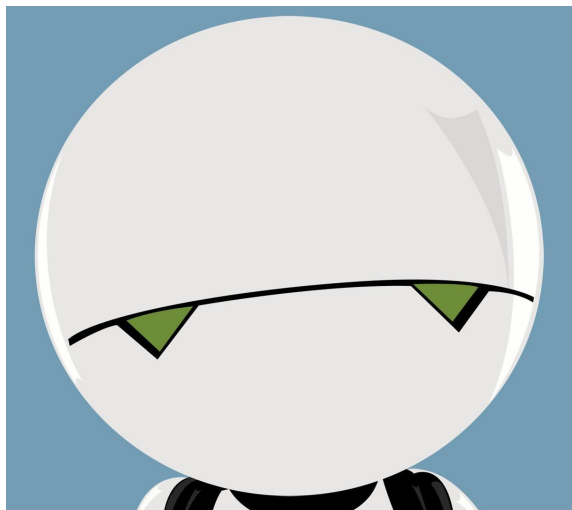


Le meilleur développeur est-il un UX engineer ?

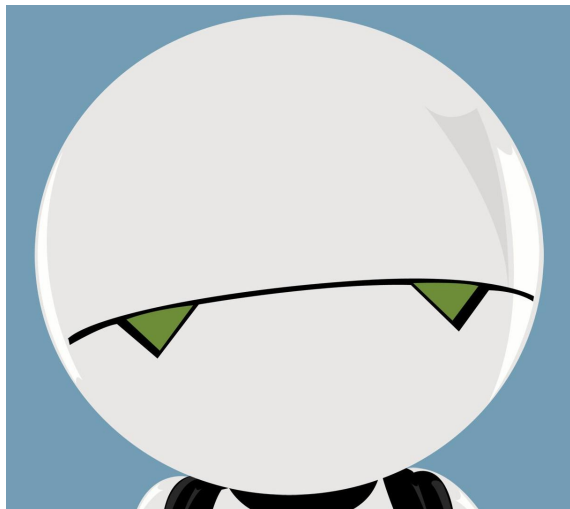
tldr: oui



Moi

Ce que les gens oublient

- ▼ Ingénieur en Cognitique de formation
- ▼ UX Engineer pour faire simple
- ▼ Data Engineer & SRE dans la vraie vie
- ▼ Craftman convaincu selon mon manager



Craft. Don't talk to me
about craft.

Un peu d'histoire

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.**

C'est à dire qu'en recherchant les éléments de gauche, nous avons trouvé que les éléments de droite sont indispensables.

