



M2L Product

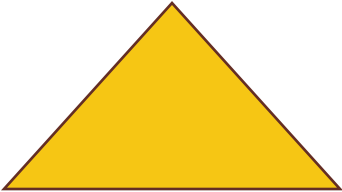
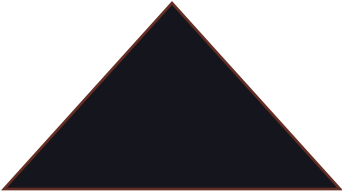
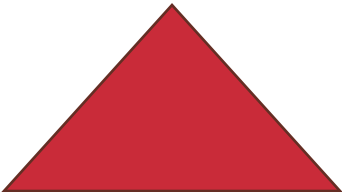
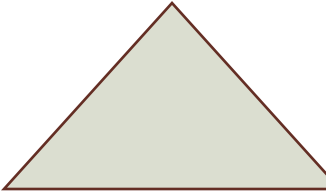
Site pour la M2L

By AllSecurity



Charte Graphique



			
Woodsmoke	Candlelight	Brick Red	Moon Mist
RGB(246,198,20)	RGB(21,22,29)	RGB(201,43,57)	RGB(219,222,208)
HEX : #f6c614	HEX : #15161d	HEX : #c92b39	HEX : #dbded0

Police :

Contexte :

Cette ressource pédagogique est destinée aux enseignants. Elle constitue une matière d'œuvre à partir de laquelle ils pourront placer leurs étudiants dans différentes situations professionnelles décrites dans le référentiel des activités professionnelles du BTS SIO.

Le contexte proposé est celui de la Maison de Ligues de Lorraine (M2L) qui a pour mission de fournir des espaces et des services aux différentes ligues sportives régionales et à d'autres structures hébergées.

Ce contexte est associé à différentes propositions de projets et missions susceptibles d'être réalisés par les étudiants dans l'horaire de PPE à différents moments de la formation.

Il peut aussi être utilisé pour illustrer certains savoirs ou savoir-faire associés à différents modules d'enseignement, ceci aussi bien pour les modules communs (SI) que pour les modules spécifiques des parcours SISR et SLAM.

Les choix d'exploitation pédagogique de ce contexte sont laissés à la libre initiative des professeurs.

Problématique :

L'entreprise M2L souhaite promouvoir le sport dans sa région en mettant en vente des produits sportifs dans un but non-lucratif ?

Solution :

Développement d'un site e-commerce en REACT JS !

