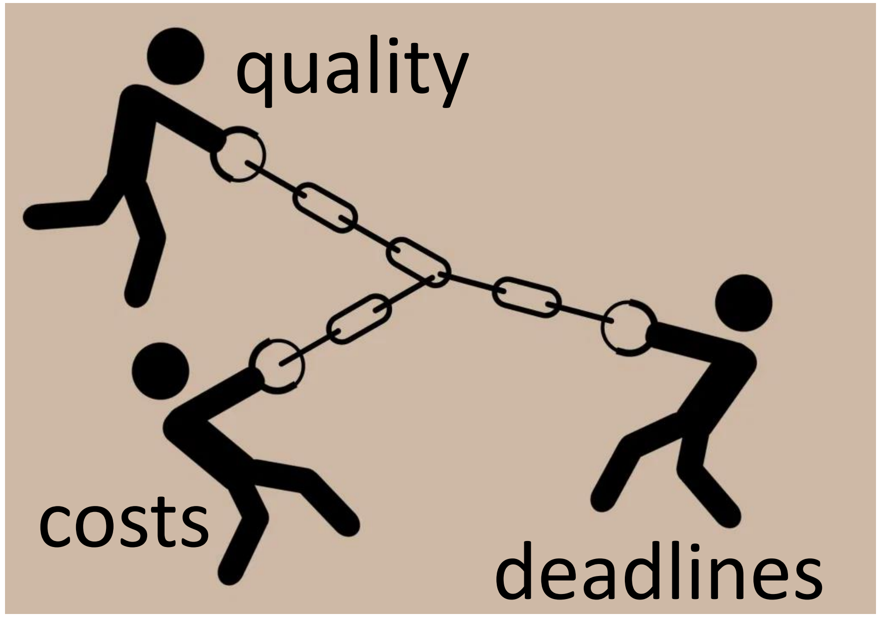
Software Application Engineering

Software Quality: criteria, measures and tools

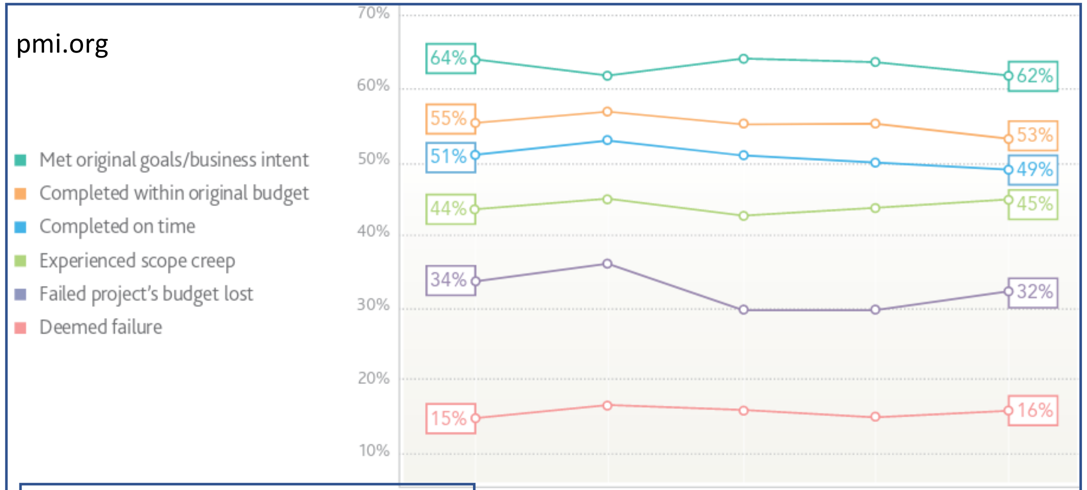
Goal(s)

"Success" of software projects = achievement of three conflicting goals

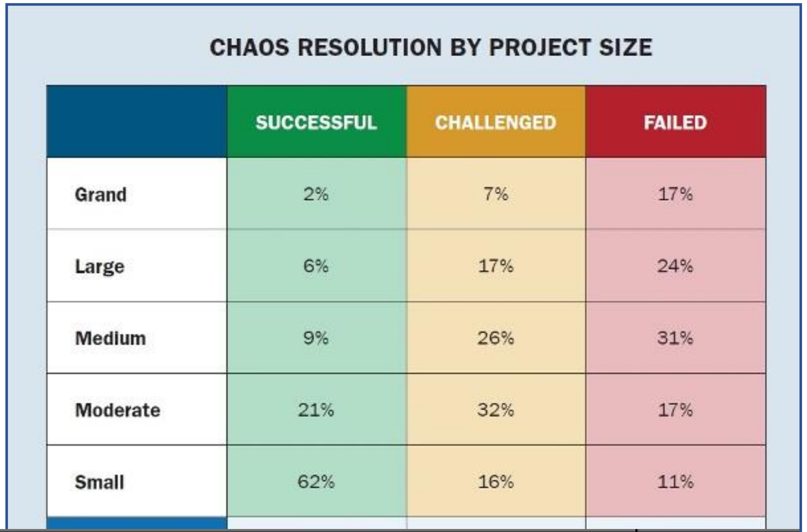
BON MAÎNTENANT IL FAUT DÉCIDER, C'EST LA DATE, LE COÛT OU LES FONCTIONNALITÉS...
BEN ON VEUT LES TROIS...
OK! ON FAIT...



Success rate of IT projects?

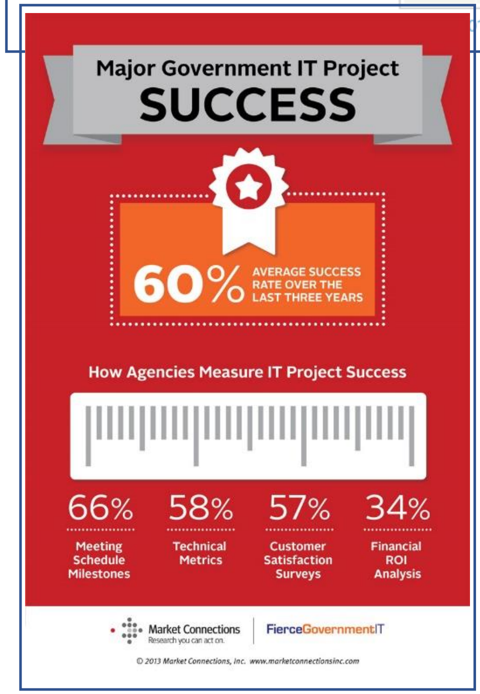


pmi.org



CHAOS RESOLUTION BY PROJECT SIZE

	SUCCESSFUL	CHALLENGED	FAILED
Grand	2%	7%	17%
Large	6%	17%	24%
Medium	9%	26%	31%
Moderate	21%	32%	17%
Small	62%	16%	11%



Major Government IT Project
SUCCESS

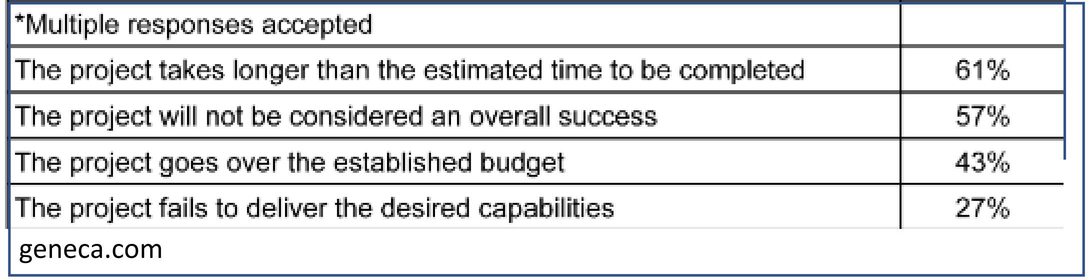
60% AVERAGE SUCCESS RATE OVER THE LAST THREE YEARS

How Agencies Measure IT Project Success

66%	58%	57%	34%
Meeting Schedule Milestones	Technical Metrics	Customer Satisfaction Surveys	Financial ROI Analysis

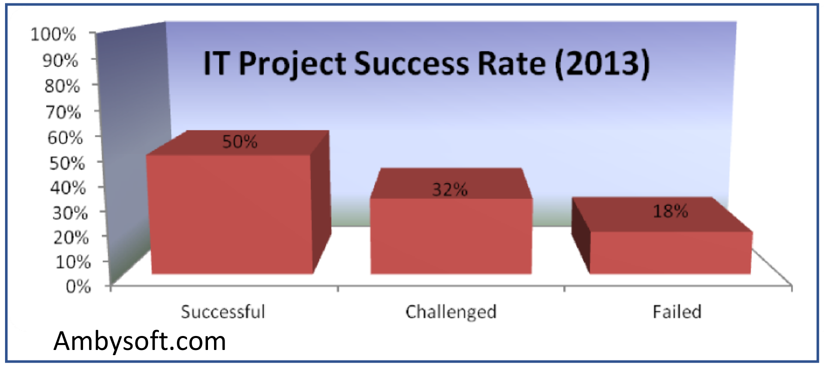
Market Connections Research you can act on | FierceGovernmentIT

© 2013 Market Connections, Inc. www.marketconnectionsinc.com

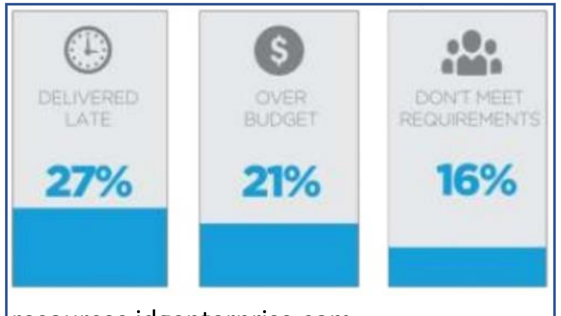


*Multiple responses accepted	
The project takes longer than the estimated time to be completed	61%
The project will not be considered an overall success	57%
The project goes over the established budget	43%
The project fails to deliver the desired capabilities	27%

geneca.com



Ambysoft.com

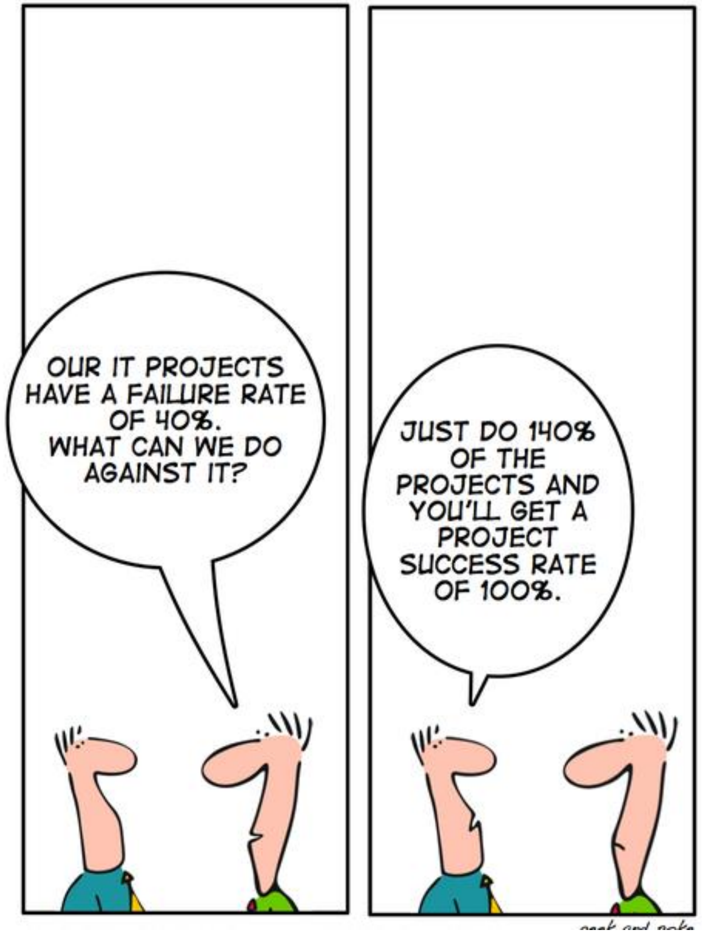


resources.idgenterprise.com

questions

- What is software quality?

