

Annexes — Guide de lecture

À propos de ces annexes

Ces 3 annexes approfondissent les concepts architecturaux fondamentaux vus en cours.

Elles ne sont **pas obligatoires** pour réussir le module,
mais elles vous aideront à :

- comprendre **pourquoi** ces principes existent
- éviter les erreurs classiques
- développer une **intuition architecturale**



Comment les utiliser

Option 1 : Lecture linéaire recommandée

Si c'est votre première lecture, suivez cet ordre :

1. `annexe_01_dependances_et_inversion.md`

→ Comprendre les dépendances et apprendre à les inverser

2. `annexe_02_decoupage_et_responsabilites.md`

→ Maîtriser couplage, cohésion et SRP



Comment les utiliser (suite)

3. annexe_03_tests_revelateur_architectural.md

→ Comprendre ce que les tests révèlent de votre architecture

Pourquoi cet ordre ?

Les concepts s'appuient les uns sur les autres :

- Les **dépendances** → doivent être inversées pour protéger le métier
- Le **découpage** (couplage/cohésion/SRP) → structure le code correctement



Comment les utiliser (suite 2)

- Les **tests** → révèlent la qualité du découpage et des dépendances

Option 2 : Lecture par besoin

Vous rencontrez un problème spécifique ? Consultez directement :

Problème rencontré	Annexe à consulter
"Mon test est difficile à écrire"	Annexe 3 (Tests) puis Annexe 1 (Dépendances)
"Une classe fait trop de choses"	Annexe 2 (Découpage - partie SRP)
"Mon code métier dépend de la BDD"	Annexe 1 (Inversion)

Option 2 : Lecture par besoin (suite)

Problème rencontré	Annexe à consulter
"Un changement casse plein de trucs"	Annexe 2 (Découpage - partie Couplage)
"Je ne sais pas où mettre ce code"	Annexe 2 (Découpage - partie SRP et Cohésion)
"Mes tests sont lents"	Annexe 3 (Tests) puis Annexe 1 (Inversion)



Interconnexions entre les annexes

Ces concepts ne sont pas isolés, ils forment un **système cohérent** :

Annexe 1 : Dépendances + Inversion

↓

Annexe 2 : Découpage (Couplage, Cohésion, SRP)

↓

Annexe 3 : Tests (révèlent tout)



Interconnexions entre les annexes (suite)

Message clé :

Chaque principe renforce les autres.

Une bonne maîtrise vient de leur **compréhension globale**.



Comment lire ces annexes

Ce qu'elles ne sont pas

- **✗** Des règles rigides à appliquer mécaniquement
- **✗** Des recettes toutes faites
- **✗** Une liste de bonnes pratiques à mémoriser



Comment lire ces annexes (suite)

Ce qu'elles sont

- Des questions à se poser pendant la conception
- Des signaux d'alarme à reconnaître
- Des exercices mentaux pour développer l'intuition



Conseils de lecture

1. Prenez votre temps

Ces concepts demandent de la réflexion.

Mieux vaut lire une annexe lentement que tout survoler rapidement.



Conseils de lecture (suite)

2. Reliez à votre code

Après chaque annexe, regardez votre projet `ticketing` :

- Identifiez des exemples concrets
- Repérez les points d'amélioration
- Testez les "exercices mentaux" sur votre code



Conseils de lecture (suite 2)

3. Revenez-y plus tard

Ces annexes prennent du sens **avec l'expérience**.

Relisez-les après avoir codé les TD2, TD3, TD4 :
vous y verrez de nouvelles choses.

4. Discutez-en

Les concepts architecturaux se comprennent mieux :

- en équipe
- en confrontant les points de vue
- en débattant sur des cas concrets



Pièges à éviter

Ne pas tomber dans le dogmatisme

Ces principes sont des **guides**, pas des lois absolues.

Il peut y avoir des exceptions,
mais elles doivent être **justifiées et conscientes**.



Pièges à éviter (suite)

Ne pas chercher la perfection immédiate

Une bonne architecture se construit **progressivement**.

L'important est de :

- reconnaître les problèmes
- comprendre pourquoi ils existent
- savoir comment les corriger



Utilisation pendant les TD

Pendant TD1 (Domain)

Concentrez-vous sur :

- Annexe 2 (SRP) : Une classe = une responsabilité
- Annexe 2 (Cohésion) : Règles métier groupées logiquement

Pendant TD2 (Use Cases + Ports)

Ajoutez :

- Annexe 1 (Dépendances) : Le domaine ne dépend de rien
- Annexe 1 (Inversion) : Les ports définissent les besoins



Utilisation pendant les TD (suite)

Pendant TD3 (SQLite)

Approfondissez :

- **Annexe 2 (Couplage)** : Le métier reste indépendant de la BDD
- **Annexe 3 (Tests)** : Le domaine se teste sans infrastructure

Pendant TD4 (API REST)

Consolidez :

- **Toutes les annexes** travaillent ensemble
- **Annexe 3 (Tests)** : Pyramide équilibrée (unit, integ, e2e)



Comment savoir si vous avez compris

Vous avez compris quand vous pouvez :

- 1. Expliquer avec vos mots pourquoi un principe existe**
- 2. Reconnaître quand il est violé dans du code**
- 3. Justifier vos choix de conception avec ces concepts**
- 4. Débattre de cas limites sans réponses toutes faites**

Lien avec le cours

Ces annexes **complètent** le cours magistral,
elles ne le remplacent pas.

- Le CM donne la structure globale
- Les annexes approfondissent les mécanismes
- Les TD permettent de pratiquer
- Les retours sur votre code ancrent la compréhension



En résumé

Structure des annexes :

- **Annexe 1** : Dépendances + Inversion (comprendre et maîtriser)
- **Annexe 2** : Découpage (Couplage + Cohésion + SRP)
- **Annexe 3** : Tests comme révélateur architectural

Temps estimé :

30-40 minutes par annexe (lecture attentive)



En résumé (suite)

Meilleur moment :

- Après le CM1 (vue d'ensemble)
- Pendant les TD (application concrète)
- Avant le QCM (consolidation)

Signe de réussite :

Vous commencez à vous poser ces questions **naturellement** pendant que vous codez.

Bonne lecture ! 