



Université Hassan II  
Ecole Nationale Supérieure  
d'Electricité et de Mécanique  
Casablanca



# Rapport du Mini projet

Application d'un système expert  
d'aide au diagnostic de pannes d'un  
PC

Département : Mathématique-Informatique

Filière : Génie logiciel et digitalisation

Réalisé par :

Loubna NAKHLAOUI

Encadré par :

Mme.BENHADOU



Université Hassan II  
Ecole Nationale Supérieure  
d'Electricité et de Mécanique  
Casablanca



## Remerciement :

Je tiens à exprimer ma sincère gratitude à ma professeure, Madame Benhadou, pour son encadrement et son soutien tout au long de ce module d'introduction à l'intelligence artificielle. Ses conseils précieux et son expertise ont grandement contribué à la réussite de ce travail. Merci infiniment pour votre dévouement et votre inspiration.



## Table de matière :

<b>Remerciement :</b>	2
Introduction générale :	3
CHAPITRE I : Conception de système expert	6
Introduction :	7
1. Spécification des besoins :	7
2. Les différents diagrammes UML :	7
2.1. Diagramme de Cas d'utilisation :	7
2.1.1. Identification des acteurs :	7
2.1.3. Schéma de diagramme de cas d'utilisation :	8
2.2. Diagramme de séquence :	9
2.2.1. Schéma de diagramme de séquence de cas d'utilisation (S'Authentifier).....	9
2.2.2. Diagramme de séquence de cas d'utilisation (Etablir diagnostic) :	11
2.2.3. Diagramme de séquence de cas d'utilisation (Ajouter règle).....	11
3. Diagramme de classes :	12
Conclusion :	13
CHAPITRE II : Réalisation de système expert	14
Introduction :	15
1. Choix de langage de développement :	15
2. Description de l'application :	16
2.1. Session Expert :	17
2.2. Session utilisateur :	20
3. Choix de la représentation des connaissances et le mode de raisonnement :	22
Conclusion :	23
Conclusion générale :	24



## Introduction générale :

Dans le domaine de la maintenance informatique, le diagnostic rapide et précis des pannes est essentiel pour minimiser les temps d'arrêt et assurer la continuité des opérations. Dans ce contexte, notre projet vise à concevoir et réaliser un système expert d'aide au diagnostic de pannes d'un ordinateur. Ce système expert se propose d'assister les utilisateurs, ainsi que les experts en maintenance informatique, dans l'identification des problèmes matériels potentiels rencontrés par un ordinateur.

L'objectif principal de ce projet est de développer un système intelligent capable de diagnostiquer les pannes d'un ordinateur en se basant sur les symptômes observés par l'utilisateur. Ce système se veut intuitif et efficace, offrant une interface conviviale pour la saisie des symptômes ainsi que pour la manipulation des données de diagnostic.

Le système expert à réaliser repose sur une architecture interactive, divisée en deux sessions distinctes : la session utilisateur et la session expert. La session utilisateur est dédiée à l'interaction avec les utilisateurs finaux, leur permettant de saisir les symptômes observés et de recevoir un diagnostic précis en retour. D'autre part, la session de l'expert est conçue pour les professionnels de la maintenance informatique, offrant des fonctionnalités avancées telles que la gestion de la base de règles du système et la personnalisation des diagnostics.

Ce projet implique plusieurs étapes clés, dont la sélection de la présentation des connaissances, le choix du type de raisonnement (chainage avant ou arrière) et la conception de l'interface utilisateur. De plus, le développement du système sera itératif, avec des tests et des ajustements continus réalisés en collaboration entre l'ingénieur de connaissances et l'expert du domaine.



Université Hassan II  
Ecole Nationale Supérieure  
d'Electricité et de Mécanique  
Casablanca



À travers ce rapport, nous détaillerons les différentes phases du projet, depuis la conceptualisation jusqu'à la réalisation du prototype. Nous présenterons également un scénario d'utilisation du système expert, illustrant son fonctionnement et sa valeur ajoutée dans le processus de diagnostic des pannes informatiques.



Université Hassan II  
Ecole Nationale Supérieure  
d'Electricité et de Mécanique  
Casablanca



# CHAPITRE I :

## Conception de système expert



## Introduction :

Dans ce chapitre, nous allons recourir à l'UML pour concevoir et modéliser notre système. Ce langage trouve son utilité dans un large éventail de domaines reposant sur la description de systèmes, offrant une diversité de diagrammes tels que les diagrammes de cas d'utilisation, de séquence, d'activités et de classes. En tant qu'outil, l'UML permet de spécifier, visualiser, modifier et construire les documents nécessaires au développement efficace d'un logiciel orienté objet. Il constitue un standard de modélisation reconnu pour représenter l'architecture logicielle.

## 1. Spécification des besoins :

Pour concevoir notre système expert nous avons suivi une démarche basée sur :

- Choisir la façon de la présentation de connaissances.
- Déterminer le type de raisonnement et la stratégie utilisée (chainage avant ou arrière).
- Concevoir l'interface utilisateur.
- Le prototype doit être testé et affiné par l'ingénieur de connaissances et l'expert de domaine en même temps.
- Le prototype peut être complété au fur et à mesure en ajoutant des nouveaux éléments dans la base de connaissance.
- Utilisation l'UML pour la conception.
- Les pannes détaillées d'un micro-ordinateur.

