ACCESSIBLE CONTINUOUS INTEGRATION

SECURITY AND COMPLIANCE EDITION

CivicActions

ACCESSIBLE CONTINUOUS INTEGRATION | OUTLINE

- ➔ Nerdstein
- ➔ Continuous Integration
- ➔ Current Limitations
- ➔ A Future Vision
- ➔ Accessible Continuous Integration
- ➔ Security and Compliance
- ➔ Case Studies
- ➔ A Call to Action

Nerdstein (Adam)

- → Associate Director of Engineering, CivicActions
- → Masters of Science, Information Systems Security
- → Drupal 8 Maintainer of Taxonomy Menu, Password Policy, Key, Encrypt, Field Encrypt

Continuous Integration

DevOps automates solutions to longstanding CI problems

- ➔ **Continuous learning into applied problem solving**
- ➔ **Consistency equates to predictability and stability**
- ➔ **Automation over error-prone manual processes**
- ➔ **Have no barriers:** *release management, security scanning, log analysis, 508 compliance, automated testing, quality assurance, code reviews, on-demand environments*

You offer a better service to your users with CI practices

Current Limitations

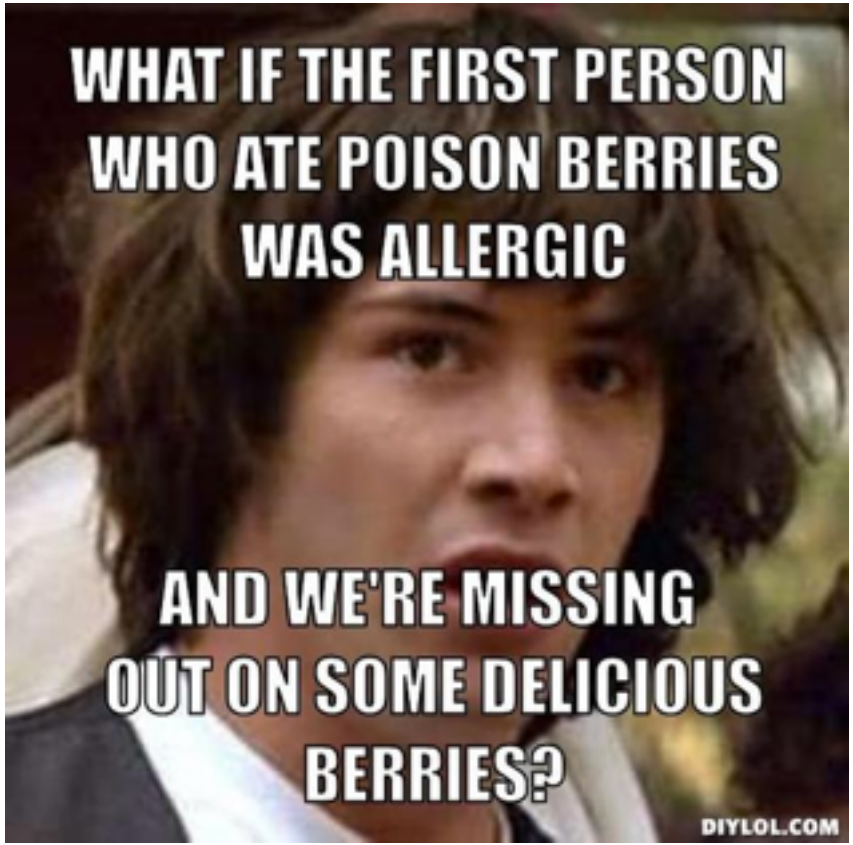
- ➔ Technical people solving technical problems
- ➔ But, outcomes are intended to improve service
- ➔ Services are for those you serve
- ➔ Can we agree there may be some assumptions made if ONLY technical people are engaged?

We’re missing those

we serve in the process

How do you know your actually improving?

A Future Vision


WHAT IF THE FIRST PERSON WHO ATE POISON BERRIES WAS ALLERGIC
AND WE'RE MISSING OUT ON SOME DELICIOUS BERRIES?
DIYLOL.COM

Innovation is not just technical breakthroughs

- ➔ The problem is not the tools, it's that they are not approachable
- ➔ Open a dialogue with those you serve

Build both technical and social bridges

Digital Enablement

Let's promote digital enablement for Continuous Integration…

- ➔ Emphasize ease of adoption and removal of barriers to entry
- ➔ Promote effective information sharing and transparency
- ➔ Drive toward usability of your services and tools
- ➔ Deliver comprehensive and streamlined services

How can we frame high level goals…

- ➔ SIMPLE - Processes void of encumbrance
- ➔ USEFUL - Solve meaningful problems
- ➔ FLEXIBLE - Build robust, long-term, unassuming solutions
- ➔ TRANSPARENT - Communicate concisely and frequently

We aim to take CI to the masses

Accessible Continuous Integration

CivicActions

Wikipedia defines accessibility as…

The process of creating products that are usable by people with the widest possible range of abilities, operating within the widest possible range of situations.

