



Projet Web

Clara Balzano
Julien Calabrese
Louise Delrieu
Léo Genot
Elise Gondange

I) Présentation de l'équipe

Pour réaliser ce projet, nous avons décidé de nous impliquer à 5.
Nous venons chacun de parcours différents, les niveaux et expériences des uns et des autres en développement back-end sont donc disparates.

Notre objectif était que les élèves plus expérimentés mettent à profit leurs connaissances pour aider les élèves débutants, cela a permis à chacun de travailler de façon plus organisée et de gagner en expérience.

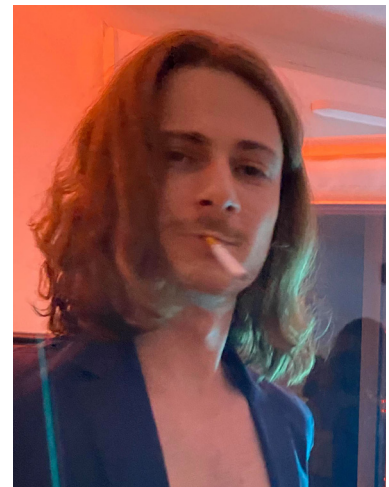


Léo Genot: Chef de projet, DUT informatique, niveau +++



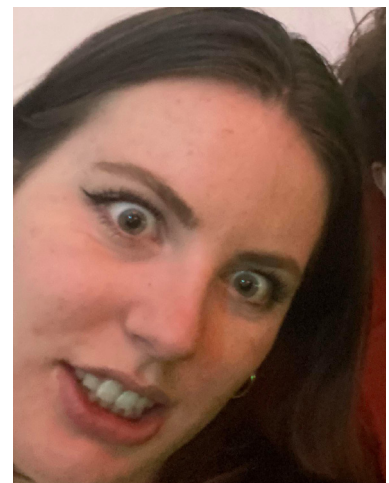
Clara Balzano: prépa, débutait au début de semestre.

Julien Calabrese: DUT MMI, niveau +



Louise Delrieu: prépa, débutait au début du semestre

Elise Gondange: DUT MM, niveau ++



II) Présentation du projet

Dans le cadre du cours de web du S2 à l'IMAC, nous devons créer un site web dynamique et respectant différentes contraintes.

Étant amateurs d'alcool et de jeux de soirées, nous nous sommes demandés comment concilier nos deux objectifs, à savoir répondre à notre projet de site web et boire un maximum tout en s'amusant. Nous possédions déjà quelques jeux de soirée basés sur un système de carte à piocher. Une idée nous est donc venue : pourquoi ne pas créer un site de jeu de soirée ? Sur ce site on pourrait "piocher" virtuellement des cartes qui nous donneraient une action à effectuer ou quelques gorgées de boisson alcoolisée (ou non pour Clara) à boire. Ce site nous permettrait de ne plus jamais oublier notre jeu de cartes à piocher puisqu'on pourrait l'avoir partout avec nous depuis notre smartphone.

Le site s'inscrivant dans le cadre de notre formation, son nom a très vite été trouvé : "Alcool'IMAC". À travers ce titre (et venant de la formation), on identifie très rapidement le fait que ce site web va être destiné aux étudiants (ou professeurs) de l'IMAC voulant s'amuser en soirée.

À travers cette réalisation, nous souhaitons permettre à des utilisateurs réguliers de jouer de manière simple et ludique, que ce soit depuis leur smartphone ou depuis leur ordinateur.

L'intérêt principal est d'avoir un site facile d'accès et d'utilisation pour donner envie aux utilisateurs de jouer depuis Alcool'IMAC.

Description et fonctionnement du site :

Pour commencer une partie, un joueur (le maître du jeu : celui qui a l'accès à l'ordinateur) se connecte pour accéder au jeu. Il va ensuite lui être demandé d'entrer les prénoms des autres joueurs de la partie.

Il est important d'avoir accès aux prénoms des joueurs de la partie car cela permettra par la suite d'avoir des cartes personnalisées prenant directement en compte la partie actuelle. (fonctionnalité non incluse dans notre projet mais dont il a été réflexion)

Une fois les joueurs entrés, le maître du jeu arrive sur l'accueil du site. Ici, plusieurs choix s'offrent à lui :

Pour commencer la partie, il peut directement se rendre sur l'onglet "Carte random" qui va afficher aléatoirement une carte présente dans le jeu. Après que le joueur concerné a réalisé l'action de la carte ou bu ses gorgées, le prochain joueur aura juste à cliquer sur le bouton "Changer de cartes" ou à cliquer de nouveau sur l'onglet "Carte random" pour afficher la carte suivante.

Les joueurs peuvent indiquer leurs cartes préférées à l'aide de la fonction de "Like"

(dépendante du maître du jeu connecté) présente au bas de chaque carte. Il peut également cliquer sur le bouton "J'ai bu" afin d'ajouter les gorgées qu'il a bues au compteur commun de gorgées de la partie.

Le maître du jeu peut également aller lui-même choisir les cartes qu'il va donner aux joueurs, pour cela, il peut se rendre dans l'onglet "Liste des cartes" dans lequel toutes les cartes du jeu sont répertoriées. De même que pour l'onglet carte random, chaque joueur pourra marquer ses cartes préférées à l'aide du like.

