VA.gov Testing

Discovery Sprint

James Chasia, Naomi Marcussen, Keifer Furzland, Peter Hill // **Nov 26, 2019**

DIGITAL SERVICE at VA

Sprint Objectives

Define a holistic, scalable solution for the challenges teams face performing tests and demos on VA.gov.

How might we enable:

- VSP and VFS teams to QA all product path scenarios
- Call center reps to easily replicate user issues
- VFS teams to demo their apps

Content Overview

1 Discovery Sprint Team

2 Who We’ve Talked to

3 What We Did

4 What We Learned

5 Next Steps/ Recommendations

…

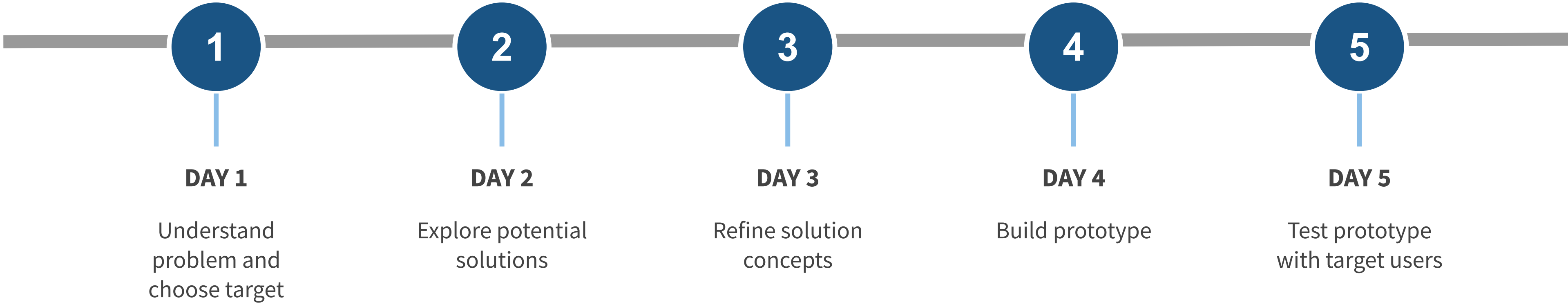
1. DISCOVERY SPRINT TEAM

- James Chasia – Product
- Naomi Marcussen – Design
- Keifer Furzland – BE Engineer
- Peter Hill – QA Engineer

2. WHO WE TALKED TO

- Trevor Pierce - FE Engineer [VSP/508]
- Erik Hansen - FE Engineer [VSA/Auth’d Exp.]
- Chante Lantos-Swett - [Call Center Lead]
- Tze-chiu Lei - QA Engineer [VSA/Support]
- Kevin Hoffman - [DEPO/Benefits]
- Bill Ryan - Operations [VSP/Tools]
- Anna Carey - BE Engineer [VSP/Tools]
- Lindsey Hattamer - BE Engineer [VSP/Tools]

3. WHAT WE DID



1

DAY 1

Understand problem and choose target

2

DAY 2

Explore potential solutions

3

DAY 3

Refine solution concepts

4

DAY 4

Build prototype

5

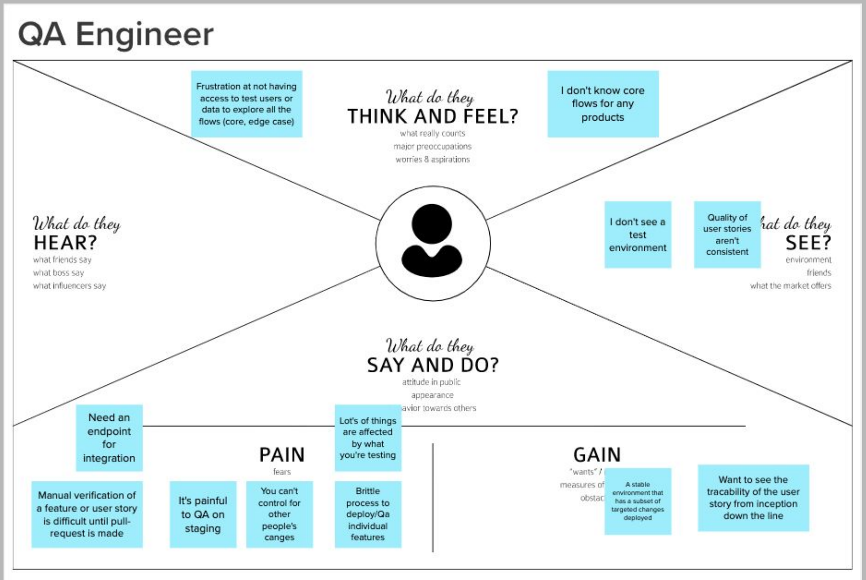
DAY 5

Test prototype with target users

DAY 1

Understand the problem and choose a target

Empathy Maps



QA Engineer

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

Frustration at not having access to test users or data to explore all the flows (core, edge case)

I don't know core flows for any products

What do they HEAR?
what friends say
what boss say
what influencers say

What do they SEE?
environment
friends
what the market offers

I don't see a test environment

Quality of user stories aren't consistent

What do they SAY AND DO?
attitude in public
appearance
avior towards others

PAIN
fears

Need an endpoint for integration

Lot's of things are affected by what you're testing

Manual verification of a feature or user story is difficult until pull-request is made

It's painful to QA on staging

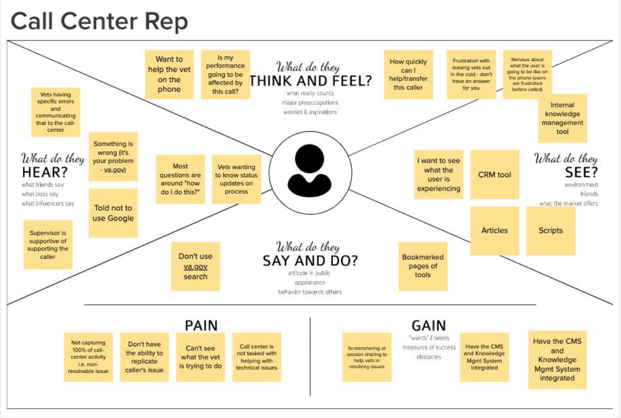
You can't control for other people's canges

Brittle process to deploy/Qa individual features

GAIN
"wants" /
measures of
obstac

A stable environment that has a subset of targeted changes deployed

Want to see the tracability of the user story from inception down the line



Call Center Rep

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

Want to help the vet on the phone

Is my performance going to be affected by this call?