Robert C. Martin Series



Clean Code

A Handbook of Agile Software Craftsmanship

Foreword by James O. Coplien

Robert C. Martin

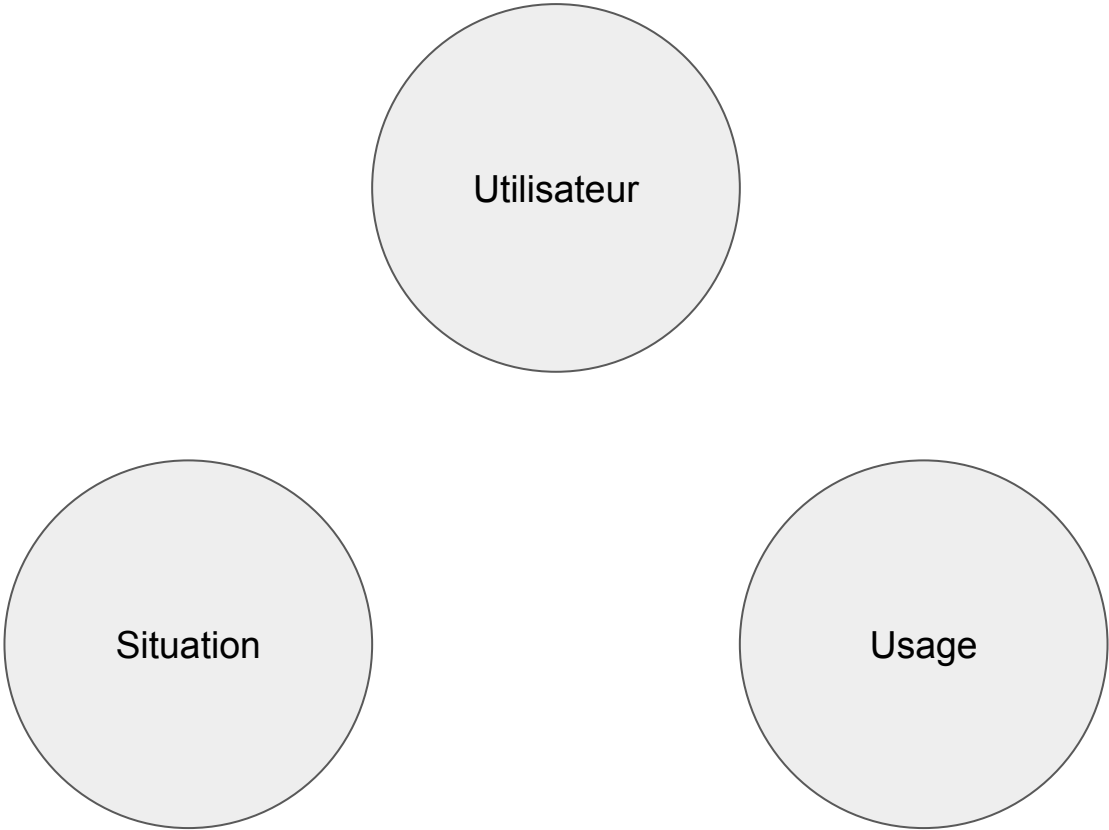


| Œufs ou poules

- ▼ 1988 : The Design of everyday things, Don Norman
- ▼ 1993 : Heuristique Bastien et Scapin
- ▼ 1999 : The Inmates are running the asylum, Alan Cooper
- ▼ 1999 : The Pragmatique Programmer, Andrew Hunt & David Thomas
- ▼ 2001 : Software Craftmanship, Pete McBreen
- ▼ 2008 : Software Craftmanship manifesto, Uncle Bob & Co
- ▼ 2008 : Clean code

L'UX level 0

comment avoir son diplôme sans bachoter

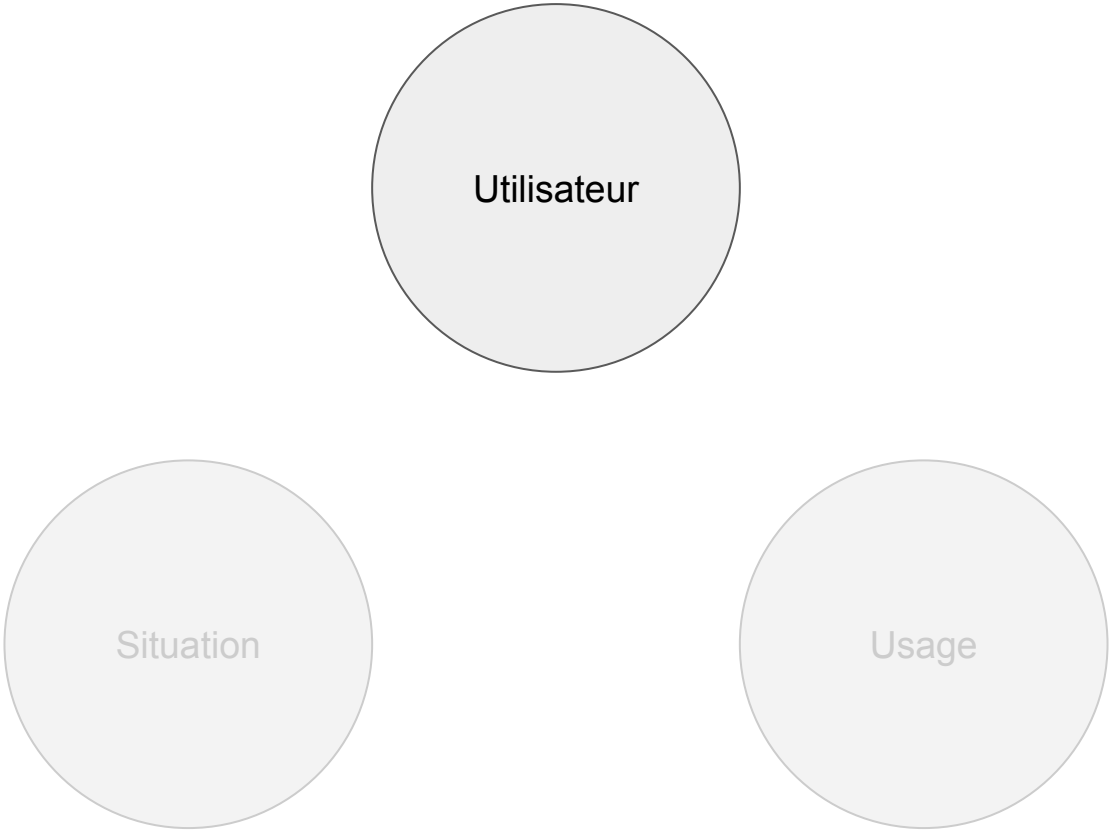


A diagram consisting of three light gray circles arranged in a triangle. The top circle is labeled 'Utilisateur', the bottom-left circle is labeled 'Situation', and the bottom-right circle is labeled 'Usage'. All circles have a thin black outline.

