

A woman with red hair, likely Arya Stark from Game of Thrones, is shown in a close-up shot. She is holding a sword and looking off to the side with a serious expression. The background is a blurred outdoor setting.

Tu ne sais rien de la qualité logicielle,
JonSnow, DSI à
MarcheurBlancConsulting

Arthur Magne

CTO & Co-fondateur

@

 rQMyze



Le software craftsmanship ?

- Responsabilité
- Autonomie
- Amélioration continue
- Échange et partage
- Satisfaction du client



Manifesto for Software Craftsmanship

Elever le niveau

En tant qu'aspirants Artisans du Logiciel, nous relevons le niveau du développement professionnel de logiciels par la pratique et en aidant les autres à acquérir le savoir-faire. Grâce à ce travail, nous avons appris à apprécier :

Pas seulement des logiciels opérationnels,
mais aussi **des logiciels bien conçus.**

Pas seulement l'adaptation aux changements,
mais aussi **l'ajout constant de la valeur.**

Pas seulement les individus et leurs interactions,
mais aussi **une communauté de professionnels.**

Pas seulement la collaboration avec les clients,
mais aussi **des partenariats productifs.**

10 ans plus tard ...



Manifesto for Software Craftsmanship

Elever le niveau

En tant qu'aspirants Artisans du Logiciel, nous relevons le niveau du développement professionnel de logiciels par la pratique et en aidant les autres à acquérir le savoir-faire. Grâce à ce travail, nous avons appris à apprécier :

Pas seulement des logiciels opérationnels,
mais aussi **des logiciels bien conçus.**

Pas seulement l'adaptation aux changements,
mais aussi **l'ajout constant de la valeur.**

Pas seulement les individus et leurs interactions,
mais aussi **une communauté de professionnels.**

Pas seulement la collaboration avec les clients,
mais aussi **des partenariats productifs.**

UNE COMMUNAUTÉ

Groupes (Okiwi / JUG / Software Crafters / etc.)

Meetups

Brown Bag Lunch

Coding Dojo

Conférences

Manifesto for Software Craftsmanship

Elever le niveau

En tant qu'aspirants Artisans du Logiciel, nous relevons le niveau du développement professionnel de logiciels par la pratique et en aidant les autres à acquérir le savoir-faire. Grâce à ce travail, nous avons appris à apprécier :

Pas seulement des logiciels opérationnels,
mais aussi **des logiciels bien conçus.**

Pas seulement l'adaptation aux changements,
mais aussi **l'ajout constant de la valeur.**

Pas seulement les individus et leurs interactions,
mais aussi **une communauté de professionnels.**

Pas seulement la collaboration avec les clients,
mais aussi **des partenariats productifs.**

Développement => Création



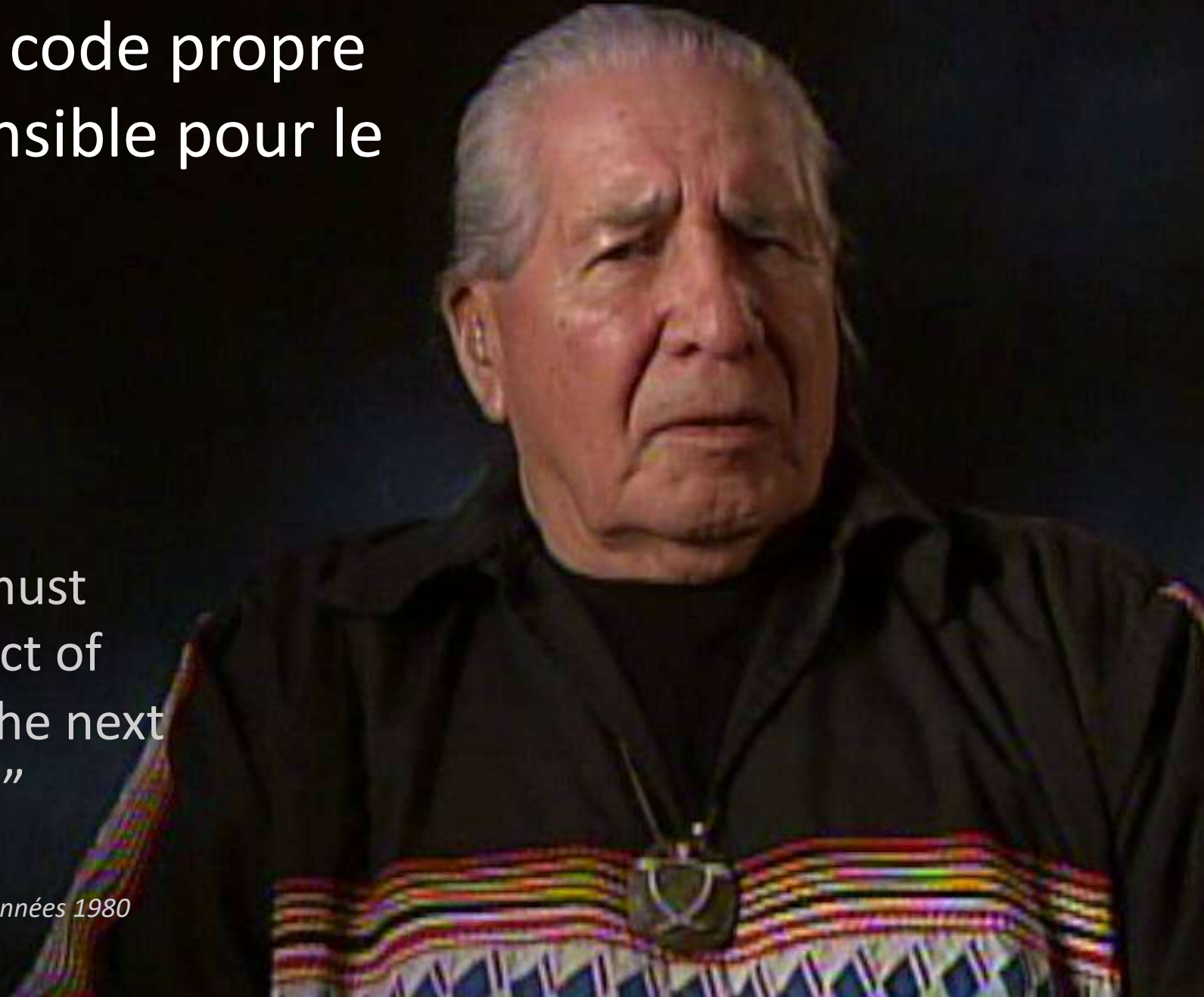


A NEW FEATURE IS COMING

Maintenir un code propre
et compréhensible pour le
futur

“In our every
deliberation, we must
consider the impact of
our decisions on the next
seven generations”

Oren Lyons – Iroquois Onondaga – années 1980



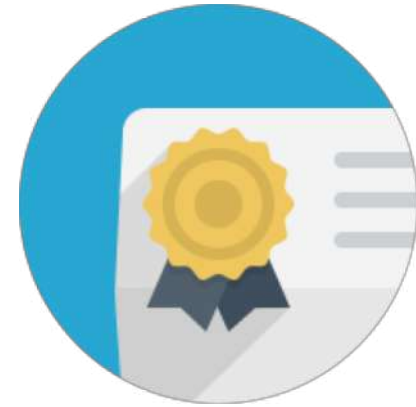
Qualité logicielle ?



