DOCUMENT DE SPÉCIFICATIONS

Plateforme de Gestion et Monitoring
No-Code/Low-Code

DigitalBank - Projet de Groupe

Groupe : DevForce

Date : Janvier 2026

Membres : 6 personnes

Formation : ESIS

Examen - Partie 2 : Travail de Groupe

Table des matières

1 INTRODUCTION

1.1 Contexte du Projet

DigitalBank souhaite mettre en place une plateforme complète de gestion des données bancaires et de monitoring de sécurité. Cette plateforme doit permettre de visualiser les données en temps réel, détecter les activités frauduleuses, gérer les accès utilisateurs et monitorer la sécurité du système.

Le projet s'inscrit dans une démarche de transformation digitale visant à améliorer la sécurité et l'efficacité opérationnelle de la banque.

1.2 Objectifs

Les objectifs principaux de cette plateforme sont :

— Visualiser les données clients et transactions en temps réel

— Détecter et alerter sur les activités frauduleuses

— Gérer les accès utilisateurs avec un système RBAC (Role-Based Access Control)

— Monitorer la sécurité et les performances du système

— Générer des rapports automatisés

1.3 Portée du Projet

Le projet couvre le développement d'une solution no-code/low-code complète incluant :

— **Backend et API** : Supabase (PostgreSQL + API REST auto-générée)

— **Dashboards de visualisation** : Metabase + Grafana

— **Système de détection de fraude par IA** : API Flask + Machine Learning

— **Automatisation des workflows** : n8n

— **Monitoring et logs centralisés** : Prometheus + Grafana

— **Sécurité et conformité** : RBAC, MFA, Audit logs

2 DÉFINITION DES PersonneS

2.1 Personne 1 : Administrateur Système

2.1.1 Objectifs

— Maintenir la sécurité et la disponibilité de la plateforme
— Gérer les utilisateurs et leurs permissions
— Superviser l'infrastructure technique
— Assurer la conformité avec les réglementations (RGPD, PCI-DSS)

2.1.2 Besoins

— Gérer les rôles et permissions (RBAC)
— Monitorer l'état de l'infrastructure (CPU, RAM, disque)
— Consulter les logs d'audit et de sécurité
— Configurer les alertes système
— Gérer les accès à la base de données
— Effectuer des backups et restaurations

2.1.3 Frustrations

— Trop de systèmes séparés pour gérer l'infrastructure
— Alertes trop nombreuses et peu pertinentes
— Manque de visibilité sur les actions des utilisateurs
— Processus manuels chronophages

2.1.4 Niveau d'accès requis

— Accès complet à tous les modules
— Droits d'administration sur la base de données
— Capacité à créer/modifier/supprimer des utilisateurs
— Accès aux logs système et d'audit

2.2 Personne 2 : Analyste de Sécurité

2.2.1 Objectifs

— Détecter les fraudes en temps réel
— Analyser les patterns de transactions suspectes
— Réduire le taux de fausses alertes
— Protéger les clients contre les activités frauduleuses

2.2.2 Besoins

— Dashboard temps réel des alertes de fraude
— Visualisation géographique des transactions
— Score de risque pour chaque transaction
— Historique des fraudes détectées
— Outils d'analyse des patterns (graphiques, statistiques)
— Système d'alerte automatique (email, SMS)
— Capacité à bloquer une carte rapidement

2.2.3 Frustrations

— Délai entre la fraude et la détection
— Trop de fausses alertes (faux positifs)
— Interface complexe pour analyser les données
— Manque d'outils de visualisation

2.2.4 Niveau d'accès requis

— Lecture sur toutes les transactions
— Accès aux données clients (limité)
— Capacité à marquer une transaction comme frauduleuse
— Accès aux dashboards analytiques
— Pas de droits d'administration

2.3 Personne 3 : Agent du Service Client

2.3.1 Objectifs

— Répondre aux demandes clients rapidement
— Résoudre les problèmes liés aux comptes et transactions
— Assister les clients en cas de fraude suspectée
— Maintenir un haut niveau de satisfaction client

2.3.2 Besoins

— Rechercher un client rapidement (par nom, email, numéro de compte)
— Visualiser les informations complètes d'un client
— Consulter l'historique des transactions
— Vérifier le solde des comptes
— Bloquer/débloquer une carte en urgence
— Interface simple et intuitive
— Temps de réponse rapide

