MÉTHODE SAKITI · ARCHITECTE DU SUCCÈS NUMÉRIQUE

CURSUS 4 · AXE A — DIRECTION TECHNIQUE

DU TEST À L'ARCHITECTURE

Du testeur à l'architecte : trois jalons, aucune passerelle,
la voie de l'expertise pure — le sommet technique sans le management

28 → 31 → 40 · F2 → F2 → F2

MONOGRAPHIE APPROFONDIE · CURSUS & VOIE D'EXPERTISE

ANCRAGE UEMOA · GÉNIE LOGICIEL · QUALITÉ

SOMMAIRE

01 Le cursus en un coup d'œil

02 Pourquoi cette voie — le contexte UEMOA

03 Jalon 1 · Ingénieur de Test — QA (28)

04 Jalon 2 · Ingénieur Qualité Logicielle (31)

05 Jalon 3 · Architecte Logiciel Global (40)

06 La voie de l'expertise — un cursus sans passerelle

07 Plan de mise en œuvre sur 30 mois

08 Étude de cas — la trajectoire d'Ibrahim

09 Activer ce cursus — lecture RH & DSI

10 Conclusion

A Annexe A — récapitulatif des trois jalons

B Annexe B — compétences & certifications

C Annexe C — glossaire & repères

D Annexe D — variantes, bifurcations & cursus voisins

E Annexe E — indicateurs de progression par jalon

F Annexe F — ressources de montée en compétences

G Annexe G — anti-modèles à éviter dans ce cursus

H Annexe H — les deux échelles de carrière

I Annexe I — questions fréquentes

J Annexe J — trois profils d'architectes

Cette monographie approfondit un unique cursus — le quatrième de la collection, et le seul 100 % intra-famille — jalon par jalon, de l'entrée au sommet d'expertise.

RÉSUMÉ EXÉCUTIF

Trois jalons prouvés, aucune passerelle : comment le test mène, en quelques années, au sommet de l'expertise logicielle.

Le cursus **« Du test à l'architecture »** conduit un ingénieur de test jusqu'au poste d'Architecte Logiciel Global. Il enchaîne trois trajectoires internes à la Famille 2 — Ingénieur de Test, Ingénieur Qualité Logicielle, puis Architecte — qui portent un même métier, la santé du logiciel, à une échelle croissante : du test du comportement à la conception du système.

Sa singularité fait son sujet : c'est le seul cursus de l'axe à rester entièrement dans une famille, sans passerelle ni changement de métier. Son sommet n'est pas un poste de direction mais un poste d'expert. C'est sur cette nature — la voie de l'expertise, et l'échelle de carrière technique qui doit la reconnaître — que cette monographie insiste le plus, car c'est elle qui décide si l'organisation retient, ou perd, ses meilleurs techniciens.

Pour les organisations de la zone UEMOA, ce cursus est un instrument de fiabilité et de rétention : former des architectes qui conçoivent testable et maintenable, et offrir aux experts une progression qui ne les contraint pas à devenir managers. Ce document en détaille chaque jalon, le plan de mise en œuvre sur trente mois, une étude de cas et la lecture RH & DSI pour l'activer.

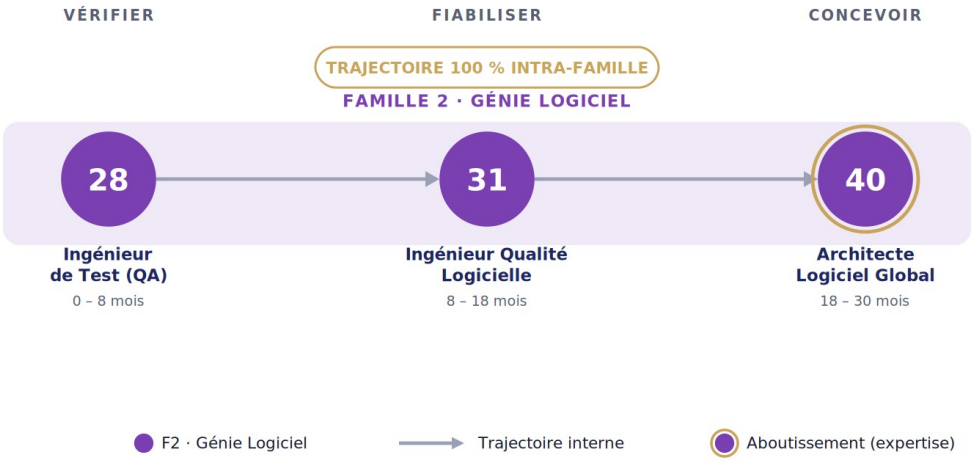
> **LE FIL CONDUCTEUR**
>
> Un principe traverse ce cursus : **la preuve, pas la promesse**. Chaque jalon se valide par un livrable déployable — une suite de tests qui arrête les régressions, des portes de qualité qui font baisser la dette, une architecture adoptée. C'est cette discipline qui sécurise la progression, du premier test jusqu'au sommet de l'expertise.

01 LE CURSUS EN UN COUP D'ŒIL

Du test à l'architecture : trois jalons, aucune passerelle, la voie de l'expertise pure — celle qui mène au sommet technique sans passer par le management.

Le cursus **« Du test à l'architecture »** occupe une place à part dans l'axe « Direction technique ». Il part d'un poste accessible — l'**Ingénieur de Test** — et conduit, en trois jalons prouvés, jusqu'au poste d'**Architecte Logiciel Global**. Sa singularité : il reste **entièrement dans la Famille 2 (Génie Logiciel)**. Aucun changement de famille, aucune passerelle. C'est le seul cursus de l'axe à tracer une voie d'expertise pure, sans détour par le management.

Les trois mouvements relèvent d'une même logique : approfondir, élargir le périmètre, prouver. Mais là où les autres cursus de l'axe culminent dans un rôle de direction, celui-ci culmine dans un rôle d'expert — l'architecte. Son enjeu n'est donc pas un franchissement vers le management, mais la montée en maturité d'un même métier : la qualité du logiciel, du test du comportement à la conception du système.