Utilisateur

Situation

Usage

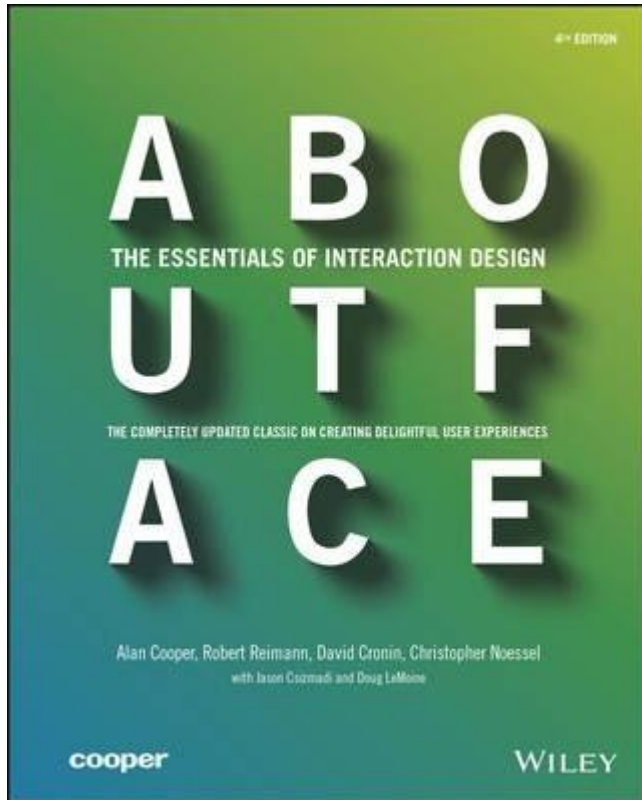


A diagram illustrating the Three-Component Model of User-Computer Interaction. It consists of three light gray circles arranged in a triangle. The top circle is labeled 'Utilisateur' in bold black text. The bottom-left circle is labeled 'Situation' in gray text. The bottom-right circle is labeled 'Usage' in gray text. The circles are connected by thin gray lines, forming a triangular structure.

