133 – Développer des applications WEB

Documentation

Version 1 du 31.03.2025

Créé le 24.03.2025

Borgeat Nathan, Schmitt Maxime

|  |  |
| --- | --- |
| Une image contenant texte  Description générée automatiquement | Module du 17.03.2025 au 15.04.2025 |

[1 Introduction et contexte du project 5](#_Toc194399178)

[2 Analyse 6](#_Toc194399179)

[2.1 Use case client et use case Rest 6](#_Toc194399180)

[2.1.1 Use case user 6](#_Toc194399181)

[2.1.2 Use case admin 7](#_Toc194399182)

[2.1.3 Use case APIGateway 7](#_Toc194399183)

[2.1.4 Use case Rest user 9](#_Toc194399184)

[2.1.5 Use case Rest admin 11](#_Toc194399185)

[2.2 Diagrammes d’activité 11](#_Toc194399186)

[2.2.1 Diagramme d’activité user 11](#_Toc194399187)

[2.2.2 Diagramme d’activité admin 12](#_Toc194399188)

[2.3 Maquettes du projet 12](#_Toc194399189)

[2.3.1 Maquette user 12](#_Toc194399190)

[2.3.2 Maquette admin 13](#_Toc194399191)

[2.4 Diagrammes de séquence système 13](#_Toc194399192)

[2.4.1 Diagramme de séquence système user 13](#_Toc194399193)

[2.4.2 Diagramme de séquence système admin 15](#_Toc194399194)

[3 Conception 16](#_Toc194399195)

[3.1 Diagrammes de classes 16](#_Toc194399196)

[3.1.1 Diagramme de classes user 16](#_Toc194399197)

[3.1.2 Diagramme de classes admin 16](#_Toc194399198)

[3.1.3 Diagramme de classes gateway 17](#_Toc194399199)

[3.1.4 Diagramme de classes Rest user 17](#_Toc194399200)

[3.1.5 Diagramme de classes Rest admin 18](#_Toc194399201)

[3.2 Diagrammes de navigation 19](#_Toc194399202)

[3.2.1 Diagramme de navigation user 19](#_Toc194399203)

[3.2.2 Diagramme de navigation admin 20](#_Toc194399204)

[4 Bases de données 21](#_Toc194399205)

[4.1 Schémas ER 21](#_Toc194399206)

[4.1.1 Schéma ER global 21](#_Toc194399207)

[4.2 Schémas relationnels (MySQL) 21](#_Toc194399208)

[4.2.1 Schéma relationnel user 21](#_Toc194399209)

[4.2.2 Schéma relationnel admin 21](#_Toc194399210)

[5 Implémentation des applications <User> et <Admin> 22](#_Toc194399211)

[5.1 Une descente de code user 22](#_Toc194399212)

[5.2 Une descente de code admin 22](#_Toc194399213)

[6 Implémentation de l'application <API Gateway> 23](#_Toc194399214)

[6.1 Une descente de code APIGateway 23](#_Toc194399215)

[7 Implémentation des applications <REST user> et <REST admin> 24](#_Toc194399216)

[7.1 Une descente de code de l'API REST user 24](#_Toc194399217)

[7.2 Une descente de code de l'API REST admin 24](#_Toc194399218)

[8 Hébergement 25](#_Toc194399219)

[9 Installation du projet complet avec les 5 applications 26](#_Toc194399220)

[10 Outils, langages (versions, définitions, installations) 27](#_Toc194399221)

[11 Tests de fonctionnement du projet 28](#_Toc194399222)

[12 Auto-évaluations et conclusions 29](#_Toc194399223)

[12.1 Auto-évaluation et conclusion de Nathan Borgeat 29](#_Toc194399224)

[12.2 Auto-évaluation et conclusion de Maxime Schmitt 29](#_Toc194399225)

# Introduction et contexte du project

Voici le contenu du module :

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

# Analyse

## Use case client et use case Rest

### Use case user

Voici le diagramme use case pour la partie utilisateur :

Une image contenant texte, capture d’écran, diagramme, cercle

Le contenu généré par l’IA peut être incorrect.