Les trois jalons

JALON	RÔLE (N°)	FAMILLE	NIVEAU	DURÉE
1	Ingénieur de Test — QA (28)	F2 · Génie Logiciel	Vérifier	0 – 8 mois
2	Ingénieur Qualité Logicielle (31)	F2 · Génie Logiciel	Fiabiliser	8 – 18 mois
3	Architecte Logiciel Global (40)	F2 · Génie Logiciel	Concevoir	18 – 30 mois

Profil d'entrée, profil de sortie

À l'entrée, un testeur qui vérifie que le logiciel fait ce qu'il doit. À la sortie, un architecte qui conçoit des systèmes entiers — élégants, mais aussi testables et maintenables, parce qu'il vient du test et de la qualité. Entre les deux, une transformation continue — **du vérifier au fiabiliser, puis au concevoir**.

> **UN CURSUS SANS PASSERELLE**
>
> Contrairement aux autres cursus de l'axe, celui-ci ne franchit aucune frontière de famille : il reste **100 % dans le Génie Logiciel**. Son sommet n'est pas un poste de direction mais un poste d'expert. C'est la voie de celles et ceux pour qui la maîtrise technique est une destination, et non une étape vers le management. La section 06 lui est entièrement consacrée.

02 POURQUOI CETTE VOIE — LE CONTEXTE UEMOA

Dans le paiement, un défaut logiciel se paie en argent et en confiance. La qualité et l'architecture deviennent des actifs stratégiques — et l'expert technique, un profil aussi rare que le dirigeant.

Dans la zone UEMOA, la couche applicative porte la valeur — et le risque. Un bug dans un service de paiement, c'est une transaction perdue, un client mécontent, parfois un incident réglementaire. La qualité du logiciel n'est pas un luxe d'ingénieur : c'est une condition de la confiance, donc de l'adoption. Or les profils capables de garantir cette qualité à l'échelle — du test à l'architecture — sont rares, et souvent mal valorisés faute de voie de carrière lisible.

Ce cursus répond à trois besoins structurels du marché régional. D'abord la **fiabilité par construction** : concevoir des systèmes testables et maintenables, plutôt que de corriger sans fin. Ensuite la **maîtrise de la dette** : garder un code sain pour évoluer vite et sûr, dans un secteur qui bouge. Enfin la **rétention des experts** : offrir aux meilleurs techniciens une perspective d'évolution qui ne les oblige pas à quitter la technique.

Trois moteurs de la demande

MOTEUR	CE QU'IL IMPOSE	RÔLE DU CURSUS
Fiabilité des services de paiement	Un logiciel testé, sûr, qui ne casse pas en production.	Forme le testeur, puis l'ingénieur qualité qui industrialise.
Vitesse d'évolution maîtrisée	Évoluer vite sans accumuler une dette ingérable.	Construit la qualité et l'architecture qui rendent le changement sûr.
Rétention des talents techniques	Des experts qu'on ne perd pas faute de perspective.	Trace une voie d'expertise lisible jusqu'à l'architecture.

Pour une direction des systèmes d'information de la zone, ce cursus est un instrument de fiabilité et de rétention. L'architecte issu du test et de la qualité conçoit des systèmes sains dès l'origine — et incarne une voie de carrière qui retient les experts que le marché s'arrache.

À qui ce cursus profite dans la zone

EMPLOYEUR	POSTE-CIBLE	POURQUOI CE CURSUS
FinTech / agrégateur de paiement	Architecte, ingénieur qualité senior	Fiabilité des API, vitesse maîtrisée, testabilité native.
Banque / établissement financier	Architecte SI, référent qualité	Maîtrise de la dette, conformité, robustesse applicative.
ESN / éditeur de logiciels	Architecte logiciel, SDET senior	Qualité produit, dette maîtrisée, standards techniques.
Grande entreprise / télécom	Architecte, expert qualité	Industrialisation de la qualité sur de gros systèmes.

FIL ROUGE — LA BANQUE WAARIPAY

Tout au long de ce document, nous illustrerons le cursus par le cas d'une banque régionale fictive mais représentative, **WaariPay**, dont la plateforme de paiement exige une qualité logicielle sans faille, et qui cherche à former — et à retenir — l'expert capable d'en concevoir l'architecture.

LE DIFFÉRENCIATEUR DU PROFIL

Un architecte issu de ce cursus possède un atout que peu partagent : il a **d'abord testé, puis fiabilisé le code, avant de concevoir le système**. Là où d'autres architectes dessinent des schémas élégants mais difficiles à tester, lui conçoit des systèmes **testables et maintenables par construction**. Dans un secteur où chaque défaut coûte cher, cette assise vaut de l'or.

03 JALON 1 · INGÉNIEUR DE TEST — QA (28)

Famille : F2 · Génie Logiciel **Mouvement :** Départ **Niveau :** Vérifier **Durée :** 0 – 8 mois

Le point d'entrée du cursus : garantir que le logiciel fait ce qu'il doit, du test manuel à l'automatisation industrialisée.

Le rôle

L'Ingénieur de Test conçoit et automatise les tests : stratégie de test, tests automatisés (unitaires, d'intégration, de bout en bout), tests d'API et de contrat, non-régression, intégration dans la chaîne d'intégration continue. Il est le garant que le logiciel fait ce qu'il doit — et qu'il continue de le faire après chaque changement. C'est un rôle de « vérifier » : il ne se contente pas de cliquer, il construit le filet de sécurité automatisé qui protège la production.

Beaucoup d'ingénieurs entrent dans la filière qualité par ce poste, souvent dans une fintech où la non-régression des paiements est vitale. Il ne demande pas une expertise rare, mais une discipline et une stratégie : on ne teste pas tout, on teste **ce qui compte**, au bon niveau de la pyramide des tests. C'est là que se prend une habitude décisive : penser la testabilité, qui irriguera tout le reste du cursus.

Missions principales

MISSION	EN QUOI ELLE CONSISTE
Définir la stratégie de test	Choisir quoi tester, à quel niveau (pyramide des tests).
Automatiser	Écrire des tests automatisés fiables et rapides.
Tester les API & contrats	Vérifier les interfaces et les intégrations entre services.
Garantir la non-régression	Empêcher qu'un changement ne casse l'existant.
Intégrer à la CI	Faire tourner les tests à chaque modification, en automatique.

Compétences à acquérir

- Stratégie de test : pyramide, niveaux, ce qu'il faut — ou non — automatiser.
- Automatisation : frameworks de test, tests stables et rapides.
- Tests d'API et de contrat : garantir les intégrations.
- TDD / BDD : écrire les tests comme moteur de conception.
- Intégration continue : exécution automatique des tests.
- Un langage de programmation pour écrire des tests robustes.