- How to measure it?
- How to improve it?

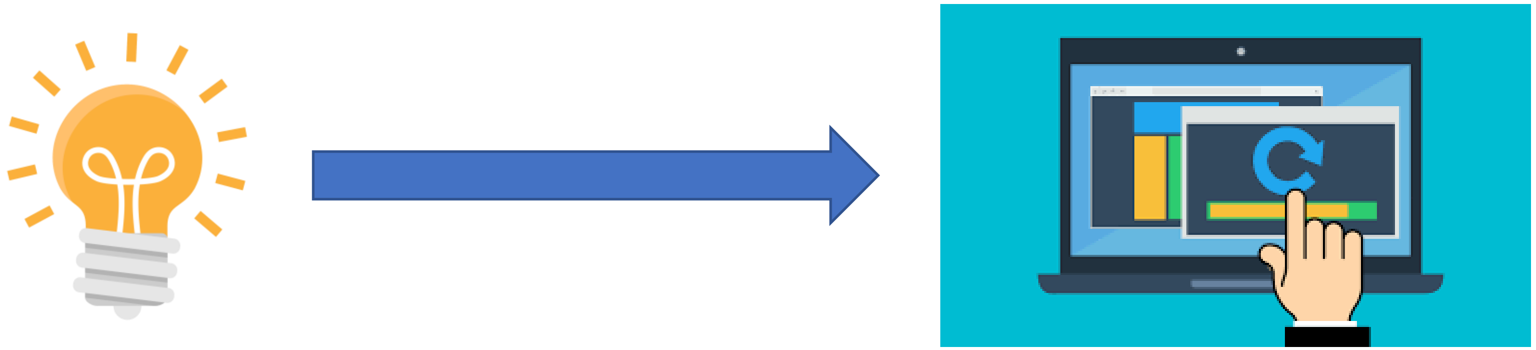

OUR IT PROJECTS HAVE A FAILURE RATE OF 40%. WHAT CAN WE DO AGAINST IT?
JUST DO 140% OF THE PROJECTS AND YOU'LL GET A PROJECT SUCCESS RATE OF 100%.
geek and poke

According to you,
what makes the quality of a software solution is its...

(20 characters, multiples answers are possible)

answergarden.ch/1356863

Software stakeholders

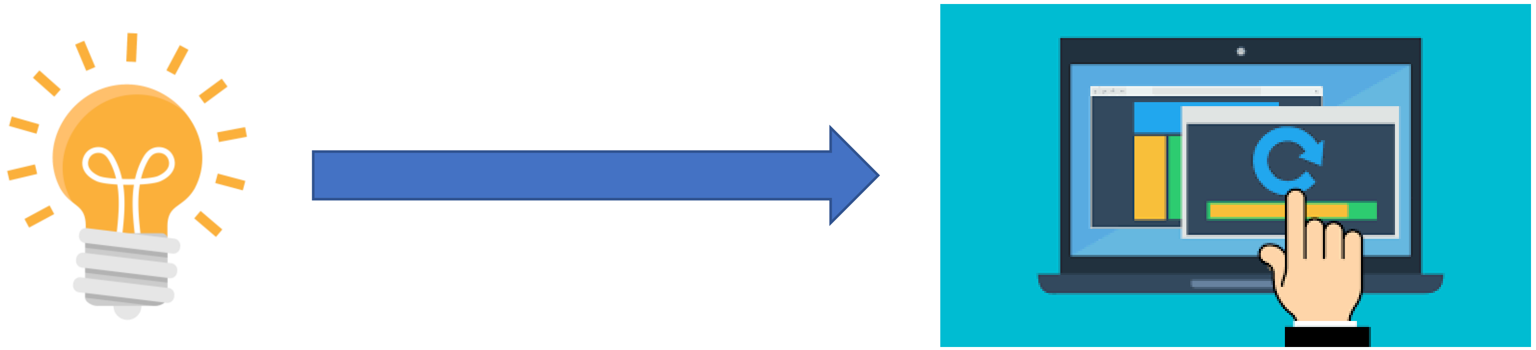


Sponsors / funders

Developers

Users

The 3 aspects of software quality



Process quality

- Meeting budgets
- Meeting delivery date
- Repeatable and sustainable development process

Structural quality

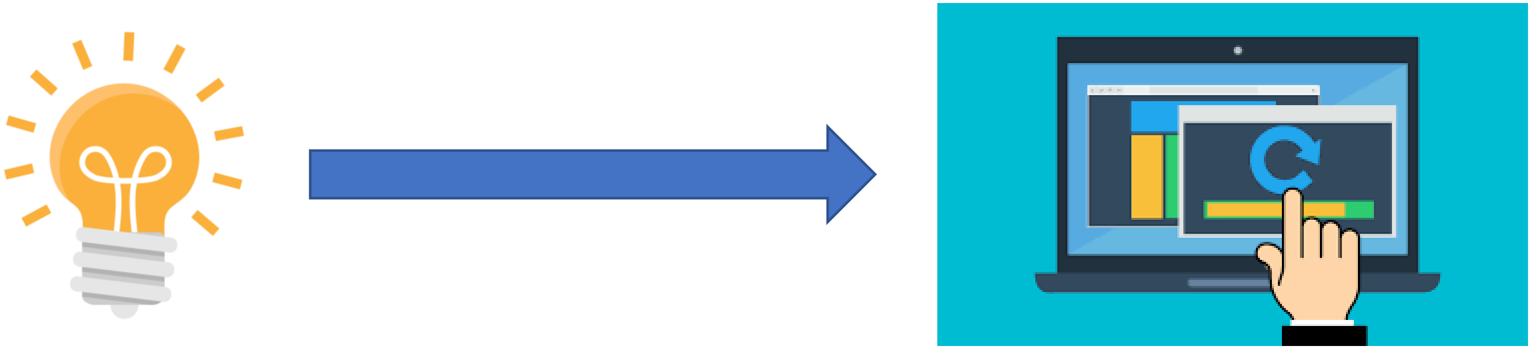
- Code testability
- Code maintenability
- Code understandability
- Code efficiency
- Code security

Functional quality

- Meets the requirements
- Few defects
- Good enough performance
- Easy to use
- Looks good

Adapted from D Chappell’s white paper “THE THREE ASPECTS OFSOFTWARE QUALITY: FUNCTIONAL, STRUCTURAL, AND PROCESS”

Different emphasis



Process quality

- Meeting budgets
- Meeting delivery date
- Repeatable and sustainable development process

Sponsors / funders

Structural quality

- Code testability
- Code maintenability
- Code understandability
- Code efficiency
- Code security

Developers

Functional quality

- Meets the requirements
- Few defects
- Good enough performance
- Easy to use
- Looks good

Users

ISO/IEC 25010 software quality-in-use (1/2)

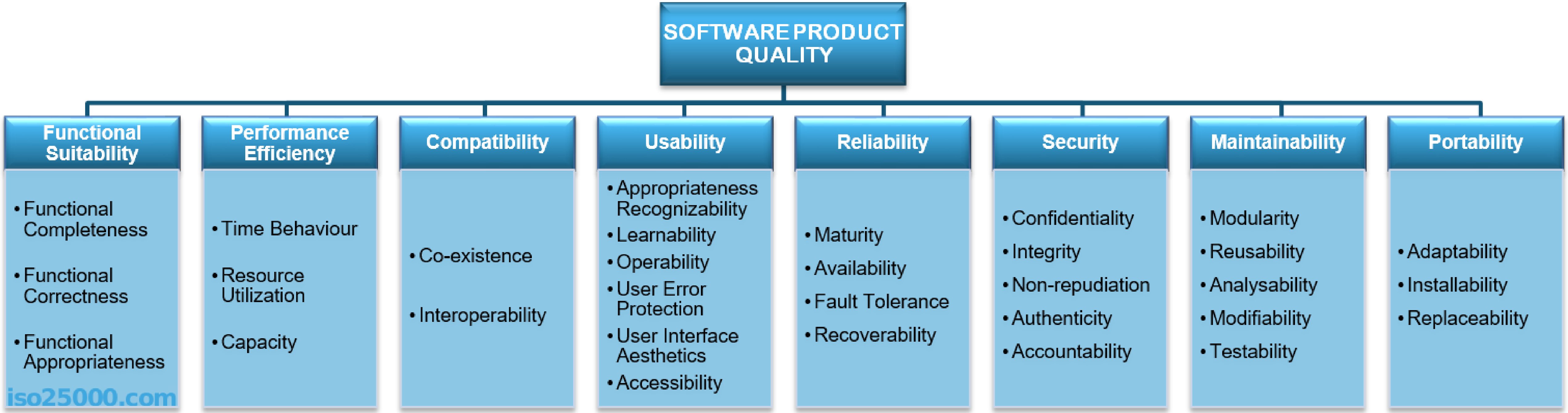
defines characteristics that relate to the outcome of interaction when a product is used in a particular context of use

- Effectiveness: accuracy and completeness with which users achieve specified goals
- Efficiency: resources expended in relation to goals achievements: time to complete the task (human resources), materials, financial cost of usage…
- Satisfaction: user’s response to interaction with the software
    - Usefulness: results/consequences of use achieve pragmatic goals
    - Trust: confidence that the software will behave as intended
    - Pleasure: pleasant experience resulting from the use of the software
    - Comfort: physical comfort

ISO/IEC 25010 software quality-in-use (2/2)

- Freedom from risk: degree to which a software mitigates potential risks
  - economic risk: risk to financial status, efficient operation, commercial property, reputation or other resources
  - health and safety risk
  - environmental risk
- Context coverage: degree to which a software can be used in both specified contexts of use and in contexts beyond those initially explicitly identified
  - Context completeness: software can be used in all the specified contexts of use. Example: the extent to which software is usable using a small screen, with low network bandwidth, by a non-expert user...
  - Flexibility: software can be used in contexts beyond those initially specified
    - by adapting the software for additional user groups, tasks and cultures
    - enables software to take account of circumstances, opportunities and individual preferences that had not been anticipated in advance
    - If a product is not designed for flexibility, it might not be safe to use the product in unintended contexts.