How quickly can I help/transfer this caller

Frustration with leaving vets out in the cold - don't have an answer for you

Nervous about what the user is going to be like on the phone (users are frustrated before called)

What do they HEAR?
what friends say
what boss say
what influencers say

Vets having specific errors and communicating that to the call-center

Something is wrong (it's your problem - va.gov)

Told not to use Google

Supervisor is supportive of supporting the caller

Most questions are around "how do I do this?"

Vets wanting to know status updates on process

What do they SEE?
environment
friends
what the market offers

Internal knowledge management tool

I want to see what the user is experiencing

CRM tool

Articles

Scripts

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

Don't use va.gov search

Bookmarked pages of tools

PAIN

Not capturing 100% of call-center activity i.e. non-resolvable issue

Don't have the ability to replicate caller's issue

Can't see what the vet is trying to do

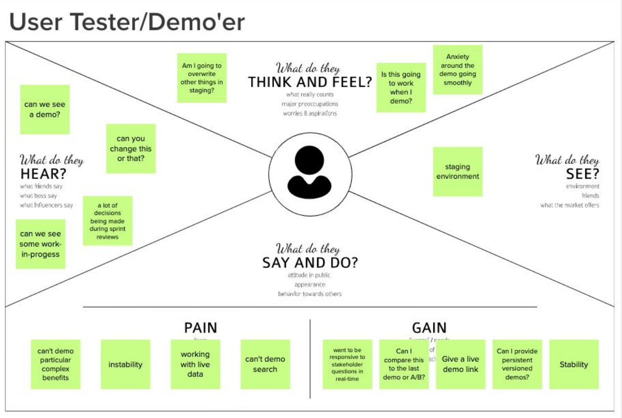
Call center is not tasked with helping with technical issues

GAIN
"wants" / needs
measures of success
obstacles

Screensharing or session sharing to help vets in resolving issues

Have the CMS and Knowledge Mgmt System integrated

Have the CMS and Knowledge Mgmt System integrated



User Tester/Demo'er

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

Am I going to overwrite other things in staging?

Is this going to work when I demo?

Anxiety around the demo going smoothly

What do they HEAR?
what friends say
what boss say
what Influencers say

can we see a demo?

can you change this or that?

a lot of decisions being made during sprint reviews

can we see some work-in-progess

What do they SEE?
environment
friends
what the market offers

staging environment

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

PAIN

can't demo particular complex benefits

instability

working with live data

can't demo search

GAIN

want to be responsive to stakeholder questions in real-time

Can I compare this to the last demo or A/B?

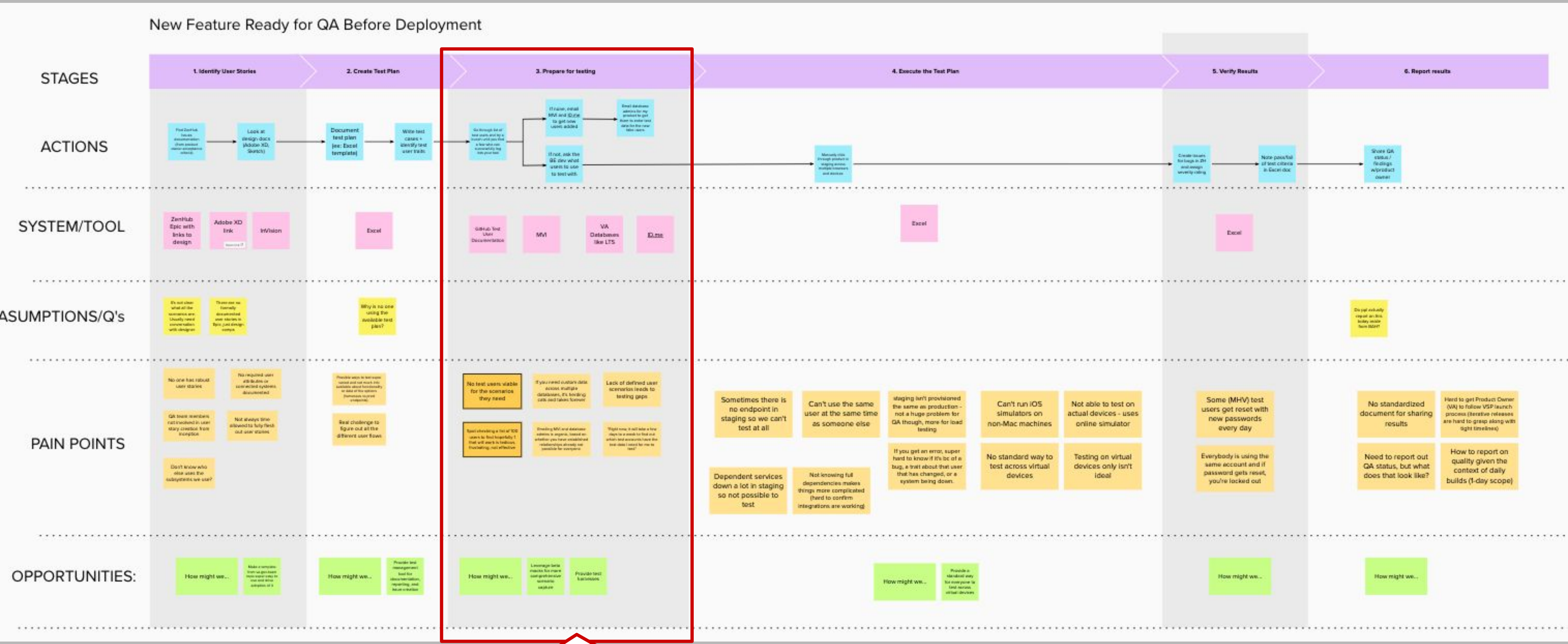
Give a live demo link

Can I provide persistent versioned demos?

