Professor: Dominique Genoud

Project requirements

Work description:

Re-créer le projet de 0

Avoir X et Y ? C'est à nous de choisir.

Create a **new** application based on your previous projects.
We want a program that does 0 to 3 filters and 0 to 3 edge detections.
The code must have the maximum possible code coverage.
The use of gnenrative pre-trained transformers (GPT) is authorized but must be documented precisely.

Features expected:
- Load an image from the disk
- Perform 0 to 3 filters and 0 to 3 edge detections
- Save the modified image into a file

Filters : on peut faire plusieurs fois une couche différente sans souci

EdgeDetetion :
On ne fait que une fois. Si on refait, on re-set l'image et on refait le edgeDétection

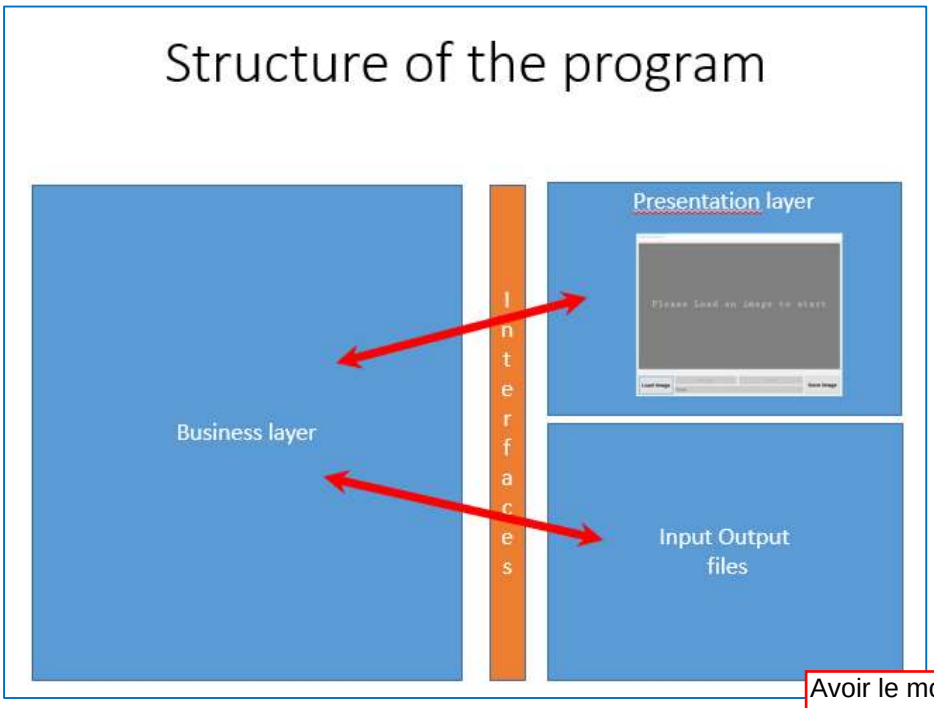
Mandatory structure of the code (At least one of each)
- Use *interfaces* to load and save files
  (You should be able to implement save to "file system" or "to database" but only the implementation for the file system is required)
- Use interfaces to separate the code from the presentation layer
- The following elements **must** be seen in the code
  - Exceptions
  - Methods that return void
  - Methods that return classes

Mandatory elements in the unit tests (At least one of each):

- Use Nsubstitute to test the code and substitute interfaces
- Use Nsubstitute to test exceptions
- Use Nsubstitute to test void methods
- Use Nsubstitute to test methods that return classes
- Cover 100% of the business layer code
- Short unit tests doing one thing at a time with explicit names

- Recréer e projet de 0
- Si on utilise des IA, le mentionner
- On peut repartir de notre code
- On doit couvrir un MAXIMUM notre code partout
- UTILISER INTERFACES
  - VOIR TEST DOUBLES

- Dans la BLL, avoir le plus gros



IO

Implementation du code dans la BLL

Interface dans le Bloc IO

PResentation apelle interface je crois

Avoir le moins de code possible dans presentation & I/O.

La classe Load&save image, on en crée une interface, puis on fait un test unitaire avec le substitute

Figure 1: structure of the application

Work to do:

Séparation des tâches

Developer A:

- Create an input/output file manipulation layer using interfaces
- Create the unit tests that goes with it (mandatory)
- Review the code of the developer B, correct and add necessary unit tests
- **It is mandatory to use Nsubstitute calls for the unit tests, all the calls to the I/O should be covered**

Developer B:

- Create a very simple user friendly GUI that follows the specifications
- Create the presentation layer based on interfaces
- Create the necessary unit tests
- Review the code of the developer A, correct and add necessary unit tests.
- **It is mandatory to use Nsubstitute calls for the unit tests, all the calls to file system should be covered**

Both developers together:

- Create the sketch of your application
- Define the required common classes and interfaces
- Create the Business layer and the unit tests that goes with it
- Prepare the necessary interfaces and prototypes
- Prepare the presentation

Tasks	Arthur	Elias	Benjamin		
Refactor	X	X	0		
Unit Testing	0	X	X		
NSubstitute TEST DOUBLES			X	X	0
Documentation	X	X	0		

Deliverables:

- **1 running application in production state.**
  - Commented code
  - The application must behave properly without bugs (prove it)
  - The code coverage should be 100% for the business layer.
  - The unit tests should include the necessary **test doubles (use Nsubstitute)** to test the file access and the presentation layer
- ***No useless code or libraries should remain***

- **Project presentation**

présentation de 15 minutes

  - **All the developers should present their project**
  - ***Demo of the application first***
  - Presentation of all the unit tests mandatory
  - During this presentation the professor should be able to clearly determine the contribution of each student.
  - A short user guide for your application

Organization:

Implementation / Development: Programming language: C# on Visual Studio 2022, use of Nsubstitute, give back a full solution.

As feed back of your work we would like a zip file containing:

- **The complete source code**
- **The presentation**
- **A short user guide for your application**