

|  |  |
| --- | --- |
| **CFI Montigny le Bretonneux** | |
| **Reconnaissance de pièces** | | |
| **Méthode Agiles** |

|  |
| --- |
| *Jean-Hilaire OBAME, Clément AUBRY, Marie LANOUE, Karam YELLES*  **21/12/2018** |



**Sommaire :**

[1. Cadrage du projet 3](#_Toc533089734)

[1.1 Méthode Agile : 3](#_Toc533089735)

[1.2 Scrum : 4](#_Toc533089736)

[1.3 Contexte 6](#_Toc533089737)

[1.4Objectif 6](#_Toc533089738)

[1.5 Organisation du projet 6](#_Toc533089739)

[2. Etude 8](#_Toc533089740)

[2.1 Trello 8](#_Toc533089741)

[2.1 GitHub : 10](#_Toc533089742)

[3. Conception 12](#_Toc533089743)

[3.1 ArgoUML 12](#_Toc533089744)

[3.2 SolidWorks 14](#_Toc533089745)

[4. Réalisation 15](#_Toc533089746)

[4.1 Interface graphique 16](#_Toc533089747)

[4.2 Acquisition d’image 18](#_Toc533089748)

[5. Perspectives d’améliorations 19](#_Toc533089749)

[6. Conclusion 20](#_Toc533089750)

[7. Annexes 21](#_Toc533089751)

[7.1 Annexes 1 21](#_Toc533089752)

[7.1 Annexe 1 23](#_Toc533089753)

# **Cadrage du projet**

Durant notre première année de master Manager des systèmes informatiques et robotiques, nous avons participé à différents projets dans un cadre pédagogique afin de nous préparer à nos futures responsabilités professionnelles.

Le projet qui fera l’objet de ce rapport est une problématique dans lequel nous devions appliquer une méthode de gestion de projet appelé méthode agile :

## 1.1 Méthode Agile :

Une méthode Agile est une approche interactive et collaborative, capable de prendre en compte les besoins initiaux du client et ceux liés aux évolutions du projet.

La méthode Agile se base sur un cycle de développement qui porte le client au centre. Le client est impliqué dans la réalisation du début à la fin du projet. Grâce à la méthode agile le client obtient une meilleure visibilité de la gestion des travaux.

L’implication du client dans le processus permet à l’équipe d’obtenir des feedback réguliers afin d’appliquer directement les changements nécessaires. Cette méthode vise à accélérer le développement d’un logiciel. De plus, elle assure la réalisation d’un logiciel fonctionnel tout au long de la durée de sa création.

Le principe de base consiste à proposer une version minimale du logiciel puis à intégrer des fonctionnalités supplémentaires à cette base, par un processus de répétition. Ce processus regroupe une séquence d’instructions à répéter autant de fois que possible, selon le besoin.

La méthode agile repose sur quatre grands principes :

* **COLLABORATION** : Communication et cohésion d’équipe passent avant les outils et les processus.
* **EQUIPE** : Le privilège de la relation équipe/client est mis en avant plutôt que la négociation contractuelle.
* **APPLICATION** : Préférer une application bien construite à une documentation détaillée.
* **ACCEPTATION** : Le choix de l’acceptation du changement et de la flexibilité au détriment d’un plan rigide.

## **1.2 Scrum :**

En effet, lorsque l’on emploie le terme « méthode agile » au singulier on parle d’un concept, une approche. Cependant il existe plusieurs méthodes agiles qui se différencient les unes des autres.

De nos jours, la méthode « Scrum » est la méthode agile la plus populaire. Cette méthode s’appuie sur des « sprints » qui sont des espaces temps pouvant aller de quelques heures pour les plus courts à un mois pour le plus longues. Généralement et de préférence un sprint s’étend sur deux semaines. A la fin de chaque sprint, l’équipe présente ce qu’elle a ajouté au produit au client.