## 2. Les différents diagrammes UML :

### 2.1. Diagramme de Cas d'utilisation :

#### 2.1.1. Identification des acteurs :





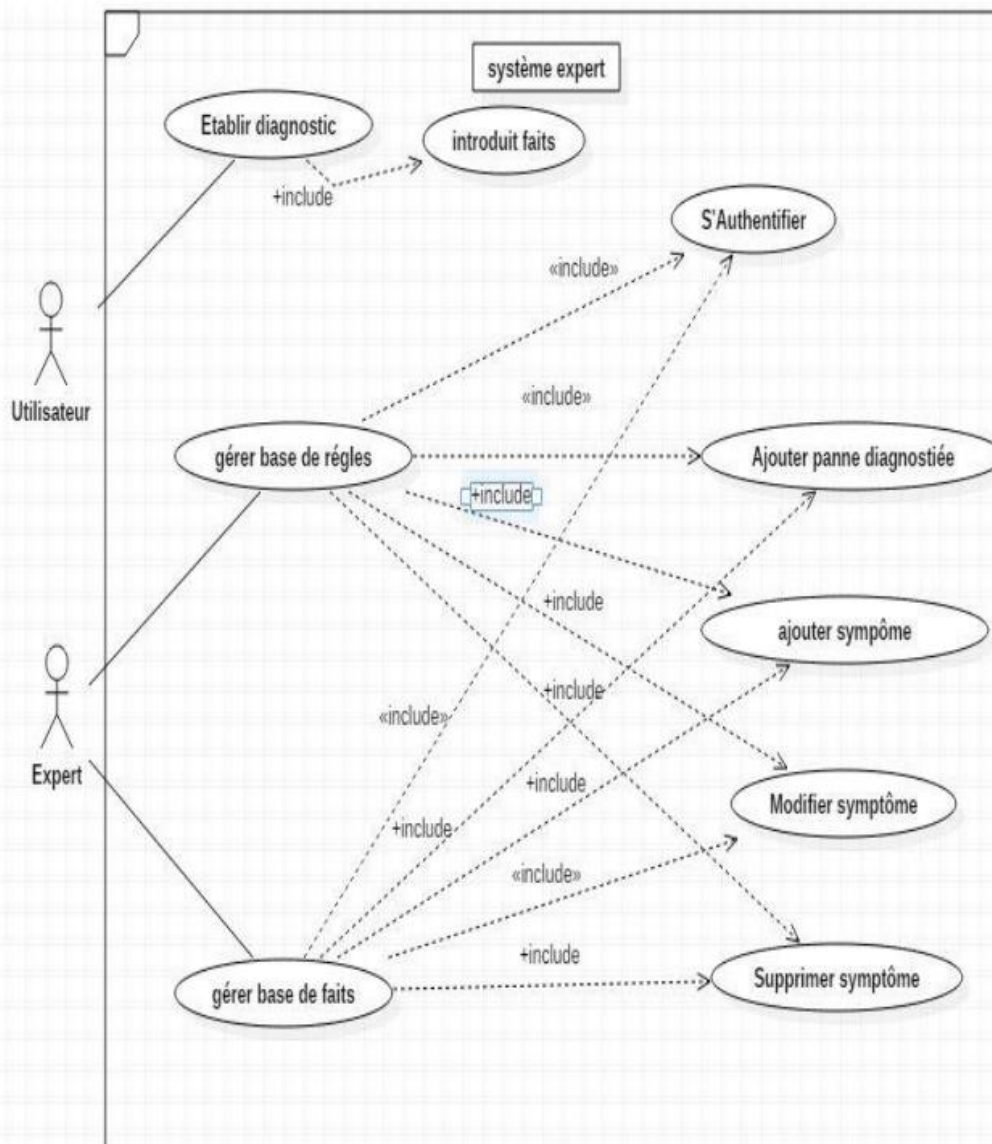


Figure 1 : Diagramme de cas d'utilisation

## 2.2. Diagramme de séquence :

### 2.2.1. Schéma de diagramme de séquence de cas d'utilisation (S'Authentifier)

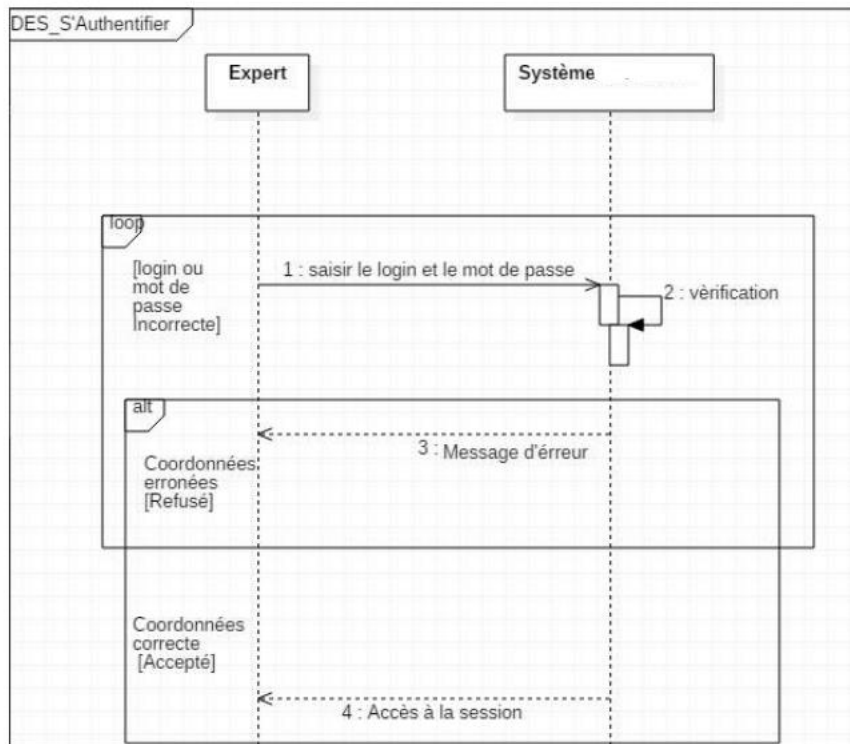


