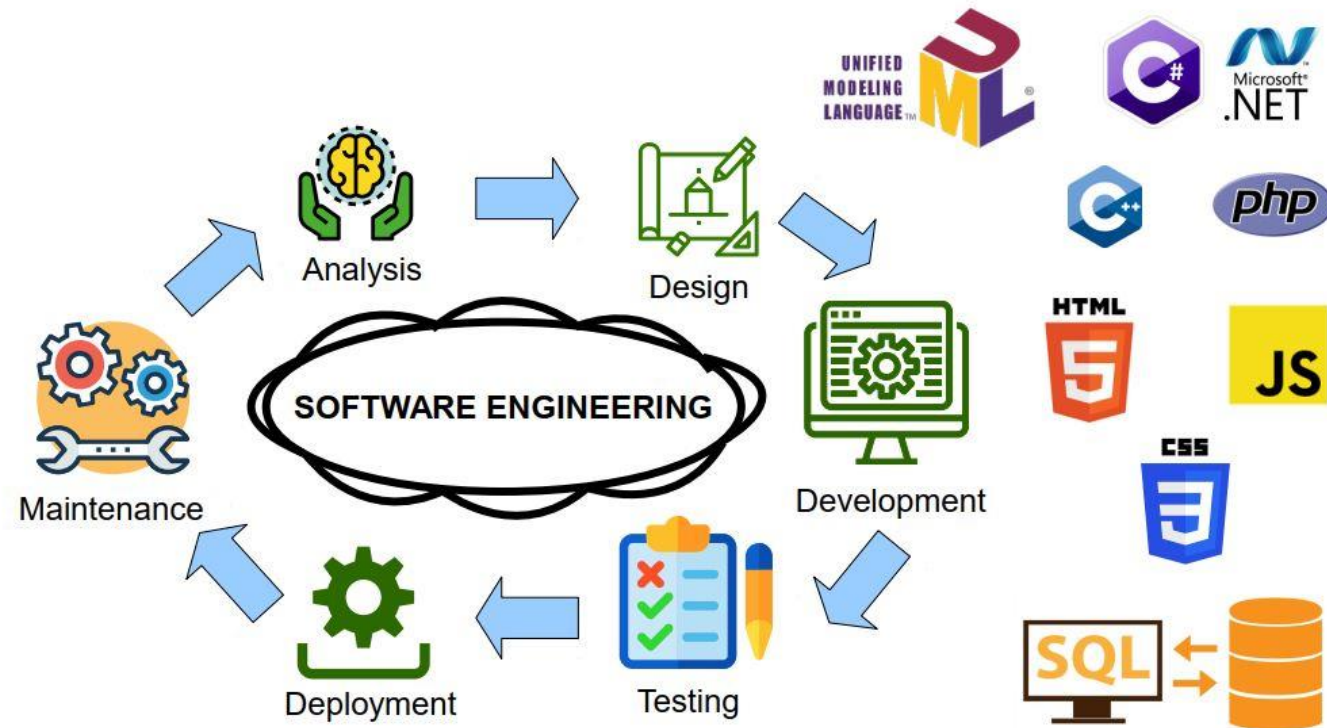


# Introduction au Génie Logiciel



---

Dr El Hadji Bassirou TOURE  
Ecole Supérieure Polytechnique (ESP)  
2020 - 2021

# Le terme « génie logiciel »

- Apparue en 1969, à l'issue d'une conférence pour discuter de la « crise du logiciel »
- Les raisons de cette crise :
  - Les coûts du matériel chutaient et le matériel était de plus en plus performant
  - Les coûts du logiciel grimpaient en flèche, les logiciels de plus en plus complexes.
  - Il fallait de nouvelles techniques et de nouvelles méthodes pour contrôler la complexité inhérente aux grands systèmes logiciels