Stability

Empathy Maps

Mapping the QA Journey


New Feature Ready for QA Before Deployment
STAGES
1. Identify User Stories
2. Create Test Plan
3. Prepare for testing
4. Execute the Test Plan
5. Verify Results
6. Report results
ACTIONS
Look at design docs (Adobe XD, Sketch)
Document test plan (ex: Excel template)
Write test cases + identify test user traits
If none, email MVI and ID.me to get new users added
If not, ask the BE dev what users to use to test with
Create issues for bugs in ZH and assign severity rating
Note pass/fail of test criteria in Excel doc
Share QA status / findings w/product owner
SYSTEM/TOOL
ZenHub Epic with links to design
Adobe XD link
InVision
Excel
GitHub Test User Documentation
MVI
VA Databases like LTS
ID.me
Excel
Excel
ASUMPTIONS/Q's
Why is no one using the available test plan?
PAIN POINTS
No one has robust user stories
No required user attributes or connected systems documented
QA team members not involved in user story creation from inception
Not always time allowed to fully flesh out user stories
Don't know who else uses the subsystems we use?
Real challenge to figure out all the different user flows
No test users viable for the scenarios they need
If you need custom data across multiple databases, it's herding cats and takes forever
Lack of defined user scenarios leads to testing gaps
Sometimes there is no endpoint in staging so we can't test at all
Can't use the same user at the same time as someone else
staging isn't provisioned the same as production - not a huge problem for QA though, more for load testing
Can't run iOS simulators on non-Mac machines
Not able to test on actual devices - uses online simulator
If you get an error, super hard to know if it's bc of a bug, a trait about that user that has changed, or a system being down.
No standard way to test across virtual devices
Testing on virtual devices only isn't ideal
Dependent services down a lot in staging so not possible to test
Not knowing full dependencies makes things more complicated (hard to confirm integrations are working)
Some (MHV) test users get reset with new passwords every day
Everybody is using the same account and if password gets reset, you're locked out
No standardized document for sharing results
Hard to get Product Owner (VA) to follow VSP launch process (iterative releases are hard to grasp along with tight timelines)
Need to report out QA status, but what does that look like?
How to report on quality given the context of daily builds (1-day scope)
OPPORTUNITIES:
How might we...
How might we...
How might we...
Provide test harnesses
How might we...
Provide a standard way for everyone to test across virtual devices
How might we...
How might we...

We chose to target the most painful part of the experience for QA

Prepare for Testing

QA User Journey Map

Call Center User Journey Map

Pain Points

- Spot checking a list of 100 users to find hopefully 1 that will work is tedious, frustrating, not effective
- There are no viable test users for the scenarios they need to test
- Can't use the same user at the same time as someone else
- Need to find someone who has a personal vet account to log in to help replicate user issues
- Not knowing which test accounts had data for which applications in staging

See all pain points

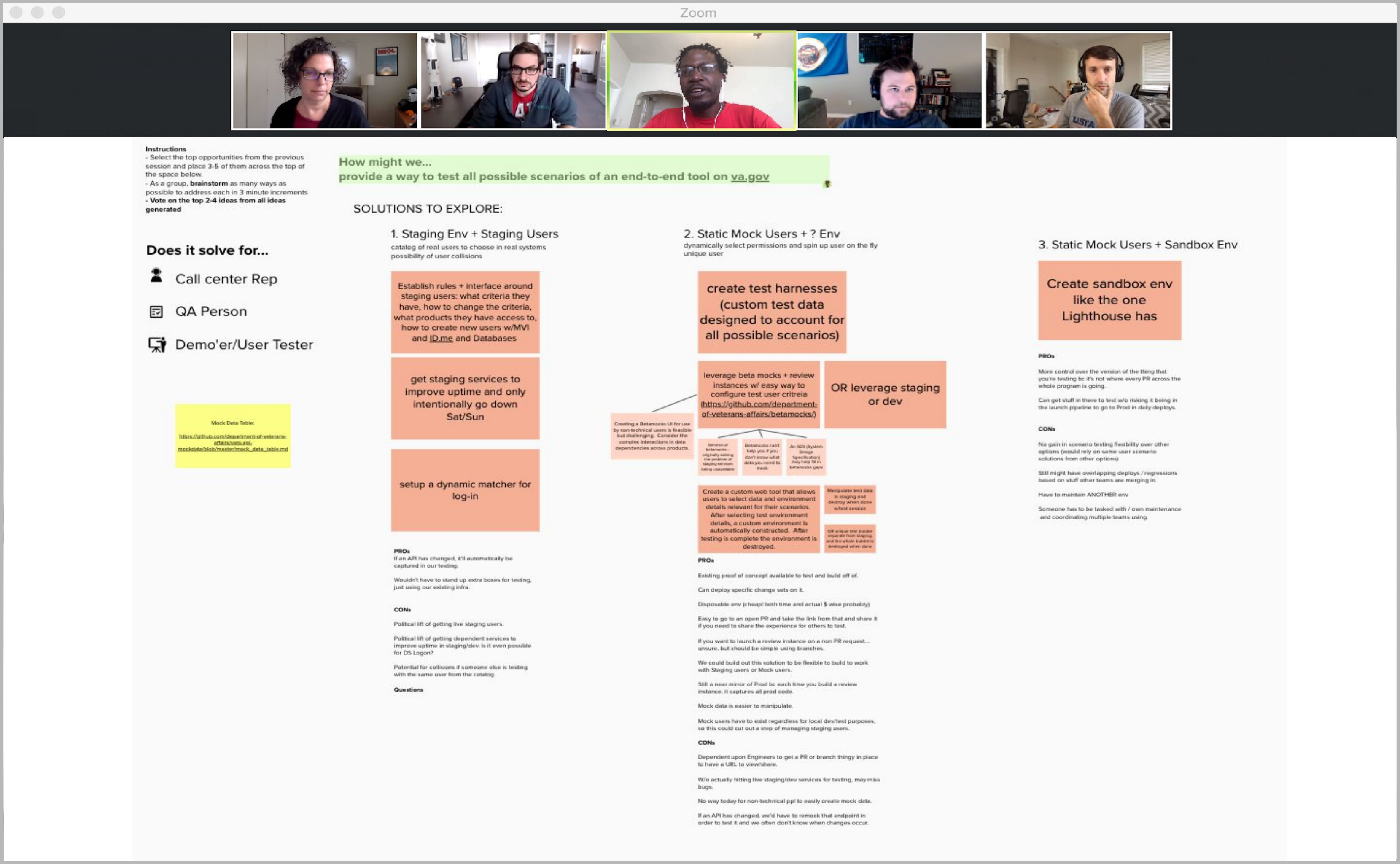
How might we..

