

"FOWL TUNE"

"Mais c'est quel oiseau ça ?"



PRÉSENTÉ PAR

Romane Bellomo
A. Medan
P. Vincent
M. Comminges

Résumé

Problème soulevé :

Plus le temps passe, plus le monde s'urbanise et moins il reste d'espaces et de lieux de vie pour l'espèce animale. Notre monde subit également des changements climatiques (parfois lourds) chaque année, ce qui a également un impact sur le développement de la faune et de la flore. Nous soulevons aujourd'hui un point essentiel et voulons nous focaliser sur un regroupement d'espèces en particulier : les oiseaux. Les oiseaux font parties intégrantes de nos villes et il n'est pas rare d'en croiser un à chaque coin de rue ou de les entendre piailler dès notre réveil. Nous souhaitons aujourd'hui mettre en lumière la très grande présence d'oiseaux dans nos villes, permettre à la population de se rendre compte de leur très grande présence et d'aider les scientifiques à étudier leur présence et leur cycle migratoire pour qu'ils puissent les aider à vivre dans un milieu qui puisse leur convenir.

Solution proposée :

Ici, nous proposons une solution innovante qui permettra de répondre à notre problème. Nous souhaitons créer une application à but collaboratif, entre scientifiques, passionnés d'ornithologie et amateurs en soif de découverte. Celle-ci aura pour objectif d'arriver à identifier à quel oiseau appartient un chant enregistré. Ce chant sera enregistré par les utilisateurs. et le répertoriera sur une carte interactive de l'application, où l'oiseau a pu être identifié. L'utilisateur pourra ainsi, garder en mémoire tous les oiseaux rencontrés depuis qu'il utilise l'application. Il pourra également consulter des fiches d'informations pour en apprendre plus sur les espèces rencontrées.

Valeur ajoutée :

Ce projet se veut en tous points, impactant dans plusieurs domaines. En déployant ce projet à échelle nationale, puis mondiale dans le futur, celui-ci se veut participatif avec toutes les personnes désirant y participer. Il s'inscrit dans le cadre de l'observation de la faune, et plus particulièrement dans l'étude ornithologique. Dans un monde en constante évolution où des villes sont de plus en plus urbanisées, ce projet permettra également d'étudier l'impact que de nouvelles constructions pourront avoir sur l'habitat des oiseaux. Enfin, les changements climatiques étant de plus en plus importants et alarmants, on pourra également observer si cela a un impact sur les migrations des oiseaux et sur des lieux autrefois fréquemment fréquentés par eux.

SOMMAIRE

CONTEXTE	4
NOS CONVICTIONS	5
NOTRE PROJET	6
DONNÉES UTILISÉES	7
<i>Web Scraping</i>	8
<i>NA Bird</i>	8
<i>Xeno-canto</i>	8
<i>Localisation</i>	9
MODÉLISATION DES DONNÉES	10
DATA ARCHITECTURE	14
CAS D'USAGE	
<i>Cas d'usage 1 – Identifier un oiseau à partir de son chant</i>	18
<i>Cas d'usage 2 – Étendre l'application à d'autres territoires</i>	19
<i>Cas d'usage 3 – Créer une base de données d'observations d'oiseaux</i>	20
<i>Cas d'usage 4 – Permettre aux utilisateurs de localiser les oiseaux en un coup d'oeil</i>	21
ANNEXES	23

Contexte

Nous sommes une équipe française de 4 personnes vivant à Toulouse et travaillant ensemble depuis quelques années sur des sujets portant sur l'IA. Fort d'expérience dans notre domaine, plusieurs de nos projets ont été mis en production et ont rencontré un succès très satisfaisant.

Dans le cadre de la politique « Toulouse Ville Verte », la Mairie de Toulouse développe et soutient la création de forêts urbaines : des îlots de verdure au cœur de la ville et les zones périurbaines, afin de répondre aux exigences des habitants en termes de régulation de la température, d'amélioration de la qualité de l'air, et de la promotion de la biodiversité urbaine.

« Toulouse Ville Verte » souhaite inciter les Toulousains à davantage prêter attention à leur environnement immédiat et a lancé en décembre 2022 un appel à projets : la mairie étudiera tout particulièrement les projets ayant pour objectif la mise en avant du patrimoine zoologique urbain et périurbain de la ville.

C'est dans ce cadre que nous présentons notre projet :

Une application qui, grâce à l'intelligence artificielle et l'apprentissage machine, associe le chant des oiseaux à proximité de l'utilisateur à leur nom et les stocke ensuite dans une « collection » propre à chaque utilisateur.

L'application permettra d'identifier à quel oiseau appartient le chant enregistré et le répertoriera sur une carte interactive de l'application, où l'oiseau a pu être identifié. L'utilisateur pourra ainsi, garder en mémoire tous les oiseaux rencontrés depuis qu'il utilise l'application. Il pourra également consulter des fiches d'informations pour en apprendre plus sur les espèces rencontrées.

Pour ce faire, notre équipe est donc composée d'un chef de projet, d'un data engineer, d'un data analyst et d'un expert big data.

Dans le développement suivant, nous détaillerons ce qui nous a mené à ce projet, les besoins rencontrés et comment nous allons y répondre.

NOS CONVICTIONS

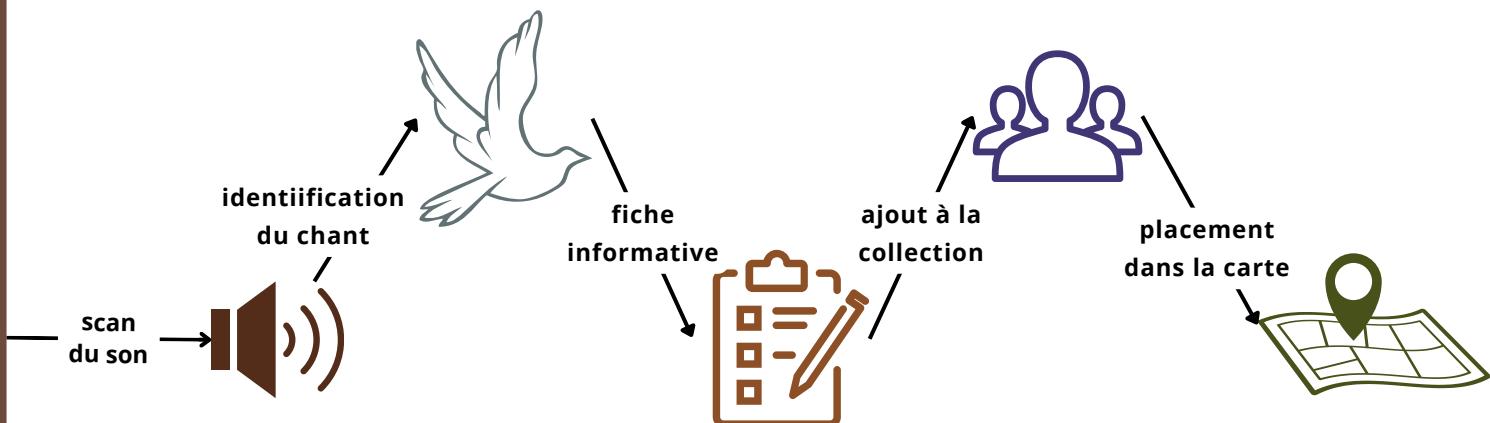
- **Penser l'ornithologie amateur comme un jeu de collection** : notre proposition se distingue de l'offre déjà présente sur le marché par une approche radicalement gamifiée du birdwatching (ou ornithologie amateur). Nous sommes convaincus que le succès de jeux mobiles de collection en réalité augmentée (comme Pokémon Go par Niantic) est transposable à l'activité de birdwatching, toutes générations confondues.
- **Une gestion de projet incrémentale et itérative** : voir grand, commencer petit, voici comment nous envisageons la gestion de ce projet. En commençant sur un périmètre restreint, et en prenant en compte les retours des utilisateurs dans nos améliorations, nous serons en mesure de créer le produit qui correspond réellement aux attentes des Toulousains et des Toulousaines.
- **Inscrire l'application dans une démarche de science participative** : inscrire l'application dans la lignée des initiatives scientifiques visant à davantage impliquer les citoyens dans des opérations scientifiques. L'ornithologie amateur est d'ailleurs pionnière de ces techniques, avec le projet FeederWatch du Cornell Lab of Ornithology dans les années 1970. Nous souhaitons intégrer la communauté scientifique de Toulouse dans cette démarche en leur fournissant une base de données dynamique des populations aviaires dans les îlots de verdure urbains, afin de mesurer les impacts de la politique Toulouse Ville Verte.