Diagramme de Gantt

Untitled Gantt Project

5 déc. 2023

Diagramme de Gantt

4

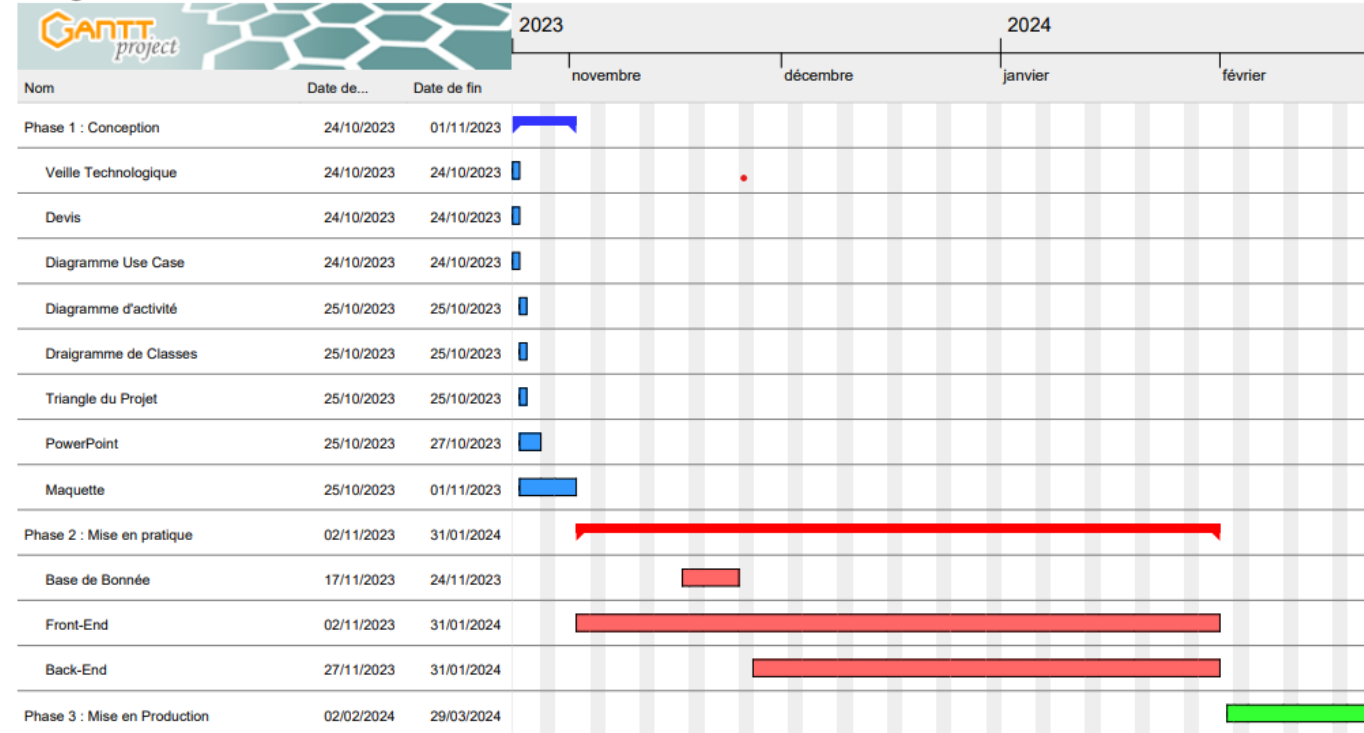


Diagramme de classe

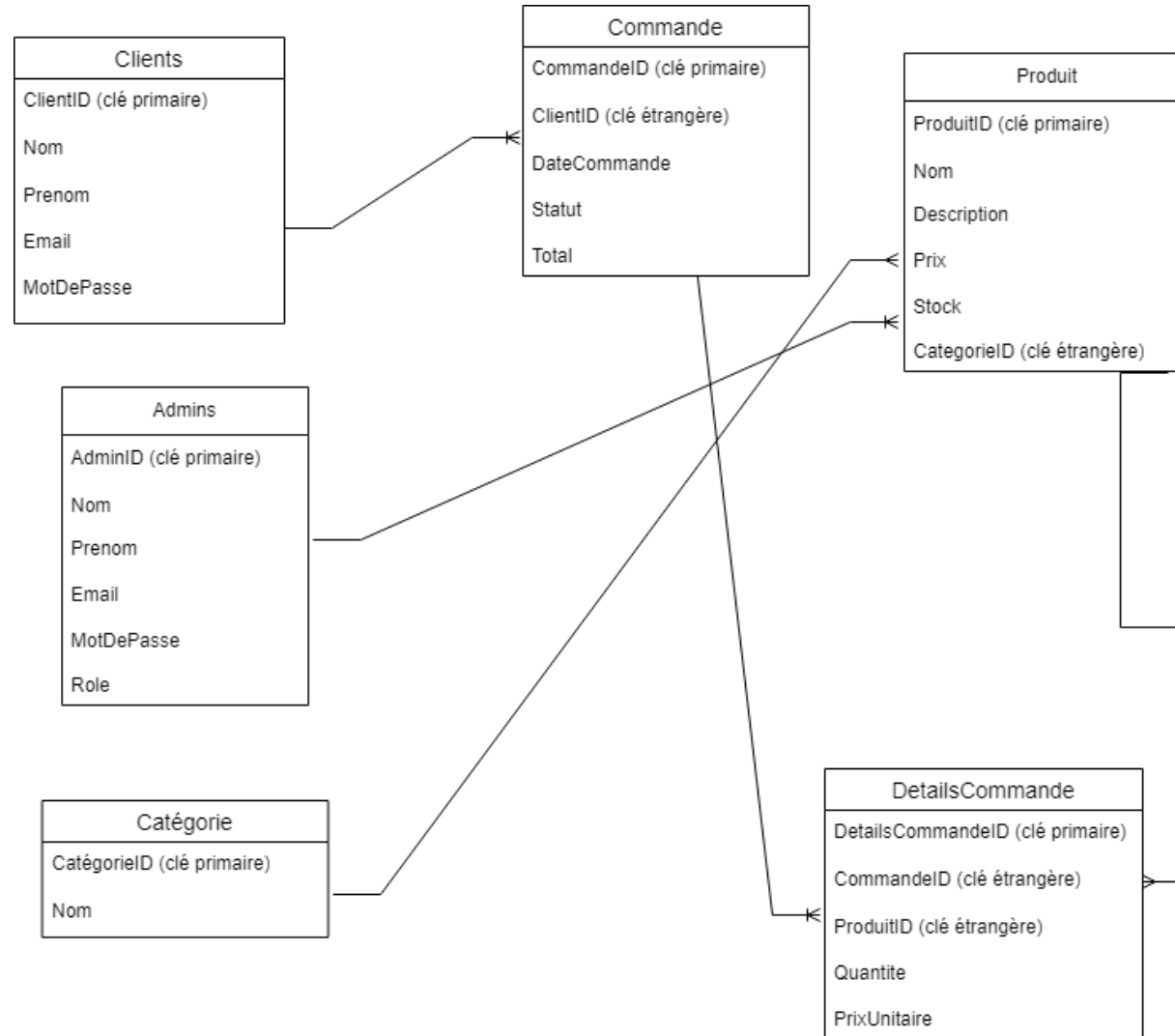


