

# Application de classification de fleurs

Analyse fonctionnelle  
présentée à  
Jonathan Rousseau

par  
Yann Roubeau  
et  
Thomas Saudemont

Le 2 février 2021

Cégep de Matane



<b>Résumé de l'application</b>	<b>3</b>
<b>Interfaces</b>	<b>4</b>
Ecran 1	4
Ecran 2	5
Ecran 3	6
<b>Cas d'utilisation</b>	<b>7</b>
Diagramme de cas d'utilisation	7
Fiche de cas d'utilisation A	8
Fiche de cas d'utilisation B	9
Fiche de cas d'utilisation C	10
<b>Grille de test fonctionnel</b>	<b>11</b>
<b>Produits similaires sur le marché</b>	<b>12</b>
<b>Analyse comparative des technologies</b>	<b>12</b>
<b>Le défi que cela représente pour l'équipe</b>	<b>13</b>
<b>Ressources</b>	<b>14</b>
Lien vers les sources	14
Lien vers les outils	14
Outil de prototypage d'écran	14
Outil UML	14



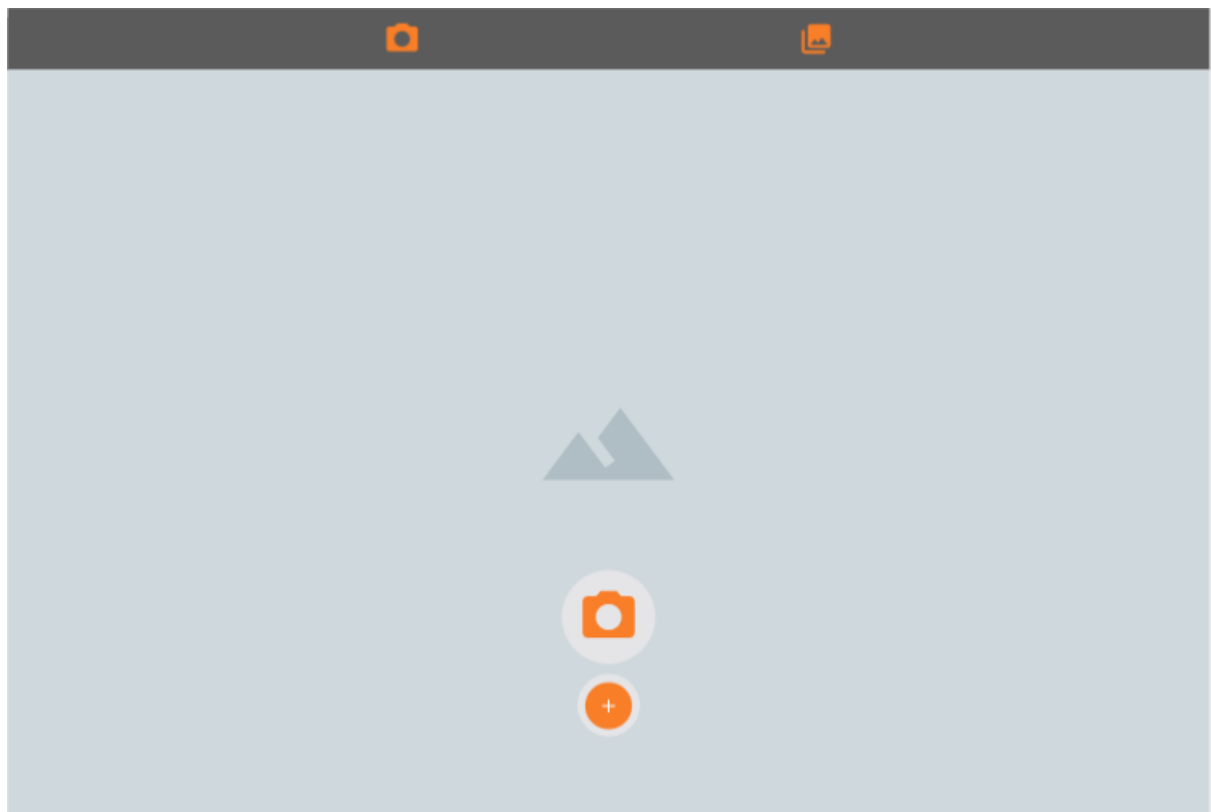
## Résumé de l'application

L'application permet de classifier des fleurs selon leur espèce. L'utilisateur prend une photo de fleur ou en importe une depuis son ordinateur et l'application détermine ensuite l'espèce de la fleur. L'utilisateur a accès à l'historique des classifications qu'il a effectué (photo, type de fleur, date) et peut ajouter une note ou la supprimer.