Figure 2 Diagramme de séquences de cas d'utilisation « S'Authentifier



### 2.2.2. Diagramme de séquence de cas d'utilisation (Etablir diagnostic) :

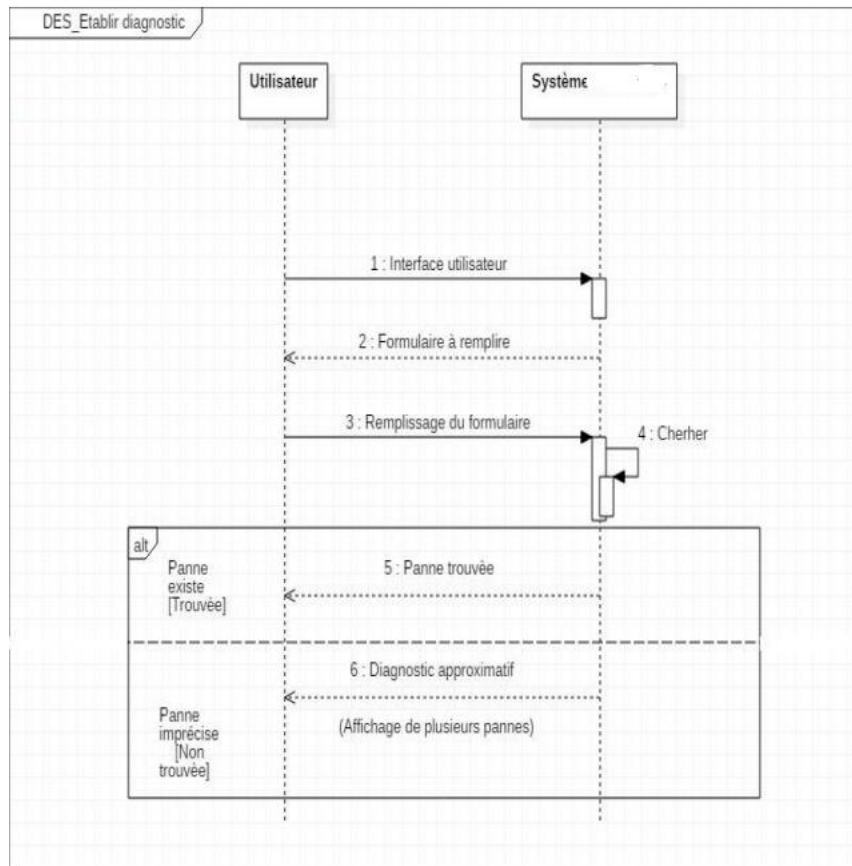


Figure 3 : Diagramme de séquence de cas d'utilisation « Etablir diagnostic »

### 2.2.3. Diagramme de séquence de cas d'utilisation (Ajouter règle)

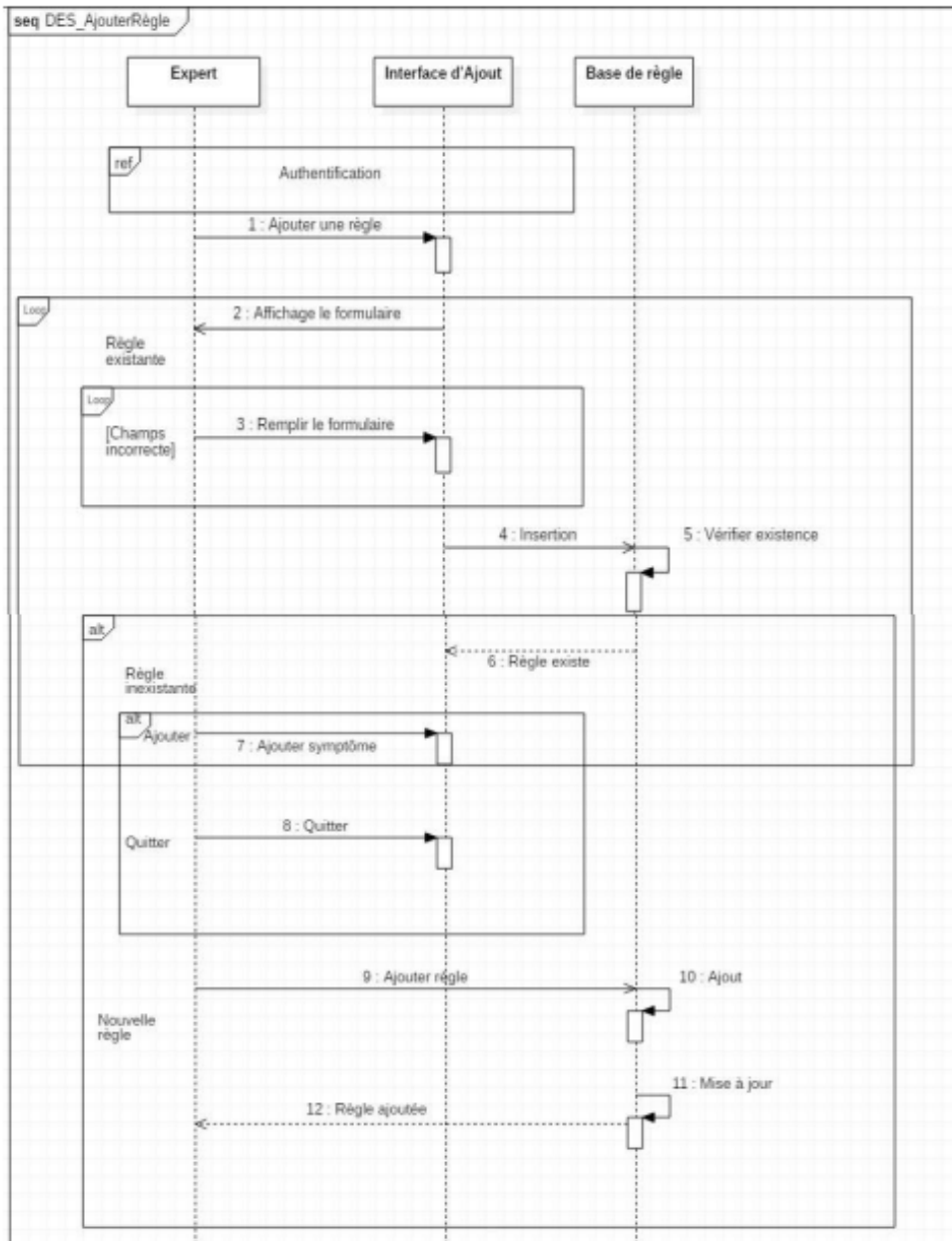
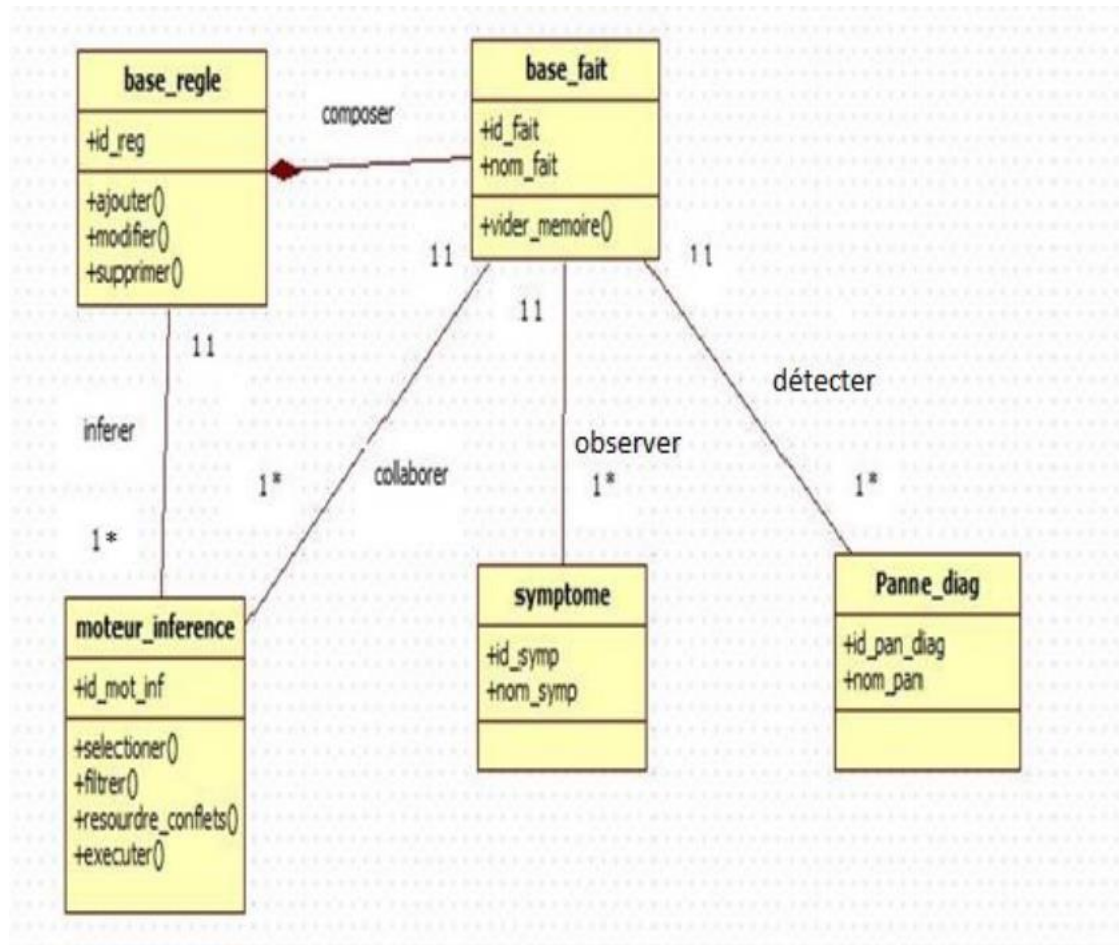


