

VA



U.S. Department  
of Veterans Affairs

# Engineering Onboarding

VA.gov | VSA

**Bill Tran, VSP Frontend Engineer**  
[bill@oddball.io](mailto:bill@oddball.io)

**Trevor Pierce, VSP Accessibility Expert**  
[trevor@adhocteam.us](mailto:trevor@adhocteam.us)

August 5, 2019

# Welcome!

---

# Presentation Slides

- We will be sharing the PDF version of the slides at the end

# Agenda

- Introductions
- Getting setup
- VSP engineering best practices
- Resources
- Feedback
- Questions

# Introductions

---

# Who we are



**Bill Tran**

Frontend Engineer  
VSP Product Support  
Slack: @Bill John Tran



**Trevor Pierce**

Accessibility Expert  
VSP Product Support  
Slack: @Trevor Pierce



**Anna Carey**

Backend Engineer  
VSP Product Support( Temporary)  
Slack: @Anna Carey

ENGINEERING ONBOARDING

# Introductions

One minute each:

1. Name
2. Role
3. If you couldn't be a developer anymore  
what would you be?

# VSP Development Support Team

- VSP & VFS
  - Veteran Services Platform (VSP)
  - Veteran Facing Services (VFS)
- Names and roles
  - Andrea Hewitt - Project Manager
  - Layla Soileau - Designer
  - Emily Waggoner - Designer
  - Bill Tran - Front-end Engineer
  - Trevor Pierce - Accessibility Expert / Front End Engineer
  - Peter Hill - Test Engineer
  - Anna Carey - Temporary Back-end Engineer
- DSVa Slack
  - [#vfs-platform-support](#) channel



# Getting Setup

---

# Access, Credentials & Local Envs

- [VA Digital Service - Developer Documentation](#)
- [Onboarding for External Contractors](#)

# VSP Engineering Patterns

# VSP responsibility

- Developers experience
- Code quality
- Code consistency
- Prevent VA.gov breakage

# Documentation

- [Engineering Best Practices](#)

# Motivation

- Provide direction, resources and support
- Set expectations for a successful partnership and launch

# Tickets/Issues/User Stories

# Tickets/Issues/User stories

- Refine tickets until they are atomic
- Guideline: no larger than a week's worth of effort



# Tickets/Issues/User stories

Use epics to tether together common bodies of atomic work

# Tickets/Issues/User stories

- Use the GitHub issue template
- Include background that provides context, resources, links, images, and/or POCs
- Identify clear, concise, achievable acceptance criteria
- When applicable, identify risks that could impact delivery, resources, quality, etc.

# Tickets/Issues/User stories

- Use tickets as a forum to discuss and capture implementation ideas, questions, patterns, access, etc.
- In particular when implementation path is not clear, favor discussion prior to submitting a PR

## Example

- <https://github.com/department-of-veterans-affairs/vets.gov-team/issues/8738>

# Tickets/Issues/User stories

Maintain accurate ticket status details (i.e. pipelines, assignees, blockers, etc.)

# Tickets/Issues/User stories

Subscribe relevant VSP team members for visibility and agile support

## Slack/GitHub Groups

- frontend-review-group
- backend-review-group

# Pull Requests

# Pull Requests

- Post/update code in a PR frequently
- Multiple times per week, **no less than once a week**
- Be it work-in-progress, or a PR ready for review

# Pull Requests

- Favor multiple, smaller, focused PR's over large PR's
- The master branches of vets-website and vets-api are deployed [every weekday](#)



# Pull Requests

- Use the GitHub pull request template
- We prefer more details than assumptions
- When applicable, include screenshots
- New API endpoint docs, new UI, etc.

