## Remise #4 – tests unitaires, diagramme de cas d’utilisation et Interface graphique

## RAPPEL – Charge de travail

N’oubliez pas de prendre en note le temps que vous mettez sur votre projet. Vous aurez besoin de ces informations pour compléter le tableau d’estimation de la charge de travail et pour votre présentation.

## RAPPEL – Git, GitHub et vos dépôts (commits)

Rappelez-vous également de faire des dépôts dans Git et des push dans GitHub après chacune des étapes de développement de votre application.

## Tests unitaires

Ajouter les tests unitaires qui permettent de tester les modificateurs / accesseurs (setters/getters) et toutes les méthodes de votre application. Pour cela, utiliser le module pytest.

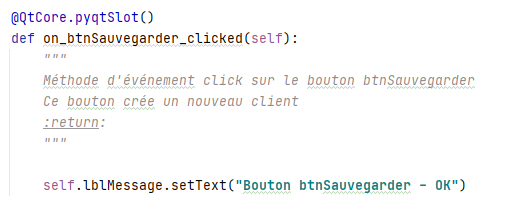
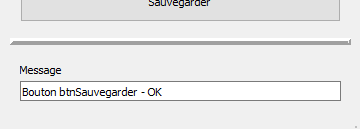
## Diagramme de cas d’utilisation

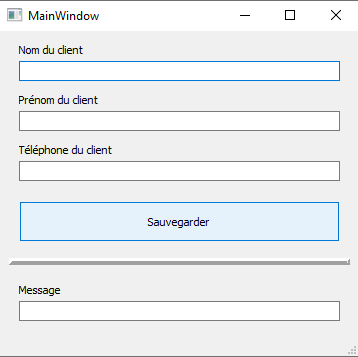
Produire un diagramme de cas d’utilisation qui décrit le fonctionnement de votre application.

## L’interface graphique Qt Designer

Après avoir codé vos classes et modélisé votre diagramme de cas d’utilisation, vous avez déjà une bonne idée du contenu nécessaire de votre interface graphique. Créer une interface graphique avec Qt Designer en lien avec le sujet de l’application que votre UML présente.

À cette étape, il n’est pas nécessaire que les fonctionnalités de votre application (créer des objets, faire des calculs, etc.) soient développées. Ce qui est exigé ici c’est de pouvoir confirmer que les mécaniques de votre interface fonctionnent correctement, c’est-à-dire que les gestionnaires d’évènements s’exécutent.

Exemple : Supposons qu’il y ait un bouton pour créer un nouveau client. En cliquant sur ce bouton, un message pourrait être affiché dans un label à message. Si le message s’affiche correctement, c’est que le click sur le bouton appelle la bonne méthode d’événement.



## Documents à remettre

Vous devez remettre vos tests unitaires, votre diagramme de cas d’utilisation et votre prototype de programme (programme principal, codes des boîtes de dialogue, fichiers .ui de QtDesigner et fichiers convertis en .py). Remettre votre travail sur LÉA sous forme de dossier compressé. Vos fichiers doivent aussi être accessibles sur votre dépôt GitHub.