

# Fitness Functions

Creating Consistency across Hundreds of Services

Bijan Chokoufe Nejad -  
StaffPlus London 2024

**JIMDO**

# Good software, everywhere?



# Good software, everywhere?





# ENGINEERING GUIDELINES



Содержит, определяет, устанавливает и утверждает:

- 1. Назначение, цели, задачи и задачи.
- 2. Структуру, состав, функции и полномочия.
- 3. Порядок, сроки и методы выполнения работ.
- 4. Ответственность за выполнение работ.
- 5. Порядок взаимодействия с другими подразделениями.
- 6. Порядок контроля и отчетности.
- 7. Порядок внесения изменений.
- 8. Порядок хранения и использования документов.
- 9. Порядок обучения и повышения квалификации.
- 10. Порядок проведения ревизий и аудитов.
- 11. Порядок проведения мероприятий по улучшению качества.
- 12. Порядок проведения мероприятий по обеспечению безопасности.
- 13. Порядок проведения мероприятий по обеспечению экологической безопасности.
- 14. Порядок проведения мероприятий по обеспечению пожарной безопасности.
- 15. Порядок проведения мероприятий по обеспечению безопасности жизнедеятельности.
- 16. Порядок проведения мероприятий по обеспечению информационной безопасности.
- 17. Порядок проведения мероприятий по обеспечению энергетической безопасности.
- 18. Порядок проведения мероприятий по обеспечению радиационной безопасности.
- 19. Порядок проведения мероприятий по обеспечению безопасности при чрезвычайных ситуациях.
- 20. Порядок проведения мероприятий по обеспечению безопасности при проведении работ в условиях повышенной опасности.





# ENGINEERING GUIDELINES



Содержит, определяет, устанавливает и  
описывает:

- 1. Методы и средства измерения качества
- 2. Средства измерения качества
- 3. Средства измерения качества
- 4. Средства измерения качества
- 5. Средства измерения качества
- 6. Средства измерения качества
- 7. Средства измерения качества
- 8. Средства измерения качества
- 9. Средства измерения качества
- 10. Средства измерения качества



# What's a fitness function?

Building Evolutionary Architectures, 2017, Ford, Parsons, Kua

*An architectural fitness function provides an **objective integrity assessment** of some architectural characteristic(s).*

We can write **tests** that verify that **our system** obeys our **nonfunctional requirements**

# When to run fitness functions?

In 2019, Paula and Wang (Thoughtworks) described how to implement a **fitness function-driven development** by incorporating checks in build pipelines

# When to run fitness functions?

In 2019, Paula and Wang (Thoughtworks) described how to implement a **fitness function-driven development** by incorporating checks in build pipelines

But what if nobody actively works on a project?

We can run tests centrally for all projects, ensure ownership for all services and inform owning teams of required changes



**Jimdo**: the website builder from 2007  
turned business builder for small businesses

17 years

30M websites

**Jimdo**: the website builder from 2007  
turned business builder for small businesses

17 years

30M websites





214 services

70 engineers

# Failed experiment: **Feature Flag Janitor**

- Checks age of all feature flags of a team and notifies them *when they configure it*

# Failed experiment: **Feature Flag Janitor**

- Checks age of all feature flags of a team and notifies them *when they configure it*
-  **Complicated setup:** ~10 steps to set it up correctly
-  **No support:** Nobody followed up with the teams
-  **No Alignment:** Was created in 2021 were many engineering leaders were still betting on explosive growth instead of prioritizing tech debt
-  **Alert fatigue:** Teams learned to ignore the notification that came twice a week as nobody followed up on them

# Success case: **Deployment notifier**

- Looks for services without regular deployments and notifies teams

# Success case: **Deployment notifier**

- Looks for services without regular deployments and notifies teams
- **✓ No setup cost for the teams:** Tool figures everything out automatically.
- **✓ Dedicated support:** I get a summary of the alerts and join team channels to understand if the works gets done or is at least prioritized
- **✓ Alignment with CTO + Directors:** “Quality is not negotiable”
- **✓ Alerts need to be acted on:** To avoid me asking annoying questions

# Success case: **Deployment notifier**

## Results:

- **15 %** of services were violating the rule of regular deployments from the engineering guidelines initially
- **3 %** of services could be deleted because they were no longer needed
- **3 %** of services didn't have clear ownership (missing/outdated alerting channels)
- Now **we know exactly** that all services have a working CI/CD pipeline because **we test our infrastructure weekly** for this attribute

# Summary

- **Fitness functions** shine a light of knowledge and clarity into the fog of uncertainty and ambiguity
- To make them **effective** in driving change, be sure to
  - **Align** with your engineering leaders first
  - Make it **easy** for your teams
  - Follow up and **support**

**Thank you for your time**