

**UNIVERSITÉ DE LORRAINE
IUT NANCY-CHARLEMAGNE
DÉPARTEMENT INFORMATIQUE**



RAPPORT DE STAGE

Découverte de la transformation digitale et des métiers de l'informatique bancaire

BANK OF AFRICA

Présenté par
Formation
Année universitaire
Période du stage
Tuteur en entreprise

Zayd BENZAKOUR
BUT Informatique - 1re année
2025-2026
Du 22 juin au 17 juillet 2026
Soumia Fares

Déclaration de respect des droits d'auteur

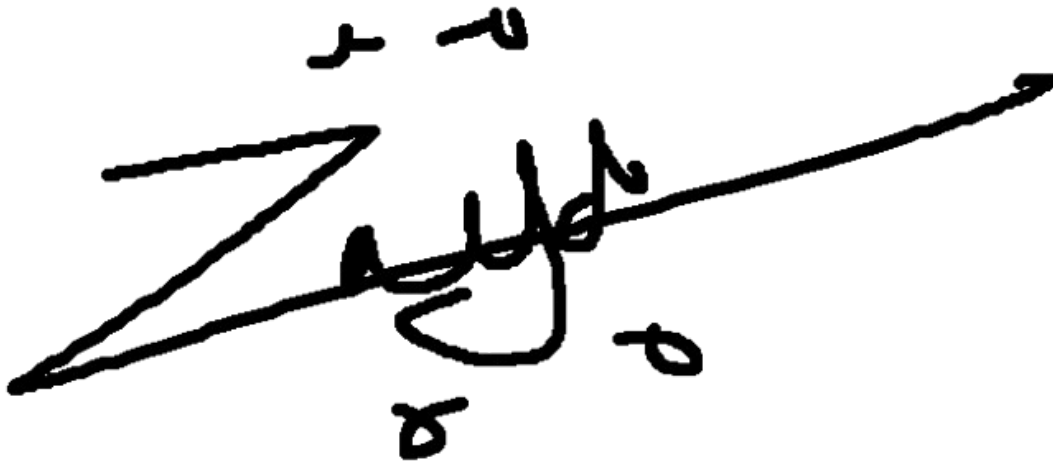
Par la présente, je déclare être l'auteur de ce rapport et assure qu'aucune autre ressource que celles indiquées n'a été utilisée pour sa réalisation. Tout emprunt, citation ou référence à un document publié ou inédit est signalé comme tel.

L'usage d'un outil doté d'intelligence artificielle pour m'aider à structurer, reformuler et corriger ce document est mentionné et reste sous ma responsabilité. J'ai relu l'ensemble du contenu et je demeure responsable de l'exactitude des informations présentées.

Fait à : Casablanca, Maroc

Le : 18/07/26

Signature :

A handwritten signature in black ink, appearing to read 'Zayd Benzakour', with a long horizontal stroke extending to the right.

Remerciements

Je tiens à remercier l'ensemble des collaborateurs de BANK OF AFRICA qui m'ont accueilli pendant la période du 22 juin au 17 juillet 2026. Malgré leurs responsabilités et leurs contraintes professionnelles, plusieurs d'entre eux ont pris le temps de me présenter leur métier, leurs méthodes de travail et les projets auxquels ils participent.

Je remercie particulièrement Oussama El Hachimi pour la présentation de la plateforme de crédit à la consommation et des technologies utilisées ; Zineb Khadraoui pour la découverte du métier de QA testeur et de JMeter ; Soumia Fares pour m'avoir permis d'assister à une réunion ; Youssef pour la présentation du design produit et de Figma ; Mehdi pour ses explications sur la gestion de projet et le projet de néo banque ; Youssef pour la découverte des traitements batch et du développement back-end ; Ayoub pour la présentation du développement iOS ; ainsi que le collaborateur du domaine de la data qui m'a expliqué les principaux volets du traitement et de la valorisation des données dans un environnement bancaire.

Je remercie également mon tuteur de stage, Soumia Fares, pour son accompagnement

Résumé

Ce stage d'immersion au sein de BANK OF AFRICA avait pour objectif de découvrir le fonctionnement d'un département informatique et digital dans le secteur bancaire. Réalisé du 22 juin au 17 juillet 2026, il m'a permis d'observer l'organisation en squads, la méthode Agile et les principales étapes du cycle de vie d'un projet numérique, depuis l'analyse du besoin jusqu'au déploiement en production.

Au cours de la période, j'ai échangé avec plusieurs professionnels intervenant dans le développement web et mobile, les tests logiciels, le design UX/UI, la gestion de projet, les traitements batch et la data. J'ai notamment découvert les environnements de développement, de recette et de production, les tests de performance avec JMeter, la conception de maquettes avec Figma, les enjeux du time to market, ainsi que l'ingestion et la transformation de données.

