
Trombinoscope

20 décembre 2021

Auteurs :

Nathan FAUDEIL
Quentin GUICHOUX
Fabio CASSIANO



Table des matières

1	Arbre de décision	2
2	Guide utilisateur	3
2.1	Captures d'écran de la solution	3
2.2	Explication du fonctionnement	3
2.2.1	<i>Version 1.0</i>	3
2.2.2	<i>Version 2.0</i>	4
3	Présentation du schéma de la base de données	5
3.1	Les difficultés	5
3.2	Les solutions	5
4	Présentation du code Python	6
4.1	Les difficultés	6
4.2	Les solutions	6
5	Apprentissage	6
5.1	Fabio	6
5.2	Quentin	6
5.3	Nathan	6
6	Conclusion	7
6.1	Que souhaitez-vous apprendre sur SQL et Python ?	7
6.2	Quelles évolutions envisagez-vous sur l'application ?	7
6.2.1	<i>Version 1.0</i>	7
6.2.2	<i>Version 2.0</i>	7

1 Arbre de décision

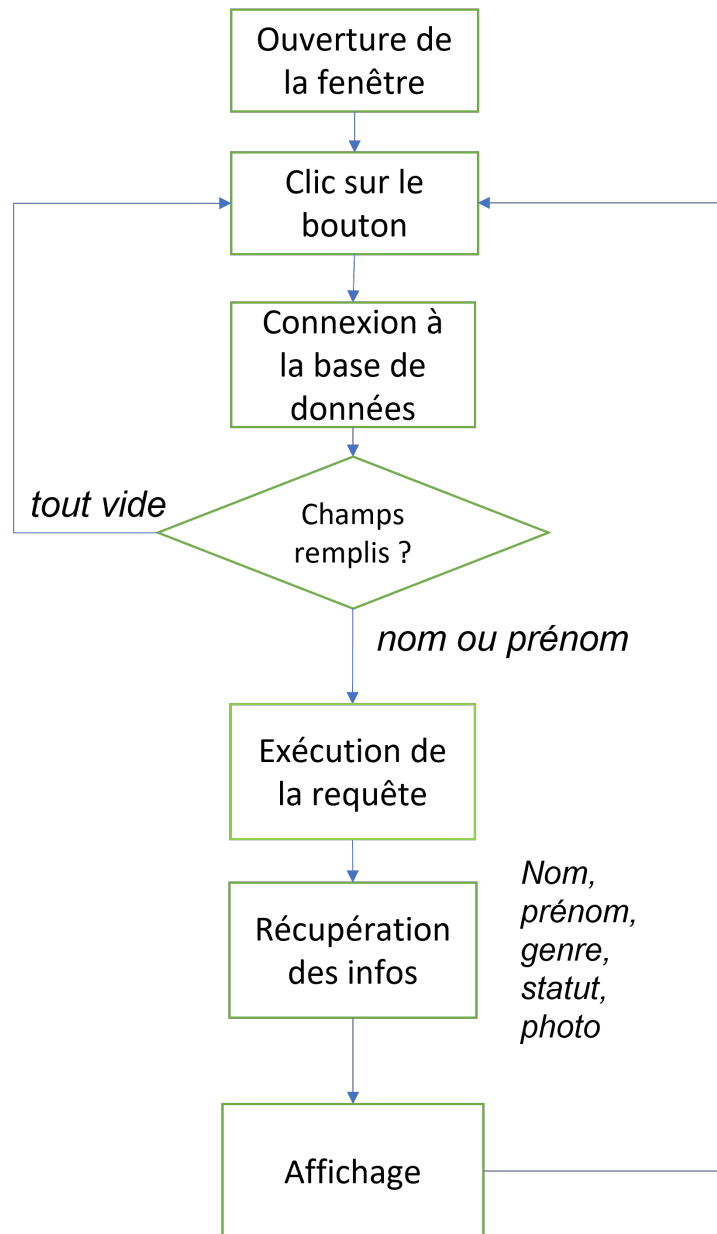
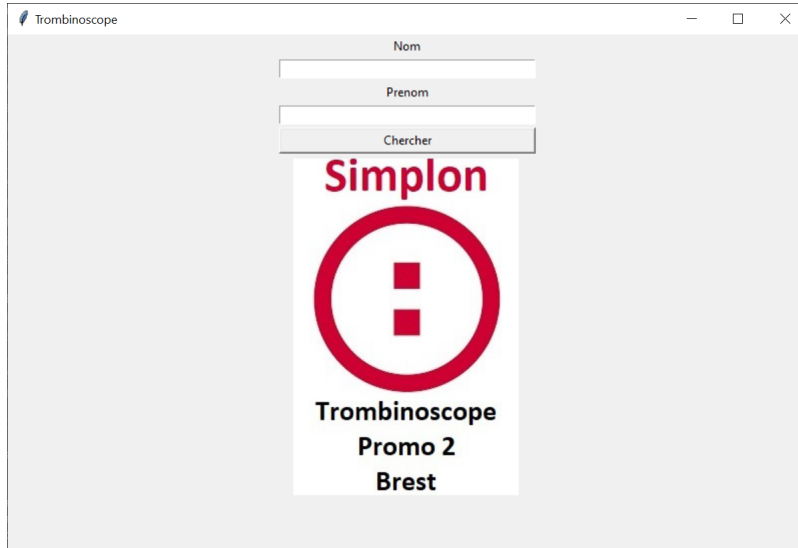