NOTRE PROJET

Notre projet est donc de créer une application capable (principalement) de :

- Scanner un son enregistré par un utilisateur
- Déterminer si ce son correspond au chant d'un oiseau
- Associer cet oiseau à une fiche informative courte
- Placer l'oiseau dans la collection de l'utilisateur
- Placer l'information du lieu, l'heure, et du nom de l'oiseau dans une base de données destinée aux scientifiques.

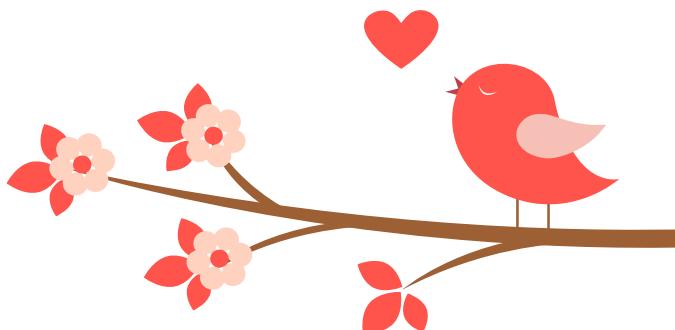


Dans le futur, cette application pourrait :

- Envoyer une notification dès que l'utilisateur passe à proximité d'un îlot de verdure susceptible d'accueillir des oiseaux
- Déterminer les meilleurs lieux où entendre spécifiquement certains oiseaux

On peut également imaginer que d'autres communes dans la même démarche que Toulouse pourraient être intéressées par le dispositif, le généralisant à toutes les villes de France.

De même, des zoos ou des réserves ornithologiques cherchant à davantage impliquer leurs visiteurs pourraient être intéressés par l'application : ils pourraient créer des parcours spécifiques ou engager les visiteurs à scanner tous les chants d'oiseaux présents avant de partir, afin d'augmenter le temps de visite.



DONNÉES UTILISÉES

Les sources de données prévues à l'utilisation de notre projet, ou déjà utilisées, sont présentées ci-dessous. Nous possédons principalement des données structurées et non-structurées.

DataSource	Description	Type	Lien
Web Scraping	Méthode pour accéder aux informations sur les oiseaux et localisation.	Semi-structurées	https://ebird.org/hotspots?env.minX=-0.326875030999872&env.minY=42.333442689&env maxX=4.8453440670001&env.maxY=45.0446662900002&yr=all&m=
NA Birds	Dataset de 48 000 photos d'oiseaux représentant 400 espèces ; données labellisées.	Données non-structurées	https://vu.fr/FIYZ
Xeno-canto	Base de données de sons d'oiseaux avec descriptif du type de son (parade nuptiale, chant) mais aussi les noms scientifiques, le pays d'enregistrement. Licence Creative Commons libre.	Semi-structurées	https://xeno-canto.org/explore/api
Localisation	Données de localisation de certains oiseaux	Structurées	Status and trends of bird populations: datasets from Article 12, Birds Directive 2009/147/EC reporting – European Environment Agency (europa.eu)

DONNÉES UTILISÉES

Pour mieux comprendre comment notre projet va être mis en place, voici ci-joint une description de l'utilisation de nos données.

Web scraping

Les utilisateurs pourront voir les données en tant réel sous la forme de dashboards, implantés dans l'application. Pour partir avec une base pour la mise en place des dashboards, nous sommes partis avec une base de données que nous avons extrait d'un site web. Pour ce faire, nous avons utilisé le code source des différentes pages web utilisées et en avons extirpé les données les plus pertinentes. Nous en avons ensuite converti ces données en un fichier csv pour pouvoir l'exploiter. Le code d'extraction et le dashboard seront mis en annexe.

NA Birds

Afin de pouvoir montrer aux différents utilisateurs à quoi ressemblent les oiseaux qu'ils ont entendu, nous avons besoin d'un jeu de données répertoriant des images. Nous avons donc pris le jeu de données NA Birds. Une fois le son enregistré et identifié grâce à notre modèle d'IA, nous allons lier les images aux oiseaux identifiés. Ainsi, les utilisateurs pourront avoir un aperçu physique d'à quoi ressemble un oiseau.

Xeno-canto

Pour que notre application soit fonctionnelle, nous avons besoin de données permettant de créer notre modèle d'IA. Pour ce faire, nous allons utiliser des jeux de données de type audio. En utilisant différentes techniques de deep learning, nous allons mettre en place un algorithme permettant d'identifier le son d'un oiseau et de déterminer à quelle espèce il appartient.

DONNÉES UTILISÉES

Localisation

Enfin, pour continuer à construire nos dashboards, nous avons récupéré d'autres données de localisation d'oiseaux. Celui-ci permet d'enrichir le premier? De plus, comme il appartient à European Environment Agency, on sait que les localisations des oiseaux sont vraies et sont utilisées par les scientifiques et chercheurs pour l'étude des oiseaux.



MODÉLISATION DES DONNÉES

L'approche de modélisation retenue pour construire la base de données opérationnelle est la modélisation relationnelle. Pour rappel, une BDD relationnelle organise l'information dans des tableaux (tables) reliés entre eux par des relations.

Plusieurs raisons nous ont conduit à faire ce choix :

- L'approche relationnelle est très efficace pour traiter des données structurées. Or dans notre cas, la majorité des données utilisées sont des données structurées, d'où ce choix.
- C'est une technologie avec un bon niveau de maturité. En effet, elle est apparue vers la fin du XXème siècle. Elle a donc vu grandir un écosystème important de logiciels dits RDBMS (Relational Database Management System), qui sont très utilisés et documentés (Oracle, MySQL...). Ce sont des gages de confiance. De plus, au sein de notre équipe de développeurs, c'est l'approche qui nous est la plus familière, facilitant donc le travail de développement.
- Cette approche répond à nos besoins en « scalability » ou la capacité à ajuster les ressources en fonction de la demande. En effet, le nombre d'utilisateurs inscrits à notre application devrait fluctuer de même pour le nombre d'utilisation au quotidien. Par ailleurs, pour envisager l'export de notre application vers d'autres villes et ainsi intégrer de nouvelles zones géographiques et espèces d'oiseaux par exemple, il faut un système capable de gérer le passage à l'échelle. Il nous faut donc prendre en compte les besoins en scalabilité. L'approche relationnelle réagit assez bien au changement d'échelle (« vertical scaling » surtout).
- Enfin, les BDD relationnelles facilitent l'envoie de données vers d'autres formats, on peut penser au format Excel (Xlsx). Cet atout pourra nous être utile notamment pour échanger des informations avec les scientifiques ,plus accoutumés avec le logiciel de la suite Office.