provide a way to test all possible scenarios of an end-to-end tool on va.gov

DAY 2

Explore potential solutions

Solution Planning



Instructions
- Select the top opportunities from the previous session and place 3-5 of them across the top of the space below.
- As a group, **brainstorm** as many ways as possible to address each in 3 minute increments
- Vote on the top 2-4 ideas from all ideas generated

Does it solve for...

- Call center Rep
- QA Person
- Demo'er/User Tester

Mock Data Table:

https://github.com/department-of-veterans-affairs/vets-api-mockdata/blob/master/mock_data_table.md

How might we...
provide a way to test all possible scenarios of an end-to-end tool on va.gov

SOLUTIONS TO EXPLORE:

1. Staging Env + Staging Users

catalog of real users to choose in real systems
possibility of user collisions

Establish rules + interface around staging users: what criteria they have, how to change the criteria, what products they have access to, how to create new users w/MVI and ID.me and Databases

get staging services to improve uptime and only intentionally go down Sat/Sun

setup a dynamic matcher for log-in

PROs
If an API has changed, it'll automatically be captured in our testing.

Wouldn't have to stand up extra boxes for testing, just using our existing infra.

CONs

Political lift of getting live staging users.

Political lift of getting dependent services to improve uptime in staging/dev. Is it even possible for DS Logon?

Potential for collisions if someone else is testing with the same user from the catalog

Questions

2. Static Mock Users + ? Env

dynamically select permissions and spin up user on the fly
unique user

create test harnesses (custom test data designed to account for all possible scenarios)

leverage beta mocks + review instances w/ easy way to configure test user critreia (https://github.com/department-of-veterans-affairs/betamocks/)

OR leverage staging or dev

Creating a Betamocks UI for use by non-technical users is feasible but challenging. Consider the complex interactions in data dependencies across products.

Genesis of betamocks - originally solving the problem of staging services being unavailable

Betamocks can't help you if you don't know what data you need to mock

An SDS (System Design Specification) may help fill in betamocks gaps

Create a custom web tool that allows users to select data and environment details relevant for their scenarios. After selecting test environment details, a custom environment is automatically constructed. After testing is complete the environment is destroyed.

Manipulate test data in staging and destroy when done w/test session

OR unique test bubble separate from staging, and the whole bubble is destroyed when done

PROs

Existing proof of concept available to test and build off of.

Can deploy specific change sets on it.

Disposable env (cheap! both time and actual $ wise probably)

Easy to go to an open PR and take the link from that and share it if you need to share the experience for others to test.

If you want to launch a review instance on a non PR request... unsure, but should be simple using branches.

We could build out this solution to be flexible to build to work with Staging users or Mock users.

Still a near mirror of Prod bc each time you build a review instance, it captures all prod code.

Mock data is easier to manipulate.

Mock users have to exist regardless for local dev/test purposes, so this could cut out a step of managing staging users.

CONs

Dependent upon Engineers to get a PR or branch thingy in place to have a URL to view/share.

W/o actually hitting live staging/dev services for testing, may miss bugs.

No way today for non-technical ppl to easily create mock data.

If an API has changed, we'd have to remock that endpoint in order to test it and we often don't know when changes occur.

3. Static Mock Users + Sandbox Env

Create sandbox env like the one Lighthouse has

PROs

More control over the version of the thing that you're testing bc it's not where every PR across the whole program is going.

Can get stuff in there to test w/o risking it being in the launch pipeline to go to Prod in daily deploys.

CONs

No gain in scenario testing flexibility over other options (would rely on same user scenario solutions from other options)

Still might have overlapping deploys / regressions based on stuff other teams are merging in.