FIGURE 1 – Arbre de décisions de l'application trombinoscope.

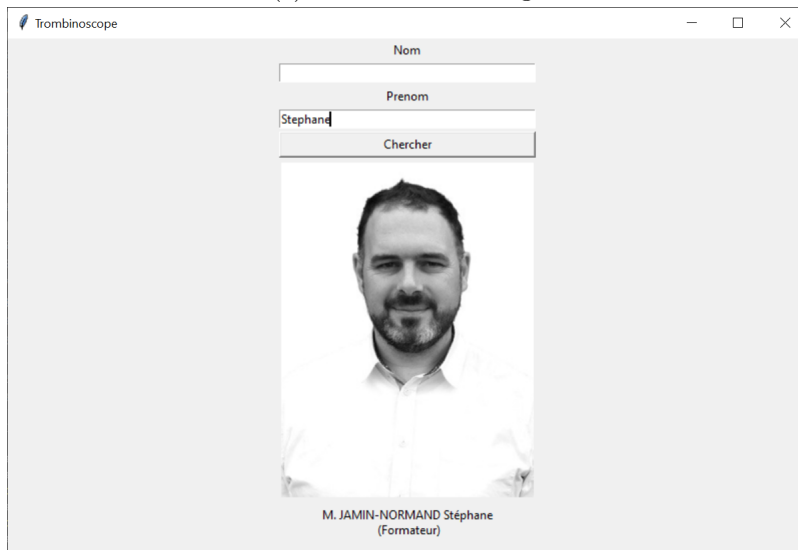
Les champs de saisie correspondent au nom, prénom, genre, statut, photo et l'interrogation de la base de données jusqu'à l'affichage se fait de manière invisible pour l'utilisateur

2 Guide utilisateur

2.1 Captures d'écran de la solution



(a) Fenêtre de démarrage



(b) Recherche d'une personne par le prénom

FIGURE 2 – Interface de l'application trombinoscope

2.2 Explication du fonctionnement

2.2.1 Version 1.0

Le trombinoscope que nous avons créé fonctionne par la recherche du **Nom** ou du **Prénom** de la personne. Pour se faire un champ de saisie est mis à disposition de l'utilisateur pour chacun de ces deux paramètres (Fig. 2a). Le bouton **recherche** permet

ensuite d'activer une requête SQL qui utilise les champs remplis pour retrouver dans la base de données la personne correspondante et de récupérer les informations suivantes :

1. Nom de la personne
2. Prénom de la personne
3. Photo associée
4. Genre associé (M., Mme., Autre)
5. Statut associé (Étudiant P2, Formateur, Chargé de Formation)

Ces informations sont ensuite renvoyées sur l'interface utilisateur, en affichant la photo associée (Fig. 2b), puis les informations de la personne sous la forme :

Genre Nom Prénom (Statut)

2.2.2 Version 2.0

La seconde version du trombinoscope a subi une légère modification de l'interface. La partie recherche se trouve maintenant à gauche de la partie affichage.

Il est maintenant possible de réaliser une recherche par le genre ou le statut. Cette fonctionnalité se matérialise par deux *listebox* en dessous du champ prénom.

Coté affichage il est maintenant possible d'afficher plusieurs photo ainsi que les informations associés.