# La « crise du logiciel »

- Qualité du logiciel livré était souvent déficiente
- Performances moyennes (temps de réponse trop lents)
- Non-respect des délais prévus de livraison
- Pas moyen d'estimer les coûts
- Logiciels pas conformes aux attentes
- Maintenance difficile, voire inexistante

# Génie logiciel

- Le génie logiciel s'intéresse aux théories, méthodes et outils pour le développement professionnel de logiciels.
- De plus en plus de systèmes sont contrôlés par le biais de logiciels.
- Les économies des nations développées sont aujourd'hui influencées par les dépenses sur les logiciels
- Ces dépenses représentent une part importante du PNB de tous les pays développés

# Coûts logiciels

- Les coûts logiciels dépassent souvent les coûts du dispositif informatique.
- Généralement, un logiciel coûte plus lors de la maintenance que lors du développement. Surtout pour des logiciels avec une longue durée de vie.
- Le coût du logiciel est estimé par rapport aux coûts effectifs lors du développement

# Raison des échecs de projets logiciels

- Croissance de la complexité des systèmes
  - Etant donné que les nouveaux logiciels nous permettent d'avoir des systèmes plus grands et plus complexes, les exigences évoluent elles aussi. Les systèmes doivent être construits et délivrés rapidement; sont de plus en plus grands, et surtout des systèmes plus complexes sont requis; Les systèmes doivent avoir de nouvelles aptitudes initialement crues impossible.
- Non utilisation des méthodes du génie logiciel
  - Il est assez facile d'écrire des programmes sans utiliser les méthodes et techniques du génie logiciel. Plusieurs compagnies peuvent entamer le développement de leurs logiciels afin de s'adapter à l'évolution de leurs produits et services. Ces organisations n'adoptent pas les méthodes du génie logiciel. Au final, leurs logiciels coûtent souvent chers et sont moins fiables qu'ils ne devraient.

# Développement professionnel de logiciels

# FAQs sur le génie logiciel (exercice)

Question	Réponse
Qu'est ce qu'un logiciel ?	
Quelles sont les caractéristiques d'un bon logiciel ?	
Qu'est-ce que le génie logiciel ?	
Quelles sont les principales activités du génie logiciel ?	
Quelle est la différence entre génie logiciel et informatique (discipline) ?	
Quelle est la différence entre le génie logiciel et l'ingénierie système ?	



# FAQs sur le génie logiciel

Question	Réponse
Qu'est ce qu'un logiciel ?	C'est un programme informatique ainsi que la documentation associée. Un produit logiciel peut être développé pour un client en particulier ou pour un marché plus général.
Quelles sont les caractéristiques d'un bon logiciel ?	Un bon logiciel doit fournir les fonctionnalités et performances attendues et doit être maintenable, fiable et utilisable.
Qu'est-ce que le génie logiciel ?	Le génie logiciel est une discipline d'ingénierie qui s'intéresse à tous les aspects de la production du logiciel.
Quelles sont les principales activités du génie logiciel ?	Spécification logicielle, développement logiciel, validation logicielle et évolution logicielle.
Quelle est la différence entre génie logiciel et informatique (discipline) ?	L'informatique se concentre sur des théories et fondamentaux ; Le génie logiciel s'intéresse aux aspects pratiques du développement et de la livraison de logiciels utiles.
Quelle est la différence entre le génie logiciel et l'ingénierie système ?	L'ingénierie système s'intéresse à tous les aspects du développement de systèmes logiciels incluant le hardware, le software, les processus, etc. Le génie logiciel est une partie de ce grand ensemble.

