

Manuel d'installation

*Projet MATUI : un outil d'aide à la
décision de méthode pour recueillir des
données auprès d'utilisateurs.*

PAULINE TEOULLE
CORENTIN ROY
UNIVERSITÉ GRENOBLE-ALPES

Manuel d'installation

Projet MATUI: un outil d'aide à la décision de méthode pour recueillir des données auprès d'utilisateurs.

Identification du document

Référence du document :	MI
Version du document :	2.0
Date du document :	18/05/2021
Auteur(s) :	Pauline TEOULLE Corentin ROY

Vérification du document

Validé par :	Nadine MANDRAN Sophie DUPUY
Validé le :	15/06/2021
Soumis le :	11/06/2021
Type de diffusion :	Document électronique (.pdf)

Sommaire

Sommaire	4
1. Introduction	5
1.1. Objectifs et méthodes	5
1.2. Documents de référence	5
2. Guide de lecture	6
3. Concepts de base	6
4. Installation et paramétrage du serveur	6
5. Installation de l'application web	7
6. Paramétrage de l'application web	8
7. Installation des données	8
8. Autres informations	10
9. Annexes	10
10. Glossaire	13
11. Références	13
12. Index	13

1. Introduction

L'objectif du document est de connaître les procédures à mettre en place pour installer le serveur ainsi que l'application web développée sous PHP et React.

Toutes les procédures (serveur et application) ainsi que les paramètres à prendre en compte sont décrits dans ce document.

1.1. Objectifs et méthodes

L'objectif du projet est de développer une application web afin d'aider les chercheurs en informatique à choisir une méthode ou un outil qui les aidera à récolter des informations auprès des utilisateurs. L'objectif est de faciliter l'utilisation du diagramme MATUI et de le diffuser plus largement.

Le projet doit informatiser le diagramme MATUI sous deux aspects :

- Éditer le diagramme (Une personne peut ajouter ou supprimer une partie , une association de documents à une méthode ou un outil, etc ...)
- Utiliser le diagramme (Une personne peut l'utiliser pour accéder au résultat d'une méthode particulière et aux documents aidant à l'application de celle-ci).

Le produit du projet est une application web permettant d'une part, de modifier le diagramme et d'autre part, de l'utiliser. L'application doit permettre de supporter trois rôles distincts.

Les personnes ayant le rôle "utilisateur" auront accès à une interface afin de répondre à des questions correspondant aux alternatives de MATUI.

A la fin du questionnaire, ils obtiendront le résultat d'une méthode (ou d'un outil) à utiliser et des documents supplémentaires relatifs au résultat afin de les aider dans l'application de celle-ci.

Les personnes ayant le rôle "administrateur" auront accès à une interface de modification de MATUI. Un administrateur pourra donc ajouter des règles de décision, des liens entre les règles, des conditions, les supprimer du diagramme, les éditer...Il sera aussi capable d'ajouter des documents à chacune des méthodes ou outils présents.

Les personnes ayant le rôle "super-administrateur" auront accès à la même interface que les administrateurs. Mais ils pourront également gérer les comptes des membres (suppression et modification de rôle). Par exemple, un super-administrateur aura la possibilité de changer une personne "utilisateur" en "administrateur". Si un membre pose problème, il pourra supprimer son compte.

1.2. Documents de référence

Ce présent document a été élaboré grâce aux documents tels que le cahier des charges, le plan de développement.

2. Guide de lecture

Le personnel technique de la maîtrise d'œuvre doit avoir accès à l'ensemble du document pour pouvoir installer et déployer correctement l'application.

Les responsables de la maîtrise d'ouvrage sont Nadine MANDRAN et Sophie DUPUY. Elles doivent avoir accès à l'ensemble du document.

Les personnes voulant installer ou réinstaller l'application sur un serveur ont besoin d'avoir accès à l'ensemble du document également: principalement les parties "Installation et paramétrage du serveur", "Installation de l'application", "Paramétrage de l'application" et "Installation des données".

3. Concepts de base

En informatique centrée humain, les chercheurs en informatique doivent connaître les besoins des futurs utilisateurs pour proposer une solution correspondant à leurs attentes (interface adaptée, fonctionnalités, etc...).