La méthode Scrum regroupe trois acteurs :

* **Le Product Owner** (ou « Directeur de produit ») : il communique les objectifs premiers des clients et utilisateurs finaux, coordonne l’implication des utilisateurs et des parties prenantes, et se coordonne lui-même avec les autres product owners pour assurer une cohérence.
* **Le Scrum Master** : membre de l’équipe, il a pour but d’optimiser la capacité de production de l’équipe. Pour se faire, le scrum master aide l’équipe à travailler de façon autonome tout en s’améliorant d’avantage.
* **L’équipe** opérationnelle : Elle est composée idéalement de moins de dix personnes: la particularité d’une équipe scrum est qu’elle est dépourvue de toute hiérarchie interne. Une équipe scrum est auto-organisée.

L’avancement d’un projet scrum est pulsé par de grands principes permettant de garantir sa réussite :

**Les sprints**

Le cycle de vie Scrum est rythmé par des échéances de quelques semaines : les sprints.

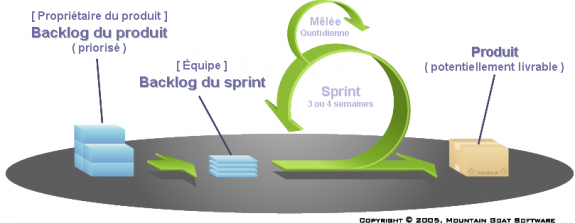
On organise, avant chaque sprint, une réunion de planification. Ce planning sélectionne dans le product backlog les exigences prioritaires pour le client. Elles seront développées, testées et livrées au client à la fin du sprint.Elles constituent le **sprint backlog, un sous ensemble du product backlog.**

**Le product backlog**

**C’est un référentiel des exigences initiales**, dressé et hiérarchisé avec le client. Il constitue ce que l’on nomme le **product backlog.** Il ne doit pas nécessairement contenir toutes les fonctionnalités attendues dès le début du projet, il peut évoluer durant le projet en fonction des besoins du client.

**Mêlée quotidienne**

Il s'agit d'une réunion de synchronisation de l'équipe de développement qui se fait debout de chaque journée. Elles ont une durée de 15 minutes maximum au cours de laquelle chacun répond principalement à 3 questions : « Qu'est-ce que j'ai terminé depuis la dernière mêlée ? Qu'est-ce que j'aurai terminé d'ici la prochaine mêlée ? Quels obstacles me retardent ? »



**Schéma d’un processus Scrum**

## 1.3 Contexte

C’est en partant de cette feuille de route que nous avons appliqué la méthode de gestion de projet scrum afin de réaliser le projet qui nous a été attribué.

Prévoyant de récolter un nombre conséquent de pièces de monnaies à l’approche des fêtes de fins d’années et dans le but de simplifier le comptage de monnaies de leurs bénévoles, une association a décidé de faire appel aux étudiants de Master 1 informatique et robotique du CFI Montigny le Bretonneux afin de trouver une solution à leur problème. Le but étant de mettre de concevoir et de mettre en place un logiciel facilitant la reconnaissance et le comptage de pièces de monnaies. Le produit devra être opérationnel et prêt à l’utilisation à partir du weekend précédant les fêtes de noël (à partir Du 22 décembre 2018).

## 1.4Objectif

Ce projet comporte plusieurs objectifs :

* Livrer un produit prêt à l’utilisation et ce dans les temps souhaitez par le client (produit prêt a être utiliser pour le 22 Décembre).
* Concevoir un produit simple d’utilisation et ludique afin que la prise en main du produit soit la plus intuitive possible pour le client.

## 1.5 Organisation du projet

Afin de répondre au mieux à la demande de notre client ainsi qu’au fondamentaux de la méthode scrum, nous avons mis en place une équipe composé de 4 membres :

* LANOUE Marie : Qui a le rôle de développeur ainsi que celui du **Scrum Master**
* AUBRY Clément : qui a le rôle de développeur ainsi que celui de **Product Owner**
* OBAME Jean-Hilaire : Qui a le rôle de Développeur
* YELLES Karam : Qui a le rôle de Développeur

Afin de répondre au mieux aux besoins de notre client, nous avons décidé d’effectuer une analyse « QQOQCCP » sur notre problématique.

