CHAPITRE I: INTRODUCTION AU GENIE LOGICIEL

Sommaire

- La crise logiciel
- Qu’est-ce qu’un logiciel?
- La Qualité du logiciel
- Le génie logiciel

La crise du logiciel

Les débuts de la crise

Destruction de Mariner 1 (1962) «Observation de Vénus »

➢ ***Cause directe***

- le programme de contrôle **ne lisse** plus les valeurs de la **vitesse** et réagit brutalement à des variations mineurs

➢ ***Impact***

- Destruction **297s** après le décollage
- Coût : 554 millions de $

➢ ***Origine***

- Erreur de transcription d'une formule dans le code source

Les débuts de la crise

Destruction de Mariner (1962)

➢ *Trouvez-vous le bug?*

```
...
   IF (TVAL .LT. 0.2E-2) GOTO 40
   DO 40 M = 1, 3
   W0 = (M-1)*0.5
   X = H*1.74533E-2*W0
   DO 20 N0 = 1, 8
   EPS = 5.0*10.0**(N0-7)
   CALL BESJ(X, 0, B0, EPS, IER)
   IF (IER .EQ. 0) GOTO 10
20 CONTINUE
   DO 5 K = 1. 3
   T(K) = W0
   Z = 1.0/(X**2)*B1**2+3.0977E-4*B0**2
   D(K) = 3.076E-2*2.0*(1.0/X*B0*B1+3.0977E-4*
  *(B0**2-X*B0*B1))/Z
   E(K) = H**2*93.2943*W0/SIN(W0)*Z
   H = D(K)-E(K)
 5 CONTINUE
10 CONTINUE
   Y = H/W0-1
40 CONTINUE
   ...
```

```
DO 5 K = 1,3
...
5 CONTINUE
```

```
DO5K=1.3
...
5 CONTINUE
```

(Fortran)
Boucle interprétée comme **l'affectation à une nouvelle variable**

C'est une autre cause invoquée par l'accident de Mariner

Les débuts de la crise

<u>Nombreux d'autres exemples</u>

➢***Convocation de centenaires à l'école***

-Codage de l'âge sur 2 chiffres.

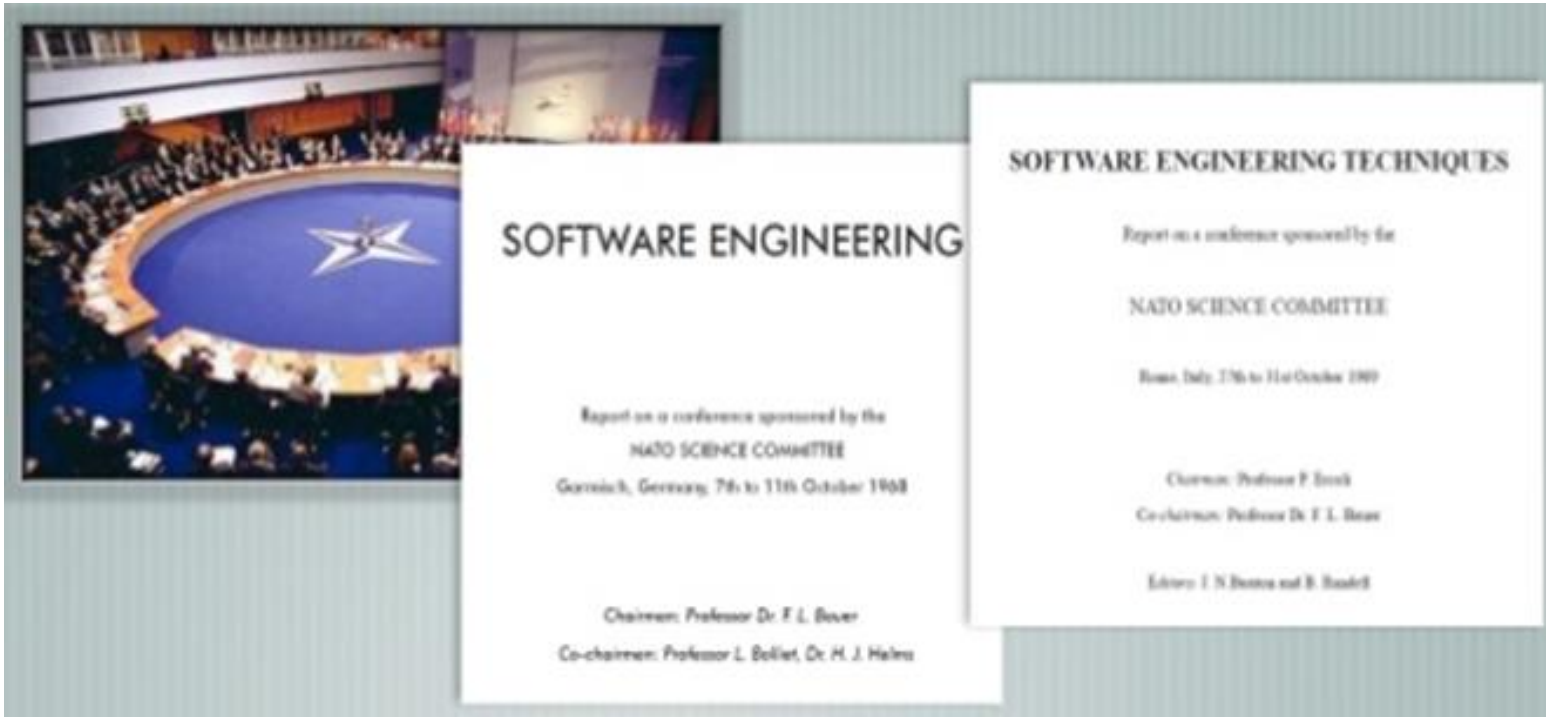
➢***Retournement sur le dos d'un Avion au passage à l'équateur***

