

Cahier des charges v1

URL de l'application : <https://totum.ovh>

Présentation du projet

Vous vouliez faire du tennis, mais votre ami, avec qui vous deviez jouer, n'est finalement plus disponible.

Vous vouliez faire un foot, mais vos amis ont finalement préféré profiter de leur dimanche aprem pour faire la sieste.

Vous voulez organiser une soirée jeux de rôle, mais l'un des participants vous a fait faux bond.

Afin de remédier à ça, il y a Totum.

Totum est une application qui met en relation des particuliers d'une même ville afin qu'il puisse avoir une personne avec qui faire des activités sportives et ludiques.

Cela permettra à des passionnés d'une même activité d'entrer en contact afin qu'ils puissent partager et pratiquer leur activité favorite.

En plus du fait de pouvoir rechercher une activité à rejoindre, les utilisateurs pourront également proposer des activités.

Définition des besoins

Les besoins :

- Beaucoup de personnes manquent de partenaires pour participer à des activités collectives sportives ou ludiques
- Totum facilite la mise en relation et l'organisation de telles activités en prenant en compte la localisation et les aspirations de chaque participant-e-s

Solution :

- Totum prend la forme d'un réseau social qui permet de s'inscrire pour rejoindre une communauté d'utilisateurs qui souhaitent partager leurs passions
- Chaque utilisateur peut proposer des activités et participer à celles proposées par les autres
- L'application permet également d'archiver les activités passées et propose un chat entre participants afin de peaufiner l'organisation de l'évènement futur
- Elle permet aussi de visualiser les activités disponibles sur une zone géographique donnée

MVP

- Pouvoir faire une recherche d'activité dans une zone
- Pouvoir faire une recherche d'activité par catégorie
- Création de compte
- Gestion du compte
- Connexion
- Se déconnecter du compte
- Suppression du compte
- Création d'activité
- Suppression d'activité
- Modification d'activité
- Accepter/ refuser participant dans l'activité
- Se désinscrire de l'activité
- Zone commentaires
- Consulter les profils utilisateurs
- Formulaire de contact
- Accès zone administrateur / visu des comptes utilisateurs, messages de réclamations, (stats) , envoyer mail

évolution potentielle :

- Ajout d'une alerte météo
- Liste d'amis
- Chat de discussion entre amis
- Connexion avec login Google et Facebook
- Sortir un participant de l'activité
- Modération du chat par IA
- Archivage des activités
- Album photos (????)
- Géolocalisation / interface carte
- Bannir utilisateur
- Mail de récupération de mdp
- Chat de discussion dans l'activité
- L'archivage des activités
- Accès au profil utilisateur
- Modification du profil utilisateur / photo / pseudo / mail

Spécifications techniques

Frontend :

- JavaScript/REACT
- Frameworks CSS : Tailwind, Bulma??

- Axios

Backend :

- JavaScript
- NodeJS
- Express
- Postgresql (database)
- MongoDB (database chat)
- Sqitch (management database)
- JSDoc
- Bcrypt

Backend et frontend :

- JEST
- Eslint
- Git / Github

Déploiement :

- Docker / Docker compose
- Github action

Définition de la cible du projet

- Les sportifs voulant faire des activités a plusieurs
- Les personnes arrivant dans une nouvelle ville et ne connaissant personne
- Les personnes voulant faire des activités ludiques à plusieurs
- Les personnes voulant proposer à d'autres personnes de les rejoindre pour participer à des activités

Les navigateurs compatibles

- Safari
- Chrome
- Firefox
- Edge
- Opera
- IE (Sans forcer)

Arborescence de l'application

Routes back :

(<https://api.totum.ovh/v1>)

