



VA Mobile App Discovery Sprint

# Week 3: Output + Outcomes

**Julia Elman**, UX Lead  
[julia@adhocteam.us](mailto:julia@adhocteam.us)

**Jonathan Julian**, Engineering Lead  
[jonathan@adhocteam.us](mailto:jonathan@adhocteam.us)

**Ambika Roos**, Product Lead  
[ambika@adhocteam.us](mailto:ambika@adhocteam.us)

March 20, 2020

# Agenda

- Where are we in the process?
- Week 3: Output + Outcomes
  - ◆ Technical Feasibility
  - ◆ Veteran Desirability
- Week 4: Looking ahead

**Where are we in the process?**

# Schedule

Week(s)	Guiding Question	Activities	Output	Outcome
0	What feature sets + frameworks should we evaluate from a technical feasibility and Veteran desirability perspective?	<ol style="list-style-type: none"> <li>Discussions with CTO's office</li> <li>Discussions with VA technical SMEs</li> </ol>	List of mobile app development frameworks and features to evaluate	Consensus on feature set + frameworks to be researched in coming weeks
1	What feature set + framework should we prototype?	<ol style="list-style-type: none"> <li>Evaluate feasibility of mobile frameworks using matrix</li> <li>Begin to research Veteran desirability of features using matrix</li> </ol>	<ul style="list-style-type: none"> <li>- Research findings</li> <li>- Recommendation on what to prototype</li> </ul>	Consensus on which technical framework and feature set to prototype
2 + 3	Is this prototype a viable option for the VA? Does it speak to Veteran needs?	<ol style="list-style-type: none"> <li>Technical experimentation</li> <li>User research and usability testing</li> </ol>	Technical and user research findings	Feedback and input on how to iterate on prototype
4	What are the pros/cons to different technical approaches and the impact of implementing them to the VA?	<ol style="list-style-type: none"> <li>Future proofing</li> <li>Synthesis</li> </ol>	Report and prioritized recommendations	Comparative analysis of technical frameworks and synthesis of Veteran desirability findings

# Technical Feasibility

# Schedule: Technical Feasibility

Planning		Week 1	Week 2	Week 3	Week 4
Frameworks	Guiding Questions	Activities		Output	Outcome
<ol style="list-style-type: none"><li>1. Fully native app (iOS + Android)</li><li>2. Cross-platform (React Native)</li><li>3. Cross-platform (Xamarin)</li><li>4. Cross-platform (NativeScript)</li><li>5. Hybrid (Ionic)</li><li>6. Hybrid (Ionic React)</li><li>7. Hybrid (Flutter)</li><li>8. PWA (Progressive Web App)</li></ol>	<ol style="list-style-type: none"><li>1. How much reuse is possible?</li><li>2. What would the investment be?</li><li>3. How does the UX change?</li></ol>	Technical analysis to develop initial hypothesis	Experimentation	Continued Experimentation and comparison	Recommendation for which framework is the most technically feasible

# What did we learn this week?

React Native doesn't allow for component re-use, but React skills are easy to source

## REUSE

- Can we reuse VA front-end components?
- What skills are required?

## UX

- How does “fit and finish” feel?
- Getting Facility Locator running
- Camera: doc upload functionality

## Questions

### Tooling

List everything you needed to install

Name	Time	Difficulty	Notes
Xcode	1hr	easy	download from App Store
Android Studio	1hr	easy	Download from <a href="https://developer.android.com/studio">https://developer.android.com/studio</a> , Install .dmg, Install Virtual Device (Pixel 3 M R), Install SD Build-Tools
Expo and related npm modules, latest Node	20m	easy	relatively straightforward

What were the challenges with tooling?

No major challenges. Android studio runs slowly on OSX / takes a lot of memory. iOS emulator large memory footprint but runs relatively quickly.

### What is the “reload/feedback loop” like?

relatively fast on iOS and web—3-5sec

### Did you notice any “clunkiness” with the UI?

UI responsive

### What specific skills did you use to be successful with React Native?

basic knowledge of iOS and understanding of iOS native and RN framework interaction, e.g. concepts

XCode set up

Read documentation

### How reusable are VA.gov React components?

1. no reuse at all
2. needed to rewrite it significantly
3. needed some edits
4. needed just a couple edits
5. worked out-of-the-box

Conclusion:

## Questions

### Tooling

List everything you needed to install

Name	Time	Difficulty	Notes
Xcode	30m	easy	download from App Store
Android Studio	45m	easy	Download from <a href="https://developer.android.com/studio">https://developer.android.com/studio</a> , Install Virtual Device (Pixel 3 Max, Android R), Install SD Build-Tools
Expo CLI	5m	easy	<code>npm i -g expo-cli</code>
Expo app on iOS	5m	easy	Download from App Store

What were the challenges with tooling?