# Interfaces

## Ecran 1

Accueil



## Ecran 2

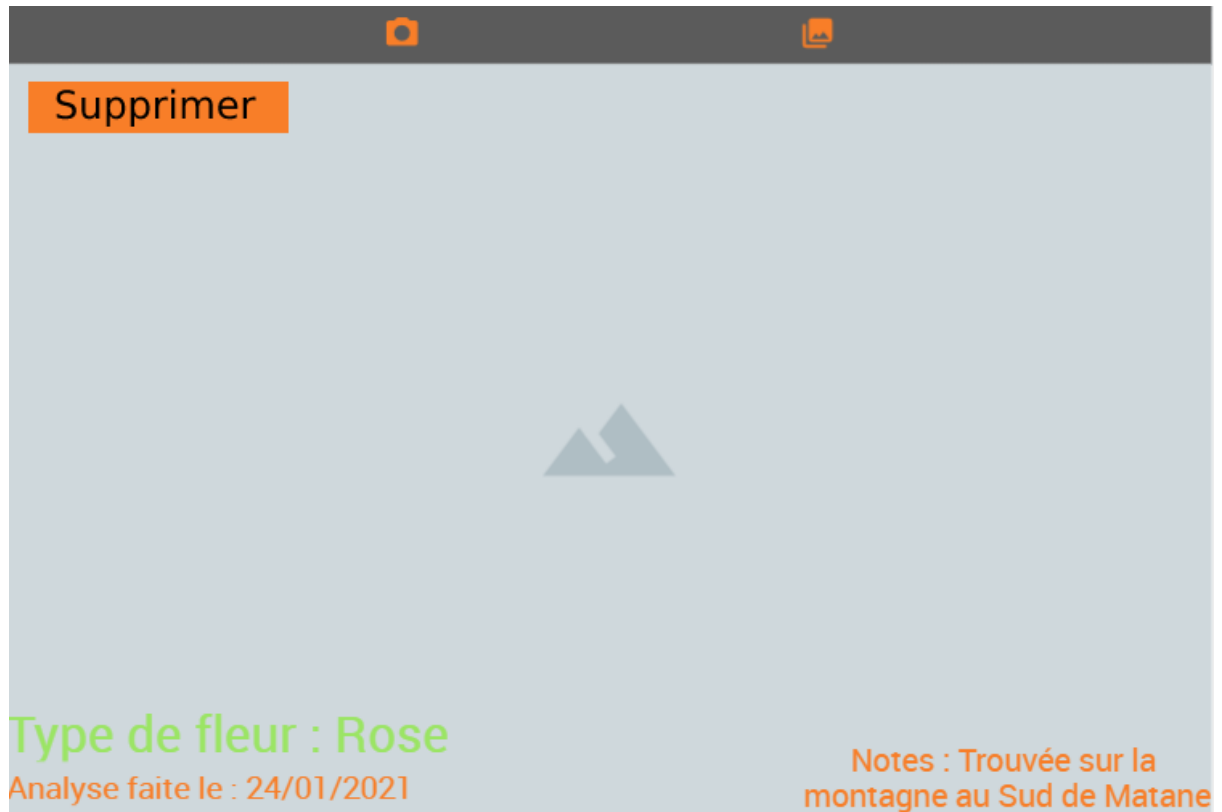
### Historique des classifications



	Rose 24/01/2021		Tulipe 24/01/2021		Tournesol 24/01/2021		Coquelicot 24/01/2021
	Tournesol 12/01/2021		Pissenlit 12/01/2021		Tulipe 06/01/2021		Rose 01/01/2021
	Jonquille 28/12/2020		Tulipe 28/12/2020		Pissenlit 28/12/2020		Tulipe 28/12/2020
	Tulipe 28/12/2020		Tulipe 28/12/2020		Rose 28/12/2020		Rose 28/12/2020
	Rose 28/12/2020		Rose 28/12/2020		Pissenlit 03/12/2020		Pissenlit 03/12/2020
	Pissenlit 03/12/2020		Pissenlit 03/12/2020		Tulipe 03/12/2020		Tulipe 03/12/2020
	Tulipe 03/12/2020		Tulipe 03/12/2020		Tournesol 03/12/2020		Tournesol 03/12/2020