Architecture



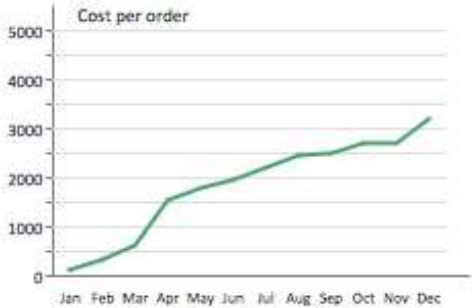
Tests



Clean Code

Key performance indicators

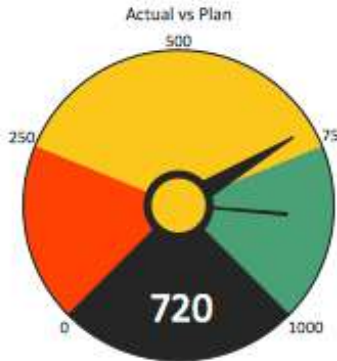
Cost per order



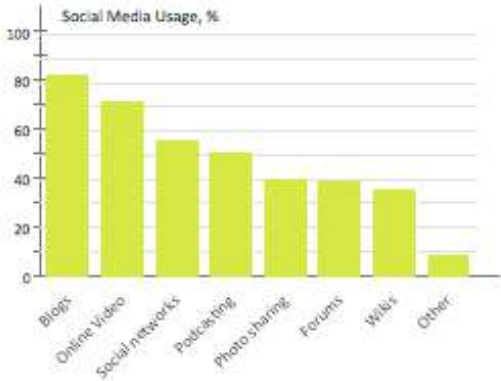
Clicks from Sales Follow-Up Emails



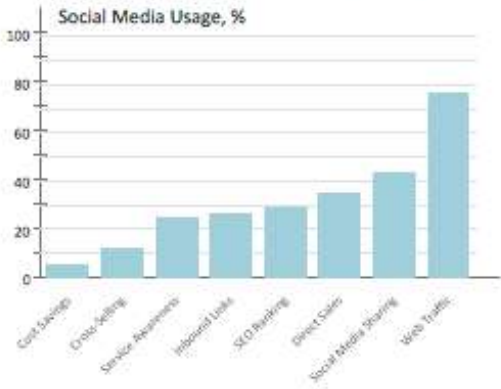
Sales Growth



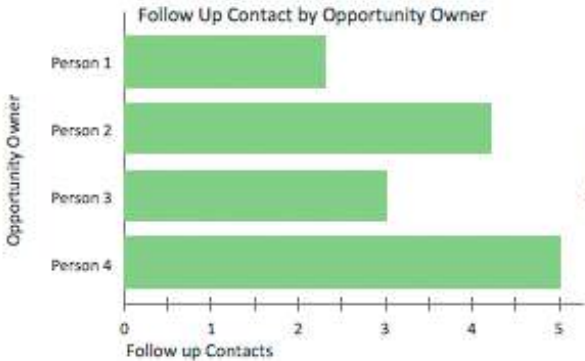
Social Media Usage



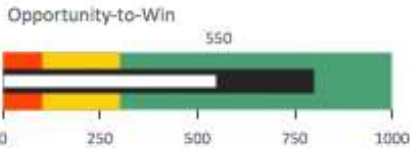
Usage Rate of Marketing Collateral



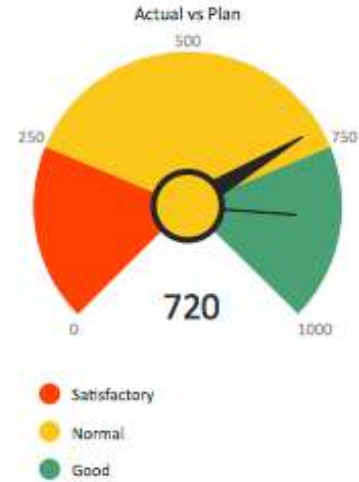
Rate of Follow Up Contact



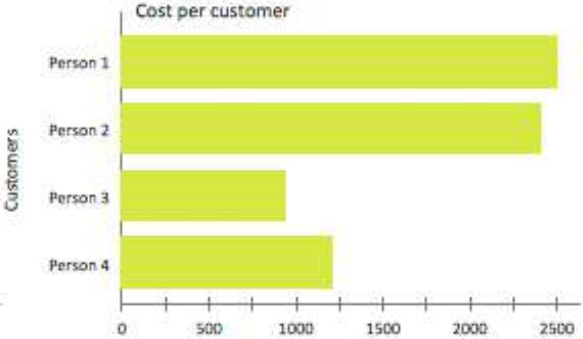
Sales Target



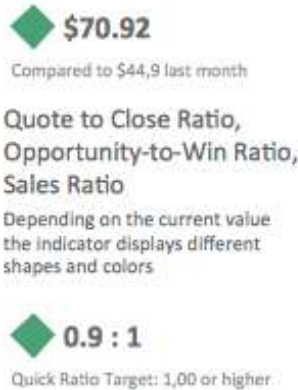
Actual vs Plan



Cost per customer



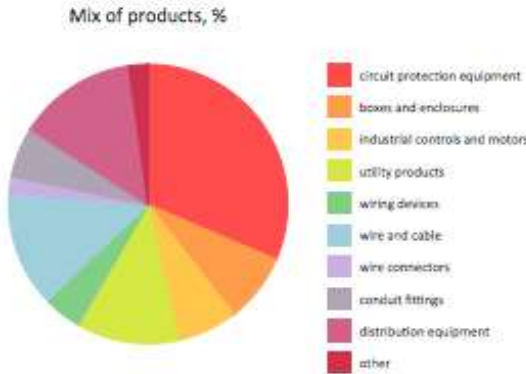
Average Purchase Value



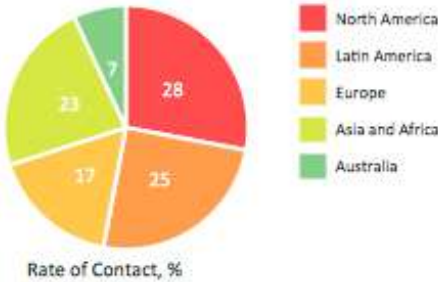
Product Performance



Product mix



Rate of Contact





Architecture



CLOSE ENOUGH





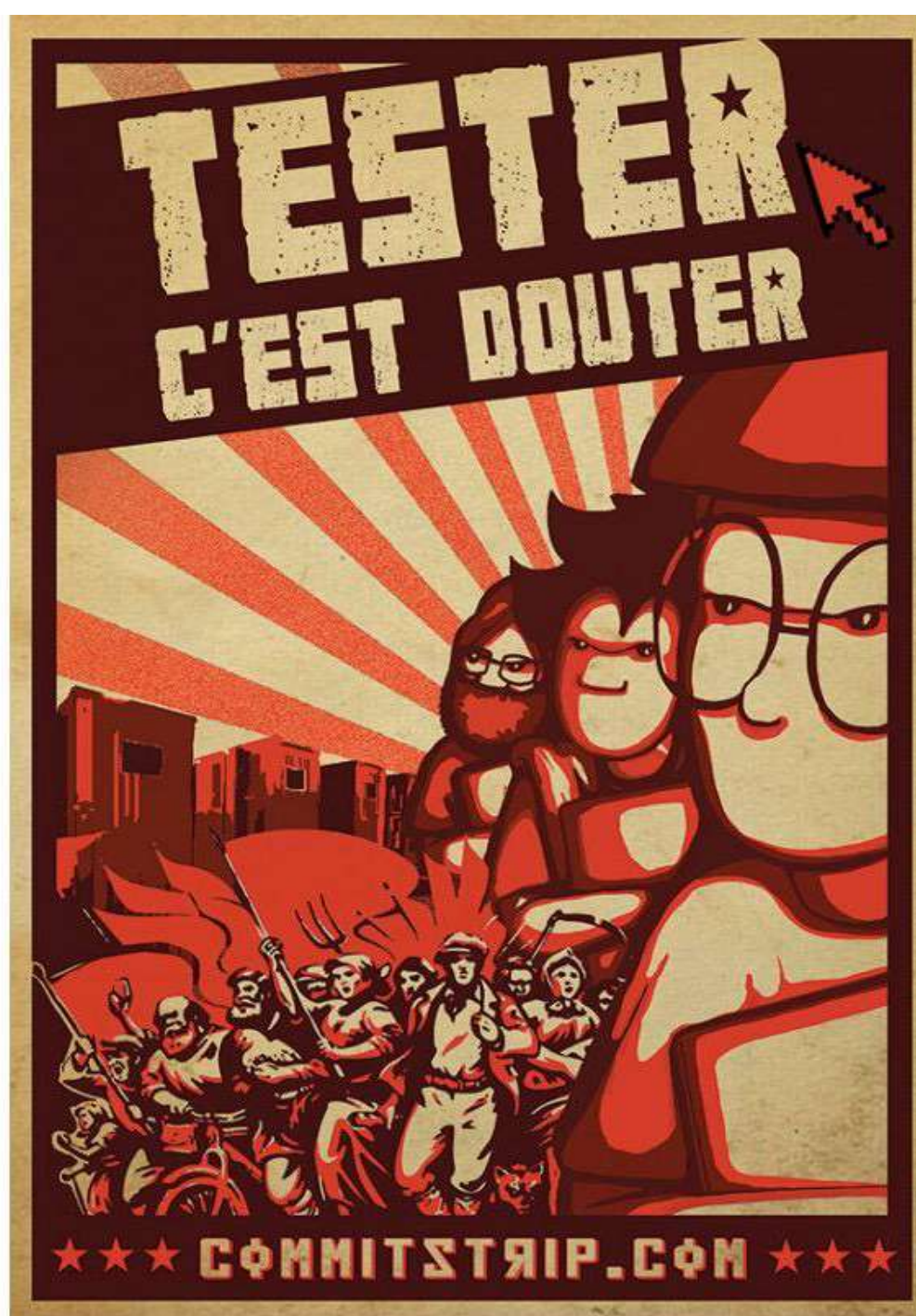
Architecture

- Pair programming
- Code review / Conception review
- Coding dojo
- Lecture



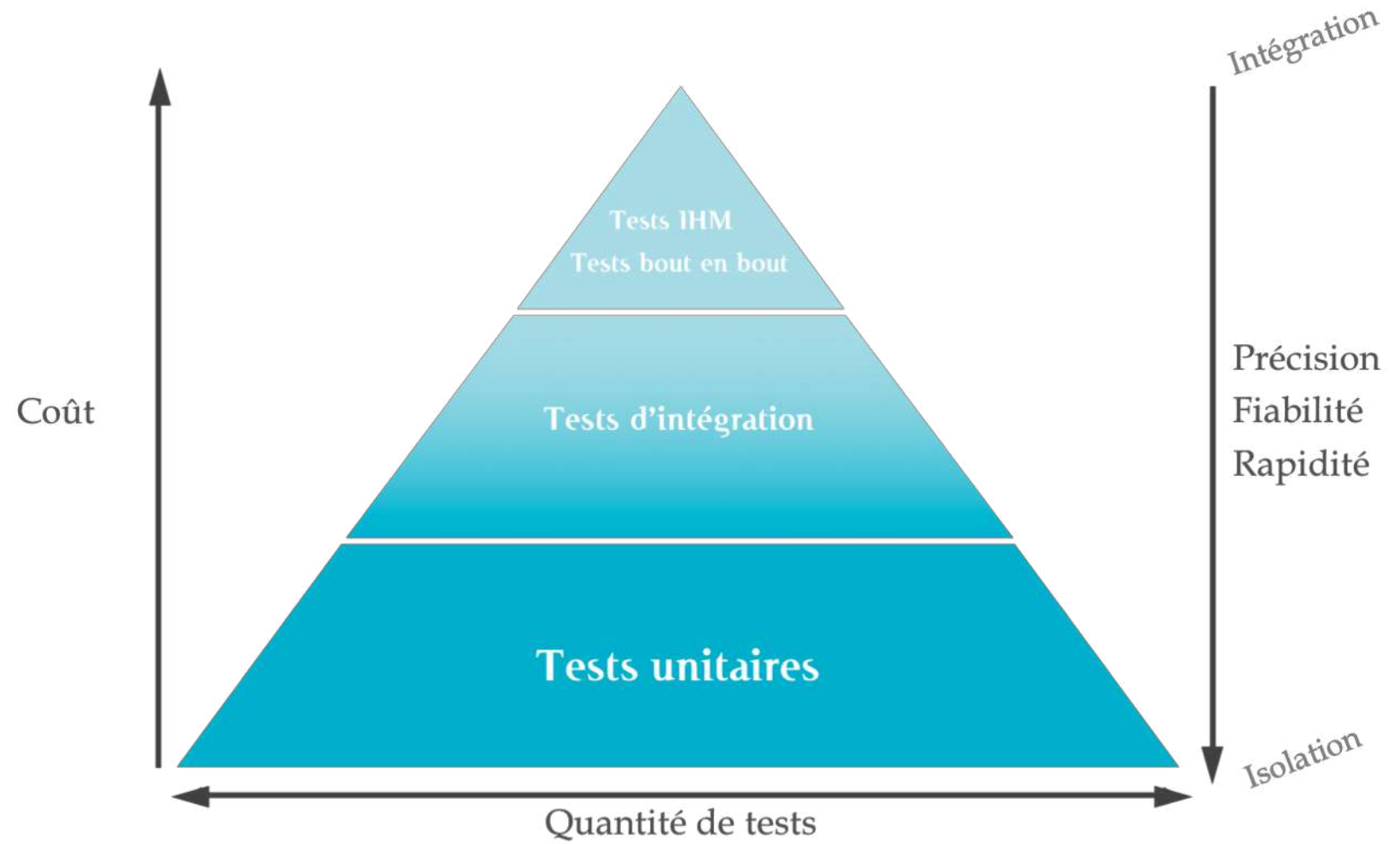


Tests



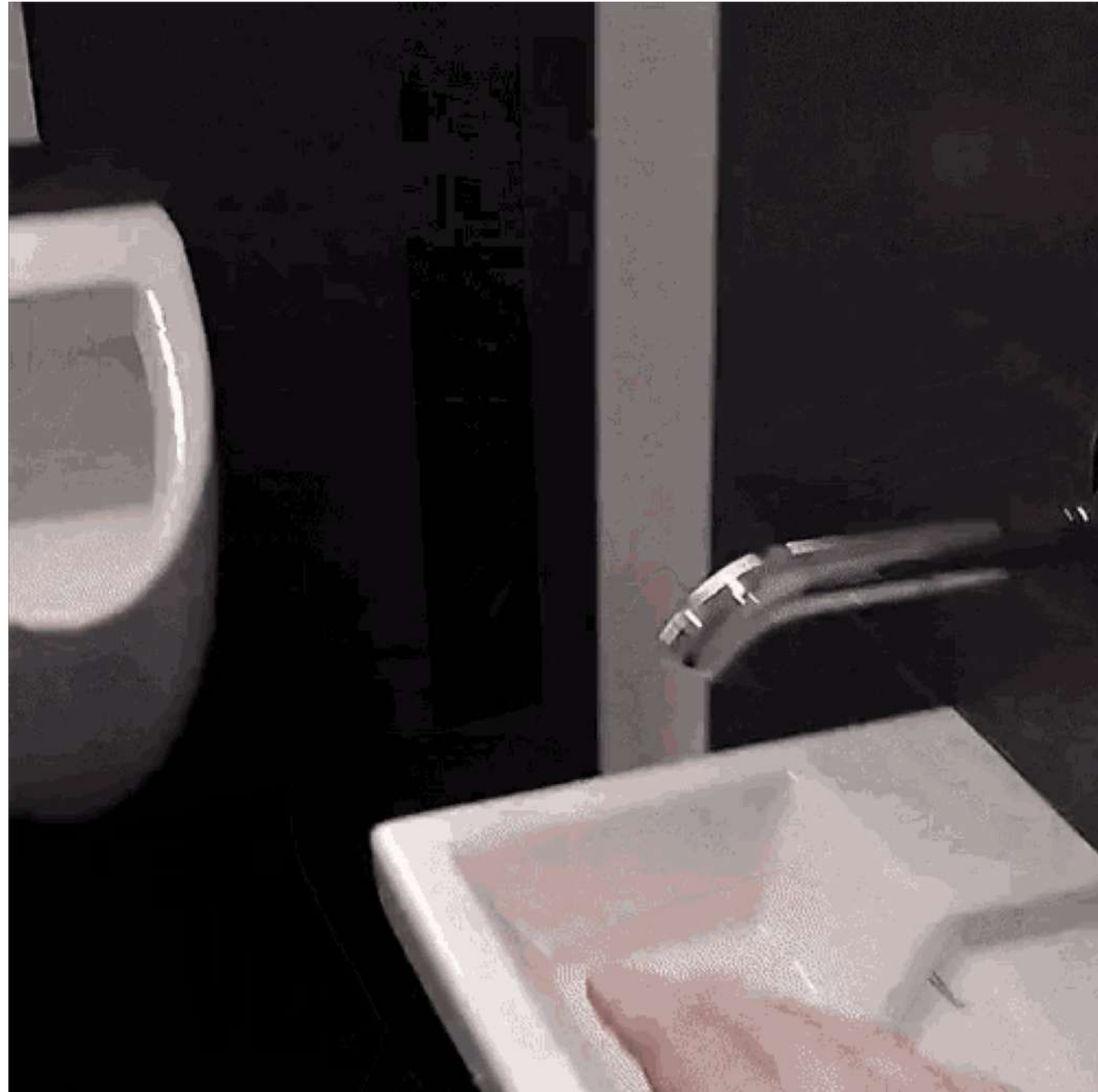


Tests



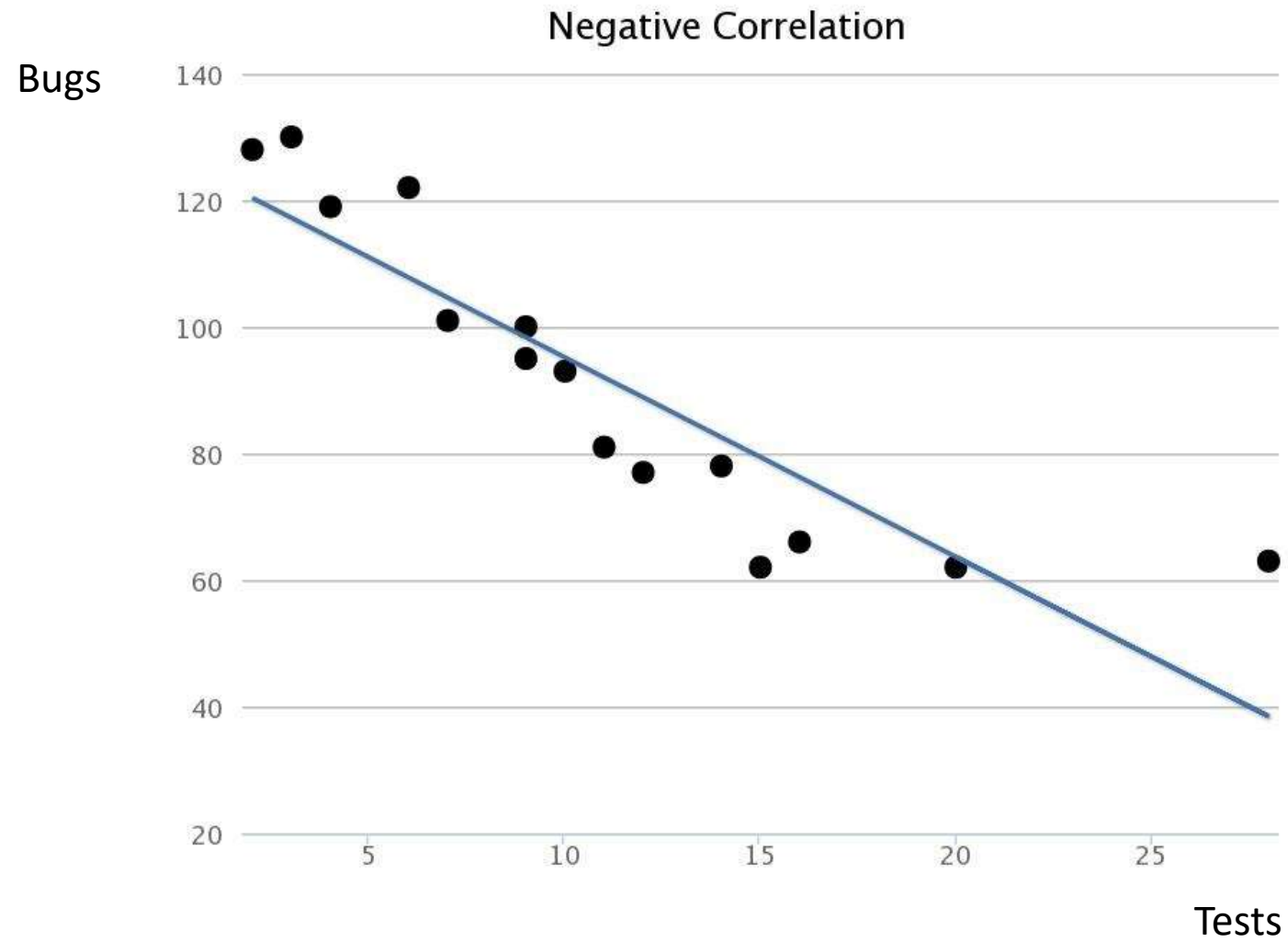


Tests





Tests



Code coverage



Tests

```
288      this.imagePlus = imagePlus;
289
290      this.stackMin = stackMin;
291      this.stackMax = stackMax;
292
293      this.bidirectional = bidirectional;
294      this.definedGoal = definedGoal;
295      this.startPaused = startPaused;
296
297      this.imageType = imagePlus.getType();
298
