
COO / POO
TRAVAUX DIRIGES & PRATIQUES

**SYSTEME DE CLAVARDAGE DISTRIBUE
INTERACTIF MULTI-UTILISATEUR TEMPS REEL**

Cahier des Exigences Utilisateur



TABLE DES MATIERES

1 INTRODUCTION 3

1.1 Mission du système 3

1.1.1 A qui rend service le système ? 3

1.1.2 Sur quoi agit le système ? 3

1.2 Condition d'utilisation du système 4

1.3 Documents de référence 4

2 EXIGENCES FONCTIONNELLES 5

2.1 Fonctionnalités relatives à l'agent 5

2.1.1 Fonctionnalités d'administration de l'agent 5

2.1.2 Fonctionnalités d'utilisation de l'agent 5

3 AUTRES EXIGENCES 6

3.1 Exigences opérationnelles 6

3.1.1 Exigences de performances 6

3.1.2 Exigences de ressources 7

3.1.3 Exigences de sûreté de fonctionnement 7

3.1.4 Exigences particulières 7

3.2 Exigences de développement 7

3.2.1 Management et gestion du projet 7

3.2.2 Exigence d'ergonomie 8

3.2.3 Exigence de qualification 8

4 ANNEXE- ELEMENTS METHODOLOGIQUES A DESTINATION DES ETUDIANTS (TD6 A TD11) 9

Version	Date	Auteur/s	Description
0.0	29/03/17	AV & DE	Rédaction initiale (idée)
1.0	09/06/17	DE & EE	Chapitres introduction (besoin)
1.1	12/07/17	FF & EF	Intégration suite à discussion client
1.2	15/09/17	FF & DE	Correction missions (clarification)
1.3	10/10/17	AV & EE	Intégration revues client défense
1.4	28/10/17	EF & AV	Révision des besoins (finalisation)
2.0	29/10/17	EE & AV	Version révisée livrée fournisseur

1 Introduction

1.1 Mission du système

Il s'agit d'un système de communication qui sert de support aux équipes et groupes de l'entreprise afin de leur permettre d'accroître leur efficacité naturelle. Le système utilise un(des) agent(s) afin d'accomplir sa mission. Il est dépourvu d'une architecture centralisée.

Le système observe l'environnement de l'utilisateur, son emploi du temps et ses habitudes d'utilisation, pour détecter des conditions particulières et déclencher des alertes. Il traite les données récoltées, de leur acquisition brute jusqu'à l'obtention de données synthétiques.

D'autre part, en tant que relais de communication, le système permet aux intervenants de se coordonner par l'échange de messages textuels entre eux mais également des illustrations sous la forme d'images, des schémas en plus de permettre l'échange de fichiers de travail.

1.1.1 A qui rend service le système ?

L'utilisateur du système est un organisme classique représentant un groupement de personnes qui ont un intérêt commun et qui collaborent entre eux afin d'arriver à leur fin. Du point de vue de l'utilisateur le système est utilisé comme un service et lui facilite la vie. En effet, un administrateur de l'organisme qui souhaite utiliser le système réalise l'installation des agents sur les postes des personnes amenées à interagir et, après une configuration minimale du poste de travail, le système est fonctionnel.

Grâce à sa fonction de support aux tâches de communication, le système rend indirectement un service indispensable à la population de l'organisme classique représentant un groupement de personnes qui ont un intérêt commun et qui collaborent entre eux afin d'arriver à leur fin.

Le système est aussi utilisé par les experts et les techniciens chargés de l'administrer et de le déployer sur les postes des personnes amenées à l'utiliser afin de réaliser plus efficacement leur mission.

Finalement, en tant que relais de communication, le système est un élément clé pour l'ensemble des utilisateurs.

1.1.2 Sur quoi agit le système ?

Le système doit permettre un déploiement dans n'importe quelle région du monde. La présence d'un réseau local au sein de l'environnement de déploiement est requis. De la

même façon, l'ensemble des participants éligibles au clavardage par l'intermédiaire de ce système devront avoir leur poste de travail situé sur le dit réseau local.

D'autre part, le système a besoin d'établir une liaison avec le réseau de télécommunications afin de servir de relais de communication. Le système devra pouvoir être déployé rapidement et sa mise en route devra être rapide sans gêner les autres systèmes déjà existants déployés au sein de la structure.