# FAQs sur le génie logiciel

Question	Réponse
Quels sont les défis auxquels le génie logiciel doit faire face ?	Faire face à la croissance de la diversité et aux exigences pour la réduction des délais de livraisons. Produire des logiciels fiables.
Quels sont les coûts du génie logiciel ?	Généralement 60% des coûts sont des coûts de développement, 40% sont des coûts dûs aux tests. Pour des logiciels personnels, les coûts de l'évolution dépassent largement les coûts de développement.
Quelles sont les meilleures méthodes et techniques du génie logiciel ?	Tout projet logiciel doit être géré et développé de manière professionnelle mais différentes techniques sont appropriées pour différents types de systèmes. Par exemple, un projet logiciel de jeu vidéo doit toujours être développé en utilisant une série de prototypes alors qu'un projet logiciel de développement de systèmes critiques requiert une spécification précise pour son développement. On ne peut donc pas dire qu'une méthode est meilleure qu'une autre.
Quelles différences le web a-t-il apportées au génie logiciel ?	Le web a conduit à la disponibilité de services logiciels et a permis le développement de systèmes distribués à base de services. Le développement de systèmes pour le web a permis d'importantes avancées au niveau des langages de programmation et de la réutilisation logicielle.

# Produits logiciels

- Produits génériques
  - Les systèmes logiciels commercialisés et pouvant être vendus à n'importe quel client souhaitant l'acheter.
  - Exemples : Logiciels PC de traitement de textes, logiciels pour un marché spécifique comme les logiciels de gestion bibliothèques.
- Produits personnalisés (sur mesure)
  - Les systèmes logiciels commandés par un client en particulier pour assurer un besoin spécifique.
  - Exemples : Les systèmes de gestion d'un business métier bien défini.

# Spécification de produits logiciels

- Produits génériques

- La spécification du logiciel est faite par le constructeur du logiciel. C'est ce dernier qui décide de quand et comment appliquer des modifications au logiciel.

- Produits personnalisés (sur mesure)

- La spécification du logiciel est faite par le client qui décide également des modifications à apporter au logiciel (avec le concours du constructeur).

# Caractéristiques importantes d'un bon logiciel

Caractéristiques du produit	Description
Maintenabilité	Le logiciel doit être écrit de sorte à ce qu'il puisse évoluer et s'adapter aux changements de ses clients. Il s'agit d'une caractéristique critique parce que les modifications logicielles sont une exigence inévitable pour des environnements métiers changeants.
Fiabilité et sécurité	La fiabilité du logiciel inclut plusieurs autres caractéristiques telles que la solidité, la sécurité et la sûreté. Un logiciel fiable ne doit pas causer des dégâts physiques ou économiques si toutefois le système tombait. Des utilisateurs malveillants ne doivent pas être capables d'accéder ou d'endommager le système.
Efficience	Le logiciel ne doit pas gaspiller les ressources du système telles que la mémoire ou les cycles du processeur. L'efficience inclut donc la réactivité, le temps de calcul, l'utilisation de la mémoire, etc.
Acceptabilité	Le logiciel doit être acceptable du point de vue des utilisateurs pour lesquels il a été conçu. Ceci veut dire qu'il doit être compréhensible, utilisable et compatible avec les autres systèmes qu'ils utilisent.

# Génie logiciel

- Le génie logiciel est une discipline d'ingénierie qui s'intéresse à tous les aspects de la production du logiciel allant de la spécification à la maintenance du logiciel.
  - Discipline d'ingénierie
    - Utilisation de théories et méthodes appropriées pour la résolution de problèmes en tenant en compte des contraintes organisationnelles et financières.
  - Tous les aspects de la production du logiciel
    - Pas seulement les procédés techniques de développement utilisés mais également la gestion de projet et le développement d'outils, de méthodes, ..., pour faciliter la production de logiciels.

# Importance du génie logiciel

- De plus en plus, les individus et la société comptent sur des logiciels sophistiqués. Il doit être possible de produire des systèmes sûrs et dignes de confiance de manière économique et rapidement.
- Il est souvent moins coûteux, dans le long terme, d'utiliser les méthodes et techniques du génie logiciel plutôt que d'écrire directement des programmes comme s'il s'agissait d'un projet de développement de logiciel personnel. Pour la plupart des systèmes, la majorité des coûts sont des coûts de modification après que le logiciel ait été soumis à utilisation.

