

**République Algérienne Démocratique et Populaire**

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**

**Université Saad Dahlab Blida 1**



**CDTA**

**Faculté des Sciences**

**Département d'Informatique**

**Mémoire de fin d'étude pour l'obtention du diplôme de Master en Informatique**

**Option : Systèmes Informatiques et Réseaux**

**Présenté par :**

**M<sup>elle</sup> DEFFAF Imene**

**M<sup>elle</sup> FORTAS Hassiba**

**Thème**

**Routage Multicritères des Flux Multimédia au Niveau  
Applicatif**

**Organisme d'accueil : Centre de Développement des Technologies Avancées  
(CDTA).**

**Soutenu le 30/09/2019 devant le jury composé de :**

**M<sup>r</sup> Ould Khaoua Mohamed**

**Président**

**M<sup>me</sup> Boutoumi Bachira**

**Examineur**

**M<sup>me</sup> AROUSSI Sana**

**Promotrice**

**D<sup>r</sup> HENTOUT Abdelfetah**

**Encadreur**

**M<sup>r</sup> TIBERKAK Allal**

**Encadreur**

**Promotion : 2018-2019**

## **Dédicaces**

C'est avec profonde gratitude et sincères mots, Que je dédie ce modeste travail de fin d'étude :

À mes chers parents, qui ont sacrifié leur vie pour ma réussite et m'ont éclairé le chemin par leurs conseils judicieux, c'est grâce à vous et pour vous que j'ai fait mon mémoire. Aucun mot sur cette page ne saurait exprimer ce que je vous dois, ni combien je vous aime. J'espère qu'un jour, je pourrais leurs rendre un peu de ce qu'ils ont fait pour moi, que dieu leur prête bonheur et longue vie.

À mon frère Mohamed et ma sœur Yousra, pour avoir contribué à la réussite de ce travail d'une manière indirecte, et pour leurs encouragements.

À mes plus fidèles amies, et sœurs de cœur, toi FatmaZohra, Houria et Manel. Je prends le temps de vous remercier pour votre soutien et encouragement, je vous en serais éternellement reconnaissante.

Merci à tous.

**DEFFAF Imene**

## **Dédicaces**

C'est avec profonde gratitude et sincères mots, Que je dédie ce modeste travail de fin d'étude :

À mes chers parents, qui ont sacrifié leur vie pour ma réussite et m'ont éclairé le chemin par leurs conseils judicieux, c'est grâce à vous et pour vous que j'ai fait mon mémoire. Aucun mot sur cette page ne saurait exprimer ce que je vous dois, ni combien je vous aime. J'espère qu'un jour, je pourrais leurs rendre un peu de ce qu'ils ont fait pour moi, que dieu leur prête bonheur et longue vie.

À Mes frères Soufiene et Hamza, mes sœurs Rafika, Sihem et Khaoula, pour avoir contribué à la réussite de ce travail d'une manière indirecte, et pour leurs encouragements.

À mes plus fidèles amies, et sœurs de cœur, toi FatmaZohra, Houda, Bouchra, Yasmine et Lydia. Je prends le temps de vous remercier pour votre soutien et encouragement, je vous en serais éternellement reconnaissante.

Merci à tous.

**FORTAS Hassiba**

## **Remerciements**

Nous tenons à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce modeste travail.

Je commence d'abord à remercier notre encadreur Monsieur TIBERKAK et Monsieur HENTOUT pour leurs aides précieuses, et disponibilité durant toute cette période d'encadrement.

J'exprime ma profonde gratitude à Madame AROUSSI Sana pour avoir accepté de nous encadrer, pour sa disponibilité tout au long de la réalisation de ce mémoire, pour la qualité de ses conseils et pour les nombreuses discussions que nous avons eues. Elle est et elle sera pour nous un exemple de rigueur et de droiture dans l'exercice de la profession.

Nos vifs remerciements aux membres du jury pour l'intérêt accordé à notre travail en l'examinant minutieusement et avec attention.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail, principalement à tout l'effectif du Centre de Développement des Technologies Avancées CDTA.

## Résumé

Les webinaires réfèrent aux réunions interactives et aux séminaires via internet (web séminaire). Récemment, les universités ont opté pour l'utilisation de ce service afin de faciliter l'enseignement à distance et le travail collaboratif. L'acceptabilité et le succès de ce service sont conditionnés par le degré de satisfaction des utilisateurs finaux mesuré en termes de la Qualité d'Expérience (Quality of Experience, QoE). Cette dernière dépend de la qualité de service (Quality of Service, QoS) offerte par le réseau de communication. Ceci nous amène à développer un système de webinaire qui prend en considération ces deux notions QoS et QoE.

L'objectif principal de ce travail consiste à développer un mécanisme de routage multicritères de flux multimédia basé sur un modèle de corrélation QoS-QoE pour les webinaires dans les réseaux IP. La solution proposée permet de sélectionner le meilleur serveur de diffusion qui offre une meilleure qualité afin d'acheminer les flux du conférencier aux participants. Les résultats préliminaires obtenus montrent le bon fonctionnement de cette solution. Les résultats préliminaires obtenus montrent l'efficacité et les performances élevées en termes de délai, de charge, de bande passante, etc.

**Mots clés :** Réseaux IP, Webinaire, Qualité de Service, Qualité d'Expérience, Routage multicritères.

## **Abstract**

Webinar refer to interactive meetings and seminars via internet (web seminar). Recently, universities have opted to use this service to facilitate distance education and collaborative work. The degree of acceptability and success of this service depend on the satisfaction of the end-users requirements, which is measured by the Quality of Experience (QoE) that is related to the selected Quality of Service (QoS). To achieve this objective, we developed a webinar system that takes into consideration both concepts QoS and QoE.

The main objective of this work is to develop a multi-criteria routing mechanism of multimedia stream based on a QoS-QoE correlation model for webinars in IP networks. The proposed solution selects the best streaming server that provides the best quality for routing data stream from the speaker to the participants. Preliminary results show that this solution works well. For this aim, we studied webinar systems with the notion of QoS and QoE to find a correlation model to estimate the QoE. Obtained preliminary results show the effectiveness and high performances in terms of delay, load balancing, bandwidth, etc.

**Keywords:** IP networks, Webinar, Quality of service, Quality of Experience, Multi-criteria routing.

## ملخص

تشير ندوات الويب إلى اجتماعات وندوات تفاعلية عبر الإنترنت (ندوة عبر الإنترنت). في الآونة الأخيرة، اختارت الجامعات استخدام هذه الخدمة لتسهيل التعليم عن بعد والعمل التعاوني. تخضع مقبولية ونجاح هذه الخدمة لدرجة رضا المستخدم النهائي المقاسة من حيث جودة التجربة (QoE). تعتمد هذه الأخيرة على جودة الخدمة (QoS) التي تقدمها شبكة الاتصالات. هذا يقودنا إلى تطوير نظام الوبينار الذي يأخذ في الاعتبار كلا هذين المفهومين QoS و QoE.

الهدف الرئيسي من عملنا هو تطوير آلية توجيه متعددة المعايير لتوجيه التدفقات تعتمد على نموذج ارتباط QoS-QoE للندوات عبر الإنترنت في شبكات IP. يقوم حلنا بتحديد أفضل خادم بث يوفر أفضل جودة لتوجيه التدفقات للمشاركين. توضح النتائج الأولية مدى جودة أداء حلنا من حيث الأداء مثل التأخير وموازنة التحميل وعرض النطاق الترددي وما إلى ذلك.

**الكلمات المفتاحية:** شبكات IP، الوبينار، جودة الخدمة، جودة التجربة، التوجيه متعدد المعايير.

# Table des matières

|  |           |
|--|-----------|
| <b>TABLE DES FIGURES :</b> .....                           | <b>11</b> |
| <b>TABLE DES TABLEAUX</b> .....                            | <b>13</b> |
| <b>GLOSSAIRE</b> .....                                     | <b>14</b> |
| <b>INTRODUCTION GENERALE</b> .....                         | <b>16</b> |
| <b>CHAPITRE 1</b> .....                                    | <b>18</b> |
| <b>SERVICE WEBINAIRE</b> .....                             | <b>18</b> |
| <b>INTRODUCTION</b> .....                                  | <b>18</b> |
| <b>1. DIFFUSION DE FLUX MULTIMEDIA EN TEMPS REEL</b> ..... | <b>18</b> |
| 1.1    MODE DE DIFFUSION.....                              | 20        |
| 1.2    PROTOCOLES MULTIMEDIAS .....                        | 21        |
| 1.2.1    TCP.....  | 21        |
| 1.2.2    UDP.....  | 22        |
| 1.2.3    RTP.....  | 22        |
| 1.2.4    RTCP.....   | 23        |
| 1.2.5    SIP.....  | 23        |
| 1.2.6    SDP.....  | 23        |
| <b>2. WEBINAIRE</b> .....                                  | <b>24</b> |
| 2.1    ÉLÉMENTS DE SYSTEMES WEBINAIRES.....                | 24        |
| 2.2    FONCTIONNALITES DE SYSTEME WEBINAIRE.....           | 25        |
| <b>3. ÉTUDE DES SYSTEMES WEBINAIRES EXISTANTS</b> .....    | <b>25</b> |
| 3.1    SKYPE.....  | 26        |
| 3.2    HANGOUTS GOOGLE+.....                               | 27        |
| 3.3    WEBRTC.....   | 28        |
| 3.4    UNIVWEBINAR .....                                   | 29        |
| 3.5    ÉTUDE COMPARATIVE.....                              | 30        |
| <b>4. CONCLUSION</b> .....                                 | <b>32</b> |
| <b>CHAPITRE 2</b> .....                                    | <b>33</b> |
| <b>QUALITE DE SERVICE VS. QUALITE D'EXPERIENCE</b> .....   | <b>33</b> |

|  |           |
|--|-----------|
| <b>1. INTRODUCTION .....</b>   | <b>33</b> |
| <b>2. QUALITE DE SERVICE .....</b>   | <b>34</b> |
| 2.1    DEFINITION.....   | 34        |
| 2.2    PARAMETRES DE LA QUALITE DE SERVICE .....                             | 34        |
| 2.2.1 <i>Calcul des paramètres de QoS</i> .....                              | 35        |
| 2.2.2 <i>Relation entre les paramètres de qualité de service</i> .....       | 36        |
| 2.3    TECHNIQUES DE MESURE DE LA QUALITE DE SERVICE .....                   | 37        |
| <b>3. QUALITE D'EXPERIENCE .....</b>   | <b>38</b> |
| 3.1    DEFINITION.....   | 39        |
| 3.2    PARAMETRES AFFECTANT LA QOE .....                                     | 39        |
| 3.3    RELATION ENTRE LA QoS ET LA QOE.....                                  | 42        |
| 3.3.1 <i>Modèle Hourglass de QoE</i> .....                                   | 42        |
| 3.3.2 <i>Modèle de corrélation QoE-QoS proposé par Kim et al. [28]</i> ..... | 46        |
| <b>4. CONCLUSION .....</b>   | <b>48</b> |
| <b>CHAPITRE 3 .....</b>  | <b>50</b> |
| <b>CONCEPTION DE LA SOLUTION PROPOSEE.....</b>                               | <b>50</b> |
| <b>1. INTRODUCTION .....</b>   | <b>50</b> |
| <b>2. DIAGRAMME DE CAS D'UTILISATION.....</b>                                | <b>50</b> |
| 2.1    GESTION DES WEBINAIRES .....  | 52        |
| 2.2    GESTION DES MCU .....   | 53        |
| <b>3. ARCHITECTURE DU SYSTEME PROPOSE .....</b>                              | <b>53</b> |
| 3.1    FONCTIONNEMENT .....  | 54        |
| 3.2    PROCESSUS DE SELECTION DU MMCU .....                                  | 56        |
| <b>4. DIAGRAMME DE SEQUENCE « GESTION DE WEBINAIRE » .....</b>               | <b>57</b> |
| <b>5. DIAGRAMME DE CLASSES.....</b>  | <b>60</b> |
| <b>6. CONCLUSION .....</b>   | <b>61</b> |
| <b>CHAPITRE 4 .....</b>  | <b>62</b> |
| <b>IMPLEMENTATION, TESTS ET RESULTATS.....</b>                               | <b>62</b> |
| <b>1. INTRODUCTION .....</b>   | <b>62</b> |
| <b>2. IMPLEMENTATION .....</b>   | <b>62</b> |

|  |           |
|--|-----------|
| <b>3. PRESENTATION DE LA PLATEFORME « EDUC-WEBINAR »</b> | <b>63</b> |
| 3.1 ESPACE ADMINISTRATEUR                                | 64        |
| 3.1.1 <i>Gestion de MCU</i>                              | 64        |
| 3.1.2 <i>Gestion des comptes</i>                         | 65        |
| 3.2 ESPACE UTILISATEUR                                   | 66        |
| <b>4. TESTS EXPERIMENTAUX</b>                            | <b>68</b> |
| 4.1 ENVIRONNEMENT DE TESTS                               | 69        |
| 4.2 SCENARIO DE TEST                                     | 69        |
| 4.2.1 <i>Configuration de la « Machine A »</i>           | 70        |
| 4.2.2 <i>Configuration de la « Machine B »</i>           | 73        |
| <b>5. RESULTATS OBTENUS</b>                              | <b>73</b> |
| <b>6. CONCLUSION</b>                                     | <b>75</b> |
| <b>CONCLUSION GENERALE</b>                               | <b>76</b> |
| <b>BIBLIOGRAPHIE</b>                                     | <b>78</b> |
| <b>WEBOGRAPHIE</b>                                       | <b>81</b> |
| <b>ANNEXE A</b>  | <b>82</b> |
| <b>LOGICIELS ET LANGAGES DE DEVELOPPEMENT</b>            | <b>82</b> |
| <b>ANNEXE B</b>  | <b>84</b> |
| <b>LOGICIEL DE TEST : NETEM</b>                          | <b>84</b> |

## Table des figures :

|   |    |
|---|----|
| Figure 1. Différents types de diffusion en temps réel [3].                  | 19 |
| Figure 2. Unité de contrôle multipoint.                                     | 20 |
| Figure 3. Pile de protocoles [8].   | 21 |
| Figure 4. Architecture générale des systèmes webinaire                      | 24 |
| Figure 5. Architecture du Skype [20].                                       | 26 |
| Figure 6. Architecture de Hangouts [20].                                    | 28 |
| Figure 7. Architecture de WebRTC [21].                                      | 29 |
| Figure 8. Architecture de UnivWebinar [1].                                  | 30 |
| Figure 9. Paramètres affectant la QoE.                                      | 40 |
| Figure 10. Le modèle <i>Hourglass</i> de QoE [38].                          | 43 |
| Figure 11. Modèle de corrélation QoS-QoE [35].                              | 46 |
| Figure 12. Diagramme général de cas d'utilisation.                          | 51 |
| Figure 13. Diagramme de cas d'utilisation « gestion des webinaires ».       | 52 |
| Figure 14. Diagramme de cas d'utilisation « gestion des MCU ».              | 53 |
| Figure 15. Architecture du système « <i>Educ_Webinar</i> ».                 | 54 |
| Figure 16. Fonctionnement de la plateforme « <i>Educ_Webinar</i> ».         | 55 |
| Figure 17. Sélection de MMCUP ou MMCUC.                                     | 57 |
| Figure 18. Différentes interactions entre le conférencier et la plateforme. | 58 |
| Figure 19. Différentes interactions entre le participant et la plateforme.  | 59 |
| Figure 20. Diagramme de classes.  | 60 |
| Figure 21. Logiciels de développement.                                      | 63 |
| Figure 22. Inscription et authentification.                                 | 64 |
| Figure 23. Administration des MCU.  | 65 |
| Figure 24. Gestion des comptes.   | 66 |
| Figure 25. Création et lancement d'un webinaire.                            | 67 |

|  |    |
|--|----|
| Figure 26.Consulter et rejoindre un webinaire..... | 68 |
| Figure 27.Environnement de tests.....              | 69 |
| Figure 28.Lacement de MCU.....                     | 72 |
| Figure 30. Lancement de serveur de gestion.....    | 72 |
| Figure 31. Exemple 1 du NetEm [47]. ....           | 84 |
| Figure 32. Exemple 2 du NetEm [47]. ....           | 85 |

## Table des tableaux

|  |    |
|--|----|
| Tableau 1. Comparaison des systèmes webinaires .....   | 31 |
| Tableau 2. Calcul des paramètres de la qualité de service selon la norme UIT [27]. .....                                 | 36 |
| Tableau 3. Tableau comparatif entre les techniques actives et les techniques passives. ...                               | 38 |
| Tableau 4. Paramètre et plage de QoS liés à la QoE [28]. .....   | 48 |
| Tableau 5. Guide pour les classes de QoS [28]. .....   | 48 |
| Tableau 6. Définitions des classes de qualité de service du réseau IP et objectifs de performances du réseau [28]. ..... | 48 |
| Tableau 7. Description du diagramme global de cas d'utilisation. ....  | 51 |
| Tableau 8. Description de cas d'utilisation « gestion des webinaires ». .....  | 52 |
| Tableau 9. Description de cas d'utilisation « gestion des webinaires ». .....  | 53 |
| Tableau 10. Tableau descriptif des classes. ....   | 61 |
| Tableau 11. Caractéristiques de machines utilisées. ....   | 69 |
| Tableau 12. Configuration des MCU. ....  | 70 |
| Tableau 13. Paramètres du réseau. ....   | 71 |
| Tableau 14. Table de tests .....   | 73 |
| Tableau 15. Table de test2. ....   | 75 |

## **Glossaire**

|              |  |
|--------------|--|
| <b>IETF</b>  | <b>Internet Engineering Task Force.</b>                |
| <b>IP</b>    | <b>Internet Protocol.</b>                              |
| <b>IPDV</b>  | <b>IP Packet Delay Variation.</b>                      |
| <b>IPER</b>  | <b>IP Packet Error Ratio.</b>                          |
| <b>IPLR</b>  | <b>IP Packet Loss Ratio.</b>                           |
| <b>IPPT</b>  | <b>IP Packet Throughput.</b>                           |
| <b>ISO</b>   | <b>International Organization for Standardization.</b> |
| <b>ITU-T</b> | <b>International Telecommunication Union.</b>          |
| <b>M2M</b>   | <b>Machin to Machin.</b>                               |
| <b>MCU</b>   | <b>Multipoint Control Unit.</b>                        |
| <b>MMCU</b>  | <b>Meilleur MCU</b>                                    |
| <b>MMCUC</b> | <b>Meilleur MCU pour le Conférencier.</b>              |
| <b>MMCUP</b> | <b>Meilleur MCU pour le Participant.</b>               |
| <b>MQTT</b>  | <b>Message Queue Telemetry Transport.</b>              |
| <b>NetEm</b> | <b>Network Emulator.</b>                               |
| <b>QoD</b>   | <b>Quality of Delivery.</b>                            |
| <b>QoDA</b>  | <b>Quality of Delivery by Application.</b>             |
| <b>QoDN</b>  | <b>Quality of Delivery by Network</b>                  |
| <b>QoE</b>   | <b>Quality of Experience.</b>                          |
| <b>QoP</b>   | <b>Quality of Presentation.</b>                        |
| <b>QoS</b>   | <b>Quality of Service.</b>                             |
| <b>RTCP</b>  | <b>Real Time Control Protocol.</b>                     |
| <b>RTP</b>   | <b>Real-Time Transfer Protocol.</b>                    |
| <b>SG</b>    | <b>Server de Gestion.</b>                              |
| <b>SIP</b>   | <b>Session Initiation Protocol.</b>                    |

|             |                                    |
|-------------|------------------------------------|
| <b>SSL</b>  | <b>Secure Sockets Layer.</b>       |
| <b>TCP</b>  | <b>Transport Control Protocol.</b> |
| <b>TLS</b>  | <b>Transport Layer Security.</b>   |
| <b>UDP</b>  | <b>User Data Protocol</b>          |
| <b>VoD</b>  | <b>Video on Demand.</b>            |
| <b>VoIP</b> | <b>Voice over IP.</b>              |

## Introduction générale

Les progrès technologiques du réseau Internet ont permis le développement de nouvelles applications multimédia. La vidéoconférence, la téléconférence et le webinaire sont devenus des domaines importants de recherche et de développement pour l'industrie des télécommunications.

Pour le travail collaboratif et l'enseignement à distance, les webinaires ont révolutionné l'art de se réunir où chacun peut maintenant dialoguer ou apprendre auprès des meilleurs, même s'il vit à l'autre bout de la planète. Pour les entreprises, le webinaire est devenu un outil de marketing redoutable qui facilite grandement le recrutement de prospects, valorise les relations avec les clients et offre un support technique particulièrement efficace.

En Algérie, le Centre de Développement des Technologies Avancées (CDTA) s'intéresse à développer un webinaire destiné pour les universités afin d'encourager et de favoriser la collaboration en recherche entre les enseignants et les étudiants. Dans le cadre de ce projet, « *Univ\_webinar* » [1] est un prototype d'une solution permettant la diffusion du flux multimédia sur plusieurs participants à travers des nœuds de diffusion. Cette solution permet d'acheminer le flux multimédia vers le participant en se basant sur la bande passante et en empruntant le plus court chemin entre les participants et les nœuds possédant le flux de la conférence. Cependant, dans la réalité, la situation est plus complexe car les services de webinaires exigent, d'une part, de fortes contraintes en termes de plusieurs paramètres de qualité de service (Quality of Service, QoS) : bande passante, délai, gigue (variation de délai) et taux de perte des paquets. D'autre part, l'acceptabilité et le succès de ces nouveaux services sont conditionnés par le degré de satisfaction de leurs utilisateurs finaux qui est déterminé par la Qualité d'Expérience (Quality of Experience, QoE). Il s'avère donc nécessaire de développer des architectures et des mécanismes capables de prendre en compte ces deux notions de qualité (QoE et QoS). C'est dans cette optique que s'inscrit ce travail de master.

L'objectif consiste à développer un mécanisme de diffusion multicritères de flux multimédia pour les webinaires. Un tel mécanisme prend en compte les deux notions de

qualité QoE et QoS, afin de fournir aux utilisateurs un meilleur service avec une meilleure qualité.