-Changement de signe non pris en compte.

Réactions à la crise

➢ ***Comment faire des logiciels de qualités***

- L'OTAN (*L'Organisation du Araité de l'Atlantique Nord*) jette les bases du génie logiciel en **1968**



La crise se poursuit…

Rejet du système socrate à la SNCF – 1990

➢ ***Cause directe***

- Importantes difficultés de déploiement et d'utilisation.

➢ ***Impact***

- Report de la clientèle vers d'autres moyens de transport.
- Coût : Non communiqué

➢ ***Origine***

- Rachat d'un système développé par une compagnie aérienne

La crise se poursuit…

La destruction d'Ariane 5 – 1996

➢ ***Cause directe***

- Conversion entier /flottant non autorisée.

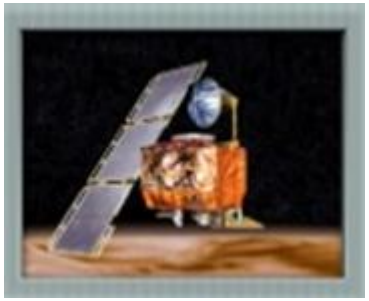
➢ ***Impact***

- Explosion **30s** après le décollage.

➢ ***Origine***

- **Reprise** du code spécifié pour Ariane 4
- **Absence de tests** de **validation**
- **désactivation** de la gestion des **exceptions**
- Conservation du **code inutile**

La crise se poursuit…



Perte de Mars Climate Orbiter 1999

➢ ***Cause directe***

- Confusion entre **pieds** et **mètres**
- Sonde, programmée pour utiliser le système **métrique**, reçoit des données en **unités de mesure anglo-saxonnes.**

➢ ***Impact***

- Destruction de la sonde à l'entrée de l'atmosphère martienne.
- Coût : 120 millions de $

➢ ***Origine***

- **Communication défaillante** entre équipe USA et GB.
- Spécifications approximatives

La crise se poursuit…

<u>Bug de l'an 2000</u>

➢ ***Cause directe***

- Codage de la date sur deux caractères

➢ ***Impact***

- Très nombreux mesures correctives et préventives
- Coût : estimé à 500 milliards de F