Ils doivent donc inclure l'humain en prenant en compte leur travail, leur environnement et doivent comprendre leurs besoins. Pour cela, ils doivent mobiliser des méthodes de production des données issues des Sciences Humaines Sociales (SHS) (entretiens, questionnaires). Or le plus souvent, ils manquent de formations pour choisir ces méthodes et les utiliser. Ils rencontrent donc des difficultés pour connaître les besoins des utilisateurs.

Afin de faciliter leur choix, Nadine MANDRAN et Sophie DUPUY qui travaillent au LIG (le Laboratoire d'Informatique de Grenoble) ont proposé un arbre de décision en 2018 qui s'appelle le diagramme MATUI et qui est disponible dans un article (Mandran & Dupuy-Chessa, 2018) situé en référence.

(http://thedre.imag.fr/wp-content/uploads/2019/08/20170208_THEDRE_MATUI.pdf)

C'est un logigramme (un schéma) qui représente un processus, un système ou un algorithme informatique. Il est utilisé pour documenter et partager des processus complexes sous la forme d'un diagramme clair et facile à comprendre. Il représente plusieurs parcours possibles qui amènent à des méthodes et outils à utiliser pour récolter des données pertinentes à leurs besoins. A partir de critères et de règles, il permet donc de trouver une méthode, ses conditions de mise en place et les outils pour arriver à récolter des données auprès d'utilisateurs.

4. Installation et paramétrage du serveur

En travaillant pour le laboratoire d'informatique de Grenoble, nous n'avons pas eu d'autre choix que de demander et d'utiliser un serveur Apache 2.4 géré par les services des moyens informatiques du laboratoire. En effet, pour des questions de sécurité, une demande de serveur a dû être effectuée par nos encadrantes pour préciser les différentes caractéristiques nécessaires au bon fonctionnement de notre application.

Il a donc fallu définir le projet en précisant les personnes responsables (Nadine MANDRAN et Sophie DUPUY-CHESSA), l'équipe de recherche (PIMLIG /EHCI) et les personnes utilisatrices en tant qu'administrateurs (maîtrise d'œuvre).

Des informations sur le site à héberger (telles que le nom d'URL, l'espace de stockage, les technologies souhaitées, le mode de mise à jour, une date de début de validité et une date de fin ou encore la durée de vie du site) ont aussi été communiquées aux personnes qui s'occupent des moyens informatiques afin de les aiguiller dans la configuration du serveur.

La demande est disponible en annexe. Les technologies à installer sur le serveur sont également mentionnées dans la demande de serveur (Annexe 1).

Nous avons dû demander à l'équipe informatique gérant le serveur d'activer le module PDO (du fichier php.ini) afin de pouvoir se connecter à la base de données et dû demander des droits d'accès supplémentaires à la base de données (SELECT par défaut) car nous avons besoin de faire des modifications sur celle-ci (INSERT, DELETE, UPDATE). Il faut également vérifier que le dossier dans lequel les documents des méthodes sont stockées est bien paramétré : accès en lecture, écriture et exécution pour les permissions publiques.

Les fichiers de l'application MATUI sont disponibles dans le dossier "matui" en suivant le chemin suivant : "/var/www/matui".

Les technologies installées sur le serveur sont les suivantes :

- packet debian 10 (php 7.3, mysql 8)
- nodejs 12
- npm 6

Le reste des librairies peuvent être installée directement lors du développement du projet avec les commandes npm :

- jquery 3.5.1
- react 17.0.1
- react-flow-renderer 8.7.0
- jwt

5. Installation de l'application web

Le serveur étant en accès distant, il est nécessaire de se connecter au VPN de l'Université de Grenoble pour garantir la sécurité des échanges depuis un accès extérieur et pour faire transiter des données professionnelles.

Une fois connecté au VPN de l'université, l'accès au serveur peut se faire de deux manières différentes : soit via une connexion SSH, soit via un client FileZilla qui gère plusieurs types de connexions (FTP, FTPS ou encore SFTP dans notre cas).

L'application ReactJS a besoin d'être construite pour pouvoir être déployée sur un serveur grâce à l'exécution de la commande NodeJS "npm run build" à la racine du projet. Cela crée un répertoire "build" dans lequel l'application est regroupée, minimisée et optimisée dans des fichiers statiques pour la production.

Le répertoire "/var/www/matui" du serveur est celui qui doit contenir les fichiers de l'application. Il faut donc déplacer les fichiers dedans pour pouvoir accéder au site via un navigateur.