Utilisateur

Situation

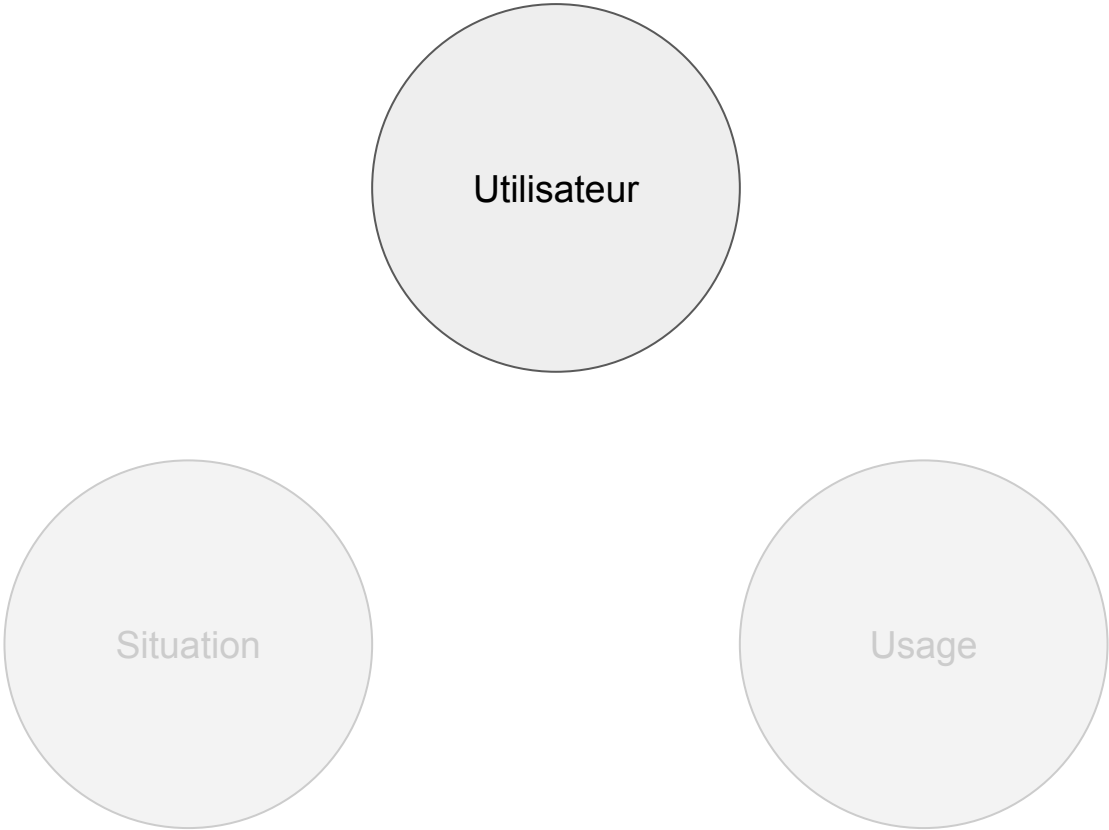
Usage



Persona

1999, Alan Cooper

- ▼ Archétype d'utilisateur possible
- ▼ Type d'utilisateur (age, sexe, profession, csp, ...)
- ▼ Besoin, Objectif
- ▼ Expertise du domaine
- ▼ Façon de se servir de l'application

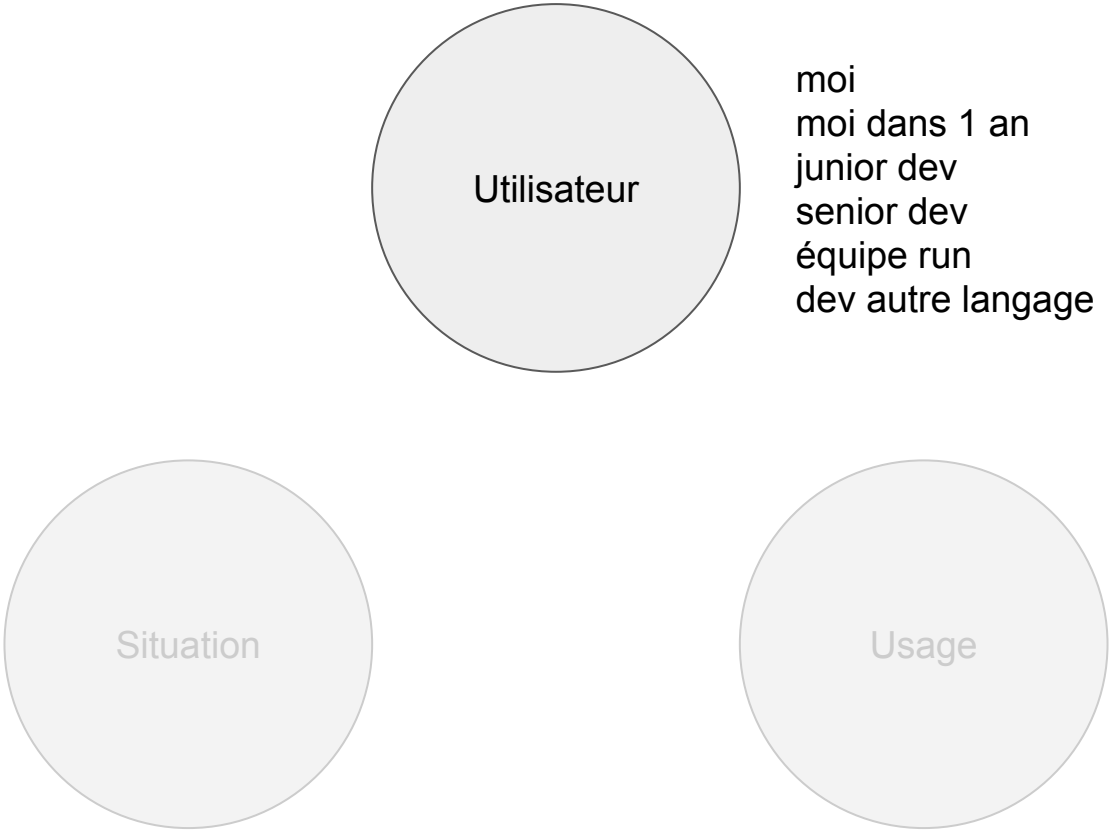


A diagram illustrating the Three-Component Model of User-Computer Interaction. It consists of three light gray circles arranged in a triangle. The top circle is labeled 'Utilisateur' in bold black text. The bottom-left circle is labeled 'Situation' in gray text. The bottom-right circle is labeled 'Usage' in gray text. The circles are connected by thin gray lines, forming a triangular structure.

