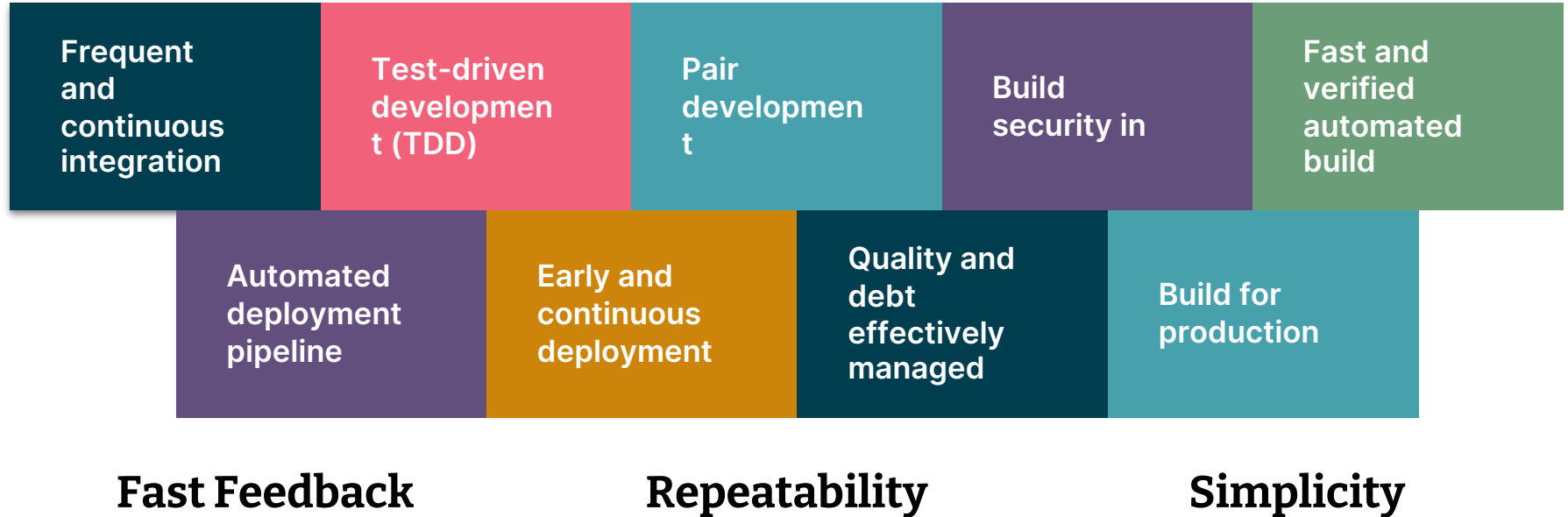


Refactorizare & Bune practici

Mihai Angheluță



Bune practici implicite



1. Frequent and continuous integration

2. Test Driven Development (TDD)

3. Pair development

4. Build security in

5. Fast and verified automated build

6. Automated deployment pipeline

7. Early and continuous deployment

8. Quality and debt effectively managed

9. Build for production

Fast Feedback

Fast feedback means being able to find out whether a change has been successful in moments not days. It might be that unit tests have passed, or that we haven't broken production, or that a customer is happy with what we've built.

Repeatability

Repeatability is the confidence and predictability that comes from removing manual tasks that introduce weird inconsistencies. We also want to spend time on activities that are more important than troubleshooting something that should have just worked.

Simplicity

We want software that contains no more complexity than it needs to do a good job. We build for what we need now, not what we think might be coming. But we make choices that allow our software to rapidly change to meet the requirements that are coming.

