

# MEMOIRE

Présenté par

**GASMI Khair Eddine**

Pour l'obtention de diplôme de

**MASTER**

Filière : Informatique

Spécialité : **Systèmes Informatiques Intelligents**

Thème

**Les applications sensibles au contexte au profit des  
personnes dépendantes**

Soutenue le : / / 2020

Devant le Jury composé de :

Qualité	Nom et Prénom	Grade	Université
Président	Mme. GASMI Ibtissem	MCB	Chadli Bendjedid El-Tarf
Rapporteur	Mme. AHMED MALEK Nada	MAA	Chadli Bendjedid El-Tarf
Examineur	Mr. TOUAHRI Djamel Eddine	MAA	Chadli Bendjedid El-Tarf

Année Universitaire : 2020/2021

# REMERCIEMENTS

---

On tient à la fin de ce travail à remercier **ALLAH** le tout puissant de nous avoir donné la foi et de nous avoir permis d'en arriver là.

En préambule à ce mémoire, on souhaitait adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leurs aides et qui ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire.

On tient à remercier sincèrement notre encadreur Mme **HAMDI-AHMED MALEK Nada**, d'avoir accepté de nous encadrer, proposer notre thème et nous avoir guidé tout au long de notre projet de master par ses précieuses directives, pour sa confiance et sa gentillesse, sa disponibilité ainsi pour l'inspiration, l'aide et le temps qu'elle a bien voulu nous consacrer.

On exprime également nos gratitudees à nos examinateurs, qui vont nous honorer en évaluant ce travail....

A l'âme de mon très cher papa TAHER (Rabi yarhmo)

A ma très chère maman AZIZA que dieu la garde et la protège

Pour tout l'amour dont vous m'avez entouré, pour tout ce que vous avez fait pour moi. Je ferai de mon mieux pour rester un sujet de fierté à vos yeux avec l'espoir de ne jamais vous décevoir. Que ce modeste travail, soit l'exaucement de vos vœux tant formulés et de Vos prières quotidiennes.

A mes très chers FRERE

A mes très chères Sœurs

A ma femme SHONA

A mes très chers amis (Sans exception)

En souvenir de nos éclats de rire, des bons moments et nuits blanches.

En souvenir de tout ce qu'on a vécu ensemble.

J'espère de tout mon cœur que notre amitié durera éternellement.

**KHAIR EDDINE**

...

# TABLE DES MATIERES

---

REMERCIEMENTS .....	2
DEDICACES .....	3
TABLE DES MATIERES .....	4
LISTE DES FIGURES.....	6
LISTE DES TABLEAUX.....	8
LISTE DES ACRONYMES .....	9
INTRODUCTION GENERALE.....	10
1. CONTEXTE DU PROJET ET PROBLEMATIQUE .....	10
2. MOTIVATION ET OBJECTIFS.....	11
2. STRUCTURATION DU MEMOIRE.....	11
CHAPITRE 1 : ETAT DE L'ART .....	13
1. INTRODUCTION.....	13
2. DEFINITION DU CONTEXTE.....	13
3. CYCLE DE VIE DU CONTEXTE .....	17
4. CATEGORISATION DU CONTEXTE.....	17
5. DEFINITION DE LA SENSIBILITE AU CONTEXTE .....	18
6. CHAMPS D'APPLICATION .....	20
7. PERSONNES A MOBILITE REDUITE ET LA SENSIBILITE AU CONTEXTE.....	21
8. CONCLUSION .....	22
CHAPITRE 2 : ETUDE DE PROJET.....	23
1. INTRODUCTION.....	23
2. ARCHITECTURE DU SYSTEME.....	23
3. STRUCTURE DU SYSTEME.....	23
4. CONTEXTE DE L'APPLICATION.....	24
5. MODELISATION DU SYSTEME.....	26

6. CONCLUSION .....	34
CHAPITRE 3 : REALISATION.....	35
1. INTRODUCTION.....	35
2. LES OUTILS MATERIELS UTILISES .....	35
3. LES LOGICIELS UTILISES : .....	42
4. IMPLEMENTATION .....	45
5. CONCLUSION .....	47
CONCLUSION ET PERSPECTIVES .....	48
REFERENCES.....	49

# LISTE DES FIGURES

---

Figure 1.1. Application sans contexte	14
Figure 1.2. Application tenant compte du contexte	19
Figure 1.3. Infrastructure d'un modèle de contexte	20
Figure 1.4. Des personnes à mobilité réduite	22
Figure 2.1 Schéma Global Du Système	23
Figure 2.2. Architecture du système	24
Figure 2.3. Contexte de l'environnement de l'exécution de l'application.	24
Figure 2.4. Contexte de développement de l'application.	25
Figure 2.5. Diagramme cas d'utilisation	28
Figure 2.6 diagramme de séquence démarrer le système	28
Figure 2.7 diagramme de séquence arrêter le système	29
Figure 2.8 diagramme de séquence envoyer informations sur la présence	29
Figure 2.9 diagramme de séquence envoyer informations sur la présence 2	29
Figure 2.10 diagramme de séquence contrôler le fonctionnement 1	30
Figure 2.11 diagramme de séquence contrôler le fonctionnement 2	30
Figure 2.12 Classification des acteurs 'actif ou passif' dans le diagramme cas d'utilisation	31
Figure 2.13 Interface packages population	32
Figure 2.14 Modèle global de structure gestionnaire TV et lumière avec les relations entre classes.	32
Figure 2.15 Modèle global de structure du gestionnaire TV et lumière sans ambiguïtés	33
Figure 2.16 Modèle global de structure avec identification du «RealTimeObject ».	34

Figure 3.1. Une carte Arduino de type UNO	35
Figure 3.2. Arduino MEGA et Arduino YUN respectivement	37
Figure 3.3 Schéma descriptive de la carte Arduino UNO	38
Figure 3.4 Driver moteur	39
Figure 3.5 Moteur DC	40
Figure 3.6 Module Bluetooth HC-05	40
Figure 3.7 Récepteur infrarouge.	41
Figure 3.8 Afficheur LCD	41
Figure 3.9 IDE d'Arduino	42
Figure 3.10 IDE Processing	43
Figure 3.11 C++	43
Figure 3.12 PYTHON	44
Figure 3.13 FLASK	44
Figure 3.14 FLUTTER	45
Figure 3.15 Partie déclaration	45
Figure 3.16 Un extrait de code de la fonction listen	46
Figure 3.17 Extrait de code de la fonction loop	46
Figure 3.18 Le prototype raccordé	47

# LISTE DES TABLEAUX

---

Tableau 2.1. Dictionnaire pour notre application.