# Processus logiciel

- Une approche de développement utilisée en génie logiciel est souvent appelée un processus logiciel.
- Un processus logiciel est une séquence d'activités qui débouchent sur la production de produits logiciels.
- Quatre principales activités sont communes à tous les processus logiciels.
  - Spécification logicielle
  - Développement logiciel
  - Validation logicielle
  - Evolution logicielle



# Activités de processus logiciel

- Spécification logicielle, où les clients et les développeurs définissent le logiciel qui doit être produit ainsi que les contraintes de sa mise en opération.
- Développement logiciel, où le logiciel est conçu et programmé.
- Validation logicielle, où le logiciel est vérifié pour assurer qu'il satisfait les exigences du client.
- Evolution logicielle, où le logiciel est modifié pour s'adapter aux changements des besoins et des exigences du marché.

# Principaux problèmes qui affectent le logiciel

- Hétérogénéité
  - De plus en plus, les systèmes nécessitent de s'exécuter comme des systèmes distribués à travers le réseau, ce qui inclut divers types d'ordinateurs et appareils mobiles.
- Changements du domaine métier et social
  - Le domaine métier et la société évoluent aussi rapidement que le développement des économies émergentes, ce qui fait que de nouvelles technologies sont disponibles. Les compagnies ont besoin de modifier leurs logiciels existant et de rapidement en développer de nouveaux.

# Principaux problèmes qui affectent le logiciel

- Sécurité et confiance
  - Etant donné que les logiciels sont liés à nos vies, il est essentiel que nous ayons confiance en eux.
- Echelle
  - Les logiciels sont généralement développés suivant diverses classes d'échelles de systèmes. Cela peut aller des petits systèmes sur des appareils portables à des systèmes à l'échelle d'internet, des systèmes opérant sur le cloud servant à de très grandes communautés.

# Diversité du génie logiciel

- Il existe différents types de logiciels et il n'y a pas un ensemble universel de techniques logicielles applicables à tous ces types de système.
- Les méthodes et outils du génie logiciel utilisés dépendent du type d'application à développer, des exigences des clients et du background de l'équipe de développement.

# Types d'application

- Applications autonomes

- Il s'agit d'applications s'exécutant sur une machine locale, telle qu'un PC. Ces applications incluent toutes les fonctionnalités nécessaires et n'ont pas besoin d'être connectées à un réseau.

- Applications interactives de transactions

- Applications qui s'exécutent sur des machines distantes et qui sont accessibles aux utilisateurs à partir de leurs propres PC ou terminaux. Ce sont généralement des applications web telles que les applications d'e-commerce.

- Systèmes embarqués

- Ce sont des logiciels qui contrôlent et gèrent le fonctionnement de périphériques matériels.
- Numériquement, ce type d'applications est plus répandu que les autres.

# Types d'application

- Systèmes de traitement par lots
  - Ce sont des systèmes conçus pour traiter des données arrivant par de grands lots. Elles traitent de grands nombres de données d'entrée distinctes pour produire des données de sortie correspondantes.
- Applications de divertissement
  - Ce sont des applications destinées à une utilisation personnelle et dont le but est de distraire l'utilisateur.
- Applications de modélisation et simulation
  - Ce sont des applications développées par des scientifiques et ingénieurs pour modéliser et/ou simuler des comportements, ou situations, ou encore phénomènes réels ou naturels.

# Types d'application

- Systèmes de collection de données

- Ce sont des systèmes qui collectent des données depuis leur environnement en utilisant un ensemble de capteurs et envoient ces données aux autres systèmes pour traitements.

- Systèmes de systèmes

- Ce sont des systèmes composés d'un certain nombre d'autres systèmes.