Utilisateur

Situation

Usage

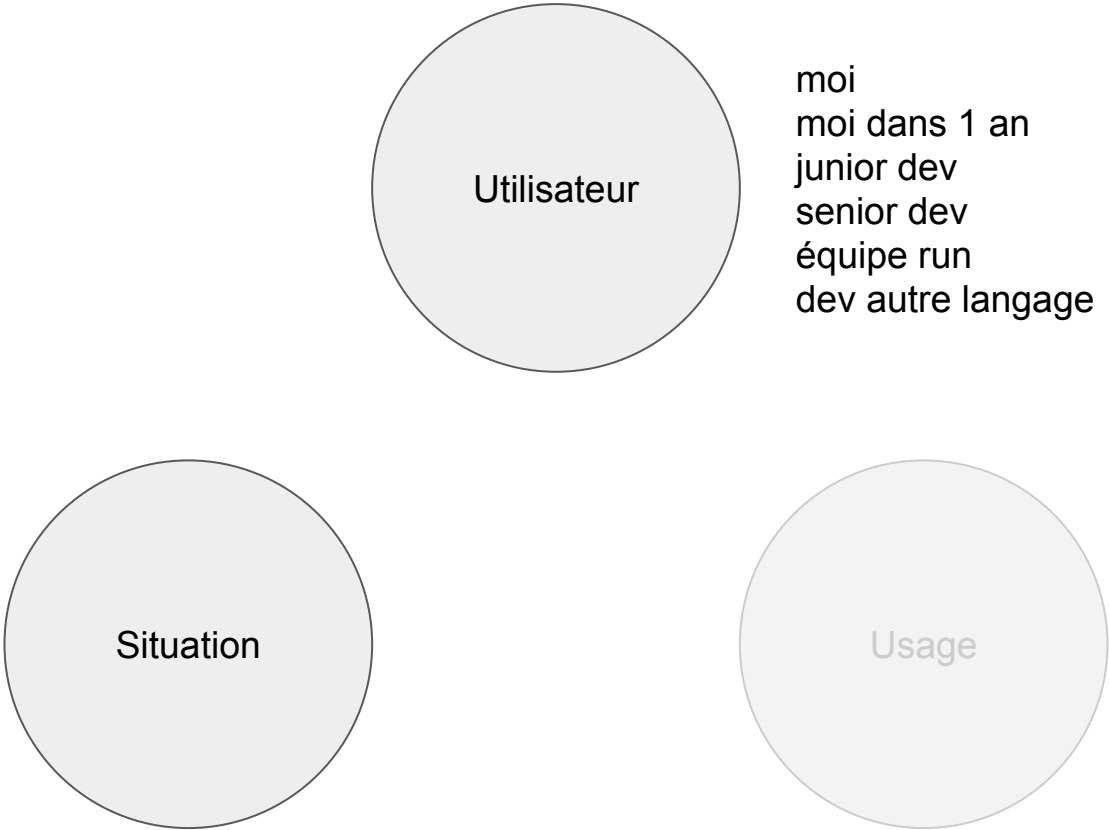


Utilisateur

moi
moi dans 1 an
junior dev
senior dev
équipe run
dev autre langage

Situation

Usage



Utilisateur

moi
moi dans 1 an
junior dev
senior dev
équipe run
dev autre langage

Situation

Usage

A diagram consisting of three light gray circles arranged in a triangle. The top circle is labeled 'Utilisateur'. The bottom-left circle is labeled 'Situation'. The bottom-right circle is labeled 'Usage'. To the right of the 'Utilisateur' circle is a list of text. To the left of the 'Situation' circle is a list of text.

Utilisateur

moi
moi dans 1 an
junior dev
senior dev
équipe run
dev autre langage

Situation

bureau
conférence
bug (stress)

Usage

Utilisateur

moi
moi dans 1 an
junior dev
senior dev
équipe run
dev autre langage

Situation

bureau
conférence
bug (stress)

Usage

A diagram consisting of three light gray circles arranged in a triangle. The top circle is labeled 'Utilisateur'. The bottom-left circle is labeled 'Situation'. The bottom-right circle is labeled 'Usage'. To the right of the 'Utilisateur' circle is a list of text items. To the left of the 'Situation' circle is a list of text items. To the right of the 'Usage' circle is a list of text items.

Utilisateur

moi
moi dans 1 an
junior dev
senior dev
équipe run
dev autre langage

Situation

bureau
conférence
bug (stress)

Usage

Nouvelle Feature
Refactoring

L'UX Level 1

améliorer sa note en retenant une dizaine de choses

Analyse Heuristique

règles pour évaluer une interface



- ▼ Nielsen
- ▼ Schneidermann
- ▼ Bastien et Scapin

Analyse Heuristique

Bastien et Scapin (cocorico)



- ▼ Charge de travail
- ▼ Guidage
- ▼ Contrôle explicite
- ▼ Adaptabilité
- ▼ Gestion des Erreurs
- ▼ Homogénéité et Cohérence
- ▼ Signifiante des codes et dénomination
- ▼ Compatibilité