Ce mémoire est structuré en deux parties, présentant respectivement la partie théorique et la partie pratique :

- La partie théorique comporte deux chapitres :
  - Le premier est consacré au service webinaire, où nous allons présenter les différents outils de webinaires existants afin de proposer une solution au problème posé.
  - Le deuxième chapitre a pour objectif de trouver un modèle de corrélation QoE-QoS sur lequel nous allons nous baser dans la gestion de la diffusion du flux multimédia.
- La partie pratique comprend elle aussi deux chapitres :
  - Le troisième chapitre décrit la solution proposée pour la diffusion du flux multimédia.
  - Le dernier chapitre concerne l'implémentation du système proposé, avec des tests expérimentaux pour présenter ses différentes fonctionnalités.

Nous terminons ce mémoire avec une conclusion générale où nous récapitulons les principaux résultats obtenus et présentons quelques perspectives futures.

# Chapitre 1

## Service webinaire

### Introduction

La popularité de services de webinaire et vidéo-conférence a augmenté de manière significative ces dernières années. Les outils de webinaires sont devenus une forme de communication indispensable et attirent de plus en plus l'attention avec l'avancement des technologies de l'apprentissage en ligne. Ces dernières facilitent la communication en temps réel et enrichissent l'interactivité dans un environnement d'apprentissage en ligne [1].

Dans ce premier chapitre, nous commençons par définir ce qu'est une diffusion de flux multimédia en temps réel, en citant leurs différents types, leur mode de diffusion, ainsi que les différents protocoles multimédias utilisés pour gérer et véhiculer le flux. Ensuite, nous présentons un service de communication qui se base sur la diffusion de flux multimédia ; il s'agit de « *Webinaire* ». Nous terminons par une étude comparative et une analyse des systèmes existants dans la littérature.

### 1. Diffusion de flux multimédia en temps réel

Par définition, un stream est un flux de données, de son ou de vidéo transporté par le réseau. Le streaming, quant à lui, est une technique de diffusion et de lecture en ligne et en continu de données multimédias, qui évite le téléchargement des données et permet la diffusion en direct. Le streaming permet la lecture d'un flux audio ou vidéo, à mesure qu'il est diffusé [2].

Comme montré par la figure 1 ci-dessous, plusieurs façons existent pour faire une diffusion en temps réel [3].

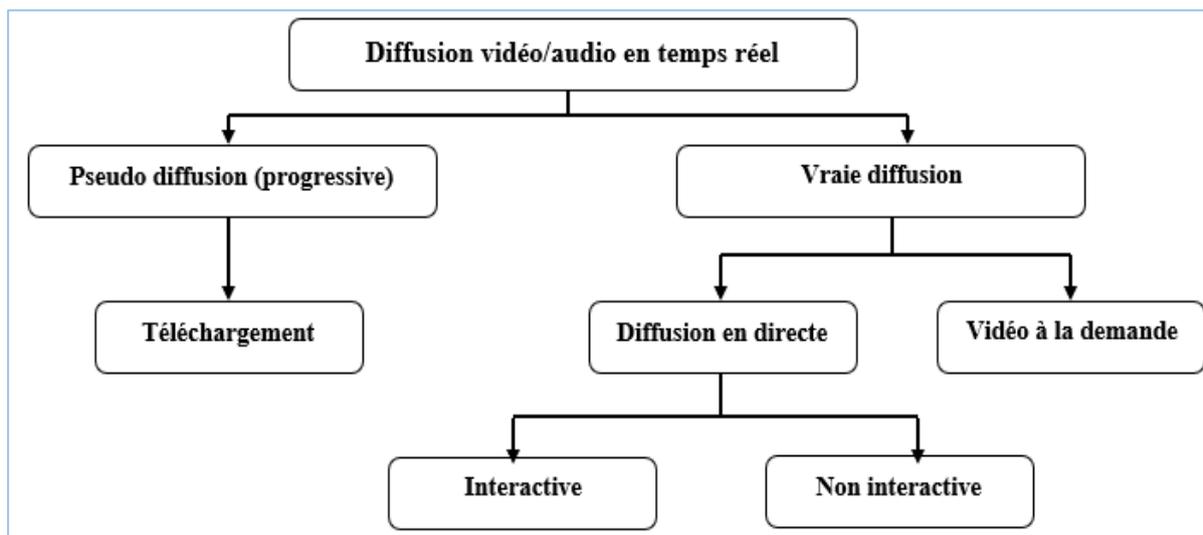


Figure 1. Différents types de diffusion en temps réel [3].

D'une part, le streaming progressif oblige l'utilisateur de télécharger le flux (par exemple, un film). La reproduction commence déjà pendant le téléchargement, mais elle n'offre pas la possibilité de gérer le flux, c'est-à-dire d'avancer ou de rembobiner le flux. D'autre part, le streaming en direct présente la vraie diffusion en temps réel, qui peut être réalisée à la demande ou en directe (live) :

- **La diffusion à la demande** : ou VoD (Video on Demand) : elle est utilisée pour diffuser des productions vidéo ou audio précédemment enregistrés ; par exemple, les chaînes de télévision fournissent des diffusions sélectionnées comme les actualités sur leurs pages web [3].
- **La diffusion en directe** : elle permet de diffuser en direct un programme d'une chaîne radio, un cours à l'université ou une conférence audio-visuelle [4]. On distingue deux types de diffusion en directe : interactive et non-interactive. Dans la diffusion interactive, la communication est bidirectionnelle afin que les utilisateurs puissent intervenir et communiquer en utilisant le module de chat ou des appels vidéo. Cette interactivité crée un lien plus fort entre les utilisateurs.

Dans ce qui suit, nous présentons les différents modes de diffusion de flux multimédia.

## 1.1 Mode de diffusion

La diffusion de flux multimédia se base principalement sur deux modes de transmission [5] :

- **L'unicast** : ce premier mode nécessite la duplication d'un flux vidéo pour chaque récepteur ; il est généralement utilisé pour des services VoD (Video on Demand) ou LoD (Live on Demand).
- **Le multicast** : un flux est transmis pour un groupe de récepteurs. Il est utilisé pour la diffusion en continu de flux temps réel, ainsi que pour les services de communication de groupe comme la vidéo-conférence, le webinaire, etc.

Le multicast de flux multimédia est garanti par des outils spéciaux qui s'appellent Unité de contrôle multipoint ou MCU (*Multipoint Control Unit*). C'est un logiciel informatique ou une machine servant à établir simultanément plusieurs communications (figure 2). Cette fonction permet à plusieurs « participants » de la session de se réunir dans une même « *salle virtuelle* » de discussion [6].

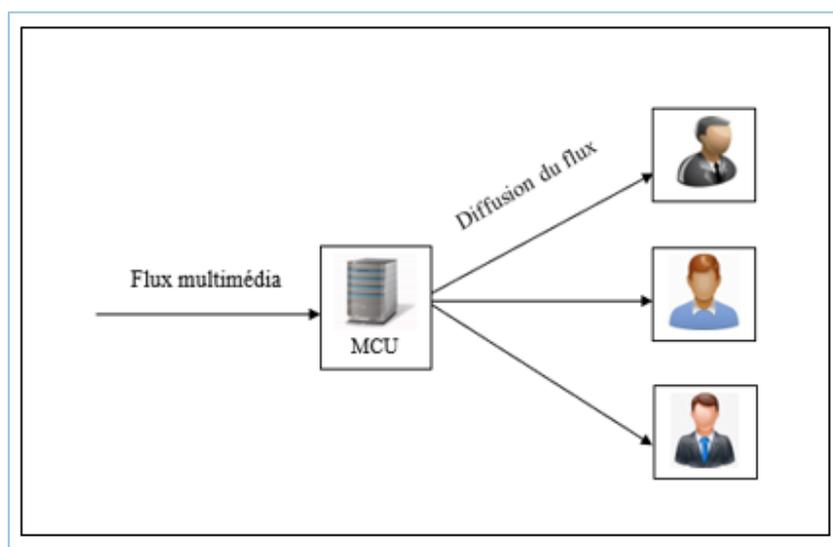


Figure 2. Unité de contrôle multipoint.

Un MCU peut faire face à différentes résolutions des codecs, des taux de trame au sein d'une session vidéo, gérer le transcodage<sup>1</sup>, et faire des redirections de flux sélective.

## 1.2 Protocoles multimédias

Le transfert de données sur Internet s'effectue par paquets de données. Cette structure repose sur l'utilisation de protocoles TCP/IP (Transport Control Protocol/Internet Protocol). De plus, les technologies de diffusion vidéo utilisées sur le réseau peuvent être classées en fonction de leur protocole de transport. La figure 3 récapitule l'ensemble de protocoles que nous allons citer.

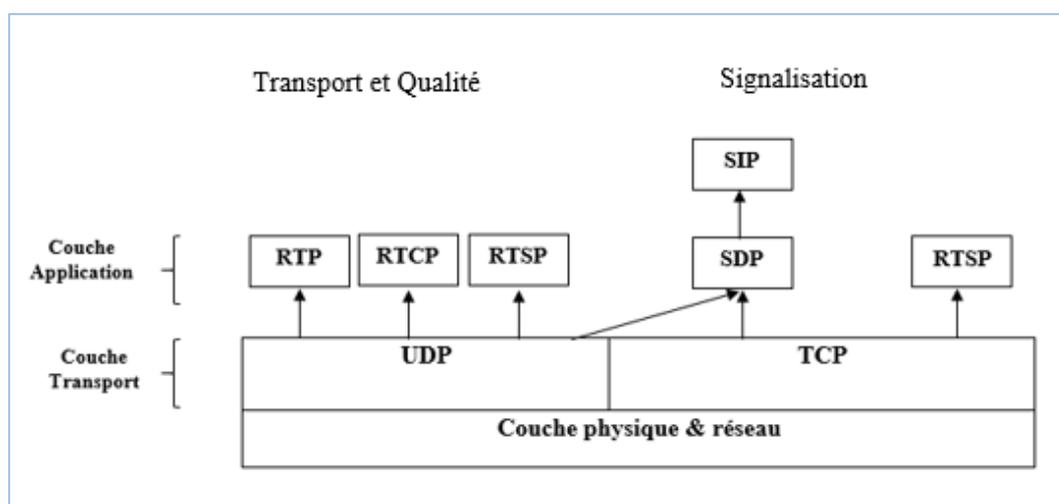


Figure 3. Pile de protocoles [8].

### 1.2.1 TCP

TCP (Transport Control Protocol) [9] est un protocole fiable en mode connecté qui livre un flux de données dans le même ordre que celui dans lequel il a été envoyé, après avoir d'abord établi une session de communication.

De nos jours, la diffusion vidéo sur Internet est habituellement basée sur TCP en raison de sa fiabilité. Bien que TCP soit actuellement le protocole de transport le plus utilisé dans

---

<sup>1</sup> Le transcodage : en vidéo ou en audio, est le fait de changer le format de codage d'un média pour compresser ou encapsuler un média audio ou vidéo dans un fichier et le transporter.

Internet ; il est généralement considéré comme inapproprié pour la diffusion multimédia pour les raisons suivantes :

- Le mécanisme d'accusé de réception du TCP qui ne se balance pas bien lorsque le nombre des destinations augmente pour le même serveur.
- Le mécanisme de retransmission qui peut entraîner des retards de transmission indésirables et peut nuire aux exigences de temps strictes pour les flux vidéo diffusés en direct.

### 1.2.2 UDP

UDP (User Data Protocol) [10] est un simple protocole de transmission sans connexion, sans dialogue et n'offre pas de garantie de livraison, de commande ou de retransmission. Il est utilisé comme protocole de transport par un grand nombre de services de diffusion vidéo.

Ces caractéristiques simplifient la transmission de données et réduisent le délai de bout en bout. Par conséquent, UDP est adapté aux applications temps réel qui sont sensibles aux retards (diffusion vidéo en direct, etc.). Il est largement utilisé avec les protocoles de couche d'application RTP et RTCP.

### 1.2.3 RTP

RTP (Real-Time Transfer Protocol) [11] permet la transmission de données en temps réel. Il fournit très peu de correction d'erreurs : les paquets perdus, en retard ou endommagés sont ignorés. En effet, il privilégie l'enchaînement de son et des images, plutôt que l'intégrité des données.

Cependant, si la vitesse de connexion est inférieure au débit de données, la lecture est « saccadée », voire impossible. De même, si le taux d'erreur est supérieur à 10, la lecture est de mauvaise qualité (le son peut par exemple être métallique).

Le but de RTP est de fournir un moyen uniforme de transmettre des données soumises à des contraintes temps réel. Son rôle principal est de mettre en œuvre des numéros de séquence de paquets IP pour reconstituer les infos de voix ou vidéo.

## 1.2.4 RTCP

RTCP « Real time control Protocol » [11] est basé sur la transmission périodique de paquets de contrôle à tous les participants d'une session. Ce protocole de contrôle des flux RTP permet de véhiculer des informations basiques sur les participants d'une session, et sur la qualité de service. Le multiplexage des paquets de données RTP et des paquets de contrôle RTCP est réalisé par le protocole sous-jacent (UDP, par exemple).

## 1.2.5 SIP

Plusieurs protocoles de signalisation existent ; le plus utilisé dans l'Internet est le protocole SIP (Session Initiation Protocol). Ce dernier utilise SDP pour décrire les détails de l'appel, sous un modèle d'offre-réponse.

SIP [12] est un protocole TCP/IP de la couche application. Il a été conçu pour établir, modifier et terminer des sessions multimédia. Il prend en charge l'authentification et la localisation de multiples participants. S'il se charge de la négociation des médias, il laisse le soin à d'autres protocoles de transporter du texte, de la voix ou de la vidéo.

## 1.2.6 SDP

SDP (Session Description Protocol) [13] est un protocole de description, conçu à l'origine pour être utilisé par des applications de diffusion. Son usage s'est généralisé avec le protocole de signalisation SIP ; il est utilisé aussi par le WebRTC au niveau du serveur de signalisation.

SDP permet de décrire les propriétés des sessions multimédias, les différents types de médias supportés et les propriétés d'un point de communication : son adresse IP, le port d'écoute, les codecs à utiliser ainsi que les différents candidats-relais (adresse IP, protocole, port, etc.) pour établir la communication.

Ces protocoles sont utilisés par différents services de communication comme le webinaire, que nous allons présenter dans la section suivante.

## 2. Webinaire

Par définition, la communication est le fait d'établir une relation ou une liaison avec autrui et de leur transférer, partager et diffuser des informations à travers la linguistique. Elle représente le pilier des séminaires soit de façon classique (c'est-à-dire, des séminaires dans des salles physiques) ou bien séminaires virtuels sur internet [14].

Le webinaire ou sa version anglophone « *webinar* » [15] est la combinaison des mots web et séminaire, ce qui signifie simplement un séminaire virtuel sur internet. C'est un service d'apprentissage en ligne visant à transmettre des connaissances et des compétences à un public cible. Il offre un mélange de l'audio, de la vidéo, du texte et d'autres formes d'interactions en directe. Il est organisé autour de stratégie d'apprentissage spécifique, identifiée à l'avance, et qui constitue la base du contenu de webinaire pertinent.

Les webinaires ont été principalement utilisés par les entreprises, et récemment introduits dans le domaine de l'éducation visant à apporter une révolution dans ce secteur.

Dans ce qui suit, nous présentons les éléments principaux de ce service, ainsi leurs différentes fonctionnalités.

### 2.1 Éléments de systèmes webinaires

Un webinaire est composé de plusieurs entités indispensables [16], comme montré par la figure 4. Ses principaux éléments sont :

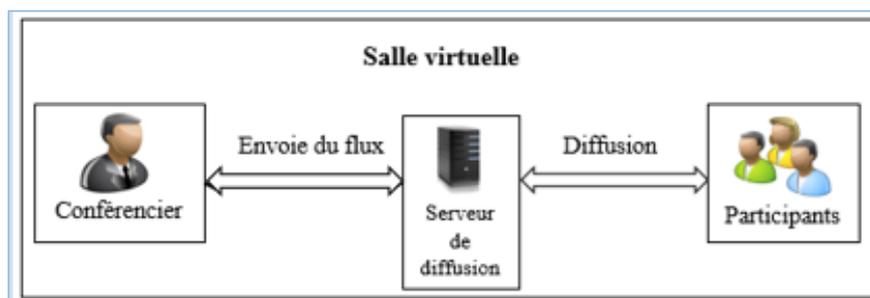


Figure 4. Architecture générale des systèmes webinaire

- **Conférencier** : c'est la personne qui ordonne et gère la réunion notamment en attribuant les droits de parole et d'intervention des participants. Il peut aussi créer

des salles privées, où un nombre restreint de participants se retrouve en toute confidentialité.

- **Participants** : ils sont conviés à la réunion par le conférencier ; ils ont deux manipulations à effectuer pour participer : se connecter à un site internet et rejoindre la salle de réunion virtuelle. En fonction des droits accordés par le conférencier, ils seront simples spectateurs, ou pourront prendre la parole, voire modifier des documents.
- **Salle de conférence virtuelle** : la quasi-totalité des systèmes de webinaire propose des salles de réunion virtuelles sur leur site. Ne peuvent y accéder que les personnes possédant un droit d'entrée, exceptées pour les réunions ouvertes. L'organisateur dispose, dans cette salle d'un tableau de commande de la réunion. En outre, les participants ont accès à différents outils (demande de prise de parole, bulletin de vote, accès aux documents, etc.).

## 2.2 Fonctionnalités de système webinaire

La caractéristique clé d'un webinaire est la capacité de discuter et partager des informations ; par exemple « *Amener des experts à interagir avec les étudiants à travers cette plateforme collaborative virtuelle* ». Les webinaires offrent des fonctionnalités qui sont citées ci-dessous [17] :

- Interactivité accrue, capacité de discuter et de partager des informations en directe.
- Diffusion live de vidéo.
- Interaction avec le présentateur (conférencier). Les participants à la session peuvent lui poser des questions pendant ou après la session.
- Réponse aux questions des participants pendant ou après la fin de la session.

## 3. Étude des systèmes webinaraires existants

Le webinaire associe généralement des outils permettant l'interaction distante et le partage des fichiers entre le conférencier et ceux qui participent à la session.

Cette section a pour but d'étudier des outils de webinaraires existants, ainsi que leurs architectures.

### 3.1 Skype

C'est un logiciel de vidéo-conférence qui permet aux utilisateurs de passer des appels vidéo. D'autres services sont offerts aussi comme : la messagerie instantanée, le webinaire, etc. [18].

Avec Skype, commencer une session webinaire revient à faire un appel vidéo entre le conférencier et les participants, qui doivent avoir au préalable un compte Skype. Pour chaque session webinaire, le conférencier doit inviter les participants en leur envoyant une invitation par email avec l'objet de la session ainsi que la date de commencement. Un lien « *participer à une réunion* » est généré et envoyé à tous les participants. En cliquant sur ce lien, ils peuvent se joindre à la session après avoir été lancé par le conférencier.

Skype est un mélange de VoIP et pair-à-pair et s'appuie précisément sur l'architecture hybride comme le montre la figure 5 [19].

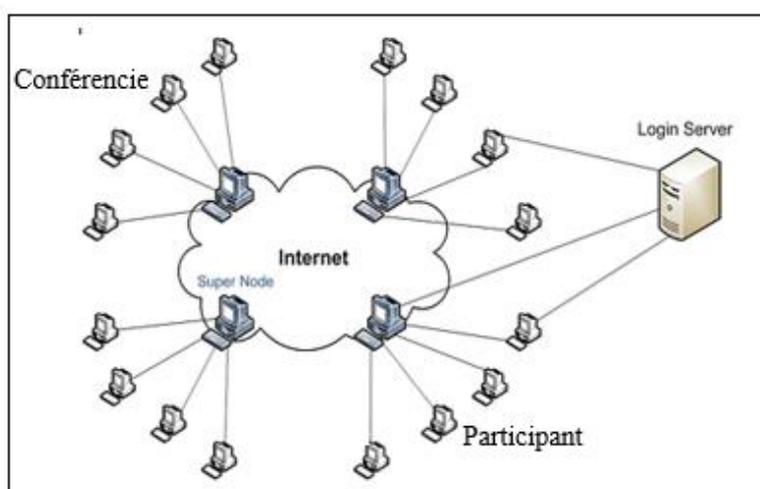


Figure 5. Architecture du Skype [20]

Il existe 3 types d'éléments au sein du réseau Skype : le serveur de login, les super-nœuds et les nœuds ordinaires [19] :

- **Serveur de login** : c'est la seule entité centralisée du protocole Skype, où tous les clients doivent s'authentifier pour s'assurer que les identifiants soient uniques sur le réseau.

- **Super nœud** : c'est le point d'arrivée d'un nœud ordinaire sur le réseau Skype. Tout nœud avec une adresse IP publique ayant suffisamment de CPU, de mémoire et de bande passante réseau est un candidat à devenir un super nœud.
- **Nœud ordinaire (participant ou conférencier)** : c'est une application qui peut être utilisée pour passer des appels vocaux et envoyer des messages texte. Un hôte ordinaire doit se connecter à un super nœud et doit s'enregistrer avec le serveur de connexion Skype pour une connexion réussie.

Skype utilise généralement UDP pour transmettre l'audio et la vidéo ; par contre, si UDP est bloqué, il bascule vers TCP.

### 3.2 Hangouts Google+

Google Hangouts [20] est un service de vidéo-conférence multi-participants, et de messagerie instantanée, intégré au réseau social Google+. Le webinaire fait partie des services fournis par Hangouts Google+.

Avec Hangouts Google+, commencer une session webinaire revient à faire un appel vidéo entre le conférencier et les participants, qui doivent avoir en premier un compte *Gmail*. Pour chaque session webinaire, le conférencier doit inviter les participants en leur envoyant une invitation par email avec l'objet de la session ainsi que la date de commencement ; ainsi, un lien sera généré et envoyé à tous les participants. Ces derniers peuvent refuser ou accepter l'invitation reçue par email. En cliquant sur ce lien, les participants rejoignent la session après qu'elle soit lancée par le conférencier.

Selon la figure 6, l'architecture de Google+ est déployée sur [20] :

- Une topologie centralisée sur les serveurs, qui sert à gérer et à diffuser les appels vidéo entre plusieurs utilisateurs.
- Un utilisateur qui envoie sa voix et sa vidéo à un seul serveur proxy dédié et reçoit également la voix et la vidéo d'autres utilisateurs à partir de ce serveur.

Chaque utilisateur ouvre quatre connexions avec son serveur proxy sur le même port de serveur (Port 19305). La plupart du temps, ces quatre connexions utilisent l'UDP. Le TCP est utilisé seulement lorsque le trafic UDP est délibérément bloqué [20].