SQuaRE: *System and Software Quality Requirements and Evaluation* ISO/IEC 25010:2011 standard



SOFTWARE PRODUCT QUALITY

Functional Suitability

- Functional Completeness
- Functional Correctness
- Functional Appropriateness

Performance Efficiency

- Time Behaviour
- Resource Utilization
- Capacity

Compatibility

- Co-existence
- Interoperability

Usability

- Appropriateness Recognizability
- Learnability
- Operability
- User Error Protection
- User Interface Aesthetics
- Accessibility

Reliability

- Maturity
- Availability
- Fault Tolerance
- Recoverability

Security

- Confidentiality
- Integrity
- Non-repudiation
- Authenticity
- Accountability

Maintainability

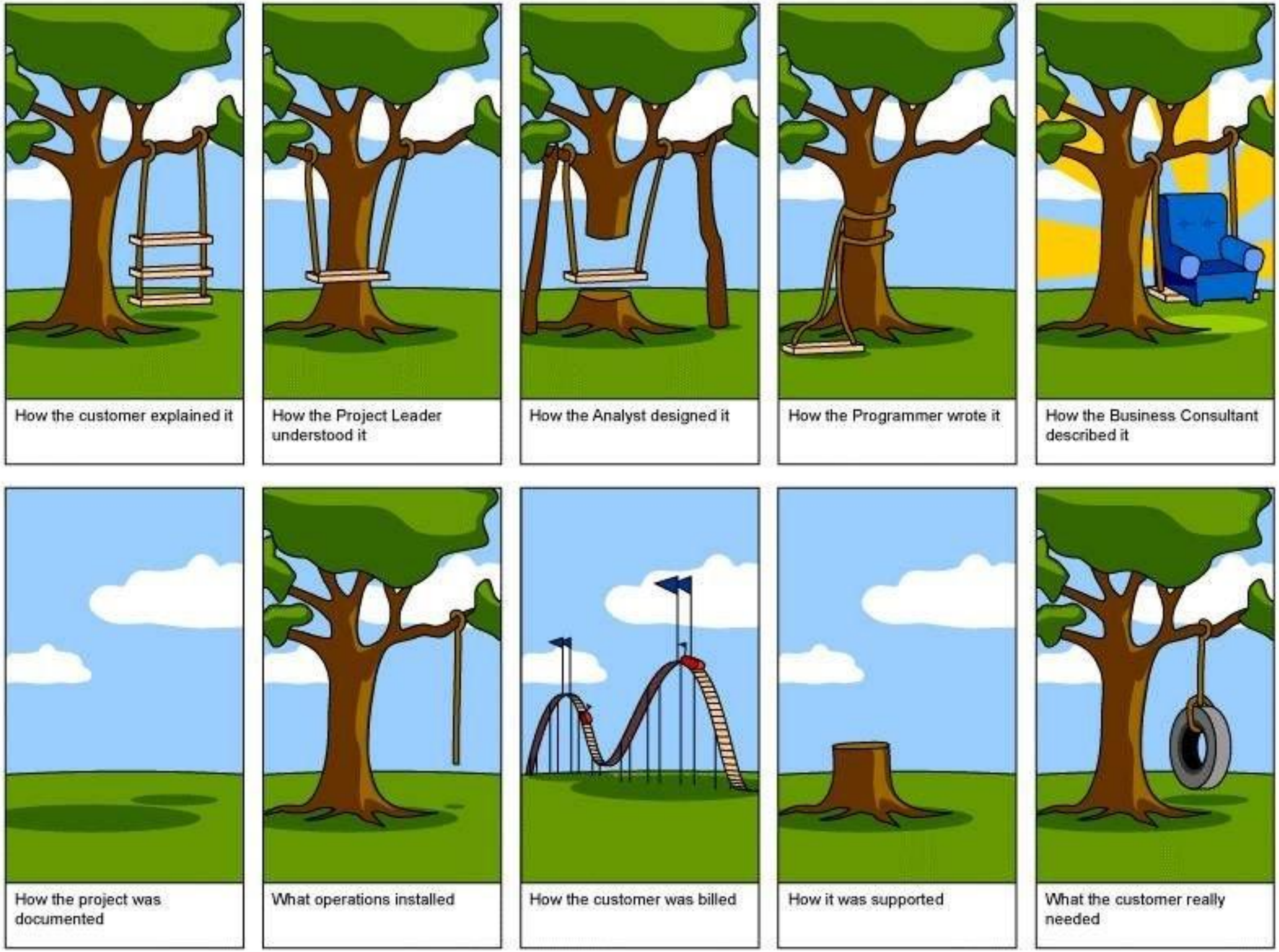
- Modularity
- Reusability
- Analysability
- Modifiability
- Testability

Portability

- Adaptability
- Installability
- Replaceability

Software quality criteria and actions

https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:en



How the customer explained it

How the Project Leader understood it

How the Analyst designed it

How the Programmer wrote it

How the Business Consultant described it

How the project was documented

What operations installed

How the customer was billed

How it was supported

What the customer really needed

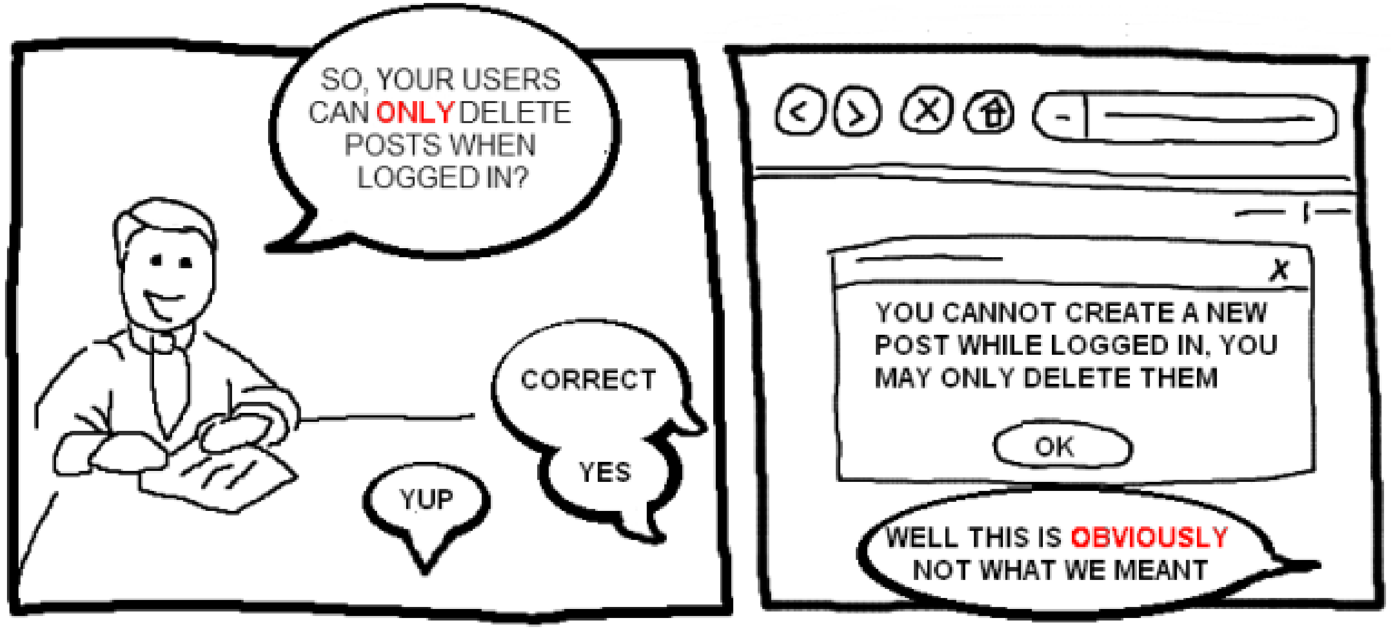
Functional suitability

meets the expressed and perceived needs, when used under specified conditions: *does it do what it is meant to do?*

- Completeness: covers all user's needs : *can we do everything we need to do?*
- Correctness: generates the right results with the required accuracy
- Appropriateness: facilitates the achievement of specific tasks and goals: *Is this the best way to do it?*

Difficulties

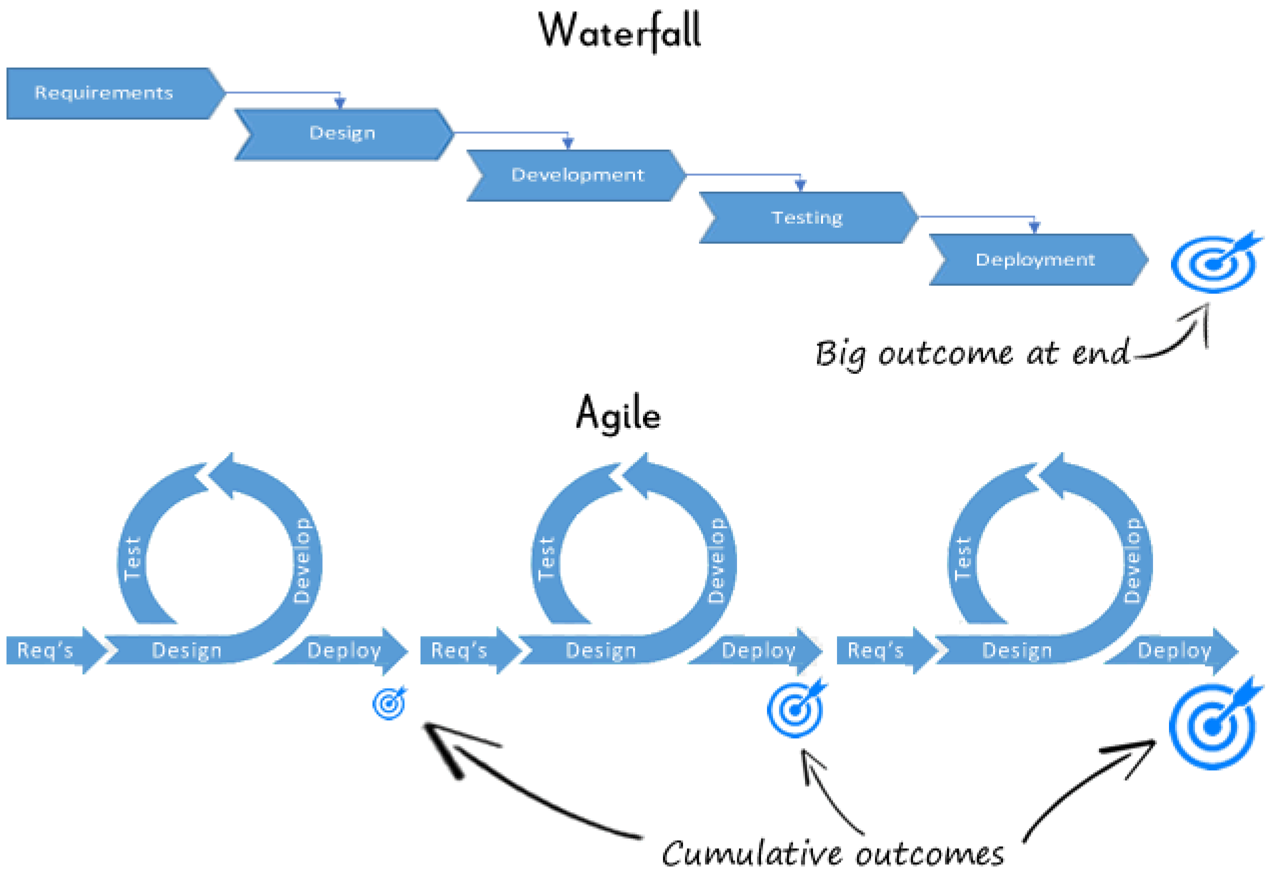
- changing requirements
- ambiguous requirements
- implicit expectations

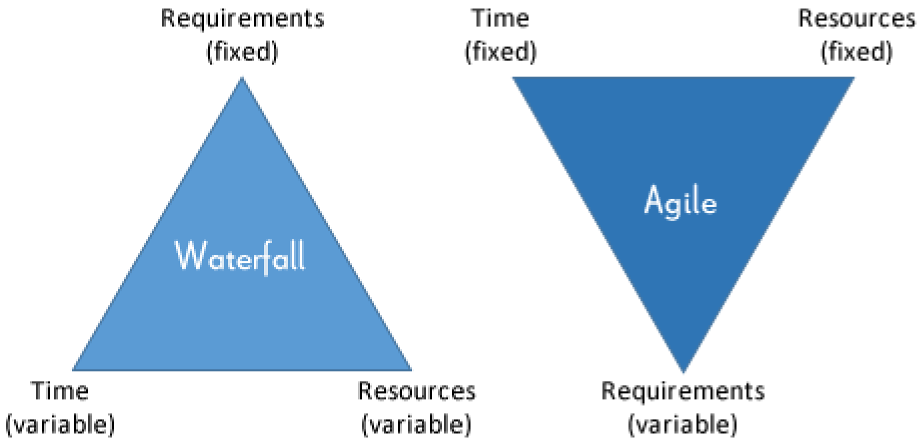

SO, YOUR USERS CAN ONLY DELETE POSTS WHEN LOGGED IN?
CORRECT
YES
YUP
YOU CANNOT CREATE A NEW POST WHILE LOGGED IN, YOU MAY ONLY DELETE THEM
OK
WELL THIS IS OBVIOUSLY NOT WHAT WE MEANT

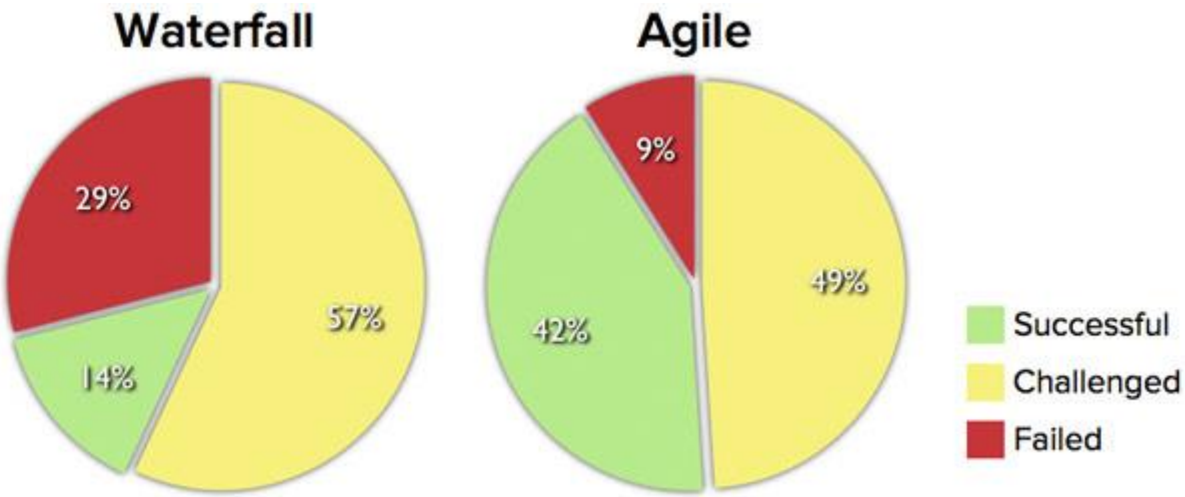
"The client kept changing the requirements on a daily basis, so we decided to freeze them until the next release."