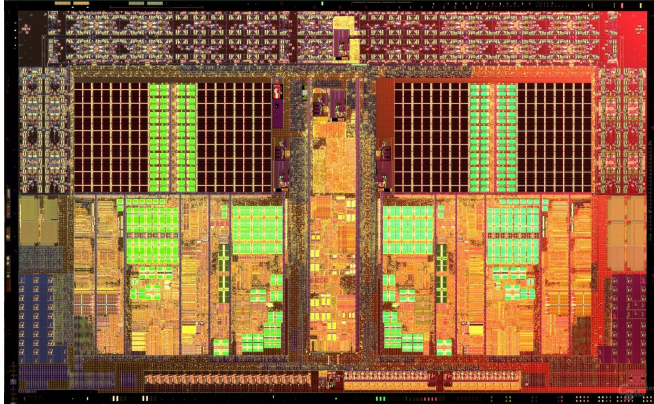
Charge de travail

“Charge de travail qui garantit la simplicité de l’interface et la pertinence des fonctionnalités tout en limitant le travail de lecture”

```
case class MyBusinessConf(  
  item: Int,  
  customer: String,  
  cdemo: Int,  
  returnAddr: String,  
  callCenter: Int,  
  catalogPage: Int,  
  shipMode: Int,  
  tax: Float)
```

Nommer les attributs de la classe ?

L'empan mnésique



- ▼ Cache L1 des êtres humains
- ▼ 4 ± 2
- ▼ on peut tricher

```
case class Item(  
  name: String,  
  color: String,  
  width: Int,  
  height: Int,  
  depth: Int,  
  posX: Int,  
  posY: Int,  
  posZ: Int,  
)
```

Nommer les attributs de la classe ?



Charge de travail

“Charge de travail qui garantit la simplicité de l’interface et la pertinence des fonctionnalités tout en limitant le travail de lecture”

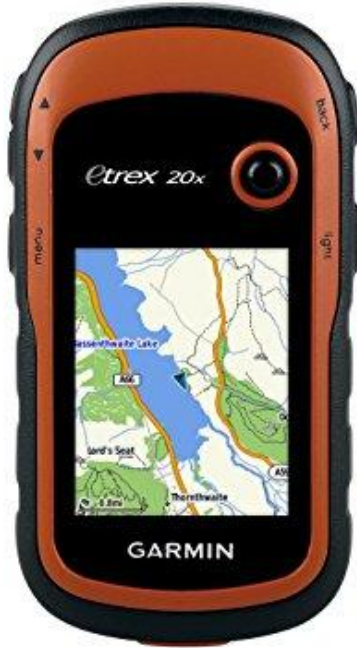
- ▼ complexité cyclomatique
- ▼ méthode courte
- ▼ single responsibility

Guidage



“Le guidage est l’ensemble des moyens mis en oeuvre pour conseiller, orienter, informer et conduire l’utilisateur lors de ses interactions avec l’ordinateur.”

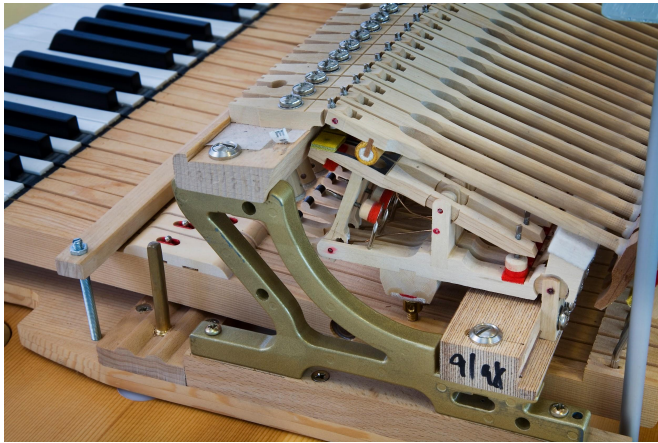
Guidage



“Le guidage est l’ensemble des moyens mis en oeuvre pour conseiller, orienter, informer et conduire l’utilisateur lors de ses interactions avec l’ordinateur.”

