

Algorithmique

Conception



© 2018 - André Aoun



Etapes d'un projet informatique

- Un projet informatique obéit à un cycle de vie ...



- ✓ Déterminer les différentes phases d'un projet informatique

© 2018 - André Aoun

- Algorithmique -

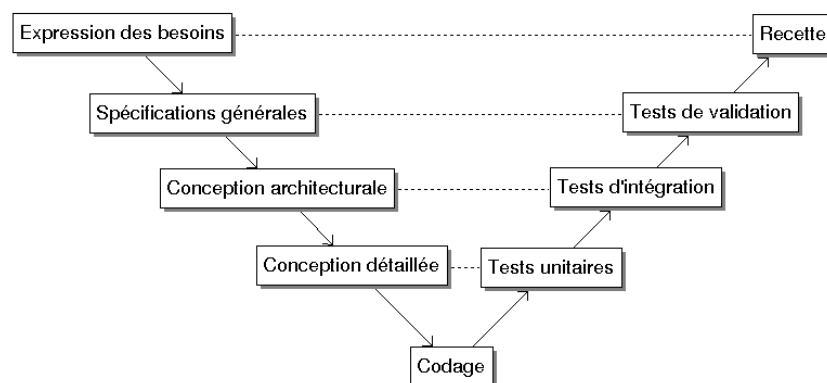


Etapes d'un projet informatique (1)

□ Un projet informatique comprend les phases suivantes :

- **Phase d'analyse** : Analyse des besoins et de l'existant, étude d'opportunité et de faisabilité
- **Phase de spécifications fonctionnelles** : Données initiales, résultats à obtenir, contraintes à respecter, cahier des charges...
- **Phase de conception** : Etude du processus de traitement des données permettant d'aboutir aux résultats demandés. Décomposition des problèmes en sous-problèmes.
- **Phase de réalisation** : Développement et Implémentation (Programmation), tests, documentation...

Etapes d'un projet informatique (2)



Etapes d'un projet informatique (3)



Comment le client a exprimé son besoin



Comment le chef de projet l'a compris



Comment l'ingénieur l'a conçu



Comment le programmeur l'a écrit



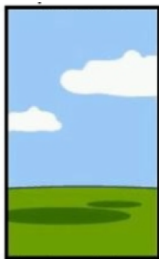
Comment le responsable des ventes l'a décrit

© 2018 - André Aoun

- Algorithmique -



Etapes d'un projet informatique (4)



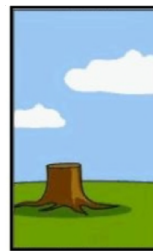
Comment le projet a été documenté



Ce qui a finalement été installé



Comment le client a été facturé



Comment la hotline répond aux demandes



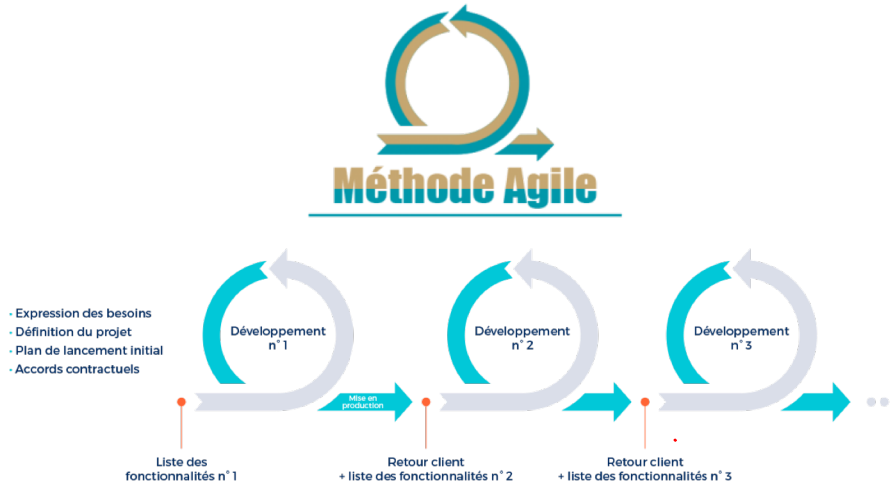
Ce dont le client avait réellement besoin

© 2018 - André Aoun

- Algorithmique -



Etapes d'un projet informatique (5)



© 2018 - André Aoun

- Algorithmique -



Caractéristiques d'un algorithme

- La conception d'un algorithme obéit à certaines règles...