Cette expérience m'a montré que la réussite d'un projet informatique bancaire ne repose pas uniquement sur le développement. Elle dépend aussi de la qualité, de la coordination entre les métiers, de la sécurité, de la réglementation et de la compréhension des besoins utilisateurs.

Mots clés : informatique bancaire, transformation digitale, méthode Agile, développement, assurance qualité, UX/UI, data, gestion de projet.

Abstract

This observation internship at BANK OF AFRICA aimed to provide an overview of how an IT and digital department operates in the banking sector. Conducted from 22 June to 17 July 2026, it allowed me to discover squad-based organisation, Agile practices and the main stages of a digital project, from business needs to production deployment.

During the internship, I met professionals working in web and mobile development, software testing, UX/UI design, project management, batch processing and data. I discovered development, testing and production environments, performance testing with JMeter, interface prototyping with Figma, time-to-market issues, and data ingestion and transformation.

This experience showed me that successful banking IT projects rely not only on technical development, but also on quality assurance, coordination, security, regulation and a clear understanding of user needs.

Keywords: banking IT, digital transformation, Agile, software development, quality assurance, UX/UI, data, project management.

Sommaire

1. Introduction
 2. Présentation du contexte professionnel
 - 2.1 Présentation générale de BANK OF AFRICA
 - 2.2 Organisation du département Data et Digital
 - 2.3 Objectifs et déroulement général du stage
 3. Découverte du fonctionnement d'un projet informatique bancaire
 - 3.1 Organisation Agile et travail en squads
 - 3.2 Cycle de vie d'une fonctionnalité
 - 3.3 Environnements de développement, recette et production
 4. Développement d'applications et traitements informatiques
 - 4.1 Plateforme de crédit à la consommation
 - 4.2 Technologies web et outils de développement
 - 4.3 Développement back-end et traitements batch
 - 4.4 Développement mobile iOS
 5. Qualité logicielle et conception centrée utilisateur
 - 5.1 Rôle du QA testeur
 - 5.2 Types de tests et découverte de JMeter
 - 5.3 Design UX/UI et méthode Design Thinking
 6. Gestion de projet et transformation digitale
 - 6.1 Rôle du chef de projet
 - 6.2 Projet de néo banque et time to market
 - 6.3 Choix d'un éditeur et contraintes organisationnelles
 7. La data dans un environnement bancaire
 - 7.1 Ingestion, traitement et restitution
 - 7.2 Reporting réglementaire
 - 7.3 Valorisation des données
 8. Analyse globale et apports du stage
 - 8.1 Complémentarité des métiers
 - 8.2 Contraintes du secteur bancaire
 - 8.3 Liens avec le BUT Informatique
 - 8.4 Apports pour mon projet professionnel
 9. Conclusion
- Glossaire
- Références bibliographiques et webographiques
- Annexes

1. Introduction

Le secteur bancaire connaît depuis plusieurs années une transformation importante liée au développement des services numériques. Les clients souhaitent consulter leurs comptes, effectuer des opérations, simuler un crédit ou accéder à un service sans nécessairement se rendre en agence. Pour répondre à ces attentes, les banques doivent faire évoluer leurs systèmes d'information tout en respectant des exigences élevées de sécurité, de fiabilité et de conformité.

C'est dans ce contexte que j'ai effectué un stage d'immersion au sein de BANK OF AFRICA, du 22 juin au 17 juillet 2026. En tant qu'étudiant en première année de BUT Informatique à l'IUT Nancy-Charlemagne, je souhaitais confronter les notions étudiées en formation à un environnement professionnel réel. Mon objectif était notamment de comprendre comment sont organisés les projets informatiques dans une grande banque, quels métiers interviennent dans leur réalisation et comment les équipes collaborent entre elles.

La durée du stage m'a permis d'alterner des temps d'observation, des échanges avec différents professionnels, la consultation de documents internes autorisés, des démonstrations d'outils et un travail personnel de prise de notes et de rédaction. Les disponibilités des collaborateurs dépendaient naturellement de leurs responsabilités et de leurs contraintes opérationnelles. Les rencontres ont donc été réparties sur l'ensemble de la période plutôt que concentrées dans un déroulement strictement journalier.

Ce stage n'avait pas pour objectif de livrer une application complète. Il s'agissait principalement d'un stage de découverte du fonctionnement d'un service informatique bancaire. Cette organisation m'a permis d'explorer plusieurs domaines complémentaires : développement web et mobile, assurance qualité, design UX/UI, gestion de projet, traitements automatisés et data.