Pour déployer l'application sur le serveur, la procédure d'installation est la suivante :

- Se connecter au VPN si besoin avec ses identifiants.
- Se connecter au serveur (SSH ou FileZilla) avec ses identifiants.
- Exécuter la commande "npm run build" à la racine du projet pour générer le dossier "build" à la racine du projet.
- Déplacer le contenu du dossier "build" de la racine du projet vers le serveur dans "/var/www/matui".
- Déplacer le dossier "API" de la racine du projet vers le serveur dans "/var/www/matui".
- Vérifier le site <http://lig-matui.imag.fr/>.

La première étape est de se connecter au VPN, la seconde est de se connecter au serveur, la troisième est d'exécuter la commande "npm run build" à la racine du projet développé afin de générer un répertoire "build". Enfin, la dernière étape est de déplacer le contenu du répertoire "build" et le contenu du répertoire "API" contenant nos fichiers PHP nécessaire pour l'accès à la base de données dans le répertoire "/var/www/matui". Le contenu du "build" doit être au même niveau que le dossier "API".

6. Paramétrage de l'application web

L'application étant divisée en deux parties (une partie API et une partie build), il faut faire attention à plusieurs choses : la connexion à la base de données au niveau de l'API PHP et les chemins utilisés par Axios dans la partie de développement React.

Lors de la connexion à la base de données, il faut renseigner l'hôte (host), le nom de la base de données (db_name), le nom de l'utilisateur de la base de données (username), et un mot de passe (password). Normalement, l'API et le build sont situés sur le même serveur donc l'hôte est "localhost".

Les logs pour se connecter à la base de données sont les suivants :

```
host = "localhost"
db_name = "api_matui"
username = "adminMatui"
password = "Pm2Mv1MDRT!"
```



```

public function getConnection()
{
    $this->connexion = null;
    try {
        $this->connexion = new PDO(
            'mysql:host='.$this->host .
            ';dbname=' . $this->db_name,
            $this->username,
            $this->password);
        $this->connexion->exec('set names utf8');
    } catch (Exception $exception) {
        echo "Erreur de connexion à la BDD : " . $exception->getMessage();
    }
    return $this->connexion;
}

```

Figure 1 : Fonction de connexion à la base de données via PDO.

Concernant l'accès aux fichiers de l'API par ReactJS, l'utilisation d'Axios requiert une adresse url pour aller chercher le fichier requis pour effectuer une action en base de données. La structure est donc la suivante : en premier lieu, le protocole et le hostname sont récupérés puis concaténés et lors de la requête axios, l'appel au bon fichier de l'API est effectué.

```

let protocol = window.location.protocol;
let host = window.location.hostname;
let url = protocol + '/' + host;
axios.get(url + '/API/Controllers/accueil/lire.php')

```

Figure 2 : Exemple d'appel à l'API via Axios pour la version de déploiement.

Attention, les chemins du projet en développement ne sont pas les mêmes que ceux nécessaires pour le déploiement. Il faut donc être attentif lors des modifications pour ne pas en oublier.

7. Installation des données

Notre application utilise une base de données MySQL compatible avec les paramètres du serveur. La base de données a été créée de façon locale sur les machines via phpMyAdmin et ensuite exportée en fichier .sql (disponible en annexe). Le fichier a ensuite été envoyé à un membre appartenant aux moyens informatiques afin qu'il puisse l'importer sur le serveur (pour une question de sécurité).

Les différentes tables utilisées au sein de la base de données, ainsi que leurs liaisons entre elles sont présentées en un modèle relationnel (Figure 3).