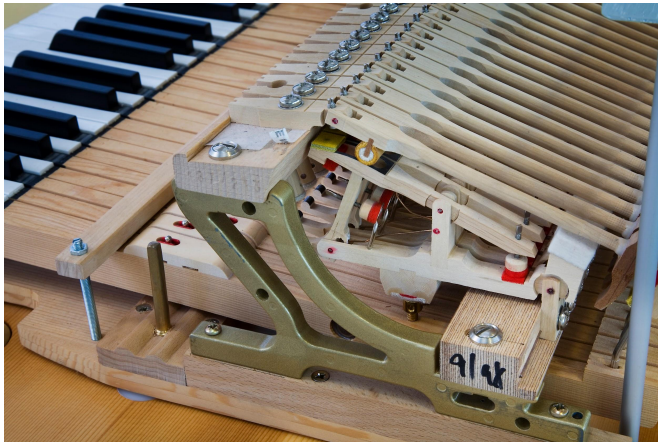
- ▼ commentaire
- ▼ nom explicite
- ▼ test
- ▼ doc

Contrôle explicite



“montrer que toute action du système correspond à une demande explicite de l'utilisateur”

Contrôle explicite



“montrer que toute action du système correspond à une demande explicite de l'utilisateur”

- ▼ exception
- ▼ logging

Adaptabilité



“Concerne la capacité de l’interface à réagir selon le context, et selon les besoins et préférences des utilisateurs”

Adaptabilité



“Concerne la capacité de l’interface à réagir selon le context, et selon les besoins et préférences des utilisateurs”

▼ VCS

▼ IDE

Homogénéité et cohérence



“respecter les choix de conception graphique effectués sur l’intégralité de l’interface”

Homogénéité et cohérence



“respecter les choix de conception graphique effectués sur l’intégralité de l’interface”

- ▼ style guide
- ▼ convention de nommage

Gestion des Erreurs



*“Tous les moyens permettant d’une part d’éviter ou de réduire les erreurs.
Elle se décline en 3 sous critères :*

- ▽ *protection contre les erreurs*
- ▽ *qualité des messages d'erreur*
- ▽ *correction des erreurs”*

Gestion des Erreurs



*“Tous les moyens permettant d’une part d’éviter ou de réduire les erreurs.
Elle se décline en 3 sous critères :*

- ▽ protection contre les erreurs*
 - ▽ qualité des messages d'erreur*
 - ▽ correction des erreurs”*
-
- ▼ Test
 - ▼ Nommage des variables
 - ▼ Erreur explicite
 - ▼ Log/observabilité

Signification des codes et dénominations



“Avoir une interface compréhensible et utile”

| Quel code est le plus adapté ?

1

```
for x in range(len(fruits)):  
    fruits[x] = f"{fruits[x]}s"
```

2

```
tmp = []  
for fruit in fruits:  
    tmp.append(f"{fruit}s")  
fruits = tmp
```

3

```
fruits = [f"{fruit}s" for fruit in fruits]
```

4

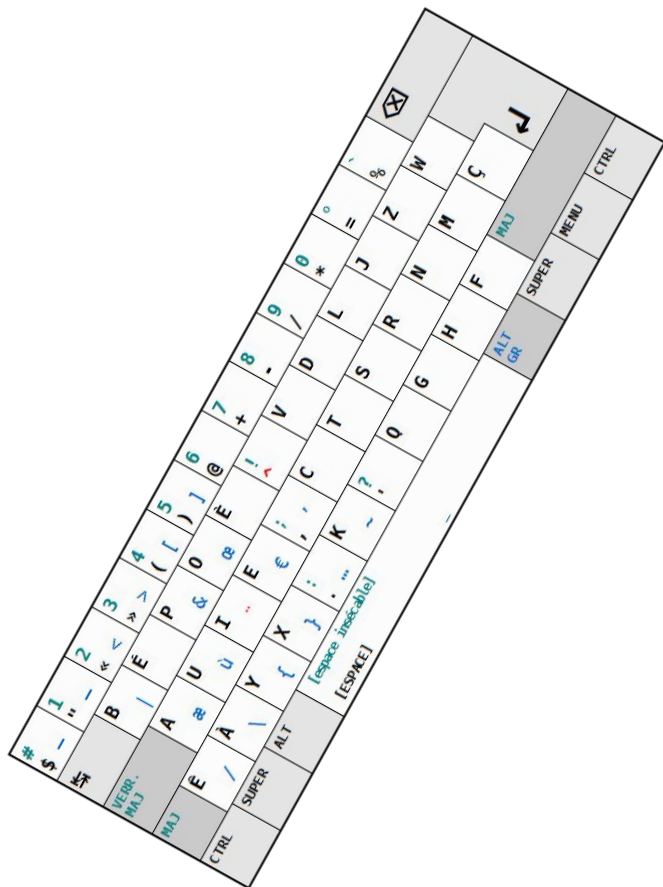
```
def pluralize(input):  
    return f"{input}s"  
  
fruits = map(pluralize, fruits)
```




Signifiante des codes et dénominations

“Avoir une interface compréhensible et utile”

- ▼ Design pattern
- ▼ Architecture (MVC, Clean, Hexagonal, Functional Core, ...)
- ▼ Convention de langage ou framework



Compatibilité

“adapter l’interface aux caractéristiques de l’utilisateur”