Figure 4 : Diagramme de séquence de cas d'utilisation « Ajouter règle »

### 3. Diagramme de classes :

Nous avons réalisé le diagramme suivant avec l'outil StarUML.



### Conclusion :

Dans ce chapitre nous avons modélisé les processus de système à l'aide de quelques diagrammes d'UML (diagramme de cas d'utilisation, diagrammes de séquence, diagramme de classes) et nous avons accompli la conception de système qui nous permettra la réalisation de l'application dans le chapitre qui suit.\*



Université Hassan II  
Ecole Nationale Supérieure  
d'Electricité et de Mécanique  
Casablanca



# CHAPITRE II :

## Réalisation de système expert

## Introduction :

Ce chapitre est dédié à la réalisation de notre application de système expert d'aide au diagnostic de pannes informatiques. Nous commencerons par discuter du choix du langage de développement, en mettant en avant les critères qui ont guidé notre décision. Ensuite, nous présenterons les différentes parties de l'application, en décrivant brièvement leurs fonctionnalités et leur importance dans le système global. Cette étape marque une avancée significative dans le projet, où nous passerons de la conception théorique à la concrétisation pratique de notre solution.

## 1. Choix de langage de développement :

Les raisons qui nous ont conduits à choisir Python comme langage de programmation pour ce projet de système expert sont les suivantes :

- Facilité d'apprentissage et de lecture : Python est reconnu pour sa syntaxe claire et concise, ce qui facilite la compréhension du code.
- Large bibliothèque standard : Python dispose d'une vaste bibliothèque standard qui offre des fonctionnalités pour divers domaines, y compris le développement d'applications graphiques avec Tkinter, le traitement des chaînes de caractères, etc. Cela nous permet de bénéficier d'une large gamme d'outils sans avoir à rechercher des bibliothèques tierces.
- Polyvalence et adaptabilité : Python est un langage polyvalent qui peut être utilisé pour une variété de tâches, de l'automatisation de





tâches simples à la création d'applications web complexes. Sa polyvalence nous permet d'étendre facilement les fonctionnalités de notre système expert à l'avenir, en fonction des besoins évolutifs du projet.

- Rapidité de développement : Python est réputé pour sa rapidité de développement grâce à sa syntaxe simple et à son approche orientée objet. Cela nous permet de prototyper rapidement notre système expert et d'itérer rapidement sur les fonctionnalités.

