

What I learned

Czyli cykl notatek z życia programisty

Aleksander Orchowski

ORCHOWSKIA#

czerwiec 2019

Spis treści

Kategoria 1

Legacy code

1.1 Podstawy pracy

1.1.1 Algorytm wprowadzania zmian

Feathers pisze o algorytmie wg. którego należy postępować podczas wprowadzania zmian do kodu legacy. Oto on:

1. Zidentyfikuj miejsca zmian.
2. Odszukaj miejsca na wstawienie testów.
3. Usuń zależności.
4. Napisz testy.
5. Wprowadź zmiany i dokonaj refaktoryzacji.

Kategoria 2

Do uporządkowania

Tu znajdują się rzeczy które nie znalazły jeszcze swojej kategorii i czekają na nią, oraz uporządkowanie.

2.1 Architektura aplikacji

To jest raczej moje przemyślenie odnośnie wytwarzania aplikacji ich struktury i działania.

Bardzo spodobało mi się podejście czy też filozofia UNIX. Ma ona kilka założeń które stoją u podstaw tego wspaniałego ekosystemu. Co prawda nie można do końca tego tak spłycać, ale zauważyć można pewną prawidłowość odnoszącą się do wytwarzania systemów i ewolucji w stronę mikroservisów.

- Każdy program powinien robić jedną rzecz. Lepiej zrobić kolejny program niż komplikować obecny.
- Spodziewaj się że output z jednego programu będzie inputem dla innego - nieznanego programu. Niech output będzie więc lakoniczny.
- Buduj oprogramowanie tak aby mogły być wypróbowane wcześniej, nawet jeśli nie wszystko jest jeszcze dopracowane. Zawsze można wyrzucić niepotrzebne fragmenty i je przebudować.
- Jeśli potrzebujesz narzędzi to je buduj, nawet jeśli ci nie po drodze. To ci pomoże przy programowaniu. Udostępnij je później, lub wyrzyc jeśli nie przynoszą wartości.(automatyzacja)