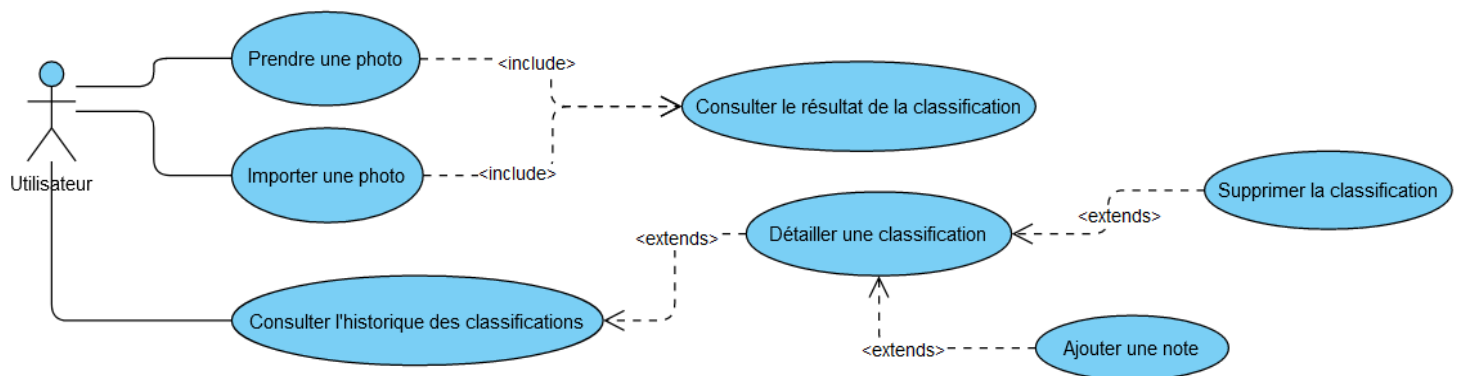
## Ecran 3

Détail d'une classification



# Cas d'utilisation

## Diagramme de cas d'utilisation



## Fiche de cas d'utilisation A

<b>Résumé</b>	L'utilisateur fait classier une fleur.	
<b>Acteur principal</b>	L'utilisateur.	
<b>Intervenants &amp; Rôles</b>	<b>Intervenant</b>	<b>Rôle</b>
<b>Préconditions</b>		
<b>Postconditions</b>	La classification a été ajoutée à l'historique des classifications.	
<b>Déclencheur</b>	L'utilisateur a importé ou pris une photo.	

Scénario nominal	Étape	Action
	1	Depuis l'écran d'accueil l'utilisateur clique sur le bouton "importer une photo" ou "prendre une photo".
	2	Le système vérifie que la photo ne dépasse pas une certaine taille.
	3	Le système affiche le résultat de la classification.

Scénarios alternatifs	Étape	Actions
	2.1	Si la photo téléchargée ou prise dépasse une certaine taille, le système refuse la photo.
	2.2	Le système affiche un message d'erreur indiquant que la photo est trop volumineuse.



## Fiche de cas d'utilisation B

<b>Résumé</b>	Ajouter ou modifier une note à une classification de l'historique.	
<b>Acteur principal</b>	L'utilisateur	
<b>Intervenants &amp; Rôles</b>	<b>Intervenant</b>	<b>Rôle</b>
<b>Préconditions</b>	Au moins une classification existe dans l'historique des classifications.	
<b>Postconditions</b>	La note a été ajoutée ou mise à jour.	
<b>Déclencheur</b>		

<b>Scénario nominal</b>	<b>Étape</b>	<b>Action</b>
	1	Depuis la page historique, l'utilisateur clique sur une classification.
	2	Le système affiche la classification (image, date, résultat, note).
	3	L'utilisateur change le contenu du champ de texte relatif à la note.
	4	Le système vérifie si la note ne dépasse pas un certain nombre de caractères.
	5	Le système ajoute ou met à jour la note dès que l'utilisateur ne focus plus le champ de texte.

<b>Scénarios alternatifs</b>	<b>Étape</b>	<b>Actions</b>
	4.1	Si la note dépasse un certain nombre de caractères, le système refuse d'enregistrer la note.
	4.2	Le système affiche un message d'erreur indiquant que la note est trop volumineuse.