Have to maintain ANOTHER env

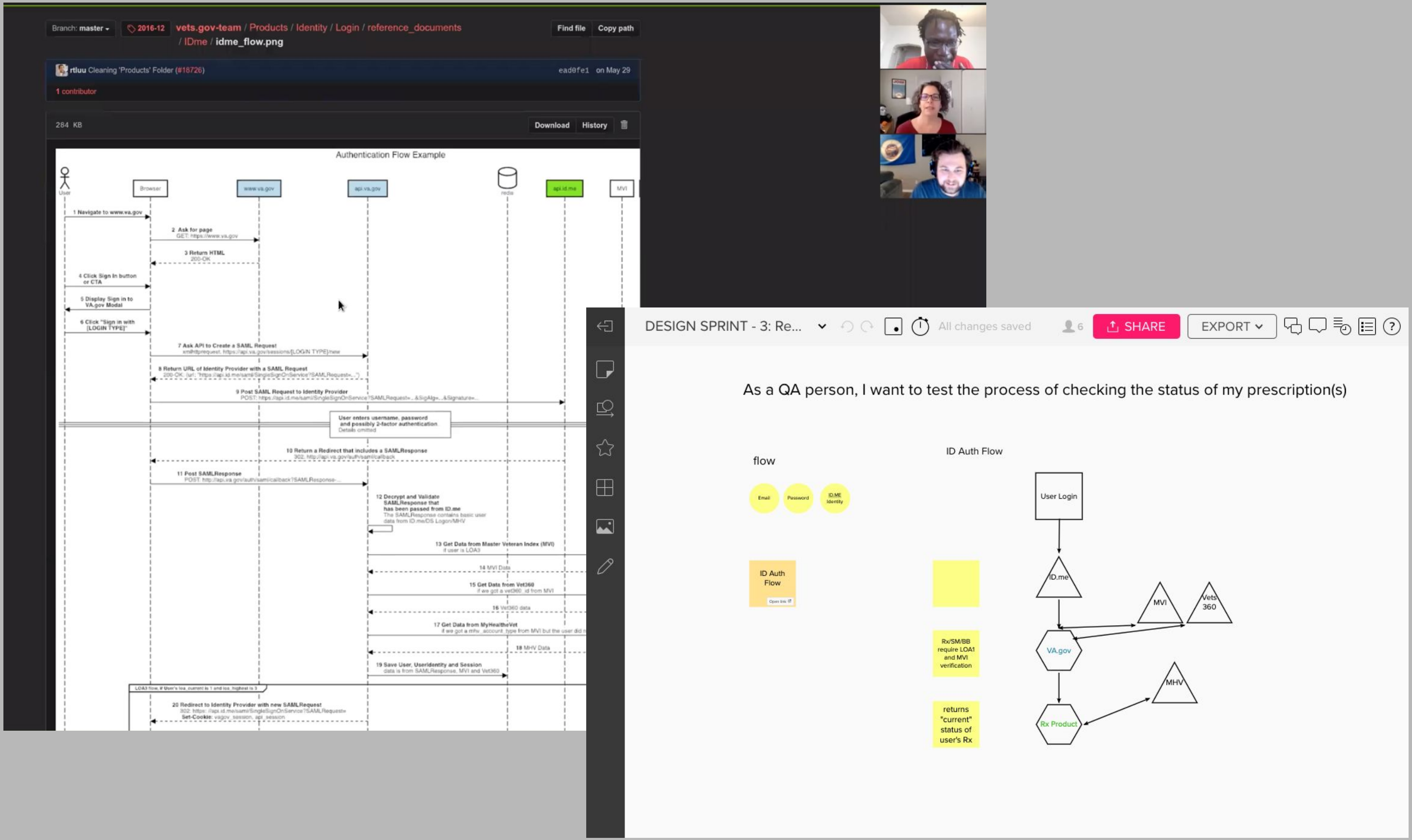
Someone has to be tasked with / own maintenance and coordinating multiple teams using.

Mural Board

DAY 3

Decide on the best solution

Decision Process



Branch: master ▾ 2016-12 vets.gov-team / Products / Identity / Login / reference_documents / IDme / idme_flow.png

rtluu Cleaning 'Products' Folder (#18726) ead0fe1 on May 29

1 contributor

284 KB

Authentication Flow Example

1 Navigate to www.va.gov

2 Ask for page
GET: https://www.va.gov

3 Return HTML
200-OK

4 Click Sign In button or CTA

5 Display Sign in to VA.gov Modal

6 Click "Sign in with [LOGIN TYPE]"

7 Ask API to Create a SAML Request
xmlhttprequest. https://api.va.gov/sessions/[LOGIN TYPE]/new

8 Return URL of Identity Provider with a SAML Request
200-OK: {url: "https://api.id.me/saml/SingleSignOnService?SAMLRequest=..."}

9 Post SAML Request to Identity Provider
POST: https://api.id.me/saml/SingleSignOnService?SAMLRequest=...&SigAlg=...&Signature=...

User enters username, password and possibly 2-factor authentication
Details omitted

10 Return a Redirect that includes a SAMLResponse
302: http://api.va.gov/auth/saml/callback

11 Post SAMLResponse
POST: http://api.va.gov/auth/saml/callback?SAMLResponse=...

12 Decrypt and Validate SAMLResponse that has been passed from ID.me
The SAMLResponse contains basic user data from ID.me/DS Logon/MHV

13 Get Data from Master Veteran Index (MVI)
if user is LOA3

14 MVI Data

15 Get Data from Vet360
if we got a vet360_id from MVI

16 Vet360 data

17 Get Data from MyHealtheVet
if we got a mhv_account_type from MVI but the user did n

18 MHV Data

19 Save User, UserIdentity and Session
data is from SAMLResponse, MVI and Vet360

LOA3 flow, if User's loa_current is 1 and loa_highest is 3

20 Redirect to Identity Provider with new SAMLRequest
302: https://api.id.me/saml/SingleSignOnService?SAMLRequest=
Set-Cookie: vagov_session, api_session