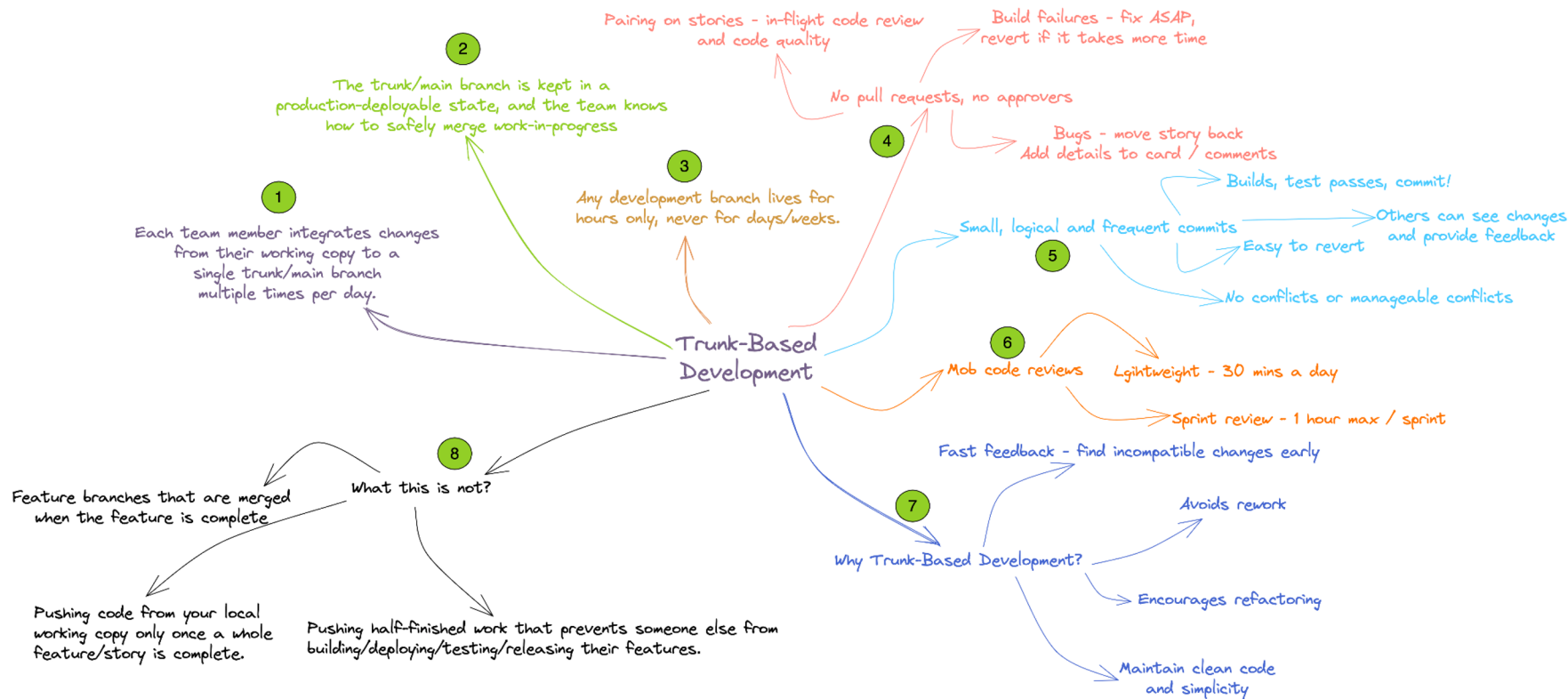
↑ Deployment Frequency

↓ MTTR

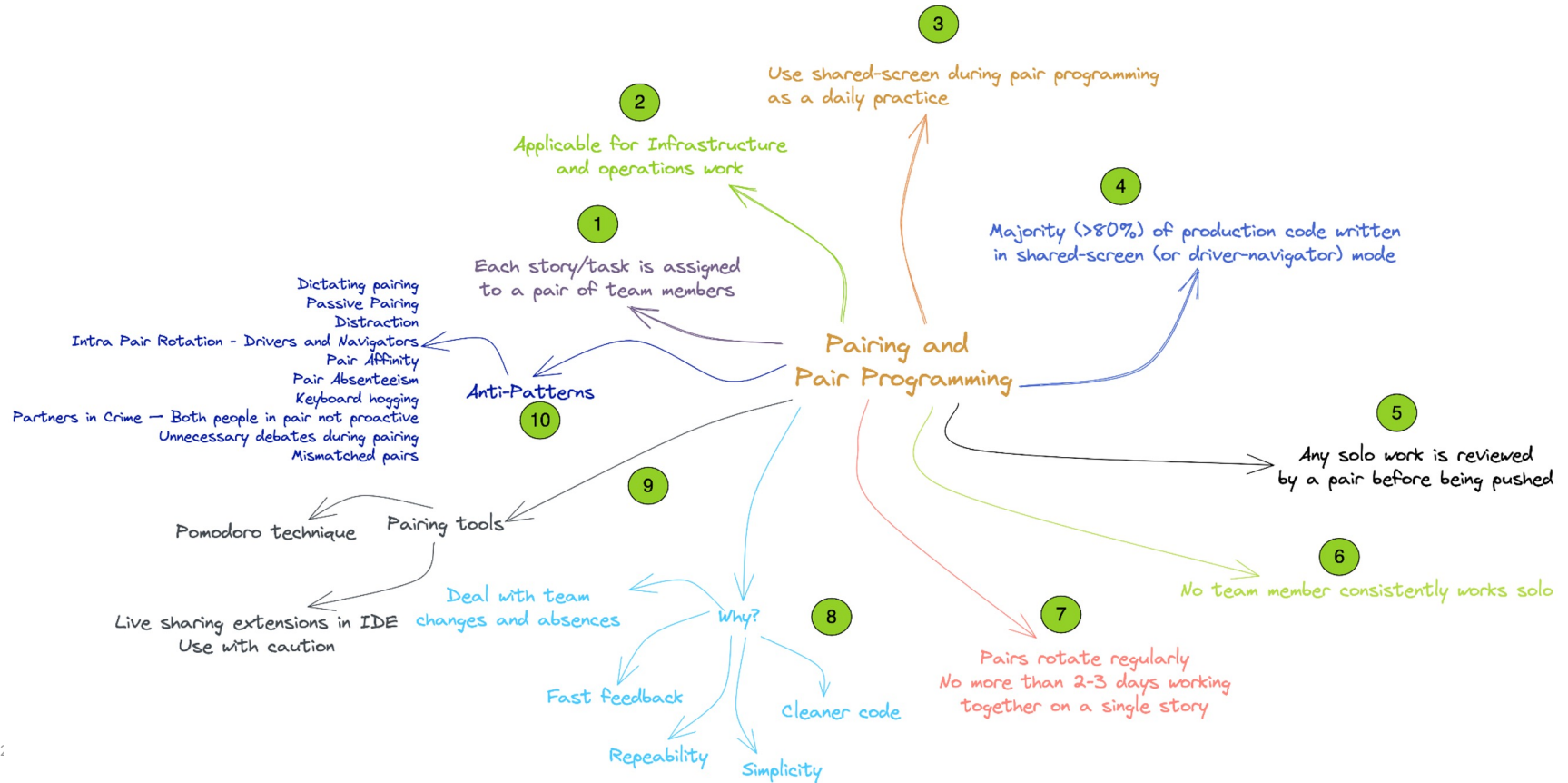
↓ Lead Time for changes

↓ Change Failure Rate