299      width = imagePlus.getWidth();
300      height = imagePlus.getHeight();
301      depth = imagePlus.getStackSize();
302
303      {
304          ImageStack s = imagePlus.getStack();
305          switch(imageType) {
306              case ImagePlus.GRAY8:
307              case ImagePlus.COLOR_256:
308                  slices_data_b = new byte[depth][];
309                  for( int z = 0; z < depth; ++z )
310                      slices_data_b[z] = (byte []) s.getPixels( z + 1 );
311                  break;
312              case ImagePlus.GRAY16:
313                  slices_data_s = new short[depth][];
314                  for( int z = 0; z < depth; ++z )
315                      slices_data_s[z] = (short []) s.getPixels( z + 1 );
316                  break;
317              case ImagePlus.GRAY32:
318                  slices_data_f = new float[depth][];
319                  for( int z = 0; z < depth; ++z )
320                      slices_data_f[z] = (float []) s.getPixels( z + 1 );
321                  break;
322          }
323      }
324
325      Calibration calibration = imagePlus.getCalibration();
326
327      x_spacing = (float)calibration.pixelWidth;
```



Code coverage



Tests



Code coverage



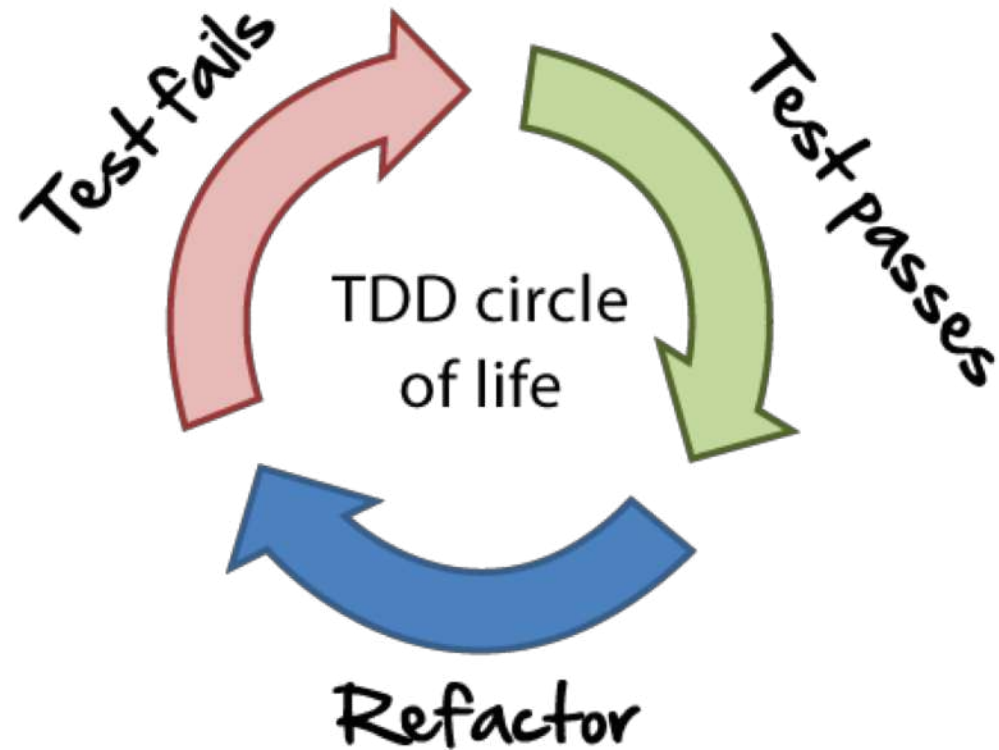
Tests



Test Driven Development



Tests



Test Driven Development



Tests

- Conception par spécification
- Code testable
- Documentation vivante
- Permet le refactoring
- Découper l'attention
- ...