## 2. Description de l'application :

L'application fournit à l'utilisateur une interface où il peut se connecter soit au tant qu'utilisateur standard ou au tant qu'expert :

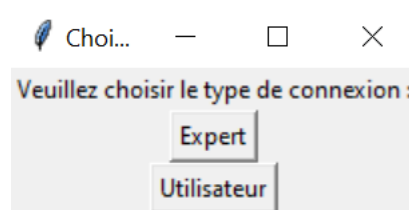


Figure 4 : Interface de connexion

Pour ce faire, l'utilisateur ou l'expert doit être inscrit dans l'un des fichiers : utilisateurs.txt ou experts.txt , selon son choix de connexion :

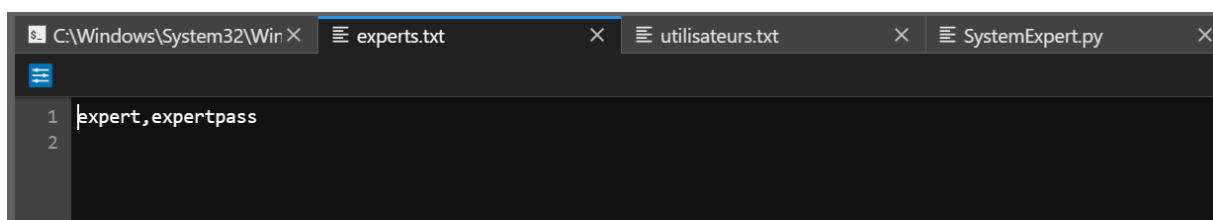


Figure 5 : Fichier experts.txt

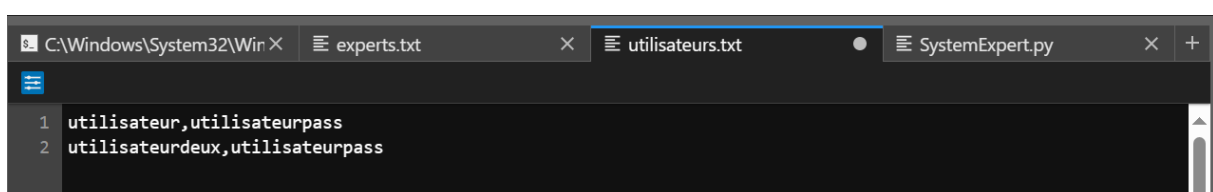






Figure 6 : Fichier utilisateurs.txt

## 2.1. Session Expert :

Dans un premier temps, nous nous connectons au tant qu'expert, en insérant le login et le mot de passe, il y'a aussi un bouton « ajouter expert », ce dernier permet l'ajout d'un nouveau expert, et un bouton retour qui permet le retour à la page de connexion.

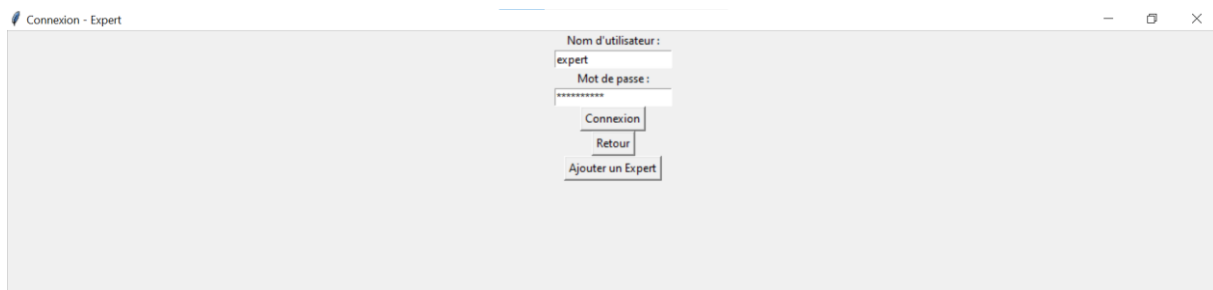


Figure 7 : interface de connexion expert

Après la connexion, nous trouverons l'interface ci-dessous, qui contient des boutons d'ajout, de modification, de suppression et d'affichage des pannes

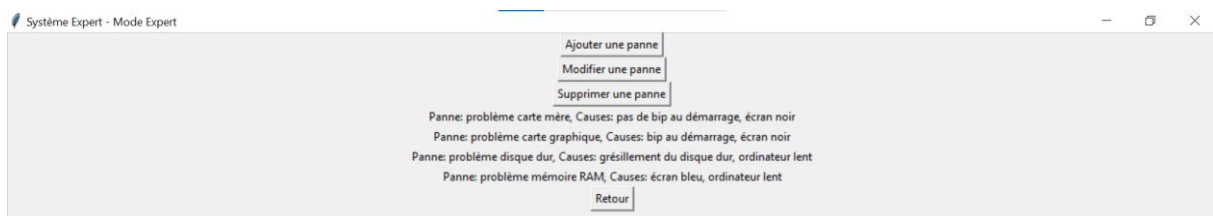


Figure 8 : Interface des opérations CRUD pour l'expert

Ici par exemple, nous tentons d'ajouter une nouvelle panne :

- Conditions : "Ordinateur ne s'allume pas", "Aucun voyant allumé"

- Action : "Problème d'alimentation"

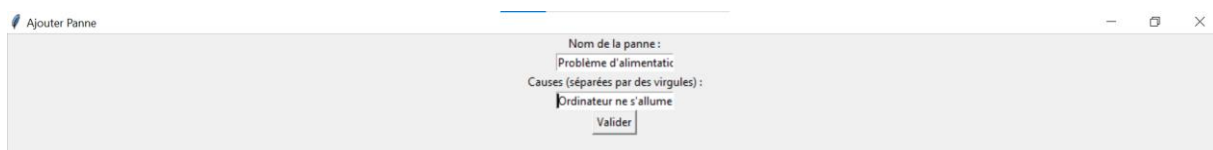


Figure 9 : Interface d'ajout d'une nouvelle panne

Voici le résultat :