**Erreur ! Signet non défini.7**

# LISTE DES ACRONYMES

---

<b>ACCORD/UML</b>	UML-based methodology for real-time systems development
<b>DAM</b>	Modèle d'analyse détaillée
<b>ECU</b>	Electronic Control Unit.
<b>PAM</b>	Modèle d'analyse préliminaire
<b>PMR</b>	Personne à mobilité réduite

# INTRODUCTION GENERALE

---

Autrefois, l'idée d'un ordinateur dissimulé à n'importe quel environnement était inimaginable. Aujourd'hui, il semble plausible avec la transformation des ordinateurs de l'époque précédente en petites puces qui nous entourent, et la réussite de les intégrer dans notre quotidien à un point tel que nous les utilisons sans penser consciemment à leurs sujets. Actuellement, notre environnement est truffé de dispositifs informatiques équipés d'une certaine capacité de décision autonome, qu'on les utilise souvent à notre insu, que ce soit dans les lieux publics avec les systèmes de repérage permettant par exemple, la détection des vols et la lutte contre le vandalisme, ou bien dans nos coins et recoins les plus privés.

Que ce soit notre maison anticipant quand les lumières doivent être allumées ou éteintes, quand nous réveiller ou commander notre nourriture préférée du supermarché, une station de transport facilitant les déplacements, ou une chambre d'hôpital aidant à prendre soin d'un patient, il y a de fortes raisons de croire que nos vies vont être transformées dans les prochaines décennies par l'introduction d'une large gamme d'appareils qui équiperont de nombreux environnements avec une forte puissance de calcul.

Ces dispositifs informatiques devront être coordonnés par des systèmes intelligents pour comprendre les événements et le contexte pertinent d'un espace spécifique et de prendre des décisions judicieuses en temps réel, rendant ainsi l'environnement intelligent. Cette convergence de sujets a conduit à l'introduction de la notion de la sensibilité au contexte.

## 1. CONTEXTE DU PROJET ET PROBLEMATIQUE

---

Cette notion de sensibilité au contexte a été adoptée à une grande variété de domaines préexistants de l'informatique ainsi que de l'ingénierie. Étant donné la diversité des applications potentielles, cette relation s'étend naturellement à d'autres domaines de la science, comme l'éducation, la santé, les services sociaux, le divertissement, le sport, etc.

La domotique comme les autres, s'est vu bénéficier de la sensibilité au contexte en le rendant plus intelligent et plus flexible. La domotique ne se date pas à aujourd'hui l'homme a toujours envie d'automatiser sa maison : ouverture et fermeture automatiques des volets, ouverture du portail électrique, gestion du chauffage, gestion de l'éclairage, etc... Ainsi, avant l'ère des smartphones, il était par exemple possible d'activer son chauffage à distance en passant un coup de téléphone à sa maison, ou encore en lui envoyant un SMS. C'était tout à fait réalisable. Seulement, une telle installation était relativement compliquée à mettre en place, et il faut bien

l'avouer, coûteuse. Cette époque a malheureusement laissé des traces, puisque pour beaucoup encore aujourd'hui, domotique rime avec cher et compliqué. Pourtant, ce domaine a énormément évolué, et de nombreuses solutions simples à mettre en place et tout à fait abordables pour le grand public existent.

Mais surtout, la domotique en elle-même a évolué, si bien que le terme de domotique est quelque peu dépassé lui-même : la domotique servait à automatiser sa maison ; aujourd'hui, on parle de domotique 2.0, ou de "maison intelligente", pour bien marquer le fait que ce monde a évolué. Les différents domaines de la maison ne se contentent plus d'être automatisés et pilotables, ils communiquent ensemble, permettant à la maison de réagir selon différents évènements.

Tout particulièrement, les personnes à mobilité réduite seront les plus grands bénéficiaires de ces avancées technologiques. Malheureusement, l'Algérie comme tous les pays sous-développés n'utilisent pas beaucoup les nouvelles technologies, ni elles détiennent d'équipements sociaux importants. Ainsi, il est temps de mettre fin à cette situation dramatique par l'introduction et la mise en place des équipements sociaux, de nouvelles mesures comprenant en compte cette tranche de la société.

## 2. MOTIVATION ET OBJECTIFS

---

En dépit des mesures coercitives jugées peu fiables, et avec les nouvelles technologies, et tout particulièrement, la sensibilité au contexte, les handicaps peuvent aujourd'hui se sentir libres et indépendants.

Mon travail, après étude de fiabilité, et dans un souci de mieux intégrer les personnes à mobilité réduite dans la société, j'ai opté délibérément pour un système sensible au contexte gérant la lumière et la TV au niveau d'une pièce.

## 2. STRUCTURATION DU MEMOIRE

---

Dans ce travail, on a étudié en premier lieu la faisabilité de notre système tout en déterminant ses composants essentiels et en proposant une architecture globale du système espéré. Vient ensuite, la modélisation du système proposé avec la méthode ACCORD-UML. En dernier lieu, par manque de matériels, on a opté pour la création d'un prototype avec la carte Arduino pour prouver l'efficacité et la fiabilité de notre système.

Ce mémoire est structuré comme suit :

1. Une introduction générale définissant le contexte de notre travail. Ensuite, on a posé le problème et les contributions apportées, suivie d'une structuration du mémoire.