MODÉLISATION DES DONNÉES

Construction de la Base de données

Modèle conceptuel de données (MCD)

Le modèle conceptuel de données (MCD) est une des premières étapes de la construction d'une base de données. Cette étape intervient après avoir déterminé l'ensemble des données nécessaires dans le dictionnaire des données. Elle consiste à réaliser une représentation haut niveau et abstraite des données utiles à notre application. Pour ce faire, on définit les entités qui sont les éléments caractéristiques de notre application ainsi que leurs attributs respectifs qui les définissent.

Dans notre cas, on distingue 3 entités :

- Utilisateur : représente une personne qui vient interagir avec l'application ;
- Oiseau : représente la liste des oiseaux considérés ;
- Zone : représente un repère géographique ;

A noter que nous avons une association ternaire car le point concourant à ces 3 entités est le fait de "reconnaitre". En effet, on pourrait résumer ceci par 1/N utilisateurs reconnaissent 1/N oiseaux dans 1/N zones géographiques.

Pour construire ce MCD, nous nous sommes assurés de respecter les trois formes normales afin de d'éviter la redondance des données et de répondre aux enjeux d'intégrité des données.

Ci-dessous la représentation du MCD :

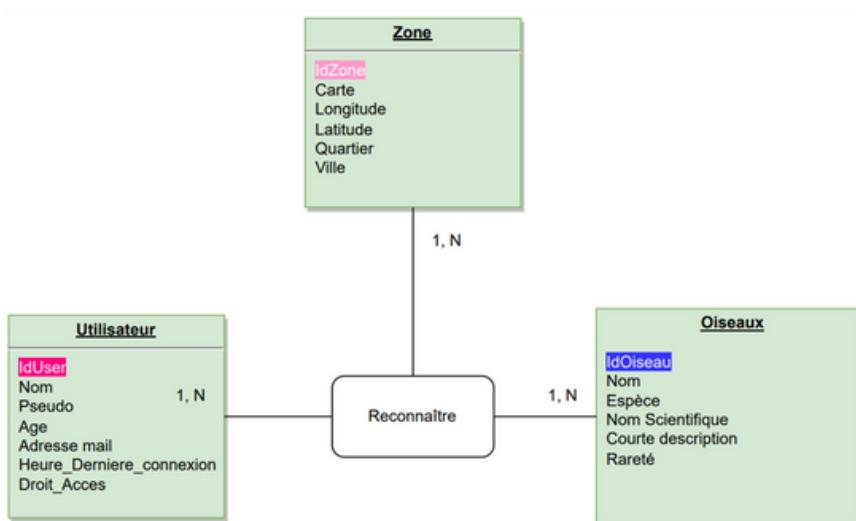


Figure 1: Schéma entité relation de notre BDD opérationnelle

MODÉLISATION DES DONNÉES

Modèle conceptuel de données (MCD)

Le MCD ne peut être implanté comme suit. Il faut ajouter un niveau de détail supplémentaire. C'est pourquoi, on définit le Modèle Logique de Donnée (MLD). Nous avons choisi une modélisation relationnelle. Nous allons donc le construire suivant certains principes : les données seront représentées sous formes de tables, et nous allons respecter 3 règles :

- Règle de transformation des entités en table ;
- Règle de transformation des relations 1 :N ;
- Règle de transformation des relations N :N ;

Le MLD est représenté ci-dessous :

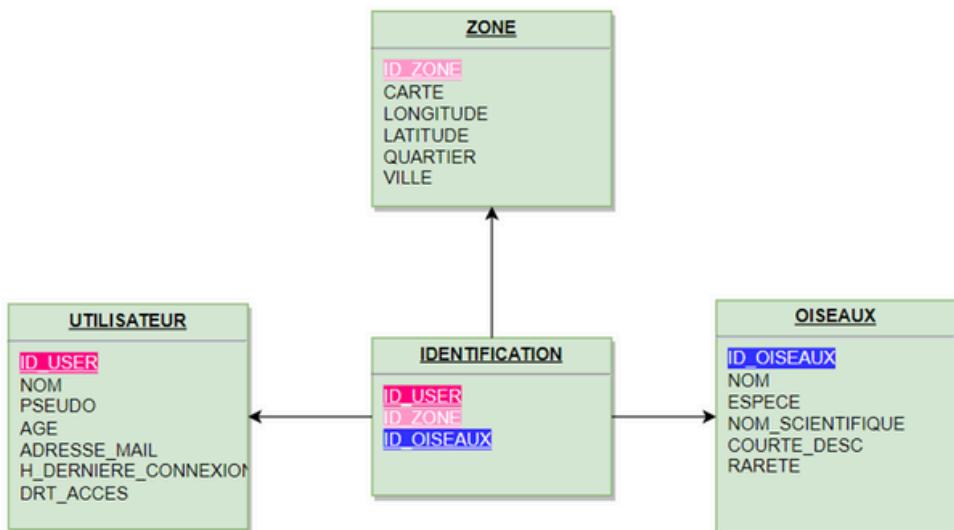


Figure 2: MLD de la base opérationnelle.

On a donc créé les tables suivantes :

- UTILISATEUR (ID_USER, NOM, PSEUDO, AGE, ADRESSE_MAIL, , DRT_ACCEES H_DERNIERE_CONNEXION)

ID_USER → à Clé primaire table UTILISATEUR

- ZONE (ID_ZONE, CARTE, LONGITUDE, LATITUDE, QUARTIER, VILLE)

ID_ZONE → à Clé primaire table ZONE

- OISEAUX (ID_OISEAUX, NOM, ESPECE, NOM_SCIENTIFIQUE, COURTE_DESC, RARETE)

ID_OISEAUX → à Clé primaire table OISEAUX

- IDENTIFICATION (ID_USER, ID_ZONE, ID_OISEAUX)

ID_USER, ID_ZONE, ID_OISEAUX → à Clé primaire composée de la table IDENTIFICATION

MODÉLISATION DES DONNÉES

Modèle physique de données (MPD)

Enfin, la dernière couche de détail consiste à la mise en place réelle de notre base de données. C'est donc le modèle Physique de données (MPD). Le MPD décrit donc le SGBD, le type des variables et l'implémentation pratique des BDD.

Nous utiliserons ici le Système de Gestion de base de données SQL lite afin de pouvoir créer notre base de données. Le langage de programmation est donc le SQL. Ci-dessous, les instructions pour l'implémentation de nos tables :

Pour le cas particulier de la table identification, on construira une clé primaire composite. C'est une clé primaire avec plusieurs champs.

Ci-dessous l'arborescence ainsi que l'implémentation de la base de données associé au MPD.