I didn't run into any challenges with tooling. Running their “beginner” app showed several warnings in console but app was completely functional.

### What is the “reload/feedback loop” like?

1-2sec tested on iOS Expo app and on browser

### Did you notice any “clunkiness” with the UI?

None when testing on phone or in browser

### What specific skills did you use to be successful with React Native?

React. Although the styling and JSX are a bit different, having used React extensively makes RN easy to use.

### How reusable are VA.gov React components?

1. no reuse at all
2. needed to rewrite it significantly
3. needed some edits
4. needed just a couple edits
5. worked out-of-the-box

Explain: 1. no reuse at all

Styling and JSX components are different from web React. Each component from `formation-react`

## Questions

### Tooling

List everything you needed to install

Already had Xcode and Android Studio installed from the Ionic evaluation.

Name	Time	Difficulty	Notes
CocoaPods update	1.5 hours	medium	Received an error while running React Native command for starting a new project, <code>react-native init</code> , which would prevent me from building to an iOS emulator. Downgraded CocoaPods from 1.9.1 to 1.8.0 and ran <code>pod install</code> from <code>project-root/ios</code> to fix it, but not really any great help online.
Watchman	30 minutes	medium	Received a permissions error when installing via HomeBrew and had to run <code>chmod</code>
Expo CLI and app	15 minutes	easy	No errors, just make sure you're using a later version of Node during install

What were the challenges with tooling?

I first installed by following the first page of the docs, which presented this command - `npm react-native init MyTestApp`. I ran into some install difficulties but nothing too bad. I didn't have any luck Google searching my errors - I ended up getting lucky by downgrading a dependency and running `pod install` again.

As I proceeded in the docs I landed on the Getting Started page, which then offered two ways of running the app - through React Native CLI Quickstart or Expo CLI. This is a little confusing. At first, I ran the app in an iOS emulator but got a stacktrace. From this, I determined not having Watchman installed was the issue. After installing Watchman, I got the app running in the emulator okay.

I was curious about the Expo CLI, so I tried that next. I to make an account on expo.io, download the Expo app to my phone, install the expo-cli globally on my computer as a Node module, and sign in from my terminal too. I then created a new project using `expo init AwesomeProject`. I ran that project with the Expo app open on my phone. First, my computer's browser opened <http://localhost:19002/> with kind of a neat-looking debug page for the Expo stuff. On the Expo app on my phone, I saw the project listed as an app I can open. I opened it, and then it begins downloading and building the JavaScript from my computer. The app then begins running on my iPhone. Really cool. There's also a QR code on my localhost and in my terminal for devices to scan in order to start the app.

### What is the “reload/feedback loop” like?

Pretty much instant for just changing text, on the emulator and my phone.

Did you notice any “clunkiness” with the UI?

