

Le cahier des charges

- **Les rôles individuels**

Product Owner - Alice

Connaît le produit et représente les intérêts et les besoins du client/des utilisateurs purement fonctionnels (pas techniques)

Tranche en cas de conflits fonctionnels (pas techniques)

En général c'est le porteur du projet (s'il est dans le groupe)

Scrum Master - Anissa

Garant de la méthode du projet : il gère le respect des conventions définies dans le groupe

S'assure que toutes les tâches sont bien attribuées, suivies, accomplies

Assure la communication au sein du groupe : il vérifie que tout le monde a les bonnes informations

Gère l'outil de suivi du projet

Anime la réunion du matin et gère l'avancement du projet

Lead dev Back - Floriane

Choisit les orientations importantes, choix techniques importants

S'assure du bon fonctionnement de sa partie du projet

Lead dev Front - Sam

Choisit les orientations importantes, choix techniques importants

S'assure du bon fonctionnement de sa partie du projet

Référents techniques : git - Jeanne

Git master

Garant du bon fonctionnement du versioning avec Git.

Responsable du bon fonctionnement du versionning, vérifie les PR et merge, gère les conflits etc.

- **Le CDC : contenu**

La présentation du projet :

Adopte un arbre est une plateforme de parrainage d'arbres qui a pour but de connecter des personnes souhaitant contribuer à des projets de reforestation avec des organisations ou des initiatives locales. Les utilisateurs peuvent acheter un ou plusieurs arbres en finançant leur plantation et leur entretien. Ils auront la possibilité de recevoir des informations régulières sur la croissance et le développement de leurs arbres parrainés et des projets déjà existants (arbres déjà plantés/objectif à atteindre) avec des photos régulières et l'inscription à une newsletters.

Les besoins et des objectifs du projet :

Il y a énormément de cas de déforestation ces dernières années, c'est pourquoi il est important de sensibiliser à la santé de notre planète en contribuant à la reforestation, en ouvrant la possibilité d'avoir une communauté soudée, prête à participer financièrement ainsi que de s'investir dans le renouveau de notre planète.

Les fonctionnalités du projet (spécifications fonctionnelles) :

Le MVP (Minimum Viable Product) :

- Pouvoir adopter (parrainer) un arbre
- Consulter les arbres à parrainer
- Consulter la page des projets en cours
- Pouvoir s'identifier
- Pouvoir créer un compte
- Consulter son profil avec nos arbres adoptés
- Backoffice pour les admins, et organismes
- Créer un arbre, projet, organisme en backoffice
- Compteur d'arbre déjà plantés

Évolutions potentielles :

- Newsletter
- Mise à jour des évolutions des arbres adoptés (photos, descriptions..)
- Page qui regroupe les associations / organismes
- Page de contact
- Projets finis / réussis

La liste des technologies utilisées pour le projet (spécifications techniques) :

- REACT
- SASS
- Symfony
- SQL
- HTML / SCSS

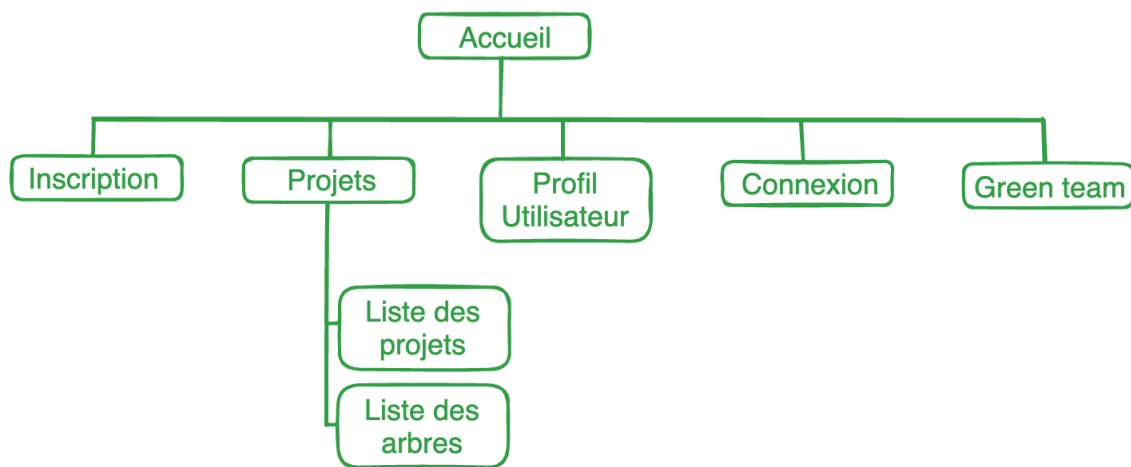
La cible du projet (le public visé) :