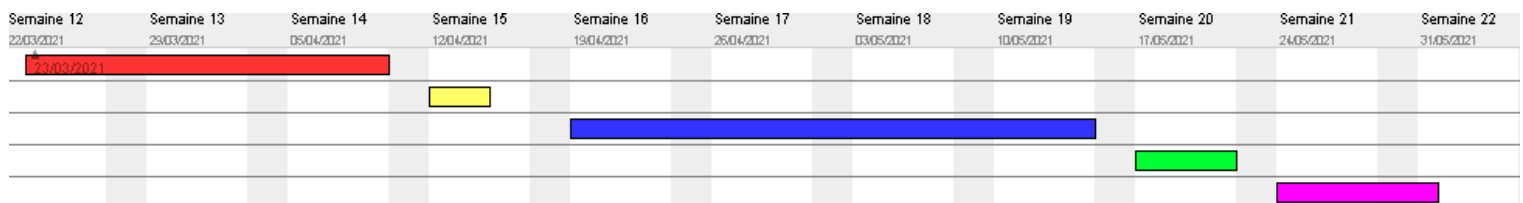
Une autre fonctionnalité présente sur cette page est celle de l'ajout de cartes. En effet, les premières cartes du jeu sont ajoutées par les modérateurs, mais leur imagination a des limites. Les joueurs peuvent donc créer des cartes eux-mêmes lorsqu'ils ont des idées à l'aide du formulaire présent tout en bas de la liste des cartes.

Une autre fonctionnalité présente sur le site est l'accès à différentes listes de boissons. Il nous est tous déjà arrivé de ne pas savoir quoi prendre à boire, ces listes permettront donc aux joueurs de s'inspirer et de découvrir de nouvelles saveurs. 3 listes de boissons différentes sont présentes : les cocktails, les alcools et les diluants (ou softs). Un cocktail est une boisson composée d'un alcool et d'un diluant. Le joueur aura donc accès dans l'onglet cocktails à une liste d'inspiration sur les meilleurs mélanges à tester. Les cocktails seront ajoutés directement par les joueurs, pour en créer un, il leur suffira de choisir un alcool, un diluant et d'inscrire le nom du créateur du cocktail. Toutefois, chaque alcool présent dans un cocktail devra tout d'abord être présent dans la liste des alcools (onglet alcools) et chaque diluant devra tout d'abord être présent dans la liste des diluants (onglet diluant). Pour créer un cocktail, le joueur aura peut être besoin d'aller d'abord créer son alcool et son diluant. La liste des cocktails sera accessible pour tous les joueurs de n'importe quelle partie.

Une fois sa partie terminée, il suffira au maître du jeu de se déconnecter de son compte grâce à l'onglet "Logout" pour fermer sa session.

III) Répartition du groupe / Planning

Nom	Date de début	Date de fin
• Réflexions préliminaires, choix du projet	23/03/2021	09/04/2021
• BDD: Merise et création sur serveur	12/04/2021	14/04/2021
• création du back-end	19/04/2021	14/05/2021
• Debugage et ajout de cartes de jeu à la BDD	17/05/2021	21/05/2021
• Ajout CSS et dernières modifications	24/05/2021	31/05/2021



Le but de notre groupe était de permettre aux moins expérimentés de prendre en main le projet pour gagner en expérience et compétence. Pour ce faire, nous avons créé à 5 (depuis discord) la structure du back-end afin d'être sûr que nous étions d'accord sur les bases, puis Clara et Louise ont été les premières à se lancer sur le projet. Julien, Elise et Léo étaient tout de même présents sur le projet pour suivre leurs avancées et les aider en cas de difficultés. Par la suite, le code s'est compliqué notamment lorsqu'il a été question de session, nous avons donc décidé de laisser Clara et Louise coder le projet tout en étant présents en vocal (grâce à discord) avec elles pour les guider dans les différentes étapes à effectuer. Cela leur a permis de coder elles-mêmes et de vraiment découvrir le back-end. Pour nous, il était plus intéressant de travailler dans ce sens là, même si c'était parfois un peu plus long que ça n'aurait pu l'être, que de les laisser regarder un étudiant confirmé dans la matière coder pendant des heures.

IV) Recherches graphiques

Nous ne nous sommes pas attardés sur la partie visuelle du site, c'est d'ailleurs une partie qui pourrait être grandement améliorée. Un logo a tout de même été rapidement créé car cela nous tenait à cœur de mettre une image sur notre projet. Pour l'ergonomie et la facilité d'utilisation du site, un CSS léger a été mis en place. Cela permet de rendre des textes plus ou moins importants et d'aligner le contenu du site (avec flexbox). Nous avons trouvé que c'était le minimum à faire pour avoir un site lisible.

Le but pour le futur est que le site soit facilement utilisable sur tous les supports. Il faut que les joueurs aient envie d'aller dessus et comprennent directement comment utiliser ce site. Pour qu'ils aient envie d'aller dessus, il fallait que le site soit agréable à regarder, sans forcément paraître magnifique. Un dégradé violet pastel que nous apprécions tout particulièrement a été utilisé en background, cela donne un aspect très tendance et moderne au site.