## Fiche de cas d'utilisation C

<b>Résumé</b>	Supprimer une classification.	
<b>Acteur principal</b>	L'utilisateur.	
<b>Intervenants &amp; Rôles</b>	<b>Intervenant</b>	<b>Rôle</b>
<b>Préconditions</b>	Au moins une classification existe dans l'historique des classifications.	
<b>Postconditions</b>	La classification a été supprimée de l'historique des classifications.	
<b>Déclencheur</b>		

Scénario nominal	Étape	Action
	1	Depuis la page historique, l'utilisateur clique sur une classification.
	2	Le système affiche la classification (image, date, résultat, note).
	3	L'utilisateur clique sur le bouton "Supprimer".
	4	Le système redirige vers la page historique des classifications. La classification n'apparaît plus dans la liste.

Scénarios alternatifs	Étape	Actions



## Grille de test fonctionnel

Test	Instructions	Résultat	Ré uss i	Éch oué	Signatur e
<b>Classifica tion d'une fleur</b>	L'utilisateur importe une image de fleur ou en prend une en photo.	L'application classe la fleur et affiche le résultat de la classification.			
<b>Ajouter une note</b>	L'utilisateur détaille une classification depuis l'historique et modifie la note associée à cette classification.	La note a été enregistrée, elle s'affiche aux prochaines ouvertures de la classification.			
<b>Supprime r une classificat ion</b>	L'utilisateur détaille une classification depuis l'historique et supprime la note associée à cette classification.	La note a été supprimée et n'apparaît plus dans l'historique.			



## Produits similaires sur le marché

tf-flower-classificator : <https://github.com/juanabascal/tf-flower-classificator> : classificateur de fleurs utilisant Tensorflow

plantsnap : <https://www.plantsnap.com> : application mobile permettant d'identifier des fleurs en prenant une photo

## Analyse comparative des technologies

Bien que ce projet aurait pu être réalisé en Java ou encore en C++, nous avons choisi d'utiliser Python car c'est le choix de prédilection des développeurs pour le deep learning. Python propose notamment un nombre de bibliothèques important garantissant qu'il existe une solution à chaque problème. Python favorise une augmentation de la productivité en réduisant le temps de codage et de débogage par sa simplicité. Nous avons également choisi Python car c'est un langage très populaire qui serait intéressant d'apprendre. Cette popularité garantit par ailleurs une quantité importante d'entrées le concernant sur des forums, facilitant l'apprentissage et la résolution d'erreurs.

Concernant la librairie, nous avons choisi Tensorflow, développé par Google, car c'est un framework très bien documenté qui a une très grande communauté. Tensorflow intègre de plus Keras qui permet de créer facilement des modèles. Nous aurions pu nous orienter vers PyTorch, un autre framework, développé par Facebook, et offrant des fonctionnalités similaires à Tensorflow. Nous avons néanmoins choisi Tensorflow pour sa popularité et car il intègre Keras.



## MACHINE LEARNING LANGUAGES SHOOT OUT — HOW TO PICK THE RIGHT LANGUAGE

% of machine learning developers / data scientists who use or prioritise each language (n = 2,022)



Source: State of the Developer Nation Q1 2017 | <http://vmob.me/DE1Q17> | Copyright VisionMobile | All rights reserved

## Le défi que cela représente pour l'équipe

Yann : Ce projet sera pour moi ma première approche du langage Python mais aussi ma première approche du Deep Learning et de l'intelligence artificielle approfondie en général.

Thomas : Bien que j'ai des bases théoriques en intelligence artificielle, ce projet est l'occasion de m'initier à Python et d'apprendre à implémenter un réseau de neurones de classification via Tensorflow.



## Ressources

### Lien vers les sources

State of the developer nation Q1 2017 :

<http://s3-eu-west-1.amazonaws.com/de-blog/uploads/2017/03/VisionMobile-Developer-Economics-State-%E2%80%93-of-the-Developer-Nation-Q1-2017.pdf>

### Lien vers les outils

Outil de prototypage d'écran

[www.fluidui.com](http://www.fluidui.com)

Outil UML

<https://online.visual-paradigm.com/fr/diagrams/solutions/free-use-case-diagram-tool/>