Test Driven Development



Tests



Mutation testing



Tests



Mutation testi

Block Statement

Removes the content of every block statement. For example the code:

```
function saySomething() {  
  console.log('Hello world!');  
}
```

becomes

```
function  
{
```

Assignment Expression

Original	Mutated
+=	-=
	+=
	/=
	*=
	*=
	>>=
	<<=
&=	=
=	&=

Arithmetic Operator

Mutated
a - b
a + b
a / b
a * b
* b

Boolean Literal

Original	Mutated
false	true
true	false
a == b	a != b

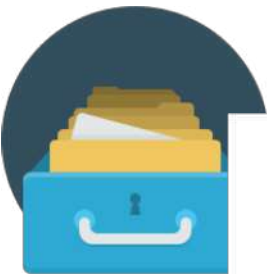
Logical Operator

Original	Mutated
a && b	a b
a b	a && b

Génération de mutants

Génération du rapport

Code source



/gameOfThronesSurvivalPredictor.js



```
1  import CacheManager from './cacheManager';
2
3  export function computeOrAddToCacheSurvivalScore(characterFamilyName) {
4      let survivalScore = 50;
5
6      if (characterFamilyName === "Stark") {
7          survivalScore = 8;
8      } else {
9          CacheManager.addToCache({
10             characterFamilyName
11          });
12      }
13
14      return survivalScore;
15  }
16
```



/gameOfThronesSurvivalPredictor.spec.js



```
1  import { expect } from 'chai';
2
3  import { computeOrAddToCacheSurvivalScore } from '../../server/testStryker/
4
5  export default function () {
6
7      describe('computeOrAddToCacheSurvivalScore()', function () {
8
9          it('should return the survival score', function () {
10             const youngStarkScore = computeOrAddToCacheSurvivalScore("Stark");
11             expect(youngStarkScore).to.equal(8);
12
13             const oldStarkScore = computeOrAddToCacheSurvivalScore("Lannister");
14             expect(oldStarkScore).to.equal(50);
15
16             });
17
18         });
19
20     }
```



/gameOfThronesSurvivalPredictor.js



```
1  import CacheManager from './cacheManager';
2
3  export function computeOrAddToCacheSurvivalScore(characterFamilyName) {
4      let survivalScore = 50;
5
6      if (characterFamilyName === "Stark") {
7          survivalScore = 8;
8      } else {
9          CacheManager.get(characterFamilyName)
10             .then((characterFamilyName) => {
11                 // ...
12             });
13
14         return survivalScore;
15     }
16 }
```

Couverture de code : 100%



gameOfThronesSurvivalPredictor.js – Stryker report



[All files](#) / gameOfThronesSurvivalPredictor.js

File / Directory	Mutation score	# Killed		# Survived		# Timeout		# No coverage		# Runtime errors		# Transpile errors		Total detected		Total undetected		Total mutants	
gameOfThronesSurvivalPredictor.js	<div><div>75.00 %</div></div>	6	2	0	0	0	0	0	0	6	2			8					

☒ Survived (2) ☐ Killed (6) [Expand all](#)

```
import CacheManager from './cacheManager';

export function computeOrAddToCacheSurvivalScore(characterFamillyName) {
  let survivalScore = 50;

  if (characterFamillyName === "Stark") {
    survivalScore = 8;
  } else 6 {} {
    CacheManager.addToCache(7 {
      characterFamillyName
    });
  }

  return survivalScore;
}
```

#	Mutator	State	Location	Original	Replacement
0	Block	Killed	2 : 71	{ ... ; }	}
1	IfStatement	Killed	5 : 8	=== " "	
2	IfStatement	Killed	5 : 8	=== " "	
3	BinaryExpression	Killed	5 : 8	=== " " !	=== " "
4	StringLiteral	Killed	5 : 33	" "	""
5	Block	Killed	5 : 42	{ ... }	{ }
6	Block	Survived	7 : 11	{ ... }	{ }
7	ObjectLiteral	Survived	8 : 32	{ ... }	{ }



```
1 import { expect } from 'chai';
2 import sinon from 'sinon';
3
4 import { computeOrAddToCacheSurvivalScore } from '../../server/testStryker/gameOfThronesSurvivalPredictor';
5 import CacheManager from '../../server/testStryker/cacheManager';
6
7 const addToCacheSpy = sinon.spy(CacheManager, 'addToCache');
8
9 export default function () {
10
11   describe('computeOrAddToCacheSurvivalScore()', function () {
12
13     it("should return the survival score", function () {
14       const starkScore = computeOrAddToCacheSurvivalScore("Stark");
15       expect(starkScore).to.equal(8);
16
17       const lannisterScore = computeOrAddToCacheSurvivalScore("Lannister");
18       expect(lannisterScore).to.equal(50);
19     });
20
21     it("should call the addToCache method of the CacheManager if the character name doesn't exist yet", function () {
22
23       computeOrAddToCacheSurvivalScore("Lannister");
24
25       expect(addToCacheSpy).to.have.been.called;
26       expect(addToCacheSpy).to.have.been.calledWithMatch({
27         characterFamilyName: "Lannister"
28       });
29     });
30   });
31
32 });
33
34 }
```

Ajout d'un nouveau cas de test

gameOfThronesSurvivalPredictor.js – Stryker report



[All files](#) / gameOfThronesSurvivalPredictor.js

File / Directory	Mutation score												
		# Killed	# Survived	# Timeout	# No coverage	# Runtime errors	# Transpile errors	Total detected	Total undetected	Total mutants			
gameOfThronesSurvivalPredictor.js	<div><div>100.00 %</div></div>	100.00	8	0	0	0	0	0	0	8	0	8	

☐ Killed (8) [Expand all](#)

```
import CacheManager from './cacheManager';

export function computeOrAddToCacheSurvivalScore(characterFamilyName) {
  let survivalScore = 50;

  if (characterFamilyName === "Stark") {
    survivalScore = 8;
  } else {
    CacheManager.addToCache({
      characterFamilyName
    });
  }

  return survivalScore;
}
```

#	Mutator	State	Location	Original	Replacement
0	Block	Killed	2 : 71	{ ... ; }	{ }
1	IfStatement	Killed	5 : 8	=== " "	
2	IfStatement	Killed	5 : 8	=== " "	
3	BinaryExpression	Killed	5 : 8	=== " "	!== " "
4	StringLiteral	Killed	5 : 33	" "	""
5	Block	Killed	5 : 42	{ ... }	{ }
6	Block	Killed	7 : 11	{ ... }	{ }
7	ObjectLiteral	Killed	8 : 32	{ ... }	{ }



Tests

Mutation testing : tools

- C / C++
- C#
- Clojure
- Go
- Java
- Javascript
- PHP
- Python
- Ruby
- Rust
- Scala
- Swift