2.3.3 Frustrations

— Systèmes lents et interfaces complexes
— Informations dispersées dans plusieurs outils
— Manque de formation sur les nouveaux outils
— Procédures trop longues pour des actions simples

2.3.4 Niveau d'accès requis

— Lecture sur les données clients
— Lecture sur les transactions
— Modification limitée (bloquer/débloquer carte)
— Pas d'accès à la configuration système
— Pas d'accès aux données sensibles (mots de passe, codes PIN)

2.4 Personne 4 : Client

2.4.1 Objectifs

— Consulter ses comptes et soldes
— Vérifier ses transactions récentes
— Gérer son budget personnel
— Être alerté en cas d'activité suspecte

2.4.2 Besoins

— Visualiser le solde de ses comptes
— Consulter l'historique de ses transactions
— Rechercher une transaction spécifique
— Recevoir des notifications (fraude, solde faible)
— Interface simple et responsive
— Sécurité de ses données

2.4.3 Frustrations

— Ne pas être informé rapidement en cas de fraude
— Interfaces bancaires trop complexes
— Temps de chargement lent
— Manque de transparence sur les frais

2.4.4 Niveau d'accès requis

— Lecture uniquement sur ses propres données
— Aucun accès aux données d'autres clients
— Aucun droit d'administration
— Consultation limitée (ses comptes uniquement)

3 USER STORIES

3.1 Méthodologie

3.1.1 Format utilisé

Toutes les user stories suivent le format standard :

> **En tant que** [Personne], **je veux** [action], **afin de** [bénéfice].

3.1.2 Priorisation selon MoSCoW

— Must have : Fonctionnalités essentielles, sans lesquelles le système ne peut pas fonctionner
— Should have : Fonctionnalités importantes mais non critiques
— Could have : Fonctionnalités souhaitables si le temps le permet
— Won't have : Fonctionnalités hors périmètre pour cette version

3.2 Liste des User Stories (Priorisées)

3.2.1 Must Have - Priorité Haute

US-001 : Authentification sécurisée

En tant qu'utilisateur (tous profils), **je veux** me connecter de manière sécurisée avec authentification multi-facteurs (MFA), **afin de** protéger mon accès et mes données sensibles.

Critères d'acceptation :

— Login avec email/mot de passe
— Activation MFA (TOTP ou SMS)
— Session sécurisée avec JWT
— Déconnexion automatique après inactivité

Personne : Tous | **Module :** Authentification | **Points :** 8

US-002 : Visualisation temps réel des alertes de fraude

En tant qu'analyste de sécurité, **je veux** voir en temps réel toutes les alertes de fraude avec score de risque, **afin de** réagir rapidement et bloquer les transactions suspectes.

Critères d'acceptation :

— Dashboard affichant les alertes en temps réel
— Score de risque visible (0-100%)
— Filtrage par période, montant, catégorie
— Rafraîchissement automatique toutes les 30 secondes

Personne : Analyste Sécurité | **Module :** Dashboard Analytics | **Points :** 13

US-003 : Gestion des permissions utilisateurs (RBAC)

En tant qu'administrateur système, **je veux** gérer les rôles et permissions de chaque utilisateur, **afin de** respecter le principe du moindre privilège et sécuriser les accès.

Critères d'acceptation :

— Créer/modifier/supprimer des utilisateurs
— Assigner des rôles (admin, analyst, customer_service, customer)
— Définir des permissions granulaires par table
— Historique des modifications de permissions

Personne : Administrateur | **Module :** RBAC / Auth | **Points :** 13

US-004 : Recherche et consultation client

En tant qu'agent du service client, **je veux** rechercher un client par nom, email ou numéro de compte, **afin d'**accéder rapidement à ses informations pour l'assister.

Critères d'acceptation :

— Barre de recherche avec auto-complétion
— Résultats en moins de 2 secondes
— Affichage des informations complètes du client
— Historique des 20 dernières transactions

Personne : Service Client | **Module :** Dashboard Service Client | **Points :** 8

US-005 : Consultation des données personnelles

En tant que client, **je veux** consulter uniquement mes propres comptes et transactions, **afin de** suivre mon activité bancaire.