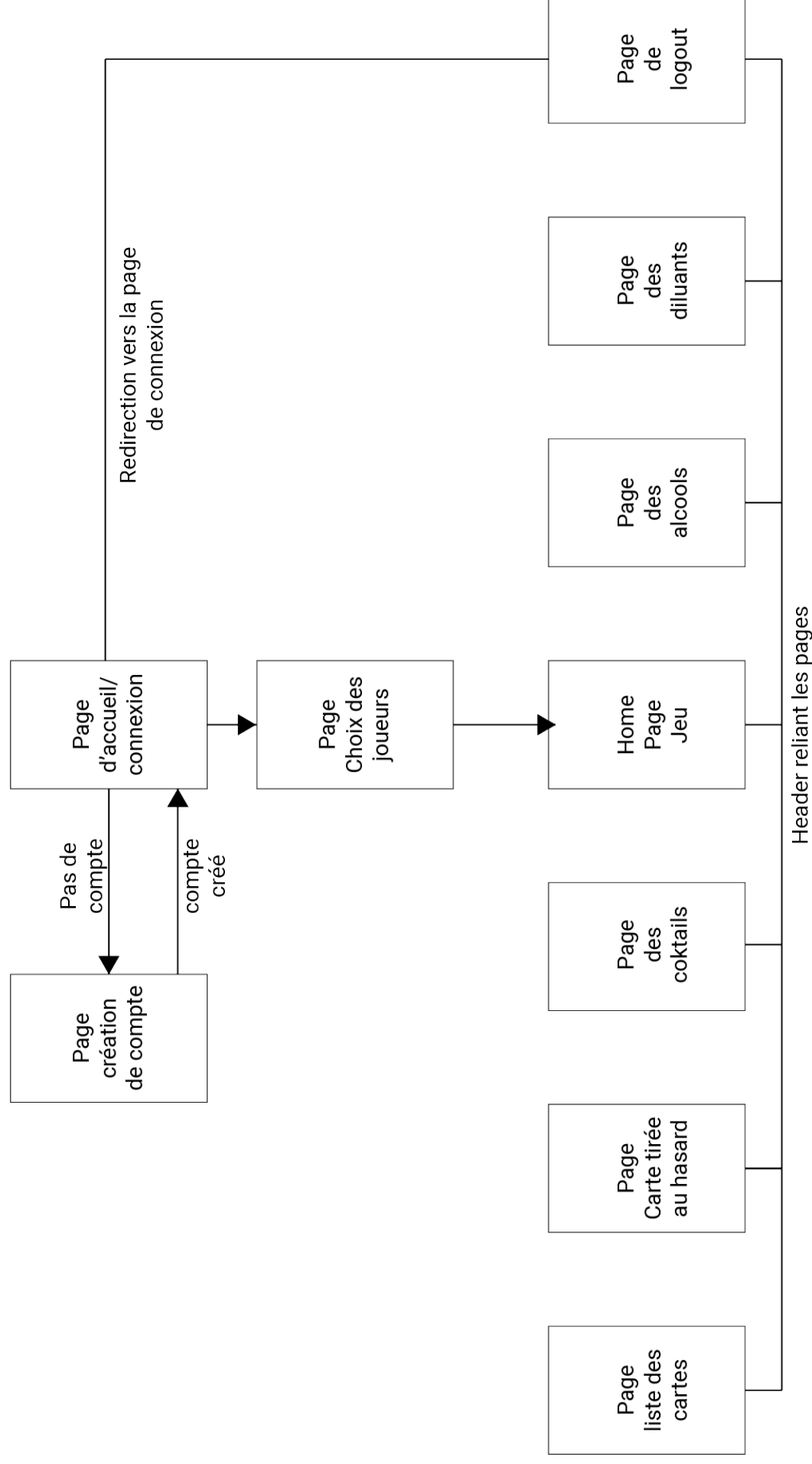
Visuel du site Alcool'IMAC



Logo du site Alcool'IMAC

V) Architecture et gabarit

A) Arborescence



B) Schéma de la BDD

alcool

Colonne	Type	Null	Valeur par défaut	Commentaires
id (<i>Primaire</i>)	int(11)	Non		
name	varchar(100)	Non		
degree	int(11)	Non		
dose	int(11)	Non	1	

Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	9	A	Non	

card

Colonne	Type	Null	Valeur par défaut	Commentaires
id (<i>Primaire</i>)	int(11)	Non		
date	timestamp	Non	CURRENT_TIMESTAMP	
content	text	Non		
shots	int(11)	Non		

Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	17	A	Non	

cocktails

Colonne	Type	Null	Valeur par défaut	Commentaires
id (<i>Primaire</i>)	int(11)	Non		
alcool_id	int(11)	Non		
diluant_id	int(11)	Non		
user	varchar(100)	Non		

Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	2	A	Non	
cocktails_alcool_foreign_key	BTREE	Non	Non	alcool_id	2	A	Non	
cocktails_diluant_foreign_key	BTREE	Non	Non	diluant_id	2	A	Non	

diluant

Colonne	Type	Null	Valeur par défaut	Commentaires
id (<i>Primaire</i>)	int(11)	Non		

name	varchar(100)	Non		
dose	int(11)	Non	1	

Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	8	A	Non	

likes

Colonne	Type	Null	Valeur par défaut	Commentaires
id (<i>Primaire</i>)	int(11)	Non		
user_id	int(11)	Non		
card_id	int(11)	Non		

Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	1	A	Non	
likes_card_foreign_key	BTREE	Non	Non	card_id	1	A	Non	
likes_user_foreign_key	BTREE	Non	Non	user_id	1	A	Non	

user

Colonne	Type	Null	Valeur par défaut	Commentaires
id (<i>Primaire</i>)	int(11)	Non		
username	varchar(100)	Non		
email	varchar(100)	Non		
password	varchar(100)	Non		

Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	0	A	Non	

user_temp

Colonne	Type	Null	Valeur par défaut	Commentaires
id (<i>Primaire</i>)	int(11)	Non		
user_id	int(11)	Non		
username	varchar(100)	Non		

Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	0	A	Non	
user_id	BTREE	Non	Non	user_id	0	A	Non	

<i>temp</i>
id
user_id
username

<i>user</i>
id
username
email
password

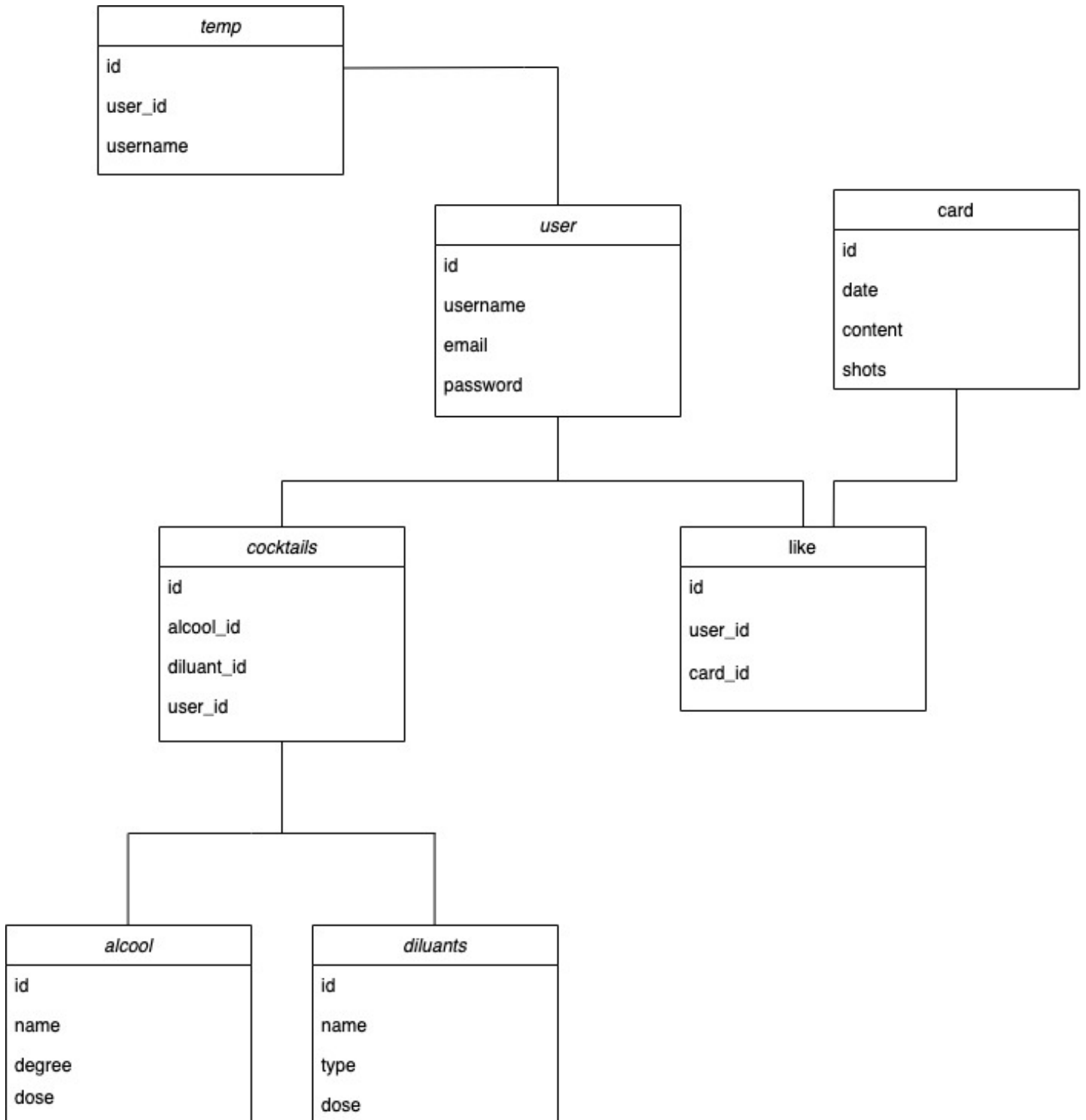
<i>card</i>
id
date
content
shots

<i>cocktails</i>
id
alcohol_id
diluant_id
user_id

<i>like</i>
id
user_id
card_id

<i>alcohol</i>
id
name
degree
dose

<i>diluants</i>
id
name
type
dose



VI) Tâches et fonctionnalités à implémenter (endpoints)