# Principes fondamentaux du génie logiciel

- Quelques principes fondamentaux s'appliquent à tous les types de systèmes, peu importe les techniques de développement employées :
  - Les systèmes devraient être développés en utilisant un processus de développement bien maîtrisé et compris. Bien sûr différents processus sont utilisés pour différents types de logiciel.
  - La fiabilité et la performance sont importantes pour tout type de systèmes.
  - La maîtrise et la compréhension des spécifications logicielles et des exigences sont importantes.
  - Quand c'est possible, on doit réutiliser les logiciels déjà développés plutôt que d'en écrire de nouveaux.



# Génie logiciel d'Internet

- Le web est maintenant une plateforme pour l'exécution d'applications et les organisations développent de plus en plus de systèmes basés sur le web plutôt que des systèmes en local.
- Les services web permettent d'accéder aux fonctionnalités d'une application via le web.
- Le cloud computing est une approche permettant l'approvisionnement de services où des applications s'exécutent à distance sur le cloud.
  - Les utilisateurs n'achètent pas le logiciel mais paient pour son utilisation.

# Ethique du génie logiciel

# Ethique du génie logiciel

- Le génie logiciel implique de plus grandes responsabilités qu'une simple utilisation de compétences techniques.
- Les ingénieurs du logiciel doivent se comporter de manière honnête et (éthiquement) responsable s'ils veulent se faire respecter comme des professionnels.
- Un comportement éthique est plus que le respect d'un ensemble de lois mais implique également le suivi d'un ensemble de principes moraux.

# Enjeux d'une responsabilité professionnelle

- Confidentialité

- Les ingénieurs doivent normalement respecter la confidentialité de leurs employeurs ou clients, peu importe si un accord formel de confidentialité a été signé ou pas.

- Compétence

- Les ingénieurs ne doivent pas usurper leur niveau de compétence. Ils ne doivent pas accepter un travail qui est en dehors de leurs compétences.

# Enjeux d'une responsabilité professionnelle

- Droits de propriété intellectuelle
  - Les ingénieurs doivent être conscients des lois locales gouvernant l'utilisation de propriétés intellectuelles telles que les brevets, copyrights, ... Ils doivent s'assurer que les droits de propriété intellectuelle sont respectés.
- Méusage des ordinateurs
  - Les ingénieurs ne doivent pas utiliser leurs compétences techniques pour mésuser les ordinateurs d'autres personnes. Le méusage d'ordinateurs peut aller du trivial (jouer sur des machines de son employeur) à du très sérieux (propager du virus)

# Code d'éthique de l'IEEE/ACM

- Les organisations professionnelles des USA ont coopéré pour produire un code de pratiques éthiques.
- Les membres de ces organisations signent le code en adhérant.
- Le code contient huit principes et est établi par les professionnels du génie logiciel (praticiens, enseignants, chercheurs, étudiants, etc.).

# Code d'éthique de l'IEEE/ACM

## Software Engineering Code of Ethics and Professional Practice

ACM/IEEE-CS Joint Task Force on Software Engineering Ethics and Professional Practices

### PREAMBLE

The short version of the code summarizes aspirations at a high level of the abstraction; the clauses that are included in the full version give examples and details of how these aspirations change the way we act as software engineering professionals. Without the aspirations, the details can become legalistic and tedious; without the details, the aspirations can become high sounding but empty; together, the aspirations and the details form a cohesive code.

Software engineers shall commit themselves to making the analysis, specification, design, development, testing and maintenance of software a beneficial and respected profession. In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to the following Eight Principles:

# Code d'éthique de l'IEEE/ACM

1. PUBLIC - Software engineers shall act consistently with the public interest.
2. CLIENT AND EMPLOYER - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
3. PRODUCT - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.
4. JUDGMENT - Software engineers shall maintain integrity and independence in their professional judgment.
5. MANAGEMENT - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
6. PROFESSION - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
7. COLLEAGUES - Software engineers shall be fair to and supportive of their colleagues.
8. SELF - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.