DESIGN SPRINT - 3: Re... All changes saved 6 SHARE EXPORT

As a QA person, I want to test the process of checking the status of my prescription(s)

Rx/SM/BB require LOA1 and MVI verification

returns "current" status of user's Rx

Day 3

Explore potential solutions

OPTION 1

Catalog Existing Users on staging

PROs

- Hitting live partner external APIs gives visibility into changes
  - No maintenance of mock endpoints
- Wouldn't have to stand up extra infra for testing or demos
- User data already exists, we're making better use of it
- **Low risk, high usability for future efforts**

CONs

- Political lift of creating/modifying live test users could limit our available user pool
- Political lift of getting dependent services to improve/maintain uptime in staging.
- Potential for data collision if testing with the same user

OPTION 2

Build Users Dynamically on staging

PROs

- Less reliance on external APIs for testing
- Could be made flexible enough to work w/ live *or* mock users
- Full control over the mock user data flow

CONs

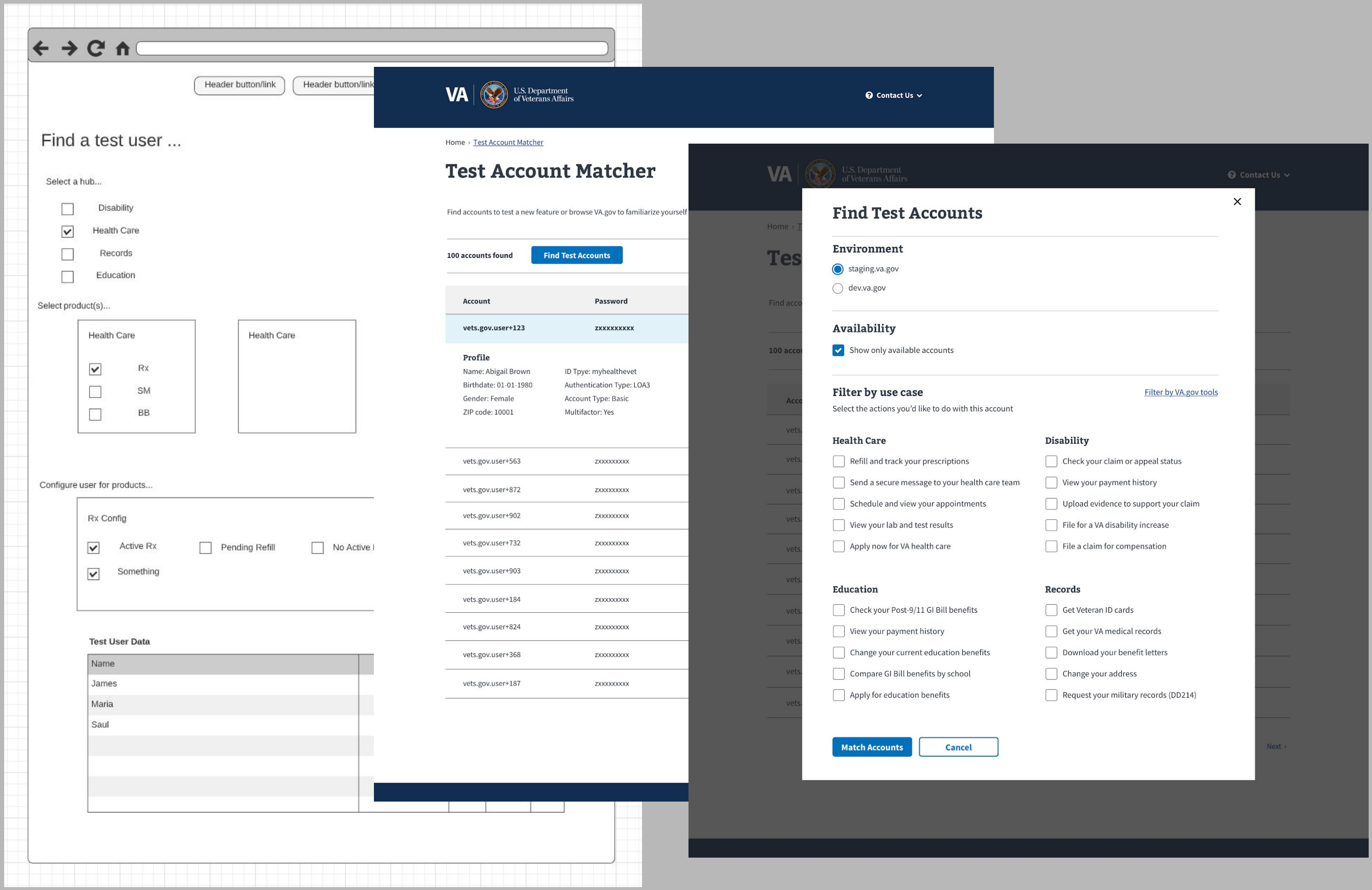
- **Much more lift to implement mock servers**
- W/o actually hitting live staging/dev services for testing, may miss bugs
- If an API has changed, we'd have to remock that endpoint in order to test it and we often don't know when changes occur
- More infrastructure == more costs
- Still have to present users to Call Center, etc

See whiteboard

DAY 4

Build a prototype

Wireframe & Prototyping



Header button/link

Header button/link

Find a test user ...

Select a hub...

□ Disability
☑ Health Care
□ Records
□ Education

Select product(s)...

Health Care

☑ Rx
□ SM
□ BB

Health Care

Configure user for products...