<https://github.com/theofidry/awesome-mutation-testing>



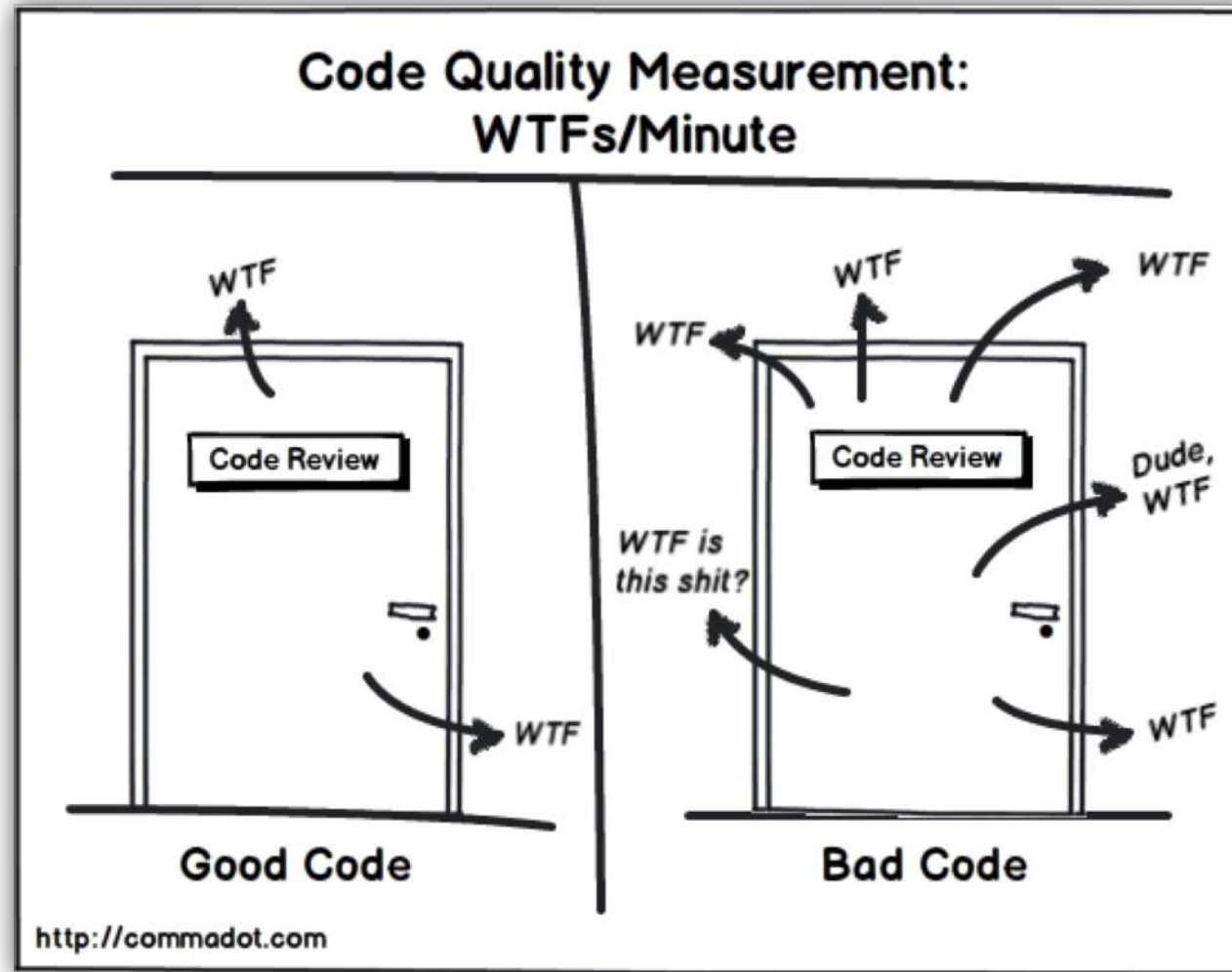


Clean code





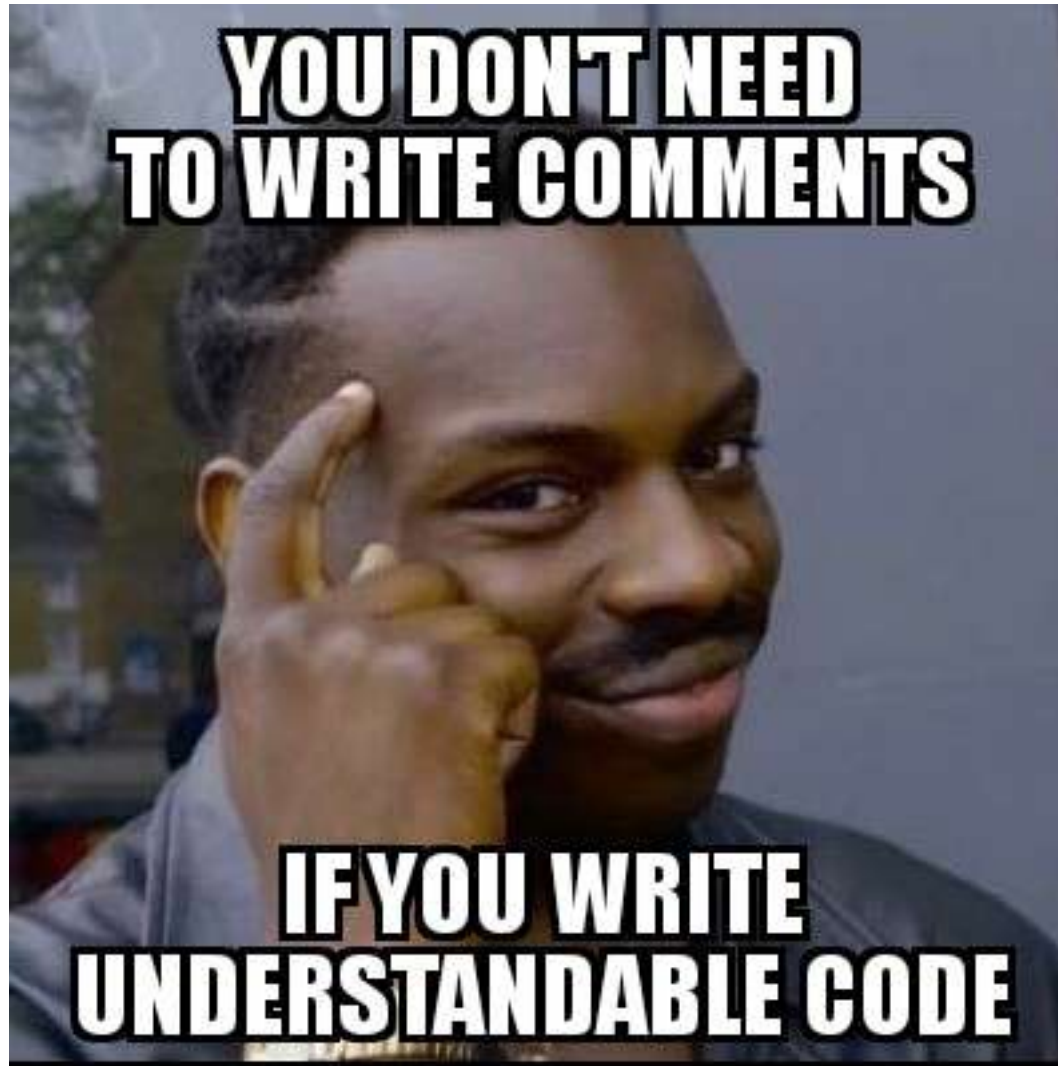
Clean code



Principes de base



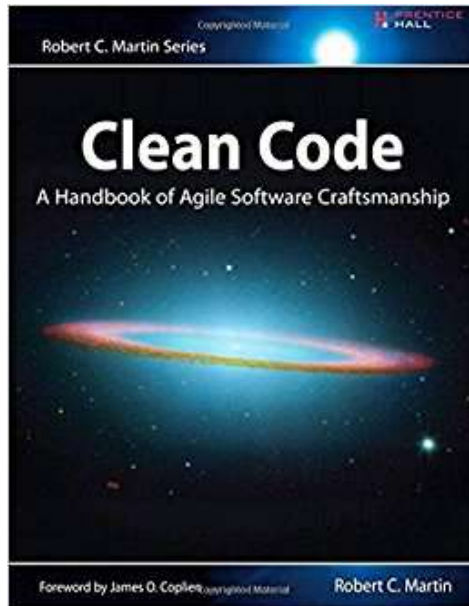
Clean code



Code smells



Clean code



- Taille des méthodes
- Complexité des méthodes
- Taille des fichiers
- Nommage
- Duplication de code
- ...



Code smells : impact



Clean code

```
class Shop {
  constructor(items=[]){
    this.items = items;
  }
  updateQuality() {
    for (var i = 0; i < this.items.length; i++) {
      if (this.items[i].name != 'Aged Brie' && this.items[i].name != 'Backstage passes to a TAFKAL80ETC concert') {
        if (this.items[i].quality > 0) {
          if (this.items[i].name != 'Sulfuras, Hand of Ragnaros') {
            this.items[i].quality = this.items[i].quality - 1;
          }
        }
      } else {
        if (this.items[i].quality < 50) {
          this.items[i].quality = this.items[i].quality + 1;
          if (this.items[i].name == 'Backstage passes to a TAFKAL80ETC concert') {
            if (this.items[i].sellIn < 11) {
              if (this.items[i].quality < 50) {
                this.items[i].quality = this.items[i].quality + 1;
              }
            }
          }
          if (this.items[i].sellIn < 6) {
            if (this.items[i].quality < 50) {
              this.items[i].quality = this.items[i].quality + 1;
            }
          }
        }
      }
    }
  }
}
```

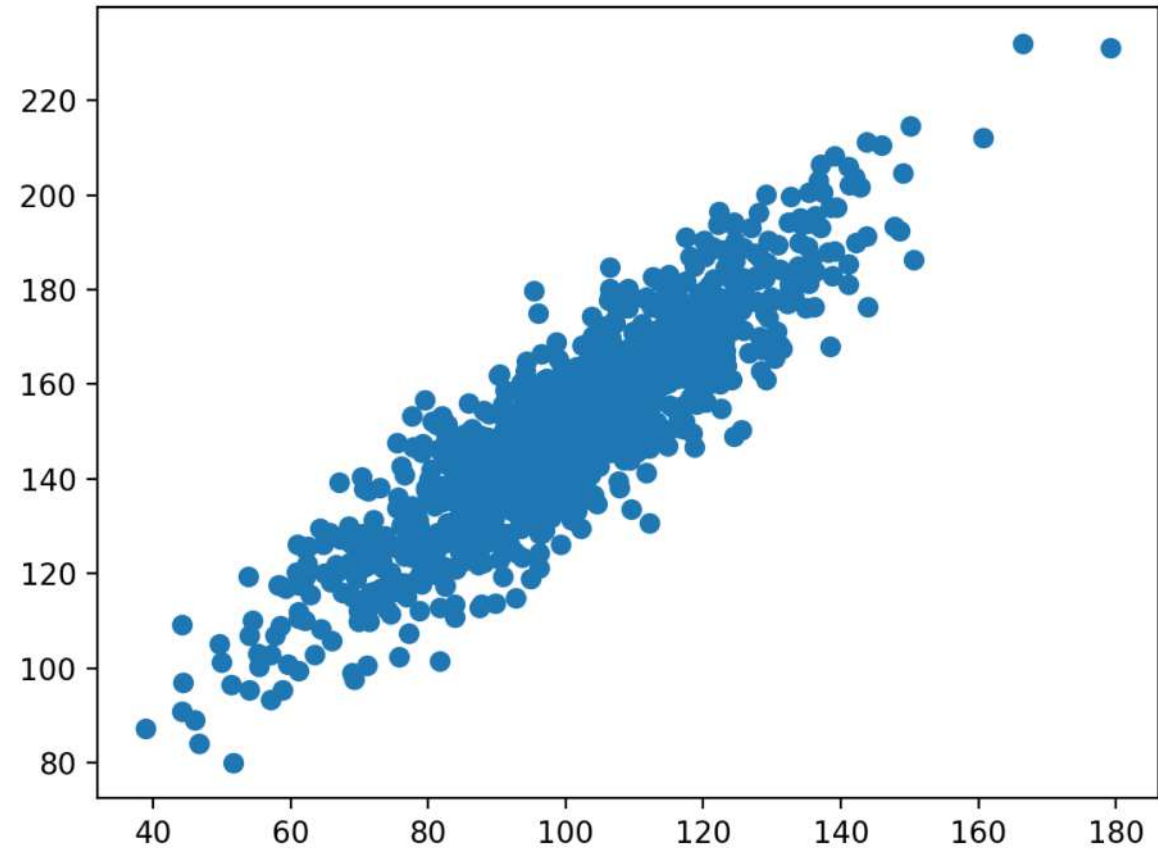


Code smells : impact



Clean code

Bugs



Code smells



Code smells : détection



Clean code

```
function updateArrowProperty() {
  for (var i in $scope.filesOrders) {
    if ($scope.filesOrders.hasOwnProperty(i)) {
      if (i === $scope.filesOrder) {
        if ($scope.filesReverse) {
```

Refactor this code to not nest more than 3 if/for/while/switch/try statements. ... last year ▾ L62 3 🔗

🔗 Code Smell ▾ 🚩 Major ▾ 🔵 Open ▾ Not assigned ▾ 10min effort Comment 🧠 brain-overload ▾

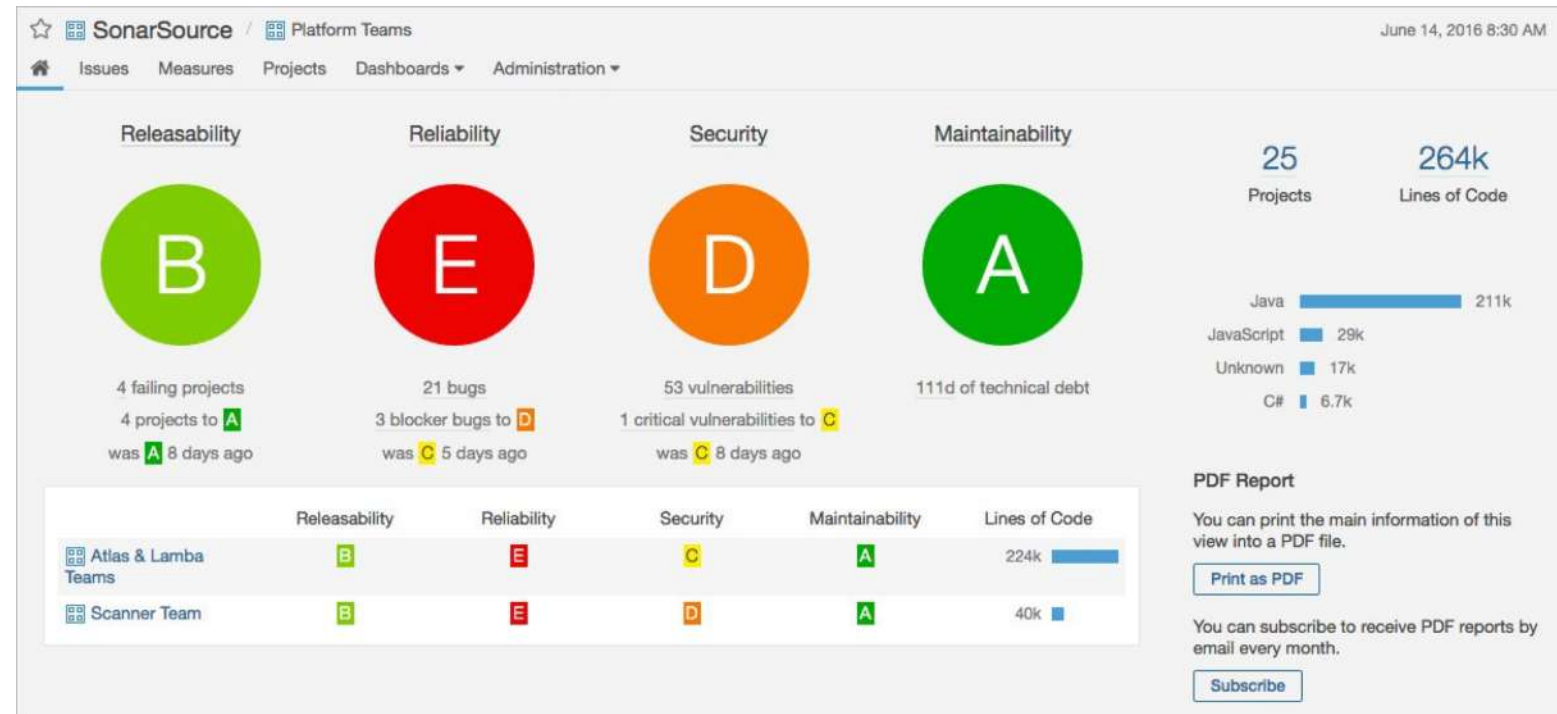
```
        $scope.filesOrders[i] = "active-reversed-filter";
      } else {
        $scope.filesOrders[i] = "active-filter";
```