> **LA PREUVE ATTENDUE**
>
> Livrer une suite de tests automatisés intégrée à la chaîne d'intégration continue qui détecte les régressions avant la production, et faire baisser mesurablement le taux de défauts livrés. Le critère : un changement risqué est arrêté par les tests avant d'atteindre les clients, pas découvert après.

Une semaine-type

ACTIVITÉ	PART DU TEMPS
Conception et écriture de tests automatisés	≈ 45 %
Stratégie de test, analyse de couverture utile	≈ 20 %
Tests d'API, de contrat et de non-régression	≈ 20 %
Intégration CI, rapports de qualité, collaboration	≈ 15 %

Le mouvement

C'est le **départ** du cursus. Le professionnel capitalise un socle de qualité par le test — stratégie, automatisation, testabilité — qui servira à chacun des jalons suivants. La qualité de ce socle conditionne la suite : un testeur qui pense déjà « testabilité » et « ce qui compte » deviendra un bien meilleur ingénieur qualité que celui qui empile des tests fragiles.

> **ANCRAGE UEMOA — WAARIPAY**
>
> Chez WaariPay, ce poste protège les API de paiement contre les régressions : un test qui échoue, c'est un incident évité en production. C'est le vivier du cursus : la banque y repère les testeurs stratèges, qui automatisent ce qui compte, et les oriente vers la voie qualité.

Pièges à éviter

- Tester tout à la main : ne pas construire le filet automatisé qui protège la production.
- Viser 100 % de couverture : confondre une métrique de vanité avec la vraie qualité.
- Inverser la pyramide : tout tester en bout-en-bout, des suites lentes et fragiles.

04 JALON 2 · INGÉNIEUR QUALITÉ LOGICIELLE (31)

Famille : F2 · Génie Logiciel **Mouvement :** Trajectoire interne (F2 → F2) **Niveau :** Fiabiliser **Durée :** 8 – 18 mois

La qualité ne se teste pas, elle se construit : industrialiser la santé interne du code, de la dette à la sécurité.

Le rôle

L'Ingénieur Qualité Logicielle industrialise la qualité du code lui-même. Là où le testeur vérifie le comportement externe, lui garantit la santé interne : standards de code, revues efficaces, analyse statique, portes de qualité (quality gates) dans la chaîne CI/CD, dette technique mesurée et pilotée, sécurité du code (analyse SAST, dépendances), fiabilité des pipelines. C'est un rôle de « fiabiliser » : il rend la qualité visible, mesurable et opposable — il l'outille.

Ce jalon repose sur une conviction : **la qualité ne s'ajoute pas après coup, elle se construit en continu**. On ne peut pas « tester » la maintenabilité d'un code à la fin ; on l'obtient par des standards, des revues et des garde-fous automatisés tout au long. L'ingénieur qualité conçoit ces garde-fous — mais avec discernement, car une qualité punitive, faite de portes qui bloquent sans aider, est contre-productive.

Missions principales

MISSION	EN QUOI ELLE CONSISTE
Définir les standards	Établir les règles de lisibilité, de structure et de qualité du code.
Outiller la qualité	Analyse statique, linters, portes de qualité dans la CI/CD.
Piloter la dette technique	Mesurer, prioriser et rembourser la dette de façon continue.
Sécuriser le code	Analyse SAST et des dépendances (SCA), correction des vulnérabilités.
Animer les revues	Faire des revues de code un levier d'apprentissage, pas un contrôle.

Compétences à acquérir

- Qualité du code : lisibilité, maintenabilité, couplage, complexité.
- Analyse statique et portes de qualité bien calibrées.
- Gestion de la dette technique : mesure, priorisation, remboursement.
- Sécurité applicative : SAST, analyse des dépendances (SCA).
- Refactoring : améliorer le code sans en changer le comportement.
- Mentorat qualité : élever le niveau de l'équipe par les revues.

> **LA PREUVE ATTENDUE**
>
> Instaurer des portes de qualité et des standards qui font baisser mesurablement la dette et les défauts, sans bloquer la livraison. Le critère : la qualité interne du code s'améliore et devient pilotable, et les équipes adhèrent aux garde-fous au lieu de les contourner.

Une semaine-type

ACTIVITÉ	PART DU TEMPS
Standards, revues de code, mentorat qualité	≈ 35 %
Outillage : analyse statique, portes de qualité, pipelines	≈ 30 %
Pilotage de la dette technique et refactoring	≈ 20 %
Sécurité du code (SAST, dépendances)	≈ 15 %

Le mouvement

Trajectoire interne **F2 → F2** (28 → 31). Du test à la qualité : de la vérification du comportement externe à la garantie de la santé interne. Ce jalon est le préalable de l'architecture : on n'architecture bien que du code sain. L'ingénieur qui sait rendre un code maintenable acquiert le regard — couplage, dépendances, structure — qui fonde la pensée architecturale. On ne conçoit bien que ce que l'on sait garder sain.

> **ANCRAGE UEMOA — WAARIPAY**
>
> Chez WaariPay, la qualité du code des services de paiement conditionne la vitesse d'évolution : une dette maîtrisée permet d'ajouter un moyen de paiement sans tout casser. L'ingénieur qui l'outille devient le gardien de la capacité d'évolution de la plateforme.

Pièges à éviter

- Qualité punitive : des portes qui bloquent sans aider, que les équipes contournent.
- Mesurer pour mesurer : empiler des métriques sans décision derrière.
- Oublier l'humain : réduire la qualité à des outils en négligeant les revues.

05 JALON 3 · ARCHITECTE LOGICIEL GLOBAL (40)

Famille : F2 · Génie Logiciel **Mouvement :** Trajectoire interne (F2 → F2) · Aboutissement **Niveau :** Concevoir **Durée :** 18 – 30 mois

Le sommet du cursus et de l'expertise logicielle : passer du composant au système, et concevoir une architecture saine, évolutive et testable par construction.

Le rôle

L'Architecte Logiciel Global conçoit l'architecture d'ensemble du système : découpage en domaines métier (bounded contexts), choix des patterns, contrats d'API, modes de communication, cohérence transverse, maîtrise de la dette et standards de qualité. Il ne code plus seulement ses propres services : il définit les règles du jeu pour toutes les équipes, et arbitre les décisions structurantes. C'est le sommet de ce cursus — et l'aboutissement de l'expertise dans la Famille 2.

