

Cahier des charges

Le cahier des charges est un document recensant les spécifications/exigences. Il résulte de l'analyse et est contractuel entre le client et l'entreprise informatique qui va réaliser le logiciel. Il doit donc être validé par les deux. *Qualités attendues* : non ambigu, complet, vérifiable, cohérent, modifiable, traçable, utilisable durant la maintenance, indépendant des solutions (voir notamment la norme IEEE 830).

Plan type :

1. introduction : présentation générale, motivations, définitions des termes
2. contexte : environnement matériel et humain, acteurs et utilisateurs, interaction avec d'autres systèmes et logiciels, existant
3. spécifications fonctionnelles : grandes fonctionnalités du système, acteurs et autres systèmes qu'elles impliquent
4. spécifications non fonctionnelles, contraintes :
 - charte graphique
 - matériel : marques, RAM, débit de connexion Internet...
 - interfaçage : protocoles de communication, formats de fichiers, etc., pour l'interaction avec des matériels, logiciels, systèmes d'exploitation...
 - performances : temps réel...
 - sécurité : sauvegardes, confidentialité, ...
 - charge à supporter : volume des données, nombre d'utilisateurs simultanés, ...
 - comportement en cas de panne...
5. priorités relatives des spécifications, versions à prévoir, délais
6. évolutions à prévoir
7. annexes

A.1 Capture et analyse des besoins, rédaction

Rédaction d'un cahier des charges : être précis et organisé, s'aider de notations et diagrammes standards, des normes. En particulier :

- norme IEEE 830 (1998) pour la rédaction des spécifications ;

- organisation : numérotation, tableaux, schémas, exemples. . . (ce n'est pas un roman!);
- diagrammes UML : en particulier des cas d'utilisation pour présenter les fonctionnalités et les acteurs qu'elles impliquent ;
- autres schémas : écrans types, diagramme de déploiement UML...
- scénarios (d'interaction) : illustration des cas d'utilisation complexes par un exemple d'interaction le réalisant ; spécifier les acteurs impliqués et l'enchaînement des interactions sur un exemple (éventuellement par un diagramme de séquence UML), ainsi que le traitement des erreurs et les éventuelles préconditions et postconditions ;

Outils pour la capture des besoins :

- entretiens réguliers avec le client, validations ;
- observations des futurs utilisateurs dans leur pratique actuelle ;
- scénarios d'interaction ;
- exemples d'écrans ;
- prototypes ;