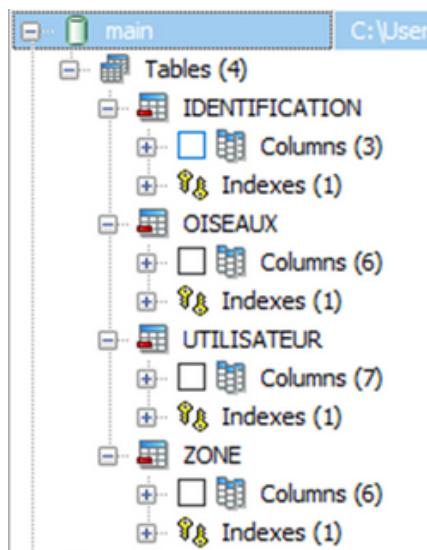
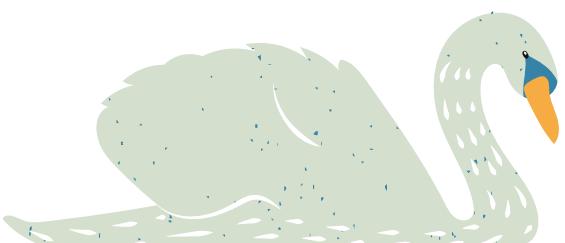


Figure 3: Arborescence du MPD de la BDD opérationnelle

Le code complet associé au MPD est donnée en annexe.



DATA ARCHITECTURE

La data architecture désigne la structure de l'organisation des données de notre application et plus généralement, de notre « entreprise ». Cela regroupe les processus et technologies autour de la transformation, la distribution et de l'utilisation des données par notre application et les utilisateurs qui l'utilisent.

La data architecture de notre projet est représentée par le diagramme suivant que nous allons expliquer plus en détails dans les paragraphes suivants :

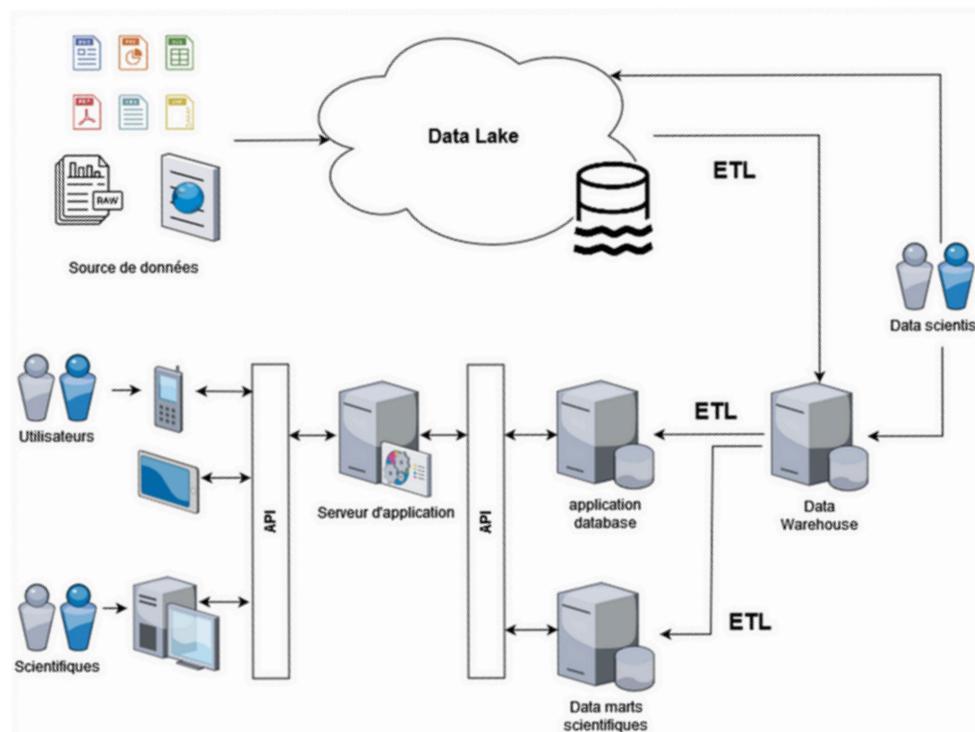


Figure 4 : Data architecture

Avant d'aborder le cœur de la data architecture, nous allons décrire le fonctionnement logiciel de l'application qui utilise une architecture trois tiers classique.

L'architecture à trois tiers est un modèle couramment utilisé en informatique pour concevoir des logiciels. Elle se décompose en trois couches :

DATA ARCHITECTURE

- Couche de présentation : également appelé couche d'interface utilisateur, représenté par les différents appareils du diagramme. Ce niveau est responsable de la présentation visuelle de l'application pour l'utilisateur final. Il gère les interactions entre l'utilisateur et l'application, en affichant les informations et en permettant à l'utilisateur d'entrer des données. Dans le cas de l'application sur téléphone, la couche de présentation permettra à l'utilisateur d'enregistrer et de transmettre les sons enregistrés mais également de visualiser ses résultats.
- Couche de traitement : également appelé couche logique, représenté par notre serveur d'application. Ce niveau est responsable du traitement des données et de la logique métier de l'application. Il s'agit du cœur de l'application, où les données sont manipulées et où les calculs sont effectués. C'est également là que ce gère les droits d'accès aux données en fonction des différents profils d'utilisateur (joueurs, scientifiques, développeurs). Elle a donc une fonction de sécurité dans notre application. Elle gère toutes les requêtes provenant des utilisateurs (par le biais de la couche de présentation). Les deux façades API du diagramme font partie de la couche de traitement et permettent le lien entre les trois couches de l'architecture.
- Couche de la persistance des données : également appelé couche de stockage, représenté par le serveur de base de données de l'application. Ce niveau est responsable de la gestion des données stockées par l'application. Il stocke les données de manière persistante, ce qui signifie que les données sont conservées même si l'application est arrêtée ou redémarrée. En termes plus simples, c'est la base de données de notre application.

En plus de ses avantages en termes de sécurité grâce à la séparation de chaque partie, l'architecture à trois niveaux permet aux développeurs de segmenter les différentes fonctions de l'application en couches distinctes, ce qui simplifie la maintenance, la mise à l'échelle et la mise à jour de l'application.

Les machines représentées dans le schéma ne sont pas forcément des machines physiques, mais des fonctions. En réalité, les bases de données et les serveurs d'applications seront probablement redondants et il y en aura plusieurs pour répondre à la charge de travail qui sera de plus en plus importante avec l'augmentation des utilisateurs.

DATA ARCHITECTURE

Nous avons fait des choix précis dans notre architecture de data en utilisant un data lake, un data warehouse et plusieurs data marts en fonction de nos besoins.