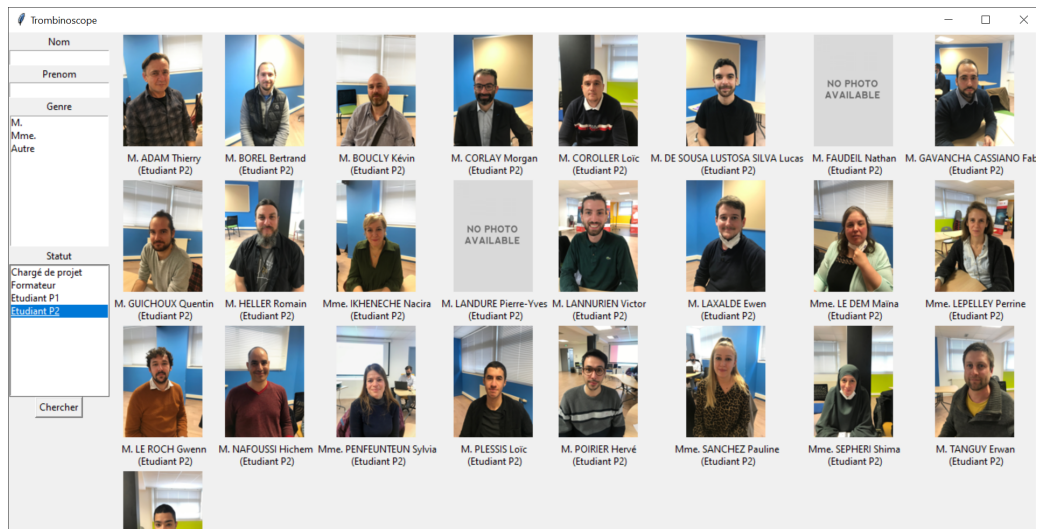


FIGURE 3 – Trombinoscope version 2.0

Représentation de l'interface du trombinoscope version 2.0 avec la recherche de toute la promotion 2 (Les *Breizhmeiz*)

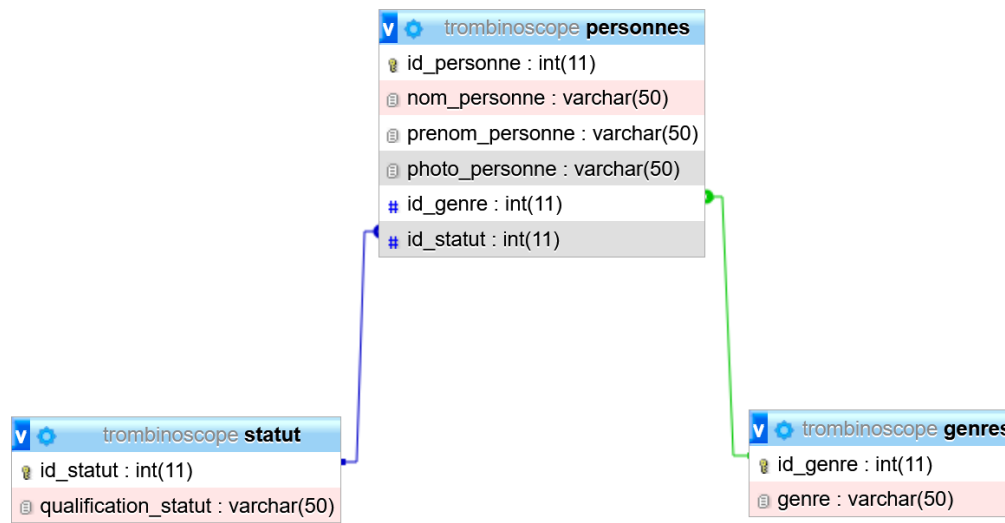


FIGURE 4 – Schéma de la base de données trombinoscope

3 Présentation du schéma de la base de données

3.1 Les difficultés

L’objectif de la base de données était de créer un document contenant les informations des différentes personnes présentes à la formation.

La création du schéma relationnel de la base de données n’a pas posé de réels problèmes, la base étant simple et de petite taille.

3.2 Les solutions

Le choix de renseigner le genre et le statut de la personne à rendu obligatoire la création de plusieurs tables. Une personne ne pouvant avoir qu’un seul genre et qu’un seul statut, des tables intermédiaires auraient été superflues, et nous avons donc créé trois tables :

1. *Personnes*, contenant les noms, prénoms et identifiants des apprenants et formateurs
2. *Statut*, contenant les différents rôles qu’une personne peut avoir (apprenant ou formateur)
3. *Genres*, contenant les différents genres possibles (M., Mme., autre). Pour lier ces différentes tables, nous avons créé des identifiants dans chacune d’entre-elles, tous mis en clé primaire, puis nous avons rajouté `id_genre` et `id_statut` en clés étrangères dans la table *Personnes* pour pouvoir accéder à toutes les informations.

4 Présentation du code Python

4.1 Les difficultés

Nous avons rencontré une difficulté lors de l’affichage des images. En effet, lors de la recherche d’une personne, la photo apparaissait mal centrée dans le cadre dédié à son affichage.

