Section:**
 - kubeversion:** Version of Kubernetes (Type: int, Value: Enter a kubeversion, Sensitive:
 - resource_group:** The resource group for provisioning (Type: string, Value: Enter or select a resource, Sensitive:
 - plan:** The type of plan (Type: string, Value: Enter or select a plan, Sensitive:
 - datacenter:** The datacenter to provision the cluster (Type: string, Value: Enter or select a datacenter, Sensitive:
 - c_pub_vlan:** C Public VLAN (Type: int, Value: 981423, Sensitive:
 - machine_type:** The type of the machine flavor (Type: string, Value: b2c16x64, Sensitive:
 - private_vlan_id:** Private VLAN ID (Type: int, Value: 104242, Sensitive:
 - public_vlan_id:** Public VLAN ID (Type: int, Value: 149991, Sensitive:
 - sl_user_name:** Username (Type: string, Value: jswan17, Sensitive:
 - pool_size:** No of workers in default pool (Type: int, Value: 1, Sensitive:
 - region:** Enter the IBM Cloud region (Type: string, Value: us-central, Sensitive:
 - temp_key:** Temporary key (Type: int, Value: 149005, Sensitive:
 - profile:** Profile (Type: string, Value: bx208x32, Sensitive:
 - basename:** Basename (Type: string, Value: cluster, Sensitive:

Screenshot 2: Variables Section with Resource Selection

- Variables Section:**
 - resource_group:** The resource group for provisioning (Type: string, Value: Enter or select a resource, Sensitive:
 - plan:** The type of plan (Type: string, Value: Enter a plan, Sensitive:
 - datacenter:** The datacenter to provision the cluster (Type: string, Value: Enter a datacenter, Sensitive:
 - c_pub_vlan:** C Public VLAN (Type: int, Value: Enter a VLAN ID, Sensitive:
 - machine_type:** The type of the machine flavor (Type: string, Value: Enter a machine type, Sensitive:
 - private_vlan_id:** Private VLAN ID (Type: int, Value: Enter a VLAN ID, Sensitive:
 - public_vlan_id:** Public VLAN ID (Type: int, Value: Enter a VLAN ID, Sensitive:
 - sl_user_name:** Username (Type: string, Value: Enter a username, Sensitive:
 - pool_size:** No of workers in default pool (Type: int, Value: Enter a pool size, Sensitive:
 - region:** Enter the IBM Cloud region (Type: string, Value: Enter a region, Sensitive:
 - temp_key:** Temporary key (Type: int, Value: Enter a temporary key, Sensitive:
 - profile:** Profile (Type: string, Value: Enter a profile, Sensitive:
 - basename:** Basename (Type: string, Value: Enter a basename, Sensitive:

Screenshot 3: Save Variables and Values Dialog

- A modal dialog titled 'Save variables and values' is displayed.
- The dialog contains the following text: 'Would you like to save these values to use again in the future?' and 'You can view and remove saved values in Variable Management settings.'
- Two buttons are present: 'Not now' and 'Save'.
- The 'Save' button is highlighted with a blue background.



Thank You!