2. Deux parties essentielles, dont la première dénommée « l'état de l'art », dévoile en détails la notion de la sensibilité au contexte.

3. La deuxième partie baptisée « étude conceptuelle », constituée de deux chapitres :

Le chapitre 2, est dédié à la conception. Nous présentons les détails de notre démarche, aussi bien l'architecture globale du système que sa modélisation. Le chapitre 3 et le dernier, consiste en la création d'un Prototype sous Arduino.

4. Le mémoire se termine comme à l'accoutumée par une conclusion. Nous indiquons aussi quelques perspectives possibles. Les pistes qui se dégagent concernent aussi bien la poursuite de travaux déjà engagés que des travaux sur de nouvelles problématiques.

# CHAPITRE 1 : ETAT DE L'ART

---

## 1. INTRODUCTION

---

Un système intelligent doit percevoir le contexte de l'environnement pour interagir adéquatement et naturellement avec l'utilisateur. Un système est dit sensible au contexte s'il est capable de capter en permanence le contexte courant et d'en tenir compte pour mieux répondre aux besoins et aux attentes des utilisateurs. Tout d'abord, le contexte est capturé par des équipements spécialisés (caméras, microphones, capteurs biométriques). Les données du contexte sont en général perçues dans leurs formes brutes. Ce qui nécessite des mécanismes de description sémantique du contexte afin de le rendre compréhensible et intelligible par le système. Ces mécanismes se basent essentiellement sur des méta-données décrites dans des langages sémantiques avec un haut niveau d'abstraction des connaissances permettant ainsi de décortiquer le vrai sens du contexte.

La sensibilité au contexte est une caractéristique délicate et indispensable pour un système intelligent. Effectivement, elle reflète la capacité de réagir proprement suite aux modifications du contexte et de changer le comportement de façon dynamique pour aboutir à des services conformes aux attentes des utilisateurs.

Dans une première partie, on présentera une synthèse des définitions données pour l'informatique contextuelle. Ensuite, on donnera un tour d'horizon des utilisations possibles du contexte en présentant des projets de recherches tout en insistant sur l'utilisation du contexte pour les handicapés.

## 2. DEFINITION DU CONTEXTE

---

### 2.1. VERS LA NOTION DE CONTEXTE

Les humains sont assez doués lorsqu'il s'agit de communiquer entre eux, de transmettre des idées, de réagir aux propos d'un interlocuteur. La richesse du discours permet de transmettre des notions très subtiles. Mais deux personnes ou un groupe de personnes communicantes ont aussi une sorte de notion implicite du monde qui les entoure et qui leur permet d'ajouter un cadre au dialogue. L'Homme se sert en permanence, et même sans s'en rendre compte, de tous ces artifices qui viennent enrichir et rendre le dialogue beaucoup plus simple et efficace.

C'est de cette prise en compte de l'environnement au sens large dans nos modes de communications que découle l'idée du contexte.

Son utilisation en informatique est nouvelle. Jusqu'ici, dans les interactions humains machine, l'homme réduit intentionnellement ses mécanismes de communications, en s'adaptant à la pauvreté des moyens de communication, simples périphériques d'entrée sortie dont dispose la machine (ordinateur ou autre). Cet appauvrissement volontaire du contexte rend le dialogue entre homme et machine artificiel, complexe [1].



Figure 1.1. Application sans contexte.

Les applications informatiques sont traditionnellement conçues comme des boîtes noires (figure 1.1), prévues pour réagir à des entrées et produire des sorties sur le mode stimulus - réponse. A une entrée donnée correspond un état de sortie. En fait, ce qui a fait la force de l'informatique a justement été la recherche d'une certaine indépendance au contexte d'utilisation. C'est la notion clef d'abstraction, le comportement d'un système doit être prédictif, quel que soit son cadre d'utilisation [2]. Cette idée a été revue depuis les années 80, avec l'apparition des premières interfaces graphiques, des logiciels reconfigurables comme Emacs mais surtout avec MacOS des premiers Macintosh. Une première prise en compte du contexte d'utilisation a vu le jour dans les logiciels de bureautique par exemple où l'utilisateur pouvait enregistrer des préférences ou définir des modèles de documents. Le système s'en remettait toutefois complètement à l'utilisateur pour la définition de contextes d'utilisation [2]. Les premières utilisations réelles du contexte sont apparues au début des années 1990 avec les premières recherches sur l'informatique mobile ou embarquée dans les laboratoires Olivetti Research, avec le projet Active Badge et au PARC Xerox avec les premières expérimentations d'informatique ubiquitaire. [3]. L'utilisation du contexte a suivi dès lors deux voies parallèles :

➤ L'utilisation du contexte pour améliorer les interactions entre homme et machine du point de vue de l'ergonomie des interfaces et de la connaissance d'informations sur l'utilisateur s'est considérablement développée avec l'arrivée du Web et l'utilisation des Cookies, la complétion automatique des champs de formulaires, la notion d'historique d'utilisation. Les interfaces graphiques suivent le même chemin et se reconfigurent en fonction des fonctions les plus utilisées par l'utilisateur. Les menus contextuels savent proposer une série de fonctionnalités disponibles uniquement dans un contexte donné. Les traitements de textes actuels proposent certaines fonctions de corrections automatiques comme l'accentuation ou la mise en majuscule des têtes de phrases. Les derniers systèmes d'exploitation proposent une gestion personnalisée par utilisateurs, avec des gestionnaires de coordonnées, de mots de passe qui vont pouvoir être

réutilisés par le système pour renseigner des formulaires, envoyer du courrier... Cette approche du contexte se sert uniquement de l'expérience acquise avec l'utilisation ou de modèles contextuels conçus dès le départ (correcteurs automatiques) pour simplifier et guider l'utilisateur, celui-ci doit pouvoir se concentrer sur des tâches de plus haut niveau en laissant le soin au système de factoriser les tâches répétitives [2].