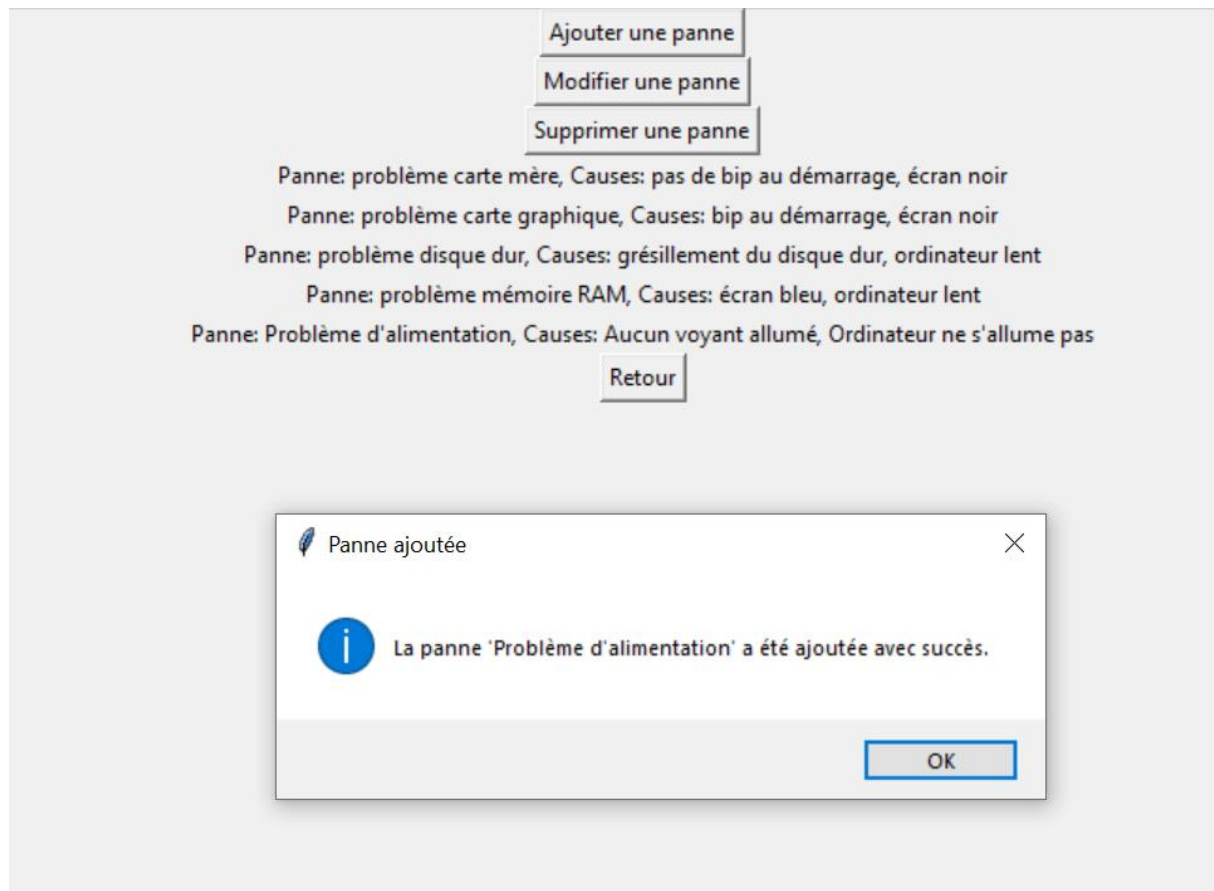


Figure 10 : Validation d'ajout d'une nouvelle panne

Pour modifier cette erreur , nous utilisons le bouton modifier , comme illustré ci-dessous :