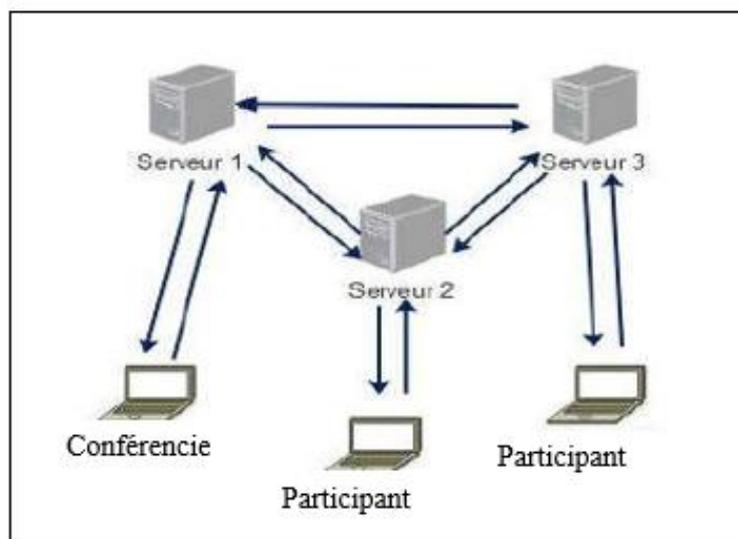


Figure 6. Architecture de Hangouts [20].

Google+ utilise également le protocole RTP pour transmettre la voix et la vidéo. Les deux autres flux sont conformes au format du protocole RTCP [20].

### 3.3 WebRTC

Communication en temps réel via le web (Web Real-Time Communication) [21], est un standard du W3C, comme leur nom l'indique, qui offre des communications audio et vidéo en temps réel. Il est dédié spécialement au service webinaire.

Avec WebRTC, l'organisation des sessions webinaraires se déroule comme suit : D'abord, chaque utilisateur doit avoir un compte. Le conférencier doit créer préalablement une session, en entrant son objet et sa date de déroulement ; il peut ensuite inviter des participants à la rejoindre. De leurs parts, les participants reçoivent un lien d'invitation ; en cliquant dessus, ils peuvent rejoindre la session après qu'elle soit lancée par le conférencier.

L'étape initiale d'établissement de cette communication requiert de la coordination fournie par le serveur de signalisation. La signalisation permet à deux ou plusieurs navigateurs Web disposant de la capacité WebRTC de se joindre, d'échanger les informations de contact, et de négocier une session qui définit la manière dont ils vont

communiquer. Finalement, l'établissement des canaux media pair-à-pair pour le transport des flux media échangés directement entre eux.

La signalisation est le mécanisme qui gère des ressources nécessaires pour l'établissement d'un canal de communication et permet d'envoyer et recevoir des messages de contrôle via le protocole SIP ou SDP.

Pour que deux clients puissent communiquer, ils doivent se mettre d'accord sur le type de media à échanger et un canal commun. Si le « *client A* » souhaite communiquer avec le « *client B* », alors ce dernier doit fournir une « *adresse* » à laquelle le « *client A* » peut le joindre, comme le montre la figure ci-dessous :

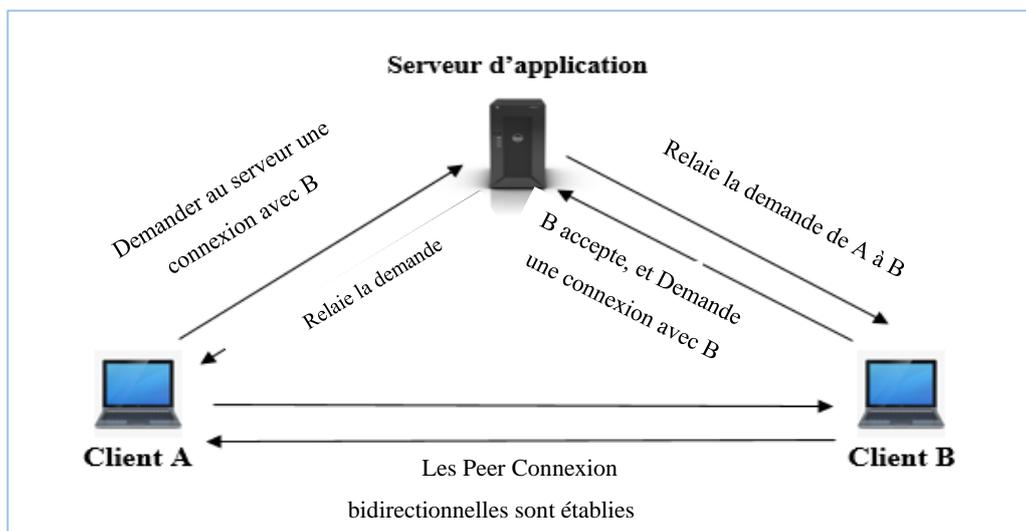


Figure 7. Architecture de WebRTC [21].

### 3.4 UnivWebinar

*UnivWebinar* [1], comme son nom l'indique, est un service de webinaire dédié au secteur de l'enseignement supérieur. Il permet aux utilisateurs de se réunir dans un environnement d'apprentissage collaboratif.

Les utilisateurs doivent avoir un compte pour bénéficier des services offerts par ce système. On peut avoir deux types d'utilisateur : conférencier et participant. Le conférencier s'occupe de la création des sessions webinaires en indiquant son objet, sa date et heure de diffusion et les affiche dans une table. Les participants peuvent facilement consulter puis rejoindre une session, en cliquant sur celle choisie.

*UnivWebinar* se déploie sur une architecture réseau hybride. Le réseau est constitué des éléments suivants (figure 8) :

- Serveur de signalisation où tous les éléments du système doivent passer avant d'effectuer une communication entre eux.
- Nœuds qui prennent en charge la diffusion de flux multimédia.
- Utilisateurs.

*UnivWebinar* utilise le protocole TCP pour se connecter au serveur de signalisation et le protocole UDP pour la diffusion de flux multimédia.

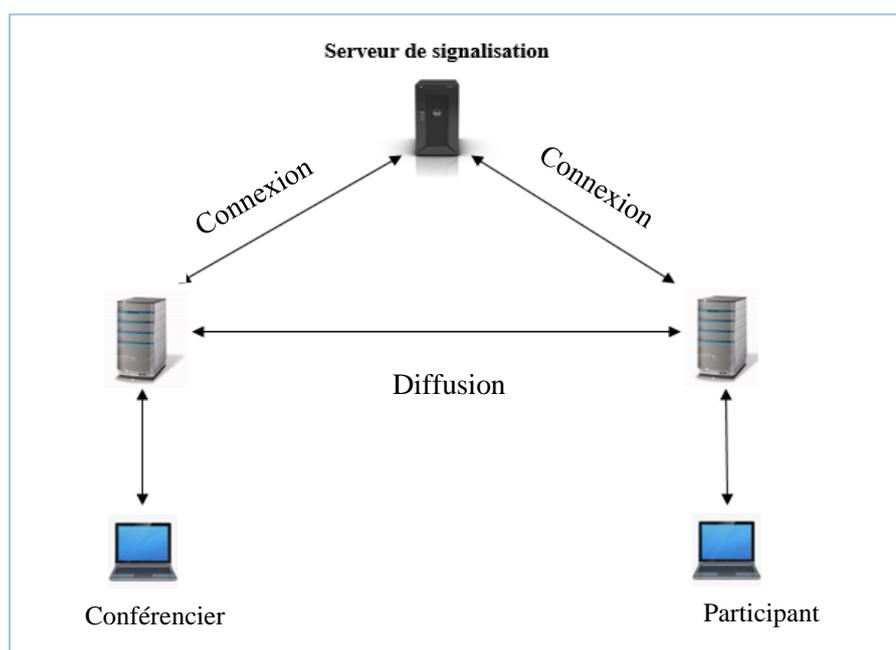


Figure 8. Architecture de UnivWebinar [1].

### 3.5 Étude comparative

Dans cette section, nous comparons les différents outils décrits dans les sections précédentes. Le tableau 1 récapitule d'une façon générale leurs principales caractéristiques, ainsi que les avantages et inconvénients de leurs architectures.

À partir de ce tableau, on peut remarquer que le système « *UnivWebinar* » a des points forts et des points faibles par rapport aux autres architectures.

| <b>Outils</b>                           | <b>Skype</b>  | <b>Google Hangout+</b>  | <b>WebRTC</b>  | <b>UnivWebinar</b>   |
|---|---|---|--|--|
| <b>Services offerts</b>                 | Messagerie instantanée<br>VOIP <sup>2</sup><br>Vidéo-conférence<br>Webinaire  | Messagerie instantanée<br>VOIP<br>Vidéo-conférence<br>Webinaire   | Webinaire seulement  | Webinaire seulement  |
| <b>Interactivité (Oui/Non)</b>          | Oui   | Oui   | Oui  | Oui  |
| <b>Session programmée par</b>           | Invitation  | Invitation  | Invitation   | /  |
| <b>Nombre de participants supportés</b> | 25  | 25  | /  | Non limité   |
| <b>Domaines d'utilisations</b>          | Générale  | Générale  | Générale   | Éducation supérieur  |
| <b>Avantages</b>                        | Bénéficie d'une meilleure bande passante.<br>N'utilise pas les trames de broadcast.<br>Le trafic de requête est réduit. | Offre une vue d'ensemble très complète du réseau.   | Gestion de la bande passante.  | Réduction très importante des besoins en bande passante (MCU). |
| <b>Inconvénients</b>                    | Risque de défaillance d'un Super-nœud.<br>Le serveur login risque de s'arrêter, donc perte de données                   | Le risque de surcharge sur le serveur central (gestion et diffusion) qui peut devenir un point de défaillance du système. | Un seul serveur centralisé, risque de s'arrêter donc perte de connexion. | Risque de défaillance du serveur de signalisation.             |

Tableau 1. Comparaison des systèmes webinaires

---

<sup>2</sup>VOIP : est une technique qui permet de communiquer par voix à distance via le réseau Internet.

## **4. Conclusion**

Dans ce chapitre, nous avons défini le champ de notre étude. Nous avons aussi présenté les différents systèmes webinaire, suivi d'une étude critique afin de réaliser notre système.

Dans le chapitre suivant, nous présenterons la notion de la qualité d'expérience et de service, dont nous allons se baser afin de proposer la solution proposée pour le routage de flux multimédia dans les systèmes de webinaire.

## Chapitre 2

### Qualité de Service Vs. Qualité d'Expérience

#### 1. Introduction

La notion de Qualité de l'Expérience (Quality of Experience, QoE) a vu le jour au cours de la dernière décennie avec l'intention d'impliquer la perception subjective des utilisateurs finaux dans l'évaluation de la Qualité de Services (Quality of Service, QoS) de télécommunication. Comme cela est connu de tous, les utilisateurs finaux définissent la réussite ou l'échec d'un service. Donc, les fournisseurs des services doivent satisfaire les besoins de leurs clients en offrant les services avec haute performance afin de prévenir l'utilisateur [22].

Par ailleurs, les critères d'évaluation de la performance du service des fournisseurs diffèrent de ceux des utilisateurs. Les premiers font souvent appel aux paramètres spécifiques de la QoS. En revanche, les utilisateurs perçoivent, quant à eux, la performance du service de manière plutôt subjective que technique. C'est pourquoi, il est nécessaire de connaître l'impact de la performance du réseau sur le ressenti de l'utilisateur. Malheureusement, il est très difficile d'évaluer cet impact en temps réel sur un réseau opérationnel alors que le contrôle des paramètres de la QoS est plus facile et plus habituel. La solution idéale pour les fournisseurs de service est d'identifier la relation entre les paramètres de la QoS et l'influence individuelle et collective de ces derniers sur la QoE. En connaissant la QoE de l'utilisateur final, les fournisseurs peuvent en effet réguler les paramètres techniques du réseau afin de répondre au mieux aux attentes des clients [22].

Dans ce chapitre, nous allons définir la notion de QoS dans les réseaux IP ainsi que ses paramètres. Ensuite, nous présentons la notion de la QoE. Enfin, nous discuterons la relation entre ces deux notions.

## 2. Qualité de Service

Internet est un réseau basé sur la commutation de paquets et sur le protocole IP (Internet Protocol) qui implémente le modèle BE (Best Effort) : tous les paquets de données suivent la même politique et leur acheminement est effectué de proche en proche. Ce modèle ne permet pas de répondre aux exigences de certaines applications (par exemple, les applications multimédia), en matière de temps de réponse, de bande passante, du taux de perte, etc. Ces paramètres indiquant la performance du réseau, sont connus souvent sous le terme de la QoS, l'objet de cette section.

Tout d'abord, nous commençons par définir ce que signifie la QoS. Ensuite, nous présentons ses principaux paramètres en discutant la relation possible entre eux. Enfin, nous décrivons les différentes techniques de mesure des paramètres de la QoS.

### 2.1 Définition

Les standards de ITU (International Telecommunication Union) et ISO (International Standardization Organization) définissent la QoS comme suit : « *la QoS est un ensemble d'exigences de qualité sur le comportement collectif d'un ou de plusieurs objets* » [23].

Selon IETF (Internet Engineering Task Force), la qualité de service désigne « *la manière dont le service de livraison de paquets est fourni et qui est décrite par des paramètres tels que la bande passante, le délai de paquet et les taux de perte de paquets* » [24].

### 2.2 Paramètres de la qualité de service

Les principaux paramètres qui reflètent l'état du réseau, et permettant de mesurer la qualité de service sont cités en ordre alphabétique :

- **Bande passante** : le débit ou bande passante (bandwidth) est le volume de données maximum qui peut être maintenu entre la source et la destination. La bande passante utilisée par un service vidéo peut varier considérablement dans le temps. La gestion de la bande passante est liée principalement au choix du codec vidéo à utiliser pour un service donné [25].

- **Délai de bout-en-bout** : le délai définit le temps que parcourt un paquet entre son émission et sa réception. Il prend en compte principalement deux phases : (i) le temps que passe un paquet dans la file d'attente des nœuds le long du parcours et (ii) le temps de propagation du paquet sur le médium physique. Les applications temps réel comme vidéo-conférence ou voix sur IP, sont très sensibles à ce facteur [25].
- **Gigue** : c'est la variation du délai, ce qui provoque des différents temps d'arrivée à la destination. Elle est causée par la multiplicité des chemins pour la même destination, et par les différents algorithmes utilisés en file d'attente. Si la valeur de la gigue est importante, cela conduit à une dégradation de la qualité des applications interactives en temps réel [26].
- **Perte de paquets** : la perte des paquets est causée par la congestion du réseau ou l'arrivée tardive des paquets. En effet, après un certain délai, le paquet est jugé comme perdu même s'il arrive dans les instants suivants [26].

Ces paramètres peuvent être calculés selon plusieurs normes ; la plus connue est la norme UIT, c'est ce que nous allons voir dans la sous-section suivante.

### 2.2.1 Calcul des paramètres de QoS

De manière générale, UIT [27] adopte des définitions statistiques pour la QoS associée au transfert de paquets IP. Elle est exprimée en fonction de six paramètres : délai (IPTD), gigue (IPDV), taux d'erreur (IPER), taux de perte (IPLR), taux des paquets faux (SIPR) et débit (IPPT). Le tableau 3 récapitule la méthode de calcul de ces paramètres :

| Paramètre  | Définition   | Équation  |
|--|--|---|
| Le délai de transfert du paquet IP (IP Packet Transfer, IPTD)        | La différence entre le moment de réception et celui d'émission du paquet et doit être inférieure à $T_{max}$ .   | $IPTD = t_{réception} - t_{émission}$ où<br>$t_{réception} > t_{émission}$ $t_{réception} - t_{émission} < T_{max}$ |
| La variation du délai du paquet IP (IP Packet Delay variation, IPDV) | La différence entre son IPTD et un IPTD de référence. L'IPTD de référence peut être, par exemple, l'IPTD moyen ou l'IPTD du premier paquet                                       | $IPDV = IPTD - IPTD_{référence}$  |
| Le taux de perte des paquets IP (IP Packet Loss Ratio, IPLR)         | Le nombre total des paquets perdus ( $N_{perdus}$ ) par rapport au nombre total des paquets transmis ( $N_{transmis}$ )  | $IPLR = N_{perdus} / N_{transmis}$  |
| Taux d'erreur des paquets IP (IP Packet Error Ratio, IPER)           | Le nombre total des paquets en erreur ( $N_{en\_erreur}$ ) par rapport au nombre total de paquets reçus avec succès ( $N_{avec\_succès}$ ) plus le nombre des paquets en erreur. | $IPER = N_{en\_erreur} / (N_{avec\_succès} + N_{en\_erreur})$   |
| Le débit des paquets IP (IP Packet Throughput, IPPT)                 | Le nombre de paquets IP transférés avec succès dans un intervalle de temps $\Delta T$ .  | $IPPT = N_{succès} / \Delta T$  |

Tableau 2. Calcul des paramètres de la qualité de service selon la norme UIT [27].

D'après [28] et [29], le service de vidéo-conférence est sensible aux délais, à la gigue, et à la perte de paquet. Dans notre travail, nous allons nous baser sur ces trois paramètres de QoS pour gérer la gestion de la diffusion des flux multimédia dans le cadre de services webinaires qui sont très similaires au service vidéo-conférence.

### 2.2.2 Relation entre les paramètres de qualité de service

Le délai, la gigue, le débit et la perte de paquets, ne sont néanmoins pas indépendants ; et dans tout type de réseau, il existe une relation entre eux. Fixer un de ces paramètres crée une dépendance entre les trois autres. Ainsi pour :

- Un taux de perte fixe, on aura une augmentation de débit suivi d'une réduction de délai et de gigue.

- Un débit fixe, résulte une baisse du délai et de la gigue qui conduit à une décroissance du taux de perte.
- Un délai fixe, la gigue est aussi fixe, implique une diminution du taux de perte et du débit.

Ces relations donnent aussi des informations pour la gestion à long terme du réseau, car elles montrent quand les exigences par rapport à la perte et au délai ne peuvent pas être satisfaites. À présent, la gestion des réseaux se fait très souvent en fonction seulement des exigences de débit, et parfois sa relation avec le délai et la gigue. Mais considérer le taux de perte et son contrôle comme des parties importantes du processus de décision permet une gestion plus flexible et plus efficace [30].

### 2.3 Techniques de mesure de la qualité de service

Afin de permettre une bonne gestion de la QoS des réseaux IP, il faut d'abord pouvoir mesurer ou évaluer cette notion à travers ses paramètres. Pour ce faire, différentes techniques de mesure ont été conçues et développées dans la littérature. En générale, ces techniques peuvent être classées en trois types : passives, actives et hybrides. Dans cette partie, nous allons présenter ces principales techniques de mesure.

- **Techniques passives** : le principe des mesures passives est de surveiller le trafic et d'étudier ses propriétés en un ou plusieurs points du réseau. Ainsi, ces techniques ne sont absolument pas intrusives et ne changent rien à l'état du réseau. De plus, elles permettent des analyses très avancées. En revanche, elles génèrent des statistiques significatives lors d'une analyse de très longues périodes ce qui nécessitent des ressources mémoires importantes [31].
- **Techniques actives** : le principe des mesures actives consiste à générer d'abord du trafic dans le réseau, ensuite à observer les effets des composants et protocoles sur le trafic généré. Ce type de technique reste le seul moyen pour un utilisateur de mesurer les paramètres du service dont il pourra bénéficier. En revanche, l'inconvénient majeur de cette approche est la perturbation introduite par le trafic de mesure qui peut faire évoluer l'état du réseau et ainsi fausser la mesure. De nombreux travaux menés actuellement abordent ce problème en essayant de trouver les profils de trafics

de mesures qui minimisent les effets du trafic supplémentaire sur l'état du réseau [31].

- **Techniques hybrides** : Afin de pallier aux inconvénients des techniques précédentes et de combiner leurs avantages, des techniques hybrides sont apparues (tableau 4). Le principe de ces techniques consiste à analyser les paquets et donner des résultats de mesure. Ces paquets peuvent être générés (comme dans les techniques actives) ou peuvent faire partie des paquets ou les flux transférés par le service (comme dans les techniques passives). La différence réside sur la question comment faire l'analyse.

|                      | Mesures Actives   | Mesures Passives  |
|----------------------|---|---|
| <b>Principe</b>      | <ul style="list-style-type: none"> <li>• Générer le trafic dans un réseau et l'observer.</li> </ul>   | <ul style="list-style-type: none"> <li>• Analyser le trafic et étudier ses propriétés en un ou plusieurs points.</li> </ul>   |
| <b>Avantages</b>     | <ul style="list-style-type: none"> <li>• Permettre la mesure directe des paramètres de QoS principalement le délai, le taux de pertes et la gigue.</li> </ul> | <ul style="list-style-type: none"> <li>• Non intrusives.</li> <li>• Permettre une mesure directe des paramètres utilisés dans l'ingénierie des réseaux.</li> </ul>  |
| <b>Inconvénients</b> | <ul style="list-style-type: none"> <li>• Intrusives : le trafic de mesure peut dans certains cas. Fausser les mesures elles mêmes.</li> </ul>                 | <ul style="list-style-type: none"> <li>• Elles sont locales (relatives à un lien) et il est difficile de les étendre à la globalité du réseau.</li> <li>• Elles ne permettent pas la mesure directe des paramètres de QoS.</li> <li>• Elles nécessitent des ressources disque et mémoire vive importantes.</li> </ul> |

Tableau 3. Tableau comparatif entre les techniques actives et les techniques passives.

### 3. Qualité d'expérience

Récemment, la perception des utilisateurs retient de plus en plus l'attention des fournisseurs de services et des chercheurs. Ils sont intéressés à connaître comment les utilisateurs perçoivent la qualité de services de diffusion de flux multimédia. Pour éviter la fatigue liée aux tests subjectifs, ils sont intéressés par la modélisation de la qualité

d'expérience basée sur la QoS et d'autres paramètres mesurables. Ceci va être discuté dans cette section. Pour cela, nous commençons par une définition de la QoE, ainsi que leurs métriques. Enfin, nous discutons la relation entre la QoS et la QoE.

### 3.1 Définition

Les standards d'ITU (International Telecommunication Union) [32] définissent la QoE comme étant « *l'acceptabilité globale d'une application ou d'un service, telle qu'elle est perçue subjectivement par l'utilisateur final* ». La QoE inclut les effets complets du système de bout en bout (client, terminal, réseau, infrastructure de services, etc.) et peut être influencée par les attentes des utilisateurs et du contexte.

Selon [5], la QoE est le degré de plaisir ou d'agacement d'un utilisateur concernant une application ou un service. Il résulte de l'accomplissement de ses attentes à l'égard de l'utilité et/ou de la jouissance de l'application ou du service à la lumière de sa personnalité et de son état actuel.