Ce qui distingue l'architecte issu de ce cursus, c'est son origine : venant du test et de la qualité, il conçoit des systèmes **testables et maintenables dès l'origine**. Il pense les frontières non seulement en termes de cohérence métier, mais aussi de testabilité et de découplage. Cette double compétence — concevoir et garder sain — est rare et précieuse.

Le principe directeur : le monolithe modulaire

Un principe guide l'architecte de ce cursus : **déployer comme un, concevoir comme plusieurs**. Le monolithe modulaire est le choix par défaut recommandé — un déploiement unique, mais une conception découpée en domaines clairs et faiblement couplés. Les microservices ne sont extraits qu'en cas de besoin réel et démontré. Cette discipline évite la complexité prématurée — et reste cohérente avec la culture qualité du cursus : un système simple est un système plus testable et plus sain.

Missions principales

MISSION	EN QUOI ELLE CONSISTE
Découper le système	Définir les domaines métier (bounded contexts) et leurs frontières.
Gouverner les API	Établir les contrats, le versionnement et les règles d'intégration.
Arbitrer l'architecture	Choisir patterns, communication, monolithe modulaire ou microservices.
Maîtriser la dette	Suivre la qualité et prioriser le remboursement de la dette à l'échelle.
Documenter les décisions	Tenir des ADR (architecture decision records) et des standards.
Mentorat technique	Faire monter les équipes, diffuser les bonnes pratiques.

Compétences à acquérir

- Architecture logicielle : DDD, bounded contexts, patterns d'intégration.
- Monolithe modulaire vs microservices : savoir choisir et justifier.
- Conception pour la testabilité : des frontières qui rendent le test simple.
- Qualité et dette à l'échelle du système entier.
- Sécurité by design : la sûreté pensée dès la conception.
- Leadership technique : ADR, standards, mentorat, coordination transverse.

> **LA PREUVE ATTENDUE**
>
> Livrer une architecture de référence réellement adoptée — par exemple le découpage d'un système de paiement en domaines clairs et testables — assortie d'ADR justifiant les choix et de standards suivis. Le critère : les équipes construisent sur cette architecture sans la contourner, et le système reste sain et testable à mesure qu'il grandit.

Une semaine-type

ACTIVITÉ	PART DU TEMPS
Conception, ADR, revues d'architecture	≈ 35 %
Coordination transverse entre équipes	≈ 20 %
Code de référence et preuves de concept testables	≈ 25 %
Mentorat technique et diffusion des standards	≈ 20 %

Le mouvement

Trajectoire interne **F2 → F2** (31 → 40), et **aboutissement du cursus**. Contrairement aux autres cursus de l'axe, ce sommet n'est pas un rôle de direction mais un rôle d'expert. L'architecte est au faîte de l'expertise technique de sa famille : il conçoit, il arbitre, il fait grandir — sans pour autant manager au sens hiérarchique. C'est une destination en soi, abordée en détail à la section suivante.

> **ANCRAGE UEMOA — WAARIPAY**
>
> L'architecte de WaariPay, venu du test et de la qualité, a découpé la plateforme de paiement en six domaines métier dont la testabilité est native. Il en est le référent technique — et l'organisation a tout intérêt à le retenir comme expert, sans le forcer à devenir manager.

Pièges à éviter

- Microservices prématurés : distribuer avant de maîtriser les frontières.
- Architecture « tour d'ivoire » : concevoir loin du code et de la testabilité.
- Oublier la testabilité : concevoir un système élégant mais impossible à tester — paradoxe fatal pour ce profil.

06 LA VOIE DE L'EXPERTISE — UN CURSUS SANS PASSERELLE

Le seul cursus de l'axe qui ne franchit aucune frontière : ce que signifie monter en expertise plutôt que vers le management, et pourquoi c'est une destination à part entière.

Les trois autres cursus de l'axe « Direction technique » partagent une mécanique : des trajectoires internes qui bâtissent une expertise, puis une passerelle qui la transforme en leadership. Celui-ci fait exception. Il reste **entièrement dans la Famille 2**, de bout en bout. Son sommet n'est pas un poste de manager mais un poste d'expert. Cette singularité n'est pas un manque : c'est le sujet même du cursus.

Le fil unique : un même métier à trois échelles

Test, qualité et architecture ne sont pas trois métiers différents : ce sont trois échelles d'un même métier — la santé du logiciel. Le testeur vérifie le comportement externe ; l'ingénieur qualité garantit la santé interne du code ; l'architecte conçoit la structure saine du système entier. Chaque jalon subsume et élargit le précédent, sans jamais le renier.

JALON	QUESTION QU'IL POSE	PÉRIMÈTRE
Test (28)	Le logiciel fait-il ce qu'il doit ?	Le comportement observable
Qualité (31)	Le code est-il sain et maintenable ?	La santé interne du code
Architecture (40)	Le système est-il bien structuré et évolutif ?	La structure du système entier

C'est pourquoi l'architecte issu de ce cursus est particulier : il porte les trois échelles à la fois. Il conçoit des systèmes parce qu'il sait les garder sains, et il les garde sains parce qu'il sait les tester. Cette continuité est sa force — et un atout que ne possède pas l'architecte venu directement du développement.

Ce qui grandit, ce qui se conserve

DIMENSION	AU FIL DES JALONS	STATUT
Le périmètre	Du composant au système entier	Grandit
L'horizon	Du comportement immédiat à l'évolution longue	Grandit
L'influence	De son propre code à toutes les équipes	Grandit
L'exigence de qualité	Présente du premier au dernier jalon	Se conserve et s'amplifie

L'expertise est une destination, pas une consolation

Dans beaucoup d'organisations, la seule manière de progresser est de devenir manager. C'est une erreur coûteuse. Promouvoir un excellent architecte à un poste de management qu'il ne désire pas, c'est **perdre un excellent expert et risquer de gagner un manager médiocre** — une variante du fameux principe de Peter. La voie de l'expertise existe précisément pour éviter ce piège : elle offre une progression — en périmètre, en influence, en reconnaissance — sans quitter la technique.

Dans les organisations matures, cette voie est formalisée par une échelle de carrière technique parallèle à l'échelle de management : ingénieur, ingénieur senior, ingénieur principal (staff), architecte. À chaque niveau correspond une influence croissante — et une reconnaissance, salariale et symbolique, équivalente à celle d'un manager de niveau comparable. C'est l'objet de l'annexe H.