Elements of solution

- Functional testing
- Favor Agile approaches

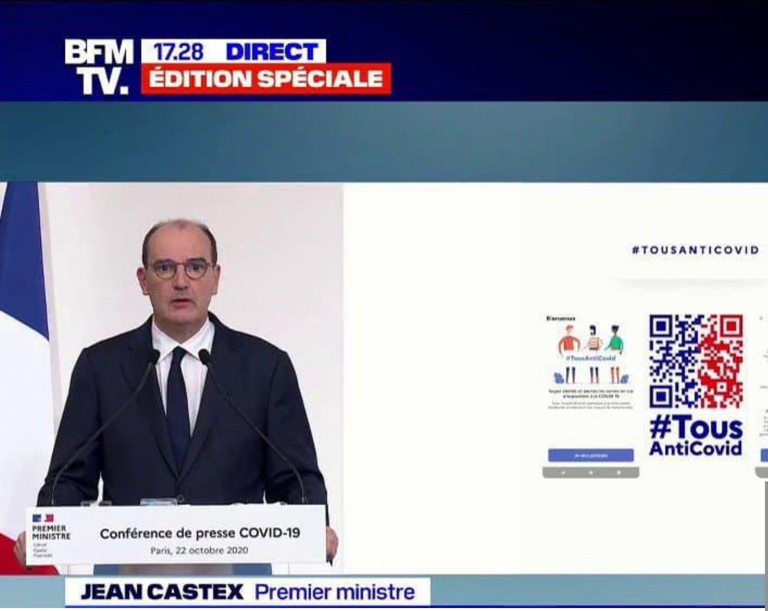


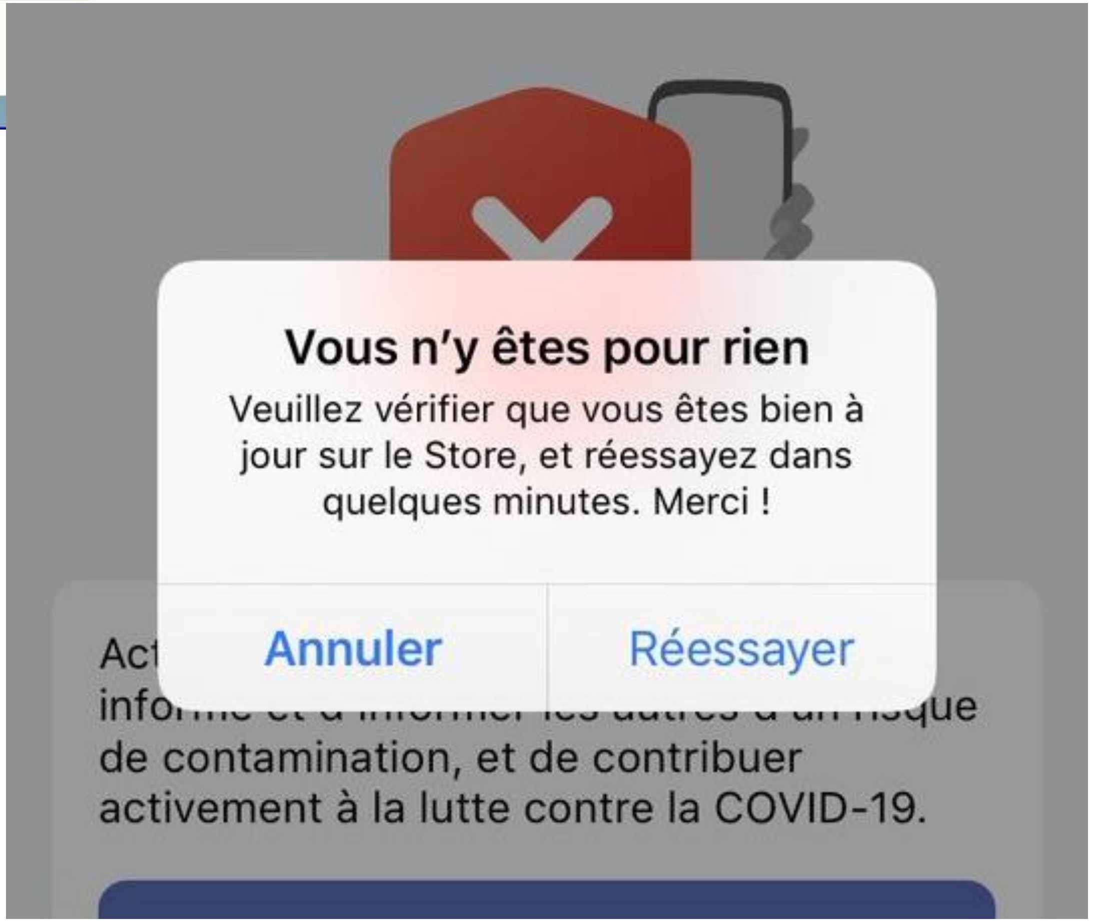




Source: The CHAOS Manifesto, The Standish Group, 2012.

http://crmsearch.eu/agile-versus-waterfall-crm.php

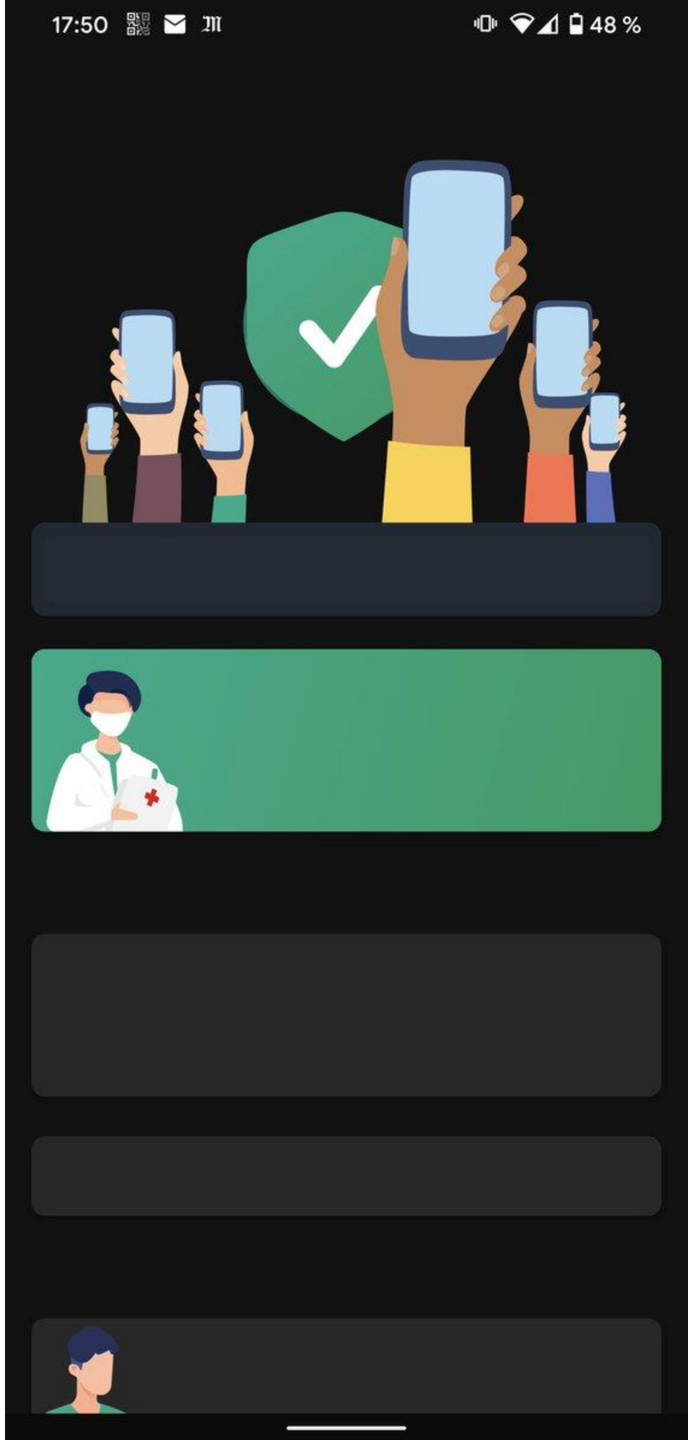




Vous n’y êtes pour rien

Veuillez vérifier que vous êtes bien à jour sur le Store, et réessayez dans quelques minutes. Merci !

Act
info
de contamination, et de contribuer activement à la lutte contre la COVID-19.



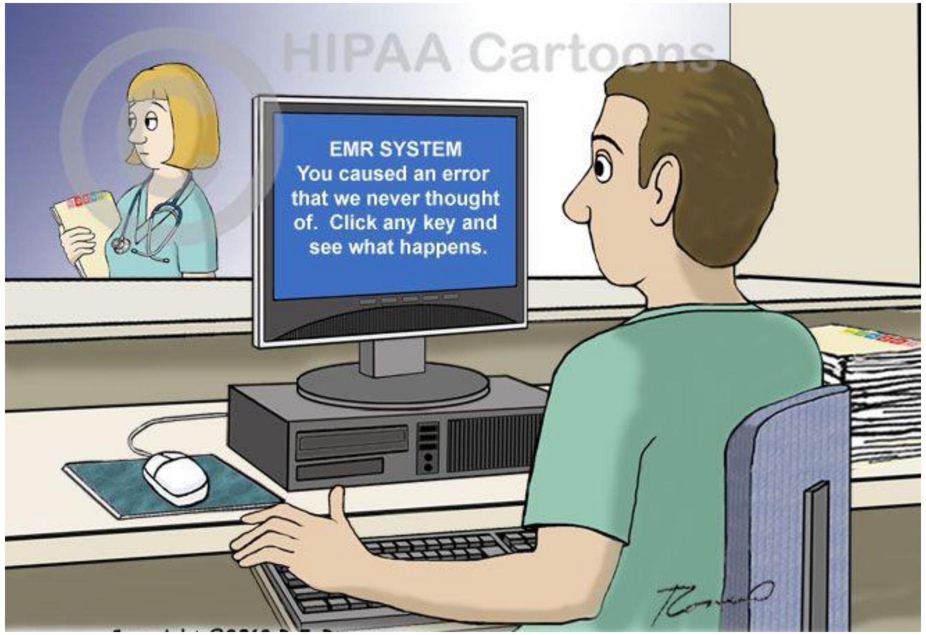
Reliability

functions under specified conditions for a specified amount of time: *How long can the system work for?*

- Maturity: is reliable under normal operating conditions: *Does it usually work?*
- Availability: is operational and accessible when one wants to use it: *Is it available when needed?*
  - expressed as a percentage of uptime in a given year: 99.9% reliability (“3 nines” class) = 10 min downtime/week
  - RTO *Recovery Time Objective*: maximal duration of service disruption admissible in order to avoid unacceptable consequences after a disaster
- Fault tolerance: works as intended despite the presence of hardware or software failures: *Does it work even when there are issues?*
- Recoverability: in the event of a failure or an error, can restore the system to the desired state and bring back the data directly involved: *Can the system recover from failure?*
  - RPO *Recovery Point Objective:* maximum tolerable amount of data to lose (how much time can occur between the last data backup and a disaster without causing serious damage?)