### 3.2 Paramètres affectant la QoE

Il y a beaucoup de facteurs qui ont un impact sur la QoE. Ces facteurs dépendent de l'application, de la technologie des réseaux, du terminal et bien sûr de l'utilisateur final. Ils peuvent ainsi être classifiés en quatre catégories comme illustré dans la figure 9.

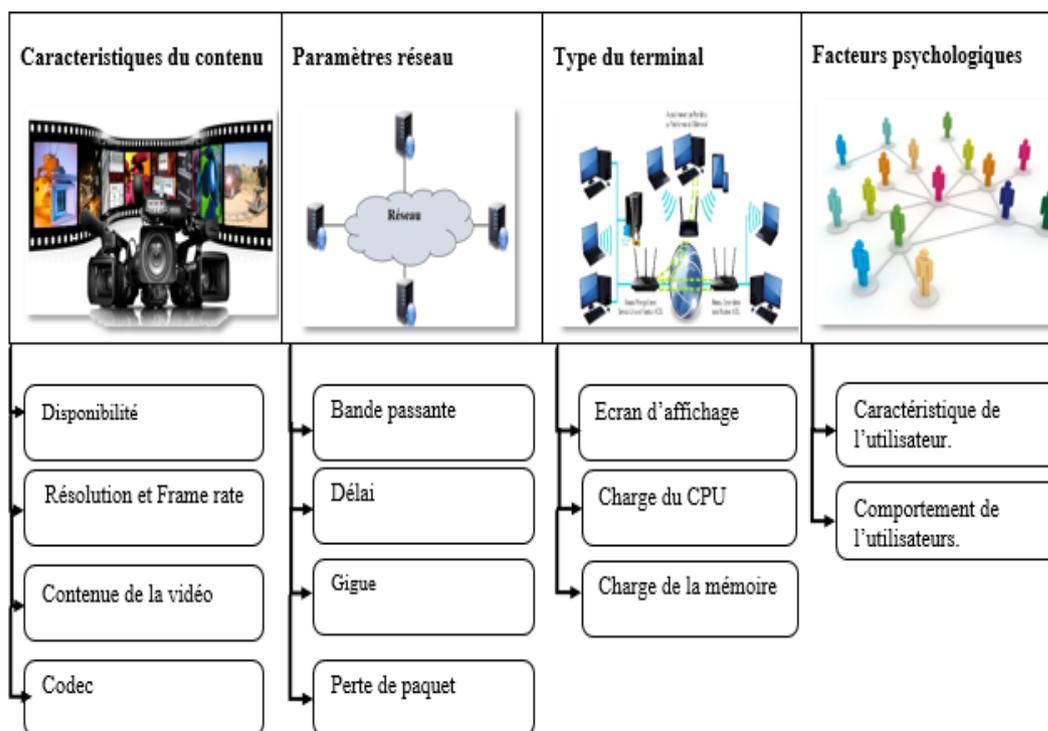


Figure 9. Paramètres affectant la QoE.

### 1) Caractéristique de la vidéo

Les propriétés et les caractéristiques de la vidéo ont un impact considérable sur la QoE. Ci-après quelques paramètres [33] :

- **Disponibilité de la vidéo** : c'est le premier facteur influençant la qualité perçue du service ; il est important d'assurer la disponibilité de la vidéo pour les utilisateurs.
- **Résolution et frame rate** : l'ajustement de la résolution de la vidéo ainsi que le nombre d'images par seconde peut impacter directement la QoE, du moment qu'ils jouent un rôle important dans le séquençement de la vidéo et sa qualité de visualisation.
- **Contenu de la vidéo** : il est important de noter que le thème de la vidéo peut différencier le jugement des utilisateurs, car chaque utilisateur peut avoir un centre d'intérêt différent. De ce fait, le sujet traité dans la vidéo à une importance relative.
- **Codec** : l'envoi d'une vidéo non compressée sur le réseau nécessite de grandes quantités de bande passante et d'espace de stockage. Par conséquent, les vidéos sont compressées avec des algorithmes de compression très efficaces. Un grand nombre

de codecs vidéo sont pris en charge par les services vidéo tels que H.262, H.263, H.264, WMV3 et WVID.

## 2) Type du terminal

Avec les progrès de l'industrie des télécommunications, des services multimédias sont déployés sur différents types de terminaux (Smartphone, PC portable, PC bureau et appareils mobiles). Ces équipements se différencient les uns des autres par leur résolution, leur processeur et leur taille d'affichage. Ci-après quelques paramètres de l'appareil affectant la QoE [33].

- **Écran d'affichage** : deux utilisateurs avec les mêmes conditions de réception peuvent avoir des qualités différentes uniquement en raison de la résolution de leur terminal.
- **Charge du CPU** : elle décrit les ressources de calcul utilisées par le système et l'application et affecte directement la QoE de l'application. En fait, lorsque le CPU est trop chargé, les informations reçues par l'appareil ne peuvent être traitées facilement, et donc, la vidéo sera bloquée avec des images figées.

## 3) Facteurs psychologiques

La QoE est une mesure subjective de l'adéquation d'un service par rapport à ce qu'en attend un client. Elle mesure le « rendu » de l'utilisation d'un service. L'estimation des variations du niveau de qualité d'une personne à l'autre dépend de plusieurs facteurs qui sont liés à la personne, ses préférences et son environnement. Nous pouvons préciser certains de ces paramètres [33] :

- **Caractéristique de l'utilisateur** : le terme « *personnalité* » est utilisé pour décrire les caractéristiques d'une personne qui comptent pour un schéma cohérent de sentiments, de pensées et d'agissements. Ces caractéristiques se différencient d'un utilisateur à un autre selon leurs connaissances, expériences, langue, âges, sexe, etc.
- **Comportement de l'utilisateur** : le terme « *état courant* » est quant à lui utilisé pour signifier les changements temporels ou de situation dans les sentiments, les pensées ou le comportement d'une personne. C'est le degré d'intérêt pour le contenu.

#### 4) Paramètres réseau

La QoS est un élément principal dans la conception du réseau et de la gestion en général. En règle générale, les paramètres QoS comprennent la perte de paquets, le délai, la gigue et les facteurs de bande passante. L'effet de ces paramètres sur la qualité perçue dépend essentiellement du type de l'application multimédia. Par exemple, si un service temps réel est nécessaire, le taux de perte de paquets est le paramètre réseau le plus important. La retransmission et la mise en mémoire tampon jouent un rôle important aussi. S'il y a une interactivité (par exemple, lors d'un appel vidéo), le délai et la gigue ont également un rôle important, en ajoutant l'écho et la perte de synchronisation audio/vidéo. Les paramètres du réseau reflètent l'état du réseau et ont impact direct sur la QoE. La dégradation de chaque paramètre réseau conduit à la dégradation de la qualité de vidéo, ainsi qu'à la satisfaction de l'utilisateur du service [34].

### 3.3 Relation entre la QoS et la QoE

Récemment, la perception des utilisateurs retient de plus en plus l'attention des fournisseurs de services, des opérateurs, des fabricants et des chercheurs. Ils sont intéressés à savoir comment les utilisateurs perçoivent la QoS de diffusion de flux multimédia. Pour éviter la fatigue liée aux tests subjectifs, ils sont intéressés par la modélisation de la QoE basée sur la QoS et d'autres paramètres mesurables. Cependant, ces paramètres permettant d'estimer la QoE peuvent différer selon les services. De ce fait, différentes propositions ont été faites en vue de trouver une relation entre la QoE et la QoS. Parmi ces propositions, on trouve le modèle *Hourglass* de QoE qui explique cette relation à travers l'architecture classique d'Internet.

#### 3.3.1 Modèle Hourglass de QoE

Le modèle *Hourglass* de QoE [35] est inspiré du modèle *Hourglass* internet classique, qui peut être divisé en cinq couches (Figure10) selon le modèle TCP/IP (physique, réseau, transport et application). Il formalise les facteurs affectant la QoE, à partir de la couche réseau à la couche application. Par conséquent, il se compose de quatre niveaux de qualité :

Qualité de service (QoS), Qualité de délivrance (Quality of delivery, QoD), Qualité de présentation (Quality of Presentation, QoP) et Qualité d'expérience (QoE).

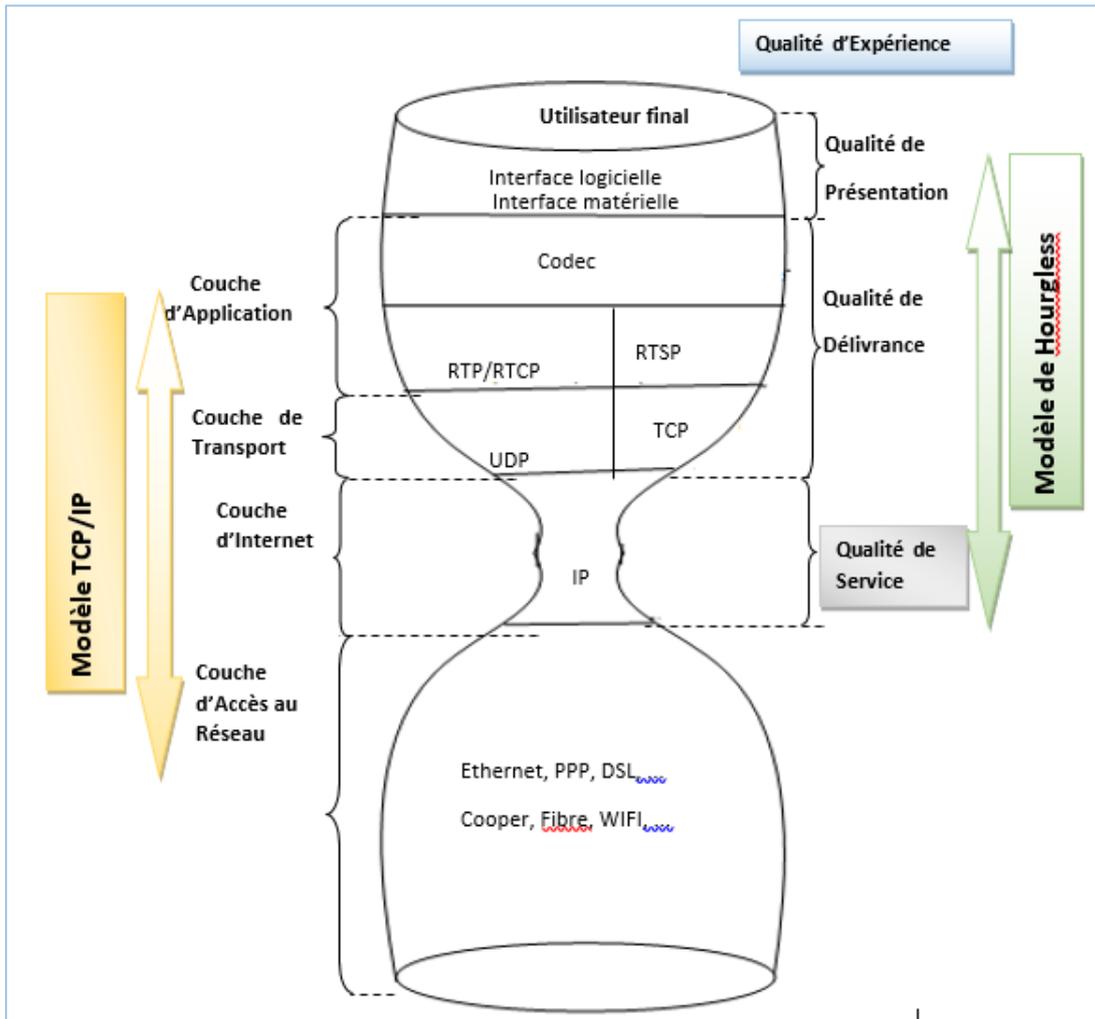


Figure 10. Le modèle *Hourglass* de QoE [38].

Dans le cas d'un service réseau ou d'un flux vidéo, la QoE est une expérience complète de l'utilisateur final qui dépend de la QoP, QoD et de la QoS. Selon le modèle *Hourglass* de la QoE, la QoE dépend de la QoS, où l'utilisateur final a une interaction directe et une expérience des périphériques d'affichage et de l'interface des applications. Tous comportement défectueux ou imprévu d'un appareil ou d'une application peut nuire à la perception de l'utilisateur, même avec d'excellentes performances réseau. Avec la contribution de QoP à la perception de l'utilisateur, cela dépend de la QoD.

### 1) Qualité de service

Comme cité précédemment, la QoS est la capacité à véhiculer un type de trafic donné dans des bonnes conditions, en termes de bande passante (BP), délais de transmission (D), gigue (J) et du taux de perte de paquet (P). Ces mesures changent au fil du temps, mais sont mesurables à des instants particuliers.

$$QoS = f(Bp, D, J, P) \quad (1).$$

### 2) Qualité de délivrance

La QoD mappe la couche de transport, la couche application et partiellement la couche application du modèle de *Hourglass*.

Les données fournies par la couche de transport sont traitées et présentées par une application à l'utilisateur final. Si cette application est livrée avec des procédures de reprise sur erreur, la perte de paquets ou toute erreur récupérable de données ont un effet négligeable sur la qualité. Sinon, ces anomalies peuvent endommager la qualité de manière significative. La QoD peut être divisé en deux parties :

- **QoD du réseau (Network QoD, QoD<sub>N</sub>)** : c'est l'effet cumulatif des paramètres de QoS et du traitement des données par les protocoles de la couche transport.
- **QoD de l'application (Application QoD, QoD<sub>A</sub>)** : c'est le résultat du traitement des données par l'application pour améliorer la performance de bout en bout.

Donc, la QoD est exprimée comme suit :

$$QoD = f(QoD_N, QoD_A) \quad (2).$$

Sachant que :

$$QoD_N = g(QoS) \quad (3).$$

Donc :

$$QoD = f(g(QoS), QoD_A) \quad (4).$$

### a) Qualité de présentation

QoP s'adresse à la deuxième couche du modèle *Hourglass* de la QoE. C'est la couche dans laquelle l'utilisateur final possède avec les dispositifs d'affichage et d'application. Cette couche traite principalement avec le logiciel et le matériel de l'interface et leur facilité d'utilisation ainsi que le contenu transmis. Même avec une excellente qualité d'image, toute présentation médiocre ou toute interface médiocre peut dégrader la QoE.

Donc, la QoP dépend de la QoD et la QoP<sub>o</sub>, où la QoP<sub>o</sub> dépend du logiciel et du matériel de l'interface et la facilité d'utilisation. De plus, la QoD et la QoP<sub>o</sub> sont indépendantes.

La QoP est exprimée comme :

$$QoP = h ( QoD, QoPo ) \quad (5).$$

### b) Qualité d'expérience

Dans le modèle *Hourglass* de QoE, la QoE est le résultat de tous les facteurs de qualité apportés par les couches sous-jacentes. Elle capture l'expérience de l'utilisateur résidant dans la partie supérieure, qui mesure le véritable succès d'un service. L'utilisateur final expérimente la QoE par l'interaction avec l'équipement et l'application, qui est décrit par la QoP où :

$$QoE = k ( QoP ) = k ( h ( QoD, QoPo ) ) \quad (6).$$

La QoP<sub>o</sub> est la contribution de la QoP et ne dépend pas du processus de la livraison de données. Donc, en fixant ce facteur indépendant, l'équation (6) devient :

$$QoE = k ( h ( QoD | QoPo ) = k ( h ( f ( g(QoS), QoDA ) | QoPo ) ) \quad (7)$$

En outre, la QoD<sub>A</sub> dépend de l'application, et elle n'est pas directement liée au processus de livraison de données de réseau. Ainsi, pour une application particulière, elle peut être considérée comme invariante :

$$QoE = k ( h ( f ( g(QoS) | QoDA ) | QoPo ) ) = R QoDA, QoPo (QoS) \quad (8)$$

Ainsi, la QoE peut être modélisée seulement à partir des paramètres de la QoS, avec des facteurs connus ou fixés de la QoDA et la QoPo. Cela signifie que la QoE peut être évaluée objectivement à partir de la QoS sans que cela n'entraîne d'autres facteurs comme le montre la figure suivante :

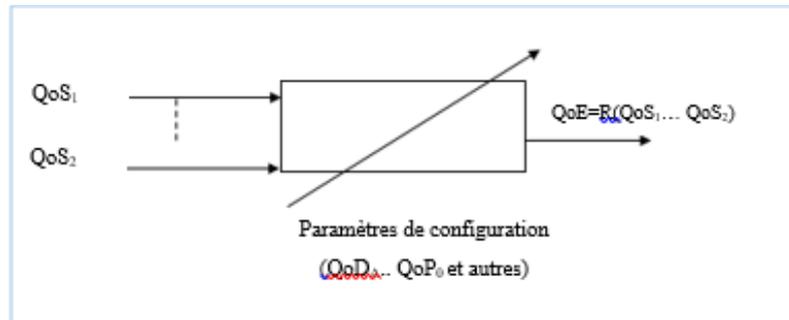


Figure 11. Modèle de corrélation QoS-QoE [35].

Dans ce contexte, plusieurs modèles de corrélation QoE-QoS ont été proposés dans la littérature, comme le modèle de pentagramme et d'autres modèles basés sur l'apprentissage artificiel, et sur les réseaux de neurones [22]. Ces modèles ont pour but de prédire la relation entre la QoS et QoE, et sont dédiés à d'autres services que le webinaire [22]. Dans notre étude, nous avons opté pour le modèle proposé par Kim et al. [28], qui est dédié au service webinaire, avec l'utilisation d'une combinaison de plus d'un paramètre de QoS.

### 3.3.2 Modèle de corrélation QoE-QoS proposé par Kim et al. [28]

D'après Kim et ses collègues [28], l'évaluation subjective de la QoE nécessite un temps et un coût de traitement trop importants, et dans le but d'établir un lien entre les conditions objectives de service du réseau et la perception humaine. Ils ont proposé une méthode pour évaluer la QoE des utilisateurs dans les réseaux convergents<sup>3</sup>.

Kim et al. [28] ont proposé un modèle de corrélation QoE-QoS donnée par la formule (9) dans laquelle la constante  $\alpha$  représente la classe de qualité QoS au niveau du réseau

---

<sup>3</sup>**Réseau convergent** : c'est un réseau qui supporte à la fois les données, la voix et la vidéo et/ou différents protocoles de réseau.

(Tableau 6), le paramètre  $\beta$  est déterminé selon la classe de service (Tableau7), et la constante K est déterminée par le degré de satisfaction des utilisateurs.

L'équation (9) suivante montre comment calculer la QoE :

$$QoE(QoS) = K \left\{ \frac{e^{QoS-\alpha} + e^{-QoS+\alpha}}{e^{QoS-\alpha} + e^{-QoS+\alpha} + \beta} + 1 \right\} \quad (9)$$

Le score normalisé de la QoS du service qui peut être fourni, peut être obtenu par l'équation suivante :

$$QoS = S(D) + S(J) + S(L) + S(E) + S(B) \quad (10)$$

Dans ce modèle, le score (S) de la QoS peut être obtenu en utilisant les indices de qualité au niveau du réseau. Cela devient une application de telle sorte que le délai (D), la gigue (J), le taux de perte (L), le taux d'erreur (E), la bande passante (B), constituent les paramètres de qualité objective et que la valeur pondérée de ces paramètres de qualité soit différente du service utilisé. Après avoir mesuré ces paramètres et défini pour chaque valeur un score, la somme de ces scores décrit la qualité de service, comme le montre le tableau 5.

| Paramètres                       | Plage                   | Score           | Qualité    |            |
|----------------------------------|-------------------------|-----------------|------------|------------|
| Délai (D) (ms)                   | $0 < D < 70$            | 10              | Excellente |            |
|                                  | $70 < D < 100$          | 8               | Bien       |            |
|                                  | $100 < D < 150$         | 6               | Moyenne    |            |
|                                  | $150 < D < 200$         | 4               | Médiocre   |            |
| Gigue (J) (ms)                   | $0 < J < 30$            | 10              | Excellente |            |
|                                  | $30 < J < 50$           | 5               | Bien       |            |
|                                  | $50 < J < 60$           | 3               | Moyenne    |            |
|                                  | $60 < J < 70$           | 3               | Médiocre   |            |
| Taux de perte de paquets (L) (%) | $< 10^{-5}$             | 10              | Excellente |            |
|                                  | $10^{-5} < L < 10^{-4}$ | 7               | Moyenne    |            |
|                                  | $10^{-4} < L < 10^{-3}$ | 3               | Médiocre   |            |
| Taux d'erreur de paquets (E)     | $< 10^{-6}$             | 10              | Excellente |            |
|                                  | $10^{-6} < T < 10^{-5}$ | 7               | Moyenne    |            |
|                                  | $10^{-5} < T < 10^{-4}$ | 4               | Médiocre   |            |
| Bande de passante (B) (Kbps)     | Voix                    | $256 < B < 128$ | 10         | Excellente |
|                                  |                         | $128 < B < 80$  | 8          | Moyenne    |
|                                  |                         | $80 < B < 64$   | 6          | Médiocre   |
|                                  | Vidéo                   | $3 < B < 2$     | 10         | Excellente |
|                                  |                         | $2 < B < 1$     | 8          | Moyenne    |

|  |  |                       |   |          |
|--|--|-----------------------|---|----------|
|  |  | 1 Mbps < B < 512 Kbps | 5 | Médiocre |
|--|--|-----------------------|---|----------|

Tableau 4. Paramètre et plage de QoS liés à la QoE [28].

| Classes de QoS | Applications (Ex)   | Techniques de réseau                  |
|----------------|---|---------------------------------------|
| Classe 0       | Temps réel, sensibilité à la gigue, interaction élevée (VoIP, VTC).             | Contrainte de réseau et distance.     |
| Classe 1       | Temps réel, sensible à la gigue, interactif (VoIP, VTC).                        | Routage moins contraint et distances. |
| Classe 2       | Données de transaction, très interactives. (Signalisation)                      | Contrainte de réseau et distance.     |
| Classe 3       | Données de transaction, interactif.   | Routage moins contraint et distances. |
| Classe 4       | Faible perte seulement (courte transactions, données en vrac, streaming vidéo). | Toute route/chemin.                   |
| Classe 5       | Applications traditionnelles de réseaux IP par défaut.                          | Toute route/chemin.                   |

Tableau 5. Guide pour les classes de QoS [28].

| Paramètres de Performance du réseau | Paramètres de QoS  |                 |                 |                 |                 |          |
|-------------------------------------|--------------------|-----------------|-----------------|-----------------|-----------------|----------|
|                                     | Classe 0           | Classe 1        | Classe 2        | Classe 3        | Classe 4        | Classe 5 |
| <b>IPTD</b>                         | 100 ms             | 400 ms          | 100 ms          | 100 ms          | 1s              | U        |
| <b>IPDV</b>                         | 50 ms              | 50 ms           | U               | U               | U               | U        |
| <b>IPLR</b>                         | $1 \times 10^3$    | $1 \times 10^3$ | $1 \times 10^3$ | $1 \times 10^3$ | $1 \times 10^3$ | U        |
| <b>IPFR</b>                         | $1 \times 10^{-4}$ |                 |                 |                 |                 | U        |

Tableau 6. Définitions des classes de qualité de service du réseau IP et objectifs de performances du réseau [28].