1.2 Condition d'utilisation du système

Tout d'abord **un système démonstrateur est développé**. Ce démonstrateur réalisera un test opérationnel en situation simulée. **Le but de la démonstration est de prouver les capacités opérationnelles du système.**

Une fois le succès du système démontré, une ou plusieurs versions améliorées vont être mises en production selon la demande des clients.

Le démonstrateur est considéré comme le premier exemplaire opérationnel à viabilité minimale.

La phase finale du cycle de vie de l'agent concerne le retrait de ce dernier. Il a lieu dès lors que son utilisation en situation n'est plus requise, sur analyse du comité exécutif et action de l'administrateur.

Le système est déployé sans restrictions du système d'exploitation et offre une série de fonctionnalités de base accessible par plusieurs interfaces homme machine qui pourront être étendues par des modules futurs développés séparément et déployés eux même par l'administrateur le moment venu. Aucune configuration autre que celle propre au module déployé ne doit être requise.

1.3 Documents de référence

- Effective Java (2nd Edition): Joshua Bloch, ISBN 8601404288357
- UML et les Design Patterns : Craig Larman
- UML 2 par la pratique: études de cas et exercices corrigés, Pascal Roques

2 Exigences Fonctionnelles

2.1 Fonctionnalités relatives à l'agent

2.1.1 Fonctionnalités d'administration de l'agent

[CdC-Bs-1] Le système doit pouvoir être déployé sur un poste de travail fonctionnant sur le système d'exploitation Windows

[CdC-Bs-2] Le système doit pouvoir être déployé sur un poste de travail fonctionnant sur le système d'exploitation Linux

[CdC-Bs-3] Le système doit pouvoir être déployé sur un poste de travail fonctionnant sur le système d'exploitation OS X

[CdC-Bs-4] Le système doit pouvoir être déployé sur un poste de travail fonctionnant sur le système d'exploitation Android

[CdC-Bs-5] Le déploiement du système devra se limiter à la copie sur le poste de travail d'une série de fichiers et la création d'un raccourci pour l'utilisateur

[CdC-Bs-6] La taille globale de l'ensemble des ressources composant le système ne devra pas excéder 50 Méga-Octets sous une forme non compressée quel que soit le système d'exploitation sur lequel il est déployé

2.1.2 Fonctionnalités d'utilisation de l'agent

[CdC-Bs-7] Le système doit permettre à l'utilisateur de choisir un pseudonyme avec lequel il sera reconnu dans ses interactions avec le système

[CdC-Bs-8] Le système doit permettre à l'utilisateur d'identifier simplement l'ensemble des utilisateurs pour lesquels l'agent est actif sur le réseau local

[CdC-Bs-9] Le système doit permettre à l'utilisateur de démarrer une session de clavardage avec un utilisateur du système qu'il choisira dans la liste des utilisateurs pour lesquels l'agent est actif sur le réseau local

[CdC-Bs-10] Le système doit garantir l'unicité du pseudonyme des utilisateurs pour lesquels l'agent est actif sur un même réseau local

[CdC-Bs-11] Tous les messages échangés au sein d'une session de clavardage seront horodatés

[CdC-Bs-12] L'horodatage de chacun des messages reçus par un utilisateur sera accessible à celui-ci de façon simple

[CdC-Bs-13] Un utilisateur peut mettre unilatéralement fin à une session de clavardage

[CdC-Bs-14] Lorsqu'un utilisateur démarre une nouvelle session de clavardage avec un utilisateur avec lequel il a préalablement échangé des données par l'intermédiaire du système, l'historique des messages s'affiche

[CdC-Bs-15] L'utilisateur peut réduire l'agent, dans ce cas, celui-ci se place discrètement dans la barre des tâches sous la forme d'une icône lorsque le système d'exploitation permet cette fonctionnalité

[CdC-Bs-16] Le système doit permettre à l'utilisateur de changer le pseudonyme qu'il utilise au sein du système de clavardage à tout moment

[CdC-Bs-17] Lorsqu'un utilisateur change de pseudonyme, l'ensemble des autres utilisateurs du système en sont informés

[CdC-Bs-18] Le changement de pseudonyme par un utilisateur ne doit pas entraîner la fin des sessions de clavardage en cours au moment du changement de pseudonyme

[CdC-Bs-19]

3 Autres Exigences

3.1 Exigences opérationnelles

3.1.1 Exigences de performances

[CdC-Bs-20] Le déploiement du système doit être réalisable en 2 heures à partir de la prise de décision de déploiement.