User

- Possibilité de se connecter avec une session
- Possibilité d'ajouter les joueurs à la partie
- Possibilité de se déconnecter de sa session

Cartes

- Possibilité d'afficher une liste carte
- Possibilité d'afficher une carte aléatoirement
- Possibilité d'ajouter une carte
- Possibilité d'enlever une carte

Cocktails

- Possibilité d'afficher une liste de cocktails
- Possibilité d'ajouter un cocktail
- Possibilité d'enlever un cocktail

Alcools

- Possibilité d'afficher une liste d'alcools
- Possibilité d'ajouter un alcool
- Possibilité d'enlever un alcool

Diluants

- Possibilité d'afficher une liste de diluants
- Possibilité d'ajouter un diluant
- Possibilité d'enlever un diluant

VII) Gestion de projet

Gestion du temps

Dès le lancement du projet, nous avons estimé le temps qu'il nous faudra par étapes et nous l'avons reporté sur un diagramme de Gantt (cf partie III), que nous avons essayé de respecter au maximum au cours des différentes semaines de travail. Cette organisation préalable nous a permis de nous fixer des deadlines pour ne pas être pris de cours le jour du rendu. De plus, le fait que des rendus intermédiaires soient mis en place par le professeur nous a forcé à avancer à un rythme suffisamment soutenu pour que les différentes étapes puissent être rendues aux bons moments. Ces dates intermédiaires ont d'ailleurs été prises en compte dans la création du diagramme de Gantt.

Organisation de travail

Comme expliqué plus tôt, nous avons fait le choix de créer l'arborescence du site et sa structure tous ensemble. Cela nous a permis de nous mettre tous d'accord et d'avoir une idée commune sur ce que nous devons faire.

Nous avons commencé par travailler sur la description de notre projet, et notamment de notre base de données. Clara et Louise étant novices en back-end mais ayant déjà de bonnes connaissances en SQL, il leur a été très rapide de comprendre le fonctionnement d'une base de données. Par soucis de facilité, nous avons décidé d'héberger notre site sur le serveur personnel d'Elise, nous avons donc pu directement créer la base de donnée depuis phpMyAdmin qui a un aspect visuel qui a facilité la compréhension de Clara et Louise, puis à la suite de cela nous avons créé le schéma de Merise de notre base.

Pour le commencement du code, les moins expérimentés ont pu lancer le projet et être aidés / débloqués par les plus expérimentés lorsque cela était nécessaire. De cette manière, les débutants ont progressé en faisant face à plusieurs problèmes globalement assez facilement identifiables. Les autres ont pu s'entraîner à résoudre les problèmes plus complexes qui se posaient, tout en expliquant leur méthode de résolution aux autres membres du groupe. Ainsi, chacun a pu suivre l'avancement du projet et y apporter sa touche, en suivant les cours et en résolvant des problèmes plus ou moins grands mais toujours à son niveau. Les débutants ont donc grandement approfondi leur expérience, les confirmés se sont confrontés aux plus grosses questions.

Une des difficultés rencontrée concerne les sessions d'utilisateurs. Étant donné que c'est une notion que Louise et Clara ne connaissaient pas, nous avons décidé de les guider mais de les laisser réaliser une grosse partie de celle-ci. Cela leur a évité de regarder quelqu'un d'expérimenter coder et leur a permis de bien comprendre cette notion.

Les élèves du groupe plus expérimentés ont tout de même eu leur rôle à jouer dans le code car parfois nous étions un peu à court de temps et nous avons besoin de réaliser des choses un peu compliquées de manière assez rapide. Cependant, dès que l'occasion se présentait, Clara et Louise n'ont pas hésité à poser des questions si elles ne comprenaient pas quelque chose, et nous faisons au mieux pour leur expliquer.

VIII) Rendu, satisfaction du projet, post mortem

Finalité du projet

Finalement, nous sommes globalement satisfaits du projet réalisé. Nous avons pu mettre en place la majorité de nos idées/envies. Cependant, il y a tout de même quelques fonctionnalités que nous n'avons pas pu mettre en place par manque de temps.

En effet, les échéances s'accumulent à la fin du semestre et nous avons été pris un peu de court quant aux autres projets que nous avions à rendre. C'est pour cela que plusieurs fonctionnalités auxquelles nous avions pensé n'ont pas pu être implémentées au projet.

Premièrement, comme expliqué précédemment, nous souhaitions mettre en place des cartes personnalisées, c'est d'ailleurs pour cela que nous avons créé une page récupérant le nom de tous les joueurs d'une partie. Ces cartes personnalisées auraient permis de créer une proximité entre le jeu et les joueurs et de les fidéliser plus rapidement.

Nous souhaitions aussi créer un système de statistiques. C'est-à-dire que nous aurions pu prendre en compte les noms de chaque joueur pour leur créer des stats de jeux personnels. Ces stats comprennent le nombre de gorgées que le joueur a bu par parties, le nombre de parties auxquelles il a joué...