Dans ce rapport, je présenterai d'abord le contexte professionnel et l'organisation du département. J'expliquerai ensuite le fonctionnement global d'un projet informatique bancaire, avant de développer les principaux domaines observés. Enfin, je proposerai une analyse des apports du stage, de ses liens avec ma formation et de son influence sur mon projet professionnel.

2. Présentation du contexte professionnel

2.1 Présentation générale de BANK OF AFRICA

BANK OF AFRICA est un groupe bancaire d'origine marocaine, historiquement connu sous le nom de BMCE Bank. Il propose des services destinés aux particuliers, aux professionnels et aux entreprises : comptes bancaires, moyens de paiement, crédits, services à distance et accompagnement financier.

La transformation digitale occupe une place importante dans l'évolution de la banque. Elle vise à améliorer l'accessibilité des services, à simplifier les parcours clients et à centraliser certaines fonctionnalités qui étaient auparavant réparties entre plusieurs plateformes.

Durant le stage, j'ai constaté que cette transformation ne consiste pas uniquement à développer de nouvelles interfaces. Elle implique également une évolution des processus métiers, des systèmes techniques, des méthodes de travail et des dispositifs de contrôle. Les projets doivent tenir compte

des besoins des utilisateurs, de la sécurité, des contraintes réglementaires et des dépendances entre différentes entités de l'organisation.

Le secteur bancaire présente une sensibilité particulière : les applications manipulent des informations personnelles et des opérations financières. La fiabilité, la confidentialité et la traçabilité occupent donc une place centrale dans toutes les étapes d'un projet.

Certaines informations observées pendant le stage concernent l'organisation interne de la banque. Le présent rapport reste volontairement général et ne contient aucune donnée client, aucun secret technique et aucun élément confidentiel.

2.2 Organisation du département Data et Digital

Le stage s'est déroulé au contact d'un département orienté Data et Digital. On m'a présenté sa structure, la répartition des responsabilités et la manière dont les équipes travaillent autour des projets.

Le fonctionnement repose notamment sur des squads, c'est-à-dire des équipes pluridisciplinaires réunies autour d'un produit ou d'un ensemble de fonctionnalités. Une squad peut inclure des développeurs, des testeurs, des designers, des profils data, un Product Owner, un Proxy Product Owner, un Scrum Master ou encore un chef de projet.

Cette organisation permet de réunir dans une même équipe les compétences nécessaires pour comprendre un besoin, concevoir une solution, la développer, la tester puis la faire évoluer. Chaque membre dispose d'une responsabilité principale, mais doit aussi comprendre les contraintes des autres métiers afin de faciliter la collaboration.

J'ai également observé que les tâches ne sont pas isolées. Une modification technique peut avoir des conséquences sur l'expérience utilisateur, les tests, les données ou les contraintes de sécurité. La coordination entre les profils est donc indispensable.

2.3 Objectifs et déroulement général du stage

L'objectif principal du stage était de découvrir l'environnement informatique d'une grande entreprise bancaire et de mieux comprendre la diversité des métiers présents dans un projet numérique.

Pour cela, j'ai été amené à observer l'organisation du service, à échanger avec plusieurs professionnels et à assister à des démonstrations d'outils. Les thèmes abordés ont été répartis sur l'ensemble du stage selon les disponibilités des équipes.

Les principaux objectifs pédagogiques étaient de comprendre le cycle de vie d'un projet, d'identifier les rôles des différents acteurs, de découvrir des technologies utilisées en entreprise et de faire le lien entre les enseignements du BUT Informatique et les pratiques professionnelles.

La période a également comporté des temps de travail autonome. Je les ai utilisés pour organiser mes notes, rechercher la signification de certaines notions, préparer mes questions et rédiger progressivement le rapport. Cette méthode m'a permis de mieux exploiter les échanges réalisés avec les collaborateurs.

3. Découverte du fonctionnement d'un projet informatique bancaire

3.1 Organisation Agile et travail en squads

Les projets observés suivent une organisation inspirée des méthodes Agiles. Contrairement à une approche dans laquelle l'ensemble du projet serait défini une seule fois au début, l'Agile permet de progresser par étapes, de recueillir des retours et d'adapter les priorités.

Les besoins sont regroupés dans un backlog. Ils sont ensuite analysés, classés par priorité et souvent formulés sous forme de User Stories. Une User Story décrit un besoin du point de vue de l'utilisateur et précise la valeur attendue. Des critères d'acceptation permettent ensuite de vérifier objectivement que la fonctionnalité répond bien au besoin.