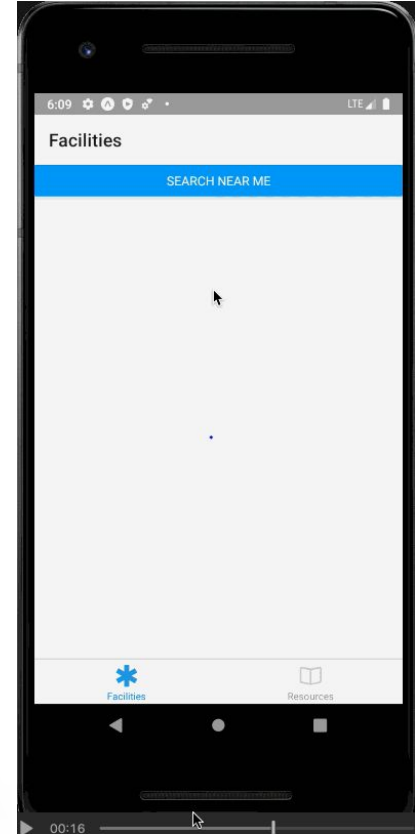
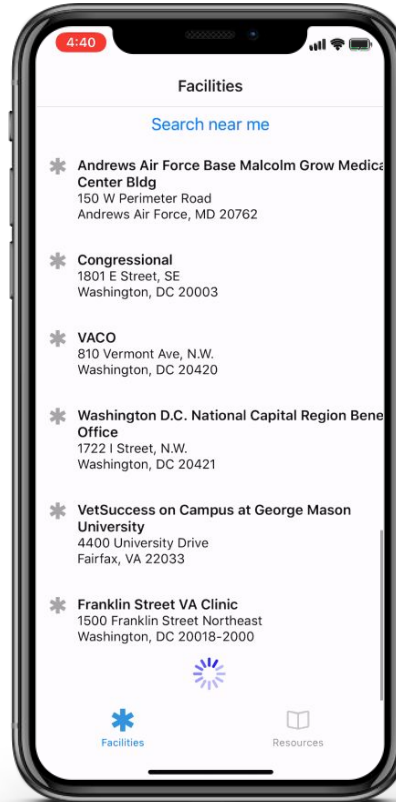


# Component Reuse

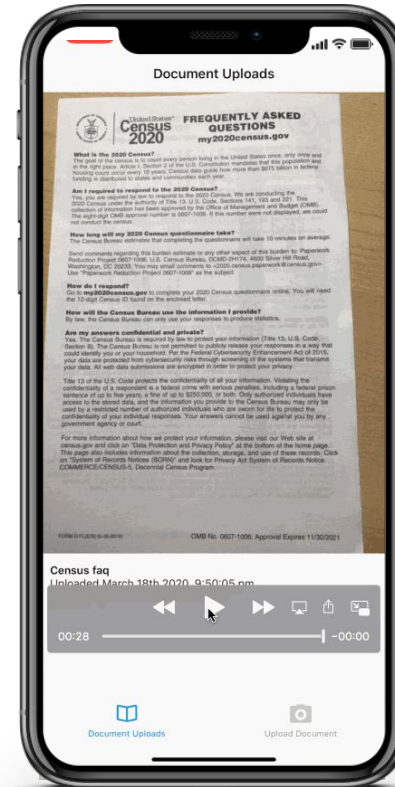
- There's no ability to reuse code from VA.gov or VA Design System
  - ◆ So we'd likely need to do some heavy stylizing in React Native or create a VA component kit, since the web-based design system won't be re-usable
- Web React skills would transfer relatively easily to using React Native
- It would likely still be necessary to have someone with hardware platform expertise on the team

## React Native UX

- React Native will just feel more like a native app
- Some orgs have decided they are doing enough custom work that they might as well go full native



- Smooth integration with the device camera
- Cutting-edge device features may require additional effort



# Developer tooling

- Expo.io allows a fast, on-device development experience
- “[The] developer experience is very nice, especially with Expo. Ability to develop/test on device, simulator, and browser allow for a lot of convenience and flexibility”
- “I had my phone plugged in charging with the Expo app running on it and my computer. For the most part, hot-reload was very fast and would give me a stacktrace if I had an error.”

# **Veteran Desirability**

# Schedule: Veteran Desirability

Planning		Week 1	Week 2	Week 3	Week 4
Feature Areas	Research questions	Phase One	Output	Phase Two	Outcome
<b>Login</b>	What are Veterans preference when logging in to VA mobile?	Card sorting (unmoderated)	Use card sorting results to develop prototype ideas	Finalizing prototype, perform usability testing with Veterans on prototype	Prototype and recommendations for VA mobile login experience
<b>Initial screen/ Dashboard</b>	What initial screen do Veterans expect/need to see after they log on to VA mobile?				Prototype and recommendations for VA initial VA mobile screen
<b>Taking an action on the app</b>	What interactions do Veterans want to be able to perform on VA mobile?				Prototype and recommendations for VA mobile interactions

# Phase Two: Usability testing

## Background

As outlined in **Phase Two** of the [VA Mobile App Discovery: UX Research Plan](#), we performed a series of usability tests to help validate our findings from our card sorting activity into a testable prototype. Our research questions for this phase include:

- **What are Veterans expecting to see** when they initially log on to a VA mobile application?
- **What is the most important secondary action** that a Veteran would do want to when leveraging VA mobile?

