

* **Rapport de mini projet Programmation Systéme et Réseaux**



* **Messenger**

**Réalisé par** : **Elabed Sabrine**

**Jraidi Houssem**

**Ben Grira Wassim**

**Bouchelliga Jalel**

**Élèves ingénieur : 2ème année Génie informatique groupe C**

**Table des matières**

**Introduction ……………………………………….….......……...…..3**

**Partie théorique**

**Réseaux informatique…………………………..........………………4**

**Threading et la synchronisation……………..........………………...6**

**Partie pratique**

**Méthodologie utilisée dans le développement………………………7**

**Choix techniques……………………………………………………..7**

**Gestion de données…………………………………………………...8**

**Les difficultés rencontrées…………………………………………...9**

**Conclusion générale……………………………………………...........……….......9**

Introduction :

La messagerie instantanée est une moyen pour vous de communiquer instantanément avec vos ami(e)s.

La messagerie instantanée requiert l’emploi d’un [logiciel client](https://fr.wikipedia.org/wiki/Logiciel_client" \o "Logiciel client) qui se connecte à un [serveur](https://fr.wikipedia.org/wiki/Serveur_informatique" \o "Serveur informatique) de messagerie instantanée. Elle diffère du [courrier électronique](https://fr.wikipedia.org/wiki/Courrier_%C3%A9lectronique" \o "Courrier électronique) du fait que les conversations se déroulent instantanément (quasiment en temps réel, les contraintes temporelles n’étant pas fortes dans ces systèmes). La plupart des services modernes offrent un système de notification de présence, indiquant si les individus de la liste de contacts sont simultanément en ligne et leur disponibilité pour discuter.

Dans les tout premiers programmes de messagerie instantanée, chaque lettre apparaissait chez le destinataire dès qu’elle était tapée, et quand des lettres étaient effacées pour corriger une faute, cela se voyait également en temps réel. Cela faisait ressembler la communication à un coup de téléphone plutôt qu’à un échange de messages. Dans les programmes modernes de messagerie instantanée, le destinataire ne voit le message de l’expéditeur apparaître que lorsque celui-ci l’a validé.

La plupart des applications de messagerie instantanée permettent de régler un message de statut, qui remplit la même fonction qu’un message de répondeur téléphonique, par exemple pour indiquer la cause d’une indisponibilité.

I - **Partie théorique**

**Réseaux informatique :**

un réseau informatique est un ensemble d'équipement informatiques (ordinateur et périphériques) reliés entre eux grâce à des supports de communication (cable :réseau câblé, ou onde :réseau sans fil..) permettant la communication (transfert des informations électroniques) et le partage de ressources( matérielles et logicielles).

*Intérêt d'un réseau:*

* *Communiquer entre personne (messagerie, discussion en direct)*
* *Diminuer les coûts (par le partage des ressources matérielles(imprimante, disques,graveurs,,,)*
* *partager des données et d'applications*
* *permettre une communication efficace, rapide et peu coûteuse,*

*Ressource logicielles :*

* *système d'exploitation pour réseau*
* *protocoles de communication*
* *applications pour réseau (messagerie, logiciels de sécurité, logiciels de gestion de réseaux,*

*\*Ressources matérielles:*

* *connectique: câble, prise, fiche de connexion*
* *cartes: carte réseau, carte modem,..*
* *concentrateurs(hubs), routeurs, commutateurs(switch),point d'accès pour wifi(Wireless Fidelity)..*

*Protocoles Réseau :*

*-Pour communiquer dans le réseau, les différents ordinateurs et périphériques ont besoin d'un protocole.   
-un protocole réseaux est une norme régissant l'ensemble de communication entre différentes entités connectées à un réseau.   
Exemple de protocoles : TCP/IP, IPX/SPX, NetBios.*

*\* TCP/IP : (Transmission Control Protocol)/Internet Protocol)*

* *Protocole d'inter-connections de réseau utilisé par Internet et permettant la communication entre différents ordinateurs.*
* *Intégré dans la plupart des systèmes d'exploitation.*
* *Assure la communication entre les réseaux hétérogènes.*
* *Assure la communication entre les réseaux hétérogènes.*
* *Dans un réseau utilisant TCP/IP, chaque ordinateur est identifié d'une manière unique par une adresse IP.*

**Threading et la synchronisation :**

Les *threads* se distinguent du multi-processus plus classique par le fait que deux processus sont totalement indépendants et isolés les uns des autres, et ne peuvent interagir qu'à travers une Interface de programmation fournie par le système. D'un autre côté, les *threads* partagent une information sur l'état du processus, des zones de mémoires, ainsi que d'autres ressources.