> **LA DIFFÉRENCE ESSENTIELLE**
>
> Les autres cursus de l'axe répondent à la question « comment passer de l'ingénierie au management ? ». Celui-ci répond à une question tout aussi importante, et trop souvent négligée : **comment progresser sans cesser d'être ingénieur ?** Pour une organisation de la zone, savoir répondre aux deux est la clé pour retenir tous ses talents — pas seulement ceux qui veulent diriger.

Reconnaître l'impact de l'expert

L'objection la plus fréquente à l'échelle technique est qu'elle serait difficile à évaluer : un manager se mesure à son équipe, mais un expert, à quoi ? La réponse existe, et elle est concrète. L'impact d'un architecte se lit dans le système et dans les équipes qui s'en servent.

L'IMPACT DE L'EXPERT SE MESURE À...	EXEMPLE CONCRET
La portée de ses décisions	Une architecture adoptée par toutes les équipes, qui tient dans la durée.
La dette qu'il évite	Des choix qui épargnent des mois de correction futurs.
Les ingénieurs qu'il fait grandir	Une équipe qui code mieux parce qu'il a posé les standards.
Les risques qu'il désamorce	Une fausse bonne idée arrêtée avant qu'elle ne coûte cher.

Aucune de ces contributions ne se mesure à un effectif encadré — toutes se mesurent à un effet sur le système et sur les autres. C'est précisément ce qui rend l'expert aussi précieux qu'un manager, et tout aussi évaluable.

07 PLAN DE MISE EN ŒUVRE SUR 30 MOIS

Un jalon à la fois, une preuve à chaque étape : la cadence qui sécurise la progression du testeur à l'architecte.

Le cursus se déploie sur une trentaine de mois, du test à l'architecture. La règle est constante : chaque jalon mobilise plusieurs mois, le temps de produire une preuve déployable avant d'ouvrir le suivant. À la différence des autres cursus de l'axe, il n'y a pas de phase de passerelle : la progression est continue, à l'intérieur d'un même métier qui s'élargit.

PHASE	MOIS	OBJECTIF	PREUVE ATTENDUE
Jalon 1	M1 – M8	Maîtrise du test et de l'automatisation	Suite de tests en CI qui arrête les régressions
Jalon 2	M9 – M18	Industrialisation de la qualité du code	Portes de qualité + dette en baisse mesurable
Jalon 3	M19 – M30	Conception de l'architecture d'ensemble	Architecture de référence adoptée + ADR + standards

Le détail mois par mois

PÉRIODE	FOCUS	LIVRABLE DE LA PÉRIODE
M1 – M4	Stratégie de test, pyramide, automatisation	Premiers tests automatisés fiables en CI
M5 – M8	Tests d'API, de contrat, non-régression	Filet de non-régression sur un service critique
M9 – M12	Standards, revues, analyse statique	Portes de qualité en place sur un périmètre
M13 – M18	Dette technique, sécurité du code, refactoring	Dette mesurée et en baisse, vulnérabilités traitées
M19 – M24	DDD, découpage en domaines, contrats	Première architecture de domaine testable + ADR
M25 – M30	Architecture globale, dette, standards	Architecture de référence adoptée par les équipes

GOUVERNANCE DU PARCOURS

Chaque jalon est encadré par un **sponsor** (un dirigeant ou un architecte qui porte le cursus) et un **mentor** (un pair plus avancé). La preuve de fin de jalon est revue formellement avant d'ouvrir le suivant. Cette gouvernance légère transforme une intention de carrière en trajectoire pilotée.

08 ÉTUDE DE CAS — LA TRAJECTOIRE D'IBRAHIM

Comment WaariPay forme — et surtout retient — en trente mois l'architecte de sa plateforme, sans le contraindre à devenir manager.

Le point de départ

Ibrahim est ingénieur de test chez WaariPay : cinq ans d'expérience, une vraie rigueur, et l'art d'automatiser ce qui compte. Il aime profondément la technique et n'a aucune envie de manager. Mais il plafonne : dans son organisation, progresser semblait vouloir dire encadrer une équipe. La DSI, qui veut fiabiliser sa plateforme et retenir ses meilleurs techniciens, active le cursus « Du test à l'architecture » et lui ouvre une voie d'expertise.

La trajectoire, trimestre par trimestre

TRIMESTRE	JALON	PREUVE PRODUITE
T1 – T2	Ingénieur de Test (28)	Suite de tests en CI arrêtant les régressions des API de paiement.
T3 – T6	Ingénieur Qualité Logicielle (31)	Portes de qualité et dette en baisse sur les services critiques, vulnérabilités traitées.
T7 – T10	Architecte Logiciel Global (40)	Découpage de la plateforme en six domaines testables (monolithe modulaire), ADR à l'appui.

Le bilan

En une trentaine de mois, Ibrahim est passé d'un rôle de testeur à celui d'architecte de la plateforme — sans jamais devenir manager, et sans le vouloir. Chaque étape s'est validée par une preuve déployable. Surtout, la banque a retenu un expert qu'elle aurait pu perdre faute de perspective : reconnu, rémunéré et influent au niveau d'un manager, mais sur la voie technique.

Les trois décisions-clés de la DSI

La réussite d'Ibrahim ne tient pas au hasard mais à trois décisions de la DSI. **D'abord, reconnaître une voie d'expertise** : ne pas exiger qu'il devienne manager pour progresser. **Ensuite, garantir le temps et l'accompagnement** : un mentor architecte et un temps protégé à chaque jalon. **Enfin, valoriser à parité** : aligner reconnaissance et rémunération sur celles d'un manager de niveau comparable.

Le calcul économique

	STATU QUO (PAS DE VOIE EXPERTE)	CURSUS INTERNE (IBRAHIM)
Risque de départ	Élevé : l'expert part faute de perspective	Maîtrisé : une voie claire le retient
Coût	Remplacer un expert rare et coûteux	Accompagnement (mentor, temps, certifs) bien moindre
Qualité	Perte d'une expertise testabilité rare	Architecture saine et testable par construction
Effet d'entraînement	Aucun	Élève le niveau de toute l'équipe

Les obstacles rencontrés

La trajectoire d'Ibrahim n'a pas été sans heurts. Au jalon qualité, ses premières portes de qualité, trop strictes, ont braqué les équipes : il a dû les recalibrer pour qu'elles aident au lieu de bloquer. À l'arrivée à l'architecture, la tentation a été de **tout repenser** plutôt que de partir de l'existant ; son mentor l'a ramené à une approche incrémentale. Ces écueils, surmontés, font partie de l'apprentissage — et illustrent pourquoi le mentorat est décisif à chaque jalon.