➢ ***Origine***

- Mauvaise perception de la **durée de vie des logiciels**

Bilan de la crise

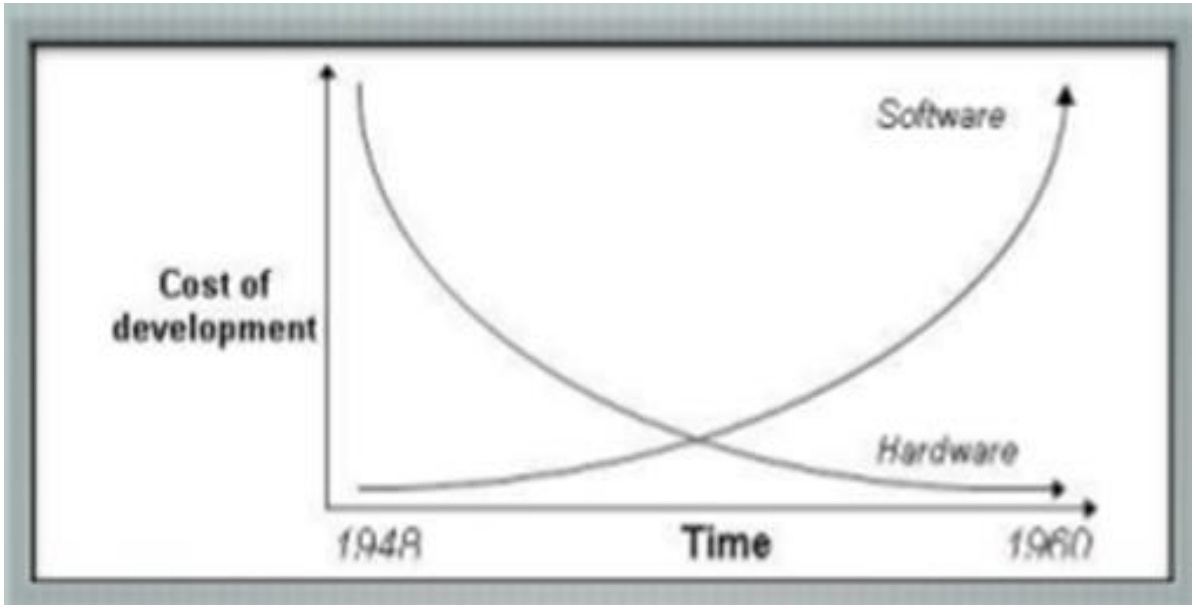
Causes

- Incapacité à faire face à la **complexité croissante** des logiciels
- **Incompréhension des besoins** des utilisateurs
- Incapacité à prendre en compte **l'évolution des besoins**
- **Absence de pratiques standards** ayant fait leur preuves

Bilan de la crise

Conséquences : les logiciels sont …

- **Inadaptés aux besoins** des utilisateurs.
- **Figés** ou très difficiles à faire évoluer
- De plus en **plus chères** à développer
- De **faible qualité**



La qualité d'un logiciel

- Qu'est ce que la **qualité** d'un logiciel?
- Comment faire des logiciels de qualité?

Qu’est ce qu’un logiciel?

Les logiciels sont omniprésents

- Utilisés par des milliards d'être humains chaque jour

- Présent dans tous les **secteurs** de la société
  - économie, transports, énergie, santé, recherche, télécommunication, enseignement, média….

- Au cœur des **systèmes critiques**
  - centrales nucléaires, trafic aérien, armement, radiothérapie….

Qu’est ce qu’un Logiciel?

Ensemble constitué de:

- **Programmes**
  - Code source (instructions et commentaires)
  - Code Exécutables
- **Données**
- **Documentations**
  - Spécification
  - Dossier de conception
  - Règle de codage
  - Manuel d’utilisation
  - Dossier de teste
  - Notice d’installation
  - ….

Qu’est ce qu’un Logiciel?

Produit fabriqué par :

- **Les éditeurs de logiciels**
  - Dans une logique d’offre pour les **utilisateurs génériques** (marché)
  - Les **spécification** sont choisies par **l’éditeur**
  - **Licences commerciales**

- **Prestataire en développement**
  - Dans une logique de demande pour les **clients spécifiques**
  - Les **spécifications** sont choisies par le **client**
  - **Licences commerciales**

Qu’est ce qu’un Logiciel?

Produit fabriqué par :

- **Une communauté de développeurs (Open Source)**
  - Dans une logique de partage pour les **utilisateurs génériques**
  - Les **spécifications** sont choisies par les **développeurs**
  - **Licences libres**

Catégories de logiciels

Logiciels système	Système d'exploitation, SGBD, logiciels réseaux, outils de programmations,..
Applications	Bureautique, Jeu, Web, Progiciels métier, logiciel scientifique, Intelligence artificielle, Multimédia, embarqué, temps réel…..
pourriciels	Virus, …

La qualité du logiciel

Qualité du logiciel

- **Norme ISO 9126**

- **Deux types de caractéristiques qualité**
  - **Externes**: Vues par l'utilisateur du logiciel
  - **Internes**: Vues par le producteur

- **Ces caractéristiques sont:**
  - Interdépendantes
  - Complémentaires
  - Souvent difficilement mesurables

Caractéristiques externes

Capacité fonctionnelle

- **Adéquation**
  - capacité du logiciel à réaliser ce qu'il a été prévu (spécifications)

- **Exactitude**
  - Capacité des fonctionnalités à fournir un comportement exacte

- **Interopérabilité**
  - Capacité du logiciel à fonctionner avec d'autres logiciels (protocoles, format de fichiers…)

- **Sécurité**

Capacité du logiciel à protéger ces fonctionnalités, ses données et ses programmes contre les accès non autorisés.