- Écologistes
- Personnes ayant une sensibilité environnementale
- Personnes majeures et actives (moyens financiers)

Les navigateurs compatibles :

- Chrome / Firefox / Safari

L'arborescence de l'application (le chemin de l'utilisateur) :



La liste des routes prévues :

Routes FRONT

URL	Composant	Title	Content	Comment
/	Accueil	Adopte un arbre	Page d'accueil	-
/greenteam	GreenTeam	Green Team	Présentation de l'organisation et de son équipe ainsi que de sa mission et de ses valeurs.	-
/arbres	Participer	Participer	Liste des arbres	-
/projets	ProjetsList	Projet en cours	Liste des projets en cours	-
/projets/[id]	Projet	[Nom du projet] en cours	La page d'un projet	[id] représente l'id du projet
/login	Connexion	Connexion	Formulaire de connexion	
/profile/[firstName]	ProfilUtilisateur	Mon Compte	Espace utilisateur	[firstName] représente le prénom de l'utilisateur

/inscription	Inscription	Inscription	Formulaire d'inscription	-
/*	Erreur 404	Erreur	Routes imprévues dans le site	-

Routes BACK

Route API:

Endpoint	HTTP method	Name	Description	Parameters
/api/trees	GET	app_tree_list	Get all trees	-
api/trees/[id]	GET	app_tree_read	Get a single tree details	id - ID of the tree to show
/api/trees	POST	app_tree_add	Create a new tree	-
/api/trees/[id]	PUT, PATCH	app_tree_edit	Update a tree	id - ID of the tree to update
/api/trees/[id]	DELETE	api_tree_delete	Delete a tree	id - ID of the tree to delete

/api/trees/[id]/adopt	POST	api_tree_adopt	Adopt a tree	id - ID of the tree to adopt
/api/trees/count	GET	api_tree_count	The counter of all adopted trees	
/api/users	GET	api_user_browse	Get all user	-
/api/users/[id]	GET	api_user_read	Get a single user details	id - id of the user to show
/api/users	POST	api_user_add	Create a new user	-
/api/users/[id]	PUT, PATCH	api_user_edit	Update a user	id - id of the user to update
/api/users/[id]	DELETE	api_user_delete	Delete a user	id - id of the user to delete
/api/organisms	GET	api_organism_list	Get all organism	-
/api/organisms/[id]	GET	api_organism_read	Get a single organism details	id - ID of the organism to show
/api/organisms	POST	api_organism_add	Create a new organism	-

/api/organisms/[id]	PUT, PATCH	api_organism_edit	Update a organism	id - ID of the organism to update
/api/organisms/[id]	DELETE	api_organism_delete	Delete a organism	id - ID of the organism to delete
/api/species	GET	api_specie_list	Get all species	-
/api/species/[id]	GET	api_specie_read	Get a single specie details	id - ID of the species to show
/api/species	POST	api_specie_add	Create a new specie	-
/api/species/[id]	PUT, PATCH	api_specie_edit	Update a specie	id - ID of the species to update
/api/species/[id]	DELETE	api_specie_delete	Delete a Species	id - ID of the species to delete
/api/projects	GET	api_project_list	Get all project	-
/api/projects/[id]	GET	api_project_read	Get a single project details	id - ID of the project to show
/api/projects	POST	api_project_add	Create a new project	-

/api/projects/[id]/species	GET	api_project_specie	Show species by project	id - ID of the project
/api/projects/[id]	PUT, PATCH	apl_project_edit	Update a project	id - ID of the project to update
/api/projecst/[id]	DELETE	api_project_delete	Delete a project	id - ID of the project to delete
/api/pictures	GET	api_picture_list	Get all pictures	-
/api/pictures/[id]	GET	api_picture_read	Get a single picture details	id - ID of the picture to show
/api/pictures	POST	api_picture_add	Create a new picture	-
/api/pictures/[id]	PUT, PATCH	api_picture_edit	Update a picture	id - ID of the picture to update
/api/pictures/[id]	DELETE	api_picture_delete	Delete a picture	id - ID of the picture to delete

Route Backoffice:

URL / Endpoint	HTTP method	Name	Description
----------------	-------------	------	-------------