## Veteran-centered method

We used moderated, remote usability testing to quickly understand from each Veteran how they interpret each part of our prototype.

**NN Group**, “Usability Testing 101”

<https://www.nngroup.com/articles/usability-testing-101/>

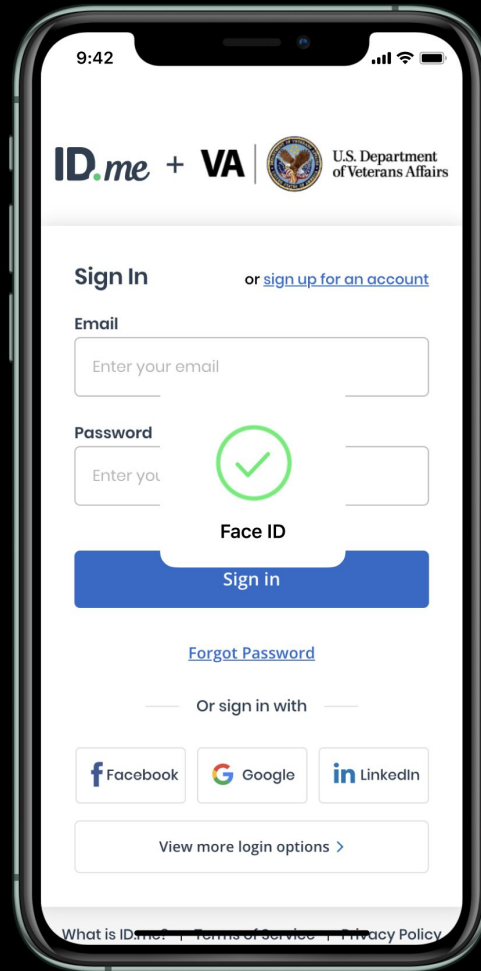
### Research Plan

<https://docs.google.com/document/d/1x366wACqlyLiLWu9frsDzXM-f9BbkAUiKOUrOghWMw4/edit>

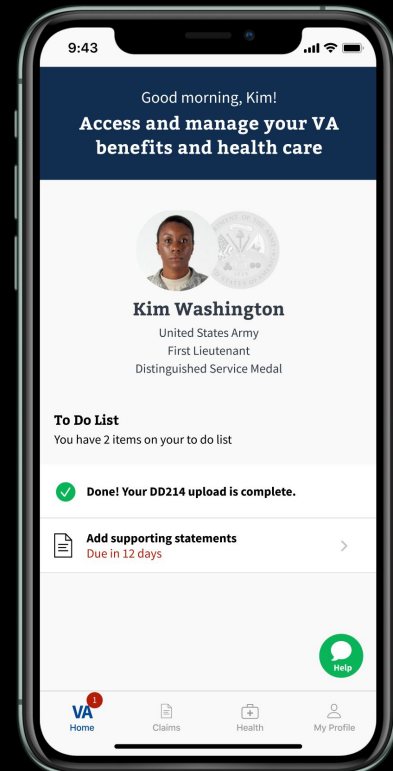
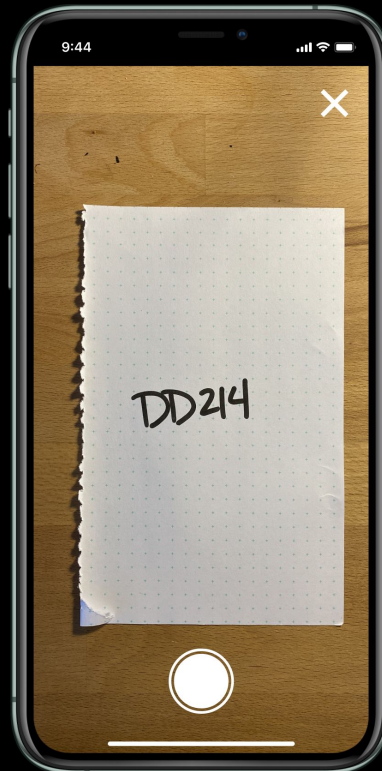
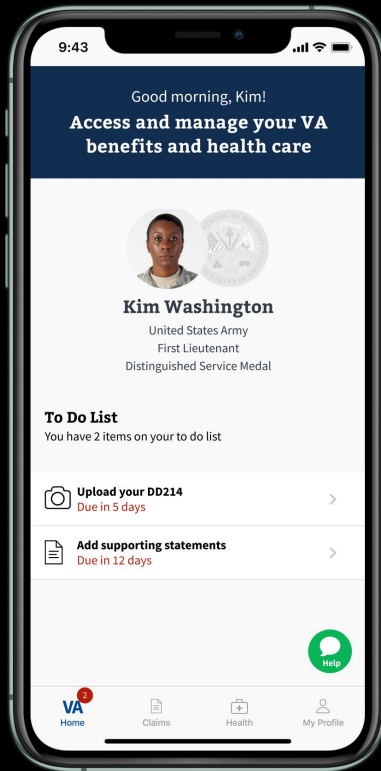
**Prototype**