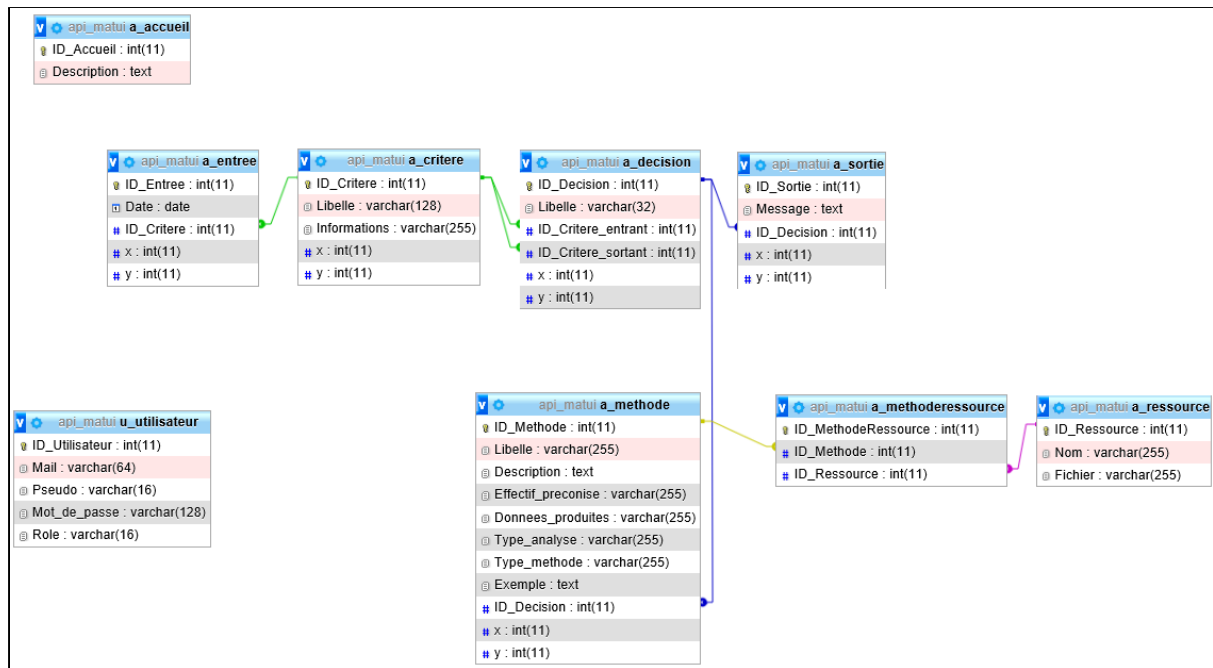


Figure 3: Modèle relationnel de la base de données api_matui

La base de données s'organise en plusieurs tables :

a_accueil: elle représente le texte de la page d'accueil et est définie par un identifiant "ID_Accueil" et une description "Description".

a_entree: elle représente le nœud de début du diagramme MATUI qui est défini par un identifiant "ID_Entree", une date d'enregistrement "Date", l'id d'un critère "ID_Critere", et une position "x" et "y".

a_sortie: elle représente le nœud de fin du diagramme MATUI qui est défini par un identifiant "ID_Sortie", un message "Message", l'id d'une décision "ID_Ddecision", et une position "x" et "y".

a_critere : elle représente les critères (questions) du diagramme MATUI qui sont définies par un identifiant "ID_Critere", un libelle "Libelle", éventuellement par une information "Informations", et une position "x" et "y".

a_decision: elle représente les liens entre les critères du diagramme MATUI qui sont définies par un identifiant "ID_Ddecision", un libelle "Libelle", l'id d'un critère entrant "ID_Critere_entrant", l'id d'un critère sortant "ID_Critere_sortant" (pouvant être null), et une position "x" et "y".

a_methode: elle représente les méthodes du diagramme MATUI qui sont définies par un identifiant "ID_Methode", un libelle "Libelle", une description "Description", l'effectif préconisé "Effectif_preconise", le type d'analyse "Type_analyse", le type de méthode "Type_methode", un exemple "Exemple", l'id d'une décision "ID_Ddecision", et une position "x" et "y".

a_methoderessource: elle représente les liens entre les ressources et les méthodes qui sont définis par un identifiant "ID_MethodeRessource", l'id d'une méthode "ID_Methode" et l'id d'une ressource "ID_Ressource".

a_ressource: elle représente les documents utiles aux méthodes et sont définis par un identifiant "ID_Ressource", un nom "Nom" et un fichier "Fichier".

u_utilisateur: elle représente les comptes utilisateurs qui sont définis par un identifiant "ID_Utilisateur", un mail "Mail", un pseudo "Pseudo", un mot de passe hashé "Mot_de_passe" et un rôle "Role".