Routes	Contrôleur	GET	POST	PATCH	DELETE	DESCRIPTION
Routes utilisateurs						
/user/login	user/login	✗	✓	✗	✗	connexion et création de session
/user/logout	user/logout	✓	✗	✗	✗	destruction de session
/user/createNew	user/createUser	✗	✓	✗	✗	création nouvel utilisateur
/user/:id/manage	- user/getOneUser - user/updateUser - user/removeUser	✓	✗	✓	✓	gestion profile utilisateur
/user/:id/manage/passwordUpdate	user/updatePassword	✗	✓	✗	✗	modification mot de passe utilisateur
/user/:id/manage/emailUpdate	user/updateEmail	✗	✓	✗	✗	modification email utilisateur
/user/:id/activity	user/getActivity	✓	✗	✗	✗	liste des activités liées à un utilisateur
Routes activités						
/activities	activity/getAll	✓	✗	✗	✗	récupération de toutes les activités
/activities/:category/category	activity/getByCategory	✓	✗	✗	✗	récupération des activités par catégorie
/activities/:geo/geo	activity/getByGeo	✓	✗	✗	✗	récupération des activités par géo-localisation
/activities/:search/search	activity/getBySearch	✓	✗	✗	✗	recherche activité par mot clé

/activity/createNew	activity/createActivity	✗	✓	✗	✗	création nouvel activité
/activity/:id/manage	- activity/getOneActivity - activity/updateActivity - activity/removeActivity	✓	✗	✓	✓	gestion des activités
/activity/:id/user	activity/getUser	✓	✗	✗	✗	liste des utilisateurs liées à une activité
Routes commentaires						
/comment/:id/user	comment/getByUser	✓	✗	✗	✗	récupération des commentaires par utilisateur
/comment/:id/activity	comment/getByActivity	✓	✗	✗	✗	récupération des commentaires par activité
/comment/createNew	comment/createComment	✗	✓	✗	✗	ajouter un commentaire à une activité
/comment/:id/manage	- comment/getOneComment - comment/updateComment - comment/removeComment	✓	✗	✓	✓	gestion des commentaires
Routes catégories						
/categories	category/getAll	✓	✗	✗	✗	récupérer toutes les catégories
/category/createNew	category/createCategory	✗	✓	✗	✗	ajouter une catégorie
/category/:id/manage	- category/getOneCategory - category/updateCategory - category/removeCategory	✓	✗	✓	✓	gestion des catégories
Routes inscription activité						
/register	register/createRegister	✗	✓	✗	✗	inscription d'un utilisateur à une activité
/register/:id/manage	- register/getOneRegister - register/updateRegister	✓	✗	✓	✓	gestion des inscriptions