Critères d'acceptation :

— Accès uniquement à ses propres données (RLS)
— Affichage du solde de tous ses comptes
— Liste des transactions avec filtres (date, montant, catégorie)
— Impossibilité d'accéder aux données d'autres clients

Personne : Client | **Module :** API + RLS | **Points :** 8

US-006 : Détection automatique de fraude par IA

En tant qu'analyste de sécurité, **je veux** qu'un modèle de machine learning analyse automatiquement chaque transaction, **afin de** détecter les fraudes sans intervention manuelle.

Critères d'acceptation :

— Modèle ML entraîné sur données historiques

— Prédiction en temps réel (<500ms)

— Score de fraude calculé pour chaque transaction

— Marquage automatique des transactions à risque élevé

Personne : Analyste Sécurité | **Module :** API Flask ML | **Points :** 21

US-007 : Alertes automatiques par email

En tant qu'analyste de sécurité, **je veux** recevoir une alerte email immédiate en cas de fraude détectée, **afin d'**être notifié même si je ne suis pas sur le dashboard.

Critères d'acceptation :

— Email envoyé automatiquement si score fraude $> 80\%$

— Contenu : détails transaction, client, score de risque

— Lien direct vers le dashboard

— Délai d'envoi < 1 minute

Personne : Analyste Sécurité | **Module :** Workflows n8n | **Points :** 5

US-008 : Monitoring de l'infrastructure

En tant qu'administrateur système, **je veux** monitorer en temps réel les métriques système (CPU, RAM, réseau), **afin de** détecter les problèmes de performance avant qu'ils n'impactent les utilisateurs.

Critères d'acceptation :

— Dashboard Grafana affichant CPU, RAM, disque, réseau

— Historique sur 7 jours minimum

— Alertes automatiques si seuil dépassé (CPU $> 80\%$)

— Temps de réponse API (latence moyenne, p95, p99)

Personne : Administrateur | **Module :** Monitoring Grafana | **Points :** 13

3.2.2 Should Have - Priorité Moyenne

US-009 : Visualisation géographique des transactions

En tant qu'analyste de sécurité, **je veux** voir une carte du monde avec les transactions en temps réel, **afin de** détecter visuellement des patterns géographiques suspects.

Critères d'acceptation :

— Carte interactive (Leaflet, Mapbox)

— Points colorés selon le risque (vert, orange, rouge)

— Zoom sur une région pour voir les détails

— Filtrage par période

Personne : Analyste Sécurité | **Module :** Dashboard Analytics | **Points :** 13

US-010 : Blocage de carte en urgence

En tant qu'agent du service client, **je veux** bloquer ou débloquer une carte en un clic, **afin de** protéger un client en cas de fraude signalée.

Critères d'acceptation :

— Bouton "Bloquer/Débloquer" visible

— Confirmation avant action

— Notification au client par email/SMS

— Logs de l'action dans l'audit trail

Personne : Service Client | **Module :** Dashboard Service Client | **Points :** 8

US-011 : Rapport quotidien automatisé

En tant qu'analyste de sécurité, **je veux** recevoir chaque matin un rapport automatique des activités de la veille, **afin d'**avoir une vue d'ensemble sans devoir consulter le dashboard.

Critères d'acceptation :

— Rapport envoyé par email à 8h chaque jour

— Contenu : nb transactions, nb fraudes, top 5 marchands, revenus

— Format PDF ou HTML lisible

— Graphiques et statistiques clés

Personne : Analyste Sécurité | **Module :** Workflows n8n | **Points :** 8

US-012 : Consultation des logs d'audit

En tant qu'administrateur système, **je veux** consulter tous les logs d'audit des actions sensibles, **afin de** tracer qui a fait quoi et quand (conformité RGPD).

Critères d'acceptation :

— Table audit_logs contenant toutes les actions

— Filtrage par utilisateur, action, date

— Export CSV pour analyse externe

— Rétention des logs sur 1 an minimum

Personne : Administrateur | **Module :** Audit Logs | **Points :** 8

US-013 : Statistiques et graphiques analytiques

En tant qu'analyste de sécurité, **je veux** visualiser des graphiques d'évolution (transactions par heure, par catégorie), **afin d'**identifier des tendances et patterns.

Critères d'acceptation :

— Graphique : Évolution du nombre de transactions par heure

— Graphique : Répartition par catégorie de marchand (camembert)

— Graphique : Top 10 transactions par montant

— Export des données en CSV

Personne : Analyste Sécurité | **Module :** Dashboard Analytics | **Points :** 8

3.2.3 Could Have - Priorité Basse

US-014 : Alerte Slack/Discord pour l'équipe

En tant qu'analyste de sécurité, **je veux** recevoir des alertes dans Slack/Discord en cas de fraude critique, **afin de** collaborer rapidement avec l'équipe.

