

# Analyse (R3.03)

## Plan de cours

Halim Djerroud

révision 2.2

### Introduction

La ressource **R3.03 Analyse** vise à doter l'étudiant des méthodes et outils nécessaires pour transformer un besoin exprimé par un client en spécifications techniques claires et exploitables. Il s'agit d'apprendre à recueillir les exigences, à les formaliser à l'aide de modèles (UML, cas d'utilisation, cahier des charges) et à en évaluer la faisabilité dans le cadre d'un projet logiciel.

Cette ressource constitue un maillon essentiel entre la compréhension des besoins métiers et la conception technique, et prépare directement aux travaux réalisés dans les **SAÉ du semestre 3**.

**Volume horaire** :  $\approx 23\text{h}$  (10h cours + 13h TD/TP).

#### Compétences visées (Apprentissages Critiques) :

- **AC1** : Élaborer et implémenter des spécifications.
- **AC2** : Formaliser les besoins.
- **AC3** : Identifier la faisabilité d'un projet.
- **AC4** : Évaluer l'impact des solutions.

### Évaluation

- Contrôle continu (exercices et rendus de TD/TP).
- Travaux intégrés à la **SAÉ** : mise en pratique de l'analyse des besoins et production des livrables associés (binôme ou trinôme, selon les groupes **SAÉ**).
- Devoir écrit (vérification des acquis théoriques), prévu lors de la **dernière séance** ou organisé en **amphi** dans le cas où la promotion est répartie en deux groupes.

## Cours 1 : Gestion de projet informatique (2h)

1. Contexte et objectifs d'un projet informatique.
2. Systèmes d'information, génie logiciel et activités associées.
3. Cycles de vie du logiciel : cascade, cycle en V, approche incrémentale.
4. Introduction aux méthodes agiles (XP, Scrum).
5. Processus unifié (UP).

### TD associés

- **TD1** : Réflexion sur la gestion de projet (projets tuteurés S1 et S3-S4).
- **TD2** : Gestion d'un projet logiciel complexe (organisation, contraintes).

Ces TD prolongent le cours en mettant les étudiants en situation de planifier un projet sous contraintes réelles (temps, binômes, ressources).

## Cours 2 : Recueil et analyse des besoins (3h)

1. Cahier des charges : rôle, structure, exemples.
2. Recueil des besoins métiers, identification des acteurs.
3. Exigences fonctionnelles et non-fonctionnelles.
4. Introduction aux cas d'utilisation.

### TD associés

- **TD3** : Lecture/écriture d'un cahier des charges (jeu de rôle client/développeur).

Ce TD met en pratique la formalisation des besoins en simulant une interaction client/développeur et la rédaction d'un document contractuel.

## Cours 3 : Cas d'utilisation et scénarios (3h)

1. Description des cas d'utilisation (formats, niveaux).
2. Diagrammes UML de cas d'utilisation.
3. Rédaction de scénarios et spécification par l'exemple.
4. Élaboration du document d'analyse des besoins.

### TD associés

- **TD4** : Cas d'utilisation (liste des exigences, niveaux, spécifications).
- **TD5** : Modélisation des processus métiers (problèmes simples et complexes).

Ces TD approfondissent la maîtrise des cas d'utilisation et la modélisation métier, en appliquant UML et en produisant des livrables intermédiaires.

## Cours 4 : Documentation et qualité logicielle (2h)

1. Typologie des documents (rapport, annexes techniques, supports).
2. Structure et bonnes pratiques de rédaction.
3. Importance de la qualité logicielle (discussion norme ISO).
4. Suivi de projet et validation des livrables.

## TD associés

- **TD6** : Rédaction du document d'analyse des besoins (intégration des TD3+TD4).

Ce TD final consolide les acquis : rédaction d'un document complet d'analyse des besoins en intégrant cahier des charges, cas d'utilisation et modèles UML.

## Répartition horaire indicative

- **CM** : 10h (4 séances principales).
- **TD/TP** : 13h ( $\approx$  6 séances).

## Bibliographie indicative

- A. Cockburn, *Writing Effective Use Cases*, Addison-Wesley.
- I. Sommerville, *Software Engineering*, Pearson.
- Référentiel BUT Informatique, R3.03 Analyse.