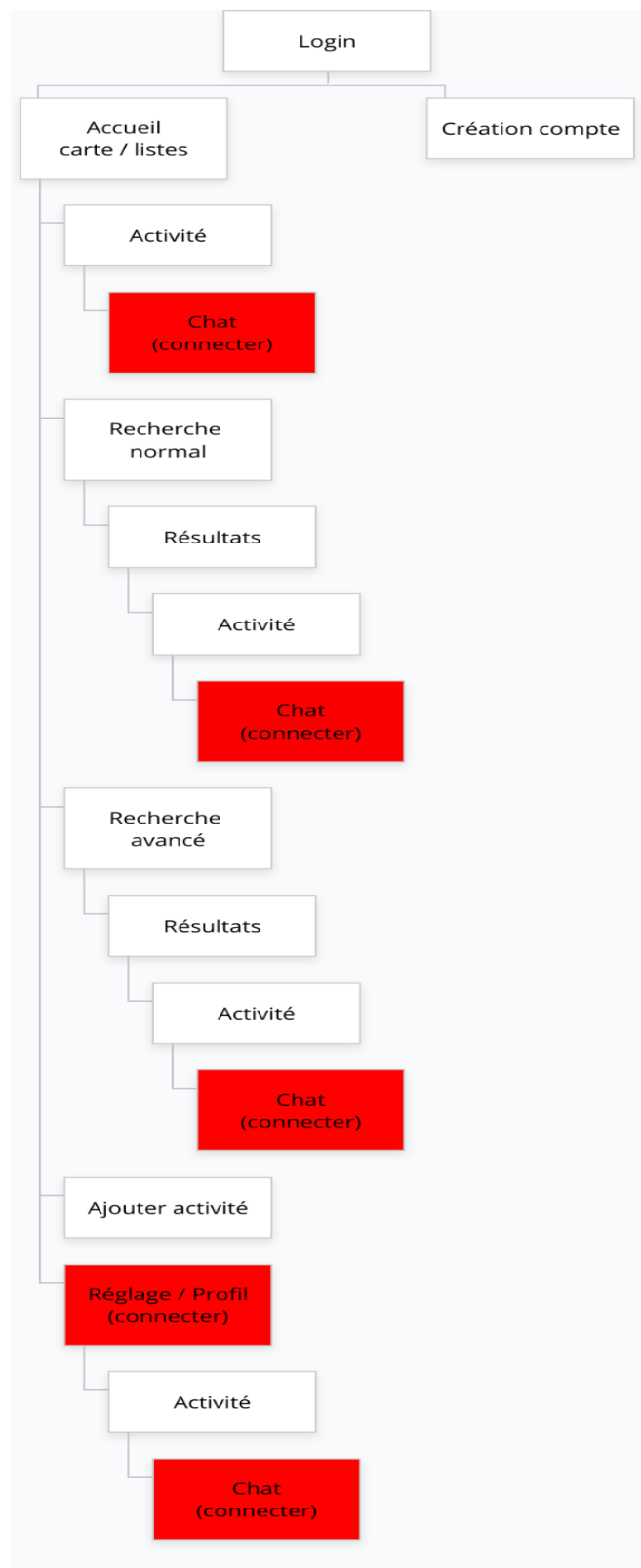
	- register/removeRegister					
Routes meta						
/meta/:id/manage	- meta/getByUser - meta/updateByUser	✓	✗	✓	✗	modification métadonnées (cookie / geo)
Route level						
/level/getAll	- level/getAll	✓	✗	✗	✗	Affiche tous les niveaux
/level/getById/:id	- level/getById	✓	✗	✗	✗	Affiche un niveau par son id

Routes front :

Routes	Description
/login	connection
/logout	déconnecter
/home	accueil desktop
/home/map	accueil carte
/home/list	accueil liste
/search	page recherche
/activity/list	liste des activités
/activity/{name}	page de l'activité
/activity/{name}/chat	chat de l'activité
/activity/member	participant activité
/activity/{name}/manage	page de gestion activité

/activity/create	page de création activité
/profil/:id	page profil
/profil/create	création de profil
/profil/params	paramètre utilisateur
/profil/param/notifications	paramètre utilisateur notification
/mentions	mentions légale
/profil/help	formulaire de contact
/profil/changeMode	choix mode sombre

Structure :



User stories

En tant que	Je veux pouvoir	Dans le but de	Version
Visiteur	avoir accès à une activité	voir l'activité en détails	
Visiteur	accéder à la liste des activités	avoir une vue d'ensemble des activités proposées	
Visiteur	faire une recherche d'activité	trouver une activité selon des critères précis	
Visiteur	choisir d'être géolocalisé	pour circonscrire ma recherche à une zone géographique	
Visiteur	créer un compte	pouvoir me loguer	
Visiteur	me loguer	changer de rôle et avoir accès aux fonctionnalités user	
Visiteur	écrire aux administrateurs	en cas de problème	
Visiteur	récupérer mdp du compte	récupérer mon mdp perdu	V2
Visiteur	visualiser les stats de connexion du site	connaître l'activité du réseau social	V2
Utilisateur	créer une activité	la proposer à la communauté	
Utilisateur	se déconnecter	fermer sa session	
Utilisateur	Supprimer mon compte	de supprimer mes accès	
Utilisateur	créer mon profil	ajouter photo/pseudo/mail... .	
Utilisateur	modifier son profil	changer photo/pseudo/mail... .	
Utilisateur	m'inscrire à une activité	participer à une activité	
Utilisateur	consulter les profils des utilisateur-ice-s	connaître les autres utilisateur-ice-s pour	