CE QUE L'EXEMPLE DÉMONTRE

Le cursus n'est pas qu'un plan de formation : c'est un instrument de rétention. Il transforme un risque — perdre un expert faute de perspective — en trajectoire prouvée et valorisée, au bénéfice conjoint du collaborateur et de l'organisation.

09 ACTIVER CE CURSUS — LECTURE RH & DSI

De l'intention au pilotage : comment une direction RH ou DSI met ce cursus en mouvement — et installe la voie d'expertise qui retient les techniciens.

Quatre usages pour une direction

USAGE	CE QUE LE CURSUS PERMET
Rétention des experts	Offrir une progression sans contraindre à devenir manager.
Fiabilité par construction	Former des architectes qui conçoivent testable et maintenable.
Échelle de carrière technique	Installer une voie d'expertise parallèle à celle du management.
Maîtrise de la dette	Garder en interne la connaissance de la santé du code.

Installer la voie d'expertise

Le principal levier de ce cursus n'est pas la formation, mais la **reconnaissance d'une échelle de carrière technique**. Sans elle, les meilleurs testeurs et ingénieurs qualité finissent par viser le management — ou partir. Avec elle, ils peuvent progresser, gagner en influence et en rémunération, et rester ce qu'ils sont : des experts. Installer cette échelle, et la valoriser à parité avec celle du management, est la décision RH qui donne tout son sens au cursus.

Une feuille de route en 90 jours

À **J+30**, repérer les testeurs et ingénieurs qualité à fort potentiel (ceux qui pensent testabilité et structure) et formaliser l'échelle de carrière technique. À **J+60**, valider l'appétence des candidats pour la voie d'expertise et désigner un mentor architecte. À **J+90**, lancer le jalon 1 avec une preuve attendue claire et une reconnaissance (titre, rémunération) alignée sur la voie de management.

Facteurs de réussite & risques

FACTEUR DE RÉUSSITE	RISQUE SYMÉTRIQUE À ÉVITER
Échelle technique reconnue et valorisée à parité	Faire de l'expertise une voie de second rang
Preuve déployable validée à chaque jalon	Valider la progression sur la seule bonne volonté
Mentor architecte désigné	Lancer le parcours sans accompagnement ni temps dédié
Respect du choix de ne pas manager	Forcer l'expert vers un rôle de management non désiré

RECOMMANDATION

Documenter ce cursus, mais surtout installer l'échelle de carrière technique qui lui donne un sommet. Sur un marché régional où les architectes et experts qualité sont rares et chers, offrir une voie de progression sans management est l'un des leviers de rétention les plus puissants — et les moins coûteux.

10 CONCLUSION

Une voie d'expertise pure, à condition de la reconnaître comme telle : la preuve à chaque jalon, et une échelle de carrière qui la valorise.

« Du test à l'architecture » est le cursus de l'expertise. Il a pour lui la cohérence : un même métier — la santé du logiciel — porté à trois échelles, du test du comportement à la conception du système. Il a contre lui un angle mort fréquent des organisations : l'absence de voie de progression technique. Sans échelle de carrière qui reconnaisse l'expert à parité avec le manager, ce cursus mène à une impasse — et l'expert finit par partir.

Pour le professionnel, ce cursus offre une perspective précieuse : progresser, gagner en influence et atteindre le sommet technique — l'architecture — sans jamais cesser d'être ingénieur. Pour l'employeur, il offre un instrument de fiabilité et de rétention : former des architectes qui conçoivent sain et testable, et retenir les experts que le marché s'arrache. Pour la zone UEMOA, il répond à un besoin structurel et négligé : valoriser l'expertise technique autant que le management, pour ne plus perdre ses meilleurs techniciens.

Trois principes à retenir

- La preuve, pas la promesse : chaque jalon se valide par un livrable déployable.
- L'expertise est une destination : progresser sans cesser d'être ingénieur.
- La qualité se conserve : du test à l'architecture, un même fil, à une échelle croissante.

Pour aller plus loin

Ce cursus dialogue avec son voisin **« Du code à la direction »** (cursus 2), qui prolonge le rôle d'architecte vers le management et la direction du SI. Les deux partent du Génie Logiciel mais bifurquent à l'architecture : l'un vers l'expertise, l'autre vers le leadership. Offrir les deux, c'est permettre à chaque talent de choisir sa voie — et les retenir tous.

> **ENGAGEMENT**
>
> Du testeur à l'architecte, un chemin lisible et pilotable, prouvé à chaque étape — et couronné par une expertise reconnue. C'est ainsi que la qualité logicielle devient une voie de carrière à part entière, et que les organisations de la zone cessent de perdre leurs experts faute de perspective.

A ANNEXE A — RÉCAPITULATIF DES TROIS JALONS

JALON	RÔLE (N°)	FAMILLE	MOUVEMENT	DURÉE
1	Ingénieur de Test — QA (28)	F2	Départ	0 – 8 mois
2	Ingénieur Qualité Logicielle (31)	F2	Interne	8 – 18 mois
3	Architecte Logiciel Global (40)	F2	Interne · Aboutissement	18 – 30 mois

Enchaînement complet : 28 → 31 → 40 • **Familles :** F2 → F2 → F2 • **Aboutissement :** Architecte Logiciel Global (40), sommet d'expertise.

> **LA SINGULARITÉ DU CURSUS**
>
> Seul cursus 100 % intra-famille de l'axe « Direction technique » : aucune passerelle, aucun changement de famille. Sa progression est une montée en maturité, pas un franchissement.

B ANNEXE B — COMPÉTENCES & CERTIFICATIONS PAR JALON

JALON	COMPÉTENCES CLÉS	CERTIFICATION / PREUVE CIBLE
Ingénieur de Test (28)	Stratégie de test, automatisation, API, TDD/BDD, CI	Certifs test/QA, automatisation
Ingénieur Qualité (31)	Qualité du code, analyse statique, dette, SAST/SCA, refactoring	Qualité logicielle, sécurité du code
Architecte Logiciel (40)	DDD, monolithe modulaire, patterns, testabilité, ADR	Architecture adoptée, certifs architecture

NOTE SUR LES CERTIFICATIONS

Les certifications matérialisent une compétence mais ne la remplacent pas. Sur cette voie d'expertise, la preuve réelle est le système livré : tests qui arrêtent les régressions, dette qui baisse, architecture adoptée. La reconnaissance se mesure à l'influence technique, pas à un effectif encadré.