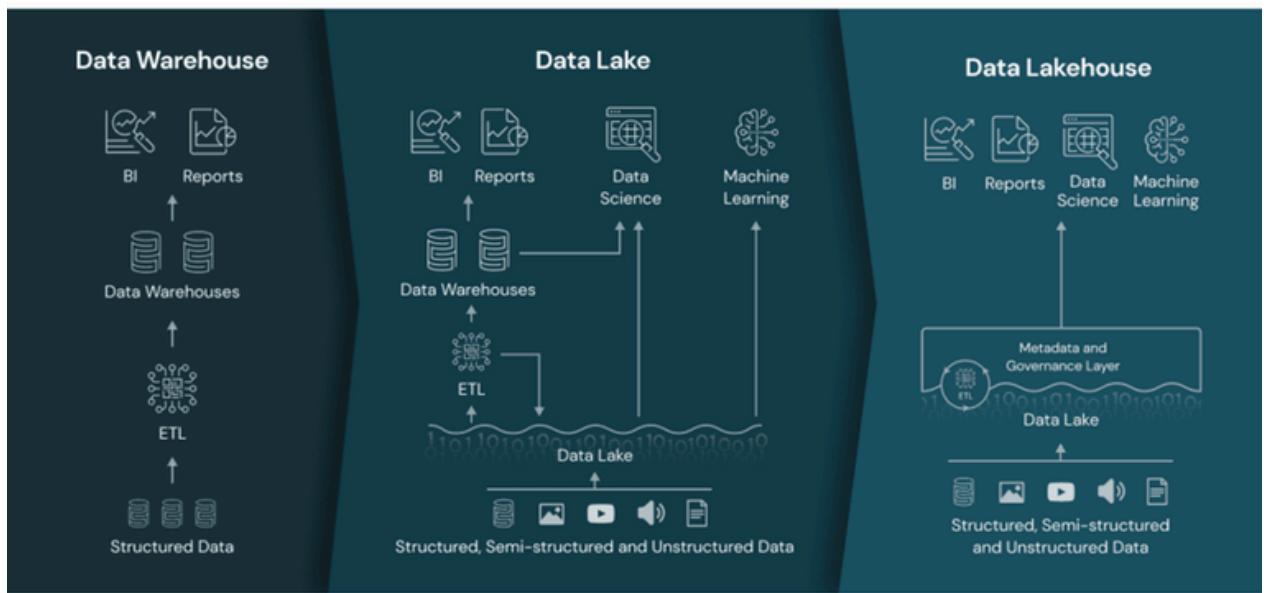


Figure 5 : Data architecture

Concernant le data lake, il permet de stocker des données dans leur forme brute, sans aucune transformation ni structuration préalable. Cela signifie que les données peuvent être stockées en vrac et dans leur format d'origine, qu'il s'agisse de données structurées, semi-structurées ou non structurées.

Cela va être utile principalement pour les data scientists qui vont travailler sur nos modèles d'IA et l'analyse des données. En effet, cela leur permet d'accéder à une grande quantité de données brutes et non traitées, ce qui peut être très utile pour l'exploration de données et la création de. Les données brutes peuvent contenir des informations qui ne sont pas disponibles dans les données transformées et structurées, ce qui peut aider les data scientists à découvrir des relations inattendues ou à identifier de nouveaux motifs dans les données.

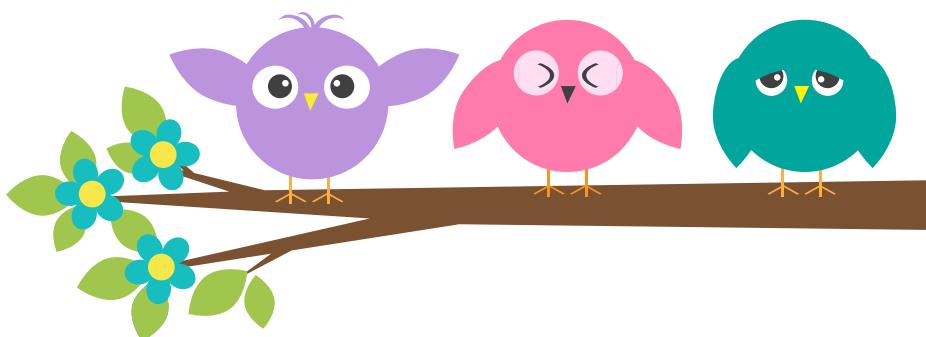
De plus, le data lake possède un avantage pour notre organisation métier. Le stockage des données brutes dans un data lake peut aider les entreprises à éviter des pertes de données potentielles. En effet, lorsqu'on transforme et structure les données, on risque de perdre des informations précieuses qui n'ont pas été identifiées au préalable. En conservant les données brutes dans un data lake, les entreprises peuvent réduire ce risque et préserver toutes les données disponibles.

DATA ARCHITECTURE

Notre data lake alimente notre data warehouse. Ce dernier stocke des données structurées, organisées et transformées pour une utilisation par l'ensemble de l'organisation. Les données stockées proviennent de différents formats, mais elles sont toutes transformées et structurées selon notre MCD par le biais des ETL.

Pour finir nous avons choisi de diviser nos bases de données en deux data marts correspondant à nos deux grands besoins utilisateurs de la plateforme : une pour les joueurs et une pour les scientifiques. Effectivement par définition le datamarts est conçu pour répondre aux besoins spécifiques d'un secteur d'activités de l'organisation. Ils sont créés en sélectionnant et en agrégeant les données pertinentes à partir de notre datawarehouse. Ce sont des sortes de sous-ensemble du data warehouse spécifiques à l'utilisation dont nos différents profils utilisateur peuvent faire.

En fournissant des informations spécifiques cela permettra une analyse plus rapide et plus précise des données par notre système.



CAS D'USAGES

Cas d'usage 1 – Identifier un oiseau à partir de son chant

Aujourd'hui - Limites des applications d'identification de musique

Shazam est une application très populaire d'identification de musique. Sa base de données de millions de chansons lui permet de reconnaître extraordinairement rapidement une chanson spécifique. Cependant, son fonctionnement est limité à l'identification de chansons enregistrées, et ne peut pas identifier les chants d'oiseaux ou d'autres sons de la nature.

Le défi - Identification des chants d'oiseaux

Identifier les chants d'oiseaux est une tâche difficile car chaque chant est unique, même pour une même espèce d'oiseau. Les chants présentent des variations régionales et individuelles, et sont influencés par les conditions environnementales changeantes. Pour résoudre ce problème complexe, on développe des systèmes d'identification des chants d'oiseaux basés sur les algorithmes de Machine Learning.

La solution – Un système d'intelligence artificielle universel et objectif de description des chants

Le Machine Learning est une technique d'apprentissage automatique qui utilise l'apprentissage supervisé pour entraîner un modèle à identifier les caractéristiques communes à des données complexes, telles que les chants d'oiseaux. Les caractéristiques sont identifiées en ajustant des centaines de paramètres analytiques pour chaque chant, y compris des paramètres qui ne sont pas détectables par l'oreille humaine. En utilisant cette méthode, le système résout le problème de l'unicité des chants de chaque oiseau : il est capable d'identifier des variations régionales ou individuelles dans les chants et de les associer tout de même à des espèces spécifiques.

Le fonctionnement - le Machine learning

Le système d'identification des chants d'oiseaux fonctionne en enregistrant un chant et en le soumettant à un algorithme qui analyse les caractéristiques acoustiques du chant. L'algorithme utilise ensuite ces caractéristiques pour trouver des similitudes avec d'autres chants, même s'ils présentent des variations régionales ou individuelles. En utilisant ces similitudes, le système peut identifier des chants qu'il n'a jamais rencontrés auparavant, grâce à une analyse en temps réel des nouvelles données sonores. En pratique, on va utiliser des techniques de reconnaissance automatique de son. Ces méthodes consistent à donner en entrée à un algorithme de reconnaissance une « image acoustique » du son. Ici on utilisera un diagramme temps fréquence qui représente les fréquences du son en fonction du temps. L'algorithme sera ensuite capable de classifier les différents sons. A partir de cette classification, on sera capable de prédire l'espèce de l'oiseau entendu.