Diagramme d'activité

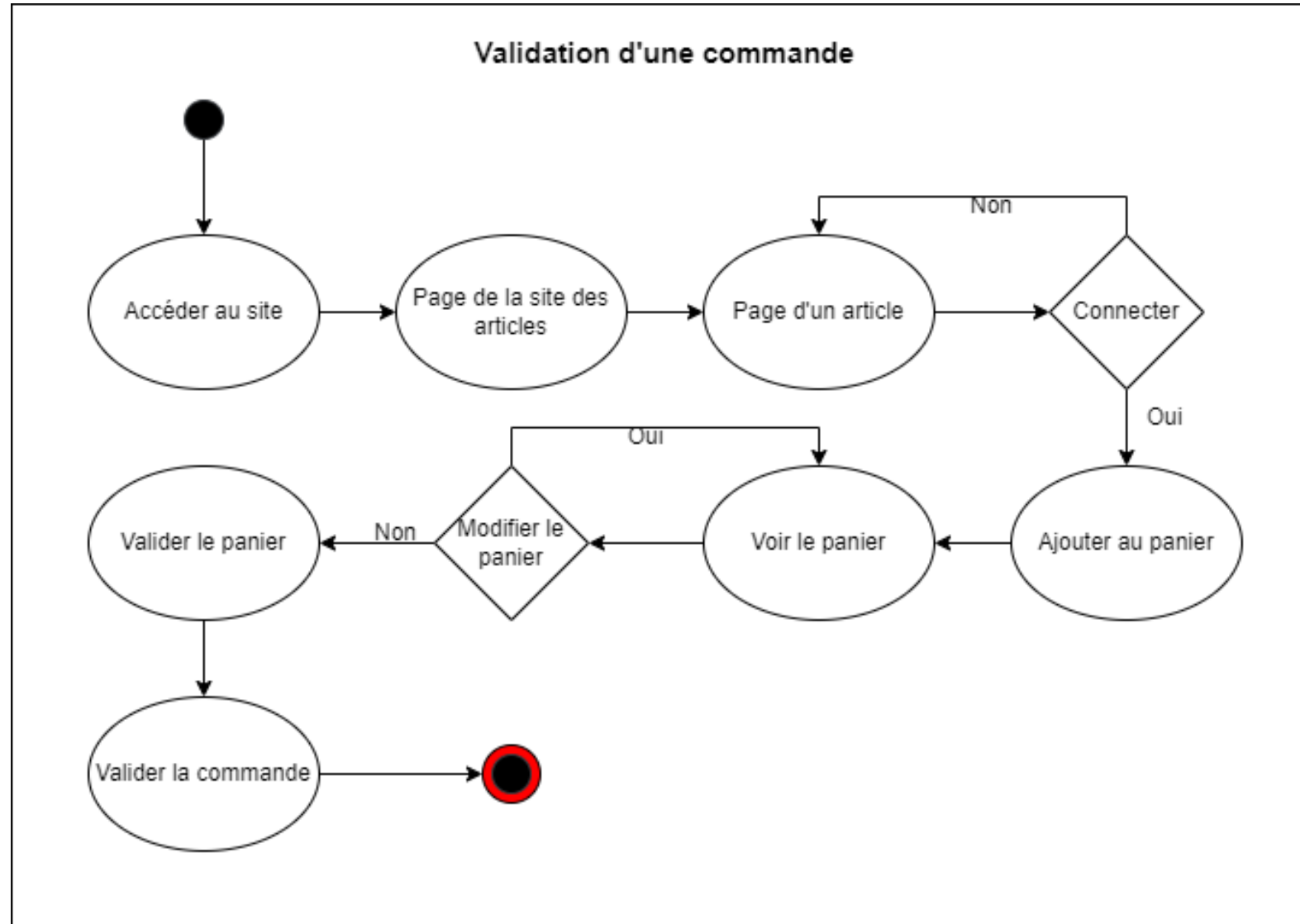
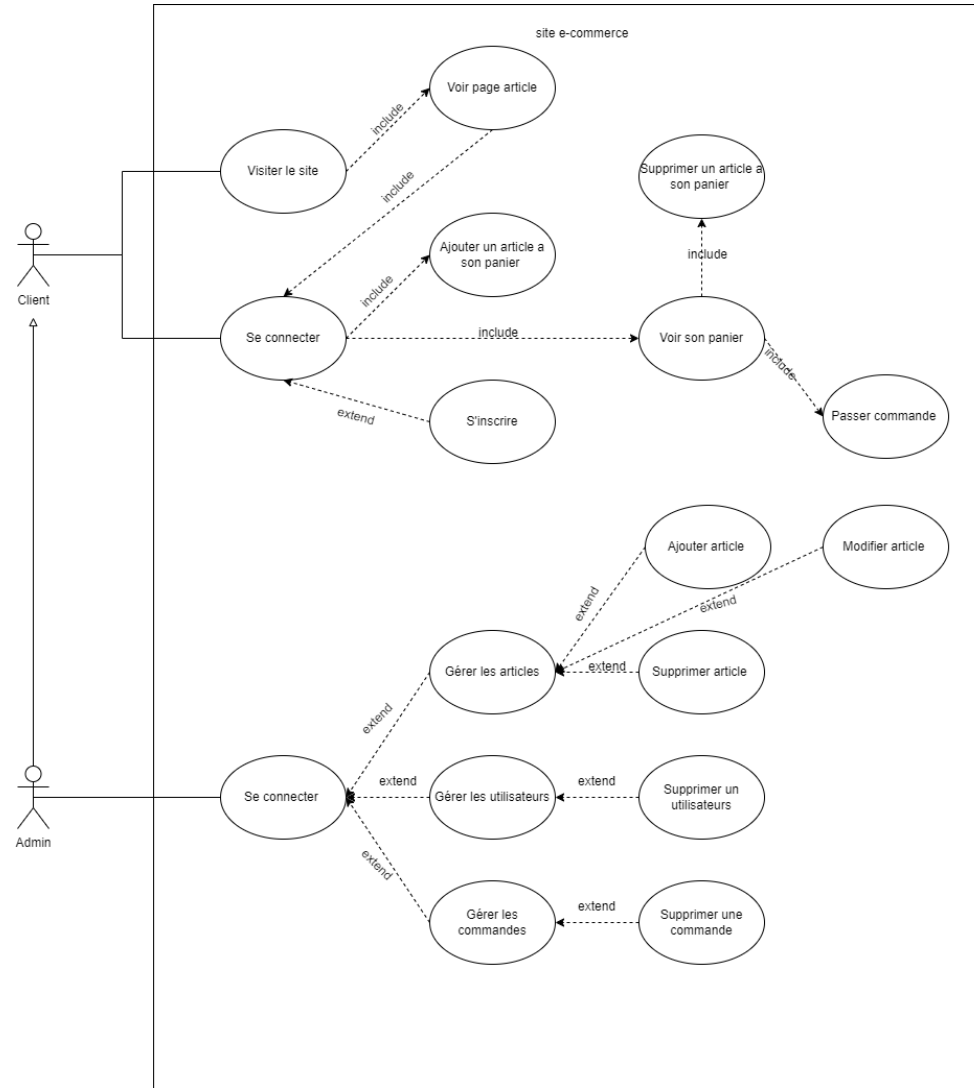