/	GET	backoffice_index	Page d'accueil du BackOffice
/backoffice/projet/creer	GET, POST	app_back_project_create	Page de création d'un projet
/backoffice/projet/modifier	GET, POST	app_back_project_edit	Page d'édition d'un projet
/backoffice/projet/supprimer	GET, POST, DELETE	app_back_project_delete	Page de suppression d'un projet
/backoffice/arbre/creer	GET, POST	app_back_tree_create	Page de création d'un arbre
/backoffice/arbre/modifier	GET, PUT, PATCH	app_back_tree_edit	Page d'édition d'un arbre
/backoffice/arbre/supprimer	GET, POST, DELETE	app_back_tree_delete	Page de suppression d'un arbre
/backoffice/organisme/creer	GET, POST	app_back_organism_create	Page de création d'un organisme
/backoffice/organisme/supprimer	GET, POST, DELETE	app_back_organism_delete	Page de suppression d'un organisme
/backoffice/organisme/modifier	GET, PUT, PATCH	app_back_organism_edit	Page d'édition d'un organisme
/login	-	app_security_login	Page de connexion d'un utilisateur
/logout	-	app_security_logout	Page de déconnexion d'un utilisateur

Les user stories :

En tant que visiteur :

- Je peux visualiser la page d'accueil afin d'en découvrir plus sur le concept du site
- Je peux voir la Green Team afin de visualiser l'équipe qui a construit le site
- Je peux voir les projets en cours afin de comprendre les actions mise en place
- Je peux me créer un compte utilisateur afin d'acheter un arbre

En tant qu'utilisateur :

- Je peux me connecter à mon espace utilisateur afin d'accéder à un profil personnalisé
- Je peux financer un ou plusieurs projets/arbres disponibles
- Je peux voir mes différents arbres parrainés

En tant qu'organisme/admin :

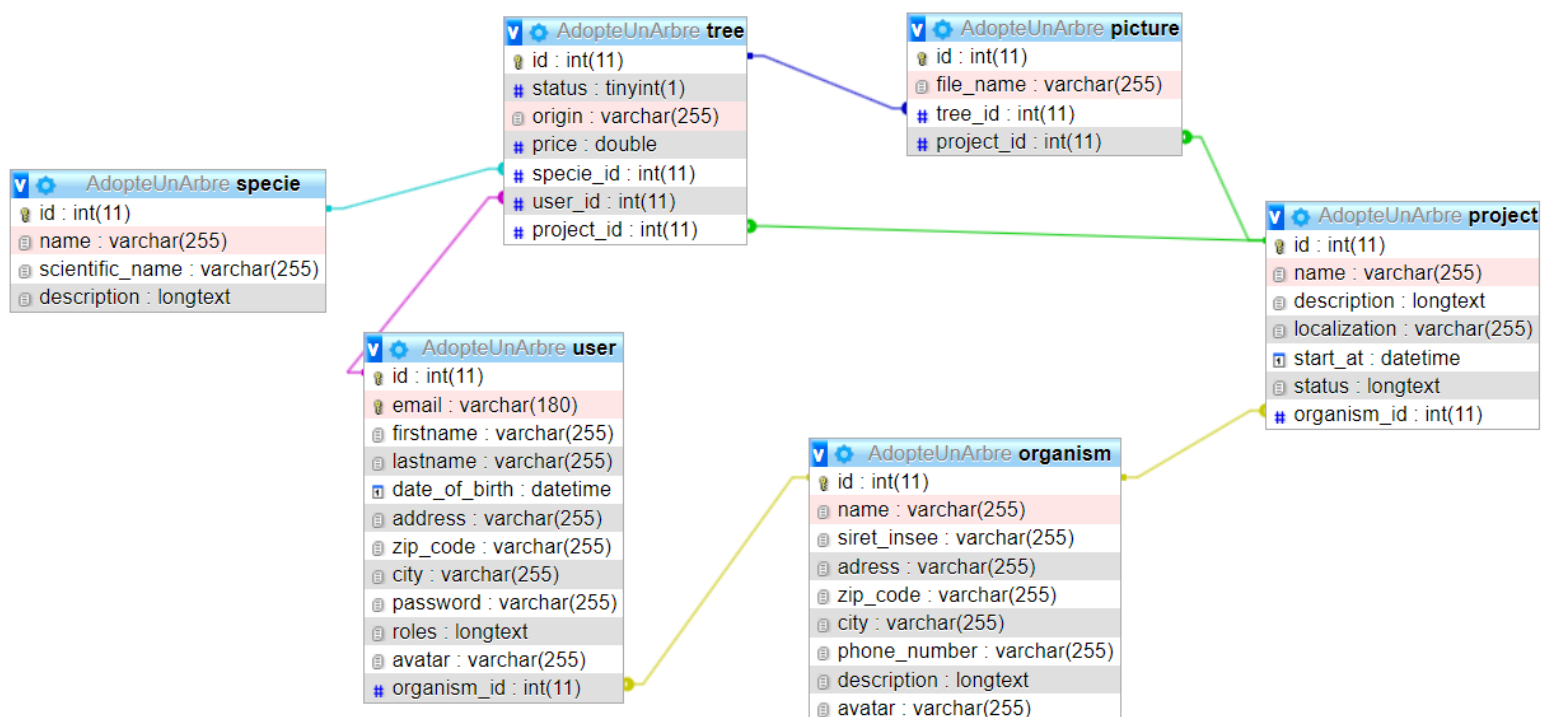
- Je peux me connecter et accéder au back office
- Je peux créer, modifier, supprimer des projets
- Je peux créer, modifier, supprimer des arbres