		une activité	
Utilisateur	modifier une activité	mettre à jour l'activité	
Utilisateur	supprimer une activité	annuler une activité	
Utilisateur	visualiser les acceptations des organisateurs	savoir si je peux participer à une activité	
Utilisateur	prévenir de la suppression d'activité	d'alerter tous les participants	
Utilisateur	accès à ma liste de camarade	voir s' ils sont connectés, discuter	
Utilisateur	communiquer avec mes camarades	d'échanger sur n'importe quel sujet	
Utilisateur	signaler utilisateur	d'alerter des mauvais comportements	
Utilisateur	Accepté les demandes de participations	Ajouter qui l'on souhaite dans l'activité	
Utilisateur	Bloquer les messages d'un utilisateur	ne plus être importuné	V2
Utilisateur	communiquer avec les participants	d'échanger sur l'activité	V2
Utilisateur	ajouter photos dans l'activité	garder des souvenirs	V2
Administrateur	accès aux messages de réclamations	régler les incidents rencontrés	
Administrateur	supprimer des activités	modérer les activités inappropriées	
Administrateur	envoyer un mail	contacter un-e user	V2

Administrateur	visualiser les stats du site	connaître l'activité du réseau social	V2
Administrateur	bannir les utilisateurs	bannir les mauvais joueurs	V2