4.2 Les solutions

Nous voulions afficher les photos au centre du canevas créé à cette fin, afin qu’elles soient centrées. Les dimensions du canevas étaient sous la forme *hauteur/largeur*. Notre erreur résidait dans le fait que nous avions indiqué dans le code que nous voulions projeter le centre de la photo en position (0,0) du canevas. Cela faisait donc apparaître le centre de l’image dans le coin nord/ouest de ce dernier, occasionnant le décalage que nous ne comprenions pas. Une fois l’erreur réalisée, nous avons tout simplement changé la position (0,0) en (*hauteur/2*, *largeur/2*), et le problème était réglé.

5 Apprentissage

5.1 Fabio

Pour ma part j’avais déjà quelques notions de SQL en ligne de commande. Grâce à ce projet j’ai pu consolider mes connaissances en création de base de données, et j’ai également pu découvrir l’utilisation de phpmyadmin pour l’administration de base de données avec une interface graphique. Pour Python, j’avais déjà utilisé ce langage pour des traitements scientifiques. Ici j’ai pu découvrir Tkinter, et ainsi la création d’interfaces sous ce langage. Les compétences développées par le biais de ce projet me permettront d’améliorer les futurs projets sur lesquels je travaillerai.

5.2 Quentin

Connaissant déjà relativement bien le SQL et phpmyadmin j’ai surtout pu découvrir de manière plus poussée Python. Avant ce projet, je ne m’en servais que d’une manière assez basique sans trop utiliser de bibliothèques. La découverte de TKinter me permet de mieux utiliser ce langage et de faire de petits projets personnels plus aboutis, et dans l’avenir utiliser ces connaissances en situation professionnelle.

5.3 Nathan

Au début du projet, je n’avais qu’une compréhension limitée de Python, le reste m’était inconnu. En créant le trombinoscope, j’ai pu découvrir SQL, phpmyadmin, ainsi que l’utilisation de la bibliothèque Tkinter. J’ai surtout eu la chance d’être bien accompagné par mes camarades qui connaissaient davantage cet environnement. L’utilisation générale de tous ces outils n’est bien évidemment pas encore totalement acquise, mais cela m’apparaît moins étranger. Je pense que toutes ces compétences pourront me servir pour mieux cerner les implications liées au domaine de l’intelligence artificielle, pour la suite de la formation mais également pour mon projet professionnel.

6 Conclusion

6.1 Que souhaitez-vous apprendre sur SQL et Python ?

La question nous semble peut-être encore prématurée, étant donné que nous ne connaissons pas toutes les implications de SQL et Python dans le domaine de l'intelligence artificielle. Le trombinoscope était donc une bonne base de départ dans l'apprentissage de ces outils, nous verrons au travers des autres projets ce qui nous intéresse davantage dans leurs usages.

6.2 Quelles évolutions envisagez-vous sur l'application ?

6.2.1 Version 1.0

Nous avons réfléchi sur plusieurs améliorations du trombinoscope :

1. Faire en sorte que le bouton "rechercher" soit activé par la touche "entrée" du clavier pour accélérer la recherche et ne pas utiliser la souris.
2. Certaines personnes de la formation ont le même nom de famille où le même prénom. Une amélioration possible serait, lors d'une recherche par nom où prénom, d'afficher la photo des deux personnes en même temps, ce qui n'est pas le cas dans le trombinoscope actuellement. (*Effectif dans la version 2.0*)
3. Faire en sorte d'afficher plus de critères de sélection. Par exemple chercher par genre ou par statut, et dans ce cas prévoir une amélioration de l'affichage et une aide à la navigation entre les différents profils. (*Effectif dans la version 2.0*)
4. Une interface homme-machine (IHM) plus aboutie et polie pourrait également être une bonne chose. Nos connaissances de TKinter étant limitées, notre interface reste relativement basique.
5. En cas d'éventuelle mise en ligne du projet une protection contre les attaques par injection SQL, bien que n'ayant pas de données sensibles ou de connexion, un "nom');DROP TABLE Personne—" serait à éviter. Dans le cadre du projet une protection contre ces attaques n'est pas utile mais pour des projets futurs ou des mise en ligne, il serait utile d'en incorporer une.

6.2.2 Version 2.0

D'autres améliorations ont été réfléchies pour la version 2.0 du trombinoscope :

1. Pouvoir effectuer des recherches en combinant les champs. Par exemple si une femme et un homme avait le même nom de famille on pourrait en afficher un seul en sélectionnant le genre associé.
2. Adapter la taille des photos afficher à l'espace disponible. Ainsi que permettre de défiler vers le bas quand les photos sont trop nombreuses. Pour le moment toutes les photos ont une taille fixe, et suffisamment petite pour permettre d'afficher une grande quantité d'image.