➤ L'informatique mobile avec d'abord l'arrivée des ordinateurs portables puis plus récemment les téléphones cellulaires, les organisateurs et enfin les ultraportables ouvrent un champ d'expérimentations très vaste pour une utilisation du contexte basée non plus uniquement sur les caractéristiques de l'utilisateur mais aussi sur l'environnement géographique, le temps... Le contexte en informatique mobile fait un usage intensif de capteurs qui viennent compléter le système existant. C'est donc un par un enrichissement des appareils que l'on arrive à capter le contexte environnemental physique. L'exemple le plus répandu de l'utilisation du contexte en informatique mobile a été introduit par les ordinateurs portables et les systèmes de mise en veille automatique après une période d'inactivité [4]. L'utilisation de l'environnement physique en informatique mobile est très à la mode ; la localisation géographique de l'utilisateur est beaucoup utilisée. [4]

## 2.2. COMPOSANTS DU CONTEXTE

Les principaux éléments qui peuvent constituer le contexte sont :

- ✚ **L'endroit** : caractérise la position et la situation d'un utilisateur ou d'une entité particulière par rapport aux éléments avoisinants et par rapport à l'espace
- ✚ **L'environnement (ou l'activité)** : est l'ensemble de tous les éléments et tous les événements qui peuvent intervenir lors de l'exécution d'une situation
- ✚ **L'identité** : représente la nature (définition) exacte d'une entité ainsi que sa description comportementale.
- ✚ **Le temps** : décrit le moment (instant) du déroulement d'une tâche

Ces éléments du contexte seront considérés comme des types primaires (premier niveau), et à partir desquels, nous pouvons dériver d'autres types de second niveau qui seront utiles pour mieux décortiquer le contexte d'une entité particulière. Les entités envisagées peuvent être réparties en trois classes :

- ✚ **Les places** : sont des régions de l'espace géographique
- ✚ **Les personnes** : peuvent être sous forme d'individus ou groupes, unies ou réparties
- ✚ **Les objets** : peuvent être des objets physiques ou des composants logiciels.

Les parties constituantes de l'environnement sont [5] :

- ✚ **Environnement informatique** : machines disponibles, processeurs, réseaux, etc.
- ✚ **Environnement de l'utilisateur** : endroit, personnes voisins, situation sociale, etc.
- ✚ **Environnement physique** : niveau de lumière, niveau de son, etc.

### 2.3. CARACTERISTIQUES DU CONTEXTE

Une information contextuelle doit présenter des propriétés de qualité très particulière pour pouvoir être prise en compte d'une manière efficace [6] :

- ✚ **Confiance** : c'est la probabilité pour que le contexte capturé soit correct.
- ✚ **Précision (exactitude)** : représente le pourcentage d'erreur des contextes capturés ou déduits.
- ✚ **Fraîcheur** : c'est l'intervalle de temps qui sépare les différentes interprétations d'un certain type de contexte.
- ✚ **Résolution (détermination)** : c'est un intervalle qui délimite l'information de localisation.

D'autres caractéristiques peuvent être liées au contexte comme celles citées dans [7] :

- ✚ Le contexte doit être abstrait pour avoir une signification claire
- ✚ Les capteurs d'acquisition du contexte peuvent être distribués et hétérogènes
- ✚ Le contexte peut avoir plusieurs représentations alternatives
- ✚ Le contexte a un caractère dynamique (dans la majorité des cas)
- ✚ L'information contextuelle est imparfaite et incertaine

Plusieurs bénéfices sont attendus de la prise en compte du contexte [8] :

- ✚ Désambiguïser l'information issue des inputs primaires,
- ✚ Fiabiliser un système,
- ✚ Augmenter les performances adaptatives d'un système.
- ✚ Amélioration du confort d'utilisation
- ✚ Rendre l'application plus proche d'un agent humain permettant de lui conférer du « bon sens »
- ✚ Prise en compte de la performance des utilisateurs par rapport au système
- ✚ Prendre en compte des besoins latents, même implicites et non formulés par l'utilisateur
- ✚ Adaptation intelligente à l'utilisateur par apprentissage incrémental, apprentissage de préférences des utilisateurs
- ✚ Eviter de demander des entrées d'information inutiles ou redondantes
- ✚ Eviter de présenter des informations en sortie inutiles ou redondantes

- ✚ Fournir des informations en sortie plus pertinentes, adaptées et personnalisées
- ✚ Adapter la présentation des informations aux conditions de consultation (terminaux disponibles)
- ✚ S'adapter à l'environnement de l'utilisateur et à l'utilisateur lui-même, à son activité, à sa situation, à son état, etc.
- ✚ Resituer le contexte distant et enrichir la conscience de l'utilisateur en situation de communication distante (adaptation non automatique).

Le contexte peut être utilisé selon plusieurs façons possibles [8] :

- ✚ Déclenchement ou arrêt automatique du système
- ✚ Adaptation automatique instantanée du système
- ✚ Apprentissage incrémental incorporé dans le système
- ✚ Information annexe fournie en sortie : état du système ou contexte distant
- ✚ Contrôle de l'activité externe du système

### 3. CYCLE DE VIE DU CONTEXTE

---

Le cycle de vie du contexte est composé de cinq étapes fondamentales :

- ✚ **Capture du contexte** : c'est l'acquisition des informations contextuelles nécessaires à l'exécution d'une action. Cette opération est faite à l'aide de capteurs externes (physiques) ou internes (logiciels).
- ✚ **Stockage du contexte** : c'est la mémorisation de toutes les informations brutes capturées pour une éventuelle exploitation immédiate ou future.

### 4. CATEGORISATION DU CONTEXTE

---

Pour bien comprendre et appliquer cette définition du contexte, il est plus simple de passer par une catégorisation des variables du contexte. Selon [9], le contexte se décompose en trois sous-classes où chacune des variables répond à l'une des questions "Où suis-je ?", "Avec qui suis-je ?", "Quelles sont les ressources de mon environnement proche?". [10] catégorisent le contexte en identité de l'utilisateur, ressources de l'environnement proche, localisation de l'utilisateur et période temporelle d'exécution de l'interaction.

