HOW TO BE A BETTER DEVELOPER!!

Description

Ever wonder how there is one developer who can do the work of 5 others?

What do they know that you don't?

Duration

16 hours

Intended Audience

- Anyone doing development
- Programmers, QA people, team leaders, managers.

Prerequisites

- Some development background

Outline

- Script everything
- This means even things you do once every year
- The practice will make you better at it

<u>Writing code</u>

- Reward developers who remove code more than those who add code
- Avoid "it's working so don't touch it" attitude
- Allow developers to rebuild everything, fast!
- Document the hard parts in your code
- Write hidden assumptions about your software which are not obvious from the code
- Reward developers who manage to redesign your code - they are the true heroes.
- Agile works, rewriting doens't.
- Keep it simple, Stupid! (your codes architecture that is).
- Have good taste.

<u>Debugging</u>

- Do not use a debugger
- Alternatives to debugging
- ∗ Reading your code
- ∗ Infrastructure for visibility
- ∗ Configurable logging

- Debugging cycles should be fast (see building your code fast above).

Do not use user interfaces

- Not reproducible
- Not scriptable
- Always changing
- Pulling\ manual instead of automatic work
- Don't let fancy GUI tempt you to the dark side

Do not write code which is not your main business

- You are not good at it
- You don't want to maintain it
- Get open source alternatives or any other alternative
- Anything which is not your main business should be open sourced

Your development environment

- Keep yourself agile regarding your tools
- Use as few tools as possible
- Be sure about why you use all of these tools
- Drop tools you no longer need to simplify your environment.
- Allow users to enter password or MFA easilty and keep security out of their way
- Dump as much as possible on automated cloud services
- Keep it simple, Stupid! (your development environment, that is)

Configuration is the root of all evil.

- Why keeping the default configuration is the best
- If you can't use the default, change the configuration as little as possible.
- Make sure your entire team uses the same configuration
- Keep it simple, Stupid! (your configuration that is)

Separate systems that should be separated

- Prefer multi-process to multi-thread
- Micro-services
- Different code repos
- Everything in computing is about separation.

Zero aural traditions

- No special urls, users, passwords, tools, that people need to gather: everything in one clear page.
- The test of building and running your code from git
- Script bootstrapping your code
- Script installing all dependencies
- Script everything!

Test, test, test

- Test especially the things which failed in the past
- Reward new tests which pass, never settle for failing tests.

Stop using MS-Windows!

- Why Linux is better for developers and facilitates all of the above

מבין לקוחותינו:

J.P.Morgan   NCR   לאומי leumi   AT&T

דיסקונט   Apple   אלביט מערכות   waze

שירות הביטחון הכללי   Webhelp   Cal.   תל אביב יפו TEL AVIV YAFO

Google   ebay   NVIDIA   SONY

בנק הפועלים   Qualcomm   kaltura

BROADCOM   harmonic   CYBERARK