✓ Caractériser un algorithme

© 2018 - André Aoun

- Algorithmique -



Caractéristiques d'un algorithme (1)

□ Un algorithme est caractérisé par :

- Une liste finie d'instructions destinées à un exécutant (processeur) sur un nombre fini de données, soit pour les créer, modifier ou supprimer, soit pour les utiliser dans l'élaboration des traitements;
- La solution du problème après exécution d'un nombre fini d'opérations;
- L'ordre dans lequel les opérations sont exécutées par le processeur peut tenir compte de l'état de l'environnement (contexte) au moment de l'exécution de chaque instruction: valeur des données, survenue d'événements, ...

Caractéristiques d'un algorithme (2)

□ Un algorithme doit être :

- **Lisible et commenté** : l'algorithme doit être compréhensible même par un non-informaticien,
- **De haut niveau** : l'algorithme doit pouvoir être traduit en n'importe quel langage de programmation, il ne doit donc pas faire appel à des notions techniques relatives à un programme particulier ou bien à un système d'exploitation donné,
- **Précis** : chaque élément de l'algorithme ne doit pas porter à confusion, il est donc important de lever toute ambiguïté,
- **Concis** : un algorithme ne doit pas dépasser une page. Si c'est le cas, il faut décomposer le problème en plusieurs sous problèmes,
- **Structuré** : un algorithme doit être composé de différentes parties facilement identifiables.

Démarche par abstraction

- La conception abstraite permet petit à petit d'approcher la réalisation concrète...



✓ Décrire la démarche par abstraction

Démarche par abstraction (1)

- La démarche par abstraction consiste à ne pas tenir compte des détails de réalisation n'influant pas sur le niveau d'analyse à un instant donné.
- Un élément dont on veut faire abstraction pourra être encapsulé dans une entité dont on définira les services offerts et les interfaces (entrée/sortie). Le traitement est réalisé au sein de cette entité et masqué au niveau d'abstraction appelant.
- Ces entités d'encapsulation appelées sous-algorithmes (procédures ou fonctions) pourront être utilisées plusieurs fois dans un algorithme. Elles pourront être regroupées dans une bibliothèque.

Démarche par abstraction (2)

- ❑ L'algorithme sera transformé par la suite en un code informatique or l'ordinateur n'est capable d'exécuter que des actions élémentaires
- ❑ Une tâche complexe sera décomposée en une suite d'actions plus simples à réaliser et cela jusqu'aux actions élémentaires. On va procéder par étapes.
- ❑ Le nombre d'étapes employées dépend de la complexité du traitement : plus ce dernier est complexe, plus le nombre d'étapes devra être important.

→ Raffinages successifs

Démarche par abstraction (3)

- ❑ **Méthode des raffinages successifs :**
Cette méthode de décomposition de problèmes permet de passer progressivement et avec un maximum de chances de réussite, de la description abstraite de l'algorithme à un niveau très précis permettant le codage.
Cette méthode de conception est aussi appelée **l'approche descendante (TOP-DOWN)**.
- ❑ On dit qu'un algorithme est au dernier niveau de raffinement lorsqu'il ne comporte que des actions élémentaires et des structures de contrôle de base permettant d'aboutir rapidement à un programme informatique.

Démarche par abstraction (4)

□ Ecrire les algorithmes correspondant :

1. Bâtir une maison de N étages
2. Dessiner une croix de 2 unités de longueur sur chaque branche

Vocabulaire :

CENTRER : met le stylet au centre de la table

LEVER : lève le stylet de la table (plus de contact)

BAISSER : baisse le stylet sur la table

NORD : déplace le stylet d'une unité vers le haut

EST : déplace le stylet d'une unité vers la droite

SUD : déplace le stylet d'une unité vers le bas

OUEST : déplace le stylet d'une unité vers la gauche

Démarche par abstraction (5)

□ Exemple de raffinages :