C’est une méthode de résolution de problème permettant d’avoir toutes les dimensions d’une problématique afin d’obtenir les informations élémentaires suffisantes pour identifier ses aspects essentiels de cette dernière. Cette méthode se réalise en répondant de manière précise à six questions :

* Quoi ? *« De quoi s’agit-il ? »*
* Qui ? *« Qui est concerné ? »*
* Où ? *« Où cela se produit-il et s'applique-t-il ? »*
* Quand ? *« Depuis quand vous avez ce problème ? »*
* Combien ? *« Comment se produit le problème ? »*
* Comment ? *« Comment se produit le problème ? »*
* Pourquoi ? « Pourquoi le problème se pose ? »

QQOQCCP du projet de reconnaissance de monnaie

Suite à cette analyse, nous avons pu établir une feuille de route qui nous servira de fil conducteur à la bonne réalisation de notre mission. Cette feuille de route nous servira de base à la construction et à la gestion de notre projet tout en appliquant les méthodologies Scrum apprises en formation.

# **2. Etude**

Afin de réaliser au mieux notre projet, nous avons utilisés différents outils tel que :

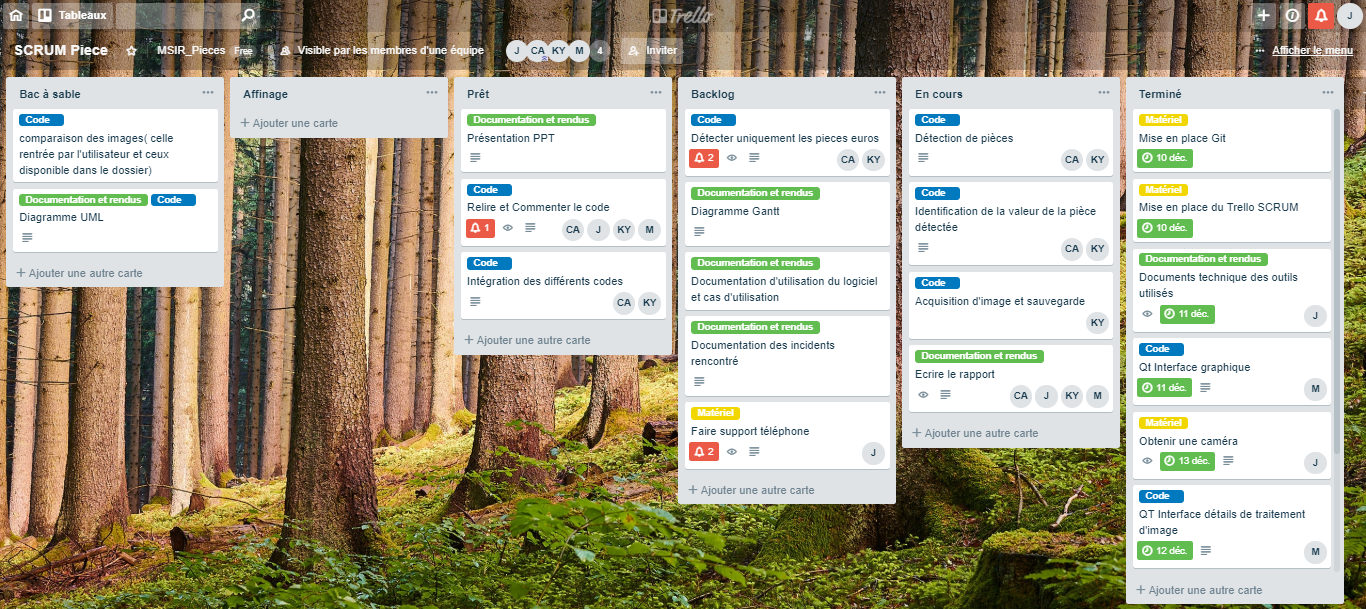
## 2.1 Trello

C’est un outil de gestion de projet en ligne basé sur une organisation des projets en colonnes listant des cartes représentant des tâches. Chaque tache peut être assignée à différents membre d’une équipe. Les taches peuvent être déplacées d’une colonne à l’autre, traduisant leur avancement. La conception de ce site est basé sur le fonctionnent des méthodes agiles et tout particulièrement à la méthode Scrum.