#### 4. Conclusion

Dans ce chapitre, nous avons présenté la notion de QoS, ainsi que ses paramètres sur lesquelles nous allons nous baser afin d'acheminer le flux multimédia entre les serveurs de diffusion. Vu que la perception des utilisateurs retient de plus en plus l'attention des

fournisseurs de services et des chercheurs, cela nous a poussé à étudier la notion de QoE ainsi que sa relation avec la QoS en se basant sur le modèle *Hourglass*. Enfin, nous avons décrit le modèle proposé par Kim et al. [28] qui sera exploité pour estimer la QoS/QoE dans notre solution. Cette solution fera l'objet du chapitre suivant.

## Chapitre 3

### Conception de la solution proposée

#### 1. Introduction

La réalisation de la plateforme doit être impérativement précédée d'une conception qui a pour objectif de formaliser les étapes préliminaires de son développement. Pour la conception du système proposé, nous allons utiliser le langage UML afin de traduire la liste des exigences en diagrammes de cas d'utilisation. Ensuite, nous décrivons l'architecture du système tout en détaillant le processus de sélection du meilleur MCU, la gestion de webinaire (avec des diagrammes de séquence) et le schéma de la base de données (BDD) en utilisant le diagramme de classes.

#### 2. Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation permet de décrire l'interaction entre l'acteur et le système. Un acteur représente un rôle joué par une entité externe (utilisation humaine, dispositif matériel ou autre système) qui interagit directement avec le système étudié [36]. Il peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données.

Dans le cas de notre plateforme, nous pouvons distinguer les acteurs (rôles) suivants :

- **Administrateur** : il prend en charge la gestion des MCU.
- **Conférencier** : il publie, lance et anime une session webinaire.
- **Participant** : il consulte et assiste à une session webinaire.

Le diagramme global suivant (figure 12) regroupe toutes les fonctionnalités pouvant être établies par l'utilisateur de la plateforme. Il est suivi d'une description détaillée des chacune d'elles (tableau 7).

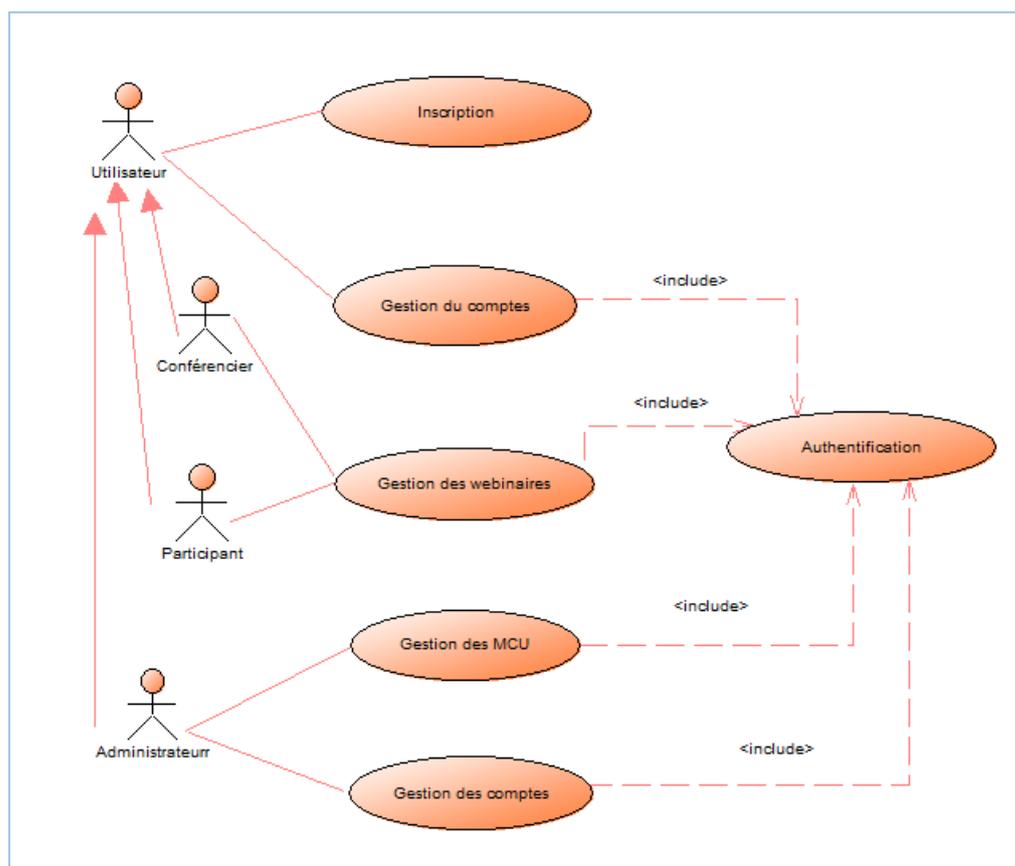


Figure 12. Diagramme général de cas d'utilisation.

| Cas d'utilisation      | Acteur         | Description du cas d'utilisation   |
|------------------------|----------------|--|
| Inscription            | Utilisateur    | Chaque utilisateur peut s'inscrire à la plateforme.                              |
| Gestion du compte      | Utilisateur    | Permet de modifier les informations personnelles ainsi de déconnecter.           |
| Gestion des comptes    | Administrateur | Permet d'ajouter, supprimer un compte.   |
| Gestion des MCU        | Administrateur | Créer, activer, modifier, supprimer, rechercher un MCU                           |
| Gestion des webinaires | Conférencier   | Permet de créer, lancer, modifier, chercher, et supprimer une session webinaire. |
|                        | Participant    | Permet de rejoindre, chercher et quitter une session webinaire.                  |

Tableau 7. Description du diagramme global de cas d'utilisation.

Dans ce qui suit, nous allons détailler les principales fonctionnalités : (i) gestion webinaires (figure 13 et tableau 8) et (ii) gestion des MCU (figure 14 et tableau 9).

## 2.1 Gestion des webinaires

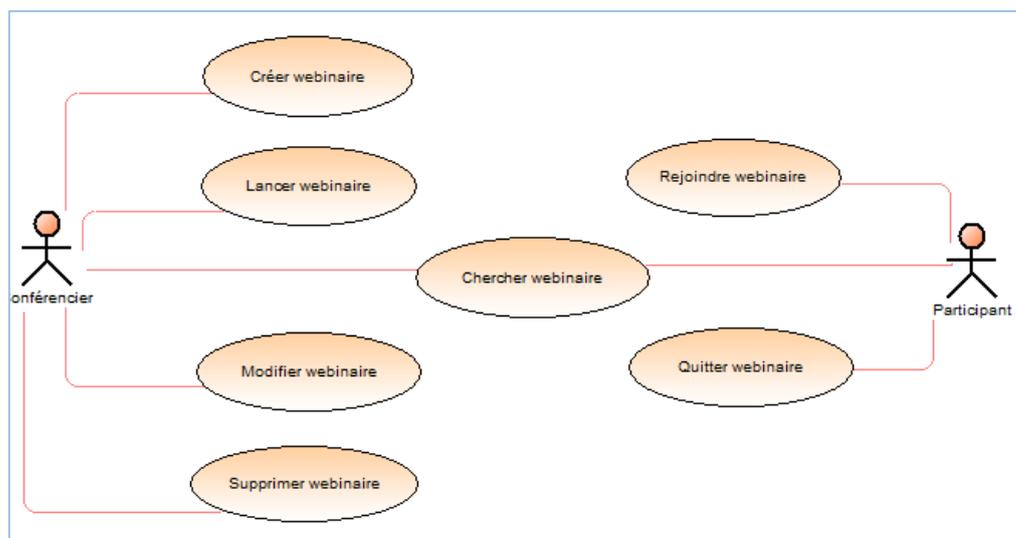


Figure 13. Diagramme de cas d'utilisation « gestion des webinaires ».

| Cas d'utilisation | Acteur                    | Description du cas d'utilisation  |
|-------------------|---------------------------|---|
| Créer             | Conférencier              | Permet de créer sa propre session webinaire.                            |
| Lancer            | Conférencier              | Permet de lancer sa propre session webinaire et l'animer.               |
| Modifier          | Conférencier              | Permet de modifier les informations d'une session webinaire déjà créée. |
| Supprimer         | Conférencier              | Permet de supprimer une session webinaire déjà créée.                   |
| Rejoindre         | Participant               | Permet de rejoindre une session webinaire.                              |
| Rechercher        | Conférencier /Participant | Permet de rechercher une session webinaire déjà créée.                  |
| Quitter           | Conférencier /Participant | Permet de Quitter une session webinaire.                                |

Tableau 8. Description de cas d'utilisation « gestion des webinaires ».

## 2.2 Gestion des MCU

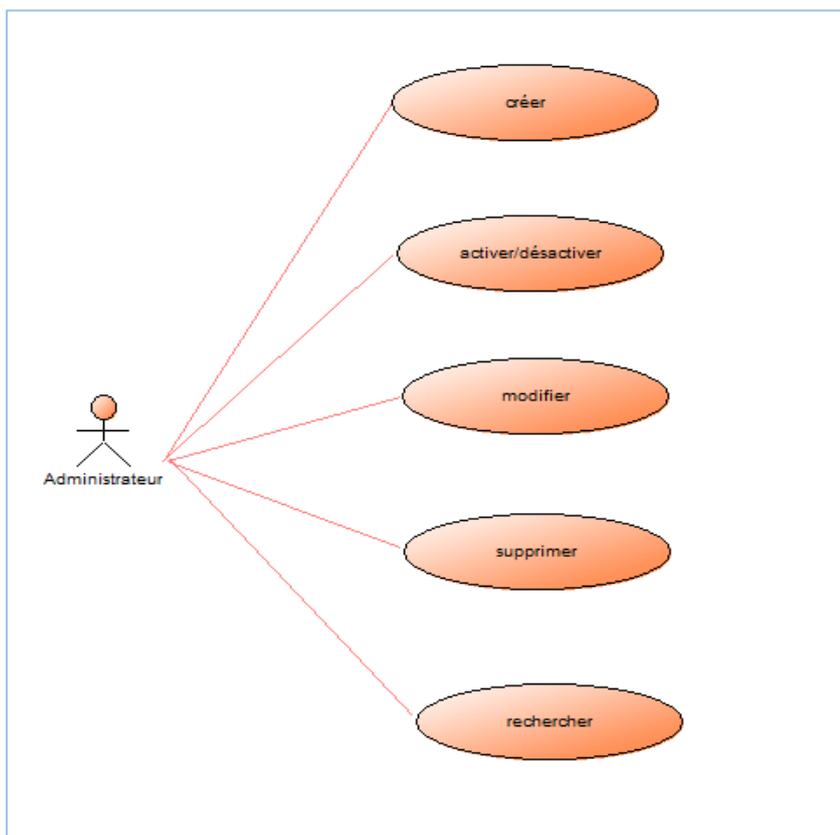


Figure 14. Diagramme de cas d'utilisation « gestion des MCU ».

| Cas d'utilisation  | Acteur         | Description du cas d'utilisation   |
|--------------------|----------------|--|
| Créer              | Administrateur | Permet de créer un MCU en remplissant un formulaire de création.                 |
| Activer/désactiver | Administrateur | Permet à l'administrateur de changer l'état de MCU.                              |
| Modifier           | Administrateur | Permet de changer le nom, le port, l'adresse et/ou la charge d'un MCU déjà créé. |
| Rechercher         | Administrateur | Permet de rechercher un MCU déjà créé.   |
| Supprimer          | Administrateur | Permet de supprimer un MCU déjà créé.  |

Tableau 9. Description de cas d'utilisation « gestion des webinaires ».

## 3. Architecture du système proposé

La figure 15 ci-dessous décrit l'architecture du système « *Educ\_Webinar* » proposé ainsi que les interactions entre ses différentes entités.

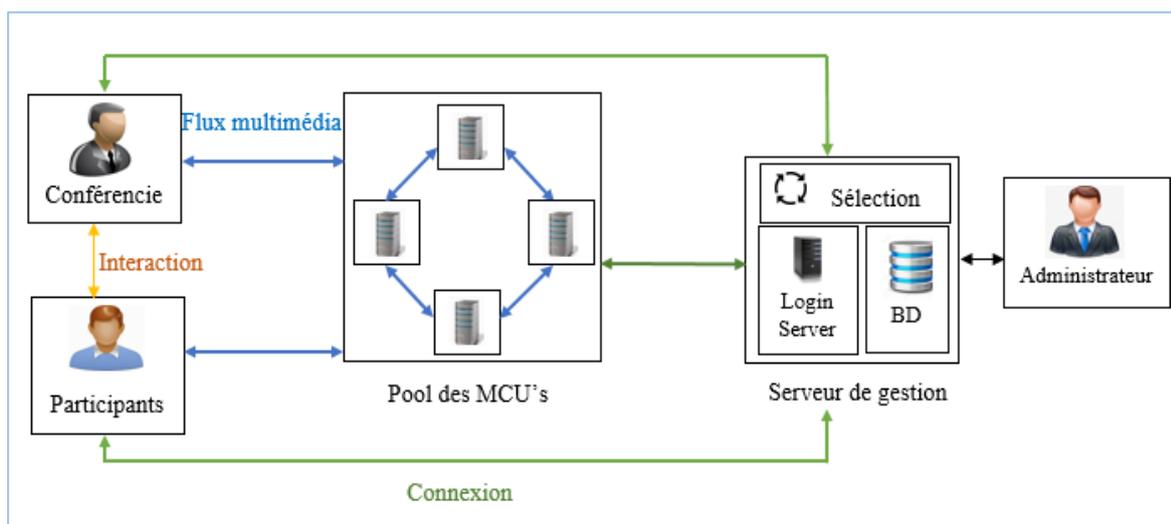


Figure 15. Architecture du système « *Educ\_Webinar* ».

Le système est constitué des entités suivantes :

- Un serveur de gestion, qui gère toutes les connexions du système, ainsi que la sélection du meilleur MCU qui répond aux besoins des utilisateurs.
- Des unités de contrôle multipoint MCU, qui s'occupent de la diffusion de flux multimédia du conférencier aux participants.
- Un ensemble d'utilisateurs (administrateur, conférencier, et participants).

Les échanges entre les utilisateurs et le serveur de gestion s'effectuent avec le protocole TCP afin de garantir le transfert de données de façon fiable et assurer une communication bidirectionnelle. Cependant, les échanges de flux multimédia entre les utilisateurs et le MCU s'effectuent avec le protocole UDP (RTP/RTCP), le plus adapté à transmettre les flux multimédias (voir section 2.3).

### 3.1 Fonctionnement

Les utilisateurs (conférenciers et participants) doivent s'inscrire à la plateforme, « *Educ\_Webinar* », afin d'y accéder et de pouvoir bénéficier des services offerts. Comme illustré dans la figure 16, le fonctionnement de cette plateforme passe par quatre phases successives :

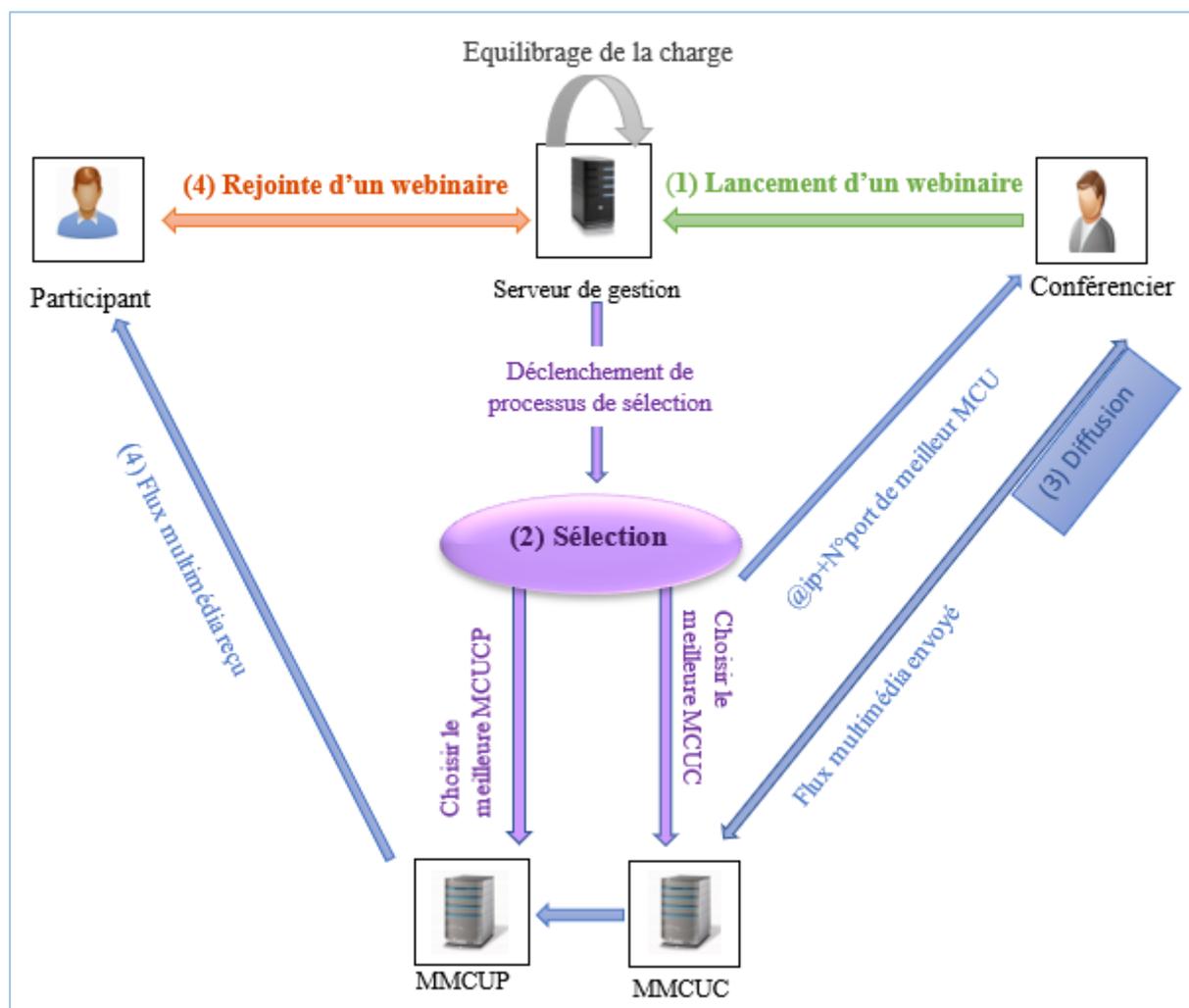


Figure 16. Fonctionnement de la plateforme « *Educ\_Webinar* ».

### (1) Phase de lancement

Lorsqu'un conférencier lance un webinaire, une requête de recherche de MCU est envoyée de manière transparente au serveur de gestion ; ainsi, le processus de sélection est déclenché.

### (2) Phase de sélection

Lors du lancement d'un webinaire, le serveur de gestion s'occupe de sélectionner le **Meilleur MCU pour Conférencier (MMCUC)** offrant une meilleure QoS et QoE. Cette phase sera détaillée dans la section suivante.

### (3) Phase de diffusion

Une fois le MMCU est sélectionné, le serveur de gestion répond à la demande du conférencier en communiquant le numéro de port de ce MMCU. Une session UDP est ensuite établie entre le conférencier et le MMCU sélectionné.

### (4) Rejoindre un webinaire

Lorsqu'un participant lance un webinaire, une requête de recherche de MCU est envoyée de manière transparente au serveur de gestion, afin de trouver le MMCU pour Participant (MMCUP) qui véhicule le webinaire concerné. Dans le cas où le meilleur MCU du participant est différent de celui du conférencier, une session de diffusion doit d'abord être établie entre les deux serveurs (MMCUC et MMCUP). Ensuite, une session de diffusion est établie entre le participant et le MMCUP.

## 3.2 Processus de sélection du MMCU

Cette procédure est effectuée par le serveur de gestion afin de sélectionner le MMCU offrant une meilleure qualité à l'utilisateur (conférencier ou participant). Pour cela, il faut estimer la QoS entre l'utilisateur et tous les MCU disponibles en envoyant des messages de contrôle et mesurer les performances du réseau en termes de délai, gigue et perte de paquet. Ces paramètres sont mesurés de manière active en utilisant la commande *ping*. Les résultats de mesure sont alors envoyés au serveur de gestion afin d'estimer la QoS et la QoE à partir du modèle de Kim (section 3.3.2), comme illustré dans la figure 17 :

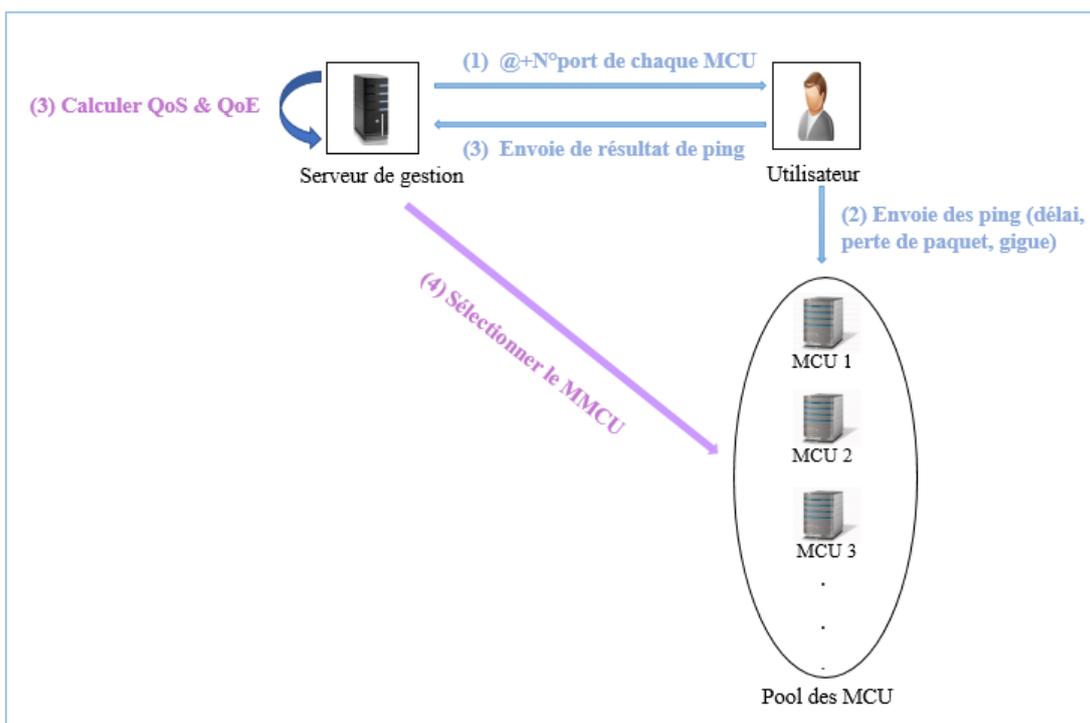


Figure 17. Sélection de MMCUP ou MMCUC.