CAS D'USAGES

Cas d'usage 2 – Étendre l'application à d'autres territoires

Aujourd'hui - Les limites du territoire toulousain

L'application sera limitée tout d'abord au territoire toulousain, en conformité avec les exigences de l'appel à projet « Toulouse Ville Verte ». □

Le défi – La création de territoires spécifiques pour l'adaptation aux zoos et réserves

L'application ayant vocation à grandir et à couvrir de nouveaux territoires sur sa carte, il est indispensable qu'il soit aisément possible d'ouvrir de nouveaux territoires. L'application doit pouvoir également héberger des cartes spécifiques, comme des parcs zoologiques ou des réserves ornithologiques, accessibles uniquement par les visiteurs de telles structures. L'application doit pouvoir donc ouvrir d'autres territoires « publics » (accessibles à tous les utilisateurs de l'application, comme l'élargissement aux montagnes, aux plages, aux chemins de randonnée), et d'autres territoires spéciaux (accessibles uniquement sous certaines conditions).

La solution – Un système d'intelligence artificielle universel et objectif de description des chants

Nous allons créer un système de carte privée. Ces cartes privées, créées en collaboration avec le client, devront avoir un accès réglementé afin de faire respecter le caractère privé de ces cartes. Les données de ces cartes ne seront pas versées dans la base de données à destination des ornithologues. L'accès à la carte et à l'oiseaudex spécifique se fera uniquement sur le territoire concerné.

Le fonctionnement - le Machine learning

Ce système de carte privée suppose la création de tables temporaires dans notre base de données « Cartographie ». Lorsque l'appareil d'un utilisateur arrive sur une donnée géographique précise (on peut également penser à un code offert par le lieu, un QR code...), il débloque alors l'expérience du lieu sur lequel il est, c'est-à-dire qu'une nouvelle table et oiseaudex temporaires se créent dans notre architecture. Les données des oiseaux trouvés durant le temps imparti à l'expérience exclusive sont versées dans l'oiseaudex permanent de l'utilisateur Joueur, mais ne seront pas versées dans la base de données destinée aux utilisateurs Scientifiques

CAS D'USAGES

Cas d'usage 3 – Créer une base de données d'observations d'oiseaux

Aujourd'hui – Absence de données en temps réel des populations d'oiseaux

Dans un contexte de changement climatique et de déclin rapide et préoccupant de la biodiversité au niveau local et mondial, il est primordial pour les ornithologues et écologues d'avoir accès à des données sur la répartition précise d'espèces d'oiseaux. Ces informations, si elles sont à jour et de qualité, permettront de compléter les données scientifiques existantes afin de pouvoir monitorer en temps quasi-réel les populations d'oiseaux.

Le défi – Une base de données riche et adaptée aux besoins des ornithologues

Les scientifiques doivent recevoir des données :

- En temps quasi-réel : une
- Datées
- Localisées : avec une précision à la « zone »
- Adaptées : le format doit être adapté aux technologies utilisées par les ornithologues

La solution – Une base de données accessible et structurée

Nous avons besoin de créer une base de données contenant des données agrégées et structurées pour répondre aux besoins spécifiques de nos utilisateurs à profil Scientifiques. Notre base de données contiendra les données collectées par les joueurs, nettoyées et transformées pour leur utilisation.

Le fonctionnement – Data Mart, Data Warehouse & ETL

Les données seront collectées et stockées dans une base de données centrale, le "Data Mart". Cette base de données sera conçue pour optimiser sa prise en main par les scientifiques, afin qu'ils puissent facilement accéder aux données pertinentes pour leurs recherches. Le data warehouse peut contenir plusieurs data marts, chacun spécifique à un domaine métier. Les données sont stockées dans le Data Mart « Scientifique » à l'aide d'un outil ETL (Extract, Transform, Load) : les données sont donc extraites de nos sources, transformées puis nettoyées avant d'être chargées dans le Data Mart. Ce type d'architecture garantit que les données sont uniformes et fiables, ce qui facilite l'analyse ultérieure des données – un critère non-négligeable par nos utilisateurs finaux. Les scientifiques peuvent alors accéder au Data Mart en utilisant leurs outils de data analyse pour exploiter les données selon leurs besoins.

CAS D'USAGES

Cas d'usage 4 – Permettre aux utilisateurs de localiser les oiseaux en un coup d'oeil

Aujourd'hui – Des observations éparses

Les communautés d'ornithologues amateurs regroupent leurs visualisations par un système de commentaires, dispersés sur une myriade de sites et de blogs. Les observations sont généralement visuelles (jumelles et appareil photo à l'appui) et les informations dispersées, non structurées. Il n'existe pas de base de données centralisées en France avec des informations vérifiées, fiables, anonymes et localisées des oiseaux. Visualiser les endroits où d'autres utilisateurs ont enregistré certaines espèces d'oiseaux est crucial pour les joueurs. C'est une incitation à la découverte et un rouage essentiel du gameplay du jeu que nous développons.

Le défi – La fiabilité, l'ergonomie, le format

Il est important que les observations puissent être mises en commun entre les joueurs, afin que les lieux à fort probabilité d'observation apparaissent facilement aux joueurs. Notre application doit fournir aux joueurs une manière fiable et ergonomique de voir en un instant où se trouvent les lieux où certains oiseaux ont une bonne probabilité d'être entendus. Pour éviter de lancer l'application avec une carte vide (car il n'y aurait pas encore d'observation), nous avons récupéré une base de données publique, qui quoiqu'un peu datée, localise des populations d'oiseaux observées par des ornithologues.

La solution – Une carte interactive

Nous avons choisi une solution sous forme de carte interactive : la carte recense les concentrations d'oiseau avec un dégradé de couleur. L'utilisateur peut également sélectionner une espèce d'oiseau, et les apparitions de cette espèce en particulier apparaîtront sous forme de cercles de couleur correspondant à une certaine probabilité de les trouver. La carte se met à jour tous les matins à 3h avec les observations de la veille. Elle offre la possibilité aux utilisateurs de manipuler la carte et de la centrer sur une localisation actuelle ou sélectionnée.

CAS D'USAGES

Le fonctionnement – Data visualisation

Notre solution se présentant sous la forme d'une application avec des utilisateurs, ceux-ci pourront d'une certaine façon collaborer et aider à remplir les cartes. Les oiseaux repérés par les utilisateurs sont répertoriés dans un jeu de données. Ce jeu de données se traduira principalement sous la forme d'une carte interactive avec une légende en dégradé. Plus il y a d'oiseaux sur un lieu, plus la zone sera foncée. On peut également filtrer la carte sur une seule espèce d'oiseau. Ainsi, on pourra voir sur quelle zone cette espèce apparaît le plus souvent. Enfin, cela permettra également aux scientifiques de suivre le cycle migratoire des oiseaux grâce aux utilisateurs.



ANNEXES

Dashboard :

Carte de base :

Pour notre application, nous possédons un certain nombre de graphiques informatifs. Ceci est la synthèse totale pour la ville de Toulouse et ses alentours. C'est sur l'ensemble des oiseaux identifiés dans chaque lieu que notre carte se positionne automatiquement.