On voit apparaître en filigranes de toutes ces catégories, comme le résumant [1], les notions de temps, d'identité, d'activité et de localisation. En fait toute variable contextuelle est utile pour éclaircir une ou plusieurs des questions "Quand ? Où ? Quoi ? Qui ?". Le système ou les concepteurs du système vont utiliser ces informations catégorisées pour répondre au "Pourquoi ?" de l'occurrence d'une situation.

Un utilisateur équipé d'un système contextuel pour, par exemple une visite de ville, et qui va s'approcher d'un monument, va voir apparaître sur ce guide des informations sur le monument. Le concepteur du système répond à la question "Pourquoi l'utilisateur s'approche t il ?" en lui envoyant des informations sur le monument. La variable contextuelle utilisée est la localisation de l'utilisateur mais aussi son environnement, c'est à dire que le monument fait aussi partie du contexte puisqu'il influe sur la réponse du système.

Temps, identité, activité et localisation sont des catégorisations primaires du contexte, ce sont les seules nécessaires pour expliciter une situation. Il est possible de les décomposer en sous variables. De l'identité, on va pouvoir trouver l'adresse, le courriel et d'autres informations personnelles. Ces sous catégories sont en fait considérées comme des attributs des catégories primaires : le numéro de téléphone d'une personne peut-être connue en passant par son identité dans un carnet de numéros. L'utilisation des catégories et des attributs donne une première idée de l'organisation que peuvent avoir des variables du contexte.

## **5. DEFINITION DE LA SENSIBILITE AU CONTEXTE**

---

On dit d'un système qu'il est sensible au contexte s'il peut tirer, interpréter et utiliser des informations issues du contexte et adapter sa réponse en fonction du contexte d'utilisation. Cette définition correspond à peu près à la première définition de la sensibilité au contexte proposé par [11]. Depuis beaucoup d'enrichissements ont été apportés à cette définition. On fait maintenant le distinguo entre les applications qui utilisent le contexte, comme par exemple un service de météo qui aura besoin d'informations de localisation et de temps pour produire un bulletin et d'autres applications qui adaptent leur comportement en fonction du contexte. L'utilisation simple du contexte n'implique pas une modification du comportement du système.

[12] définissent la sensibilité au contexte comme étant la meilleure capacité d'un système à agir en temps réel avec des données provenant du contexte. [12] limite sa définition à l'interface homme machine sans prendre en compte l'application en elle-même. D'autres définitions sont plus orientées vers l'adaptation au contexte : [13] dit d'une application sensible au contexte qu'elle doit automatiquement extraire de l'information ou effectuer des actions en fonctions du contexte utilisateur détecté par les capteurs.

Enfin [1] proposent qu'un système soit sensible au contexte s'il utilise des informations du contexte pour mettre à disposition des informations ou des services utiles à l'utilisateur, l'utilité dépendant de la tache de l'utilisateur. Avec cette définition, une application qui se contenterait d'afficher à l'écran le contenu des variables de contexte est une application sensible au contexte légitime, même si elle ne modifie pas son comportement.

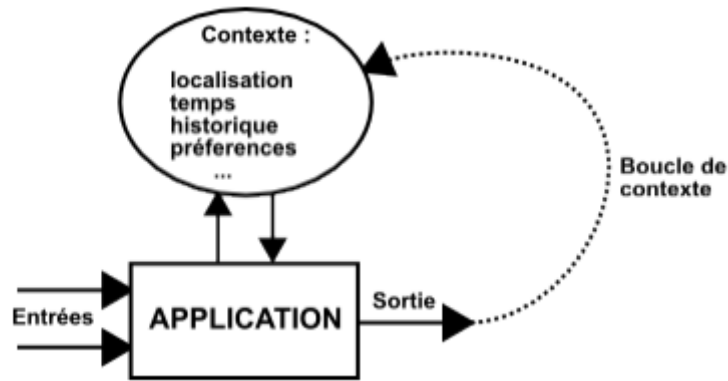


Figure 1.2. Application tenant compte du contexte

[14] complètent le schéma de la boîte noire du système informatique en ajoutant des éléments de contexte qui vont pouvoir agir sur l'application (figure 1.2). De même, il est possible pour l'application d'agir et de modifier le contexte. C'est ce qu'ils appellent la boucle de contexte où la sortie contextuelle de l'application va modifier les valeurs des variables contextuelles. On rencontre dans la littérature beaucoup de synonymes de "sensible au contexte". On parle de systèmes "adaptatifs", "réactifs", "dirigés par l'environnement", "localisés". Toutes ces dénominations sont voisines de la notion de sensibilité au contexte.

Toute la difficulté de la sensibilité au contexte provient de la capture, la représentation mémoire et le traitement des données du contexte. La plupart du temps, il faut équiper les appareils de capteurs supplémentaires. Les données des capteurs doivent être filtrées et traitées avec souvent une utilisation de système d'intelligence artificielle. Enfin, l'utilisation du contexte se fait souvent en mélangeant les données en provenance de plusieurs capteurs. La déduction indirecte d'éléments du contexte est difficile pour un système automatique. Comment par exemple faire comprendre à une application que si trois personnes se rassemblent dans une pièce à un moment donné de la semaine (informations en provenance des capteurs de contexte), il peut s'agir d'une session de travail de groupe hebdomadaire. Cette difficulté à l'abstraction explique que l'on ne trouve pas beaucoup de projets de recherche qui mélangent différentes sources d'information. En pratique, énormément de projets utilisent la localisation comme seule variable de contexte [4].

Dans le cadre de notre travail nous avons repris la définition proposée par [1] et nous avons mis l'accent sur l'aspect contextuel pour répondre aux besoins des utilisateurs. En effet, la sensibilité au contexte exige une capacité de traiter le contexte reçu pour produire un service désiré. Ceci signifie que l'application doit être en mesure de capter des informations (sous forme numérique), de les déchiffrer en chiffres pour les transformer en données symboliques et compréhensibles de haute abstraction.