[CdC-Bs-21] Le changement de pseudonyme d'un utilisateur doit être visible de l'ensemble des autres utilisateurs dans un temps inférieur à 20 secondes

[CdC-Bs-22] Le temps écoulé entre l'envoi d'un message par un utilisateur et la réception par un autre utilisateur ne doit pas excéder 1 seconde

[CdC-Bs-23] Le système doit permettre la mise en place de 1000 sessions de clavardage simultanées au sein de celui-ci

[CdC-Bs-24] L'agent doit permettre la mise en place de 50 sessions de clavardage simultanées

[CdC-Bs-25] Lorsque la vérification de l'unicité du pseudonyme de l'utilisateur échoue, l'utilisateur doit en être informé dans une période ne dépassant pas 10 secondes

[CdC-Bs-26] Le temps d'apparition des utilisateurs au sein de la liste des utilisateurs pour lesquels l'agent est actif ne doit pas excéder 10 secondes

[CdC-Bs-27] Le système doit permettre un usage simultané par au minimum 100 000 utilisateurs

3.1.2 Exigences de ressources

[CdC-Bs-28] Le système doit avoir une empreinte mémoire inférieure à 100Mo

[CdC-Bs-29] Lors de son exécution, le système ne doit pas solliciter le processeur plus de 1% du temps lorsque la mesure est réalisée sur un intervalle de 5 secondes

[CdC-Bs-30] Le système doit présenter une réactivité normale pour une application de clavardage

3.1.3 Exigences de sûreté de fonctionnement

[CdC-Bs-31] Le système doit garantir une intégrité des messages supérieure à 99,999%

[CdC-Bs-32] Une utilisation normale du système ne doit pas avoir d'impact sur le reste du système

3.1.4 Exigences particulières

[CdC-Bs-33] Le système doit permettre une extension simple des fonctionnalités par l'utilisation d'un système de modules qui fera l'objet d'une standardisation

3.2 Exigences de développement

3.2.1 Management et gestion du projet

[CdC-Bs-34] Les normes suivantes sont applicables aux travaux relatifs au projet :

- a. ISO 15288 (Ingénierie Système)
- b. ISO 9001 (Gestion de la Qualité)
- c. ISO 14001 (Environnement)
- d. ISO 10007 (Gestion de la Configuration)

[CdC-Bs-35] Un plan d'Assurance Qualité devra être fourni par le fournisseur de solution

[CdC-Bs-36] L'ensemble de la documentation produite ainsi que les artefacts devront être gérés en configuration pendant le cycle de vie.

[CdC-Bs-37] Le fournisseur devra fournir la preuve des procédures définies dans son plan d'assurance qualité.

[CdC-Bs-38] Le fournisseur devra fournir un plan de développement du système.

[CdC-Bs-39] Le fournisseur devra faire état de ses vérifications et tests.

[CdC-Bs-40] Le cycle de développement doit contenir des jalons auxquels le client doit être impliqué afin de valider les activités effectuées.

[CdC-Bs-41] Le fournisseur devra proposer une analyse du soutien logistique lors de la réponse à l'appel d'offre.

3.2.2 Exigence d'ergonomie

[CdC-Bs-42] Le système doit répondre aux standards d'ergonomie de chacune des plateformes pour lesquelles il est conçu.

3.2.3 Exigence de qualification

[CdC-Bs-43] Une version future du système pourra faire objet d'une certification en vue de son utilisation dans le milieu aéronautique.

4 Annexe- Eléments méthodologiques à destination des étudiants (TD6 à TD11)

TD6/ Diagramme des cas d'utilisation

En partant du cahier des charges et de la description qui y est faite du produit attendu, proposer une analyse et une modélisation du contexte en identifiant les principaux cas d'utilisation du système.

Vous pourrez vous aider de la démarche suivante dans votre analyse :

- Identifier les limites du système (qu'est-ce qui est dedans, qu'est-ce qui est dehors)
- Identifier les principaux services rendus par le système, les liens entre ces services
- Identifier les utilisateurs (à qui le système rend-il service ?)