En tant qu'admin :

- Je peux créer, modifier, supprimer des organismes (sécurité ?)

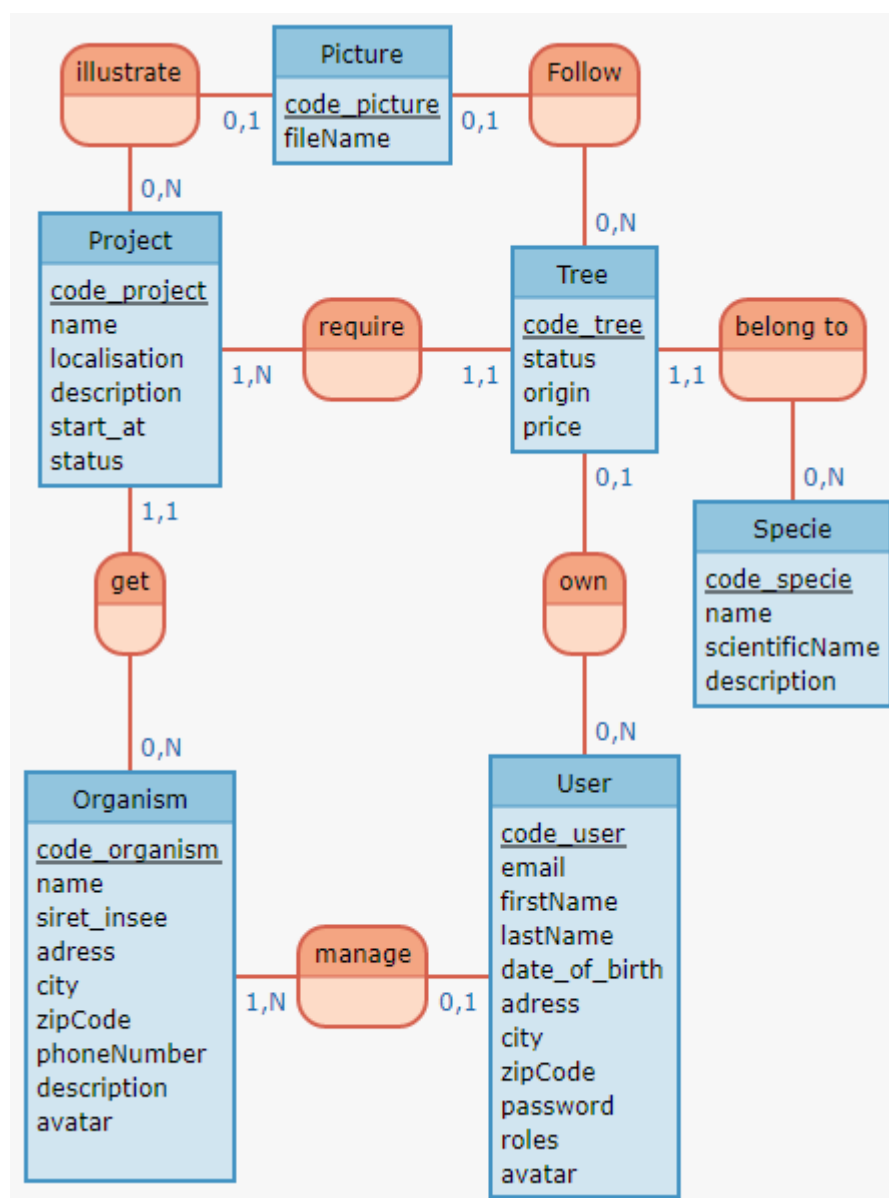
• Les documents relatifs à la BDD

Réalisé sur : phpMyAdmin



Le MCD :