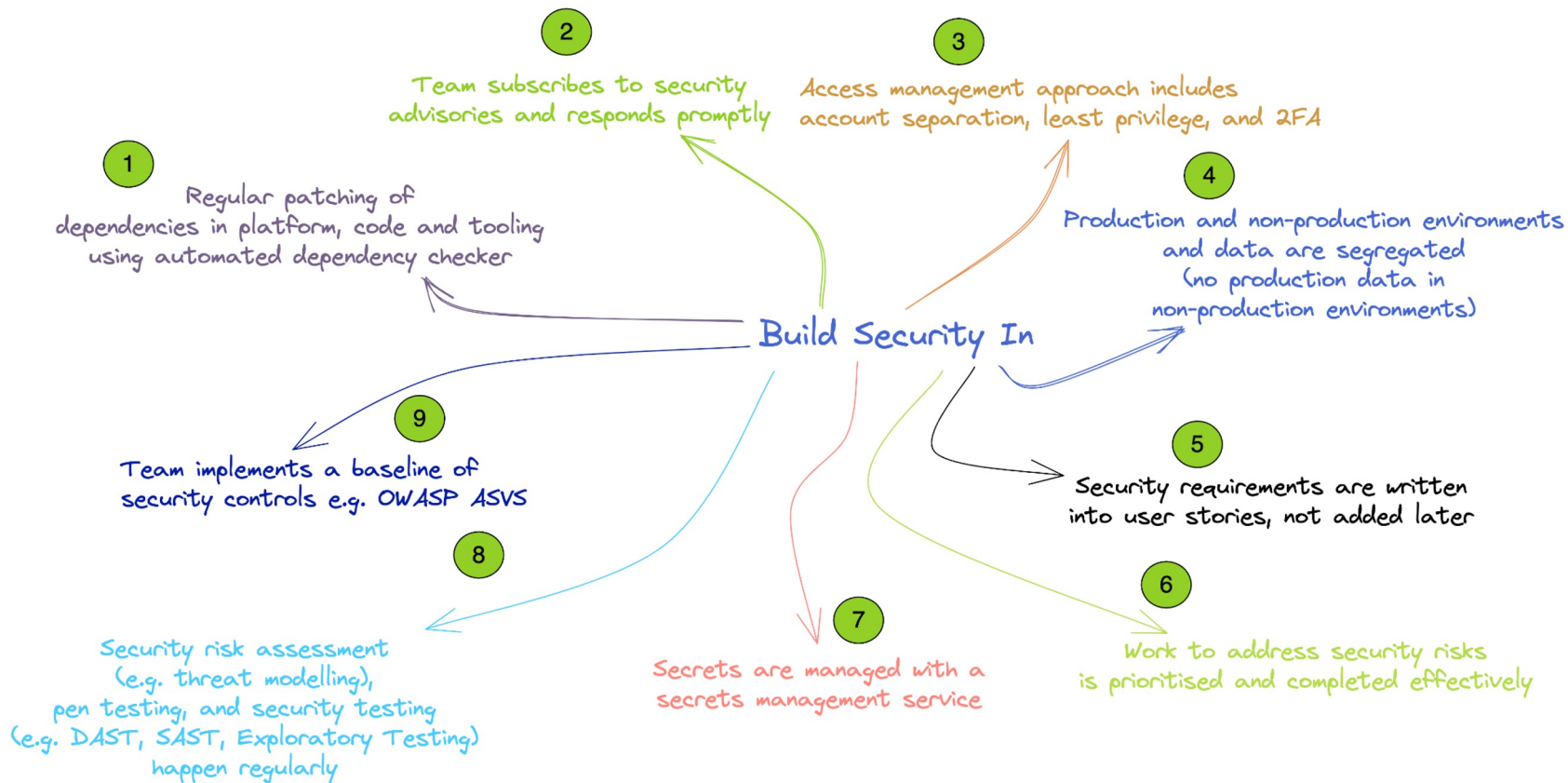
Frequent and continuous integration



Pairing and pair programming



Build Security In



Quality and debt effectively managed



Ce este refactorizarea?

De ce este necesară?

Modificarea unei secvențe de cod care funcționează corect fără a-i schimba funcționalitatea externă