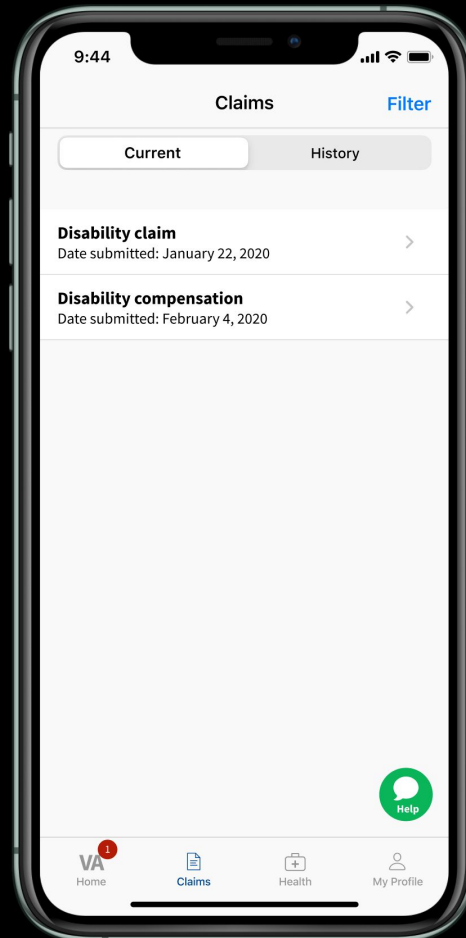


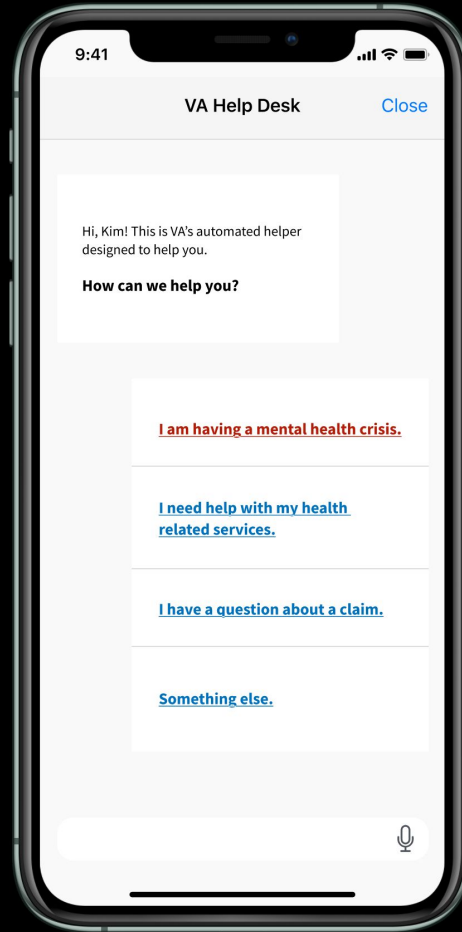




What is ID.me? | Terms of service | Privacy Policy







# Veteran feedback

Overall positive feedback on the concept of a VA companion application for mobile.

# Week 4

# Week 4:

## What you can expect from us next week:

- Overview of key takeaways
- Framework for evaluation
- Next steps for discovery



**Thank you!**

# Extra Time

## → Factors for evaluation (prioritize)

- ◆ Startup cost
- ◆ Time to MVP delivery
- ◆ Drag on VA.gov delivery
- ◆ Maintenance cost
- ◆ Code reuse vs. duplication
- ◆ User experience