Dette technique



Clean code



Dette technique : impact sur le projet



Clean code

Plus elle est grande :

- Plus le code est difficile à comprendre
- Plus la maintenance est ralentie
- Plus il y a de risques de bugs



Dette technique : impact sur le projet



Clean code

Bug de la gare Montparnasse à la SNCF :
=> plusieurs millions d'euros

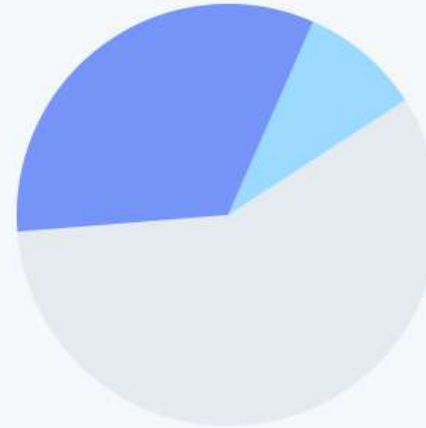


Dette technique : impact sur le projet



Clean code

LES DÉVELOPPEURS CONSACRENT 42 % DE LEUR
TEMPS À LA MAINTENANCE DE CODE



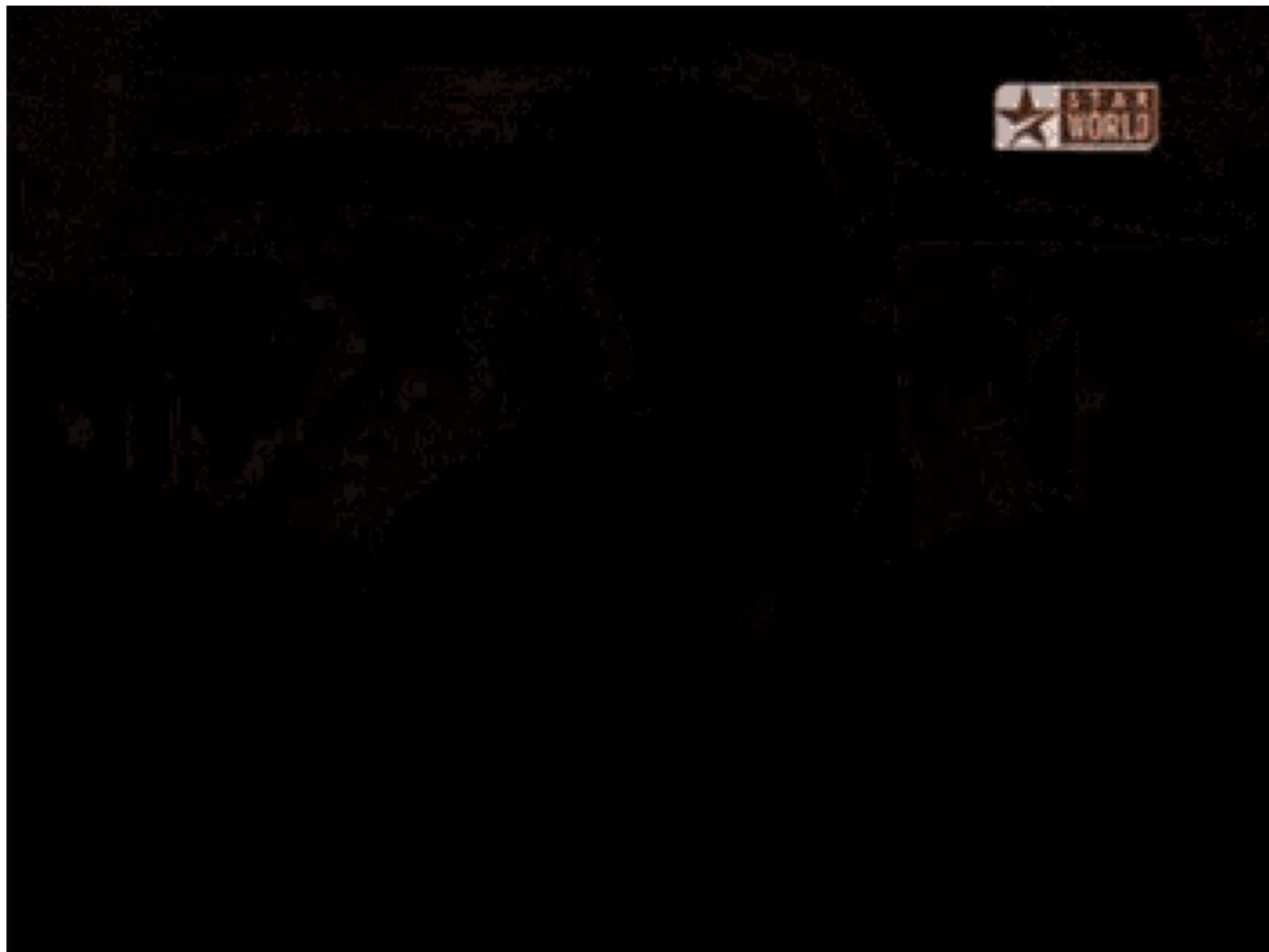
13,5 heures
Dette
technique

3,8 heures
Code de
mauvaise
qualité

Nombre total d'heures
41,1

Semaine de travail moyenne
d'un développeur



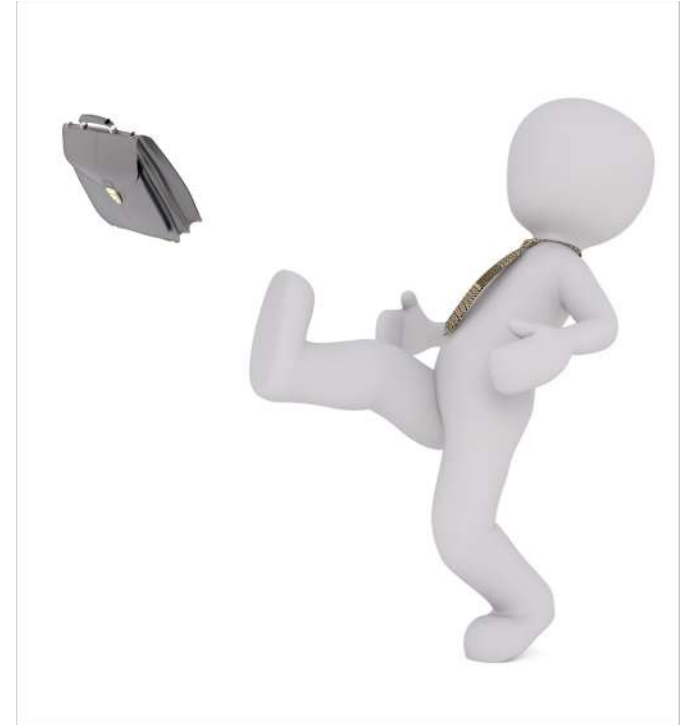


Dettes techniques : impact sur les développeurs



Clean code

- Découragement
- Lenteur
- Perte de créativité
- Érosion des compétences



Dette technique : impact sur les développeurs



Clean code

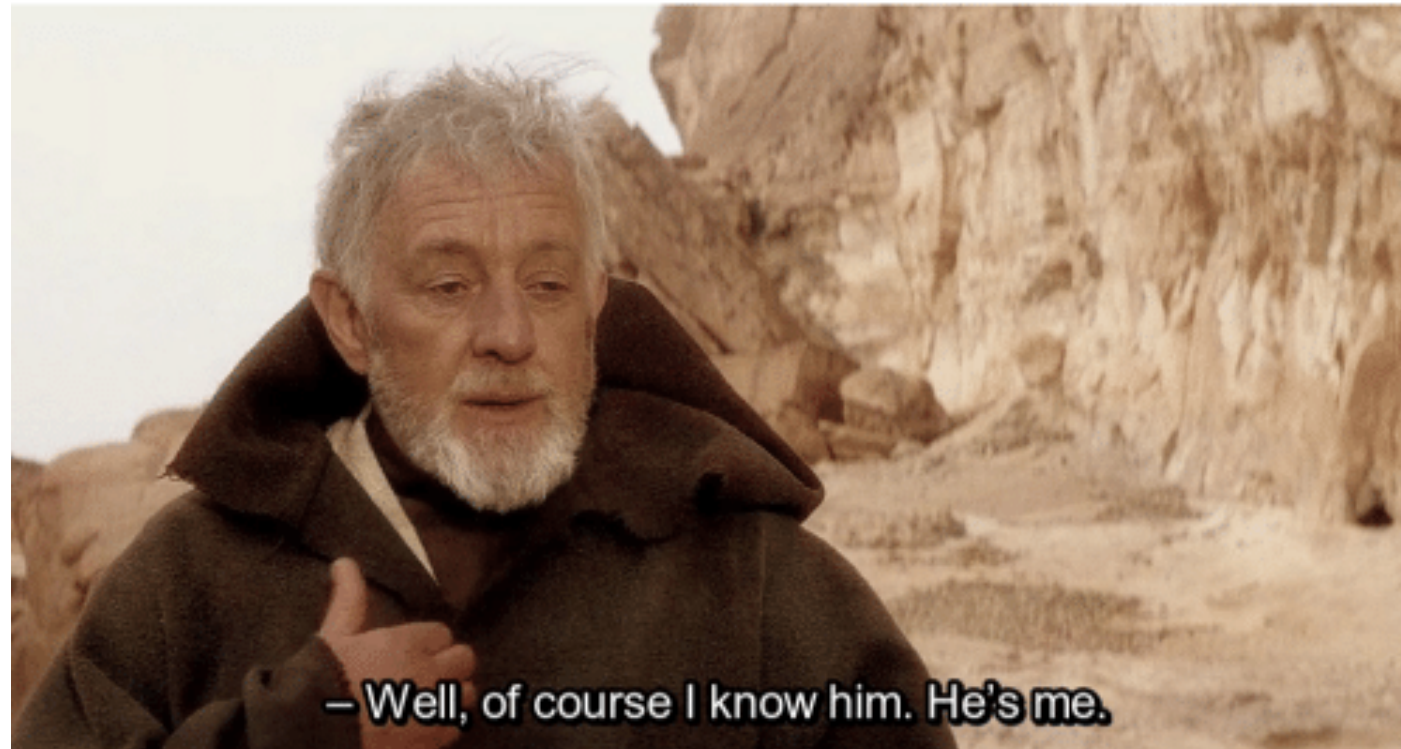


Pourquoi cette dette technique augmente ?

when your coworkers ask if you
know who has been writing bad
code all this time



Clean code



– Well, of course I know him. He's me.