Le Product Owner porte la vision du produit et établit les priorités. Le Proxy Product Owner peut assurer un suivi plus opérationnel. Le Scrum Master facilite l'organisation de l'équipe et aide à résoudre les obstacles. Les développeurs réalisent les fonctionnalités, les QA testeurs les contrôlent et les designers travaillent sur l'expérience utilisateur.

Le travail est organisé en périodes courtes appelées sprints. À l'issue d'un sprint, l'équipe dispose d'un résultat partiel qui peut être présenté, testé et amélioré. Cette organisation permet de limiter le risque de découvrir trop tard qu'une solution ne correspond pas aux attentes.

3.2 Cycle de vie d'une fonctionnalité

Les échanges avec les professionnels m'ont permis de reconstituer les principales étapes suivies par une fonctionnalité.

Tout commence par l'identification d'un besoin métier ou utilisateur. Ce besoin est ensuite clarifié pendant une phase de cadrage. Il peut être complété par des études, des ateliers, des maquettes ou des comparaisons avec des solutions existantes.

Le besoin est ensuite traduit en éléments exploitables par l'équipe technique, notamment des User Stories et des tâches. Les développeurs réalisent la fonctionnalité, puis les testeurs vérifient son comportement. Lorsqu'une anomalie est détectée, elle est décrite afin de permettre sa reproduction et sa correction.

Après les premiers contrôles, la fonctionnalité est déployée dans un environnement de recette où elle peut être validée par les équipes métier, les responsables qualité ou le client interne. Ce n'est qu'après cette validation qu'elle peut être mise en production.

Ce processus m'a permis de comprendre qu'une fonctionnalité n'est pas considérée comme terminée uniquement parce que le code a été écrit. Elle doit aussi être testée, documentée, validée et intégrée de manière sûre dans le système existant.

3.3 Environnements de développement, recette et production

Trois environnements principaux m'ont été présentés : développement, recette et production.

L'environnement de développement est utilisé par les développeurs pour construire et tester leurs modifications. Il permet de travailler sans affecter directement les utilisateurs réels.

L'environnement de recette est destiné aux validations. Les équipes métier, les QA testeurs ou les responsables du produit peuvent vérifier que le comportement correspond aux attentes. Les scénarios d'utilisation et les critères d'acceptation y sont contrôlés avant toute diffusion.

L'environnement de production correspond à la version réellement utilisée. Les changements qui y sont déployés doivent donc avoir franchi les étapes précédentes.

Cette séparation limite les risques. Dans le secteur bancaire, une anomalie en production peut affecter des utilisateurs, des opérations ou des données sensibles. Les contrôles avant déploiement sont donc particulièrement importants.

4. Développement d'applications et traitements informatiques

4.1 Plateforme de crédit à la consommation

Oussama El Hachimi m'a présenté un projet lié à la plateforme de crédit à la consommation. L'objectif est de centraliser plusieurs services qui étaient auparavant répartis entre des sites distincts.

Cette centralisation vise à proposer une expérience plus cohérente. Au lieu de devoir identifier différents sites selon le service recherché, l'utilisateur peut accéder à un parcours unifié.

Le projet illustre une problématique fréquente dans les grandes organisations : les services ont parfois été développés à des périodes différentes et sur des plateformes séparées. La transformation digitale consiste alors non seulement à créer de nouvelles fonctionnalités, mais aussi à rapprocher et moderniser l'existant.

Cette présentation m'a permis de comprendre l'importance de l'intégration. Une nouvelle interface doit fonctionner avec les systèmes déjà présents, respecter les règles métier et maintenir un niveau élevé de sécurité.

4.2 Technologies web et outils de développement

La partie front-end de la plateforme repose sur React. Cette bibliothèque JavaScript permet de construire des interfaces sous forme de composants réutilisables. Tailwind CSS est utilisé pour la mise en forme, tandis que Material UI fournit des composants d'interface prêts à être intégrés.

Le back-end utilise Java et Spring Boot. Spring Boot facilite la création d'applications et de services web en Java. Cette technologie est cohérente avec les enseignements de programmation orientée objet reçus en BUT Informatique.

GitLab est utilisé pour la gestion du code source et le travail collaboratif. Il permet de conserver l'historique des modifications, de travailler sur différentes branches et de faciliter les revues de code.

Jenkins intervient dans l'intégration et le déploiement continu. Il peut automatiser certaines étapes comme la compilation, l'exécution de tests ou le déploiement vers un environnement.

Insomnia est utilisé pour tester les API. Cet outil permet d'envoyer des requêtes, de vérifier les réponses et de contrôler le fonctionnement des échanges entre le front-end et le back-end.