Le schéma relationnel est le suivant:

a_entree (ID_Entree, Date, #ID_Critere, x, y)

a_critere (ID_Critere, Libelle, Informations, x, y)

a_decision (ID_Decision, Libelle, #ID_Critere_entrant, #ID_Critere_sortant)

a_methode (ID_Methode, Libelle, Description, Effectif_preconise, Donnees_produites, Type_analyse, Type_methode, Exemple, #ID_Decision, x, y)

a_ressource (ID_Ressource, Nom, Fichier)

a_methoderessource (ID_MethodeRessource, #ID_Methode, #ID_Ressource)

a_sortie (ID_Sortie, Message, #ID_Decision, x, y)

a_accueil (ID_Accueil, Description)

u_utilisateur (ID_Utilisateur, Mail, Pseudo, Mot_de_passe, Role)

8. Autres informations

Pour mettre à jour le site web, il suffit de remplacer les fichiers sur le serveur en suivant la même procédure que l'installation (connexion au VPN, connexion au serveur, exécuter "npm run build" à la racine du projet mis à jour et enfin déplacer les fichiers dans le dossier "/var/www/matui".

Pour se connecter à la base de donnée via SSH, il faut d'abord se connecter au serveur puis la commande suivante doit être exécutée :

"mysql -u adminMatui -pPm2Mv1MDRT! -D api_matui".

9. Annexes



Service des Moyens Informatiques et des Systèmes d'Information (MISI)

Email : contact-mi.lig@imag.fr

Demande d'hébergement d'un site de projet

Merci de saisir les informations demandées.

Ces informations sont nécessaires à la bonne traçabilité des sites hébergés par le laboratoire.

Identification du projet

Nom du projet

MATUI

Descriptif

Développement d'un outil d'aide à la décision de choix des méthodes de productions des données en sciences humaines et sociales

Personne responsable du projet

Nom : Nadine Mandran /Sophie Dupuy-Chessa

Adresse mail : nadine.mandran@univ-grenoble-alpes.fr

Lien avec le laboratoire (équipe de recherche ou autre) : PIMLIG / EHCI

Personne responsable de la mise à jour du site si différente

Autres personnes utilisatrices du site en tant qu'administrateurs

Pauline Teoulle et Corentin Roy (Stagiaires LIG)

pauline.teoulle@etu.univ-grenoble-alpes.fr

corentin.roy@etu.univ-grenoble-alpes.fr

Lien avec le laboratoire (équipe de recherche ou autre) : Stagiaires PIMLIG/LIG

Informations sur le site à héberger

Nom d'URL souhaitée : matui.imag.fr

Nombre de pages approximatif ou dimensionnement de l'espace de stockage souhaité (MO ou GO) : 1 Go

Mettre une croix devant les configurations souhaitées :

Site web statique ☐ html ☐ php
ou technologie souhaitée dans le cas d'un site web dynamique (BDD proposée par défaut : MySql) :

Pour le développement :

- jquery 3.5.1
- react 17.0.1
- react-flow-renderer 8.7.0 (librairie)
- node 12.19.0
- npm 6.14.9
- php 7.4.9
- mySql 8.0.21

Pour les tests :

- Jest 26.x
- Mocha 8.3.2
- PHPUnit 9
- react testing library 11.2.
- enzyme 3.11

Mode de mise à jour : accès ssh pour l'administrateur du site (permet l'utilisation d'un client sftp de type FileZilla)

Date de début de validité du site souhaitée : mai 2021

Date de fin de validité du site (ou durée de vie) souhaitée (*) : 4 ans

(*) Merci d'indiquer une date ou une durée, toute demande pourra être renouvelée

Le demandeur et l'administrateur du site certifient avoir signé la charte d'usage de ressources informatiques et de service internet du CNRS.

Le demandeur et l'administrateur du site s'engagent à respecter les règles de mise à jour du système qui seront mises en place par le service informatique du laboratoire sur la machine fournie, ou s'engagent à tout mettre en œuvre pour assurer la sécurité de la machine sur laquelle sera hébergé le site.

Le service MISI du laboratoire se réserve le droit de fermer l'accès du site de l'extérieur en cas de compromission, et le cas échéant d'isoler le serveur du reste des réseaux du laboratoire si la sécurité dudit réseau se trouvait compromise.

Date de la demande : 25 mars 2021

Annexe 1 : Demande de serveur pour le projet MATUI.

10. Glossaire

LIG - Laboratoire d'Informatique de Grenoble

Diagramme MATUI - Diagramme produit par Nadine MANDRAN et Sophie DUPUY qui est un logigramme permettant d'avoir des méthodes et outils pertinents pour récolter des besoins utilisateurs.