Dans le cas où plusieurs serveurs MCU offre la même QoE, nous prenons en considération leurs charges respectives (capacité de traitement : nombre des utilisateurs affectés) en choisissant le moins chargé. Ainsi, notre processus permet d'équilibrer la charge dans le pool des MCU.

#### 4. Diagramme de séquence « Gestion de webinaire »

Le diagramme de séquence permet de cacher les interactions d'objets dans le cadre d'un scénario d'un diagramme de cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets [37].

Après avoir expliqué le fonctionnement du système proposé, nous allons présenter les principales interactions entre les différentes entités du « *Educ-Web* » à l'aide d'un diagramme de séquence. La figure 18 décrit la création et le lancement d'une session webinaire par le conférencier ; quant à la figure 19, elle présente comment le participant peut rejoindre ou une session webinaire.

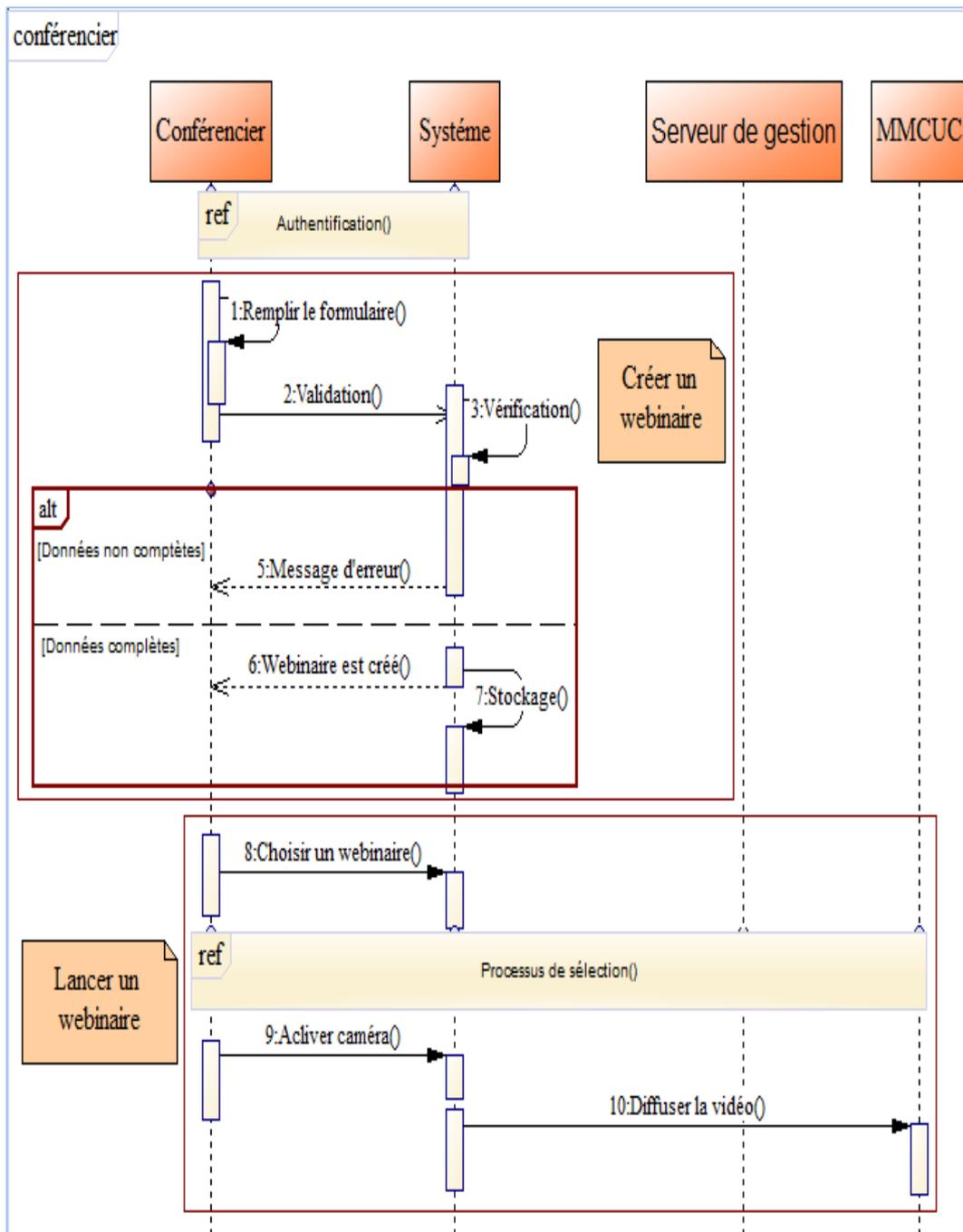


Figure 18. Principales interactions entre le conférencier et la plateforme.

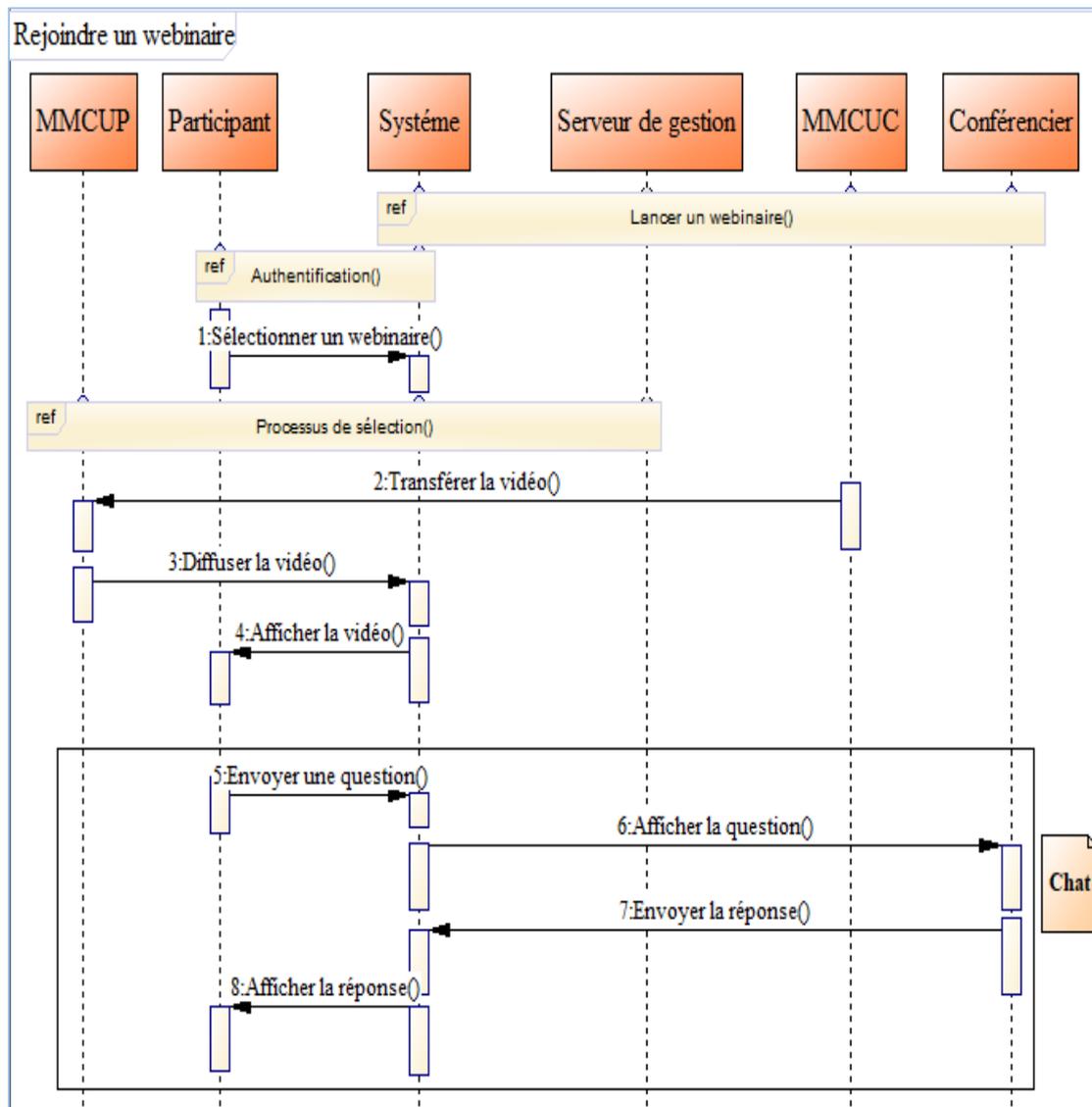


Figure 19. Principales interactions entre le participant et la plateforme.

## 5. Diagramme de classes

Le diagramme de classes présente les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques [38]. Notre diagramme de classes et sa description sont illustrés respectivement dans la figure 20 et le tableau 10.

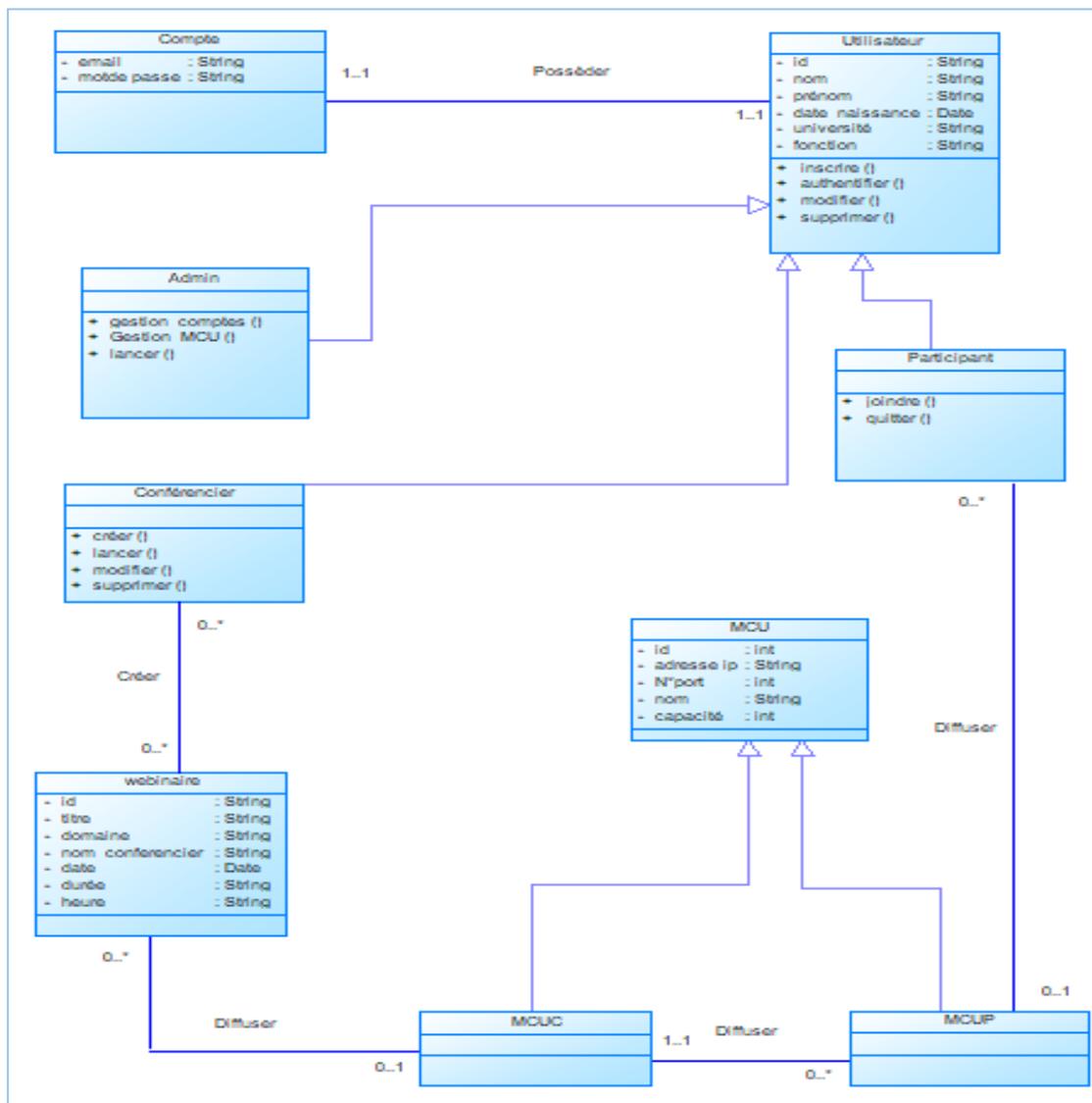


Figure 20. Diagramme de classes.

| Classe      | Attribut          | Type   | Désignation                       | Méthode        |
|-------------|-------------------|--------|-----------------------------------|----------------|
| Compte      | Email             | String | Émail de l'utilisateur.           |                |
|             | Mot de passe      | String | Mot de passe de l'utilisateur.    |                |
| Utilisateur | Id                | String | Identifiant                       | Inscrire()     |
|             | Nom               | String | Nom                               |                |
|             | Prénom            | String | Prénom                            | Authentifier() |
|             | Date de naissance | Date   | Date de naissance                 | Modifier()     |
|             | Fonction          | String | Définir une fonction              | Supprimer()    |
|             | Université        | String | Définir l'université              |                |
| Webinaire   | Id                | String | Identifiant                       |                |
|             | Titre             | String | Titre du webinaire                |                |
|             | Domaine           | String | Domaine du webinaire              |                |
|             | Nom_conf          | String | Nom du conférencier               |                |
|             | Date              | String | Date de diffusion                 |                |
|             | Heure             | String | Heure de diffusion                |                |
|             | Durée             | String | Durée de diffusion                |                |
| MCU         | Id                | String | Identifiant de MCU                |                |
|             | Adresse_ip        | String | Adresse de MCU                    |                |
|             | N° port           | String | Port de MCU                       |                |
|             | Nom               | String | Nom de MCU                        |                |
|             | Capacité          | String | Numéro du participant à supporter |                |

Tableau 10. Tableau descriptif des classes.

## 6. Conclusion

Ce chapitre était consacré à la conception détaillée de la plateforme proposée en utilisant le langage UML. En outre, nous avons spécifié les fonctionnalités de cette plateforme et modélisé avec les diagrammes UML. En plus, ce chapitre a décrit la solution conceptuelle qui répond à nos besoins.

Dans le chapitre suivant, nous détaillons les étapes suivies pour implémenter notre solution « *Educ-Web* ».

## Chapitre 4

### Implémentation, Tests et Résultats

#### 1. Introduction

Notre but étant fixé et sa conception élaborée, nous allons entamer dans ce chapitre la partie réalisation qui constitue le dernier volet de ce mémoire et qui a pour objectif d'exposer notre plateforme « *Educ\_Webinar* », une plateforme d'organisation des sessions webinaires entre les différentes universités du territoire national.

Ce chapitre se décompose en deux parties : la première partie représente l'implémentation de notre plateforme ; quant à la deuxième partie, elle expose les tests expérimentaux et les résultats obtenus.

#### 2. Implémentation

La figure 21 montre les principaux logiciels utilisés pour mettre en œuvre la plateforme « *Educ-Webinar* ». Cette plateforme est une application *Java* permettant de gérer les différentes parties :

- 1) La partie des serveurs inclut :
  - Le serveur de gestion qui s'occupe de la sélection des serveurs MCU.
  - Le serveur login qui interroge le serveur BDD « MySQL » [41] via des requêtes écrites en langage SQL.
  - Le serveur de messagerie « Mosquitto broker » [42] qui permet de gérer le chat entre le conférencier et les participants via le protocole MQTT [43].
  - Les serveurs MCU qui s'occupent de la diffusion de flux multimédia du conférencier aux participants.
- 2) La partie des utilisateurs où se trouve l'ensemble des interfaces graphiques Swing (Java) mis en place par l'environnement *NetBeans* [43]. Ces interfaces facilitent

l'utilisation de la plateforme via les différentes fonctionnalités offertes comme montré dans la section suivante.

La description des logiciels de développement utilisés (NetBeans, serveur SQL, Mosquitto broker, MQTT, java, SQL) est donnée dans l'annexe A.

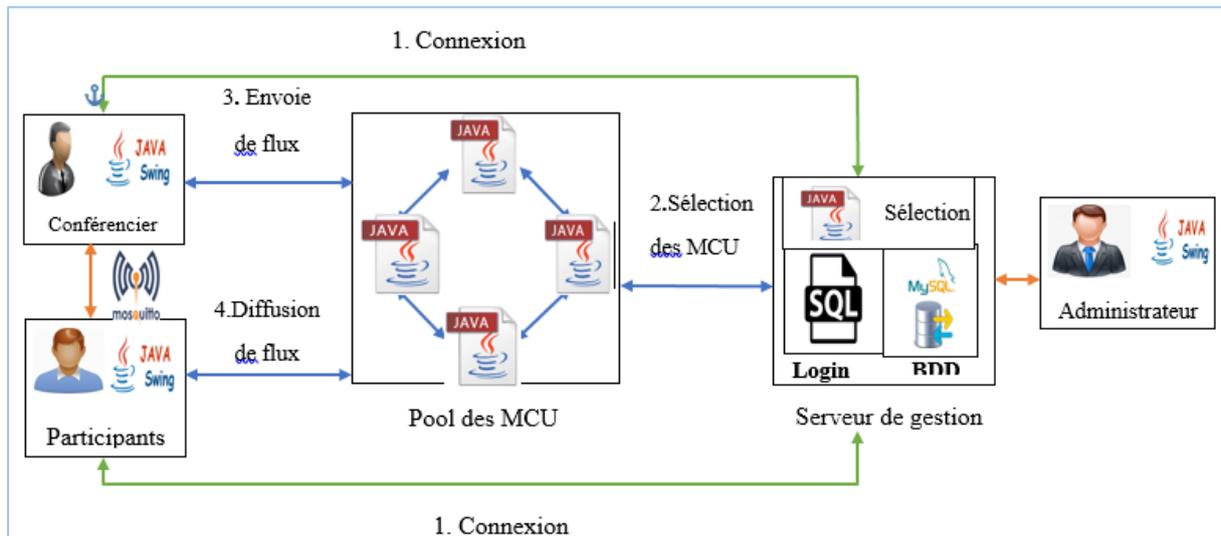


Figure 21. Logiciels de développement.

### 3. Présentation de la plateforme « Educ-Webinar »

Au lancement de la plateforme, la page d'authentification est la première page qui s'affiche pour y accéder. Elle met en avant le logo choisi pour la plateforme « *Educ-Webinar* » et permet ainsi à un utilisateur d'accéder à son propre espace d'un côté. D'un autre côté, cette page permet à un utilisateur non enregistré de s'inscrire à la plateforme pour pouvoir bénéficier de ses fonctionnalités, comme illustré dans la figure 22 :

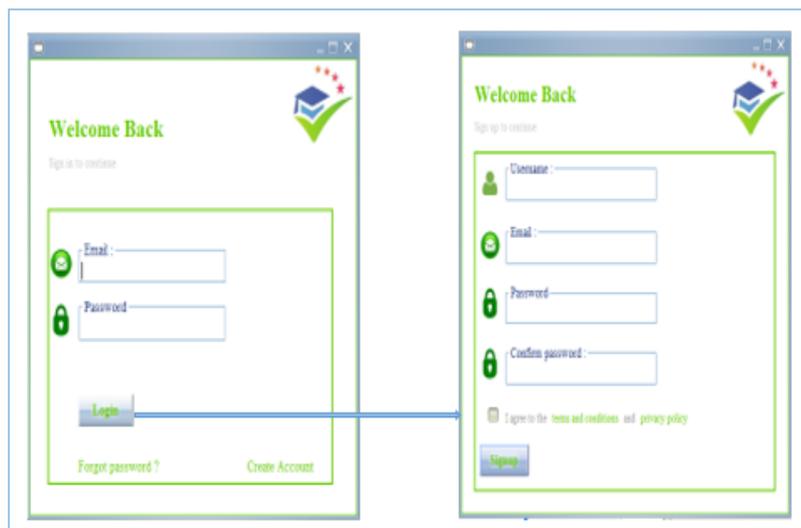


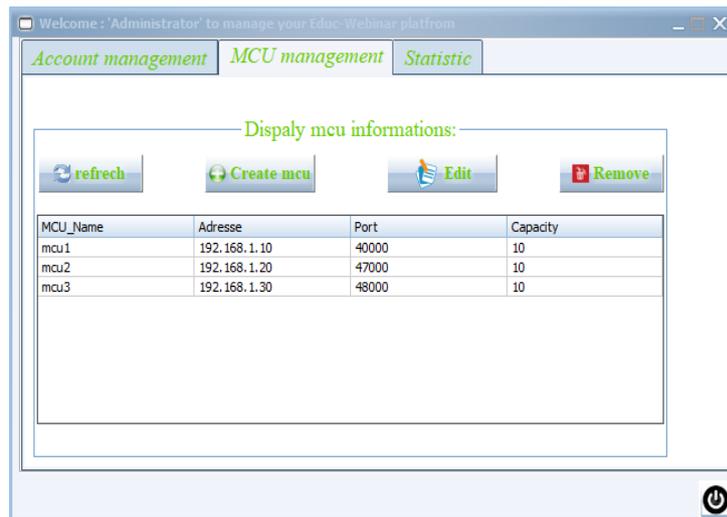
Figure 22. Inscription et authentification.