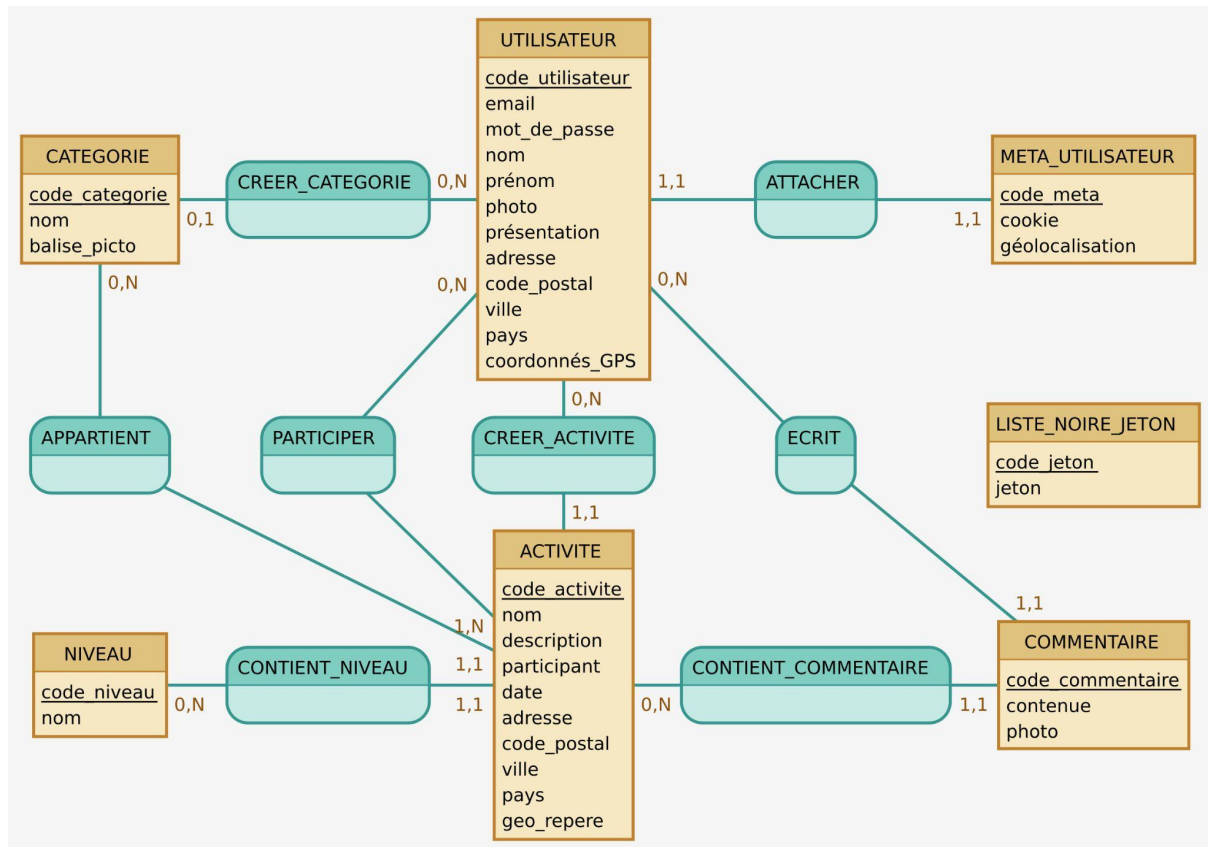
Équipe

- **Product Owner** : Anas
- **Scrum Master** : Benoit
- **Référents Techniques** : Kenny
- **Git Master** : Damien
- **Lead dev front** : Xavier
- **Lead dev back** : Kenny

Documents base de données

MCD :

- CATEGORIE: code_categorie, nom, balise_picto
- CREER_CATEGORIE, 0N UTILISATEUR, 01 CATEGORIE
- UTILISATEUR: code_utilisateur, email, mot_de_passe, nom, prénom, photo, présentation, adresse, code_postal, ville, pays, coordonnées_GPS
- ATTACHER, 11 UTILISATEUR, 11 META_UTILISATEUR
- META_UTILISATEUR: code_meta, cookie, géolocalisation
- APPARTIENT, 11 ACTIVITE, 0N CATEGORIE
- PARTICIPER, 0N UTILISATEUR, 1N ACTIVITE
- CREER_ACTIVITE, 0N UTILISATEUR, 11 ACTIVITE
- ECRIT, 11 COMMENTAIRE, 0N UTILISATEUR
- LISTE_NOIRE_JETON: code_jeton, jeton
- NIVEAU: code_niveau, nom
- CONTIENT_NIVEAU, 11 ACTIVITE, 0N NIVEAU
- ACTIVITE: code_activite, nom, description, participant, date, adresse, code_postal, ville, pays, geo_repere
- CONTIENT_COMMENTAIRE, 11 COMMENTAIRE, 0N ACTIVITE
- COMMENTAIRE: code_commentaire, contenue, photo



MLD :

- UTILISATEUR: (code_utilisateur, email, mot_de_passe, nom, prénom, photo, présentation, adresse, code_postal, ville, pays, coordonnées_GPS, #code_meta, created_at, updated_at)
- ACTIVITE: (code_activite, nom, description, participants_max, date, #code_niveau, adresse, code_postal, ville, pays, geo_repere, #code_utilisateur, #code_categorie, created_at, updated_at)
- COMMENTAIRE: (code_commentaire, contenu, photo, #code_utilisateur, #code_activite, created_at, updated_at)
- NIVEAU: (code_niveau, nom)
- META_UTILISATEUR: (code_meta, cookie, géolocalisation, #code_utilisateur, created_at, updated_at)
- CATEGORIE: (code_categorie, nom, image, #code_utilisateur, created_at, updated_at)
- PARTICIPER: (code_participer, #code_utilisateur, #code_activite, created_at, updated_at)
- LISTE_NOIRE_JETON: (code_jeton, jeton, created_at, updated_at)

Dictionnaire de données :

