GIROUD Antoine

BOIDIN Benoît

Projet de Programmation Web

MIASHS L3

Table des matières

[II. Introduction 3](#_Toc71986387)

[III. Architecture du site 3](#_Toc71986388)

[IV. Schéma conceptuel de la base de données 5](#_Toc71986389)

[A. Modèle conceptuel 5](#_Toc71986390)

[B. Modèle logique 5](#_Toc71986391)

[V. Méthodes et algorithmes significatifs 5](#_Toc71986392)

[A. Requêtes afficher. 6](#_Toc71986393)

[B. Ajout d’un site (avec l’image etc…). 6](#_Toc71986394)

[C. Connexion. 6](#_Toc71986395)

[D. Inscription. 6](#_Toc71986396)

[E. Affichage du template des commentaires avec les boutons « supprimer » et « afficher ». 6](#_Toc71986397)

[VI. Conclusion 6](#_Toc71986398)

[A. Problèmes rencontrés. 6](#_Toc71986399)

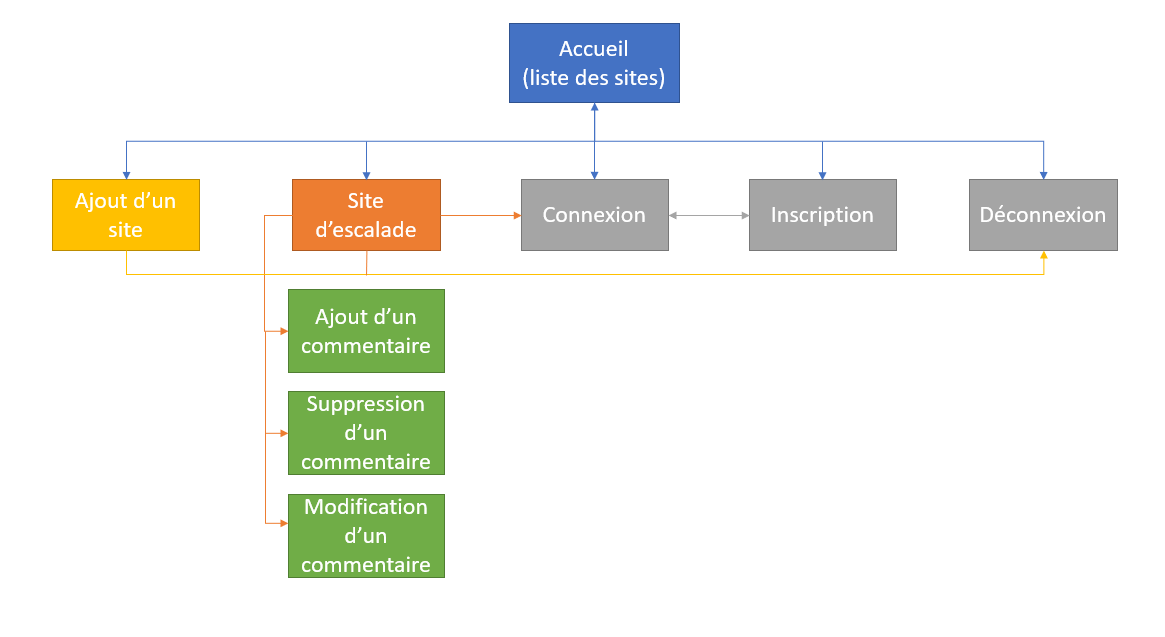
[B. Bilan. 6](#_Toc71986400)

# Introduction

Notre première réflexion lors de ce projet était de créer un site web qui pourrait nous être utile même après le rendu final. Étant tous deux passionnés d’escalade, nous avons alors eu l’idée de créer un site web répertoriant les sites d’escalade présents aux alentours de Lyon. En effet, lors de l’organisation de sortie d’escalade, il est souvent difficile de repérer le site idéal et de s’accorder sur celui-là. Notre site web regroupant la majorité des salles et falaises aux alentours, cela simplifie alors grandement la recherche. De plus, il est possible de visionner les détails de ces lieux tels que leur localisation, la difficulté des voies ainsi que leur nombre, et enfin le type d’escalade qui permet de connaitre le matériel à emporter.

Les utilisateurs ont la possibilité de rajouter des sites dans le cas où certains ont été oubliés. De plus, un espace commentaire est dédié à chacun d’eux. De ce fait, les utilisateurs peuvent donner leur avis sur les voies qu’ils ont grimpé afin d’indiquer par exemple si le niveau indiqué est correct, si le lieu est agréable ou encore d’organiser des sorties à l’avance avec les autres utilisateurs.