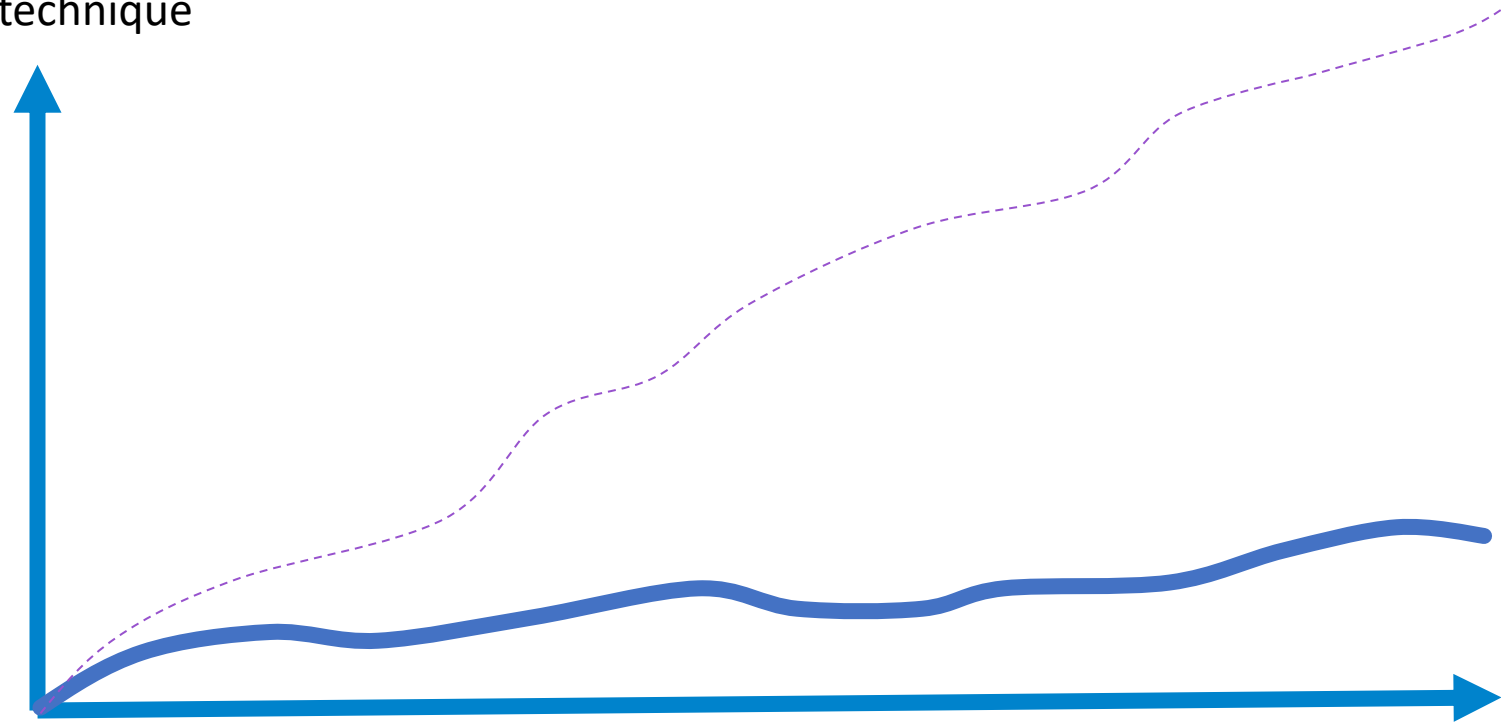


Comment maintenir un bon niveau de qualité ?



Clean code

Dette technique



Temps



A LANNISTER ALWAYS

PAYS HIS DEBTS

memegenerator.net

Revue de code



Egoless Programming

« Vous n'êtes pas votre code »

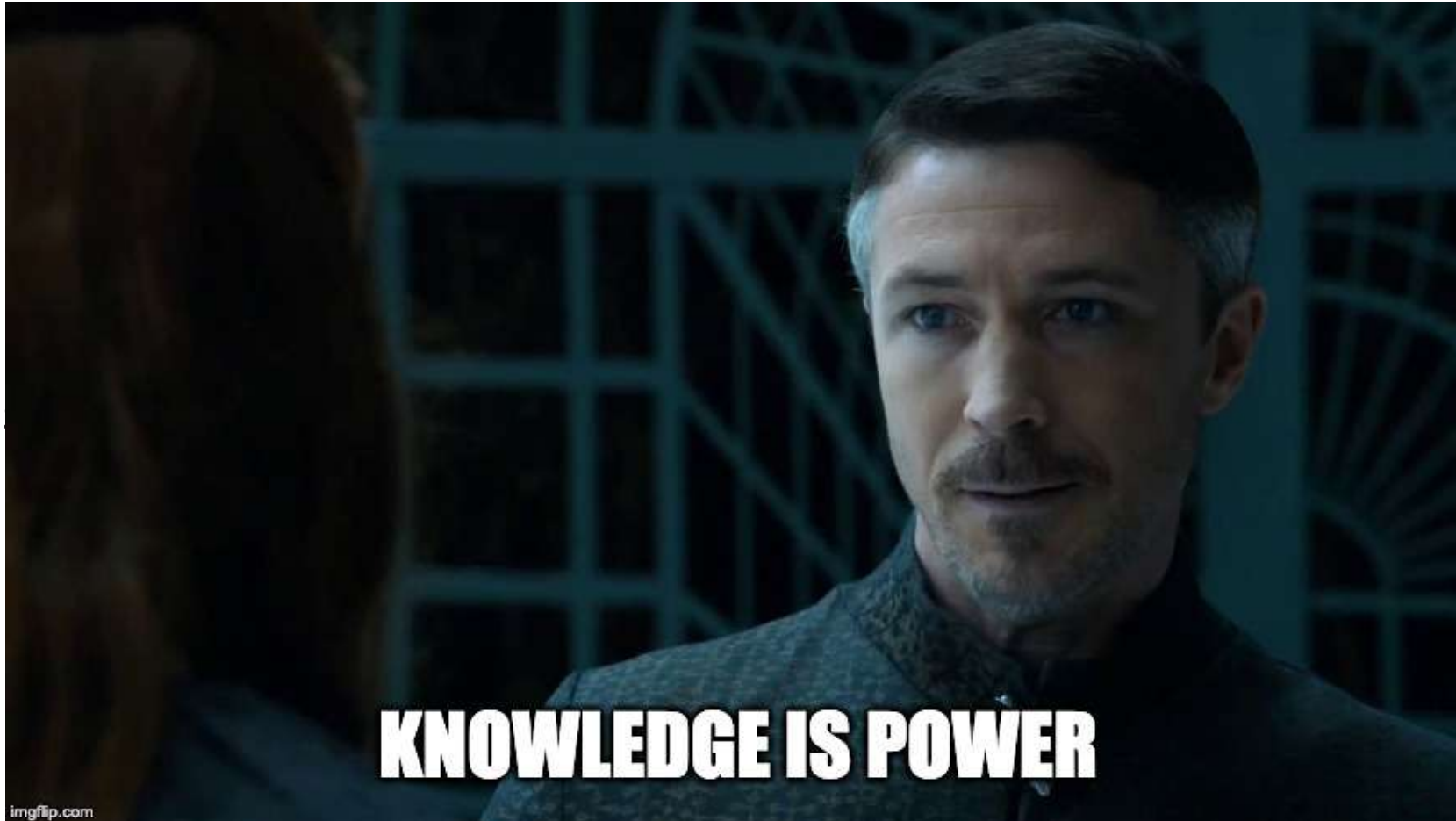
« La seule autorité acceptable vient de la connaissance, non du pouvoir »

The Psychology of Computer Programming ,1971



Egoless Programming

« La



dir »

mming ,1971



Manifesto for Software Craftsmanship

Elever le niveau

En tant qu'aspirants Artisans du Logiciel, nous relevons le niveau du développement professionnel de logiciels par la pratique et en aidant les autres à acquérir le savoir-faire. Grâce à ce travail, nous avons appris à apprécier :

Pas seulement des logiciels opérationnels,
mais aussi **des logiciels bien conçus.**

Pas seulement l'adaptation aux changements,
mais aussi **l'ajout constant de la valeur.**

Pas seulement les individus et leurs interactions,
mais aussi **une communauté de professionnels.**

Pas seulement la collaboration avec les clients,
mais aussi **des partenariats productifs.**

Avoir des actions positives sur le code





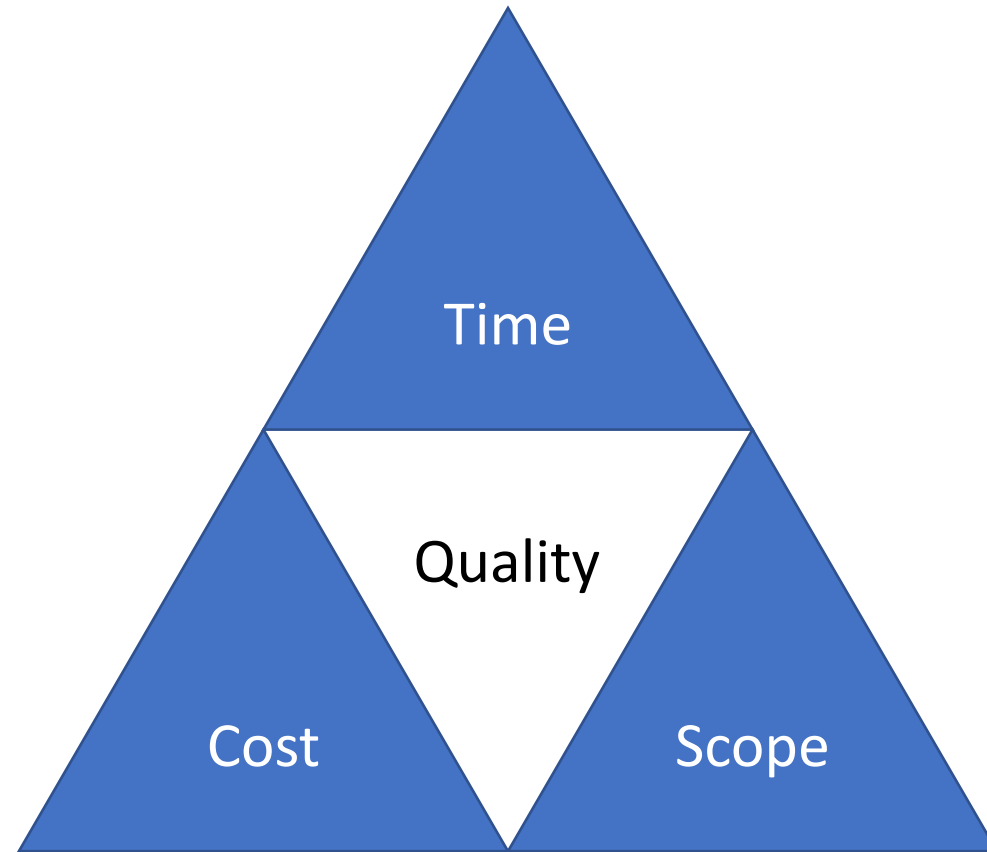
Comment est pilotée la stratégie IT de l'entreprise ?