Critères d'acceptation :

— Webhook configuré vers Slack ou Discord

— Message avec détails de la fraude

— Mention @channel pour fraudes critiques (> 10 000€)

Personne : Analyste Sécurité | **Module :** Workflows n8n | **Points :** 3

US-015 : Export des données en Excel

En tant qu'analyste de sécurité, **je veux** exporter les données de transactions en Excel, **afin d'**effectuer des analyses complémentaires hors plateforme.

Critères d'acceptation :

— Bouton "Export Excel" sur chaque dashboard

— Export au format .xlsx

— Limite : 10 000 lignes maximum

Personne : Analyste Sécurité | **Module :** Dashboard Analytics | **Points :** 3

US-016 : Notifications push pour le client

En tant que client, **je veux** recevoir une notification push sur mon téléphone à chaque transaction, **afin de** détecter immédiatement une utilisation frauduleuse de ma carte.

Critères d'acceptation :

— Notification en temps réel (<10 secondes)

— Contenu : montant, marchand, date/heure

— Option pour désactiver les notifications

Personne : Client | **Module :** Notifications | **Points :** 8

3.2.4 Won't Have - Hors Périmètre

US-017 : Application mobile native

En tant que client, **je veux** utiliser une application mobile native, **afin d'**accéder à mes comptes depuis mon smartphone.

Raison : Hors périmètre pour cette version (web responsive uniquement)
Points : 21

US-018 : Chatbot IA pour le support client

En tant qu'agent du service client, **je veux** interagir avec un chatbot IA, **afin de** répondre plus rapidement aux questions fréquentes.

Raison : Hors périmètre (nécessite un développement important)
Points : 21

US-019 : Intégration avec systèmes bancaires tiers

En tant qu'administrateur, **je veux** intégrer la plateforme avec des systèmes bancaires tiers, **afin de** synchroniser les données en temps réel.

Raison : Hors périmètre (pas d'API bancaires externes disponibles)
Points : 34

4 PRIORISATION MOSCOW - RÉCAPITULATIF

primaryblue			
mustred !30 **Must Have**	8	89	60%
shouldyellow !30 **Should Have**	5	45	30%
couldgreen !30 **Could Have**	3	14	9%
wontgray !30 **Won't Have**	3	76	Hors périmètre
shouldyellow !50 **TOTAL**	**19**	**224**	**100%**

TABLE 1 – Répartition des User Stories selon la méthode MoSCoW

4.1 Analyse de la Priorisation

4.1.1 Must Have - MVP (Minimum Viable Product)

Les 8 user stories classées en **Must Have** représentent 60% de l'effort total et constituent le cœur fonctionnel de la plateforme. Sans ces fonctionnalités, le système ne peut pas être mis en production. Elles couvrent :

— L'authentification sécurisée (US-001)
— La détection de fraude par IA (US-006)
— Les alertes en temps réel (US-002, US-007)
— La gestion des accès RBAC (US-003)
— Les dashboards pour chaque profil utilisateur (US-004, US-005)
— Le monitoring infrastructure (US-008)

4.1.2 Should Have - Fonctionnalités importantes

Les 5 user stories **Should Have** (30% de l'effort) améliorent significativement l'expérience utilisateur et l'efficacité opérationnelle :

— Visualisation géographique avancée (US-009)
— Actions rapides (blocage carte) (US-010)
— Automatisation des rapports (US-011)
— Audit et conformité (US-012)
— Analytics avancées (US-013)

4.1.3 Could Have - Nice to have

Les 3 user stories **Could Have** (9% de l'effort) sont des améliorations souhaitables mais non essentielles. Elles seront implémentées uniquement si le temps le permet.

4.1.4 Won't Have - Hors périmètre

Les 3 user stories **Won't Have** sont explicitement exclues de cette version du projet. Elles pourront être considérées pour des versions futures.