Réalisé sur : <https://www.mocodo.net/>



Dico de données :

Table Tree:

Champ	Type	Spécificités	Description
Id	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identifiant de l'arbre
Status	TINYINT(1)	NOT NULL	Statut de l'arbre (0 => pas encore planté et 1 => planté)
Origin	VARCHAR(128)	NOT NULL	Origine de l'arbre
Price	FLOAT	NOT NULL	Prix de l'arbre
specie_id	INT	NOT NULL	Id de l'espèce de l'arbre
user_id	INT		Id de l'utilisateur adoptant l'arbre

project_id	INT	NOT NULL	Id du projet de l'arbre
------------	-----	----------	-------------------------

Table User:

Champ	Type	Spécificités	Description
Id	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identifiant de l'utilisateur
Email	VARCHAR(180)	NOT NULL	Email de l'utilisateur
FirstName	VARCHAR(255)	NOT NULL	Prénom de l'utilisateur
LastName	VARCHAR(255)	NOT NULL	Nom de l'utilisateur
Date_of_birth	DATETIME_IMMUTABLE	NOT NULL	date de naissance de l'utilisateur
Address	VARCHAR(255)	NOT NULL	Adresse de l'utilisateur
Zip_code	INT	NOT NULL	Code postal de l'utilisateur
City	VARCHAR(255)	NOT NULL	Ville de l'utilisateur

Password	VARCHAR(255)	NOT NULL	Mot de passe
Rôles	LONGTEXT	NOT NULL	Rôle de l'utilisateur Exemple : ['ROLE_USER']
Avatar	VARCHAR(255)		Photo de l'utilisateur
Organism_id	INT		Id de l'organisme de l'utilisateur

Table Organism:

Champ	Type	Spécificités	Description
Id	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identifiant de l'organisme
Name	VARCHAR(255)	NOT NULL	Dénomination sociale de l'organisme
Siret_insee	VARCHAR(255)	NOT NULL	immatriculation de l'organisme
Address	VARCHAR(255)	NOT NULL	adresse de l'organisme
PhoneNumber	VARCHAR(255)	NOT NULL	numéro de l'organisme

Zip_code	VARCHAR(255)	NOT NULL	Code postal de l'organisme
City	VARCHAR(64)	NOT NULL	Ville de l'organisme
Description	LONGTEXT	NOT NULL	Description de l'organisme
Avatar	VARCHAR(255)		Image de l'organisme

Table Specie:

Champ	Type	Spécificités	Description
Id	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identifiant de l'espèce
Name	VARCHAR(255)	NOT NULL	Nom de l'espèce
ScientificName	VARCHAR(255)	NOT NULL	Nom Scientifique de l'espèce
Description	LONGTEXT	NOT NULL	Détail sur l'espèce

Table Project:

Champ	Type	Spécificités	Description
Id	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identifiant du projet
Name	VARCHAR(255)	NOT NULL	Nom du projet
Description	LONGTEXT	NOT NULL	Description du projet
Localisation	VARCHAR(128)	NOT NULL	Lieu du projet
Start_at	DATETIME_IMMUTABLE	NOT NULL	Date de début du projet
Status	INT	NOT NULL	Choix multiples : en cours de validation, validé, terminé.

Table Picture:

Champ	Type	Spécificités	Description
-------	------	--------------	-------------

Id	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identifiant de la photo
File_name	VARCHAR(128)	NOT NULL	lien du fichier contenant la photo

- **Les Wireframes**

à mieux légender + plus de description (mots clés ou bulles)

- Accueil (sam)
- Participer - liste de tous les arbres (Floriane)
- Page d'un arbre - (Alice)
- Connexion - ceux déjà inscrit - (Anissa)
- Inscription particulier (Jeanne)
- Profil particulier - tableau de bord - (Jeanne)
- Inscription organisme (Floriane)
- Profil organisme (Sam)
- La green team (Alice)
- Profil/Interface admin - (Anissa)
- Page Projet en cours (Floriane)
- Page Créer / modifier un projet (Sam)
- Page créer / modifier un arbre (Jeanne)
- Page d'un projet