La découverte de ces outils m'a montré qu'un projet professionnel s'appuie sur un ensemble coordonné de technologies. Le code n'est qu'une partie de la chaîne ; il faut aussi gérer les versions, les tests, les déploiements et la communication entre les services.

4.3 Développement back-end et traitements batch

Un développeur principalement orienté back-end m'a présenté des traitements batch. Un batch est un processus automatisé qui exécute une suite d'opérations sans intervention manuelle directe.

L'exemple observé concernait des traitements de paiement, notamment liés à des vignettes automobiles. Les traitements étaient organisés à partir d'un fichier de configuration général et de fichiers de configuration propres à chaque batch.

Des dossiers nommés « In » et « Out » permettaient de distinguer les fichiers entrants des fichiers générés ou envoyés. Cette organisation facilite le suivi du parcours des données et la détection d'un éventuel problème.

FileZilla était utilisé pour accéder aux fichiers sur un environnement distant. GitLab servait à gérer le code, tandis que Jira était cité pour le suivi des tâches, des User Stories et des anomalies.

Avant ce stage, j'associais principalement le back-end aux API et aux bases de données. J'ai découvert qu'il comprend aussi des traitements planifiés, des échanges de fichiers, des configurations et des opérations métier automatisées.

Le développeur m'a également conseillé de m'intéresser à la gestion de projet. Même lorsqu'on exerce un métier très technique, il est nécessaire de comprendre le besoin, les priorités et la place de son travail dans l'ensemble du projet.

4.4 Développement mobile iOS

Ayoub, développeur iOS, m'a présenté son activité et les technologies qu'il utilise. Il travaille principalement avec Swift et SwiftUI.

Swift est un langage de programmation conçu pour les plateformes Apple. SwiftUI est un framework qui permet de construire les interfaces des applications de manière déclarative.

Il m'a montré deux applications : une application ludique composée de plusieurs mini-jeux et une application bancaire sur laquelle il avait occupé un rôle important en tant que développeur principal.

Cette démonstration m'a permis de comparer le développement mobile au développement back-end. Dans une application mobile, l'interface, l'ergonomie, la fluidité et le comportement visible occupent une place centrale.

Les deux domaines conservent toutefois des exigences communes : qualité du code, organisation du projet, tests, gestion des versions et collaboration avec les autres équipes.

5. Qualité logicielle et conception centrée utilisateur

5.1 Rôle du QA testeur

Zineb Khadraoui m'a présenté le métier de QA testeur. Le terme QA signifie Quality Assurance, ou assurance qualité.

Le QA testeur ne se contente pas de vérifier si une fonctionnalité s'exécute. Il doit contrôler que son comportement correspond au besoin, que les erreurs sont correctement gérées et que l'application reste stable dans différents scénarios.

Lorsqu'une anomalie est détectée, elle doit être décrite avec précision : contexte, étapes permettant de la reproduire, résultat attendu, résultat obtenu et éventuelles preuves. Ce rapport permet aux développeurs de comprendre le problème et de le corriger efficacement.

La communication entre les testeurs et les développeurs est donc essentielle. Un problème mal décrit peut être difficile à reproduire, tandis qu'un rapport clair réduit le temps nécessaire à la correction.

5.2 Types de tests et découverte de JMeter

Plusieurs catégories de tests m'ont été présentées.

Les tests fonctionnels vérifient qu'une fonctionnalité correspond au besoin attendu. Ils se placent du point de vue de l'utilisateur ou du métier.

Les tests automatisés permettent de répéter certains contrôles de manière rapide et régulière. Ils sont particulièrement utiles lorsqu'une application évolue fréquemment, car ils aident à détecter les régressions.

Les tests de montée en charge étudient le comportement d'une application lorsqu'elle reçoit un volume important de requêtes ou d'utilisateurs. Ils permettent d'observer les temps de réponse, les erreurs et les limites du système.

J'ai assisté à une démonstration de JMeter. Cet outil permet de construire des scénarios de test, de simuler des requêtes et d'analyser les résultats obtenus.

Même si je n'ai pas utilisé JMeter de manière autonome, cette démonstration m'a donné une première compréhension des tests de performance et de leur importance avant la mise en production.

5.3 Design UX/UI et méthode Design Thinking

J'ai également découvert la partie design avec un professionnel utilisant Figma. Cette présentation m'a permis de comprendre que le design d'une application ne se limite pas à son apparence.

La méthode Design Thinking est centrée sur l'utilisateur. Elle peut être résumée en cinq grandes étapes : comprendre les utilisateurs, définir le problème, imaginer des solutions, créer des prototypes et les tester.