En [programmation concurrente](https://fr.wikipedia.org/wiki/Programmation_concurrente" \o "Programmation concurrente), la **synchronisation** se réfère à deux concepts distincts mais liés : la synchronisation de *processus* et la synchronisation de *données*. La synchronisation de [processus](https://fr.wikipedia.org/wiki/Processus_(informatique)" \o "Processus (informatique)) est un mécanisme qui vise à bloquer l'exécution de certains [processus](https://fr.wikipedia.org/wiki/Processus" \o "Processus) à des points précis de leur [flux d'exécution](https://fr.wikipedia.org/wiki/Structure_de_contr%C3%B4le" \o "Structure de contrôle), de manière que tous les processus se rejoignent à des étapes relais données, tel que prévu par le programmeur. La synchronisation de données, elle, est un mécanisme qui vise à conserver la [cohérence](https://fr.wikipedia.org/wiki/Coh%C3%A9rence_(donn%C3%A9es)" \o "Cohérence (données)) des données telles que vues par différents processus, dans un environnement multitâche. Initialement, la notion de synchronisation est apparue pour la synchronisation de données.

II - **Partie pratique**

**Méthodologie utilisée dans le développement :**

Nous avons utilisé la méthode de Programmation extrême (XP) pour répartir notre travail:

- Se réunir chaque jour pour vérifier: quelles tâches nous fini?

Quelles tâches sont encore annulées? Quels sont les problèmes rencontrés?

- Améliorations du code chaque fois que nous trouvons

une solution plus facile ou un meilleur moyen de résoudre le problème.

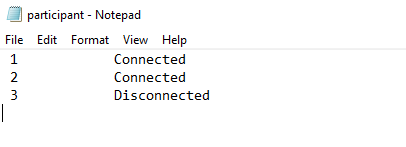
**Choix techniques :**

**Python** est un [langage de programmation](https://fr.wikipedia.org/wiki/Langage_de_programmation" \o "Langage de programmation) [objet](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet" \o "Programmation orientée objet) [interprété](https://fr.wikipedia.org/wiki/Langage_interpr%C3%A9t%C3%A9" \o "Langage interprété), multi-[paradigme](https://fr.wikipedia.org/wiki/Paradigme_(programmation)" \o "Paradigme (programmation)) et [multiplateformes](https://fr.wikipedia.org/wiki/Plate-forme_(informatique)" \o "Plate-forme (informatique)). Il favorise la [programmation impérative](https://fr.wikipedia.org/wiki/Programmation_imp%C3%A9rative" \o "Programmation impérative) [structurée](https://fr.wikipedia.org/wiki/Programmation_structur%C3%A9e" \o "Programmation structurée), [fonctionnelle](https://fr.wikipedia.org/wiki/Programmation_fonctionnelle" \o "Programmation fonctionnelle) et [orientée objet](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet" \o "Programmation orientée objet).

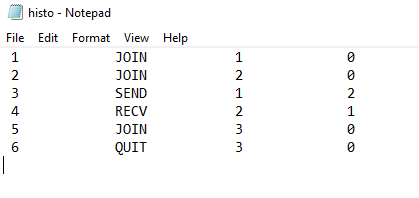
**Tkinter** est la bibliothèque graphique libre d'origine pour le langage Python, permettant la création d'interfaces graphiques.

**Gestion de données :**

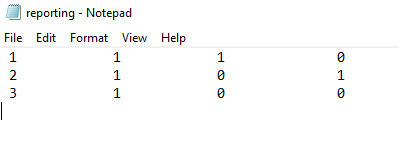
* Fichier participant.txt



* Fichier histo.txt



* Fichier reporting.txt



**Les difficultés rencontrées :**

-manipulation du langage python.

-Établir la connexion entre les ordinateurs en utilisant le protocole TCP

-manipulation des fichiers.

-Respecter la contrainte du temps

**Conclusion générale**

Au sein de ce projet, nous avons la chance de maitriser parfaitement les notions du cours :

-Comprendre l’architecture client / serveur

-Apprendre à développer avec le langage python

-Utiliser l’interface graphique de Python avec Tkinter