C’est un site totalement gratuit permettant à chaque membre de l’équipe d’y accéder et de pouvoir le modifie afin de le mettre à jour à tout moment.

Nous avons donc utilisé Trello afin de mettre en place un planning Scrum à l’intérieur duquel nous avons développé et réparti les tâches utiles à notre projet. Les différentes tâches ont été hiérarchisées, répartie en colonnes puis attribué à différents membres de l’équipe projet. Nous avons trouvé judicieux de séparer notre projet en 2 « sprints » afin d’avoir un meilleur contrôle sur l’avancement de notre projet. Ceci nous permet également de pouvoir présenter les avancé du projet au client, ce qui nous permettra de d’obtenir des feedbacks constructifs de mi projet afin que nous puissions apporter les corrections nécessaires à notre logiciel.

Pour finir, nous avons estimé un délai de réalisation pour chacune des tâches du projet en cohérence avec le délai de 2 semaines en cohérence avec notre client.



**Tableau de planification des tâches du projet reconnaissances des pièces**

Les tâches ont été divisées en 2 sprints d’une semaine :

* **Premier sprint :**
  + Mise en place des outils du projet (Trello, GitHub, QT creator ainsi que la bibliothèque OpenCV)
  + Rédaction du rapport (Document technique, Digramme de Gantt, Diagramme UML).
  + Conception de l’interface graphique du logiciel sur Qt
  + Programmation (Affichage du contour des pièces)
  + Obtention d’une caméra permettant la détection des pièces
  + Programmation (acquisition et sauvegarde d’image)
  + Programmation (détection et identification de la valeur des pièces détectées)
* **Deuxième sprint**
* Réalisation du Power Point
* Finalisation du rapport (document d’utilisation du logiciel, document d’incident rencontré, diagramme d’utilisation)
* Conception du support caméra
* Programmation (Détection unique des pièces d’euros puis faire leurs sommes)
* Mise en relation de différentes parties du code ainsi que l’interface graphique
* Relecture et commentaire du code

Le travail réalisé sur ce logiciel ainsi que le délais de livraison du produit imposé par notre client nous a permis de mettre en place un diagramme de GANTT (page 11)

## 2.1 GitHub :

Pour ce projet nous avons eu la nécessité de mettre en place un système d’archivage de données afin que chaque membre du groupe puisse se transmettre différents documents utiles dans notre progression (fichier, dossier utile au projet). Nous avons donc mis en place un logiciel de gestion des versions nommé GitHub.

C’est un logiciel permettant de stocker différents documents sur une plateforme. Il utilise le logiciel de gestion de version GIT. Permettant d’archiver les différentes versions d’un programme.

Ce logiciel nous a permis de pouvoir être informé de l’avancement en temps réel des différentes tâches gérées par chaque membre de l’équipe ainsi que d’avoir la possibilité d’apporter éventuellement des ajustements ou commentaires pertinents.