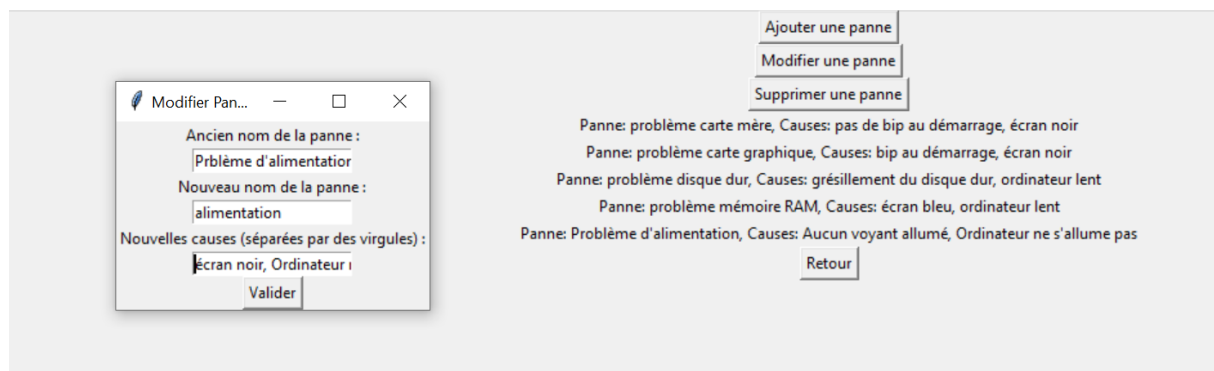


Figure 11 : Interface de modification des pannes

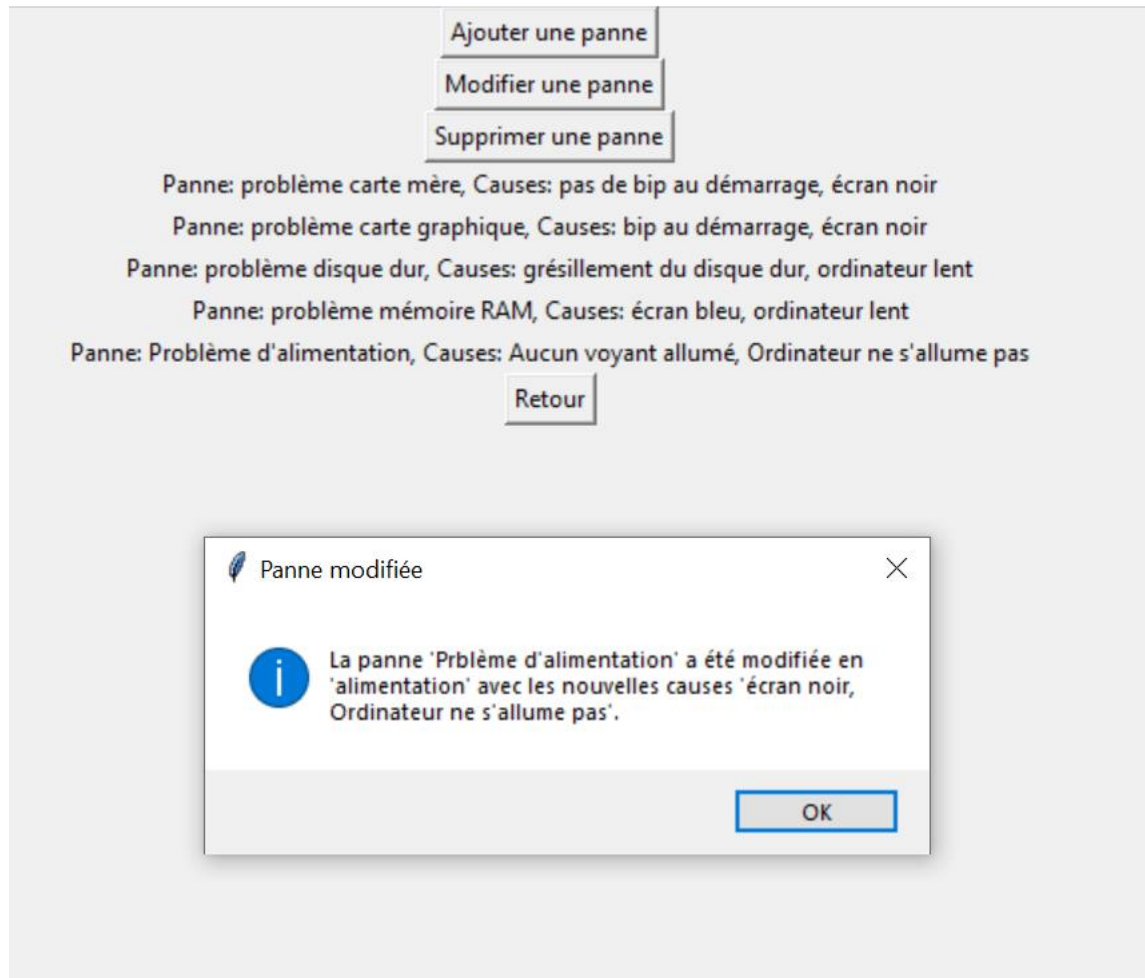


Figure 12 : Validation de modification de panne

Maintenant, nous voulons supprimer complètement cette panne :