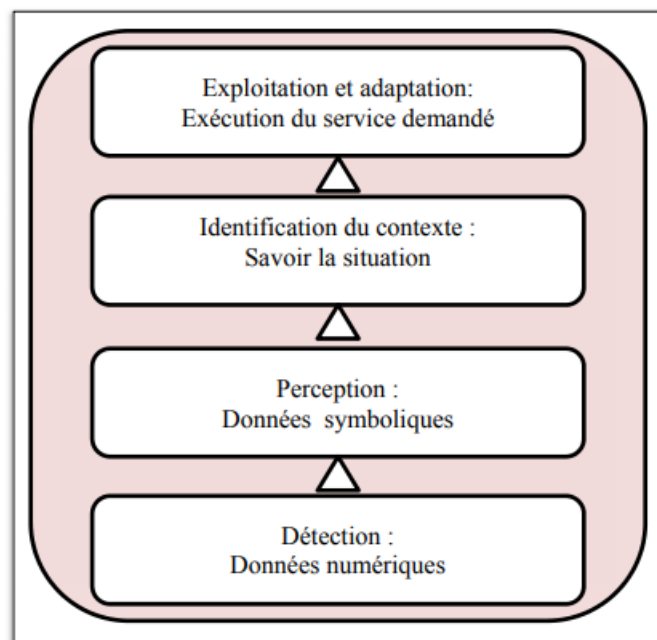


Figure 1.3. Infrastructure d'un modèle de contexte

Une fois le contexte capté, il sera bien identifié par le système et parviendra à déclencher ou à adapter un service. Autrement dit, l'application doit être en mesure de réagir en cas d'un changement de contexte et de s'adapter selon la nouvelle situation. La figure 1.3. illustre les différentes phases passées dans une application sensible au contexte.

## 6. CHAMPS D'APPLICATION

---

La sensibilité au contexte est multidisciplinaire. Elle est appliquée à de nombreux champs de recherche comme : la santé, l'éducation, la sécurité, la vidéo surveillance, la domotique, la gestion de situation de crises, le transport public, les fabriques, les chaînes de production, etc.

La sensibilité au contexte et les technologies mobiles offrent diverses possibilités pour améliorer l'efficacité et l'efficience du traitement médical dans les hôpitaux. Grâce à l'utilisation des capteurs par exemple, ces systèmes offrent un support sensible au contexte. Dans le domaine des soins de santé, les systèmes contextuels peuvent également soutenir une équipe opérationnelle à déterminer si des instruments ont été laissés dans le patient. Ces erreurs évitables conduisent à environ 17 000 décès par an en Allemagne, et les technologies peuvent aider à réduire le risque de complications chez les patients.

On adopte aussi la sensibilité au contexte dans le domaine d'apprentissage et de l'éducation afin de comprendre les besoins et les souhaits des apprenants et des enseignants, de présenter les documents pédagogiques personnalisés de manière omniprésente, de soutenir les interactions « naturelles » avec un contenu d'apprentissage numérique et traditionnelle, ainsi que de suivre la progression de l'apprentissage au niveau individuel et de la classe. De même, on l'utilise pour la

détection des difficultés et des problèmes, et en offrant une aide personnalisée. Aussi on peut l'exploiter pour soutenir les apprenants et les enseignants dans les activités de classe, et d'étendre l'apprentissage au-delà de la salle de classe. De manière similaire, on opte à l'utiliser pour que les étudiants trouvent des personnes avec les mêmes intérêts qu'eux, pour covoyer par exemple, etc.

De plus, trouver des personnes avec les mêmes intérêts ou buts, peut être appliqué à d'autres domaines comme par exemple dans un village ou un lieu particulier comme un centre de loisirs.

On emploie la sensibilité au contexte dans le domaine de vidéosurveillance pour collecter des statistiques sur les personnes, les comportements, les véhicules, etc. Ces statistiques peuvent être diffusées aux citoyens ou aux fonctionnaires publics, ou être utilisées pour des applications de planification. C'est souvent le but des systèmes de gestion du trafic basés sur la vision, tant pour les routes urbaines que les autoroutes, employés pour mesurer les files d'attente, quantifier le taux d'occupation des voies, détecter les incidents, mesurer la vitesse, pour le contrôle d'accès aux zones réglementées, et ainsi de suite.

La sensibilité au contexte est aussi employée dans le tourisme pour la personnalisation et l'adaptation des voyages pour différents touristes. Dans les usines, on l'applique pour le contrôle des chaînes de production, pour augmenter la production en fonction de la demande et pour l'organisation et la sécurité du personnel. De même, on peut l'utiliser pour la prise de décision qui est une charge importante au travail. En effet, elle est basée sur de différents points de vue, d'échange d'idées, d'arguments et elle demande de négocier, coopérer, collaborer et discuter sur le sujet.

La domotique se réfère à l'utilisation de la technologie pour rendre les tâches quotidiennes à la maison (ou au bureau) plus simples, plus sûres, ou moins chères. Les systèmes de domotique peuvent contrôler des lumières et des gradateurs, créer ou améliorer les systèmes de divertissement et de home cinéma et, renforcer la sécurité de la maison, automatiser les serrures et les portes et accroître l'efficacité énergétique.

C'est ce dernier champ qui nous intéresse dans cette étude. Plus précisément, nous focalisons notre intérêt sur la tranche des personnes à mobilité réduite (PMR).

## **7. PERSONNES A MOBILITE REDUITE ET LA SENSIBILITE AU CONTEXTE**

---

Les Personnes à Mobilité Réduite sont toutes les personnes qui ont une difficulté, telles que les personnes handicapées (comprenant les personnes ayant des incapacités sensorielles ou intellectuelles, les personnes ayant des incapacités motrices et les personnes en fauteuils

roulants), les personnes de petite taille, les personnes avec des bagages encombrants, les personnes âgées, les femmes enceintes, les personnes avec des chariots à provisions, et des personnes avec de jeunes enfants(y compris des enfants en poussette).