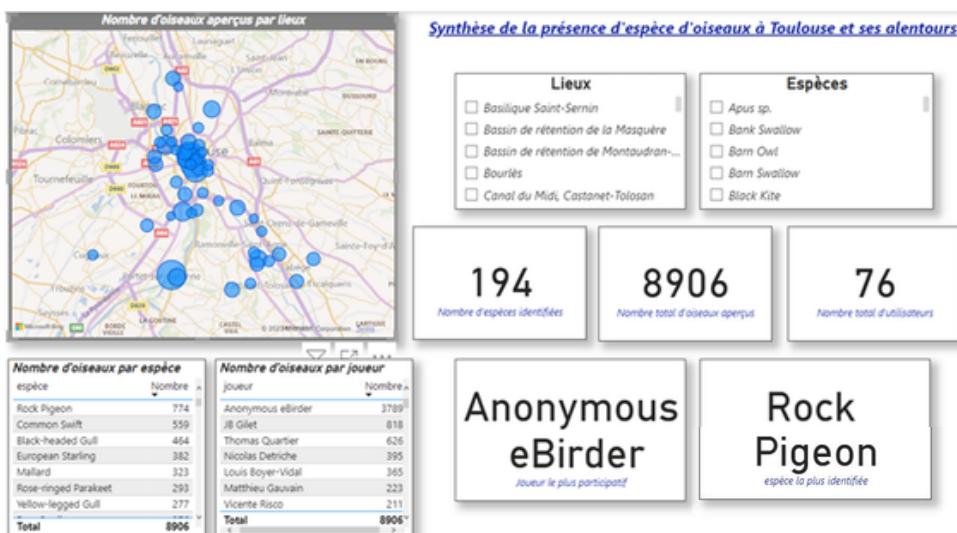


Figure 6 : Dashboard en accueil

Lorsque que nous passons notre curseur sur l'une des bulles de notre carte, nous pouvons voir les informations associées au lieu.

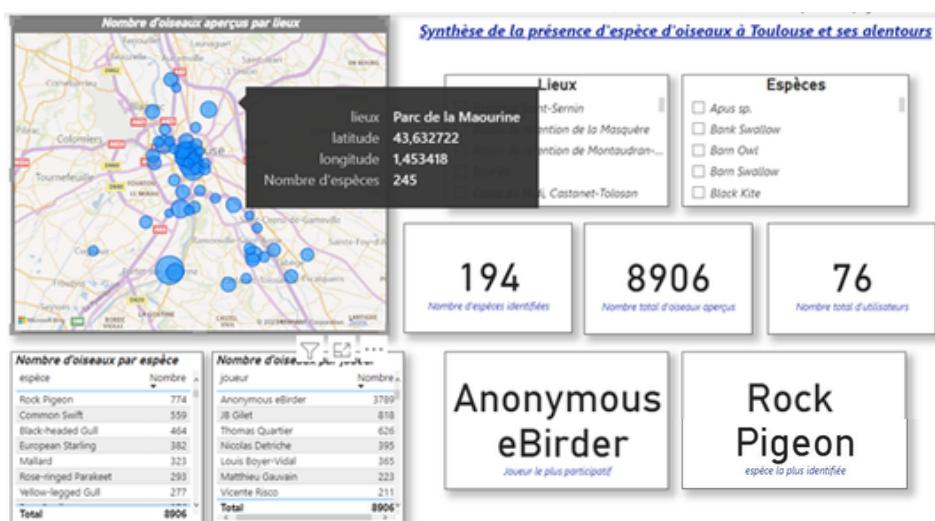


Figure 7 : pointeur sur un lieu

ANNEXES

Filtration sur un lieu :

Afin d'avoir des informations sur un lieu en particulier, nous pouvons effectuer une filtration de deux façons différentes : on peut cliquer sur l'une des bulles de la carte. Celle-ci restera foncée et les autres deviendront plus claires. Les dashboard se mettront ensuite à jour en fonction de la sélection.

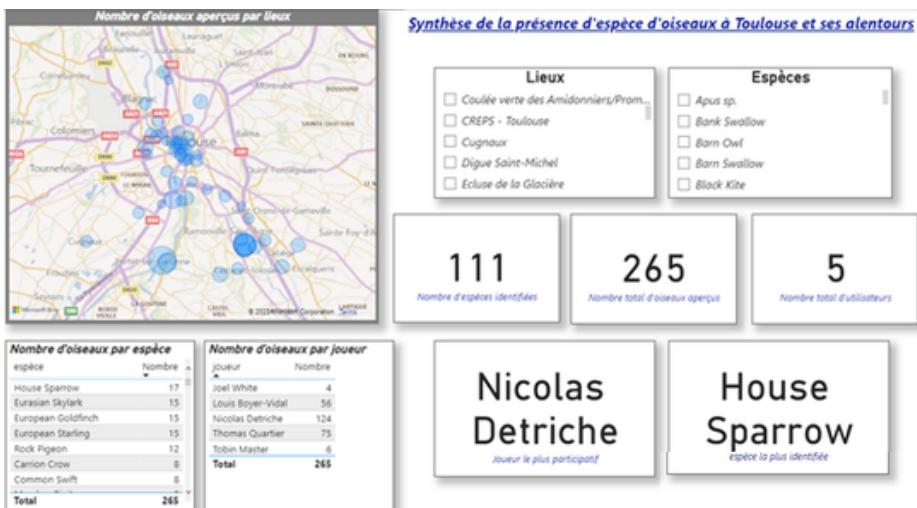


Figure 8 : Filtration via les bulles

Nous pouvons également effectuer une filtration en cliquant sur l'une des localisations dans la liste déroulante se nommant lieux.

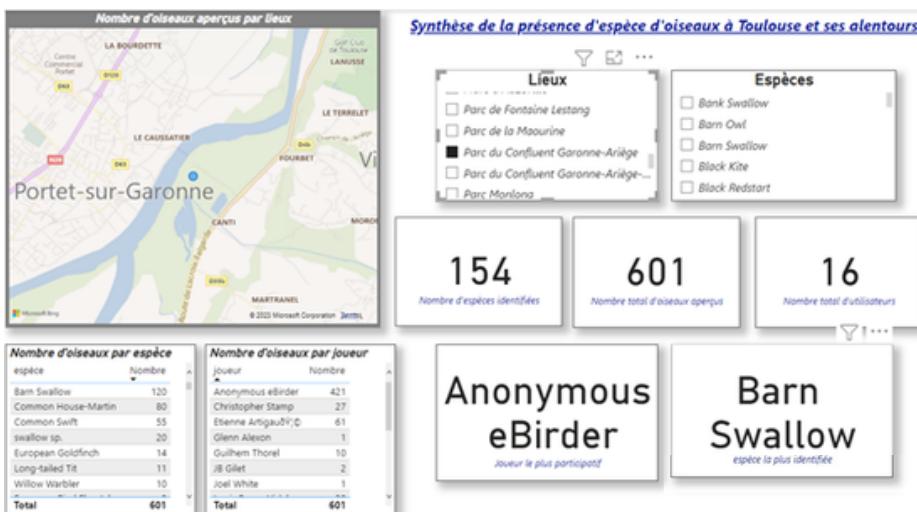


Figure 9 : Filtration avec la liste déroulante

ANNEXES

Filtration par espèce :

Nous pouvons également filtrer nos informations en fonction de l'espèce visée. Nous pouvons par exemple sélectionner une des espèces de la liste déroulante. Les autres graphiques se mettront à jour en fonction de la sélection