Rx Config

☑ Active Rx □ Pending Refill □ No Active I
☑ Something

Test User Data

Name
James
Maria
Saul

VA | U.S. Department of Veterans Affairs

Contact Us

Home › Test Account Matcher

Test Account Matcher

Find accounts to test a new feature or browse VA.gov to familiarize yourself

100 accounts found **Find Test Accounts**

Account	Password
vets.gov.user+123	zxxxxxxxxx

Profile

Name: Abigail Brown
Birthdate: 01-01-1980
Gender: Female
ZIP code: 10001

ID Tpye: myhealthevet
Authentication Type: LOA3
Account Type: Basic
Multifactor: Yes

Account	Password
vets.gov.user+563	zxxxxxxxxx
vets.gov.user+872	zxxxxxxxxx
vets.gov.user+902	zxxxxxxxxx
vets.gov.user+732	zxxxxxxxxx
vets.gov.user+903	zxxxxxxxxx
vets.gov.user+184	zxxxxxxxxx
vets.gov.user+824	zxxxxxxxxx
vets.gov.user+368	zxxxxxxxxx
vets.gov.user+187	zxxxxxxxxx

Find Test Accounts

Environment

● staging.va.gov
○ dev.va.gov

Availability

☑ Show only available accounts

Filter by use case

Filter by VA.gov tools

Select the actions you'd like to do with this account

Health Care

□ Refill and track your prescriptions
□ Send a secure message to your health care team
□ Schedule and view your appointments
□ View your lab and test results
□ Apply now for VA health care

Disability

□ Check your claim or appeal status
□ View your payment history
□ Upload evidence to support your claim
□ File for a VA disability increase
□ File a claim for compensation

Education

□ Check your Post-9/11 GI Bill benefits
□ View your payment history
□ Change your current education benefits
□ Compare GI Bill benefits by school
□ Apply for education benefits

Records

□ Get Veteran ID cards
□ Get your VA medical records
□ Download your benefit letters
□ Change your address
□ Request your military records (DD214)

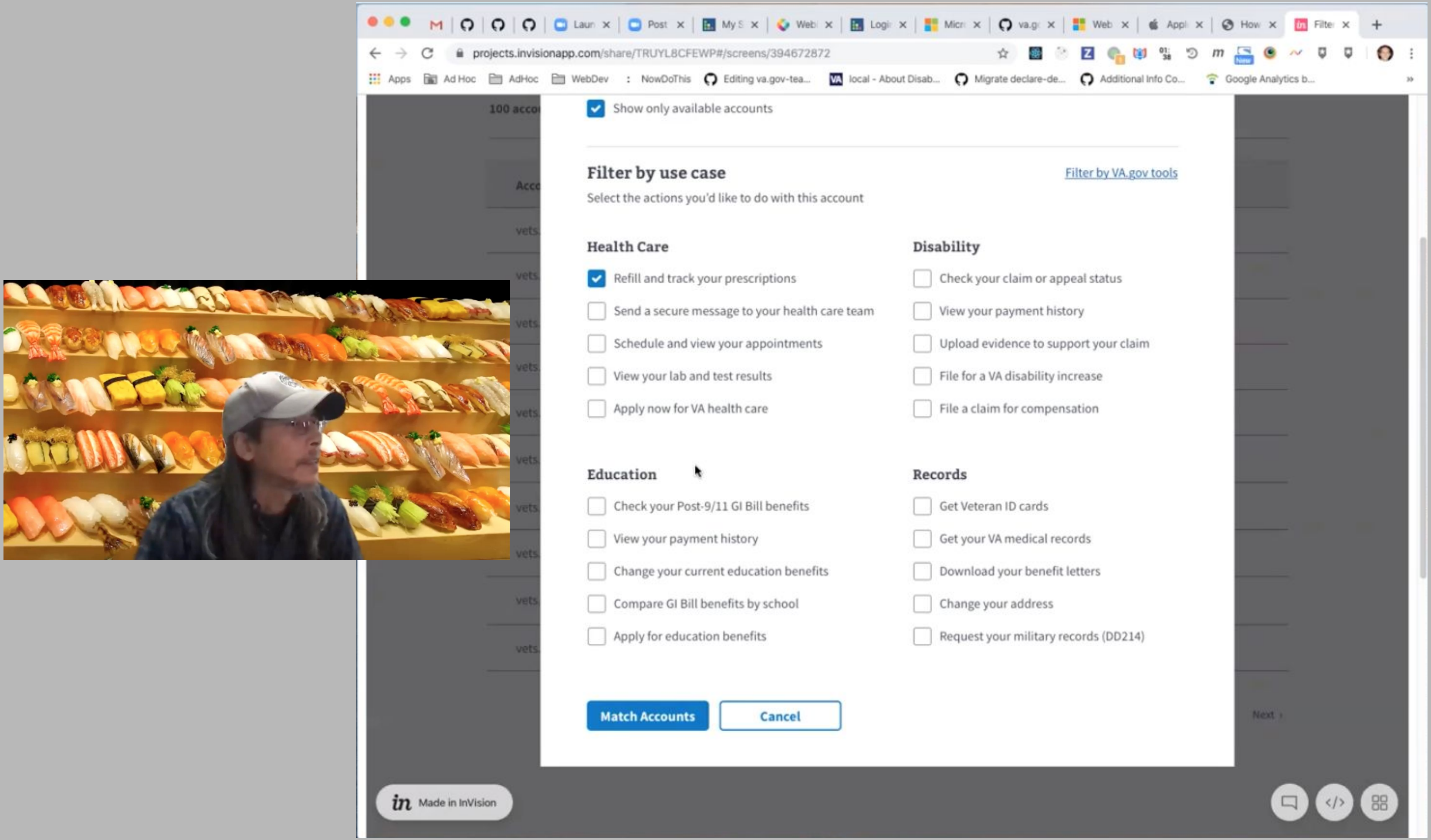
Match Accounts **Cancel**

Prototype

DAY 5

Test with users

Test with target users



☑ Show only available accounts

Filter by use case

Filter by VA.gov tools

Select the actions you'd like to do with this account

Health Care

☑ Refill and track your prescriptions
□ Send a secure message to your health care team
□ Schedule and view your appointments
□ View your lab and test results
□ Apply now for VA health care