Avant de procéder à une modélisation sous TAU, vous pouvez commencer sur papier.

TD7/ Diagramme de Séquence

Sur base du travail réalisé au TD6, il est proposé de pousser l'analyse un peu plus en procédant en deux étapes.

1/ Une première étape nommée « boîte noire » pour laquelle le système (ou les principaux sous-systèmes dans le cas d'un système complexe) sont vus dans leur ensemble, sans que leur structure interne ne soit présentée.

2/ Une seconde étape consistant à enrichir la vue « boîte noire » pour construire une vue « boîte blanche » du système et des principaux sous-systèmes au travers de laquelle les interactions entre les composants internes au système sont identifiées

Vous pourrez vous aider de la démarche suivante dans votre analyse :

- Pour chacun des use-cases du TD6, quelles sont les interactions requises entre l'utilisateur et le système afin de permettre leur réalisation ?
- Pour chacune des interactions identifiées, comment celle-ci se réalise-t-elle ? Est-elle synchrone ? Asynchrone ?
- Quelles sont les données nécessaires au système pour réaliser les services ? D'où proviennent ces données ? A quel moment et sous quelle forme ces données sont-elles acquises par le système ?
- Quels sont les principaux sous-systèmes ?
- Parmi les « composants » identifiés, lesquels sont des composants « actifs » ?

Avant de procéder à une modélisation sous TAU, vous pouvez commencer sur papier. L'objectif du TD est d'arriver à une modélisation d'un diagramme de séquence boîte blanche pour chacun des cas d'utilisation identifiés au TD6.

TD8/ Diagramme de Classes

La modélisation que vous avez réalisée au TD7 vous a permis d'identifier les principaux composants de votre système dans une approche « boîte blanche ». Nous allons maintenant nous intéresser plus formellement aux liens qui doivent exister entre ces composants afin de rendre les services que vous avez identifiés.

Construisez le diagramme de classes pour les classes principales qui composent votre système en identifiant pour chacune d'elles, leurs caractéristiques principales ainsi que les attributs et méthodes qui les caractérisent. Réfléchissez également à la visibilité de chacun de ces attributs ainsi que des classes elles-mêmes.

Vous pourrez vous aider de la démarche suivante dans votre analyse :

- Commencez par identifier les classes actives et leurs caractéristiques
- Pour chacune des classes que vous identifiez, posez-vous la question, qui construit les objets de ce type ? Qui est responsable de la destruction de ceux-ci ?
- Identifiez les liens entre les classes, faites apparaître les principaux sur le diagramme.
- Est-ce que vos diagrammes de séquence boîte blanche sont toujours d'actualité ? Doivent-ils être modifiés / enrichis pour garder la cohérence ?
- Quelles sont les principales interfaces ? Comment seront réalisées les interactions entre vos objets ? Seront-elles synchrones ? Asynchrones ? Identifier les principaux messages et les principales méthodes de vos interfaces.

TD9/ Diagramme de Structure Composite

Dans son mode nominal de fonctionnement, identifiez les principales instances des classes que vous avez décrites au TD8. Présentez celles-ci sous la forme d'un diagramme de structure composite en faisant apparaître les liens entre les instances par lesquels sont véhiculés des interactions asynchrones. Identifiez également les interactions entre les instances de vos classes et les éléments extérieurs au système si ils existent. Capturez ces éléments sur un diagramme de structure composite.

Vous pourrez vous aider de la démarche suivante dans votre analyse :

- Quelles sont les principales instances de classes présentes dans le système ?
- Combien d'instances de chacune des classes sont présentes dans le système ?
- Quelles sont les instances responsables de la communication avec l'extérieur du système ?
- Est-ce que votre modélisation jusqu'ici est cohérente du diagramme que vous venez d'ajouter ?

TD10/ Diagramme de Machine à Etats

Une classe passive n'ayant pas de comportement propre, pour chacune des classes actives identifiées au TD8 (votre système comprend au minimum une classe active sans quoi il ne pourrait exister), décrivez son comportement par le biais d'une machine à états. Faites également apparaître les interactions avec les autres composants (synchrones et asynchrones).