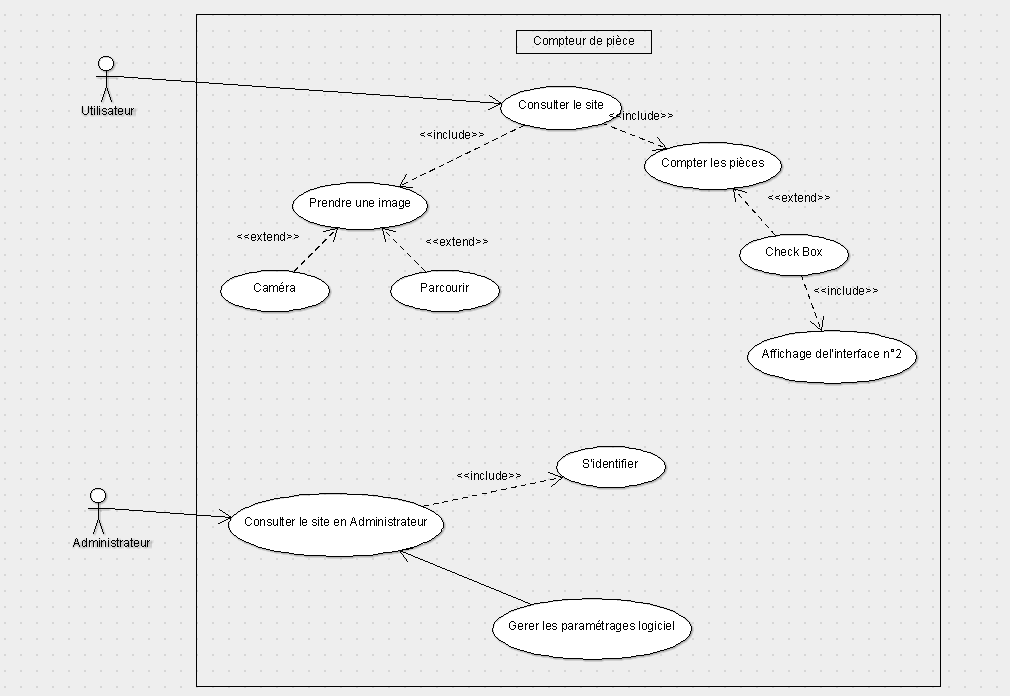
**Diagramme de GANTT du projet de reconnaissance des pièces**

# **Conception**

Avant de nous lancer dans la programmation pure de notre futur logiciel, nous avons décidé de mettre en forme de façon schématique son diagramme de classe afin de pouvoir visualiser les interactions possible entre le différentes interfaces de notre de notre logiciel. Puis un diagramme de cas d’utilisation dans le but d’obtenir une vision global du comportement fonctionnel du logiciel.

## 3.1 ArgoUML

C’est un logiciel libre spécialisé dans la création de différents types de diagramme UML (diagramme de cas d’utilisation, de classes, de séquence, d’état, de collaboration, d’activité et de déploiement).

****

**Diagramme des cas d’utilisation du logiciel**

Le diagramme des cas d’utilisation ci-dessus permet de visualiser de façon générale les différents cas d’utilisation des différentes personnes qui auront accès au logiciel.

Nous avons donc identifié deux types d’utilisateurs pour le futur logiciel :

* **Utilisateur lambda** : Aura accès aux fonctions d’utilisation classique du logiciel :
  + Application générale de comptage de pièces
  + Prise d’une image
  + Comptage des pièces par image
  + Affichage de l’interface détaillé du traitement d’image
* **Administration :** Aura accès à toutes les fonctions d’utilisations classique du logiciel et pourra accéder à la modification des paramètres du logiciel :
  + Application générale de comptage de pièces
  + Prise d’une image
  + Comptage des pièces par image
  + Affichage de l’interface détaillé du traitement d’image
  + Gérer les paramètres du logiciel

**Diagramme de classes du logiciel**

## SolidWorks

Il existe différents moyen d’acquérir une image de pièce en temps réel afin de pouvoir la traiter par la suite (appareil photo, caméra, téléphone portable, webcam,…). La solution qui nous paraissait la plus pratique et la plus économique pour notre client était l’utilisation d’une webcam sphérique pour effectuer l’acquisition d’une photo.

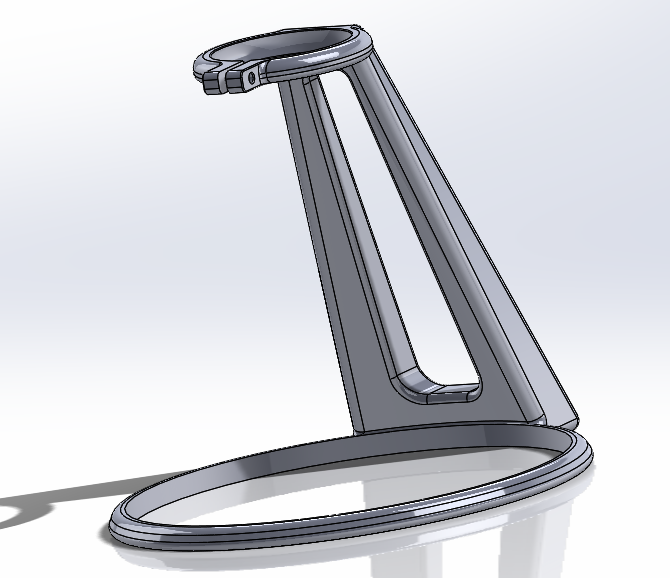
Avantage : Simple d’utilisation, facile d’acquisition, simple de paramétrage sur Qt créator.