#### *Les Personnes a Mobilité Reduite*

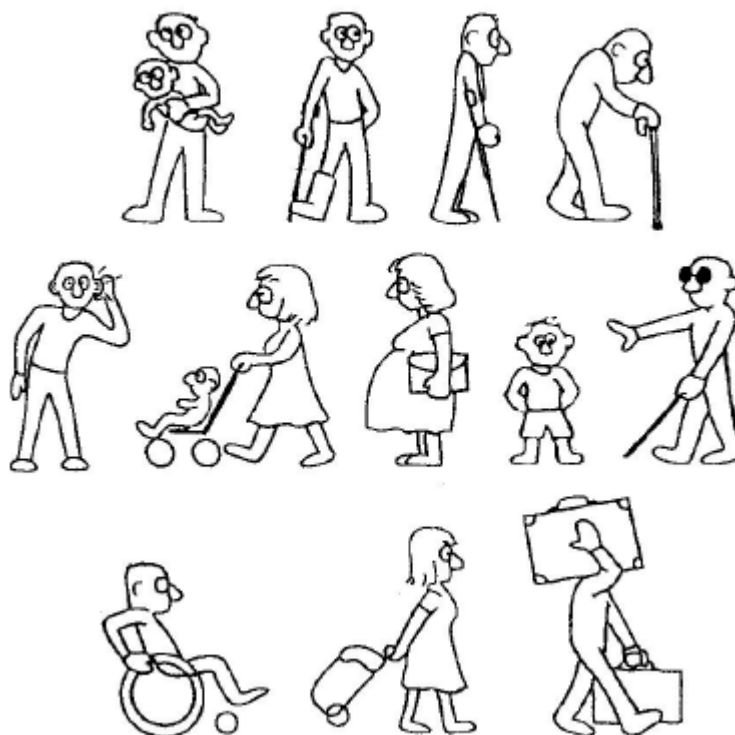


Figure 1. 4. Des personnes à mobilité réduite

Aujourd'hui et grâce à la sensibilité au contexte cette tranche peut bénéficier d'une :

- + Intégration sociale
- + Participation au développement économique

Ainsi que de

- + Passer du statut d'assisté à celui de personne active
- + Limiter le risque d'exclusion, favoriser le rassemblement des générations, garantir l'autonomie

## **8.CONCLUSION**

---

La sensibilité au contexte, cette vision futuriste qui a pour but de faciliter notre vie, couvre différents domaines incluant des applications extrêmement diverses. Mais, elle subit encore des défis qu'on doit leur faire face pour qu'on puisse appliquer cette vision à tous les domaines de notre quotidien.

# CHAPITRE 2 : ETUDE DE PROJET

## 1. INTRODUCTION

Pour aider les gens à mobilité réduite tout en profitant de la sensibilité au contexte, on a proposé un système intelligent pour gérer la lumière ainsi que la télévision à l'intérieur d'une pièce par le biais des dispositifs installés au niveau de cette dernière.

## 2. ARCHITECTURE DU SYSTEME

Dans cette section on va présenter l'architecture globale du système proposé (voir figure 2.1.)

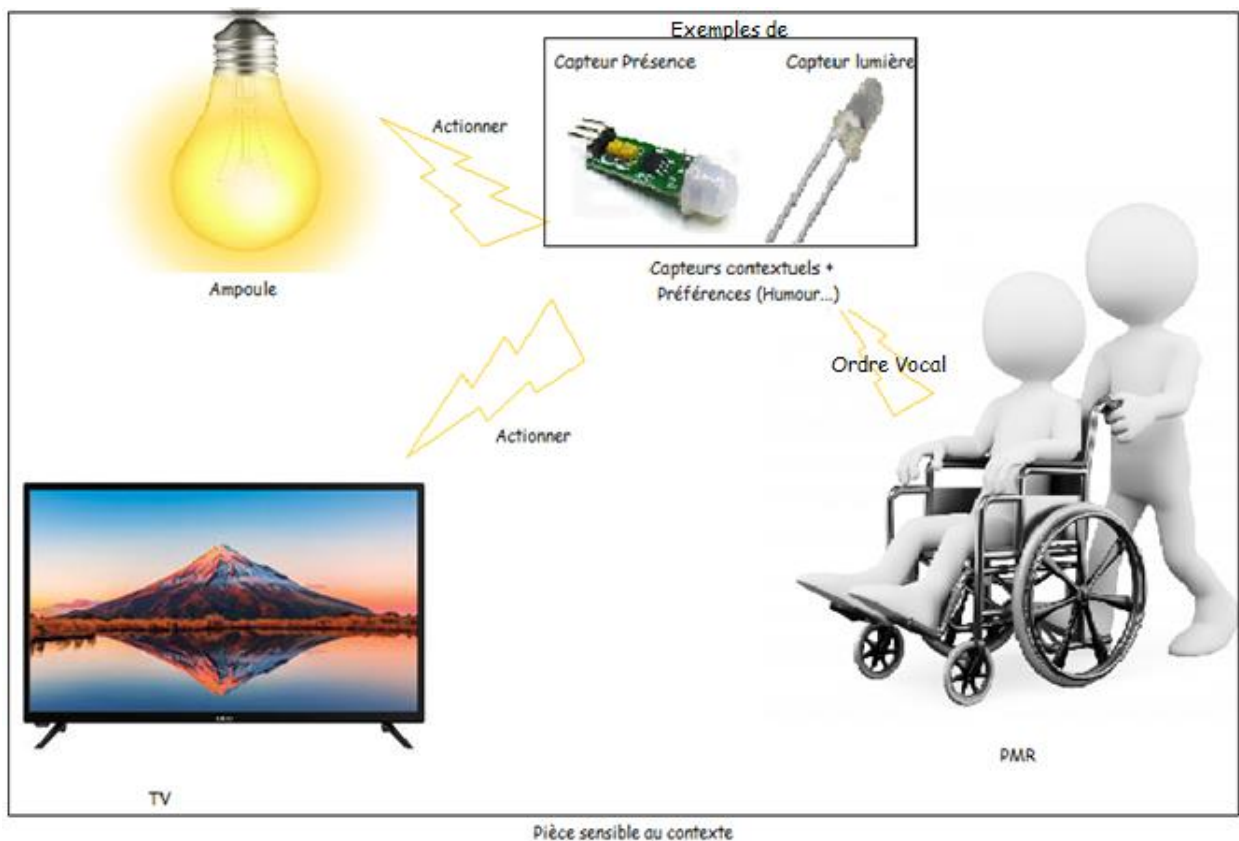


Figure 2.1. Schéma Global Du Système

## 3. STRUCTURE DU SYSTEME

La figure 2.2 représente les composants du système.