Sur le même principe que les stats, un système d'estimation de taux d'alcoolémie aurait pu être mis en place dans un but "divertissant". En prenant en paramètre le poids et la taille d'un joueur, nous pourrions calculer son taux d'alcoolémie tout au long de la partie. Cela permettrait également de prévenir les joueurs trop éméchés pour prendre la route (tout en prenant en compte qu'il s'agit d'estimation).

Au sujet de l'organisation de projet, nous sommes tous assez satisfaits de nos méthodes de travail. Malgré le rush de fin de semestre, nous avons réussi à rendre un projet qui nous ressemble : divertissant tout en étant complet et respectant les contraintes imposées.

Avis personnels

Léo : Ce projet m'a permis d'affirmer mes compétences en back, la gestion du temps et des tâches était primordiale afin que tout le groupe puisse travailler dans les meilleures conditions possibles. Appliquer l'architecture MVC m'a aussi permis de retrouver une grande organisation dans ma manière de coder. Ce fut un projet difficile mais très agréable à réaliser en groupe.

Julien : A travers ce projet, j'ai pu replonger dans l'univers du back-end, auquel je ne m'étais pas confronté depuis plusieurs mois. J'ai pu retrouver mes compétences et même les améliorer, et je suis très satisfait des fonctionnalités et de l'apparence finale du projet.

Louise : Etant débutante en back-end ce projet était un véritable défi, au début j'étais un peu perdu et je n'osais pas me lancer mais les plus expérimentés m'ont aidé et ils ont été très pédagogue. Grâce à ce projet, j'ai progressé en back-end et j'ai pu acquérir des réflexes qui me serviront dans des futurs projets.

Clara : Ce projet a été l'occasion de mettre en application les choses vues en cours d'architecture logicielle, et ainsi de mieux comprendre certaines notions théoriques. Ce qui paraissait insurmontable a finalement été rendu possible grâce à l'esprit d'équipe qui a été un réel moteur de ce projet, et également la patience dont ont su faire preuve les plus expérimentés. Le passage en programmation orientée objet a rendu les choses plus claires, à condition de bien garder une vue d'ensemble.

Elise : Je suis très satisfaite du projet que nous avons réalisé. Ayant déjà de bonnes bases en back-end, j'ai pu mettre à profit mes connaissances et me mettre au défi de partager celle-ci avec les élèves qui débutent dans ce domaine. C'est toujours plus compliqué d'expliquer notre méthode de travail que d'avancer seul sur un projet, mais c'était très enrichissant et cela m'a donné l'opportunité de revoir des notions que je n'avais pas vu depuis longtemps.