Sources of unreliability

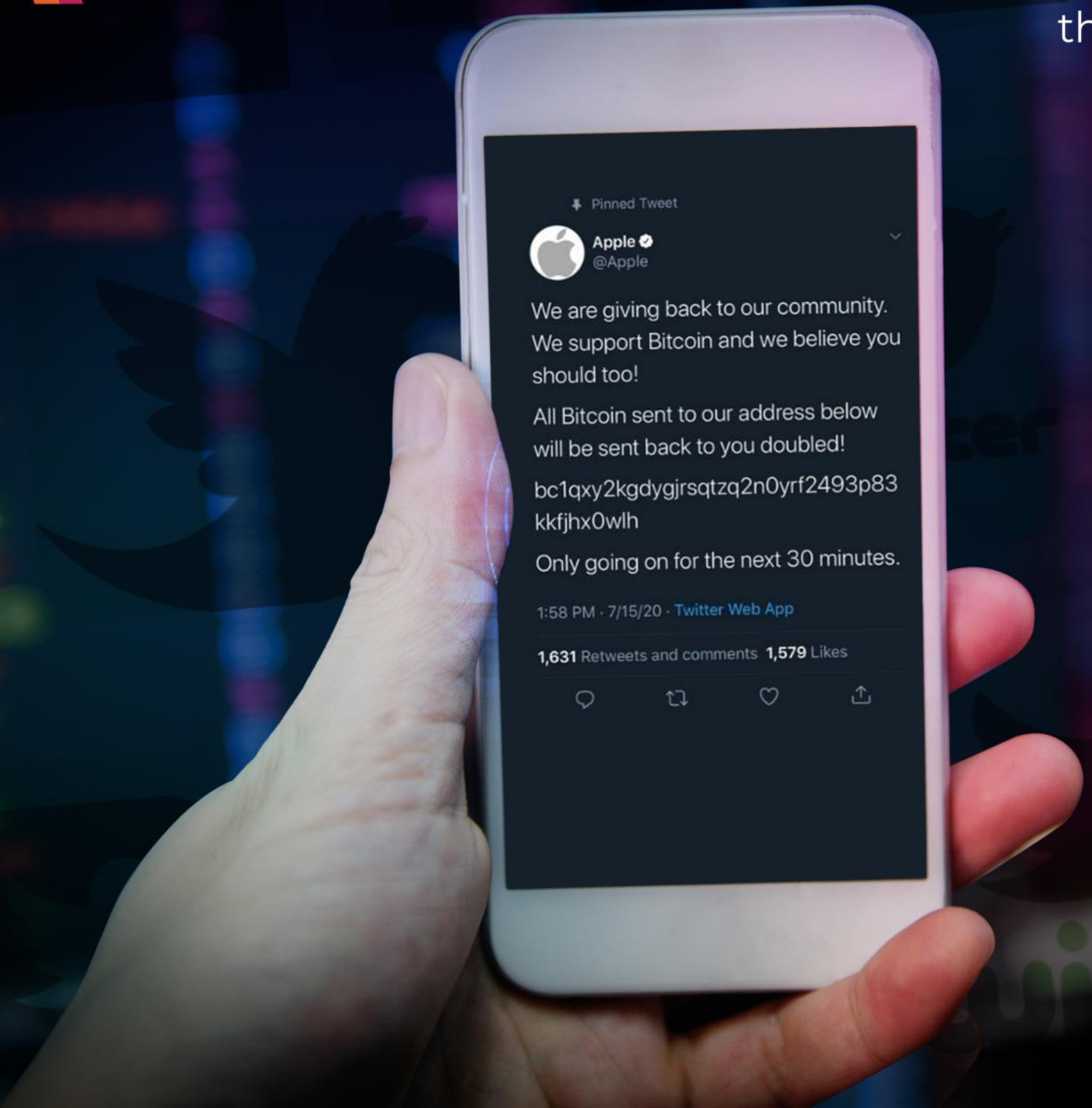
- deliberate attacks on the system, or accidents
- poor dimensioning of resources
- incident management procedure non-existent / inadequate / unknown


HIPAA Cartoons
EMR SYSTEM
You caused an error that we never thought of. Click any key and see what happens.

Elements of solution

- detect failures
- [differential] redundancy
- Hot-plugging operations (add, remove, or replace system components while the system is running) and dynamic reconfiguration (ability to adjust configuration of hot-plugged components)
- transactions, sessions
- take into account the fault tolerance capabilities of the platform, the host and the host provider
- simulate breakdowns during tests
- perform load tests
- design an elastic architecture
- design for failure, anti-fragile approach (reaction rather than prediction)

KRATIKAL
SECURE FOR SURE



Pinned Tweet

Apple
@Apple

We are giving back to our community. We support Bitcoin and we believe you should too!

All Bitcoin sent to our address below will be sent back to you doubled!

bc1qxy2kgdygjrsqtzq2n0yrf2493p83kkfjhx0wlh

Only going on for the next 30 minutes.

1:58 PM · 7/15/20 · Twitter Web App

1,631 Retweets and comments 1,579 Likes

The Twitter account hack was live on 15th July 2020 that targeted personalities like Jeff Bezos, Elon Musk, Bill Gates, Barack Obama among others.

TWITTER GOT HACKED!

It could have been you.

No technology is enough to keep you secure...
Being proactive can save you!

Stay Cyber Secure With ThreatCop
Register for a Free Trial!

Security

software and data are protected. *Can only the right people or systems see the right information or do the right actions?*

- Confidentiality: data is accessible only to those authorized
- Integrity: prevents unauthorized access to or modification of computer programs or data
- Non-repudiation and Accountability: it can be proved that actions or events have taken place, specific entity actions can be traced
- Authenticity: the identity of a subject or source can be proved as claimed

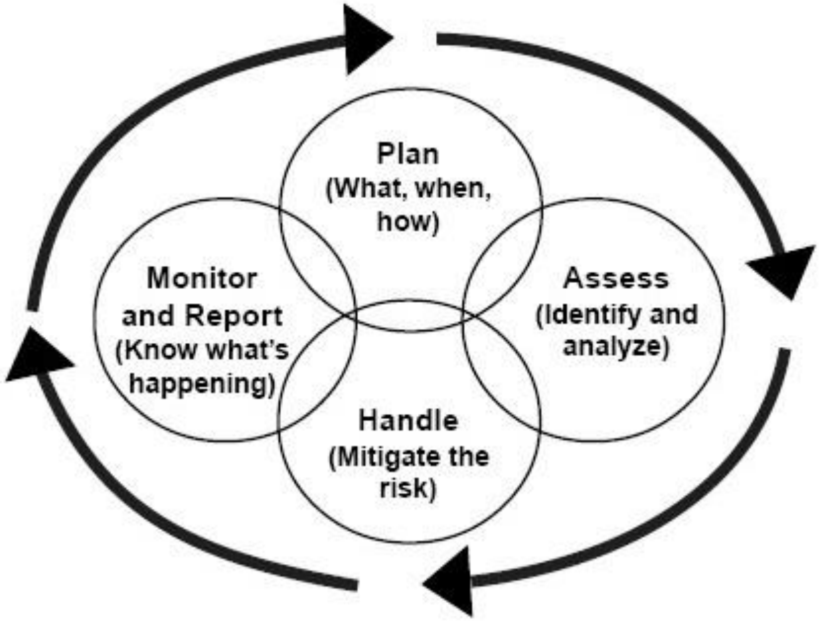
Elements of solution

- physically secure equipment
- inform and train users
- put on cracker’s shoes (do port scanning etc.)
- detect intrusions (attract crackers to decoys (honeypot) to identify them, logs analysis...)
- apply security updates
- use encryption and authentication
- use antivirus, antispam, antirootkit...
- filter network traffic (firewall)
- partition the network
- least privileged principle
- check Control-Flow Integrity
- defensive programming, check (user) inputs
  - type, value, match to regular expression
  - escape special character, remove training space or new line characters
  - use parametrized statements

Digital risk managment

- *"process of identifying vulnerabilities and threats to the information resources used by an organization in achieving business objectives, and deciding what countermeasures, if any, to take in reducing risk to an acceptable level, based on the value of the information resource to the organization“*

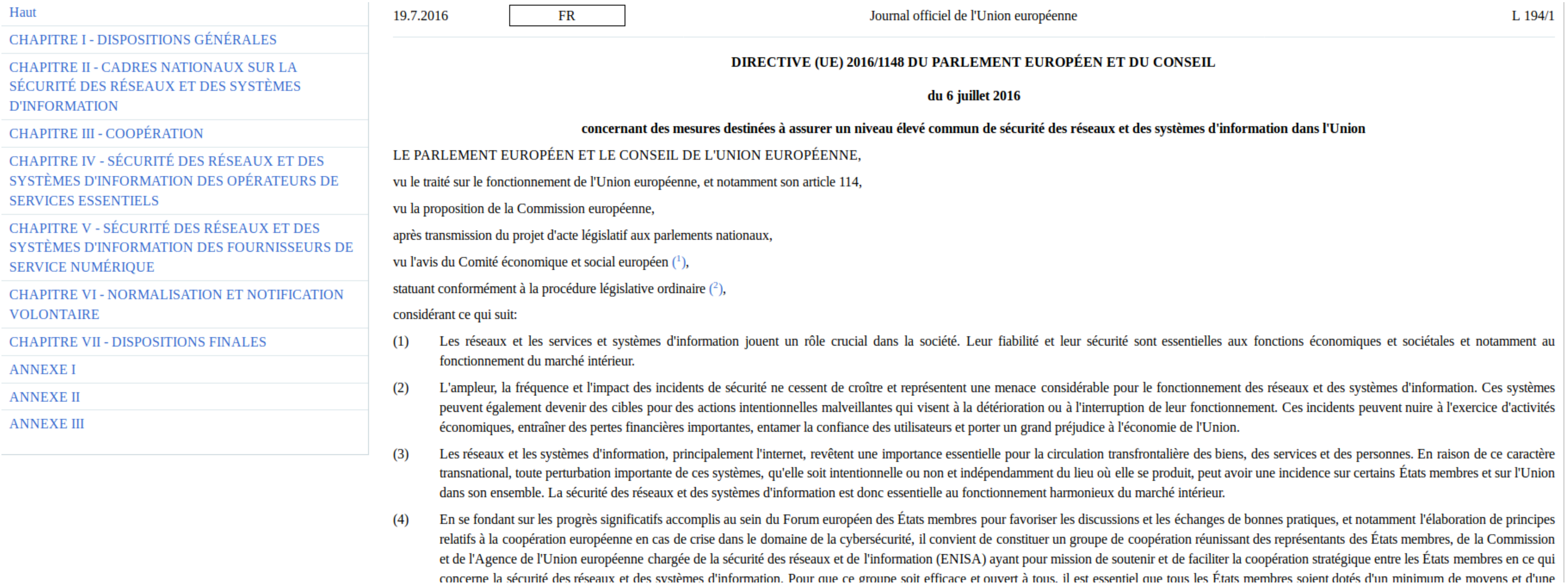
- Continuous and iterative process


Plan (What, when, how)
Assess (Identify and analyze)
Handle (Mitigate the risk)
Monitor and Report (Know what's happening)

- Several methods: EBIOS (Fr.), CRAMM…

EU directive on Network and Information System Security (NIS)

sets a range of network and information security requirements which apply to operators of essential services (enterprises in the energy, transport, banking, financial market infrastructures, health, drinking water supply and distribution) and digital service providers (DSPs)



19.7.2016 FR Journal officiel de l'Union européenne L 194/1

DIRECTIVE (UE) 2016/1148 DU PARLEMENT EUROPÉEN ET DU CONSEIL

du 6 juillet 2016

concernant des mesures destinées à assurer un niveau élevé commun de sécurité des réseaux et des systèmes d'information dans l'Union

LE PARLEMENT EUROPÉEN ET LE CONSEIL DE L'UNION EUROPÉENNE,

vu le traité sur le fonctionnement de l'Union européenne, et notamment son article 114,

vu la proposition de la Commission européenne,

après transmission du projet d'acte législatif aux parlements nationaux,

vu l'avis du Comité économique et social européen ([1]),

statuant conformément à la procédure législative ordinaire ([2]),

considérant ce qui suit:

(1) Les réseaux et les services et systèmes d'information jouent un rôle crucial dans la société. Leur fiabilité et leur sécurité sont essentielles aux fonctions économiques et sociétales et notamment au fonctionnement du marché intérieur.