Caractéristiques externes

Fiabilité

- **Maturité**
  - Fréquence de comportements anormaux

- **Robustesse (ou tolérance aux pannes)**
  - Capacité du logiciel à réagir correctement (selon spécification) à des conditions anormales (hors spécifications)

Les conditions anormales peuvent être externe(environnement) ou interne.

- **Capacité de récupération**
  - Capacité du logiciel à retourner à un état de fonctionnement Normal après une anomalie.

Caractéristiques externes

Utilisabilité

- **Facilité d'opération**
  - Effort qu'un opérateur doit fournir pour **faire fonctionner le logiciel** dans son environnement.

- **Facilité d'apprentissage**
  - Effort qu'un utilisateur doit fournir **pour apprendre à réaliser ses tâches** avec le logiciel.

- **Facilité de compréhension**
  - Effort à fournir pour se représenter **les concepts logiques** des fonctionnalités du logiciel.

- **Pouvoir d'attraction**
  - Envie suscité par le logiciel

Caractéristiques externes

Efficacité

- **Performance temporelle**
  - Temps de réalisation de la fonctionnalité.

- **Economie de ressources**
  (nécessaire pour accomplir les tâches prévues)

  - Puissance CPU
  - Mémoire
  - Débit réseau
  - …..

Caractéristiques internes

Maintenabilité

- **Flexibilité (ou modifiabilité ou extensibilité)**
  - Effort à fournir pour modifier le logiciel (pour répondre à une *évolution* des spécification ou pour *corriger* une anomalie)

- **Facilité d'analyse**
  - Effort à fournie pour analyser les **causes d'une anomalie**
  - Effort à fournie pour **identifier** une **partie à modifier**

- **Stabilité**
  - Sensibilité du logiciel à un changement effectué dans une de ses parties

- **Testabilité**
  - Effort à fournir pour tester le logiciel (préparation des données, procédures de testes..

Caractéristiques internes

Portabilité

- **adaptabilité**
  - Capacité du logiciel à s'adapter à un changement d'environnement (plateforme matérielle, OS, Compilateur…..)

- **Facilité d'installation**
  - Effort à fournir pour installer et désinstaller le logiciel

- **Facilité de migration (ou remplaçabilité)**
  - Capacité du logiciel à remplacer un autre logiciel remplissant les fonctions similaires.

- **Réutilisabilité**
  - Capacité des parties du logiciel à être utilisées dans le développement d'un autre.