CHAMP	TYPE	SPÉCIFICITÉS	DESCRIPTION
TABLE UTILISATEUR (users)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique utilisateur
email	TEXT -> DOMAIN EMAIL	NOT NULL UNIQUE	email utilisateur
password	TEXT	NOT NULL	hash mot de passe
firstname	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	prenom
lastname	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	nom
picture	TEXT		liens vers la photo de profil
about	TEXT -> DOMAIN TEXT (pas de symbole ou autre)		texte de présentation
address	TEXT -> DOMAIN TEXT (pas de symbole ou autre)		rue et numéro de rue
zip_code	INT ou TEXT -> DOMAIN ZIP_CODE	NOT NULL	code postale
city	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	ville
country	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	pays
coordinate	TEXT		coordonnés GPS
meta_id	INT	REFERENCES meta("id")	clé étrangère meta utilisateur
created_at	TIMESTAMP TZ	NOT NULL DEFAULT NOW()	date de création
updated_at	TIMESTAMP TZ		date de dernière mise à jour

TABLE ACTIVITE (activity)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique activité
name	TEXT -> DOMAIN (pas de symbole ou autre)	NOT NULL	Nom de l'activité
description	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	présentation de l'activité
max_participants	INT	NOT NULL	nombre de participant nécessaire à l'activité
date	TEXT	NOT NULL	date de l'activité
level	INT	NOT NULL REFERENCES level("id")	Niveau de l'activité
address	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	adresse activité
zip_code	INT ou TEXT -> DOMAIN ZIP_CODE	NOT NULL	code postal
city	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	ville
country	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	pays
landmark	TEXT		géo repères
id_user	INT	NOT NULL REFERENCES users("id")	identifiant unique créateur
id_category	INT	NOT NULL REFERENCES category("id")	identifiant unique catégorie
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	date de création
updated_at	TIMESTAMPTZ		date de dernière mise à jour

TABLE COMMENTAIRES (comment)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique commentaire
content	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL	corps du commentaire
picture	TEXT		liens vers l'image
id_user	INT	NOT NULL REFERENCES users("id")	clé étrangère créateur du commentaire
id_activity	INT	NOT NULL REFERENCES activity("id")	clé étrangère de l'activité
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW	date de création
updated_at	TIMESTAMPTZ		date de dernière mise à jour

TABLE NIVEAU (level)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique meta
name	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL UNIQUE	nom du niveau
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	date de création
updated_at	TIMESTAMPTZ		date de dernière mise à jour

TABLE META_UTILISATEUR (meta)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique meta
cookie	BOOLEAN	DEFAULT FALSE	cookies
landmark	BOOLEAN	DEFAULT FALSE	géolocalisation
id_user	INT	NOT NULL REFERENCES user("id")	clé étrangère utilisateur
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	date de création
updated_at	TIMESTAMPTZ		date de dernière mise à jour

TABLE CATEGORIE(category)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique catégorie
name	TEXT -> DOMAIN TEXT (pas de symbole ou autre)	NOT NULL UNIQUE	nom de la catégorie
picto	TEXT		balise react picto
id_user	INT	REFERENCES users("id")	clé étrangère utilisateur / créateur
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	date de création
updated_at	TIMESTAMPTZ		date de dernière mise à jour

TABLE PARTICIPER (user_activity)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique participé
id_user	INT	NOT NULL REFERENCES users ("id")	identifiant unique utilisateur
id_activity	INT	NOT NULL REFERENCES activity ("id")	identifiant unique activité
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	date de création
updated_at	TIMESTAMPTZ		date de dernière mise à jour

TABLE LISTE NOIR JETON (token_blacklist)			
id	INT	GENERATED ALWAYS AS IDENTITY PRIMARY KEY	identifiant unique token blacklist
token	TEXT	NOT NULL	token interdit sur l'app
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	date de création
updated_at	TIMESTAMPTZ		date de dernière mise à jour