Parte a activităților zilnice

Unelte care automatizează sunt utile dar nu esențiale



Reorganizare



Optimizare

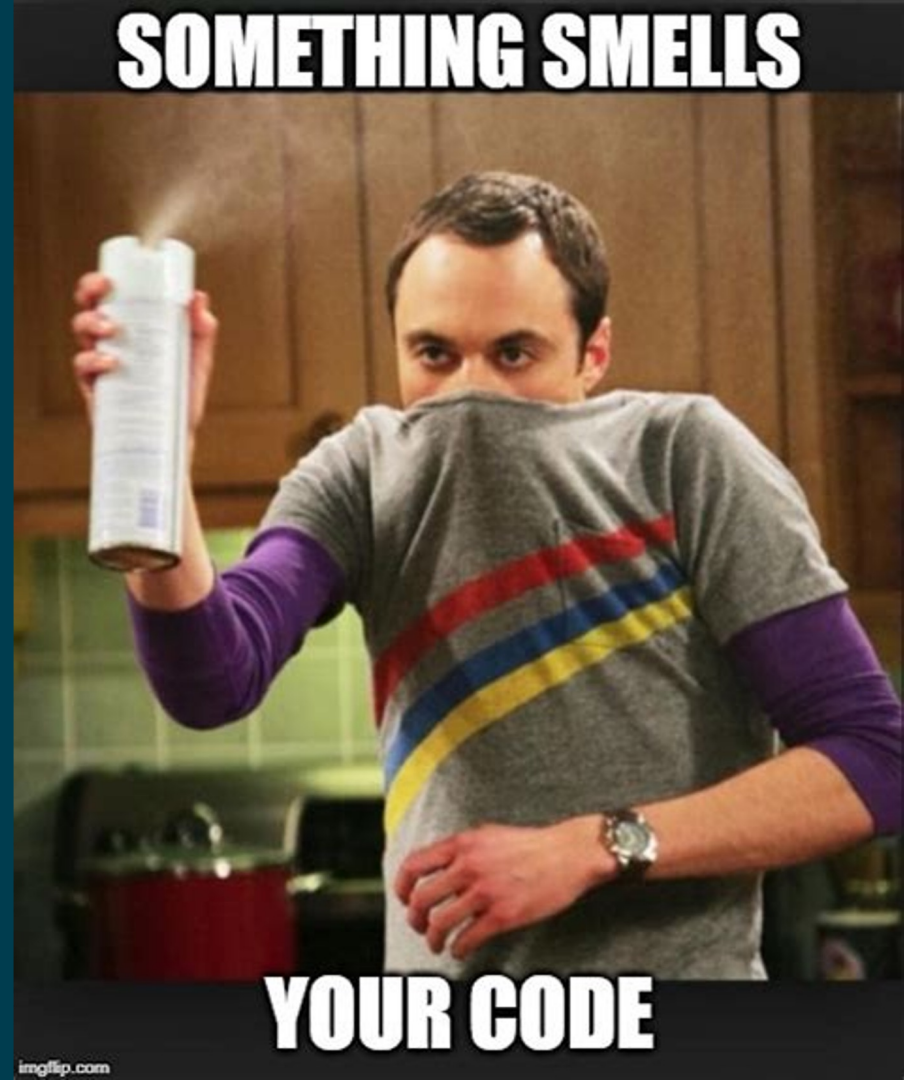


Comportamentul extern nu e alterat



Structura internă e îmbunătățită

Code smells



La nivel de aplicație

- **Duplicated code**
- **Shotgun surgery**
- **Contrived complexity**
- **Speculative generality**
- **Mysterious name**

La nivel de clase

- **God object**
- **Feature envy**
- **Lazy class**
- **Inappropriate intimacy**
- **Refused bequest**
- **Cyclomatic complexity**
- **Data clump**
- **Message chain**
- **Middleman**

La nivel de metode

- **Prea mulți parametri**
- **Metodă prea lungă**
- **Switch sau multiple blocuri de decizie**
- **Prea multe comentarii**
- **Nume prea lungi sau prea scurte**

Câte

I regret

It v

I REGRET



NOTHING!

versionare și
anterioară la

te acoperitoare

c commits)

oas

panezi. Sau invers

augi funcționalități

Mulțumesc

Mihai Angheluță

Lead Consultant

mihai.angheluta@thoughtworks.com

<https://www.linkedin.com/in/mihai-angheluta/>

<https://www.instagram.com/mihai.v.a/>