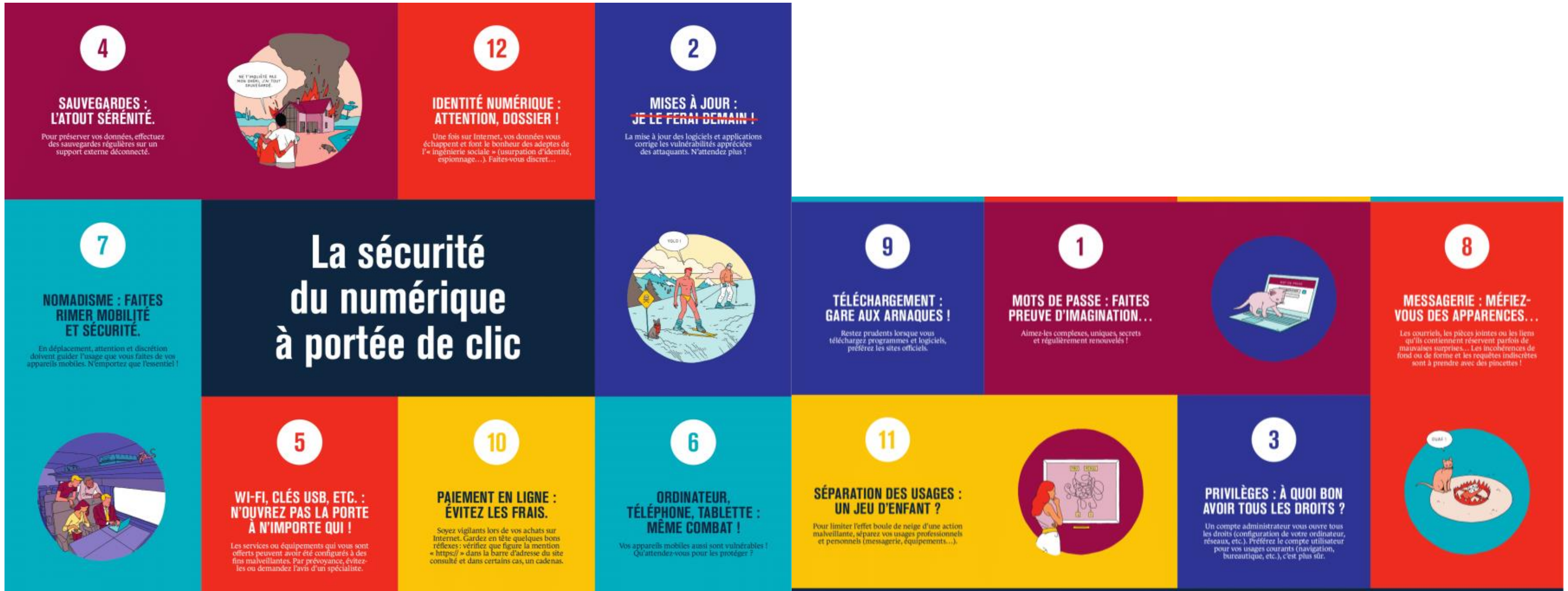
(2) L'ampleur, la fréquence et l'impact des incidents de sécurité ne cessent de croître et représentent une menace considérable pour le fonctionnement des réseaux et des systèmes d'information. Ces systèmes peuvent également devenir des cibles pour des actions intentionnelles malveillantes qui visent à la détérioration ou à l'interruption de leur fonctionnement. Ces incidents peuvent nuire à l'exercice d'activités économiques, entraîner des pertes financières importantes, entamer la confiance des utilisateurs et porter un grand préjudice à l'économie de l'Union.

(3) Les réseaux et les systèmes d'information, principalement l'internet, revêtent une importance essentielle pour la circulation transfrontalière des biens, des services et des personnes. En raison de ce caractère transnational, toute perturbation importante de ces systèmes, qu'elle soit intentionnelle ou non et indépendamment du lieu où elle se produit, peut avoir une incidence sur certains États membres et sur l'Union dans son ensemble. La sécurité des réseaux et des systèmes d'information est donc essentielle au fonctionnement harmonieux du marché intérieur.

(4) En se fondant sur les progrès significatifs accomplis au sein du Forum européen des États membres pour favoriser les discussions et les échanges de bonnes pratiques, et notamment l'élaboration de principes relatifs à la coopération européenne en cas de crise dans le domaine de la cybersécurité, il convient de constituer un groupe de coopération réunissant des représentants des États membres, de la Commission et de l'Agence de l'Union européenne chargée de la sécurité des réseaux et de l'information (ENISA) ayant pour mission de soutenir et de faciliter la coopération stratégique entre les États membres en ce qui concerne la sécurité des réseaux et des systèmes d'information. Pour que ce groupe soit efficace et ouvert à tous, il est essentiel que tous les États membres soient dotés d'un minimum de moyens et d'une

ANSSI : Agence Nationale de la Sécurité des Systèmes d'Information

- former les grands décideurs à la souveraineté numérique et à la cybersécurité
- dresser un catalogue des meilleures solutions de protection
- mettre en place une culture des gestes de base pour tous
- portail dédié créé en octobre 2017 : cybermalveillance.gouv.fr



La sécurité du numérique à portée de clic

4 **SAUVEGARDES : L'ATOUT SÉRÉNITÉ.** Pour préserver vos données, effectuez des sauvegardes régulières sur un support externe déconnecté.

12 **IDENTITÉ NUMÉRIQUE : ATTENTION, DOSSIER !** Une fois sur Internet, vos données vous échappent et font le bonheur des adeptes de l'« ingénierie sociale » (usurpation d'identité, espionnage…). Faites-vous discret…

2 **MISES À JOUR : ~~JE LE FERAI DEMAIN !~~** La mise à jour des logiciels et applications corrige les vulnérabilités appréciées des attaquants. N'attendez plus !

7 **NOMADISME : FAITES RIMER MOBILITÉ ET SÉCURITÉ.** En déplacement, attention et discrétion doivent guider l'usage que vous faites de vos appareils mobiles. N'emportez que l'essentiel !

9 **TÉLÉCHARGEMENT : GARE AUX ARNAQUES !** Restez prudents lorsque vous téléchargez programmes et logiciels, préférez les sites officiels.

1 **MOTS DE PASSE : FAITES PREUVE D'IMAGINATION…** Aimez-les complexes, uniques, secrets et régulièrement renouvelés !

8 **MESSAGERIE : MÉFIEZ-VOUS DES APPARENCES…** Les courriels, les pièces jointes ou les liens qu'ils contiennent réservent parfois de mauvaises surprises… Les incohérences de fond ou de forme et les requêtes indiscrètes sont à prendre avec des pincettes !

5 **WI-FI, CLÉS USB, ETC. : N'OUVREZ PAS LA PORTE À N'IMPORTE QUI !** Les services ou équipements qui vous sont offerts peuvent avoir été configurés à des fins malveillantes. Par prévoyance, évitez-les ou demandez l'avis d'un spécialiste.

10 **PAIEMENT EN LIGNE : ÉVITEZ LES FRAIS.** Soyez vigilants lors de vos achats sur Internet. Gardez en tête quelques bons réflexes : vérifiez que figure la mention « https:// » dans la barre d'adresse du site consulté et dans certains cas, un cadenas.

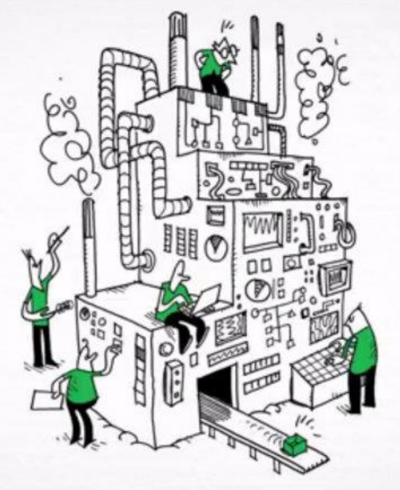
6 **ORDINATEUR, TÉLÉPHONE, TABLETTE : MÊME COMBAT !** Vos appareils mobiles aussi sont vulnérables ! Qu'attendez-vous pour les protéger ?

11 **SÉPARATION DES USAGES : UN JEU D'ENFANT ?** Pour limiter l'effet boule de neige d'une action malveillante, séparez vos usages professionnels et personnels (messagerie, équipements…).

3 **PRIVILÈGES : À QUOI BON AVOIR TOUS LES DROITS ?** Un compte administrateur vous ouvre tous les droits (configuration de votre ordinateur, réseaux, etc.). Préférez le compte utilisateur pour vos usages courants (navigation, bureautique, etc.), c'est plus sûr.

VU COMMENT ÇA RAME... JE COMPRENDS QU'ILS APPELLENT ÇA UN NAVIGATEUR...
Jitho

Performance efficiency



amount of resources used coherent with response time

- Speed (Time-behavior): processing and response times, and throughput
- Resource utilization: *How many resources (bandwidth, memory, CPU time) are used during the execution?*
- Capacity: What is the maximum concurrent users, transactions, or actions at any one time? What is the fastest, biggest transaction?

Sources of performance issues

- Wrong technological choices
- Wrong architecture
- Wrong algorithm
- Wrong implementation
- Lack of load testing
- …

Elements of solution

- Benchmarking
- Load testing
- Code complexity analysis
- Code profiling to identify CPU or memory “hot spots”

Usability

- Recognition of fitness: users can recognize whether the software is suitable for their needs
- Learnability: *How quickly and easily can a user learn how to use the software?*
- Operability: *Is it easy to use?*
- User error protection: protects users from making errors
- User interface aesthetics: user interface allows the user to have an enjoyable and rewarding experience
- Accessibility: can be used by people with a wide range of features and capabilities

Elements of solution

- respect for business habits → reuse GUI bricks?
- change management → development of tutorials, contextual help?
- ergonomics → development of GUIs, customization functions, alternative interfaces (smartphones, tablets...)
- performance / speed of startup and reaction → taking into account the hardware and software environment, load forecasting, benchmarks etc...
- easy to install and administer → automatic installation modules, graphical administration console...

"Here's your problem. The software was manufactured in November and your computer was manufactured in February. Sagittarius is incompatible with Aquarius."