Voici un tableau résumant et expliquant les actions possibles :

|  |  |
| --- | --- |
| Acteur | Action |
| Visitor | Signin : Créer un nouveau compte |
| Login : Se connecter à un compte déjà existant |
| User | Logout : Se déconnecter du compte actif |
| Ajouter un hérisson : Ajouter un hérisson à l’utilisateur connecté |
| Modifier un hérisson : Modifier les caractéristiques d’un hérisson |
| Créer un voyage : Ajouter un nouveau voyage pour un hérisson |
| GetFusees : Récupérer toutes les fusées disponibles |
| GetVoyages : Récupérer tous les voyages de l’utilisateur |
| GetHerissons : Récupérer tous les hérissons de l’utilisateur |

### Use case admin

Une image contenant texte, capture d’écran, diagramme, cercle

Le contenu généré par l’IA peut être incorrect.

|  |  |
| --- | --- |
| **Acteur** | **Action** |
| **Visiteur** | Login : Se connecter à un compte déjà existant |
| **Admin** | Logout : Se déconnecter du compte actif |
| Afficher les fusées : récupère les fusées pour les afficher |
| Afficher les vols : récupère les vols pour les afficher |
| Démonter fusée : démonte une fusée |
| Modifier fusée : modifier une fusée |
| Créer fusée : crée une fusée |

### Use case APIGateway

Voici le diagramme use case de la partie Gateway.

Une image contenant diagramme, capture d’écran, texte, cercle

Le contenu généré par l’IA peut être incorrect.

Voici un tableau résumant et expliquant les actions possibles :

|  |  |
| --- | --- |
| **Acteur** | **Action** |
| **Client** | Signin : Créer un nouveau compte |
| Login : Se connecter à un compte déjà existant |
| Logout : Se déconnecter du compte actif |
| Ajouter un hérisson : Ajouter un hérisson à l’utilisateur connecté |
| Modifier un hérisson : Modifier les caractéristiques d’un hérisson |
| Créer un voyage : Ajouter un nouveau voyage pour un hérisson |
| GetFusees : Récupérer toutes les fusées disponibles |
| GetVoyages : Récupérer tous les voyages de l’utilisateur |
| GetHerissons : Récupérer tous les hérissons de l’utilisateur |
| **Admin** | GetVoyages : Récupérer tous les voyages de l’utilisateur |
| GetFusees : Récupérer toutes les fusées disponibles |
| Créer fusée : crée une fusée |
| Modifier fusée : modifier une fusée |
| Démonter fusée : démonte une fusée |

### Use case Rest user

Voici le diagramme des cas d’utilisation de l’API Rest client :

Une image contenant texte, capture d’écran, diagramme, ligne

Le contenu généré par l’IA peut être incorrect.

Voici un tableau résumant et expliquant les actions possibles :

|  |  |
| --- | --- |
| Acteur | Action |
| APIGateway | Ajouter un hérisson : Ajouter un hérisson à l’utilisateur connecté |
| Modifier un hérisson : Modifier les caractéristiques d’un hérisson |
| Créer un voyage : Ajouter un nouveau voyage pour un hérisson |
| Signin : Créer un nouveau compte |
| Login : Se connecter à un compte déjà existant |
| GetVoyages : Récupérer tous les voyages de l’utilisateur |
| GetHerissons : Récupérer tous les hérissons de l’utilisateur |

### Use case Rest admin

Une image contenant texte, capture d’écran, diagramme, cercle

Le contenu généré par l’IA peut être incorrect.

|  |  |
| --- | --- |
| Acteur | Action |
| APIGateway | Créer fusée : crée une fusée |
| Ejecter un hérisson : éjecte un hérisson de sa fusée |
| Afficher les fusées : affiche les fusées |
| Logout : déconnecte l’admin |

## Diagrammes d’activité

### Diagramme d’activité user

Voici un diagramme d’activité sur l’ajout d’un hérisson :

Une image contenant texte, capture d’écran, diagramme, nombre

Le contenu généré par l’IA peut être incorrect.