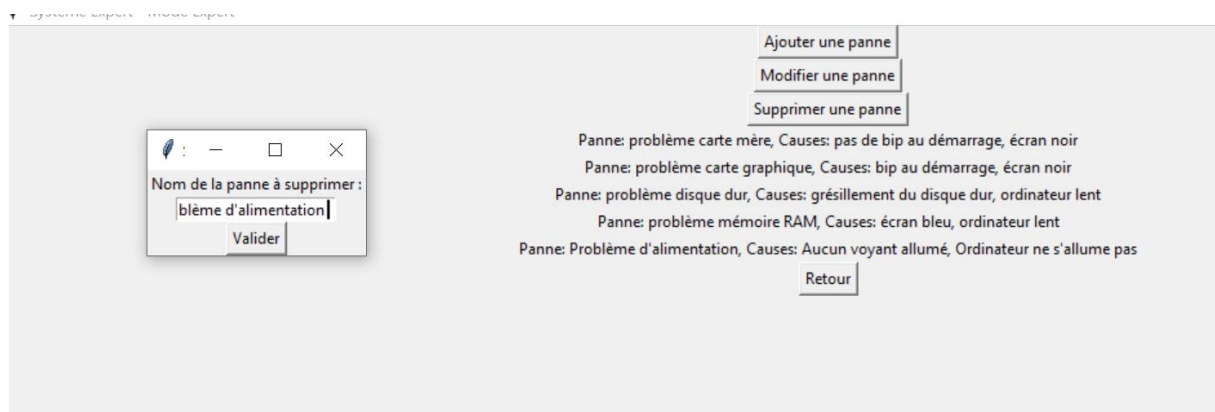


Figure 13 : Interface de suppression des pannes

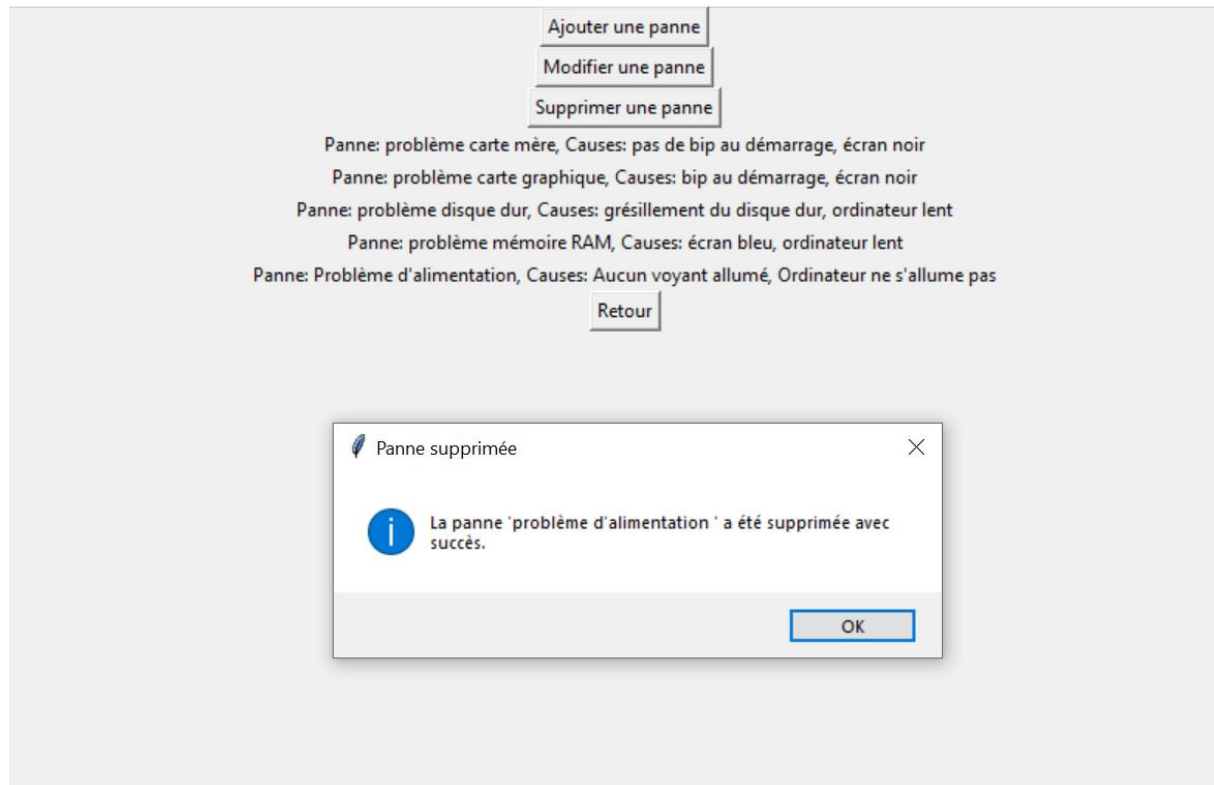


Figure 14 : Validation du Suppression de panne

## 2.2. Session utilisateur :

La première interface affichée à l'utilisateur après son choix de se connecter est la suivante, l'utilisateur peut se connecter avec son login et son mot de passe déjà enregistré, et s'il s'agit d'un nouvel utilisateur il peut s'ajouter en cliquant le bouton ajouter un utilisateur (Comme dans le cas de l'expert), pour faciliter la tâche à l'utilisateur, nous avons choisi de mettre les symptômes sous forme des boutons.



Nom d'utilisateur :  
utilisateur  
Mot de passe :  
\*\*\*\*\*  
Connexion  
Retour  
Ajouter un Utilisateur

Figure 15 ; Interface connexion utilisateur

Cette session est dédiée pour l'utilisateur standard, après avoir se connecter au tant qu'utilisateur, nous trouverons l'interface suivante, qui permet à l'utilisateur de choisir les symptômes , et lui fournit la panne , si la panne n'est pas trouvé , la phrase « Aucun organe défectueux trouvé pour les symptômes sélectionnés » , un bouton pour réinitialiser le choix est également fourni.

Système Expert - Mode Utilisateur

Veuillez sélectionner les symptômes observés :

écran noir | pas de bip au démarrage | bip au démarrage | ordinateur lent | grésillement du disque dur | écran bleu

Soumettre  
Réinitialiser  
Retour

Figure 16 : Interface pour choisir les symptômes

Veuillez sélectionner les symptômes observés :

écran noir | pas de bip au démarrage | bip au démarrage | ordinateur lent | grésillement du disque dur | écran bleu

Soumettre

Les organes susceptibles d'être défectueux sont : problème carte mère

Réinitialiser  
Retour

Figure 17 : Sélection des symptômes et affichage de l'organe défectueux



Veuillez sélectionner les symptômes observés :

écran noir	pas de bip au démarrage	bip au démarrage	ordinateur lent	grésillement du disque dur	écran bleu
------------	-------------------------	------------------	-----------------	----------------------------	------------

Soumettre

Aucun organe défectueux trouvé pour les symptômes sélectionnés.

Réinitialiser

Retour

Figure 18 : Message indiquant que la panne est introuvable

### 3. Choix de la représentation des connaissances et le mode de raisonnement :

Pour ce projet nous avons choisi le chaînage avant qui repose sur la déduction d'un fait particulier, on déclenche les règles dont les prémisses sont connues jusqu'à ce que le fait à déduire soit également connu ou qu'aucune règle ne soit plus déclenchable.

Entrée : une base de faits F, une base de règles R

Sortie : la base de faits F transformée

#### Algorithme :

répéter

S = ensemble des règles applicables de R

si  $S \neq \emptyset$  alors

a = choix d'une règle de S

application de la règle choisie a

marquer la règle a

finsi

jusqu'à  $S = \emptyset$

(Une règle applicable est une règle : non marquée et dont la prémisse est satisfaite)

Dans notre programme, nous avons implémenté cet algorithme de chaînage avant en utilisant Python avec la bibliothèque tkinter pour l'interface utilisateur. Le système expert fonctionne de la manière suivante :

1. L'utilisateur se connecte en tant qu'expert ou utilisateur.
2. L'utilisateur sélectionne les symptômes observés dans le mode utilisateur.
3. Le système utilise le chaînage avant pour déduire les organes susceptibles d'être défectueux en fonction des symptômes sélectionnés.
4. Dans le mode expert, l'expert peut afficher, ajouter, modifier ou supprimer des pannes et leurs causes.
5. Le système utilise le chaînage avant pour mettre à jour les informations en fonction des actions de l'expert.

## Conclusion :

Cette section constitue une étape cruciale dans le développement de notre application de système expert dédiée au diagnostic des pannes informatiques. Nous avons d'abord examiné attentivement le choix du langage de programmation, mettant en lumière les critères qui ont orienté notre sélection. Ensuite, nous avons présenté de manière concise les différentes composantes de l'application, en soulignant leurs fonctionnalités respectives et leur contribution essentielle au système dans son ensemble. Cette phase marque une transition significative dans notre projet, passant de la conceptualisation théorique à la mise en œuvre pratique de notre solution.



## Conclusion générale :

Après avoir parcouru les différentes étapes de conception, développement et réalisation de notre application de système expert pour le diagnostic des pannes informatiques, il est opportun de tirer quelques conclusions générales. Tout d'abord, ce projet nous a permis de mettre en pratique les concepts théoriques appris en matière d'intelligence artificielle, de programmation et de conception logicielle. En choisissant Python comme langage de développement, nous avons bénéficié de sa flexibilité, de sa richesse en bibliothèques et de sa facilité d'apprentissage, ce qui a grandement facilité le processus de création.

En détaillant les différentes parties de notre application, nous avons pu illustrer la complexité et la diversité des fonctionnalités nécessaires à la mise en œuvre d'un système expert efficace. De la gestion des utilisateurs à l'identification des pannes en passant par l'interface utilisateur conviviale, chaque aspect a été soigneusement étudié et implémenté pour offrir une expérience utilisateur optimale.

Enfin, ce projet nous a également confrontés à divers défis techniques et conceptuels, nous poussant à repousser nos limites et à rechercher des solutions innovantes. En somme, cette expérience a été enrichissante à bien des égards, nous permettant d'acquérir de nouvelles compétences, de consolider nos connaissances et de nous préparer à relever de nouveaux défis dans le domaine de l'intelligence artificielle et de l'informatique en général.