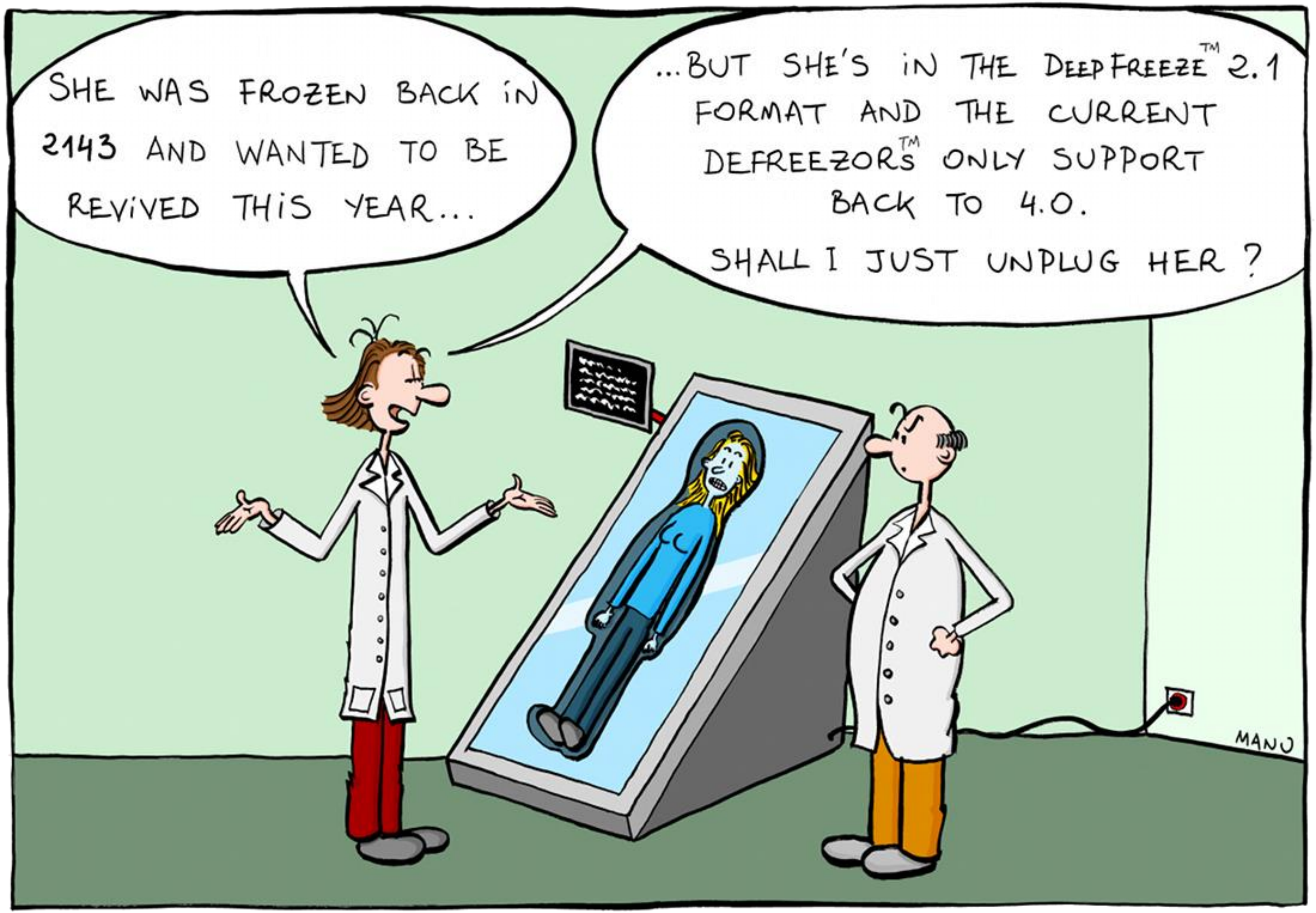
Compatibility

The software can exchange information with others, and/or it can perform while sharing the same environment. *How well does this solution interact with other solutions or applications that it must work with?*

- Co-existence: the software can perform without affecting any other product while it shares a common environment and resources with them
- Interoperability: two or more systems are able to exchange information and to use the information exchanged

Elements of solution

- Favor standard
- Integration testing


SHE WAS FROZEN BACK IN 2143 AND WANTED TO BE REVIVED THIS YEAR...
...BUT SHE'S IN THE DEEPFREEZE™ 2.1 FORMAT AND THE CURRENT DEFREEZORS™ ONLY SUPPORT BACK TO 4.0. SHALL I JUST UNPLUG HER?
MANU

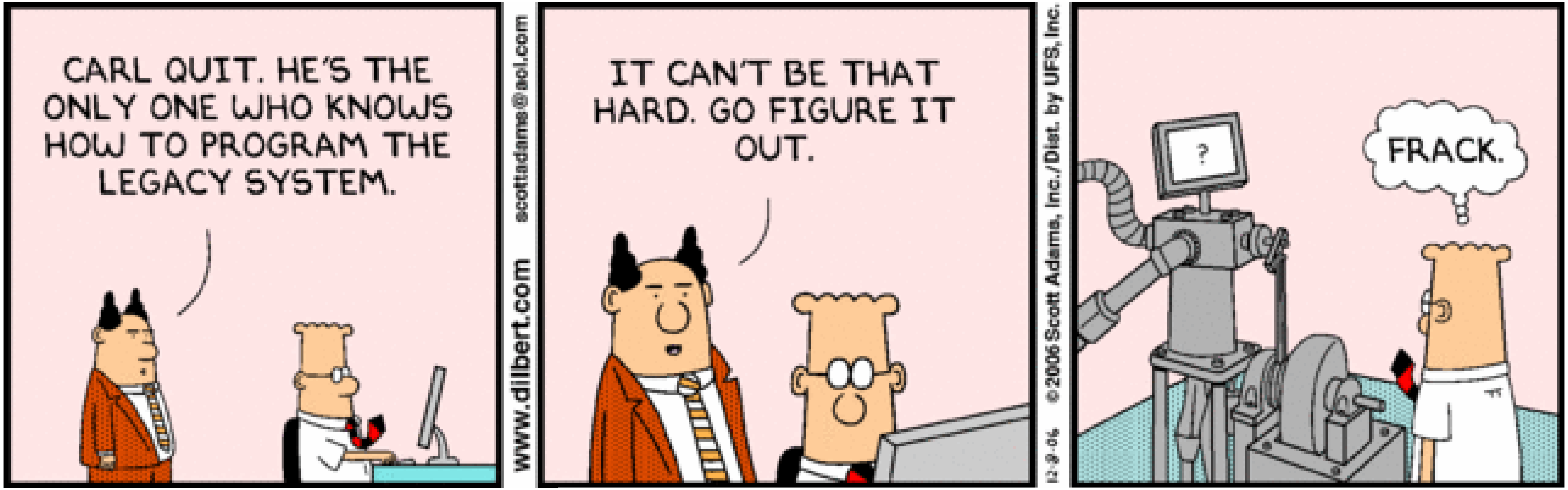
Portability

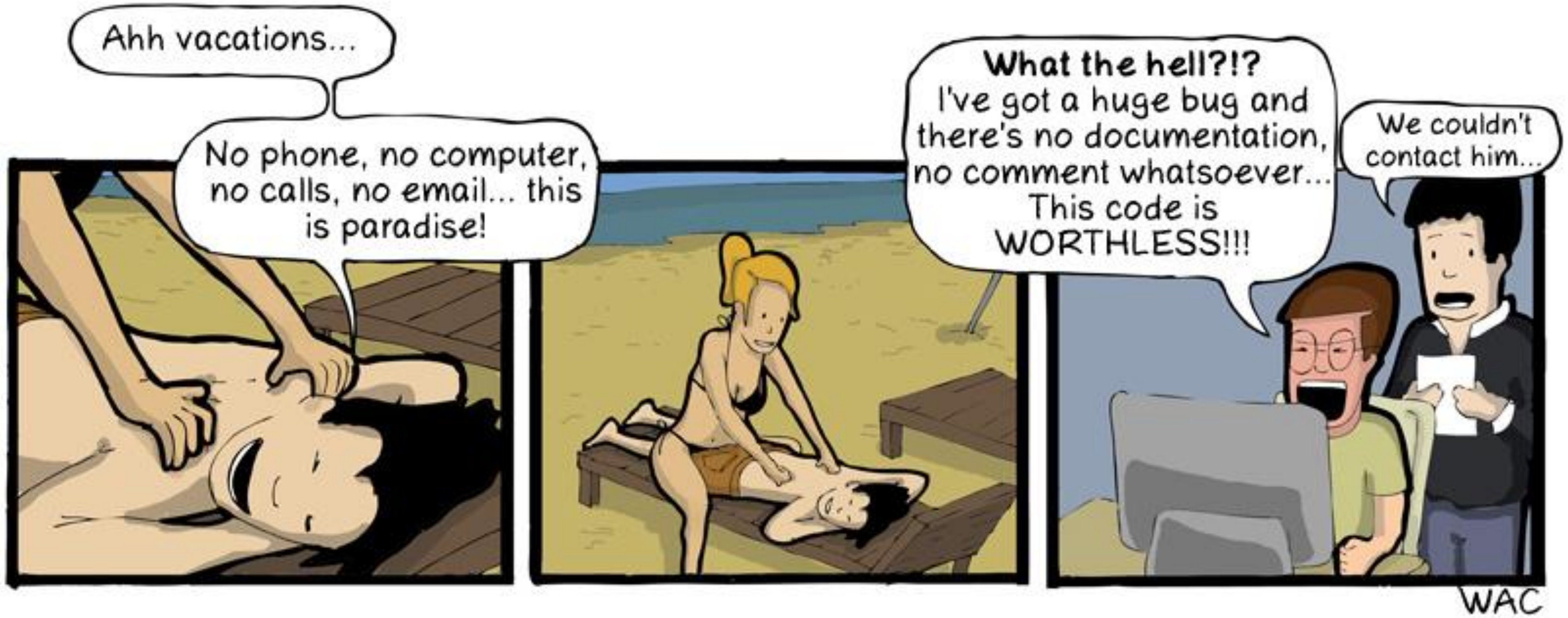
can be effectively and efficiently transferred to a different environment

- Adaptability: can be adapted to different or evolving hardware, software or other operational or operating environments
- Installability: can be installed or removed
- Replaceability: can replace another specific software product, with the same purpose in the same environment

Elements of solution

- use high-level language and stick to the standard (if it exists!)
- battery of virtual machines for testing
- i18n: internationalization of user interfaces
- provide modular delivery for encryption algorithms that are not allowed for export or different copyright protected codecs in different countries
- develop automatic installation modules, graphical administration console ...
- design for backward compatibility, manage versions


CARL QUIT. HE'S THE ONLY ONE WHO KNOWS HOW TO PROGRAM THE LEGACY SYSTEM.
scottadams@aol.com
www.dilbert.com
IT CAN'T BE THAT HARD. GO FIGURE IT OUT.
© 2006 Scott Adams, Inc./Dist. by UFS, Inc.
12-9-06
?
FRACK.


Ahh vacations...
No phone, no computer, no calls, no email... this is paradise!
What the hell?!? I've got a huge bug and there's no documentation, no comment whatsoever... This code is WORTHLESS!!!
We couldn't contact him...
WAC

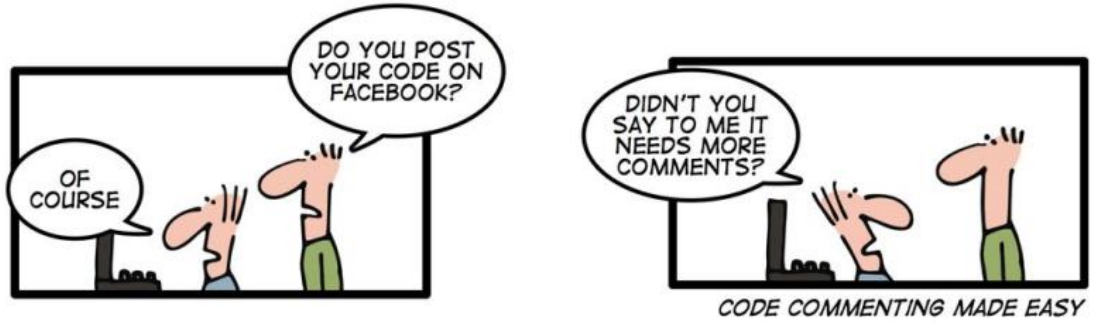
Maintainability

How easily can changes and fixes be made?