Diagramme Use Case

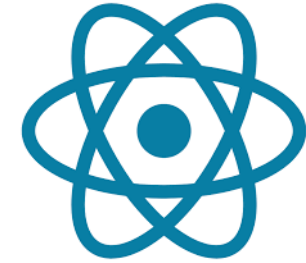
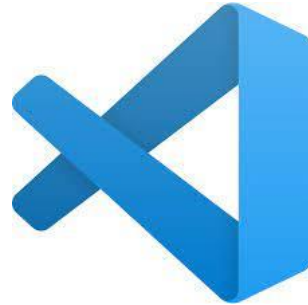


Triangle du projet

Objectifs du projet		
Objectifs		Mesure de l'atteinte de l'objectif
Contenu et qualité	Mise en place d'une interface utilisateur intuitive et esthétique. Offrir un catalogue de produits détaillé et attrayant avec des descriptions précises.	Analyse des retours clients, taux de retour de marchandises, nombre de réclamations ou problèmes rapportés.
Coût	Respecter le budget alloué de (7 500€) pour le développement et la mise en place du site.	Suivi budgétaire, comparaison coûts prévus vs coûts réels.
Délai	Lancer le site e-commerce d'ici fin janvier.	Comparaison de la date de lancement prévue vs la date de lancement réelle.

Les outils utilisés

- Visual Studio Code
- React.js
- Node.js
- Bases de données
- MySQL



Visual Studio Code

- Description :
 - Éditeur de code open source de Microsoft.
 - Offre une variété d'extensions pour prendre en charge presque tous les langages et frameworks.
- Caractéristiques clés :
 - Intégration Git native.
 - Debugging intégré.
 - Terminal intégré.
 - Live Share pour la collaboration en temps réel.
- Sources :
 - [Notes de version de VSCode]
 - [VSCode sur GitHub]

React.js

- Description :
 - React est une bibliothèque JavaScript pour la construction d'interfaces utilisateur. Il introduit le concept de "composants" pour diviser et organiser le code de l'interface utilisateur.
 - Son architecture "unidirectionnelle" avec l'utilisation de l'état et des props permet une meilleure prédictibilité du comportement de l'application.
- Avantages :
 - Rendu efficace grâce à la virtual DOM.
 - Large écosystème et grande communauté.
 - Flexibilité avec différents back-ends et architectures.
- Évolutions récentes :
 - Introduction des Hooks (useState, useEffect, etc.) qui simplifient la gestion de l'état et d'autres fonctionnalités des classes.
 - Suspense et concurrent mode pour améliorer le rendu et la performance.
- Sources :
 - [React.js Documentation]
 - [React Conférences et mises à jour]