Les wireframes basse fidélité sont des représentations simples de la structure d'une interface. Ils permettent de tester rapidement une organisation sans consacrer trop de temps aux détails visuels.

Les maquettes haute fidélité se rapprochent davantage du résultat final. Elles intègrent les couleurs, la typographie, les composants et les interactions principales.

Plusieurs notions complémentaires m'ont été présentées : benchmark, persona, empathy map, story map, roadmap et MVP. Le MVP, ou Minimum Viable Product, correspond à une première version utilisable contenant les fonctionnalités essentielles.

Figma facilite la création et le partage de maquettes. Il sert aussi de support de communication entre les designers, les équipes métier et les développeurs.

Cette partie m'a montré qu'une interface réussie repose sur une réflexion en amont. Il faut comprendre les utilisateurs, leurs objectifs et leurs difficultés avant de commencer le développement.

6. Gestion de projet et transformation digitale

6.1 Rôle du chef de projet

Mehdi, chef de projet, m'a expliqué que son rôle consiste à coordonner les besoins métier, les équipes techniques, les responsables, les prestataires et les contraintes de calendrier.

Le chef de projet doit comprendre les objectifs, suivre l'avancement, identifier les risques et faciliter les échanges. Il ne réalise pas nécessairement lui-même toutes les tâches techniques, mais doit être capable de dialoguer avec les spécialistes.

Il joue également un rôle de traduction. Les équipes métier expriment un besoin dans leur vocabulaire, tandis que les développeurs ont besoin d'éléments précis et exploitables. Le chef de projet contribue à transformer ce besoin en objectifs, User Stories et tâches.

6.2 Projet de néo banque et time to market

Le projet de néo banque présenté pendant le stage répond à l'évolution des usages et à la concurrence entre les établissements bancaires.

La notion de time to market désigne le temps nécessaire pour mettre un produit sur le marché. Lorsqu'un concurrent propose déjà une solution comparable, ce délai devient stratégique.

La banque doit cependant trouver un équilibre entre rapidité et maîtrise des risques. Dans le secteur bancaire, aller vite ne doit pas conduire à réduire les contrôles de sécurité, les tests ou les validations.

Ce projet m'a montré qu'une décision technique est aussi une décision stratégique. Le choix d'une architecture, d'un partenaire ou d'un planning dépend du budget, de la concurrence et des ressources disponibles.

6.3 Choix d'un éditeur et contraintes organisationnelles

Face aux contraintes de délai, deux possibilités principales peuvent être étudiées : développer entièrement la solution en interne ou faire appel à un éditeur externe.

Le recours à un éditeur peut permettre de gagner du temps, mais nécessite un processus de sélection rigoureux. Celui-ci peut comprendre un appel d'offres, des présentations, une première sélection, une analyse financière et une évaluation métier et technique.

Il faut aussi vérifier que la solution proposée pourra s'intégrer au système existant et respecter les contraintes de la banque.

Dans une grande organisation, certains composants techniques dépendent de plusieurs entités. L'obtention d'un nouvel outil peut donc nécessiter différentes validations. Pour réduire les délais, les équipes cherchent parfois à réutiliser et adapter les composants déjà disponibles.

Cette situation m'a fait comprendre qu'un projet peut être ralenti par des dépendances organisationnelles, même lorsque l'équipe de développement est prête. La gestion de ces dépendances fait partie du travail de pilotage.

7. La data dans un environnement bancaire

7.1 Ingestion, traitement et restitution

Un professionnel du domaine de la data m'a présenté plusieurs étapes liées à la gestion des données.

L'ingestion consiste à récupérer des données depuis un ou plusieurs systèmes afin de les intégrer dans un environnement où elles pourront être contrôlées et exploitées.

Le traitement peut comprendre le nettoyage, la transformation, l'organisation ou l'enrichissement des données. Une donnée brute n'est pas toujours directement utile ; elle doit souvent être replacée dans un contexte.

La restitution consiste à fournir les données traitées à un utilisateur, un client interne ou un autre service sous une forme exploitable. Le format dépend du besoin : tableau, indicateur, fichier ou alimentation d'un autre système.

Ces étapes montrent que la qualité de la donnée est essentielle. Une erreur lors de l'ingestion ou de la transformation peut se retrouver dans toutes les analyses qui suivent.

7.2 Reporting réglementaire

Un autre volet concerne la préparation et la transmission de certaines données à Bank Al-Maghrib.

Cette activité peut être désignée comme du reporting réglementaire. Elle consiste à préparer des informations selon des formats et des règles précises afin de répondre aux exigences d'une autorité de supervision.

Cette présentation m'a montré que la data dans une banque ne sert pas uniquement à l'analyse interne. Elle participe également au respect des obligations réglementaires et à la transparence du système financier.