Disability

□ Check your claim or appeal status
□ View your payment history
□ Upload evidence to support your claim
□ File for a VA disability increase
□ File a claim for compensation

Education

□ Check your Post-9/11 GI Bill benefits
□ View your payment history
□ Change your current education benefits
□ Compare GI Bill benefits by school
□ Apply for education benefits

Records

□ Get Veteran ID cards
□ Get your VA medical records
□ Download your benefit letters
□ Change your address
□ Request your military records (DD214)

Match Accounts | Cancel

Prototype Test Notes

4. WHAT WE LEARNED

Solving the pain point of finding test accounts will help the most users across use cases

- Allow VFS and VSP teams to QA their products
- Make it easier to find an account to demo with
- Allow Call Center Reps to familiarize themselves with the VA.gov experience

We will need to couple this tool with other initiatives to maximize improvements to the QA process

- Make the test plan template that encourages the use of this tool to define test users for each test scenario
- Establish guidelines for User Story documentation
- Tool for call center reps to replay user sessions
- Engineering use cases often require adding/modifying test users.

Prototype Tool

“ This log in button, this is amazing! ”

QA

“ This tool would be helpful during training. ”

Call Center

“ Right now, it will take a few days to a week to find out which test accounts have the test data I need for me to test. This would be an extremely useful time-saving tool. ”

QA

ITEMS FOR ROADMAP

Enhancements

- Refine filter UI through additional user testing
- Make “Availability” column clearer
- Add descriptions to test user attributes
- Filter by LOA

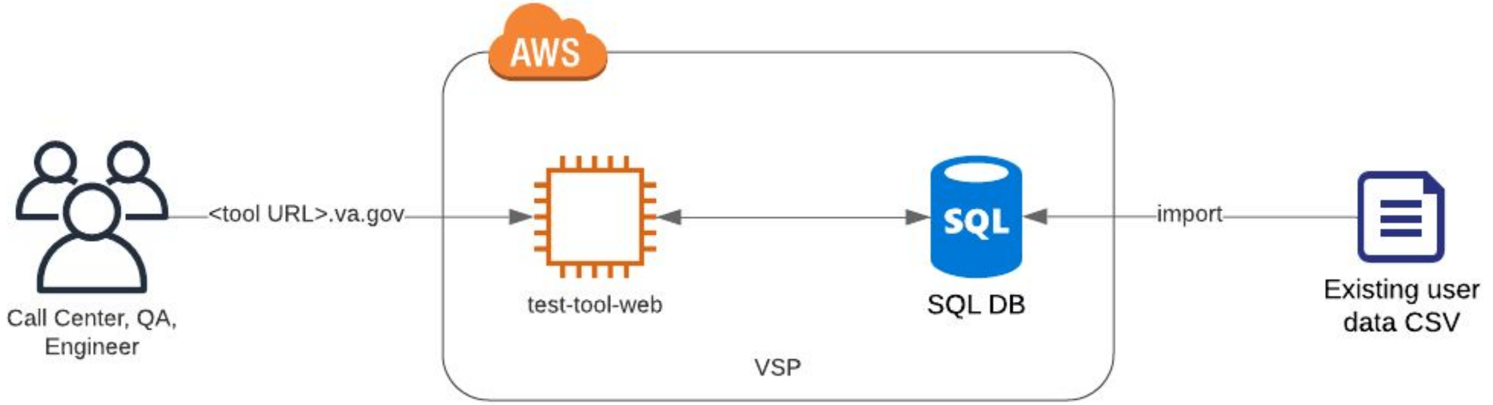
New features

- Enable keyword search queries
- Support deep linking
- Filter by login type
- Auto login
- Test user check-in/out system
- Features for adding and modifying test user accounts

5. RECOMMENDED NEXT STEPS FOR TOOL

MVP version

- Assess status of existing users on staging
- Map users to products.
- Import existing user data into a SQL database.
- Build UI leveraging DataTables library for powerful built-in filtering.



Thank you/Feedback

APPENDIX

Documents

- Survey
- Empathy Map
- As-Is-Experience Maps - QA
- As-Is-Experience Maps - Call Center Rep
- Discovery Sprint Findings

3. WHAT WE DID

Day 1: Understand problem and choose a target

- Survey
- Empathy and As-Is-Experience Maps
- Understand pain points
- “How might we…” statements

Day 2: Explore potential solutions

- Solution brainstorming and clustering
- Pros & Cons for each solution option

Day 3: Refine solution concepts

- Identify target use case for prototype - checking status of prescription refill
- Iterate on wireframes
- Plan prototype

Day 4: Build prototype

- Working UI for testing account selection tool
- Document back-end architecture proposal

Day 5: Test prototype with target users

4. WHAT WE LEARNED

Pain Points

From QA Engineers:

- “Spot checking a list of 100 users to find hopefully 1 that will work is tedious, frustrating, not effective”
- “No test users viable for the scenarios they need”
- “Emailing MVI and database admins is organic, based on whether you have established relationships already not possible for everyone.”
- “If you need custom data across multiple databases, it's herding cats and takes forever”
- “Can't use the same user at the same time as someone else”

From Call Center Lead:

- “Need to find someone who has a personal vet log in to help replicate user issues”

From Front End Engineers:

- Usually just picks test users from knowledge and experience
- “Currently test account availability is not a thing b/c people don't update the table.”

About Demos:

- “The main blocker was not knowing which test accounts had data for which applications in staging, or or having sufficient access to non VA.gov tools to DEMO something intended to be migrated to VA.gov.”