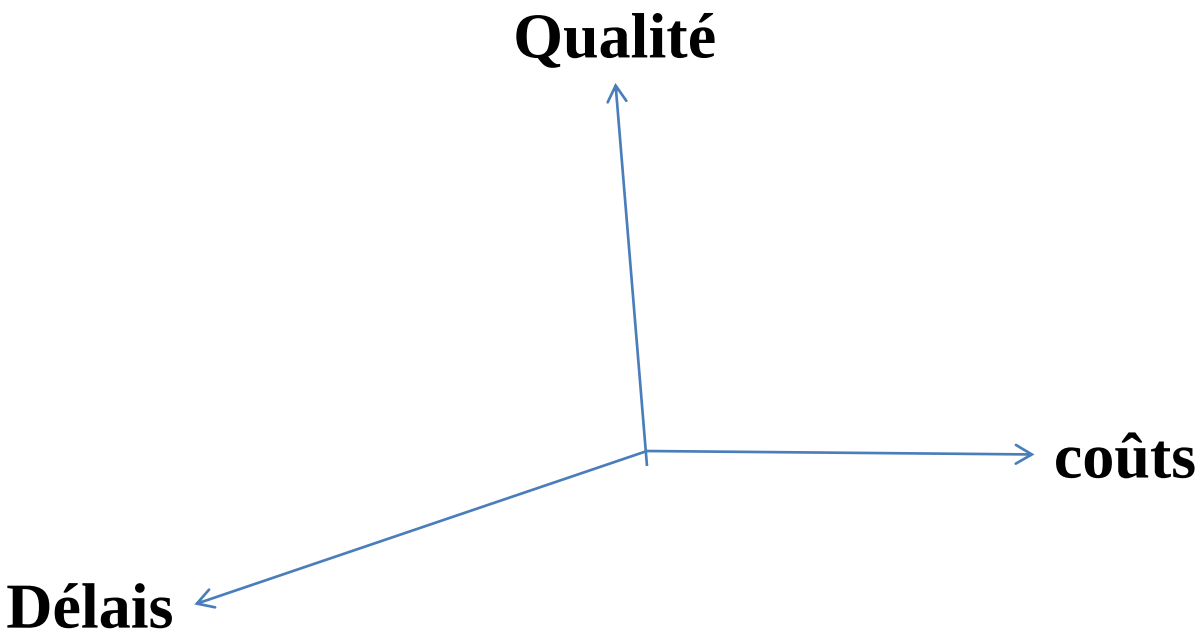
Qualité du logiciel

Et la documentation?