# Architecture du site



Le graphique ci-dessus représente l’architecture de notre site web, avec les différentes parties étant séparées sous forme de cluster. On peut donc observer dans un premier temps l’accueil (en bleu), comprenant la liste de tous les sites d’escalades de notre base de données. Il est ensuite possible d’accéder à toutes les autres pages du site web à partir de cet accueil.

En jaune est représenté la page permettant d’ajouter un nouveau site d’escalade, au sein de laquelle nous pouvons donc retrouver un formulaire où il est possible de renseigner toutes les données nécessaires (soit le nom du site, le lieu, le niveau, le nombre et le type de voies, ainsi qu’une image).

En orange, on peut retrouver la page qui permet donc d’afficher toutes ces données. Nous pouvons donc cliquer sur un des sites affichés à l’accueil, où nous serons alors renvoyés sur la page de ce site. A partir de là, il sera possible pour un utilisateur connecté d’ajouter un commentaire, qu’il pourra par la suite supprimer ou modifier.

Ensuite, nous avons affiché en gris les pages connexion, inscription et déconnexion. Il est possible de se rendre sur ces pages à partir d’un menu qui est présent sur toutes les autres pages du site. Il est donc possible de se rendre sur la page de connexion, d’inscription ou de déconnexion à partir de l’accueil ou d’un site d’escalade. Pour l’ajout d’un site, il faut forcément être connecté, donc seule la page déconnexion est accessible d’ici.

------------------ voir pour la rédaction finale de cette partie -----------------------

Pendant toute la navigation, l’utilisateur a accès à un menu dynamique, qui s’adapte en fonction de la connexion pour proposer les options cohérentes, ainsi que le pied de page avec les mentions légales.

Pour l’affichage dynamique des pages, nous avons utilisé le même moteur de template que lors des séances de TD.

Nous avons utilisé, pour la forme de notre projet, un template CSS qui donne un aspect professionnel au site.

Le site comprend tout d’abord une page d’accueil qui regroupe tous les sites répertoriés dans une grille, avec leurs images : une requête permet de récupérer les noms et les photos pour les afficher grâce à un modèle.

Pour avoir plus de renseignements, l’utilisateur peut cliquer sur un site pour accéder à une page qui affiche des informations plus précises, ainsi que les commentaires des utilisateurs. Les informations sont, de la même manière que dans la page d’accueil, requises dans la base de données.

Les commentaires peuvent être ajoutés par les utilisateurs à conditions qu’ils soient connectés. La personne à l’origine d’un commentaire peut également le supprimer ou e modifier, depuis un formulaire en bas de la page d’informations.

Le cas échant, les personnes connectées peuvent également ajouter un site d’escalade à la base de données.

Deux pages permettent respectivement la connexion et l’inscription. L’utilisateur doit renseigner un pseudonyme et un mot de passe pour avoir accès aux fonctions disponibles aux membres.

# Schéma conceptuel de la base de données

## Modèle conceptuel

## Modèle logique

Site (Idsite, Nomsite, Localisation, Niveau, Nbvoies, Image, #Idtype)

Type (Idtype, Nomtype)

Message (Idmessage, Date, Contenu, #Idgrimpeur, #Idsite)

Grimpeur (Idgrimpeur, Login, Passwd)

# Méthodes et algorithmes significatifs

J’ai mis ce à quoi je pensais mais si tu trouves que ce n’est pas pertinent ou qu’il manque des choses n’hésite pas à modifier.

## Requêtes afficher.

* Connexion à la base de données.
* Préparation puis exécution de la requête.
* Association des noms et valeurs pour l’affichage du modèle, en fonction du template voulu.

## Ajout d’un site (avec l’image etc…).

* Déplacement du fichier original dans un dossier d’images.
* Insertion des informations et du nom de l’image dans la base.

## Connexion.

* Requête pour chercher les identifiants dans la base ;
* Si les identifiants sont bons, on les met dans les variables de session.
* Redirection vers le menu.

## Inscription.

* Requête pour ajouter les identifiants
* Appel à la classe connexion.

## Affichage du template des commentaires avec les boutons « supprimer » et « afficher ».

* Vérifier que l’utilisateur est connecté.
* Ajout / modification puis redirection vers la page d’infos.

# Conclusion

## Problèmes rencontrés.

* Conflit entre les commentaires du modèle de style et le moteur de template.
* Problème pour inclure le fichier des identifiants (connect.inc.php) dans les classes qui s’en servent.

## Bilan.

* Séparer le plus possible tous les langages pour plus de clarté.
* Faire en sorte d’utiliser le moins de lignes de code possible.