- Modularity: constructed in separate components so that changes to one component has minimal impact on other components
- Reusability: an existing part can be used in more than one part of the system or in building a new one
- Analysability: it is possible to assess the impact of a planned change of one or more components
- Modifiability: mistakes or deterioration can be changed
- Testability: have identifiable criteria, and tests can determine whether those criteria are met

Elements of solution

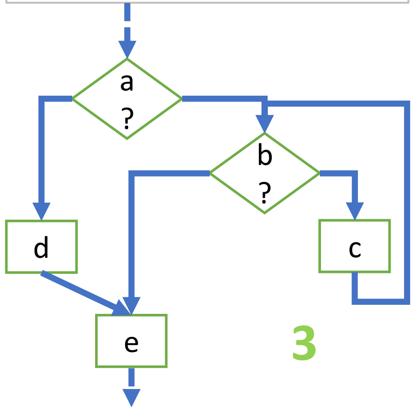
- modular design
  - small functions, micro-service architecture…
  - OO Measures to lower:
    - Coupling Between Object (CBO): number of classes used by a class
    - Lack of Cohesion of Methods (LCOM): number of pair of methods not using the same class attributes
- aim for simplicity (low cyclomatic complexity…)
- test sets
- use of patterns
- compliance with conventions
- documentation
- choice of a language or a perennial environment


DO YOU POST YOUR CODE ON FACEBOOK?
OF COURSE
DIDN'T YOU SAY TO ME IT NEEDS MORE COMMENTS?
CODE COMMENTING MADE EASY

Cyclomatic complexity

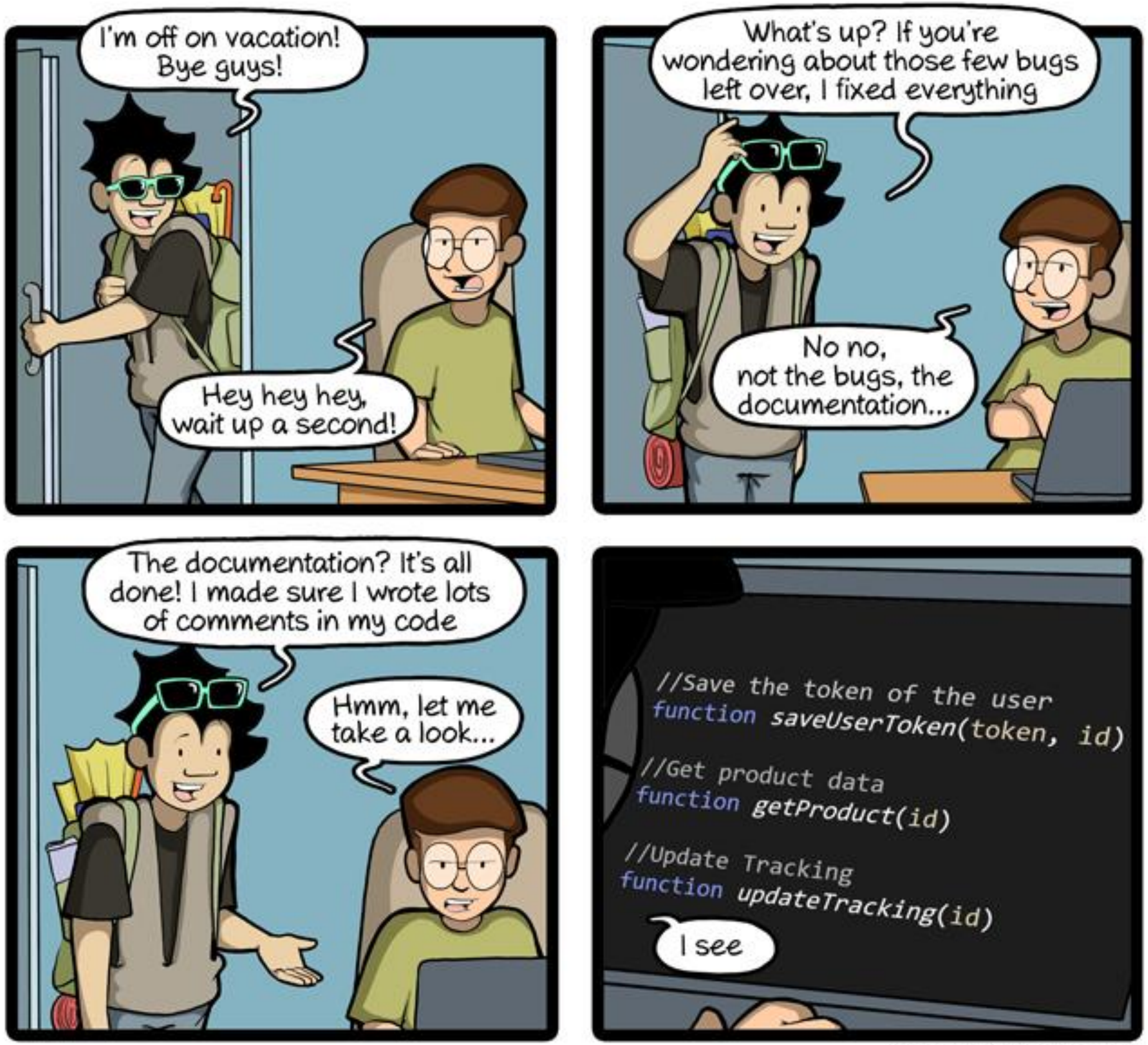
- measure of the number of linearly independent paths through a program's source code (≈ number of decision points + 1)
- developed by McCabe in 1976
- example of “rule”: break down the code (in subroutines) when cyclomatic complexity > 12
- not always meaningful (ex: switch cases construction)

```
if (a) {
    while(b) {
        c;
    }
} else {
    d;
}
e;
```



Documentation

metrics are to be used with caution


I'm off on vacation! Bye guys!
Hey hey hey, wait up a second!
What's up? If you're wondering about those few bugs left over, I fixed everything
No no, not the bugs, the documentation...
The documentation? It's all done! I made sure I wrote lots of comments in my code
Hmm, let me take a look...
I see

```
//Save the token of the user
function saveUserToken(token, id)

//Get product data
function getProduct(id)

//Update Tracking
function updateTracking(id)
```

CommitStrip.com

Some tools…

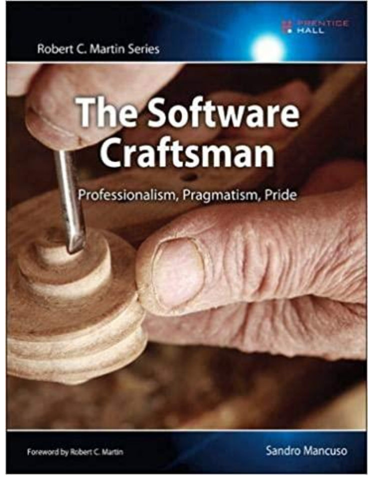
Tools that can help to improve software quality

- Code editor (syntax coloring, navigation…)
- Debugger
- Code analysis:
  - bad code smells
  - duplicate code
  - complexity
  - performance profiling
  - tests coverage
  - amount of comments
  - code churn (amount of changes)
- Code refactoring

- Unit testing
- Load and performance testing
- Documentation generation
- Bug tracking
- Source versioning
- Automatic build
- Project dashboard manager
- …

Software craftmanship

Tools are not enough



PRENTICE HALL

Robert C. Martin Series

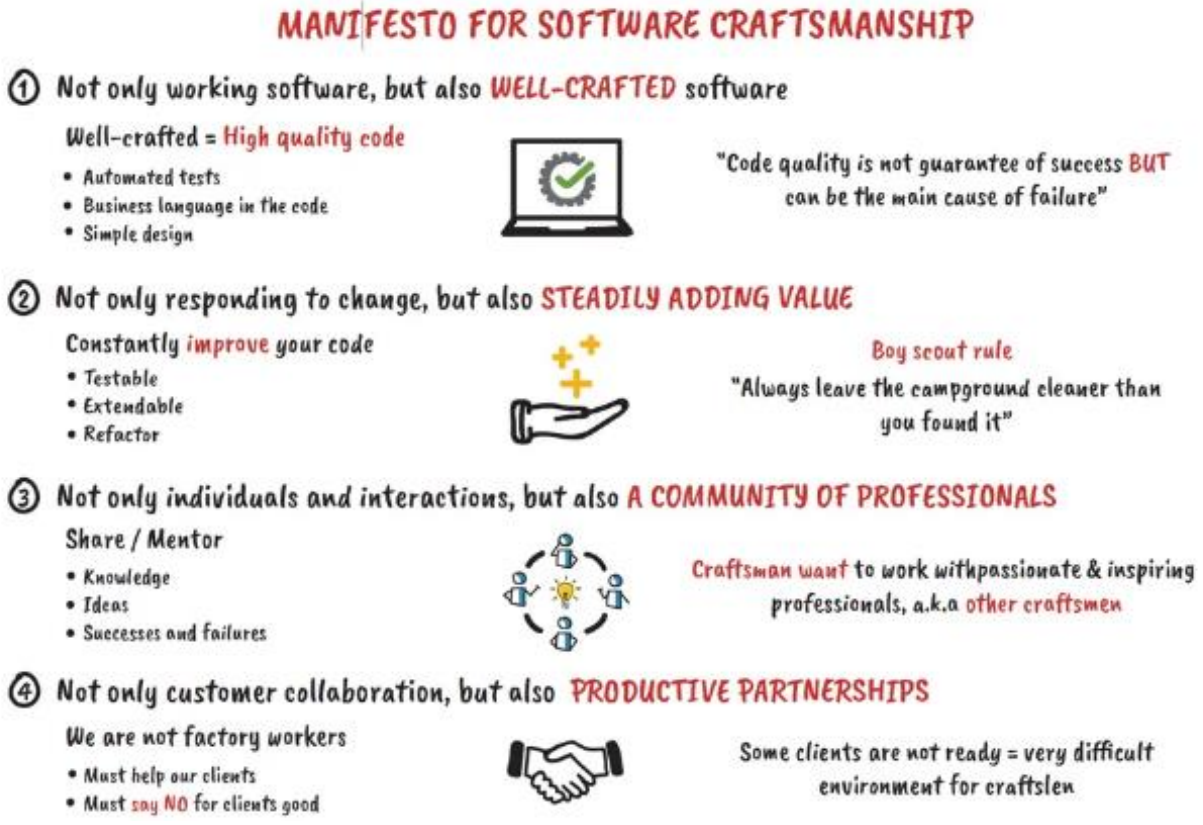
The Software Craftsman

Professionalism, Pragmatism, Pride

Foreword by Robert C. Martin

Sandro Mancuso

- Each software project requires some innovation
- Software development (still) closer to craftmanship than to manufacturing of goods
- Same tools, but non-repeatable process
- Communities of practices are important



MANIFESTO FOR SOFTWARE CRAFTSMANSHIP

① Not only working software, but also WELL-CRAFTED software

Well-crafted = High quality code

- Automated tests
- Business language in the code
- Simple design

"Code quality is not guarantee of success BUT can be the main cause of failure"

② Not only responding to change, but also STEADILY ADDING VALUE

Constantly improve your code

- Testable
- Extendable
- Refactor

Boy scout rule

"Always leave the campground cleaner than you found it"

③ Not only individuals and interactions, but also A COMMUNITY OF PROFESSIONALS

Share / Mentor

- Knowledge
- Ideas
- Successes and failures

Craftsman want to work withpassionate & inspiring professionals, a.k.a other craftsmen

④ Not only customer collaboration, but also PRODUCTIVE PARTNERSHIPS

We are not factory workers

- Must help our clients
- Must say NO for clients good

Some clients are not ready = very difficult environment for craftslen