5 MATRICE DE TRAÇABILITÉ

Cette matrice permet de suivre l'implémentation de chaque user story dans les différentes phases du projet.

primaryblue			
US-001	Tous	Authentification	Phase 2.1 + Phase 3.1
US-002	Analyste Sécurité	Dashboard Analytics	Phase 2.2
US-003	Administrateur	RBAC / Auth	Phase 3.1
US-004	Service Client	Dashboard Service Client	Phase 2.2
US-005	Client	API + RLS	Phase 2.1
US-006	Analyste Sécurité	API Flask ML	Phase 2.3
US-007	Analyste Sécurité	Workflows n8n	Phase 2.4
US-008	Administrateur	Monitoring Grafana	Phase 2.5
US-009	Analyste Sécurité	Dashboard Analytics	Phase 2.2
US-010	Service Client	Dashboard Service Client	Phase 2.2
US-011	Analyste Sécurité	Workflows n8n	Phase 2.4
US-012	Administrateur	Audit Logs	Phase 3.2
US-013	Analyste Sécurité	Dashboard Analytics	Phase 2.2
US-014	Analyste Sécurité	Workflows n8n	Phase 2.4
US-015	Analyste Sécurité	Dashboard Analytics	Phase 2.2
US-016	Client	Notifications	Phase 2.4
US-017	Client	Mobile App	Hors périmètre
US-018	Service Client	Chatbot IA	Hors périmètre
US-019	Administrateur	Intégrations	Hors périmètre

TABLE 2: Matrice de traçabilité des User Stories

6 RISQUES IDENTIFIÉS

primaryblue			
Complexité technique des outils no-code	Moyenne	Élevé	Formation préalable, tutoriels, POC
Limitations des outils gratuits	Élevée	Moyen	Choix d'outils open-source, alternatives
Déséquilibre des contributions entre membres	Moyenne	Élevé	Répartition claire, suivi hebdomadaire
Problèmes d'intégration entre outils	Moyenne	Moyen	Tests d'intégration précoces
Retards dans le planning	Moyenne	Élevé	Buffer de temps, priorisation MoSCoW
Difficultés avec le modèle ML	Moyenne	Élevé	Utiliser des modèles pré-entraînés, validation early
Problèmes de sécurité (vulnérabilités)	Faible	Très élevé	Tests sécurité réguliers, OWASP ZAP
Manque de données pour le ML	Faible	Moyen	Utiliser la base de Partie 1, données synthétiques

TABLE 3 – Analyse des risques du projet

6.1 Plan d'Action

Pour chaque risque identifié, l'équipe projet mettra en place :

— **Suivi régulier** : Réunions hebdomadaires pour évaluer l'avancement
— **Communication** : Canal Discord/Slack pour communication continue
— **Documentation** : Partage de ressources et tutoriels entre membres
— **Tests précoces** : Validation technique dès les premières semaines
— **Plan B** : Alternatives identifiées pour chaque outil critique

7 CONCLUSION

Ce document de spécifications définit les bases du projet de plateforme de gestion et monitoring pour DigitalBank. Les 19 user stories identifiées couvrent l'ensemble des besoins fonctionnels, avec une priorisation claire selon la méthode MoSCoW.

7.1 Livrables Attendus

Les 8 user stories **Must Have** constituent le MVP (Minimum Viable Product) et seront développées en priorité. Les user stories **Should Have** et **Could Have** seront implémentées en fonction de l'avancement et du temps disponible.

7.2 Approche Centrée Utilisateur

Les 4 Personnes définis permettent de guider les choix de conception et d'assurer que la plateforme répond aux besoins réels de chaque type d'utilisateur :

— **Administrateur système** : Sécurité, gestion des accès, monitoring
— **Analyste de sécurité** : Détection de fraude, alertes, analytics
— **Agent du service client** : Recherche client, assistance rapide
— **Client** : Consultation de comptes, transparence, sécurité

7.3 Technologies No-Code/Low-Code

L'utilisation d'outils no-code/low-code (Supabase, Metabase, Grafana, n8n) permettra de :

— Réduire significativement le temps de développement
— Faciliter la maintenance et les évolutions futures
— Permettre à des profils non-développeurs de contribuer
— Garantir une qualité professionnelle grâce à des outils éprouvés

7.4 Prochaines Étapes

Les prochaines étapes du projet seront :

1. Conception de l'architecture technique détaillée
2. Modélisation de la base de données (ERD)
3. Justification des choix technologiques
4. Mise en place de l'environnement de développement
5. Démarrage du développement par les user stories Must Have

Document rédigé par : DevForce

Date : Janvier 2026

Version : 1.0