### Diagramme d’activité admin

Ce diagramme représente la création de fusée. Si la fusée que l’on veut créer n’existe pas et que ses données sont correctes, on envoie au Rest admin une requête de création de fusée, puis il nous renvoie la réussite ou l’échec de l’opération.

Une image contenant texte, diagramme, ligne, capture d’écran

Le contenu généré par l’IA peut être incorrect.

## Maquettes du projet

### Maquette user

Page d’accueil une fois connecté :

Une image contenant texte, capture d’écran, nombre, conception

Le contenu généré par l’IA peut être incorrect.

### Maquette admin

Page de contrôle des fusées et des vols sur la page admin.

Une image contenant capture d’écran, Rectangle, texte, diagramme

Le contenu généré par l’IA peut être incorrect.

## Diagrammes de séquence système

### Diagramme de séquence système user

Voici le diagramme de séquence système pour l’ajout d’un hérisson :

Une image contenant texte, capture d’écran, Rectangle, diagramme

Le contenu généré par l’IA peut être incorrect.

### Diagramme de séquence système admin

Diagramme de séquence sur la fonction « créer fusée ».

Une image contenant texte, capture d’écran, diagramme, Parallèle

Le contenu généré par l’IA peut être incorrect.

# Conception

## Diagrammes de classes

### Diagramme de classes user

Une image contenant texte, capture d’écran, diagramme, Rectangle

Le contenu généré par l’IA peut être incorrect.

### Diagramme de classes admin

La page de connexion renvoie directement à la page admin de gestion. Chaque fonctionnalité de la gestion passe par un contrôleur (fusée ou vol selon la fonction) qui renvoie tout à un worker, travaillant avec le rest admin (qui a lui-même accès à la DB).

Une image contenant texte, capture d’écran, diagramme, affichage

Le contenu généré par l’IA peut être incorrect.

### Diagramme de classes gateway

Une image contenant texte, capture d’écran, Rectangle, ligne

Le contenu généré par l’IA peut être incorrect.

### Diagramme de classes Rest user

Une image contenant texte, capture d’écran, diagramme, Rectangle

Le contenu généré par l’IA peut être incorrect.

### Diagramme de classes Rest admin

Une image contenant texte, capture d’écran, diagramme, Rectangle

Le contenu généré par l’IA peut être incorrect.

## Diagrammes de navigation

### Diagramme de navigation user

Une image contenant texte, capture d’écran, diagramme, Parallèle

Le contenu généré par l’IA peut être incorrect.

### Diagramme de navigation admin

Une image contenant texte, capture d’écran, affichage, logiciel

Le contenu généré par l’IA peut être incorrect.

# Bases de données

## Schémas ER

### Schéma ER global

Dans le schéma ci-dessous, nous pouvons voir nos deux bases de données.

La première contient des utilisateurs qui peuvent créer des hérissons et des voyages.

Ces voyages utilisent des fusées qui se trouvent dans la base de données « Admin ». Cette DB contient aussi des administrateurs.



## Schémas relationnels (MySQL)

### Schéma relationnel user

Ce schéma représente les relations plus précisément que le schéma relationnel vu au-dessus.

La particularité de ce schéma est la « FK\_Fusee » dans la table « T\_Voyage », cette table devant contenir la PK d’une fusée se situant dans la base de données administrateur.

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

### Schéma relationnel admin

Voici simplement le schéma relationnel admin :

Une image contenant texte, capture d’écran, Police, nombre

Le contenu généré par l’IA peut être incorrect.

# Implémentation des applications <User> et <Admin>

## Une descente de code user

## Une descente de code admin

# Implémentation de l'application <API Gateway>

## Une descente de code APIGateway

# Implémentation des applications <REST user> et <REST admin>

## Une descente de code de l'API REST user

## Une descente de code de l'API REST admin

# Hébergement

# Installation du projet complet avec les 5 applications

# Outils, langages (versions, définitions, installations)

# Tests de fonctionnement du projet

# Auto-évaluations et conclusions

## Auto-évaluation et conclusion de Nathan Borgeat

## Auto-évaluation et conclusion de Maxime Schmitt

C’était vraiment un module.