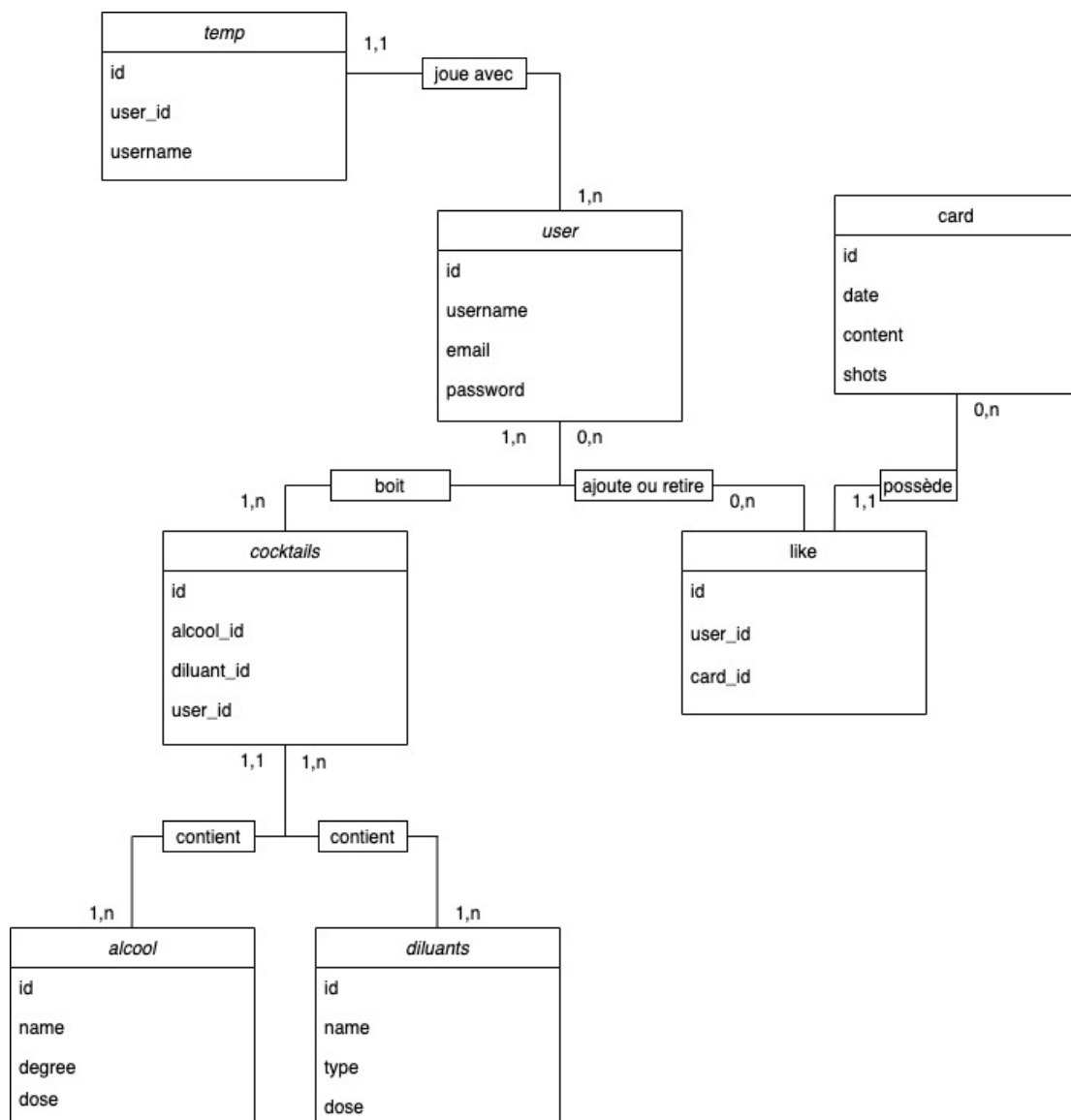
Annexes

Processus de création de la base de données:

Nous avons déjà une idée assez précise de notre projet donc la création des tables de notre base de données fut assez rapide car nous étions tous accordés sur ce que nous devons mettre en place.

Pour mettre en place notre projet nous avons décidé de commencer par travailler en local. Après avoir réalisé notre schéma Merise, nous avons créé localement sur phpMyAdmin notre base de données. Lorsqu'il a fallu passer sur le serveur, il nous a alors suffit d'importer notre base de données déjà créée dans l'interface mise en place avec le serveur. Évidemment il y a eu quelques bugs lors du passage sur serveur mais nous avons réussi à les résoudre assez rapidement.

Merise



Captures base de données depuis le serveur

IONOS

by I&I

phpMyAdmin

RECENTES

PRÉFÉRÉES

db51884372

Nouvelle table

alcohol

card

cocktails

diluant

likes

user

user_temp

← Serveur: db5002351159.hosting-data.io » Base de données: db51884372 » Table: card

Parcourir

Structure

SQL

Rechercher

Insérer

Exporter

Importer

Opérations

Déclencheurs

Structure de table

Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/>	1	id	int(11)		Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
<input type="checkbox"/>	2	date	timestamp		Non	CURRENT_TIMESTAMP			Modifier Supprimer Plus
<input type="checkbox"/>	3	content	text	utf8_general_ci	Non				Modifier Supprimer Plus
<input type="checkbox"/>	4	shots	int(11)		Non	Aucun(e)			Modifier Supprimer Plus

☐ Tout cocher Avec la sélection : [Parcourir](#) [Modifier](#) [Supprimer](#) [Primaire](#) [Unique](#) [Index](#) [Texte entier](#)

[Imprimer](#) [Suggérer des optimisations de structure](#) [Déplacer des colonnes](#) [Normaliser](#)

Ajouter

1

colonne(s)

après shots

Exécuter

Index

Action	Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
Éditer Supprimer	PRIMARY	BTREE	Oui	Non	id	17	A	Non	

Créer un index sur

1

colonnes

Exécuter

IONOS

by I&I

phpMyAdmin

RECENTES

PRÉFÉRÉES

db51884372

Nouvelle table

alcohol

card

cocktails

diluant

likes

user

user_temp

← Serveur: db5002351159.hosting-data.io » Base de données: db51884372

Structure

SQL

Rechercher

Requête

Exporter

Importer

Opérations

Procédures stockées

Déclencheurs

Filtres

Contenant le mot :

Table	Action	Lignes	Type	Interclassement	Taille	Perte
<input type="checkbox"/> alcohol	Parcourir Structure Rechercher Insérer Vider Supprimer	9	InnoDB	utf8_general_ci	16,0 kio	-
<input type="checkbox"/> card	Parcourir Structure Rechercher Insérer Vider Supprimer	17	InnoDB	utf8_general_ci	16,0 kio	-
<input type="checkbox"/> cocktails	Parcourir Structure Rechercher Insérer Vider Supprimer	2	InnoDB	utf8_general_ci	48,0 kio	-
<input type="checkbox"/> diluant	Parcourir Structure Rechercher Insérer Vider Supprimer	9	InnoDB	utf8_general_ci	16,0 kio	-
<input type="checkbox"/> likes	Parcourir Structure Rechercher Insérer Vider Supprimer	2	InnoDB	utf8_general_ci	48,0 kio	-
<input type="checkbox"/> user	Parcourir Structure Rechercher Insérer Vider Supprimer	7	InnoDB	utf8_general_ci	16,0 kio	-
<input type="checkbox"/> user_temp	Parcourir Structure Rechercher Insérer Vider Supprimer	15	InnoDB	utf8_general_ci	32,0 kio	-
7 tables	Somme	61	InnoDB	utf8_general_ci	192,0 kio	0 o