Do not get confused with

508 Compliance *Accessibility*

There are current practices that build bridges…

- ➔ Abstracting technical details
- ➔ Systems integration
- ➔ Streamlined processes
- ➔ Platform and device agnostic

1. Abstracting technical details
2. Integrate systems
3. End-user involvement
4. Platform and device agnostic

Abstracting technical details...

- ➔ KISS concept (Keep it simple, stupid)
- ➔ Systems must promote usability, account for technical literacy
- ➔ Build finely tuned user interfaces, not command lines
- ➔ Limit decision points, add help text, consistent UI design

1. Abstracting technical details
2. **Integrate systems**
3. End-user involvement
4. Platform and device agnostic

Integrate systems...

- ➔ Select systems that promote interoperability (future proof for continuous learning)
- ➔ Connect systems instead of forcing users to use multiple systems
- ➔ Make use of web services, APIs, plugin systems, and give back to communities so others can benefit

1. Abstracting technical details
2. Integrate systems
3. **End-user involvement**
4. Platform and device agnostic

End-user involvement...

- ➔ Identify systems in which your users are comfortable using -- enhance them
- ➔ Avoid forcing users to learn too much or use new systems
- ➔ Encourage user testing and feedback loops to participate in Continuous Integration discussions

1. Abstracting technical details
2. Integrate systems
3. End-user involvement
4. **Platform and device agnostic**

Platform and device agnostic...

- ➔ Users want access everywhere and immediately
- ➔ Avoid systems (like email) where communication can break down
- ➔ Adopt best of breed solutions that don't restrict platforms
- ➔ Systems should not only be on desktops, use of mobile phones, tablets, and refrigerators (Internet of Things)

Tear down the walls of CI participation


you had me at 'meat tornado'
ORANGE AFRO

Security and Compliance

So… what do your users need?

- ➔ They need confidence that you are on top of security
- ➔ They need continued assurance that you are proactive
- ➔ Security scanning
- ➔ Section 508 compliance and enablement
- ➔ Effective access control
- ➔ Process improvement and fire drills

So… what do your users expect?

THEY ASSUME YOU KNOW AND FOLLOW BEST PRACTICES

- ➔ Regular SOFTWARE updates (not just Drupal; software, OS)
- ➔ User permissions and access control auditing
- ➔ Logging site activity, auditing and management
- ➔ Install security-related modules (password policy, autologout, TFA, role watchdog, security review, paranoia, account sentinel, etc)
- ➔ Secure configuration (user registration, text formats, secure passwords)

THEY ASSUME IT'S COMPREHENSIVE

- ➔ There is a lot of trust and most user's don't know how to measure if it's successful or not
- ➔ Protected infrastructure, not just application
- ➔ Web application firewalls, load balancing, etc
- ➔ Test your processes regularly
- ➔ Crap happens. Security is risk mitigation not eradication!

What are common practices?

A COMPLETE CI SOLUTION

- ➔ Backup and restoration (and yes, you need to test this)
- ➔ Automated (predictable) deployments
- ➔ Code reviews (manual and automated)
- ➔ Recurring scanning tools
- ➔ Monitoring and alerts
- ➔ Log analysis tools (Sumo Logic, Elk)

Case Studies

Accessible CI in the wild...

1. Slack and Jenkins
2. Data Analysis
3. JIRA and Test Driven Development

Slack and Jenkins is a happy marriage...

- ➔ **Slack is highly intuitive for non-technical users**
- ➔ **Slack supports custom commands with help text**
- ➔ **Example: Run a security scan from Slack, send condensed summary back to Slack with results**
- ➔ **Commands can include running Jenkins commands**
  - ◆ *Abstract parameters and options into separate commands*
  - ◆ *Customize output to Slack so it's relevant to all users*

Accessible CI in the wild...

1. Slack and Jenkins
2. Data Analysis
3. JIRA and Test Driven Development

Promote transparency through data analysis…

- ➔ Put data at your users fingertips, saves email, development time, requests, etc.
- ➔ Provide dynamic dashboards that SUMMARIZE effective information
- ➔ Platform stability, health, and performance metrics (red, yellow, green)
- ➔ Analytics and end-user behaviors (traffic patterns, most viewed, least viewed, etc)
- ➔ Log analysis, trends, and insights

Accessible CI in the wild...

1. Slack and Jenkins
2. Data Analysis
3. JIRA and Test Driven Development

TDD can be improved by using JIRA and Repo Hooks...

- ➔ Development activities can be more secured when validated! Manual testing is not comprehensive and error prone
- ➔ Behave for JIRA plugin, integrates with code repository
- ➔ Empower users to write automated tests within JIRA tickets
- ➔ Leverage hooks in the repository to run automated tests when developers submit pull requests (TravisCI or Jenkins)
- ➔ Feedback loops are drastically shortened between user needs and developer's code

A Call to Action

Unlock the potential

Empower your users

Accessible Continuous Integration

Thank you, DrupalCon!

Questions?