Node.js

- Description :
 - **Environnement d'exécution JavaScript côté serveur :** Node.js utilise le moteur JavaScript V8 de Google Chrome pour exécuter le code JavaScript. Cela permet aux développeurs d'utiliser JavaScript pour écrire des scripts côté serveur, gérant ainsi des tâches telles que la connexion à des bases de données, la gestion de fichiers sur le serveur, ou la communication en temps réel.
 - **Non-bloquant et orienté événements :** Node.js fonctionne sur un modèle non-bloquant et orienté événements, ce qui le rend particulièrement efficace pour construire des applications nécessitant une performance en temps réel ou une haute concurrence, comme les jeux en ligne, les chatbots, ou les services de streaming. Il utilise des appels asynchrones et des opérations d'entrée/sortie (I/O) qui évitent les temps d'attente inutiles, optimisant ainsi le traitement des requêtes simultanées.
- Avantages :
 - **Performance Élevée pour les Applications en Temps Réel :** Grâce à son architecture non bloquante et événementielle, Node.js est capable de gérer un grand nombre de connexions simultanées avec une performance élevée. Cela le rend idéal pour les applications en temps réel qui nécessitent une communication rapide et bidirectionnelle entre le client et le serveur, comme les jeux en ligne, les chats en direct, et les systèmes de trading en bourse.
 - **Écosystème Riche avec npm :** Node.js bénéficie d'un vaste écosystème d'outils et de bibliothèques grâce à npm (Node Package Manager), le plus grand registre de logiciels au monde. Cela permet aux développeurs d'accéder facilement à un large éventail de modules et de paquets pour étendre les fonctionnalités de leurs applications, réduisant ainsi le temps de développement et la nécessité de réinventer la roue.
 - **JavaScript Partout :** En utilisant JavaScript tant pour le client que pour le serveur, Node.js permet aux développeurs de réutiliser le code entre le frontend et le backend, ce qui réduit les incohérences et les erreurs potentielles. Cette uniformité du langage de programmation simplifie le développement d'applications Web et réduit la courbe d'apprentissage pour les développeurs qui sont déjà familiarisés avec JavaScript.
- Évolutions récentes :
 - **Introduction des Async/Await :** Node.js a intégré les mots-clés `async` et `await` pour une écriture plus lisible et efficace des opérations asynchrones, simplifiant la gestion des promesses et des callbacks.
 - **Support de ES Modules :** Node.js a commencé à supporter nativement les ES Modules (ECMAScript Modules), permettant aux développeurs d'utiliser la syntaxe d'import/export standard d'ES6 pour organiser et charger leurs modules JavaScript.
- Sources :
 - **Documentation Officielle de Node.js :** La source principale et la plus fiable pour les informations sur Node.js est sa documentation officielle, qui est régulièrement mise à jour avec les dernières évolutions et pratiques recommandées.
 - **GitHub de Node.js :** Le référentiel GitHub de Node.js est une source précieuse où les développeurs peuvent trouver les derniers changements, contributions, et discussions sur les futures évolutions de la plateforme.

Bases de données

- Description :
 - Stockage organisé et structuré d'informations pour une récupération, une modification et une gestion efficaces.
- Types courants :
 - Relationnelles (SQL): MySQL, PostgreSQL, Oracle, SQLite.
 - NoSQL: MongoDB, Cassandra, Redis.
- Tendances :
 - La montée des bases de données en mémoire pour des performances ultra-rapides.
 - L'importance croissante des bases de données orientées graphes, comme Neo4j.
- Sources :
 - [DB-Engines Ranking]

MySQL

- Description :
 - SGBD relationnel, utilisé dans de nombreuses applications web.
- Caractéristiques:
 - Stockage basé sur des tables.
 - Utilise le langage SQL pour la manipulation de données.
 - Supporte des procédures stockées, des triggers, des vues, etc.
- Évolutions récentes :
 - Améliorations de sécurité.
 - Fonctionnalités de réplication et de clustering pour la haute disponibilité.
- Sources :
 - [MySQL Documentation]