C ANNEXE C — GLOSSAIRE & REPÈRES

TERME	DÉFINITION
Pyramide des tests	Répartition recommandée : beaucoup de tests unitaires, peu de tests de bout en bout.
TDD / BDD	Développement piloté par les tests / par le comportement : écrire les tests d'abord.
Quality gate	Porte de qualité : seuil automatisé qu'un code doit franchir dans la CI/CD.
Dette technique	Coût futur des compromis de conception ou de qualité non remboursés.
SAST / SCA	Analyse de sécurité du code source / des dépendances logicielles.
Refactoring	Améliorer la structure interne du code sans changer son comportement.
DDD	Domain-Driven Design : conception guidée par les domaines métier (bounded contexts).
Monolithe modulaire	Déploiement unique mais conception découpée en domaines faiblement couplés.
ADR	Architecture Decision Record : trace écrite et justifiée d'une décision d'architecture.
Échelle technique	Voie de carrière d'expertise (senior, staff, principal, architecte), parallèle au management.

D ANNEXE D — VARIANTES, BIFURCATIONS & CURSUS VOISINS

« Du test à l'architecture » n'est pas l'unique destin d'un testeur. Plusieurs bifurcations s'ouvrent en chemin, et son sommet peut lui-même ouvrir vers d'autres voies.

Ce cursus décrit une voie d'expertise directe, mais le graphe des compétences en autorise d'autres. L'**Ingénieur Qualité (31)** comme l'**Architecte (40)** sont des **rôles-charnières** : depuis ces points, plusieurs trajectoires divergent. Connaître ces bifurcations aide une direction à orienter chaque talent.

Trois bifurcations fréquentes

BIFURCATION	POINT DE DÉPART	VERS	CURSUS VOISIN
Vers la sécurité applicative	Qualité (31)	DevSecOps, AppSec, analyse de vulnérabilités	C3 · Du CI/CD au DevSecOps
Vers la performance / SDET	Test (28)	Test de performance, ingénierie de test avancée	B · Test & performance
Vers le management	Architecte (40)	Engineering Manager, DSI Stratège	A2 · Du code à la direction

Le sommet comme bifurcation

La particularité de ce cursus est que son aboutissement — l'architecte — est lui-même une bifurcation. De là, on peut continuer sur la voie de l'expertise (architecte principal, expert distingué) ou basculer vers le management via le cursus « Du code à la direction ». Le cursus 4 et le cursus 2 partagent ce point de jonction : l'architecte choisit alors entre approfondir l'expertise ou franchir vers le leadership.

> **LECTURE**
> Présenter ces bifurcations dès l'entrée du cursus est un atout RH : le collaborateur comprend que l'expertise n'est pas une voie sans issue, mais un carrefour riche de possibles. Cela renforce l'engagement et réduit le risque de départ.

E ANNEXE E — INDICATEURS DE PROGRESSION PAR JALON

Comment objectiver l'avancement : pour chaque jalon, des indicateurs de maîtrise et leur cible, à revoir avant d'ouvrir le suivant.

JALON	INDICATEUR DE MAÎTRISE	CIBLE DE FIN DE JALON
1 · Test (28)	Efficacité du filet de tests	Régressions arrêtées en CI avant la production
1 · Test (28)	Taux de défauts livrés	Réduction mesurable des défauts en production
2 · Qualité (31)	Évolution de la dette technique	Dette mesurée et en baisse, sans blocage du delivery
2 · Qualité (31)	Sécurité du code	Vulnérabilités critiques traitées, dépendances suivies
3 · Architecte (40)	Adoption de l'architecture	Équipes construisant sur l'architecture, sans la contourner
3 · Architecte (40)	Testabilité & traçabilité	Système testable par construction, décisions couvertes par ADR

PRINCIPE

Un indicateur n'a de valeur que s'il déclenche une décision : ouvrir le jalon suivant, prolonger le jalon en cours, ou réorienter vers une bifurcation. La revue de fin de jalon est le moment où cette décision se prend, sponsor et mentor réunis.

F ANNEXE F — RESSOURCES DE MONTÉE EN COMPÉTENCES

Pour chaque jalon, les modalités d'apprentissage les plus efficaces : la pratique d'abord, la preuve ensuite, le mentorat tout du long.

JALON	MODALITÉS DE MONTÉE EN COMPÉTENCES
Ingénieur de Test (28)	Automatisation de suites réelles, ateliers TDD/BDD, intégration en CI, lectures sur la stratégie de test ; mentorat d'un ingénieur de test senior.
Ingénieur Qualité (31)	Mise en place de portes de qualité, exercices de refactoring, analyse statique et SAST, animation de revues ; mentorat d'un ingénieur qualité confirmé.
Architecte Logiciel (40)	Conception d'architectures testables de bout en bout, lecture de DDD et patterns, rédaction d'ADR, étude de systèmes de référence ; mentorat d'un architecte.

LA RÈGLE CONSTANTE

À tous les jalons, l'ordre est le même : **pratiquer, prouver, puis formaliser**. La formation accompagne une compétence construite par la pratique ; elle ne la remplace pas. Et à mesure que l'on monte vers l'architecture, la preuve se déplace du test isolé vers la santé du système entier.

G ANNEXE G — ANTI-MODÈLES À ÉVITER DANS CE CURSUS

Les erreurs récurrentes qui font dérailler le parcours — et l'antidote correspondant, à garder en tête à chaque jalon.

ANTI-MODÈLE	POURQUOI IL NUIT	ANTIDOTE
La couverture-vanité	Viser 100 % de couverture mesure l'effort, pas la qualité.	Tester ce qui compte, au bon niveau de la pyramide.
La pyramide inversée	Tout tester en bout-en-bout : suites lentes et fragiles.	Beaucoup d'unitaires, peu de bout-en-bout.
La qualité punitive	Des portes qui bloquent sans aider sont contournées.	Des garde-fous utiles, calibrés, qui élèvent l'équipe.
La tour d'ivoire	Concevoir loin du code et de la testabilité.	Rester au contact du code et valider par l'usage.
Le sommet bouché	Pas de voie d'expertise : l'expert part ou subit le management.	Une échelle technique reconnue, valorisée à parité.
L'architecture non testable	Concevoir un système élégant mais impossible à tester.	Penser la testabilité dès la conception.