### 3.1 Espace Administrateur

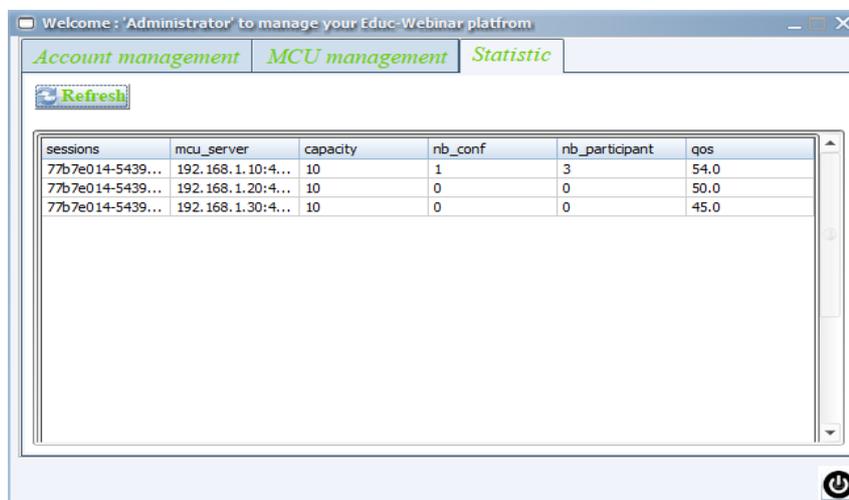
Cette espace contient les interfaces nécessaires à l'administrateur pour qu'il puisse exécuter les tâches administratives telles que la gestion des comptes et la gestion des serveurs de diffusion (MCU).

#### 3.1.1 Gestion de MCU

La gestion des MCU permet à l'administrateur de créer, modifier ou supprimer un MCU. En parallèle, un aperçu sur les webinaires qui passe sur chaque nœud, ainsi la qualité de service offerte par les MCU peuvent être affichée, comme illustré dans la figure 23 (a et b) :



(a) Interface de gestion des MCU.



(b) Statistique des MCU.

Figure 23.Administration des MCU

### 3.1.2 Gestion des comptes

La gestion des comptes permet à l'administrateur d'activer, désactiver ou de supprimer un compte déjà créé, comme illustré dans la figure 24 :

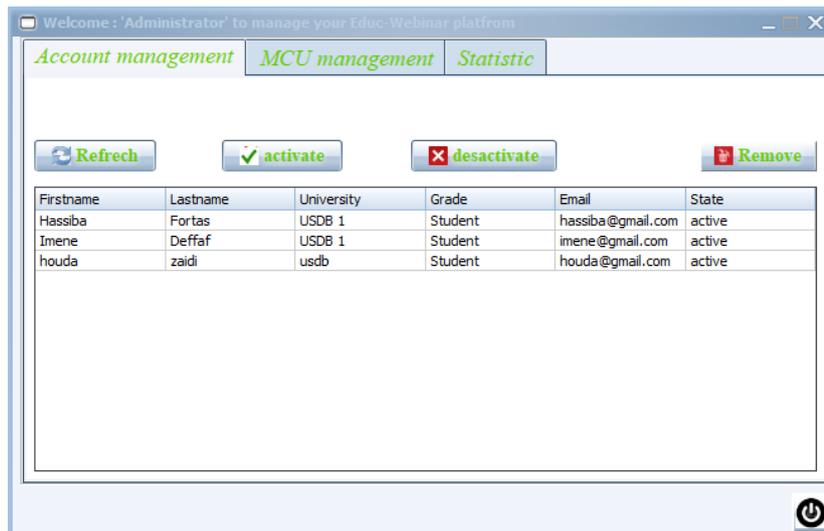


Figure 24. Gestion des comptes.

### 3.2 Espace Utilisateur

Le conférencier peut créer et lancer une session webinaire et le participant peut consulter et rejoindre cette session. Quand le conférencier choisit un webinaire à lancer, une autre interface s'affiche. Il s'agit de celle permettant de commencer la diffusion, et d'envoyer des messages aux participants assistant au webinaire, comme illustré dans la figure 25 :

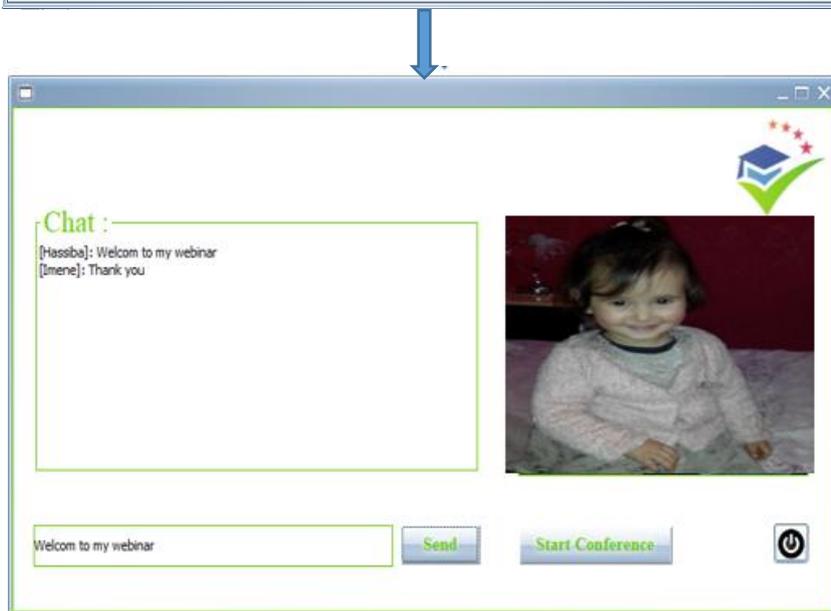
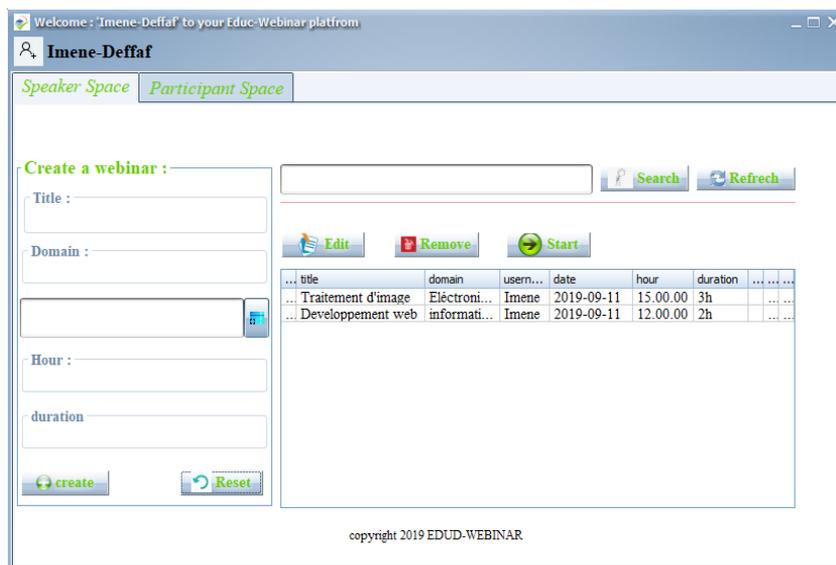


Figure 25.Création et lancement d'un webinaire

D'autre part, le participant peut consulter la liste des webinaires existant afin d'y assister. S'il choisit de rejoindre un webinaire particulier, une fenêtre sera alors affichée qui permet au participant de visionner la session et d'envoyer des questions au conférencier comme l'indique la figure 26 :

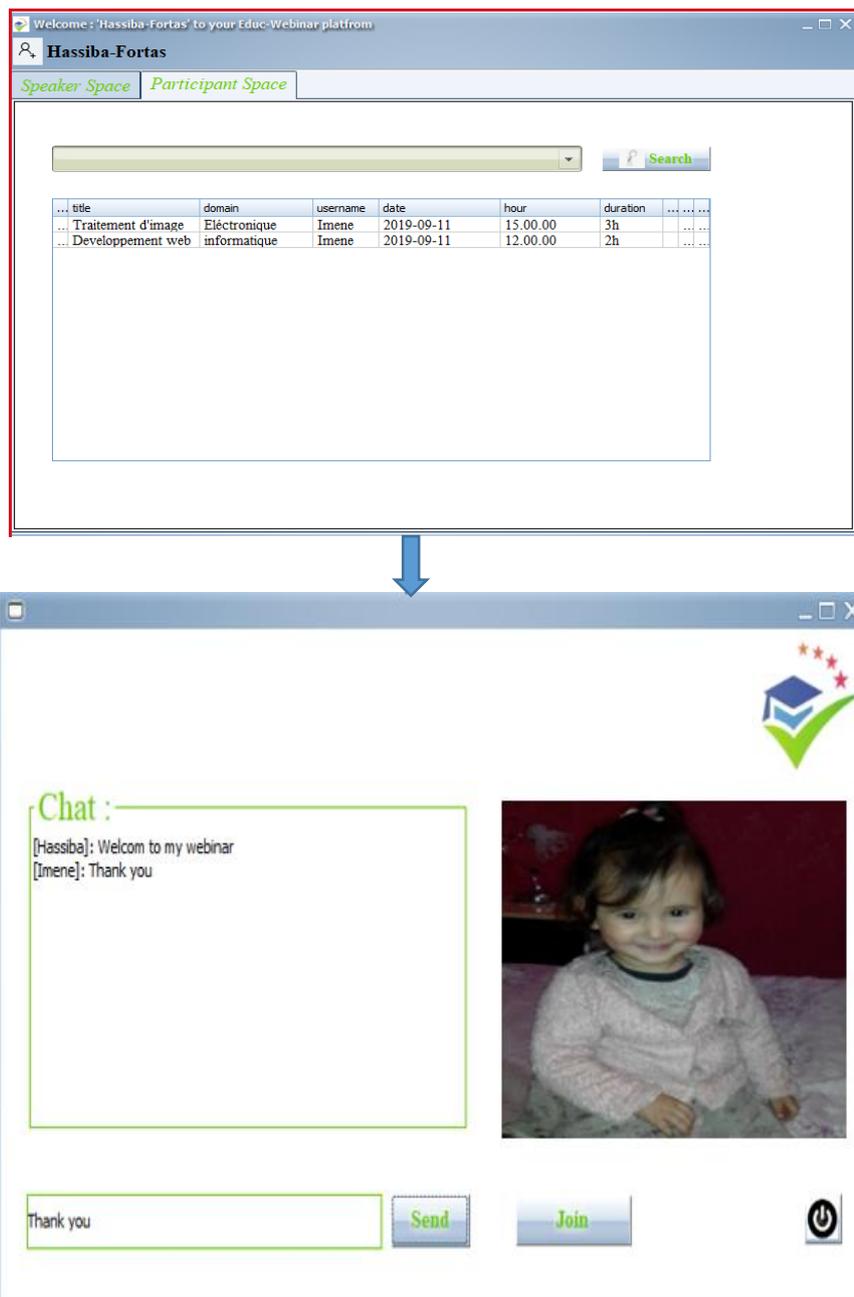


Figure 26.Consulter et rejoindre un webinaire.

#### 4. Tests expérimentaux

Dans cette section, nous allons décrire les tests expérimentaux réalisés et discuter les principaux résultats obtenus.

## 4.1 Environnement de tests

Pour la réalisation de l'environnement de tests, nous avons utilisé deux machines physiques connectées via un réseau (figure 27). La première machine « *Machine A* » regroupe les différents serveurs MCU avec l'émulateur réseau Netem (voir l'annexe B) [44]. Quant à la deuxième machine « *Machine B* », elle regroupe le serveur de gestion et les utilisateurs (conférencier et participants).

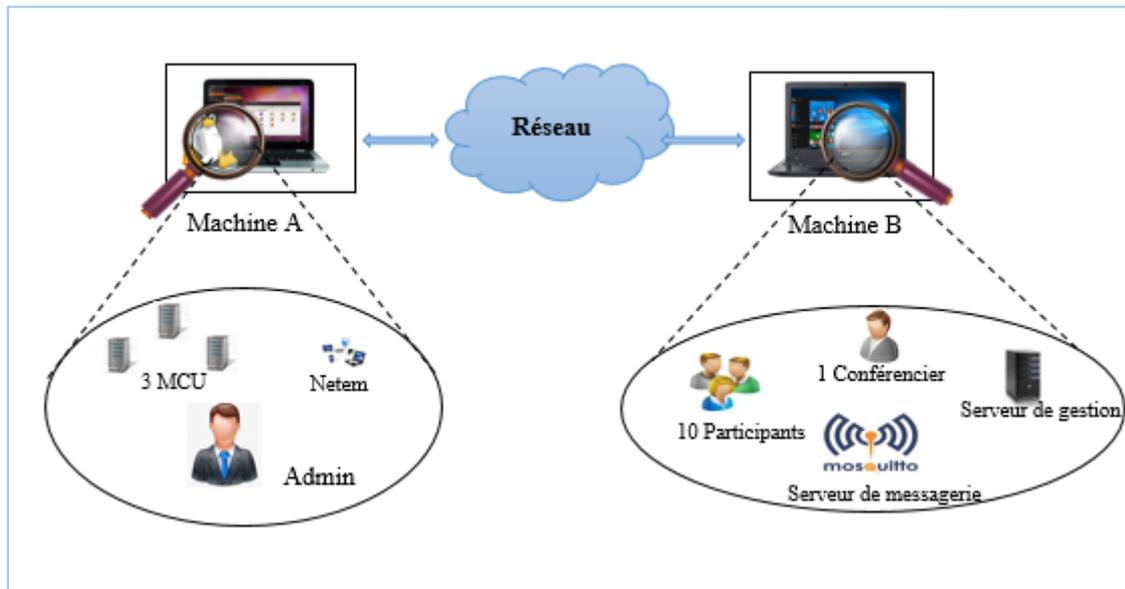


Figure 27. Environnement de tests.

Le tableau 11 montre les différentes caractéristiques de chaque machine utilisée :

| Caractéristiques       | Machine A                            | Machine B  |
|------------------------|--------------------------------------|--|
| Système d'exploitation | Linux                                | Windows ou Linux   |
| Processeur             | Intel® Atom™ CPU N2800 @ 1,86 GHZ *4 | Intel® Core™ i5-5200 CPU@2.20 GHZ 2.20 GHZ                             |
| RAM                    | 2.00 Go                              | 4.00 Go  |
| Éléments inclus        | Serveurs MCU<br>Serveur Netem        | Serveur de gestion, conférenciers,<br>participants, serveur messagerie |

Tableau 11. Caractéristiques de machines utilisées.

## 4.2 Scénario de test

Pour tester le processus de sélection de serveurs MCU proposé, nous avons utilisé :

- Trois serveurs MCU et un serveur de gestion au niveau de la « *Machine A* ».
- Dix participants, un conférencier et une session webinaires au niveau de la « *Machine B* ».

Afin de mettre en place ce scénario, nous avons suivi les étapes suivantes :

#### 4.2.1 Configuration de la « *Machine A* »

D'une part, cette machine est le fournisseur des serveurs MCU attribués avec des adresses virtuelles configurées à l'aide de la commande de configuration « *ifconfig* ». Premièrement, nous avons configuré cette machine en attribuant une adresse IP pour créer des adresses virtuelles et les utiliser par l'émulateur *Netem* afin de contrôler leur trafic via la commande « *tc* ». Nous avons utilisé la commande « *ifconfig interface @IP up* » pour attribuer les adresses IP aux interfaces utilisées dans le réseau établi à condition qu'elles soient dans la même plage d'adressage de la machine.

Dans notre cas, nous avons utilisés trois serveurs MCU, donc nous avons créé trois adresses virtuelles. La procédure de création est la suivante :

L'adresse de la « *Machine A* » est : 192.168.1.10

```
sudo      ifconfig      eth0:1      192.168.1.100      up
sudo      ifconfig      eth0:2      192.168.1.150      up
sudo      ifconfig      eth0:3      192.168.1.200      up
```

Le tableau 12 montre la liaison de chaque serveur MCU avec son interface et son adresse IP :

| MCU   | Interface | Adresse IP    |
|-------|-----------|---------------|
| MCU 1 | Eth0 :1   | 192.168.1.100 |
| MCU 2 | Eth0 :2   | 192.168.1.150 |
| MCU 3 | Eth0 :3   | 192.168.1.200 |

Tableau 12. Configuration des MCU.

Après cette configuration, il faut utiliser la commande « *tc* » qui permet de générer des perturbations sur les paquets entrant de l'interface de chaque MCU. Nous nous intéressons aux trois paramètres réseaux suivants : le délai, la gigue et la perte de paquet. Le tableau 13 présente les valeurs utilisées :

| Test         | MCU  | Délai (ms) | Gigue (ms) | Perte de paquet (%) |
|--------------|------|------------|------------|---------------------|
| <b>Test0</b> | Mcu1 | /          | /          | /                   |
|              | Mcu2 | /          | /          | /                   |
|              | Mcu3 | /          | /          | /                   |
| <b>Test1</b> | Mcu1 | 20         | /          | /                   |
|              | Mcu2 | 100        | /          | /                   |
|              | Mcu3 | 150        | /          | /                   |
| <b>Test2</b> | Mcu1 | /          | /          | 1                   |
|              | Mcu2 | /          | /          | 2                   |
|              | Mcu3 | /          | /          | 4                   |
| <b>Test3</b> | Mcu1 | 20         | 10         | /                   |
|              | Mcu2 | 70         | 30         | /                   |
|              | Mcu3 | 120        | 50         | /                   |
| <b>Test4</b> | Mcu1 | 70         | /          | 1                   |
|              | Mcu2 | 100        | /          | 2                   |
|              | Mcu3 | 130        | /          | 4                   |
| <b>Test5</b> | Mcu1 | 20         | /          | 1                   |
|              | Mcu2 | 40         | /          | 2                   |
|              | Mcu3 | 60         | /          | 4                   |
| <b>Test6</b> | Mcu1 | 100        | 20         | 1                   |
|              | Mcu2 | 130        | 30         | 2                   |
|              | Mcu3 | 160        | 50         | 4                   |

Tableau 13. Paramètres du réseau.

En utilisant NetEm, nous pouvons modifier ces paramètres comme suit :

- **Délai:** `sudo tc qdisc change dev eth0:1 root netem delay 30 ms.`
- **Gigue:** `sudo tc qdisc change dev eth0:1 root netem delay 100 ms 10ms.`
- **Perte de paquet :** `sudo tc qdisc change dev eth0 :1 root netem loss 2%.`

Enfin, l'administrateur peut créer les serveurs MCU via l'interface de gestion des MCU (figure 23) en cliquant sur « *create MCU* » ce qui permet d'afficher la fenêtre suivante (figure 28) :

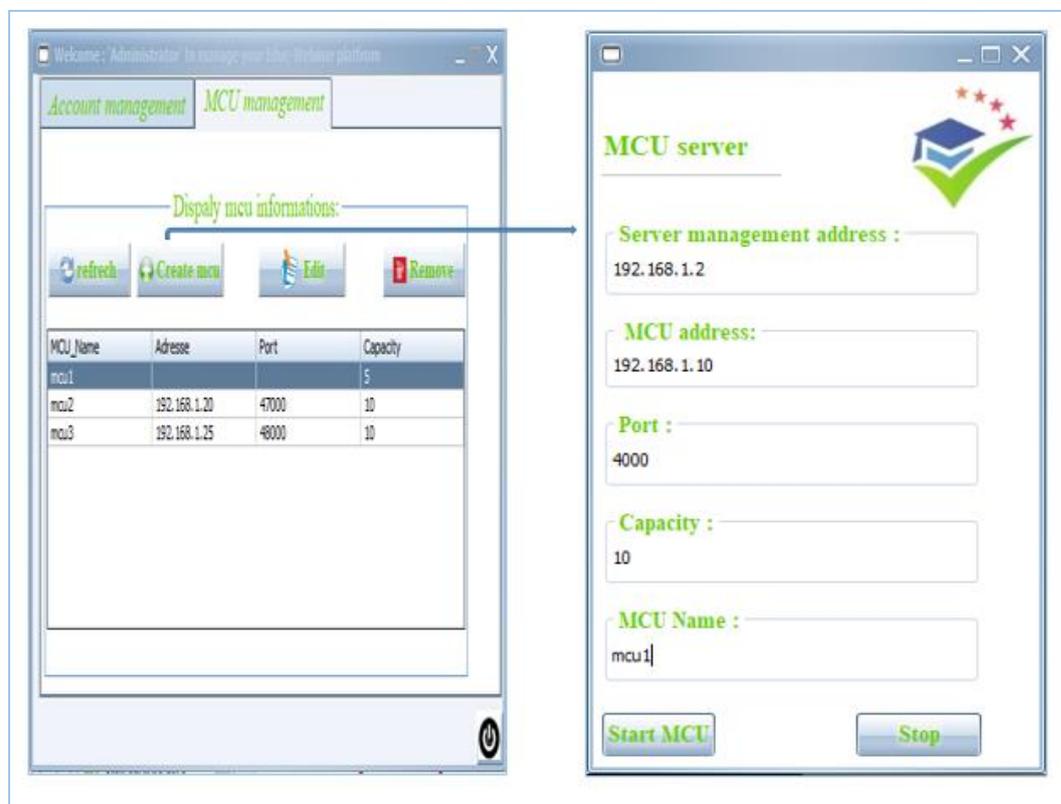


Figure 28.Lacement de MCU.

D'autre part, le serveur de gestion est lancé via l'interface suivant (figure 28):

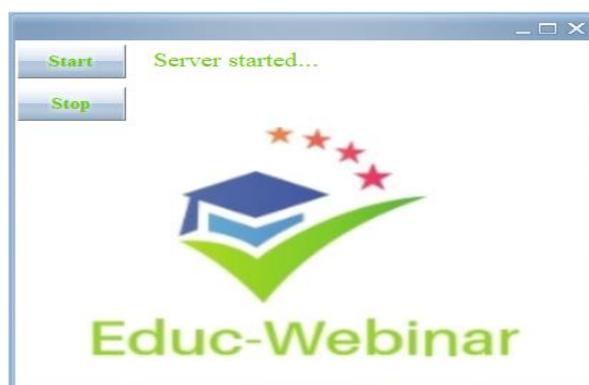


Figure 29. Lancement de serveur de gestion.