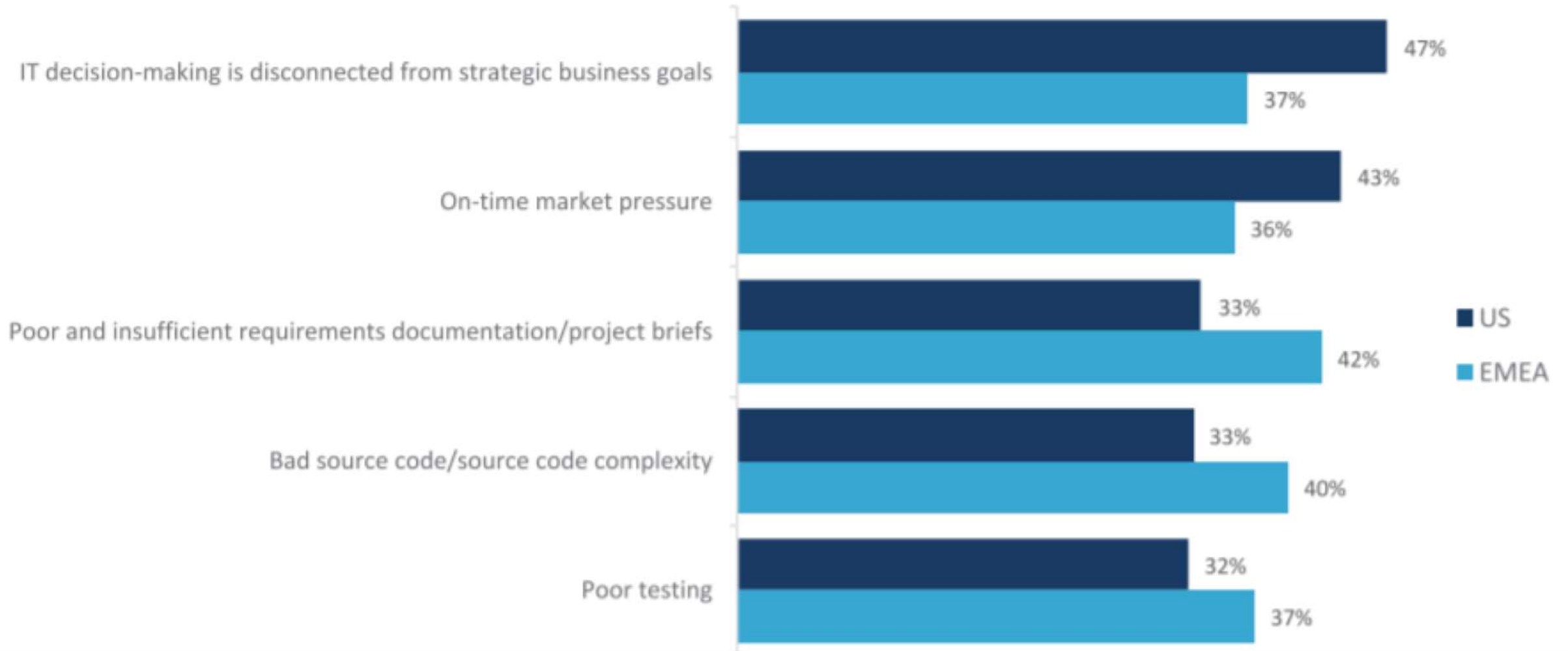
Le problème de la boîte noire



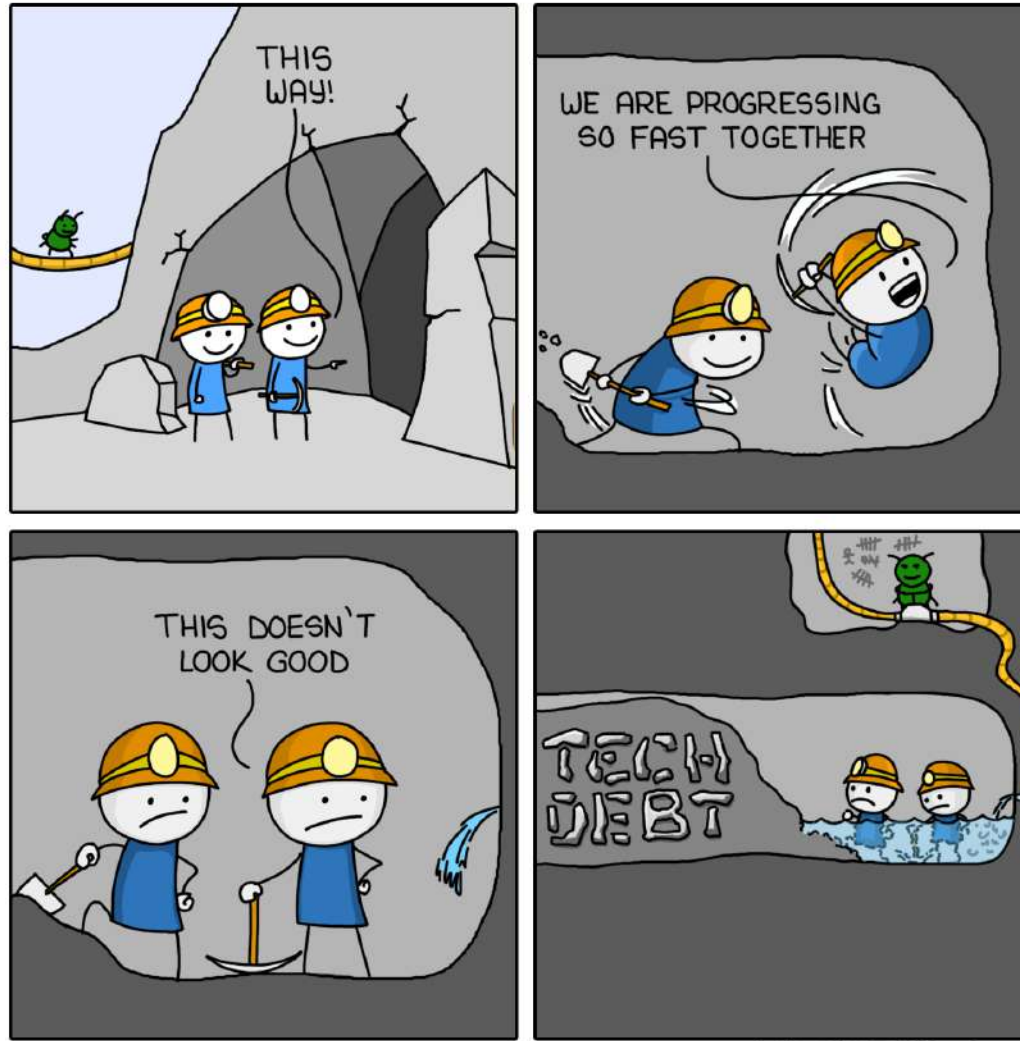
Quels sont les indicateurs de base ?



Sources of Technical Debt by Region



TECH DEBT



MONKEYUSER.COM





Quels sont les indicateurs sur la qualité du projet ?

- Nombre de bugs

- Nombre de code smells



- Couverture de code

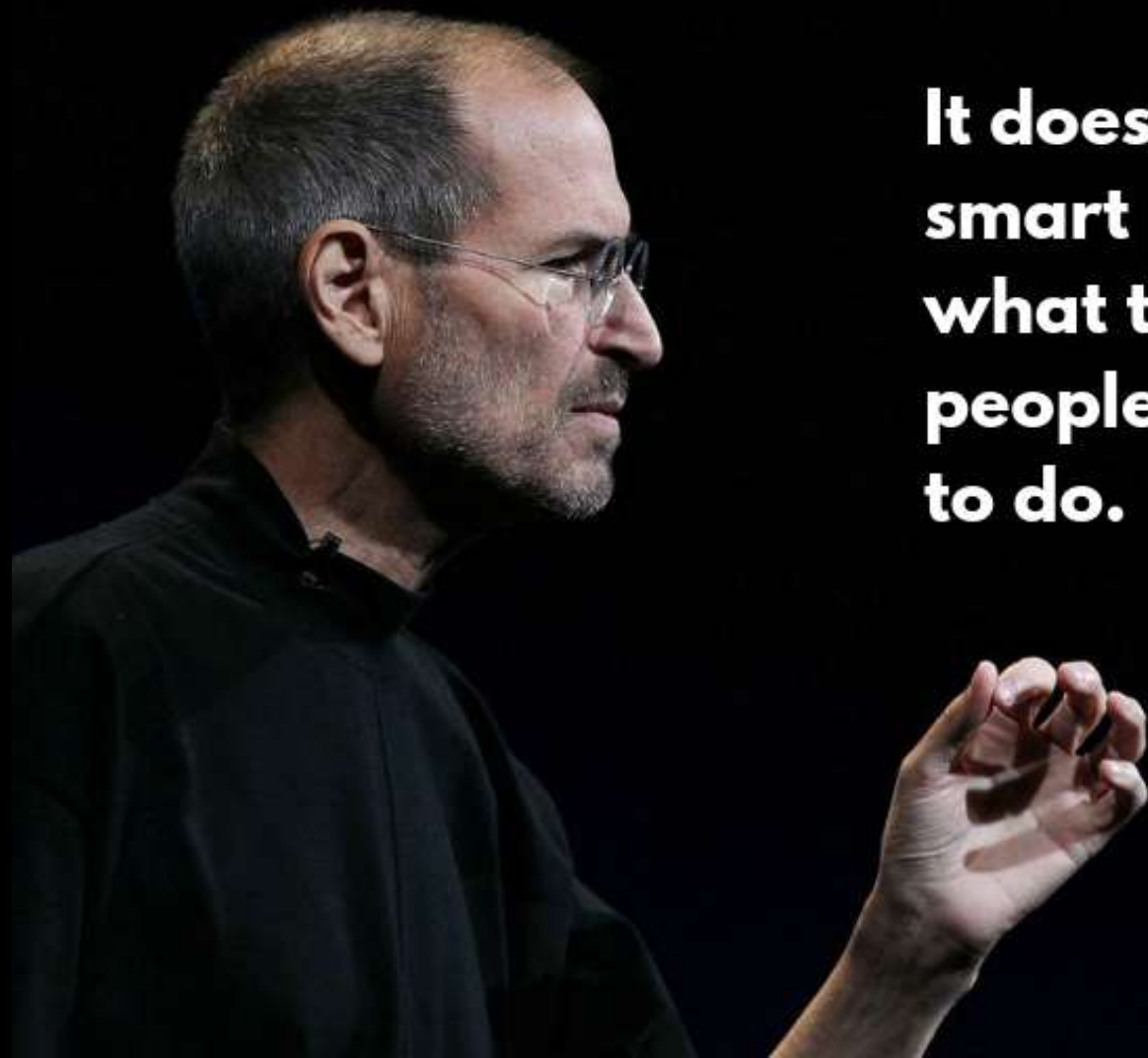


“When a measure becomes a target, it ceases to be a good measure.”

Goodhart’s Law







It doesn't make sense to hire smart people and tell them what to do; we hire smart people so they can tell us what to do.

- Steve Jobs

Tendance actuelle

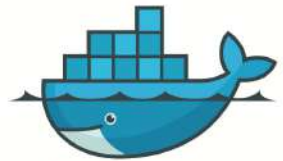
- Démarches autour de la **Qualité logicielle**
 - Réduction du risque
 - Réduction des coûts
 - Valorisation
 - RH



Tendance actuelle



Agile



docker + Jenkins



DevOps

sonarqube



Qualité logicielle



Réalité ?

- Hétérogénéité dans les équipes
- Taille de l'entreprise / Turn over / Remote
- Conduite du changement
- Problème de communication



La qualité logicielle : un objectif en 2019

- Apparition de rôle « Coach software craftsmanship »
- Organisation d'évènements
- Le but : la diffusion des bonnes pratiques
- Encore trop hétérogène entre les entreprises



The end