Afin d’utiliser de façon optimale ce type de webcam, il fallait concevoir un support permettant à la fois de soutenir notre outil de d’acquisition d’images et de pouvoir marquer la délimitation d’une zone de capture d’image des pièces à traiter. Pour cela nous décider de concevoir, sur un logiciel CAO, un support alliant les deux fonctions.

Solidworks et un logiciel CAO (Conception Assisté par Ordinateur) de type propriétaire fonctionnant sur Windows. Ce logiciel fonctionne sur la mise en relation de 3 types de fichiers relatifs à trois concepts de base : la conception d’une pièce, d’un assemblage et leurs mises en plan. Ces fichiers sont en relation. Toute modification à quelque niveau que ce soit est répercutée vers tous les fichiers concernés. Pour notre besoin, nous avons utilisé deux des trois concepts de base du logiciel :

* Conception d’une pièce : Support Webcam (dessin ci-dessous)
* Mise en plan de la pièce : Support Webcam (Présenté en annexe)

La Webcam sera positionnée et fixé grâce à un système de serrage étau. Ce système d’étau se situera à une vingtaine de centimètre de la zone de positionnements des pièces à traiter. La zone d’acquisition des pièces sera délimitée par le cercle situé à la base de notre support. Afin que la webcam capture de façon optimale les images des pièces à additionner, le point d’origine des deux cercles (support Webcam et le socle du support) a été aligné sur le même axe verticale.



**Dessin de conception du support caméra**

# **Réalisation**

* **4.1 Le logiciel Qt creator** (en utilisant la bibliothèque Open CV)

Afin de réaliser notre application nous avons utilisé le logiciel Qt creator. En effet le logiciel Qt creator offre une interface graphique permettant de configurer un visuel sur notre application de reconnaissance de pièce. C’est un logiciel orienté objet et développé en C++. Notre préférence s’est portée sur ce logiciel car il répondait au mieux à notre besoin. C’est un logiciel avec lequel nous avons pu nous familiariser au cours de notre formation et dont nous avons le plus d’aisance dans sa prise en main et sa manipulation.

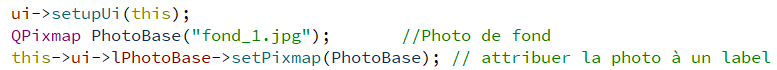
Afin de nous permettre d’exploiter au mieux les capacités du logiciel Qt, nous avons décidé d’installer la bibliothèque Open CV (Open Computeur Vision). C’est une [bibliothèque graphique](https://fr.wikipedia.org/wiki/Biblioth%C3%A8que_graphique) [libre](https://fr.wikipedia.org/wiki/Logiciel_libre), spécialisée dans le [traitement d'images](https://fr.wikipedia.org/wiki/Traitement_d%27images) en temps réel.

## 4.1 Interface graphique

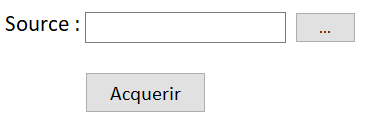
Création de l’interface 1 :

L’interface devra pouvoir récupérer l’image à traiter et attribuer cette image à un label voulu (pour l’affichage des pièces)

Fonction utiliser pour prendre une photo et l’attribuer à un label :