Logigramme - Schémas qui représente un processus, un système ou un algorithme informatique. Utilisé dans de multiples domaines pour documenter, étudier, planifier, améliorer et faire partager des processus souvent complexes ainsi transposés dans des schémas clairs et faciles à comprendre. Ils utilisent des rectangles, des ellipses, des losanges et potentiellement de nombreuses autres formes pour définir le type d'étape, ainsi que des flèches de connexion pour définir le flux et la séquence.

VPN - Réseau privé virtuel ou réseau virtuel privé, plus communément abrégé en VPN (de l'anglais: Virtual Private Network), est un système permettant de créer un lien direct entre des ordinateurs distants, qui isole leurs échanges du reste du trafic se déroulant sur des réseaux de télécommunications publics.

THEDRE - La "recherche sur la conception d'expériences humaines traçables" est une approche qui se concentre sur la recherche en informatique centrée sur l'homme. Il aide les chercheurs à intégrer des méthodes de production de données issues des sciences humaines et sociales pour l'informatique. Nom du site sur lequel le projet sera disponible.

FrontEnd - Interface graphique visuelle d'un programme qui se sert du BackEnd pour récupérer certaines informations et les afficher.

BackEnd - Programme qui traite les informations avant de les envoyer dans la partie FrontEnd (qui se servira de les afficher).

Serveur - Dispositif informatique (matériel et logiciel) qui offre des services à un ou plusieurs clients.

PHP - Langage de programmation libre impératif orienté objet, utilisé pour produire des pages Web dynamiques, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale.

React - React (aussi appelé React.js ou ReactJS) est une bibliothèque JavaScript libre permettant de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page HTML à chaque changement d'état.

PDO - Interface pour accéder à une base de données depuis PHP.

php.ini - Fichier de configuration PHP.

SSH - Protocole de communication sécurisé.

FileZilla - Client FTP, FTPS et SFTP, développé sous la licence publique générale GNU.

FTP - Protocole de communication destiné au partage de fichiers sur un réseau TCP/IP.

FTPS - Variante du FTP.

SFTP - Protocole de communication fonctionnant au-dessus de SSH pour transférer et gérer des fichiers à distance.

Node - Environnement bas niveau permettant l'exécution de JavaScript côté serveur.

API - Ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels.

Axios - Bibliothèque Javascript permettant d'effectuer des requêtes.

MySQL - Système de gestion de bases de données relationnelles (SGBDR).

Apache - Serveur HTTP créé et maintenu par la fondation Apache.

PhpMyAdmin - Application Web de gestion pour les systèmes de gestion de base de données MySQL et MariaDB, réalisée principalement en PHP et distribuée sous licence GNU GPL.

11. Références

http://thedre.imag.fr/wp-content/uploads/2019/08/20170208_THEDRE_MATUI.pdf

Nadine Mandran, Sophie Dupuy-Chessa. Supporting experimental methods in Information System research. 12th IEEE International Conference on Research Challenges in Information Science RCIS'2018, May 2018, Nantes, France. fhal-01903302

Le diagramme de MATUI est également référencé sur le site THEDRE : <http://thedre.imag.fr/wp-content/uploads/2019/06/logigrammeTHEDRE.pdf> (visité en 2021)

12. Index

Manuel d'installation - page 1
Procédure - Partie Introduction - page 5
Diagramme MATUI - Partie Objectifs et méthodes - page 5
LIG - Partie Concept de base- page 6
Logigramme - Partie Concept de base - page 6
Serveur Apache - Partie Installation et paramétrage du serveur - page 6
Sécurité - Partie Installation et paramétrage du serveur - page 6
Configuration serveur - Partie Installation et paramétrage du serveur - page 7
Base de données - Partie Installation des données - page 7
Permissions - Partie Installation et paramétrage du serveur - page 7
VPN - Partie Installation de l'application web - page 7
Connexion SSH - Partie Installation de l'application web - page 7
FileZilla - Partie Installation de l'application web - page 7
Commande Node JS - Partie Installation de l'application web - page 7
Build - Partie Installation de l'application web - page 7
Répertoire - Partie Installation de l'application web - page 7
Déployer - Partie Installation de l'application web - page 7
API - Partie Installation de l'application web - page 7
Structure - Partie Paramétrage de l'application web - page 8
Protocole - Partie Paramétrage de l'application web - page 8
mySQL - Partie Installation des données - page 8
Importer sur le serveur - Partie Installation des données - page 8
Schéma relationnel - Partie Installation des données - page 10
Mise à jour - Partie Autres informations - page 10
Déplacer - Partie Autres informations - page 10