Dans ce domaine, l'exactitude, la traçabilité et le respect des délais sont particulièrement importants.

7.3 Valorisation des données

Le professionnel m'a donné l'exemple d'une date de naissance. Seule, cette donnée est brute. Après traitement, elle peut devenir un âge, une tranche d'âge ou un indicateur utilisé dans une analyse de risque.

Cet exemple m'a aidé à comprendre la notion de valorisation. Une donnée prend de la valeur lorsqu'elle permet de répondre à une question ou d'aider à une décision.

Le travail data demande donc des compétences techniques, mais aussi une compréhension du métier bancaire. Il faut savoir quelles transformations sont pertinentes et comment interpréter les résultats.

Le collaborateur m'a expliqué que de nombreux professionnels de la data ont commencé par une formation informatique généraliste puis se sont spécialisés progressivement au fil de leurs projets.

Cette remarque m'a montré l'importance de construire des bases solides en programmation, bases de données, systèmes et analyse avant de choisir une spécialisation.

8. Analyse globale et apports du stage

8.1 Complémentarité des métiers

Le principal enseignement du stage est qu'une application bancaire ne résulte jamais du travail d'un seul métier.

Le besoin est défini par les équipes métier et produit. Le designer prépare les parcours et les interfaces. Les développeurs construisent la solution. Les QA testeurs contrôlent sa conformité. Les chefs de projet coordonnent les acteurs, les délais et les dépendances. Les spécialistes data, sécurité et infrastructure interviennent selon la nature du projet.

Cette complémentarité implique une communication constante. Une User Story mal comprise peut conduire à développer une fonctionnalité techniquement correcte mais inadaptée. Une anomalie mal décrite peut ralentir une correction. Une maquette ambiguë peut créer une différence entre le besoin et le résultat final.

La qualité du produit dépend donc autant de la précision des échanges que de la compétence technique.

8.2 Contraintes du secteur bancaire

Le secteur bancaire impose des contraintes plus fortes que de nombreux autres domaines. Les systèmes manipulent des données personnelles, des opérations financières et des informations sensibles.

La séparation des environnements, les tests, la validation en recette et la traçabilité des modifications sont donc indispensables.

La réglementation influence aussi les projets. Le reporting à Bank Al-Maghrib, les questions liées à la signature électronique et les validations nécessaires pour certains composants montrent que le système d'information est étroitement lié au cadre juridique et organisationnel.

Enfin, les banques doivent concilier deux objectifs parfois opposés : innover rapidement pour répondre aux attentes et à la concurrence, tout en conservant un niveau élevé de maîtrise des risques.

8.3 Liens avec le BUT Informatique

Ce stage m'a permis de relier plusieurs enseignements du BUT Informatique à des situations professionnelles.

La programmation Java m'a aidé à comprendre le rôle de Spring Boot et du développement back-end. Les cours de développement web m'ont permis d'identifier les responsabilités du front-end et l'utilité de technologies comme React.

Les bases de données sont directement liées aux problématiques de data, de transformation et de fiabilité des informations. Les cours de réseaux et systèmes m'aident à comprendre la communication entre services, les environnements distants et le déploiement.

Git, étudié et utilisé dans mes projets, prend une dimension plus importante en entreprise avec la collaboration, les branches, les revues et l'intégration continue.

Enfin, la gestion de projet, les tests et la conception d'interfaces montrent que le développement logiciel ne se limite pas à écrire du code.

8.4 Apports pour mon projet professionnel

Avant ce stage, je connaissais surtout l'informatique à travers les cours et mes projets personnels. Cette immersion m'a montré la diversité des métiers accessibles après un BUT Informatique.

Le développement reste un domaine qui m'intéresse, mais j'ai également pris conscience de l'importance des systèmes, de la qualité logicielle, de la gestion de projet et de la data.

Les échanges avec les professionnels ont confirmé qu'un bon profil technique doit aussi savoir communiquer, comprendre un besoin et travailler en équipe.

Cette expérience renforce mon intérêt pour des domaines situés à l'intersection du développement, des systèmes et de l'exploitation, notamment le DevOps, le DevSecOps et l'administration système.

Elle m'encourage à continuer à approfondir Linux, les réseaux, Git, Docker, Java et les méthodes de déploiement.

9. Conclusion

Ce stage au sein de BANK OF AFRICA, réalisé du 22 juin au 17 juillet 2026, m'a permis de découvrir la réalité d'un département informatique et digital dans une grande entreprise bancaire.