☐ Tout cocher Avec la sélection :

Endpoints

EndPoint	Description	Méthode	URL	Paramètres URL/Data	Réponses/erreurs
Pages/connexion	Récupérer les informations de connexion de l'utilisateur pour qu'il se connecte	POST	https://www.gondange.com/alcoolimac/projet/views/pages/connexionView.php	\$username \$email \$password (non passés dans l'URL car données perso)	Messages d'erreur (Tous les champs doivent être remplis)
Pages/register	Récupérer les informations d'inscription de l'utilisateur pour qu'il s'inscrive	POST	https://www.gondange.com/alcoolimac/projet/views/pages/registerView.php	\$username \$email \$password (non passés dans l'URL car données perso)	Messages d'erreur (Tous les champs doivent être remplis, l'email doit être au bon format et le mot de passe doit faire 6 caractères minimum)
Posts/add	Récupérer les données d'une carte pour l'ajouter à la liste des cartes	POST	https://www.gondange.com/alcoolimac/projet/index.php?controller=posts&action=index	\$content \$shots (pas par URL)	Message d'erreur
Posts/show	Récupérer un id aléatoire existant pour montrer la carte liée à cet id	GET	https://www.gondange.com/alcoolimac/projet/index.php?controller=posts&action=show&id=11	\$id (par URL)	Page d'erreur

Posts/remove	Récupérer l'id de la carte que l'utilisateur souhaite supprimer	GET	https://www.gondange.com/alcoolimac/projet/index.php?controller=posts&action=remove&id=11	\$id (par URL)	Page d'erreur
Alcool/add	Récupérer les données de l'alcool pour l'ajouter à la liste des alcools	POST	https://www.gondange.com/alcoolimac/projet/?controller=alcool&action=index	\$name \$degree \$dose (pas par URL)	Page d'erreur
Alcool/remove	Récupérer l'id de l'alcool qu'on souhaite supprimer	GET	https://www.gondange.com/alcoolimac/projet/?controller=alcool&action=remove&id=11	\$id (par URL)	Page d'erreur
Diluant/add	Récupérer les données du diluant pour l'ajouter à la liste des alcools	POST	https://www.gondange.com/alcoolimac/projet/?controller=diluant&action=index	\$name \$dose (pas par URL)	Page d'erreur
Diluant/remove	Récupérer l'id du diluant qu'on souhaite supprimer	GET	https://www.gondange.com/alcoolimac/projet/?controller=diluant&action=remove&id=11	\$id (par URL)	Page d'erreur
Cocktails/add	Récupérer les id des alcools et diluants qui composent le cocktail	POST	https://www.gondange.com/alcoolimac/projet/?controller=cocktails&action=index	\$alcool \$diluant \$createur (pas par URL)	Page d'erreur
Cocktails/remove	Récupérer l'id du cocktail qu'on souhaite supprimer	GET	https://www.gondange.com/alcoolimac/projet/?controller=cocktails&action=remove&id=11	\$id (par URL)	Page d'erreur

```

if (isset($_POST['btn_add'])) {
    $alcool = strip_tags($_REQUEST["alcool_cocktail"]);
    $diluante = strip_tags($_REQUEST["diluante_cocktail"]);
    $createur = strip_tags($_REQUEST["createur_cocktail"]);

    Cocktails::addCocktails($alcool, $diluante, $createur, "/alcoolimac/projet/");
}

```

Ajout d'un cocktail
grâce à la méthode
POST

```

<?php
function call($controller, $action) {
    require_once('controllers/' . $controller . '_controller.php');

    switch($controller) {
        case 'pages':
            $controller = new pages_controller();
            break;
        case 'posts':
            // we need the model to query the database later in the controller
            require_once('models/Post.php');
            $controller = new posts_controller();
            break;

        case 'alcool':
            // we need the model to query the database later in the controller
            require_once('models/Alcool.php');
            $controller = new alcool_controller();
            break;
        case 'diluante':
            // we need the model to query the database later in the controller
            require_once('models/Diluante.php');
            $controller = new diluante_controller();
            break;
        case 'cocktails':
            // we need the model to query the database later in the controller
            require_once('models/Cocktails.php');
            $controller = new cocktails_controller();
            break;
    }
}

```

extrait du fichier routes.php

```

<?php

class cocktails_controller {
    public function index() {
        // we store all the cocktails in a variable
        $cocktails = Cocktails::all();
        $alcools = Alcool::all();
        $diluants = Diluante::all();
        require_once('views/cocktails/index_cocktails.php');
    }

    public function remove() {
        // we expect a url of form ?controller=cocktails&action=show&id=x
        // without an id we just redirect to the error page as we need the post id to find it in the database
        if (!isset($_GET['id']))
            return call('pages', 'error');

        // we use the given id to get the right post
        $cocktail = Cocktails::deleteCocktails($_GET['id'], "/alcoolimac/projet/");
        require_once('views/cocktails/index_cocktails.php');
    }
}
?>

```

Extrait du fichier cocktails_
controller.php