- **Elle est intégrés dans la pluparts des caractéristiques précédentes**

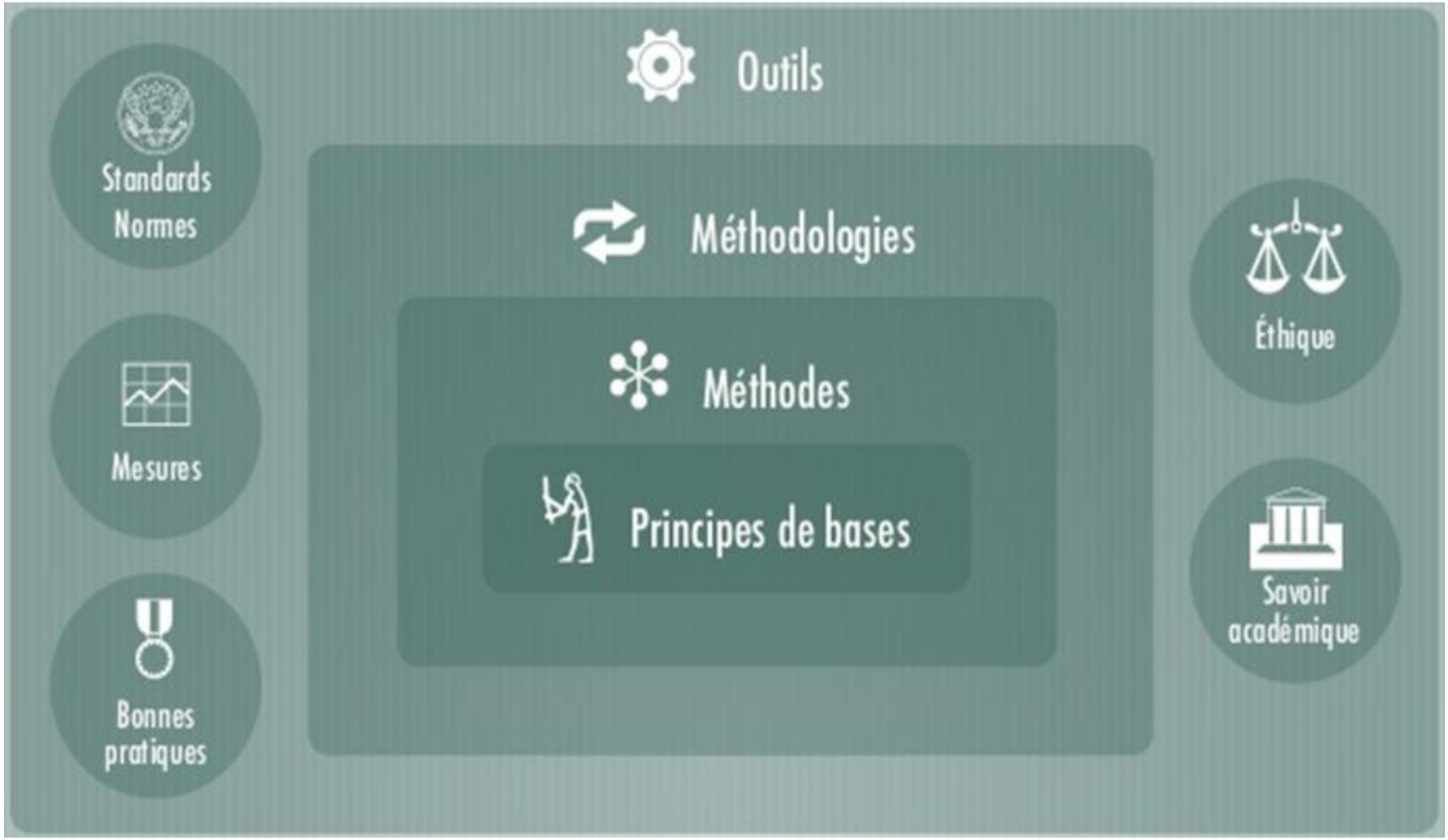
Qualité du logiciel

Qualité, coûts et délais sont étroitement liés



Le génie logiciel
L’ingénierie au service du logiciel

Qu’est ce que l’ingénierie?



Apports du génie industriel

- **Grandes étapes dans l'industrie**

  - **Division** du **travail** en tâches
  - **Division** des **produits** en sous ensembles
  - Interchangeabilité des sous-ensembles
  - Contrôle qualité
  - Qualité totale
  - ……

Une source d'inspiration pour le génie logiciel ou Ingénierie du logiciel

Qu’est-ce que le Génie Logiciel?

- L’étude des **pratiques systématiques** qui permettent **d’obtenir** des logiciels.

Qu’est-ce que le Génie Logiciel?

- **L’art et la manière de bien créer le bon logiciel**

- Le bon logiciel **répond** aux **critères** de **qualité** vues par **l’utilisateur et le producteur**

Le génie logiciel a fait son apparition dans les années 1980 aux États-Unis suite à la *crise du logiciel.*

Qu’est-ce que le Génie Logiciel?

Résume

Le génie logiciel est **l’ensemble des règles qu’il faut respecter** pour avoir une chance de développer un programme **sans trop de bug** et que l’on **pourra facilement** faire **évoluer** ultérieurement.

Activités fondamentales

Analyse

Des besoins aux **spécifications**

Clients et ingénieurs **définissent** le logiciel et son mode opératoire

Réalisation

Des spécifications aux **programmes**

Les ingénieurs conçoivent et programme le logiciel

Tests

Des programmes au produit

Les ingénieurs et le Clients contrôlent que le logiciel correspond aux besoins

Distribution, Maintenance, Evolution

Les ingénieurs adaptent le logiciel aux évolutions des clients/ du Marché

Les huit principes du génie logiciel

- Utiliser les **processus appropriés**
- Se **soucier** de **l'utilisateur**
- Concevoir pour la **qualité**
- Maîtriser les **dépendances**
- Maîtriser les **changements**
- **Tester** sans relâche
- **Mesurer** la **qualité**
- **Documenter**

Les huit principes du génie logiciel

Utiliser les processus appropriés

- Développer en suivant un processus standard
- Bien compris par l'équipe
- Adaptés:
  - A l'organisation du **projet**
  - A **l'équipe** projet
  - Au **type** du **logiciel**

Les huit principes du génie logiciel

Se soucier de l'utilisateur

- **Comprendre** et gérer les **besoins**
- **Aider l'utilisateur** à **exprimer** ses **besoins**
- Utiliser le **vocabulaire** de **l'utilisateur**
- **Prévoir** que ces besoins **évolueront**
- **Intégrer l'utilisateur** dans le **processus** de développement

Les huit principes du génie logiciel

➢ Concevoir pour la qualité

- Définir et **hiérarchiser** les objectifs de la **qualité**
- **Prévoir** et **évaluer différentes solutions** de **conception**
- Utiliser les bonnes pratiques de conception

Les huit principes du génie logiciel

➢ Maîtriser les dépendances

- Identifier et réduire les dépendances

Les huit principes du génie logiciel

➢ Maîtriser les changements

- Choisir un **processus** de développement **adapté** à la **souplesse voulue**

Les huit principes du génie logiciel

➢ Tester sans relâche

- **Tester** le plus **souvent** et le plus complètement possible
- **Tester** en **relation** avec les **exigences**
- **Tester** à **différentes niveau** de granularité
- Le **testeur ne doit pas** être un **développeur**

Les huit principes du génie logiciel

➢ Mesurer la qualité

- ***Par l'analyse technique***
  - analyse statique du source
  - analyse dynamique des l'exécutables

- ***Par les retours d'utilisations***
  -nombre de **bug** , satisfaction…

Les huit principes du génie logiciel

- **Documenter**
- Les trois aspects du logiciels
  -Comportements
  -Structures
  -Architecture

- Les testes du logiciel