Grâce aux échanges avec plusieurs professionnels, j'ai observé le cycle de vie d'un projet numérique, depuis le cadrage du besoin jusqu'à la production, en passant par le design, le développement, les tests et la recette.

J'ai compris que la transformation digitale d'une banque est un processus complexe. Elle ne dépend pas uniquement des technologies utilisées, mais aussi de l'organisation des équipes, de la qualité des échanges, de la réglementation, de la sécurité, des délais et de la stratégie de l'entreprise.

Cette expérience a également élargi ma vision des métiers de l'informatique. Elle m'a montré qu'un développeur doit comprendre le projet dans son ensemble et que les métiers de QA, de designer, de chef de projet et de la data sont indispensables à la réussite d'une solution.

Même si le stage était principalement orienté vers l'observation et comportait des périodes de travail autonome, sa durée m'a permis de prendre du recul, de consolider mes notes et d'approfondir les notions présentées. J'ai pu construire une vision plus cohérente du fonctionnement d'un système d'information bancaire.

Enfin, ce stage constitue une étape importante dans mon parcours. Il confirme mon intérêt pour l'informatique professionnelle et me donne des objectifs plus précis pour la suite : consolider mes compétences techniques, développer ma culture des systèmes et des projets, et continuer à découvrir les métiers liés au déploiement, à l'exploitation et à la sécurité des applications.

Glossaire

Agile : Méthode de gestion de projet fondée sur des cycles courts, la collaboration et l'adaptation.

API : Interface permettant à plusieurs applications de communiquer entre elles.

Back-end : Partie d'une application qui gère les traitements, les données et la logique métier.

Backlog : Liste ordonnée des besoins, fonctionnalités, corrections et tâches d'un produit.

Batch : Traitement automatisé, souvent planifié, exécutant une suite d'opérations.

CI/CD : Pratiques d'intégration et de déploiement continus visant à automatiser les contrôles et les livraisons.

Data ingestion : Processus de récupération et d'intégration de données provenant de différentes sources.

Environnement de recette : Espace dans lequel une solution est validée avant sa mise en production.

Front-end : Partie visible d'une application avec laquelle l'utilisateur interagit.

MVP : Version minimale utilisable d'un produit contenant ses fonctionnalités essentielles.

Product Owner : Responsable de la vision et de la priorisation du produit.

QA : Assurance qualité ; activité consistant à contrôler la conformité et la fiabilité d'un logiciel.

Reporting réglementaire : Préparation et transmission d'informations imposées par une autorité de contrôle.

Scrum Master : Personne facilitant l'application de la méthode Agile au sein d'une équipe.

Sprint : Période de travail courte pendant laquelle une équipe réalise un ensemble d'objectifs.

Squad : Équipe pluridisciplinaire réunie autour d'un produit ou d'un objectif.

Time to market : Temps nécessaire pour mettre un produit ou un service sur le marché.

User Story : Description concise d'un besoin formulé du point de vue d'un utilisateur.

UX/UI : Conception de l'expérience utilisateur et de l'interface graphique.

Wireframe : Représentation schématique de la structure d'une interface.

Références bibliographiques et webographiques

[1] BANK OF AFRICA, site institutionnel, pages de présentation du groupe et de ses activités, consultées pour la description générale de l'entreprise.

[2] Apache Software Foundation, documentation officielle d'Apache JMeter, consultée pour la définition générale de l'outil de test.

[3] Meta, documentation officielle de React, consultée pour la définition générale de la bibliothèque.

[4] Spring, documentation officielle de Spring Boot, consultée pour la définition générale du framework.

[5] GitLab, Jenkins, Figma et Apple Developer, documentations officielles consultées pour les définitions générales des outils et technologies.

[6] Notes personnelles prises pendant le stage et documents internes consultés sur place. Les informations confidentielles ne sont pas reproduites dans ce rapport.

Annexes

Annexe A - Synthèse thématique du stage

Domaine observé	Sujets abordés	Outils et technologies
Organisation de projet	Agile, squads, backlog, User Stories, sprints	Jira, GitLab
Développement web	Front-end, back-end, API, centralisation des services	React, Tailwind CSS, Material UI, Spring Boot, Java, Insomnia
Qualité logicielle	Tests fonctionnels, automatisés et de montée en charge	JMeter
Design UX/UI	Design Thinking, wireframes, personas, MVP	Figma
Gestion de projet	Néo banque, appels d'offres, time to market	Roadmap, User Stories
Développement back-end	Batches, configurations, fichiers entrants et sortants	FileZilla, GitLab, Jira
Développement mobile	Applications iOS et interfaces mobiles	Swift, SwiftUI
Data	Ingestion, transformation, restitution, reporting réglementaire	Outils internes