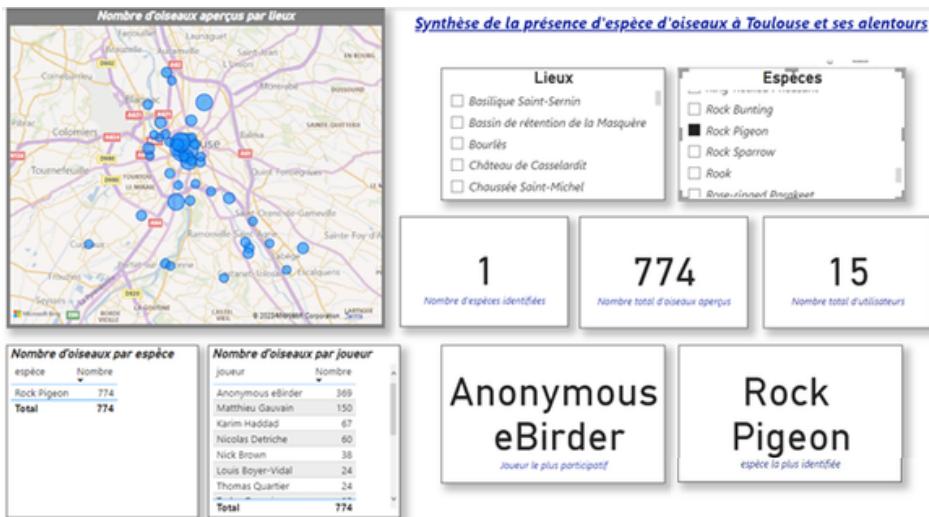


Figure 10 : Filtration via la liste déroulante des espèces

Nous pouvons également effectuer une filtration via les tableaux en bas à gauche. Pour cela, il suffit de sélectionner l'une des espèces. Les autres graphiques se mettront également à jour en fonction de la sélection.

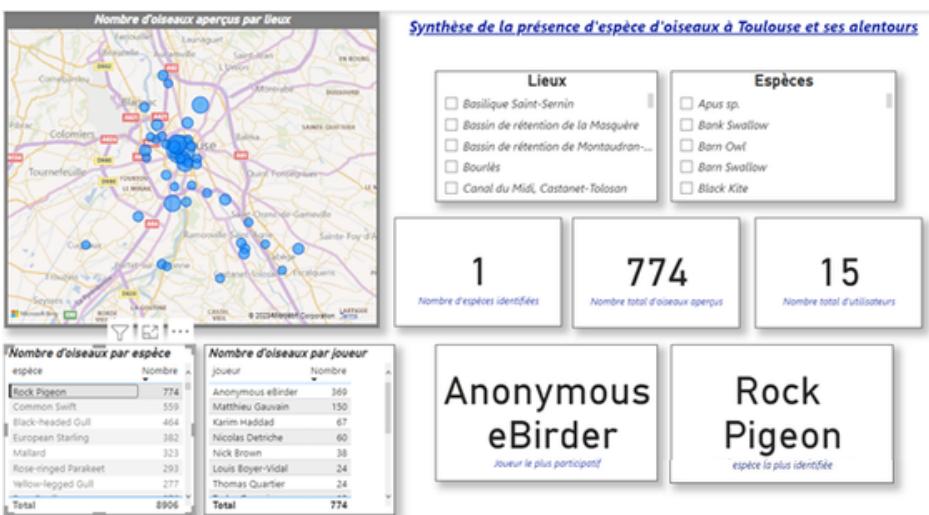


Figure 11 : Filtration avec le tableau

ANNEXES

Filtré par utilisateur :

On peut également filtrer les informations selon les joueurs. Cela peut nous indiquer si un utilisateur utilise beaucoup notre application. Pour cela, il suffira de cliquer sur un nom d'utilisateur dans le tableau en bas à gauche. Les visualisations se concentreront alors sur les données de l'utilisateur, qui peut être anonyme ou non.

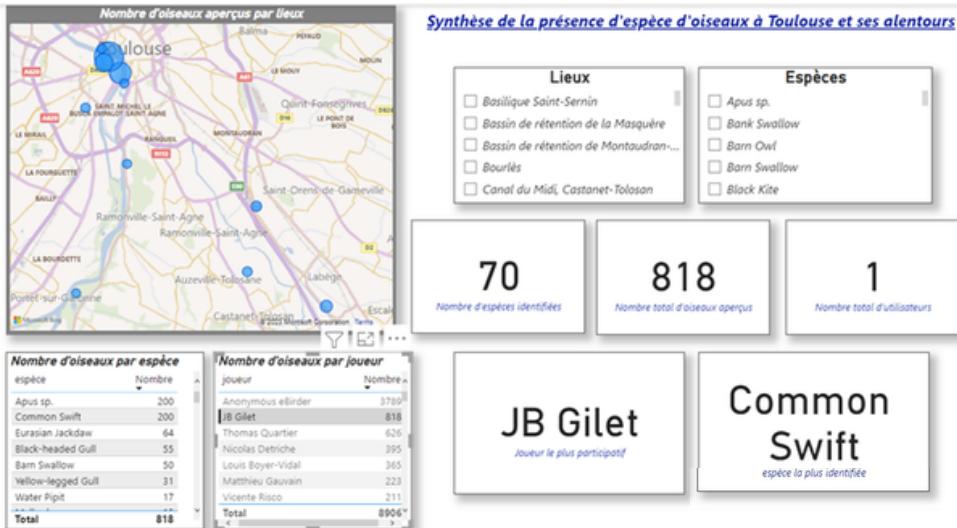


Figure 11 : Filtration en fonction de l'utilisateur

Code modélisation MPD :

```
/* Table Utilisateur */
CREATE TABLE UTILISATEUR(
    ID_USER INT AUTO_INCREMENT PRIMARY KEY,
    NOM VARCHAR(50),
    PSEUDO VARCHAR(50),
    AGE INT,
    ADRESSE_MAIL VARCHAR(100),
    H_DERNIERE_CONNEXION TIMESTAMP,
    DRT_ACCEES INT
);

/* Table Oiseaux */
CREATE TABLE OISEAUX(
    ID_OISEAUX INT AUTO_INCREMENT PRIMARY KEY,
    NOM VARCHAR(50),
    ESPECE VARCHAR(50),
    NOM_SCIENTIFIQUE VARCHAR(70),
    COURTE_DESC VARCHAR(150),
    RARETE INT
);

/* Table Zone */
CREATE TABLE ZONE(
    ID_ZONE INT AUTO_INCREMENT PRIMARY KEY,
    CARTE VARCHAR(50),
    LONGITUDE INT,
    LATITUDE INT,
    QUARTIER VARCHAR(50),
    VILLE VARCHAR(50)
);
```

ANNEXES

```
/* Table IDENTIFICATION */
CREATE TABLE IDENTIFICATION(
    ID_USER_IDT INT,
    ID_ZONE_IDT INT,
    ID_OISEAUX_IDT INT,
    /* Fusionne les 3 clés primaires en 1 clé composée */
    PRIMARY KEY (ID_USER_IDT, ID_ZONE_IDT, ID_OISEAUX_IDT),
    FOREIGN KEY (ID_USER_IDT) REFERENCES UTILISATEUR(ID_USER),
    FOREIGN KEY (ID_ZONE_IDT) REFERENCES OISEAUX(ID_OISEAUX),
    FOREIGN KEY (ID_ZONE_IDT) REFERENCES ZONE(ID_ZONE)
);
```

Figure 12 : Implémentation finale de la BDD opérationnelle

Le code ci-dessus représente l'implémentation de notre BDD afin qu'elle puisse être opérationnelle.