## Examples

- <https://github.com/department-of-veterans-affairs/vets-api/pull/3002>
- <https://github.com/department-of-veterans-affairs/vets.gov-team/issues/19143>

# Pull Requests

Identify work-in-progress PR's by adding WIP to the title, or using GitHub's [draft pull request feature](#)

# Pull Requests

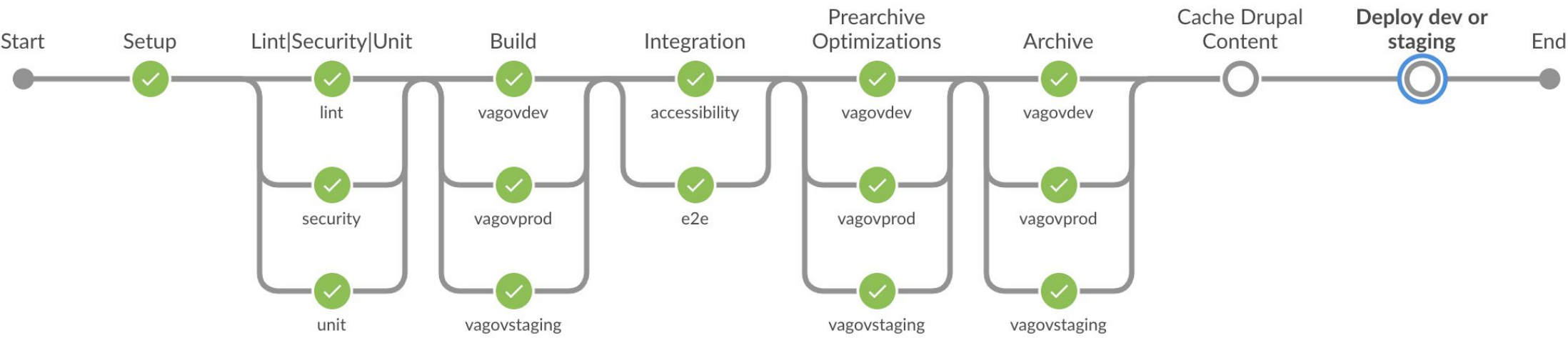
Feel free to request reviews for a work-in-progress PR, and then use GitHub's [re-request review feature](#) to get final PR approvals

# Process

---

# Process

## Deployment process



# Process

Ensure comprehensive test coverage is in place for all code additions/updates

# Process

Hook into/add monitoring and alerting for your service

# Process

Capture relevant analytics for the feature



# Process

Add documentation to support your code and service (Yard docs, product docs, POC's, etc.)

# Backend Development

---

# Backend Development

- Discuss your implementation ideas with VSP backend team members early and often
- There are many common patterns and classes that are in place in Vets-API that should be leveraged, that we can point you to

# Backend Development

- [RuboCop](#) is used for Ruby style linting
- Enforced at the CI level

# Backend Development

- [Screencast](#) on patterns and practices when adding an endpoint to Vets-API (30min.)
- [Slide deck](#) on adding endpoints to Vets-API, includes links to code in the screencast
- [Vets-API Endpoint Punch List](#)

# Frontend Development

---

# Frontend Development

- Familiarize yourself with the [Client application documentation site](#)
- It summarizes several common tasks, common components, and the high level structure of vets-website

# Frontend Development

We encourage frequent merges of small pull requests and using [environment flags](#) to keep partially complete or in progress features from appearing in production



# Frontend Development

- We rely heavily on automated testing in our continuous integration flow to detect regressions
  - e2e
  - unit
  - linting
  - axe-core
- Git hooks and CI for code quality
- It's important to follow our unit testing and end to end testing patterns to maintain a highly collaborative environment

# 508/Accessibility

---

# 508/Accessibility

- VSP takes a multiple approach to accessibility testing

# 508/Accessibility

- Add the [aXe browser plugin](#)

# 508/Accessibility

- Rendered pages must pass an aXe check in e2e suite
- [axeCheck helper function](#)

# 508/Accessibility

- Applications should be usable with just a keyboard

# 508/Accessibility

- Ensure proper keyboard focus settings for single-page apps

# 508/Accessibility

- Ensure headings are properly nested



# General

---

# General

Reminder, VSP engineers are available to review and/or discuss changes!

# Resources

---

# Resources

- Slack Channels
  - [#vfs-platform-support](#)
  - [#vfs-all-teams](#)
  - [#by-light-general](#)
- [VA Digital Service - Developer Documentation](#)
- [Onboarding for External Contractors](#)
- [Engineering best practices](#)

## [Slack/GitHub Groups](#)

- frontend-review-group
- backend-review-group

5-minute survey

[Link to survey](#)

# How are you feeling?

Has anything changed?

- What are you excited about?
- What are you nervous about?
- What questions do you have?

Thank you!

---