#### 4.2.2 Configuration de la « Machine B »

Concernant la deuxième machine, nous pouvons utiliser n'importe quel système d'exploitation (Windows, Linux, Mac), car notre application est développée en Java. Cependant, nous devons configurer la machine physique en attribuant l'adresse IP 192.168.1.20 afin qu'elle soit dans la même plage d'adressage de la « Machine A ».

### 5. Résultats obtenus

Le tableau 14 récapitule les résultats obtenus qui montre la sélection du MMCUC selon la QoS et la QoE.

| Tests        | MCU  | Délai estimé | Gigue estimé | Perte estimée de paquet | QoS | QoE | Charge | MMCUC sélectionnée |
|--------------|------|--------------|--------------|-------------------------|-----|-----|--------|--------------------|
| <b>Test0</b> | Mcu1 | 0            | 7.07         | 0                       | 54  | 4.8 | 10     | MCU1               |
|              | Mcu2 | 15           | 7.57         | 0                       | 54  | 4.8 | 10     |                    |
|              | Mcu3 | 0            | 7.57         | 0                       | 54  | 4.8 | 10     |                    |
| <b>Test1</b> | Mcu1 | 156          | 7.78         | 0                       | 46  | 4   | 10     | MCU3               |
|              | Mcu2 | 172          | 7.08         | 0                       | 46  | 4   | 10     |                    |
|              | Mcu3 | 172          | 7.78         | 0                       | 46  | 4   | 10     |                    |
| <b>Test2</b> | Mcu1 | 16           | 22.16        | 2                       | 51  | 4.5 | 10     | MCU2               |
|              | Mcu2 | 0            | 7.07         | 1                       | 54  | 4.8 | 10     |                    |
|              | Mcu3 | 15           | 21.80        | 4                       | 46  | 4   | 10     |                    |
| <b>Test3</b> | Mcu1 | 123          | 63           | 0                       | 42  | 3.8 | 10     | MCU3               |
|              | Mcu2 | 125          | 32.52        | 0                       | 44  | 4   | 10     |                    |
|              | Mcu3 | 157          | 19.15        | 0                       | 46  | 4   | 10     |                    |
| <b>Test4</b> | Mcu1 | 133          | 3.36         | 1                       | 48  | 4   | 10     | MCU1               |
|              | Mcu2 | 136          | 2.16         | 2                       | 45  | 4   | 10     |                    |
|              | Mcu3 | 142          | 3.41         | 4                       | 40  | 3.9 | 10     |                    |
| <b>Test5</b> | Mcu1 | 5            | 0.81         | 1                       | 54  | 4.8 | 10     | MCU1               |
|              | Mcu2 | 40           | 27.21        | 2                       | 51  | 4.5 | 10     |                    |
|              | Mcu3 | 14           | 40.41        | 4                       | 42  | 3.8 | 10     |                    |
| <b>Test6</b> | Mcu1 | 179          | 39.59        | 1                       | 42  | 3.8 | 10     | MCU1               |
|              | Mcu2 | 143          | 31.03        | 2                       | 41  | 3.6 | 10     |                    |
|              | Mcu3 | 127          | 45.38        | 4                       | 34  | 3   | 10     |                    |

Tableau 14. Table de tests

D'après le tableau 14, on remarque que le MMCUC sélectionné est celui qui a une meilleure QoS et QoE. Ces dernières sont calculées comme définit par le modèle de Kim et al (chapitre 2 section 3.3.2), par exemple pour le test N4 les calculs sont effectués comme suit :

Test N° 4  $\rightarrow$  QoS (MCU1) = Score (délai) + Score (gigue) + Score (perte de paquet)

$$\text{QoS (MCU1)} = \text{Classe (133)} + \text{Classe (3.36)} + \text{Classe (1)}$$

$$\text{QoS (MCU1)} = 48$$

$$\text{QoE(QoS)} = K \left\{ \frac{e^{\text{QoS}-\alpha} + e^{-\text{QoS}+\alpha}}{e^{\text{QoS}-\alpha} + e^{-\text{QoS}+\alpha+\beta}} + 1 \right\}$$

$$\text{QoE(48)} = 2 \left\{ \frac{e^{48-0} + e^{-48+0}}{e^{48-0} + e^{-48+0+0}} + 1 \right\}$$

$$\text{QoE (MCU1)} = 4$$

Notons ici que durant l'implémentation et faute de temps, nous n'avons pas pu programmer la fonction qui calcule la QoE. Les valeurs de la QoE données dans le tableau sont calculées manuellement. Nous constatons que le meilleur MCU possédant la meilleure QoS, possède aussi la meilleur QoE.

Concernant la sélection des MMCUP, nous trouvons les mêmes résultats de performances vu que nous avons mis le conférencier et les participants dans la même machine et le même réseau. Cependant, la sélection sera différente car on ne prend en considération que la charge des serveurs ; par exemple :

- Dans le test 4, le conférencier est affecté au MCU1 ainsi que les neufs premiers participants. Ensuite, comme le MCU1 devient saturé, le dernier participant est affecté au MCU2.
- Dans le test 1 où tous les serveurs MCU ont la même QoS, nous pouvons avoir le scénario suivant (tableau 15) :

| Participant <sup>4</sup> | P1   | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |   |
|--------------------------|------|----|----|----|----|----|----|----|----|-----|---|
| Charge                   | MCU1 | 1  | 1  | 1  | 1  | 2  | 2  | 2  | 3  | 3   | 4 |
|                          | MCU2 | 0  | 0  | 1  | 1  | 1  | 2  | 3  | 3  | 3   | 3 |
|                          | MCU3 | 0  | 1  | 1  | 2  | 2  | 2  | 2  | 2  | 3   | 3 |
| MMCUP                    | 1    | 3  | 2  | 3  | 1  | 2  | 2  | 1  | 3  | 1   |   |

Tableau 15. Table de test2.

Les résultats obtenus confirment aussi la relation discutée auparavant entre les paramètres réseau (chap2.section2) ainsi l'impact de ces paramètres sur la QoS et sur la qualité de vidéo. En effet, dans le test 6 la qualité de vidéo devient médiocre (blocage, lente) ce qui gêne le confort des participants.

## 6. Conclusion

Dans ce chapitre, nous avons en premier lieu présenté notre plateforme « *Educ\_Webinar* » à travers ces différentes interfaces tout en exposant les fonctions qui constituent chaque interface. Ensuite, nous avons abordé les tests, commençant par présenter l'environnement de tests ainsi que les différentes configurations nécessaires. Enfin, nous avons discuté les résultats obtenus après avoir réalisé plusieurs scénarios afin de montrer le bon fonctionnement de la solution proposée.

---

<sup>4</sup> On suppose que le participant  $P_i$  arrive avant le participant  $P_{i+1}$

## Conclusion Générale

Dans le cadre de ce travail, notre objectif consiste à développer un mécanisme de routage multicritères de flux multimédia dans le cadre de service de webinaire, afin d'offrir une meilleure qualité aux utilisateurs finaux.

Pour ce faire, nous avons en premier lieu, étudié les différentes architectures des outils webinaires existants dans la littérature ainsi leurs fonctionnements.

En deuxième lieu, nous avons étudié la qualité de service et la qualité d'expérience. Dans notre travail, nous nous sommes limités volontairement aux trois paramètres de QoS pour gérer le routage de flux multimédia, à savoir la latence, la gigue et la perte de paquets. Ceci peut être justifié par le fait que les travaux réalisés dans [28] et [29] ont arrivé à la conclusion que le service de webinaire n'est sensible qu'à ces trois paramètres. Par ailleurs, nous avons opté pour le modèle proposé par Kim et al. Dans [28], où la QoS est modélisée comme une somme pondérée des trois paramètres choisis pour estimer la QoE.

En troisième lieu, nous avons entamé la conception de notre solution en décrivant son architecture générale qui se compose de deux parties :

- Partie « utilisateurs » regroupant l'administrateur, le conférencier et les participants ;
- Partie « serveurs » incluant le serveur de gestion et les serveurs MCUs.

L'utilisateur peut gérer les webinaires de telle sorte que le conférencier peut créer et lancer un webinaire, et les participants peuvent joindre une session webinaire. La diffusion entre le conférencier et les participant est assuré via les MCUs qui sont créés par l'administrateur. Au fait, le serveur de gestion s'occupe de la sélection du meilleur MCU pour le conférencier (MMCUC) et pour le participant (MMCUP). Par la suite, une session est établie entre les différents MMCU.

En quatrième lieu, nous avons d'abord présenté l'implémentation notre système en utilisant différents outils de développement (Annexe A). Ensuite, nous avons illustré les différentes fonctionnalités de notre plateforme à travers les interfaces graphique. Après, nous avons testé notre solution dans un environnement qui contient deux machines physiques représentant toutes les entités de la plateforme (un conférencier qui crée une session webinaire, dix participants, un administrateur qui crée trois MCU, et NetEm qui perturbe les paramètres QoS dans les 7 tests). Les résultats obtenus ont montré le bon fonctionnement de la plateforme notamment le processus de sélection à base de QoS. Faute du temps, nous n'avons pas implémenté la fonction de calcul de la QoE. Aussi, il fallait aussi faire les tests sur plusieurs machines (au moins une machine par utilisateur) pour montrer l'efficacité du processus de sélection proposé.

Enfin, tout travail a ses avantages et ses inconvénients et en voici quelques points qui exposent les principales améliorations :

- Concernant le routage de flux multimédia, il serait intéressant d'ajouter d'autres critères qui affectent la QoE à l'exception des paramètres réseaux, tels que le type de terminal et les facteurs psychologiques et de tester d'autres modèles de mesures de QoE.
- Enregistrement des webinaires déjà présentés pour qu'ils soient accessibles plus tard.
- Adapter l'application au web.
- Ajouter le partage d'écran entre le conférencier et les participants.

## Bibliographie

- [1] BENBITAT Hind, CHEURFA Ihsan. Gestion des canaux de communication pour les webinaires inter-universités. Master's thesis, Université Saad Dahlab, Algeria, 1-78 2018.
- [3] Timothy BURK. Techniques de diffusion vidéo sur l'internet streaming avec RTP/RTSP. ENS Lyon, type de doc, 1-9, 2004.
- [4] Xiangyang Zhang and Hossam Hassanein. Video on-demand streaming on the internet—a survey. 25th Biennial Symposium on Communications pages 88–91. IEEE, 2010.
- [5] Antoine Lavignotte. Prise en compte de la qualité de l'expérience utilisateur au sein des protocoles de streaming HTTP adaptatifs. PhD thesis, Université de Jean Monnet, Saint-Etienne 1 - 205, 2014.
- [6] Stefano Petrangeli, Dries Pauwels, Jeroen Van Der Hooft, Matúš Žiak, Jürgen Slowack, Tim Wauters, and Filip De Turck. A scalable webrtc-based framework for remote video collaboration applications. *Multimedia Tools and Applications*, 78(6) :7419–7452, 2019.
- [7] Venkesh Kandari. Implementation and performance optimization of webrtc based remote collaboration system. Faculty of Computing at Blekinge, Institute of Technology, Karlskrona-Sweden, 1-19, 2016.
- [8] Mohamed Taib Benisse. Transmission média sur les réseaux IP en utilisant les protocoles SIP et IAX. PhD thesis, École supérieure de technologie, 17-19, 2009.
- [11] Van Jacobson, Ron Frederick, Steve Casner, and H Schulz Rinne. Rtp: A transport protocol for real-time applications. Article, Columbia university, pages 12 - 16, 2003.
- [12] M Watson. Network working group c. jennings request for comments: 3325 cisco systems categories: Informational j. peterson neustar, inc. Article, 1-15, 2002.
- [13] C Perkins. Network working group m. handley request for comments: 4566 ucl obsoletes:2327, 3266 v. jacobson category: Standards track packet design. Article, Glasgow, 1-49, 2006.
- [17] McDonald van der Merwe and Dalize van Heerden. Webinars: Ease of use and usefulness for disadvantaged students in a low bandwidth odl environment. In *International Conference on e-Learning*, page 445, 2012.
- [19] Hao Wang. Skype voip service-architecture and comparison. In *INFOTECH Seminar Advanced Communication Services (ASC)*, lieu, page 4, 2005.

- 
- [20] Yang Xu, Chenguang Yu, Jingjiang Li, and Yong Liu. Video telephony for end-consumers: measurement study of google+, ichtat, and skype. In Proceedings of the 2012 Internet Measurement Conference, pages 371–384. ACM, 2012.
- [21] Robert Bestak and Jiri Hlavacek. A videoconferencing system based on webrtc technology. In International Conference on Computer Networks, pages 245–255. Springer, 2017.
- [22] Sana AROUSSI. Méta-Routage basé sur la Qualité d’Expérience dans le cadre des Appli- cations Vidéo. PhD thesis, Ecole Nationale Supérieure d’Informatique, Oued Smar, Algérie, pages 1-151, 2012.
- [23] Ktari, Salma, Stefano Secci, and Damien Lavaux. "Bayesian diagnosis and reliability analysis of Private Mobile Radio networks." IEEE Symposium on Computers and Communications (ISCC), pages 1245-1250, IEEE, 2017.
- [25] Razvan Florin Beuran. Mesure de la qualité dans les réseaux informatiques. PhD thesis, Saint-Etienne Politehnica university and Jean Monnet University, București and Saint. Etienne, pages 1-233, 2004.
- [26] Ktari, Salma, Stefano Secci, and Damien Lavaux. "Bayesian diagnosis and reliability analysis of Private Mobile Radio networks." IEEE Symposium on Computers and Communications (ISCC). Paris, pages 1-6, IEEE, 2017.
- [28] Hyun Jong Kim, Dong Hyeon Lee, Jong Min Lee, Kyoung Hee Lee, Won Lyu, and Seong Gon Choi. The qoe evaluation method through the qos-qoe correlation model. In Fourth International Conference on Networked Computing and Advanced Information Management, lieu, volume 2, pages 719–725. IEEE, 2008.
- [29] Nadia Ghiata and Marius Marcu. Measurement methods for qos in voip review. In 3rd International Congress on Ultra- Modern Telecommunications and Control Systems and Workshops (ICUMT), lieu, pages 1–6. IEEE, 2011.
- [30] Neil Davies, Judy Holyer, and Peter Thompson. A queueing theory model that enables control of loss and delay at a network switch. Article, pages 1-12, 1999.
- [31] Philippe Owezarski. Métrologie de la qualité de service. Ecole d’été temps réel (ETR2003) - Systemes, Réseaux et Applications, Toulouse, pages 1-10, 2003.
- [32] ITI ITU. Quality of experience requirements for iptv services. Article, pages 1-35,2008.

- [33] Lahouari Benyettou. Mise en œuvre d'un modèle de corrélation qos-qoe. Article, pages 1 - 15, 2012.
- [34] Wael Cherif. Adaptation de contexte basée sur la Qualité d'Expérience dans les réseaux Internet du Futur. PhD thesis, Université de Rennes 1, Bretagne, pages 1-102, 2013.
- [35] Minhas, Tahir Nawaz, and Markus Fiedler. "Quality of experience hourglass model." International Conference on Computing, Management and Telecommunications (ComManTel). Karlskrona, Sweden, pages 1-6, IEEE, 2013.
- [37] Monalisa Sarma, Debasish Kundu, and Rajib Mall. Automatic test case generation from uml sequence diagram. In 15th International Conference on Advanced Computing and Communications, Inde, pages 60–67. IEEE, 2007.
- [38] Daniela Berardi, Diego Calvanese, and Giuseppe De Giacomo. Reasoning on uml class diagrams. Artificial intelligence, 168(1-2) :70–118, 2005.
- [41] Dalibor D Dvorski. Installing, configuring, and developing with xampp. Article, Skills Canada, 2007.
- [47] A. Jurgelionis, J. P. Laulajainen, M. Hirvonen, and A. I. Wang, "An empirical study of NetEm network emulation functionalities," Proc. - Int. Conf. Comput. Commun. Networks, lieu, page, 2011.

## Webographie

- [2] <https://doc.ubuntu-fr.org/streaming>.
- [9] <https://www.lemagit.fr/definition/Transmission-Control-Protocol-TCP>
- [10] <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203409-udp- user-datagram-protocol-definition-traduction/>.
- [14] <https://management.savoir.fr/communication/#Definition de la communication>.
- [15] [www.ccimp.com](http://www.ccimp.com).
- [16] <https://www.01net.com/actualites/lorganisation-dune-web-conference-150339.html>
- [18] <https://www.skype.com/fr/>
- [24] <https://tools.ietf.org/html/rfc3644>.
- [36] <https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-cas-utilisation>
- [39] <https://www.lemagit.fr/definition/Environnement-de-Developpement-Integre-IDE>
- [40] <https://www.techno-science.net/definition/5346.html>
- [42] <https://projetsdiy.fr/mosquitto-broker-mqtt-raspberry-pi/>
- [43] [https://www.lemagit.fr/definition/MQTT?fbclid=IwAR0eukjNkhXQ0rSWNDNtEq\\_vmUFhD0ov4v32rGNnIHWa7TjY-8A9Rs2VIT8](https://www.lemagit.fr/definition/MQTT?fbclid=IwAR0eukjNkhXQ0rSWNDNtEq_vmUFhD0ov4v32rGNnIHWa7TjY-8A9Rs2VIT8)
- [44] [www.lemagit.fr](http://www.lemagit.fr)
- [45] <https://www.futura-sciences.com/tech/definitions/internet-java-485/>
- [46] <http://dictionnaire.sensagent.leparisien.fr/Langage%20de%20requ%C3%AAte/fr-fr/>

## Annexe A

### Logiciels et langages de développement

#### 1. NetBeans

NetBeans est un environnement de développement intégré (IDE), placé en open source par Sun en juin 2000 sous licence CDDL (Common Development and Distribution License) et GPLv2. En plus de Java, NetBeans permet la prise en charge native de divers langages tels le C, le C++, le JavaScript, le XML, le Groovy, le PHP et le HTML, ou d'autres (dont Python et Ruby) par l'ajout de greffons téléchargeable du site officiel [40].

NetBeans constitue par ailleurs une plateforme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)).

#### 2. Serveur MySQL (XAMPP)

XAMPP signifie Cross Platform, Apache, Maria DB (MySQL), PHP, Perl. XAMPP est une solution de pile de serveurs web multiplateformes gratuite et open source développée par Apache Friends, composée principalement du serveur HTTP Apache, de la base de données Maria DB et d'interprètes de scripts écrits en PHP et en Perl [41].

#### 3. Mosquitto broker

Mosquitto est un serveur MQTT Open Source (Broker) que l'on peut installer sur un Raspberry Pi mais aussi sur presque toutes les plateformes (MacOs, Windows, Linux...). MQTT facilite la communication entre objets connectés (M2M) tout en économisant la batterie [42].

#### 4. MQTT

MQTT (Message Queue Telemetry Transport) est un protocole de messagerie léger qui facilite la transmission des données de télémétrie pour les clients réseau dont les ressources sont limitées. Ce protocole, qui suit un modèle de communication de type

publication/abonnement (publish/subscribe) via un broker de messages, est utilisé pour les échanges machine-à-machine (M2M) [43]. Le broker pousse ces informations vers les clients qui se sont précédemment abonnés. Pour l'utilisateur, un sujet ressemble à un chemin hiérarchique. Les clients peuvent s'abonner à un niveau spécifique de la hiérarchie d'un sujet ou à plusieurs niveaux s'ils utilisent un caractère générique [44].

Une session MQTT est divisée en quatre étapes : connexion, authentification, communication et terminaison. Un client commence par créer une connexion TCP/IP vers le broker en utilisant soit un port standard, soit un port personnalisé défini par les opérateurs du broker. Lors de la connexion, le serveur peut continuer une ancienne session s'il reconnaît une identité client précédemment utilisée. Lors de cet échange, le client peut également fournir au broker un certificat client que le broker pourra ensuite utiliser pour authentifier le client. Bien que cela ne soit pas spécifié dans la norme MQTT, les brokers prennent habituellement en charge l'authentification des clients avec leurs certificats SSL/TLS [44].

## 5. Java

Pour implémenter notre système, nous avons opté pour le langage de programmation java. En effet, Java s'annonce comme une des évolutions majeures de la programmation. Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, il permet de développer des applications client-serveur. Côté client, les applets sont à l'origine de la notoriété du langage. C'est surtout côté serveur que Java s'est imposé dans le milieu de l'entreprise grâce aux servlets, et aux JSP (Java Server Pages) [45].

## 6. Langage des requêtes SQL

Un langage de requête est un langage informatique utilisé pour accéder aux données d'une base de données ou d'autres systèmes d'information. Il permet d'obtenir les données vérifiant certaines conditions (on parle de critères de sélection), comme toutes les personnes qui habitent une ville donnée. Les données peuvent être triées, elles peuvent également être regroupées suivant les valeurs d'une donnée particulière (par exemple on va regrouper toutes les personnes qui habitent la même rue) [46].

## Annexe B

### Logiciel de Test : NetEm

Netem, Un émulateur réseau permet d'imiter les fonctions d'un réseau étendu. Il existe de nombreux émulateurs tels que NetEm, dummynet, NIST Net ou encore ALTQ. Nous avons choisi l'émulateur réseau Netem (Network Emulator) parce qu'il est gratuit, open source, le plus utilisé, riche en documentation et en fonctionnalités offertes [47].

NetEm émule un réseau étendu en donnant la possibilité de réguler soi-même les paramètres de ce réseau comme le délai, la perte de paquets, la duplication de paquets, la corruption de paquets, le ré-ordonnancement de paquets et le contrôle de débit. Par ailleurs, il permet aussi de spécifier l'interface réseau à travers laquelle le trafic sera filtré en entrée et/ou en sortie. Il est très facile à déployer car il est implémenté comme une discipline de file d'attente sous Linux et peut être activé par l'utilitaire Linux « tc » [47].

La figure 1 présente l'exemple d'une instruction NetEm permettant de générer 1% de perte avec 100 ms de latence sur les paquets émis par l'interface eth0. La figure 2 décrit un ensemble d'instructions NetEm permettant de générer les mêmes perturbations réseau (1% de perte avec 100 ms de latence) mais sur les paquets entrants de l'interface eth0.

```
# tc qdisc add dev eth0 root netem delay 100ms loss 1%
```

Figure 30. Exemple 1 du NetEm [47].

```
# modprobe ifb

# ip link set dev ifb0 up

# tc qdisc add dev eth0 ingress

# tc filter add dev eth0 parent ffff: protocol ip u32 match u32 0 0 flowid
1:1 action mirred egress redirect dev ifb0

# tc qdisc add dev ifb0 root netem delay 100ms loss 1%
```

Figure 31. Exemple 2 du NetEm [47].