À RETENIR

La plupart des échecs de ce cursus ne sont pas techniques : ils tiennent à une qualité mal pensée ou à l'absence de voie d'expertise. Nommer ces anti-modèles, c'est déjà commencer à les éviter.

H ANNEXE H — LES DEUX ÉCHELLES DE CARRIÈRE : EXPERT & MANAGER

Deux voies de progression parallèles et de valeur égale : comprendre leur équivalence est la condition pour retenir tous les talents.

Dans une organisation mature, progresser ne se résume pas à encadrer. Deux échelles coexistent, parallèles et de valeur équivalente : l'échelle de management et l'échelle technique. Ce cursus mène au sommet de la seconde. Les situer côte à côte clarifie ce qu'on offre à chacun — et évite de pousser les experts vers un management qu'ils ne désirent pas.

NIVEAU	ÉCHELLE DE MANAGEMENT	ÉCHELLE TECHNIQUE (CURSUS 4)
Confirmé	Chef d'équipe	Ingénieur senior
Avancé	Engineering Manager	Ingénieur principal (staff)
Élevé	Directeur d'ingénierie	Architecte
Sommet	DSI / CTO	Architecte distingué / expert

Le principe est l'**équivalence** : à chaque niveau, l'influence, la reconnaissance et la rémunération sont comparables d'une échelle à l'autre. Un architecte vaut un directeur d'ingénierie ; un expert distingué vaut un CTO. La différence n'est pas de valeur, mais de nature : l'un exerce son impact par les personnes qu'il dirige, l'autre par les décisions techniques qu'il porte et les ingénieurs qu'il fait grandir.

> **LECTURE**
> Sans échelle technique crédible, une organisation condamne ses meilleurs ingénieurs à un choix perdant : devenir manager malgré eux, ou partir. Installer cette échelle — et la valoriser à parité — est l'une des décisions RH les plus rentables pour une DSI de la zone.

I ANNEXE I — QUESTIONS FRÉQUENTES

Les interrogations les plus courantes des directions et des candidats au moment d'engager ce cursus.

Pourquoi ce cursus n'a-t-il pas de passerelle ?

Parce que c'est une voie d'expertise pure : elle reste entièrement dans le Génie Logiciel et ne mène pas au management, mais au sommet technique — l'architecture. L'absence de passerelle n'est pas un oubli, c'est la nature même du cursus : progresser sans changer de famille ni de métier, en montant en maturité.

Un architecte « vaut »-il vraiment un manager ?

Dans une organisation mature, oui, à niveau équivalent. L'architecte exerce son impact par les décisions techniques structurantes et par les ingénieurs qu'il fait grandir, là où le manager l'exerce par les personnes qu'il dirige. La valeur est comparable ; la nature diffère. C'est tout l'objet de l'échelle de carrière technique (annexe H).

Peut-on devenir architecte sans passer par la qualité ?

On le peut, mais ce cursus parie sur l'inverse — et c'est sa force. L'architecte qui vient du test et de la qualité conçoit des systèmes testables et maintenables par construction, là où d'autres dessinent des schémas difficiles à éprouver. La continuité test → qualité → architecture est précisément ce qui distingue ce profil.

Et si l'architecte veut finalement manager ?

C'est tout à fait possible : l'architecture est un point de jonction. De là, on peut poursuivre l'expertise ou basculer vers le management via le cursus « Du code à la direction ». Le cursus 4 ne ferme aucune porte ; il offre une voie complète qui peut, plus tard, bifurquer vers le leadership.

Combien de temps faut-il réellement ?

Une trentaine de mois pour aller du test à l'architecture. Le rythme dépend des opportunités et du candidat : le cursus trace le chemin et ses preuves, il ne garantit pas le calendrier. Mieux vaut un jalon tenu, validé par un livrable réel, que deux jalons entamés.

Ce cursus convient-il aux petites structures ?

Oui, en adaptant l'échelle. Dans une petite fintech, une même personne peut cumuler test, qualité et amorce d'architecture. L'essentiel n'est pas la taille mais la reconnaissance de la voie d'expertise : même à petite échelle, offrir une perspective technique retient les talents.

> **EN RÉSUMÉ**
>
> Ce cursus n'est ni automatique ni universel : c'est la voie de celles et ceux pour qui la technique est une vocation, à activer avec discernement — et à couronner par une échelle de carrière qui la reconnaisse.

J ANNEXE J — TROIS PROFILS D'ARCHITECTES

On peut devenir architecte par plusieurs chemins. Chacun forge une force distinctive — et un angle mort à surveiller. Comprendre d'où vient un architecte éclaire ce qu'on peut en attendre.

L'architecture logicielle n'a pas une seule porte d'entrée. Un architecte peut venir du test et de la qualité — c'est la voie de ce cursus —, du développement produit, ou de l'infrastructure. Aucun de ces chemins n'est meilleur dans l'absolu : chacun apporte une sensibilité propre. Mais ils ne produisent pas le même architecte.

PROFIL D'ORIGINE	FORCE DISTINCTIVE	ANGLE MORT À SURVEILLER
Venu du test & de la qualité (ce cursus)	Conçoit des systèmes testables et maintenables par construction.	Peut sous-estimer la performance brute et la scalabilité extrême.
Venu du développement produit	Pragmatisme, proximité du métier, sens du delivery.	Peut négliger la testabilité et laisser filer la dette.
Venu de l'infrastructure	Pense résilience, déploiement, exploitation, coût.	Peut sous-estimer la qualité interne et la lisibilité du code.

La force du profil de ce cursus est rare et recherchée : l'architecte **testability-native**, qui conçoit d'emblée des systèmes qu'on sait éprouver et faire évoluer. Dans un secteur où chaque défaut se paie, cette sensibilité change tout. L'idéal, pour une organisation, est de réunir plusieurs profils complémentaires — mais celui que forme ce cursus est souvent le plus difficile à recruter, donc le plus précieux à former en interne.

> **LECTURE**
> Connaître l'origine d'un architecte aide à composer une équipe d'architecture équilibrée : associer un profil qualité, un profil produit et un profil infrastructure couvre les angles morts de chacun. Ce cursus produit le premier — le plus rare sur le marché de la zone.

*Ce document constitue l'approfondissement du **Cursus 4 · Du test à l'architecture**, quatrième des treize cursus de la collection « Les cursus intra & inter familles » de la Méthode SAKITI.*