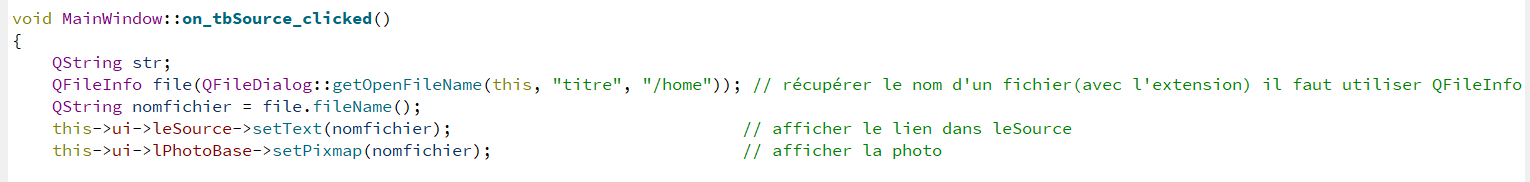






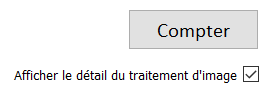
**Ecran principale du logiciel**

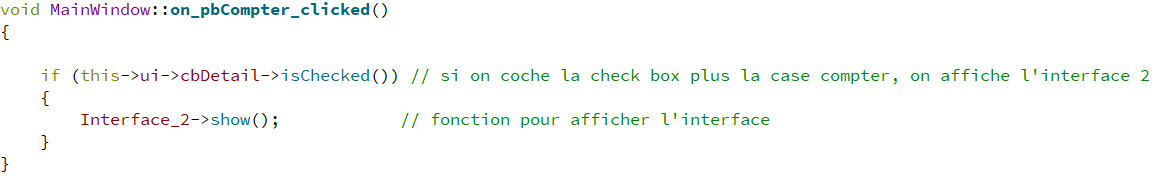
Il a fallu donc créer touche parcourir pour aller chercher une image dans un dossier et afficher le code source dans le label correspondant. Nous avons aussi trouvé important que les futurs utilisateurs puissent visualiser le nombre de pièces de chaque type.

Fonction utiliser pour la touche parcourir de l’interface ce qui permet d’aller chercher une photo dans un dossier et afficher le lien source de la photo dans le label correspondant :

Création de l’interface 2 :

Suite à cela nous avons mis en place un deuxième écran permettant de visualiser plus en détail les filtres utilisé dans le traitement de l’image souhaité.

Ci-dessous le code qui permet d’accéder à l’interface 2, ainsi que les touches correspondantes :



L’interface deux est basé sur les mêmes dimensions que l’interface 1

Une fois le check boxe cocher et que l’on appuis sur compter pour afficher l’interface suivant :



**Ecran principale du logiciel**

Cette fenêtre sert à détailler le traitement appliqué sur l’image choisis par l’utilisateur et sert aux développeurs si le traitement de l’image a été réalisé de correctement.

**Problème rencontré :**

* Faire communiquer les deux interfaces🡪 faire en sorte que l’interface 2 s’affiche seulement lorsqu’on coche la check box et que l’on appuis sur compter grâce à une boucle if
* Dimensionner l’image à la taille du label

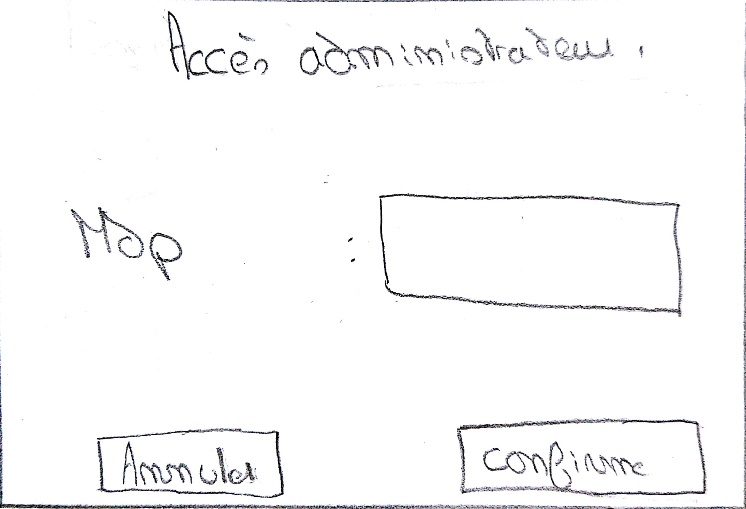
## 4.2 Acquisition d’image

# **Perspectives d’améliorations**

D’un point de vue global notre application correspond à ce que nous souhaitions réaliser. Mais à cause du temps imparti nous n’avons pu optimiser de notre application comme nous le souhaitions.