Compatibilité

“adapter l’interface aux caractéristiques de l’utilisateur”

- ▼ Limiter l’usage de la Programmation Fonctionnel
- ▼ Limiter les récursions
- ▼ Choix du langage

Vive la psychologie cognitive

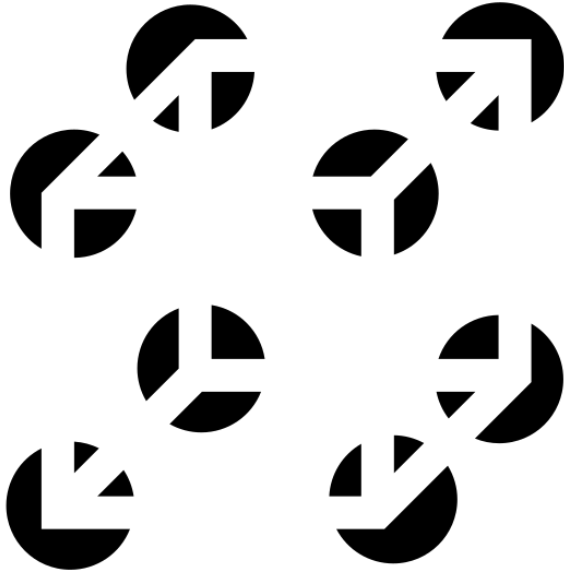
```
def isEdible(color:String,name:String,itemType:String):Boolean={  
  (itemType=="fruit"&&((color=="red"&&name=="apple")||color!="red"))|| (itemType=="vegetable"&&color=="green")  
}
```

Que fait ce code ?

```
def isEdible(color: String, name: String, itemType: String): Boolean = {  
  (  
    itemType=="fruit" && (  
      (color=="red" && name=="apple") || color!="red"  
    )  
  ) || (  
    itemType == "vegetable" && color == "green"  
  )  
}
```

Que fait ce code ?

Gestalt theory (1890)

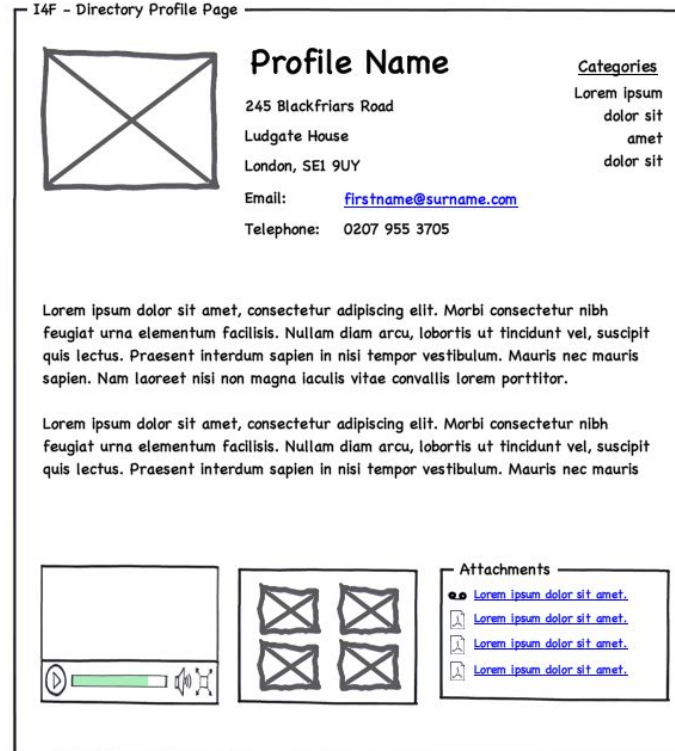


- ▼ **La loi de la bonne forme** : loi principale dont les autres découlent : un ensemble de parties informe (comme des groupements aléatoires de points) tend à être perçu d'abord (automatiquement) comme une forme, cette forme se veut simple, symétrique, stable, en somme une bonne forme.
- ▼ **La loi de continuité** : des points rapprochés tendent à représenter des formes lorsqu'ils sont perçus, nous les percevons d'abord dans une continuité, comme des prolongements les uns par rapport aux autres.
- ▼ **La loi de la proximité** : nous regroupons les points d'abord les plus proches les uns des autres.
- ▼ **La loi de similitude** : si la distance ne permet pas de regrouper les points, nous nous attacherons ensuite à repérer les plus similaires entre eux pour percevoir une forme.
- ▼ **La loi de destin commun** : des parties en mouvement ayant la même trajectoire sont perçues comme faisant partie de la même forme.

L'UX Level 2

pour les premiers de la classe

Prototype, Wireframe



| Test Utilisateur



| Guerrilla testing



| Ballade cognitive



| Focus Group



Merci pour votre attention