**Mémoire de Projet**

*Concepteur développeur d’applications*

Application BlockLumen

Plateforme de simulation de trading de cryptomonnaies.

Table des matières

[1. Présentation 3](#_Toc199338776)

[a. Introduction en français 3](#_Toc199338777)

[b. Introduction et présentation du projet en anglais 3](#_Toc199338778)

[2. Contexte 4](#_Toc199338779)

[a. Objectif 4](#_Toc199338780)

[b. Technologies 4](#_Toc199338781)

[1. Back-end 4](#_Toc199338782)

[2. Front-end 4](#_Toc199338783)

[3. OS / IDE / Utilitaires 4](#_Toc199338784)

[3. Conception et modélisation des données 5](#_Toc199338785)

[a. Clarification des besoins & dictionnaire de données 5](#_Toc199338786)

[4.Le Design 6](#_Toc199338787)

[5. Environnement de développement 6](#_Toc199338788)

[6. Développement de l’application 6](#_Toc199338789)

[7. 6](#_Toc199338790)

[8. 6](#_Toc199338791)

[9. 6](#_Toc199338792)

[10. 6](#_Toc199338793)

[11. 6](#_Toc199338794)

[12. 6](#_Toc199338795)

[13. 6](#_Toc199338796)

[14. 6](#_Toc199338797)

[15. Conclusion 6](#_Toc199338798)

[a. Remerciements 6](#_Toc199338799)

[b. Axes d’amélioration 6](#_Toc199338800)

[c. Retour d’expérience 6](#_Toc199338801)

# 1. Présentation

## Introduction en français

Je m’appelle Nicolas Lopez. En 2022, j’ai intégré la Prépa Concepteur Développeur d’Applications à l’Idem, où j’ai validé ma première année avec succès.

Dans ce cadre, j’ai effectué un stage de fin d’études chez REVIMPORT S.A.S., dont la mission était double : me familiariser avec les process en entreprise et lancer la migration de l’ERP Navision vers Microsoft Business Central. Mon implication m’a valu une gratification de stage et l’offre d’un contrat d’alternance d’un an.

Durant cette alternance, j’ai poursuivi l’analyse fonctionnelle, la conception de la base de données et le développement des premiers modules métiers. Malheureusement, REVIMPORT a été placée en liquidation judiciaire, entraînant un licenciement économique et l’interruption du projet de migration.

Aujourd’hui, dans le cadre de ma deuxième année et pour valider mon titre RNCP 6, je consacre mon mémoire à BlockLumen, une plateforme de simulation de trading de cryptomonnaies. Ce projet représente l’aboutissement de ma formation et de mes expériences.

## Introduction et présentation du projet en anglais

# 2. Contexte

Le marché des cryptomonnaies, en forte croissance, reste très volatil et complexe pour les débutants. BlockLumen propose une simulation sécurisée du trading (achats, ventes, gestion de portefeuille) enrichie de modules pédagogiques et d’une veille d’actualités pour guider l’utilisateur pas à pas. Ce nom, court et évocateur, associe “Block” (blockchain) et “Lumen” (lumière), pour symboliser l’éclaircissement et la compréhension du trading crypto

## a. Objectif

Permettre aux utilisateurs de tester leurs stratégies dans des conditions de marché réalistes (cours différés et frais inclus) sans risquer leur capital, grâce à une interface claire et responsive sur desktop et mobile.

## b. Technologies

### 1. Back-end

**Node.js & Express :** Cadre JavaScript léger et performant pour développer les API REST.

**MySQL :** Base de données relationnelle robuste, hébergeant utilisateurs, portefeuilles et historiques de transactions.

**Docker :** Containerisation du serveur et de la base de données pour garantir portabilité et isolation.

**TypeScript :** Typage statique pour renforcer la fiabilité et la maintenabilité du code.

### 2. Front-end

**React :** Bibliothèque JavaScript pour construire une interface dynamique et réactive.

**Vite :** Outil de bundling ultra-rapide, optimisé pour le développement moderne.

**Tailwind CSS :** Framework utilitaire facilitant la mise en forme et le design cohérent.

**React-Chart.js-2 / Chart.js :** Pour afficher graphiques de prix et indicateurs en temps réel.

### 3. OS / IDE / Utilitaires

**Git & GitHub :** Gestion de versions et collaboration.

**VS Code :** Environnement de développement, avec extensions TypeScript et Docker.

**Postman :** Tests et appels d’API.

**Figma :** Prototypage des écrans et design UX/UI.

**app.diagrams.net / Draw.io :** Modélisation du MCD et des architectures applicatives.

# 3. Conception et modélisation des données

Pour structurer la base de données de BlockLumen, la méthode Merise a été utilisée, fondée sur la distinction entre modèles conceptuel, logique et physique. Après une phase d’étude préalable des besoins (simulation d’ordres, suivi de portefeuilles, actualisation des cours), un MCD détaillé a été élaboré, puis traduit en MLD et enfin en schéma physique optimisé. Cette démarche assure la cohérence, la performance et l’évolutivité de la solution.

La structuration des données de BlockLumen est déclinée en quatre étapes :

1. Clarification des besoins & dictionnaire de données,
2. Modèle conceptuel de données (MCD),
3. Modèle logique de données (MLD),
4. Modèle physique SQL.

Pour les parties Merise, Base de données et développement, une convention en anglais a été adopté.

## Clarification des besoins & dictionnaire de données

La phase de clarification a pour objectif de recenser et formaliser l’ensemble des besoins fonctionnels et des flux d’information de BlockLumen, puis de définir un dictionnaire précisant chaque donnée manipulée par le système. Cette étape garantit une base solide pour la modélisation conceptuelle, en assurant cohérence terminologique et exhaustivité des entités.

L’ensemble des exigences s’articule autour de cinq processus clés :

1. **Gestion des comptes :** inscription, authentification sécurisée (JWT + bcrypt), récupération de mot de passe, et personnalisation des paramètres.
2. **Visualisation des cours :** affichage en temps réel (API CoinGecko ou dataset) de plusieurs cryptomonnaies avec sélection de périodes (1 h, 1 j, 1 sem., 1 mois).
3. **Simulation de trading :** crédit initial de 10 000 $ fictifs, module d’achat/vente avec calcul instantané des gains/pertes et mise à jour du solde.
4. **Modules pédagogiques :** séquence de leçons ordonnées pour découvrir les concepts du trading crypto.
5. **Veille d’actualités :** flux d’articles externes (RSS ou API) classés par date et résumé.

La plateforme doit rester claire et responsive (desktop, tablette, mobile) et regrouper les sections **Landing Page**, **Learn**, **Train**, **News**, **Account** et **Settings**. Ces processus et règles de gestion (API, sécurité, responsive) constituent le socle de la future modélisation Merise.

# 4.Le Design

# 5. Environnement de développement

# 6. Développement de l’application

# 7.

# 8.

# 9.

# 10.

# 11.

# 12.

# 13.

# 14.

# 15. Conclusion

## a. Remerciements

## b. Axes d’amélioration

## c. Retour d’expérience