Nous avons par ailleurs identifié plusieurs axes d’améliorations permettant d’avoir le logiciel le plus optimal possible :

* Fabrication du support en impression 3D
* Ajout d’une page administrateur afin de donner l’accès aux modifications de certains paramètres à quelques personnes habilitées



**Paramètres à modifier :**

* Distance entre les centres des pièces
* Diamètres des pièces minimale / maximale
* Echelles des photos.
* Optimisation du programme d’identification des pièces de monnaies
  + Optimisation de la détection des cercles pour permettre une mesure des diamètres des pièces plus précis.
  + Optimisation de la détection des couleurs, pour permettre de mieux détecter les types de pièces.
  + Identifier les euros parmi d’autres pièces.

# **Conclusion**

Après 2 semaines de projet de reconnaissance de pièces en appliquant la méthode agile, nous avons fait en sorte de répondre au mieux au besoin de notre client. Bien que le temps imparti fût assez cours, nous avons pu nous répartir les taches de façon judicieuse selon les compétences de chacun afin de pouvoir tous avancer vers le même objectif.

Cette expérience fut très enrichissante pour nous à tous les points :

* **Gestion de projet :**
  + Pouvoir travailler en équipe de façon agile nous à permit de gagner en autonomie dans notre prise de décision.
  + Suivre de façon rigoureuse un planning et se tenir aux délais préconisés
  + Apprendre à répartir des taches selon les compétences de chacun des membres d’une équipe
  + Réaliser un projet en appliquant une méthodologie de gestion de projet (SCRUM).
  + Appliquant des outils de gestion de projet tel que TRELLO, GitHub, diagramme de Gantt
* **Technique :**
  + Amélioration de nos compétences en programmation (C++, Qt).
  + Familiarisation avec la bibliothèque OpenCV
  + Utilisation de logiciel tel que le logiciel argoUML ainsi que SolidWorks
  + Connaissance personnelle en traitement d’image ainsi que dans l’utilisation de différents types de filtres.

L’une de nos plus grandes satisfactions durant toute la durée de ce projet est l’extrême concentration et implication dont nous avons su faire preuves. Malgré les perturbations externes plus ou moins importantes. Ceci nous servira grandement lors de nos futures projets ; autant sur le plan personnel que professionnel.

# **Annexes**

## 7.1 Annexes 1

Documentation d'utilisation du logiciel

Pour utiliser le logiciel et pour qu’il soit fonctionnelle il faut prendra prendre une photo, pour ce faire il existe deux méthode :

* Première méthode : il faut utiliser une image qui se trouve sur votre Pc que vous allez chercher via la touche parcourir présent sur l’interface.
* Deuxième méthode : il faut utiliser une caméra préalablement branché, ensuite utiliser les touches de l’interface pour afficher ce que voie la caméra puis faire une capture.

Une fois la photo prise et afficher sur l’interface, il faudra cliquer sur la touche « Compter » pour que le programme tourne pour compter les différentes pièces sur la photo et affiche sur l’interface le résultat dans la case « Somme ».



1

6

7

2

3

8

4

5

9

**Vue d’ensemble de l’interface :**

**Légende :**

1) Affichage des différentes photos sélectionner par l’utilisateur

2) Touche parcourir pour aller chercher une photo dans un dossier le dossier BUILD

3) La source de la photo s’afficheras ici

4) Index de la caméra : 0-> webcam du PC 1-> caméra ajouter en plus

5) Touche pour acquérir une image de la caméra

6) Interface ou il sera afficher le nombre de pièce par type de pièce

7) La somme de toute les pièces présente sur l’image choisis par l’utilisateur

8) Touche Compter pour lancer le programme et ainsi compter les pièces

9) Check Box à cocher si voulus par l’utilisateur pour afficher le détail du traitement faite sur l’image

## 7.1 Annexe 1

**Plan de conception du support Webcam**