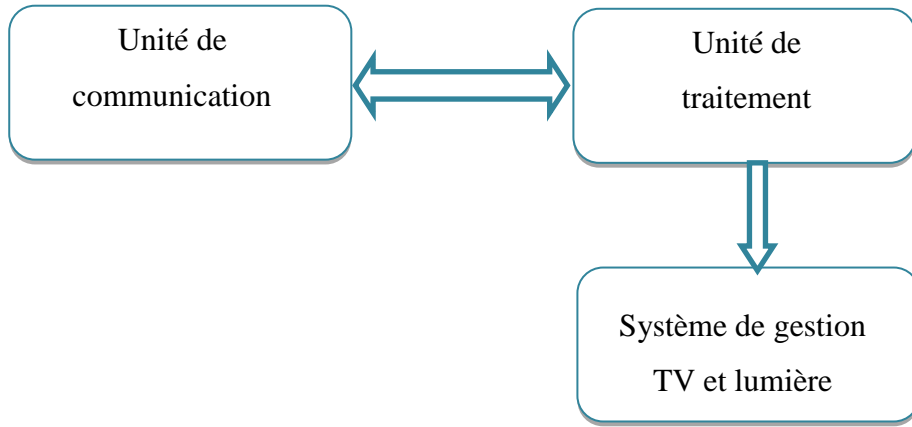


Figure 2.2. Architecture du système

- ✚ **Unité de communication** : qui sert à garantir la communication entre les capteurs et les actionneurs.
- ✚ **Unité de traitement** : cette unité actionne soit la lumière, soit la TV soit les deux après avoir filtrée les données contextuelles capturées ainsi que les préférences de la personne.
- ✚ **Système de gestion de la TV et la lumière** : selon le contexte (la présence d'une personne au niveau de la chambre ainsi qu'il fait sombre) le système allume la lumière, et en quittant la pièce le système éteint la lumière. Selon la situation (par exemple dormant ou éveillé) le système allume et éteint la TV.

#### 4. CONTEXTE DE L'APPLICATION

---

Vu l'importance du contexte dans les applications, on a proposé l'environnement suivant :

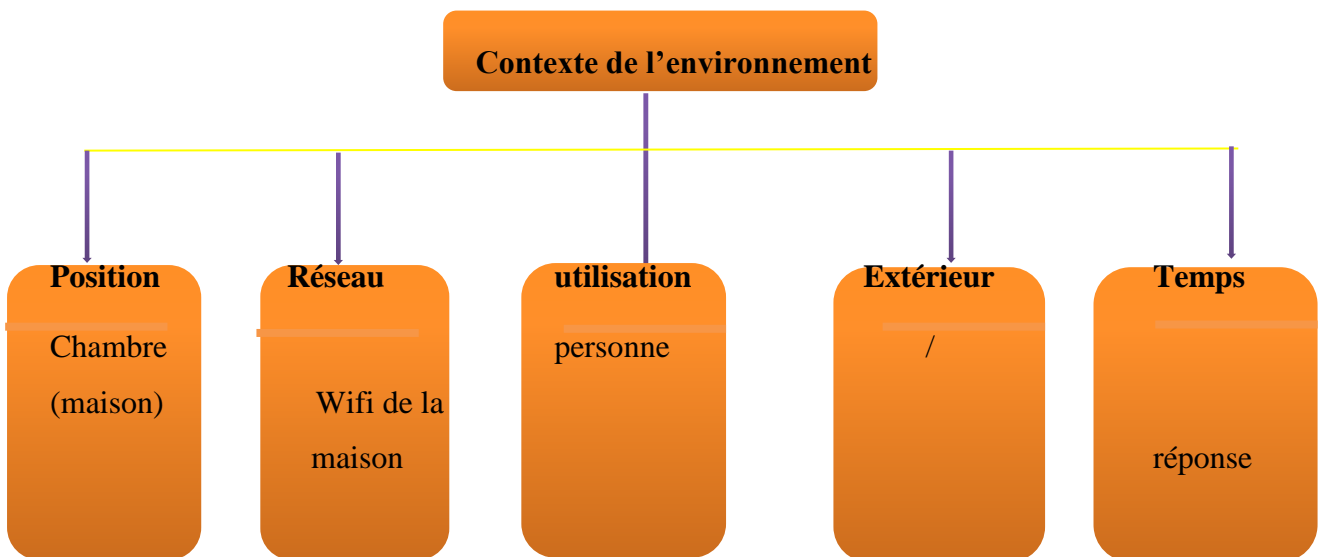


Figure 2.3. Contexte de l'environnement de l'exécution de l'application.

Comme le montre la figure 2.3, le système est utilisé principalement par une personne dans une pièce tout en profitant des réseaux sans fil de la maison pour assurer la communication entre les différents capteurs et actionneurs.

Puisqu'il s'agit d'une application non critique (soft), le système peut répondre avec un léger retard.

Le but de ce modèle de contexte est d'être utilisé dans une architecture logicielle sensible au contexte. Il est également impératif de considérer les éléments liés à la plateforme comme ceux présentés dans la figure 2.4.

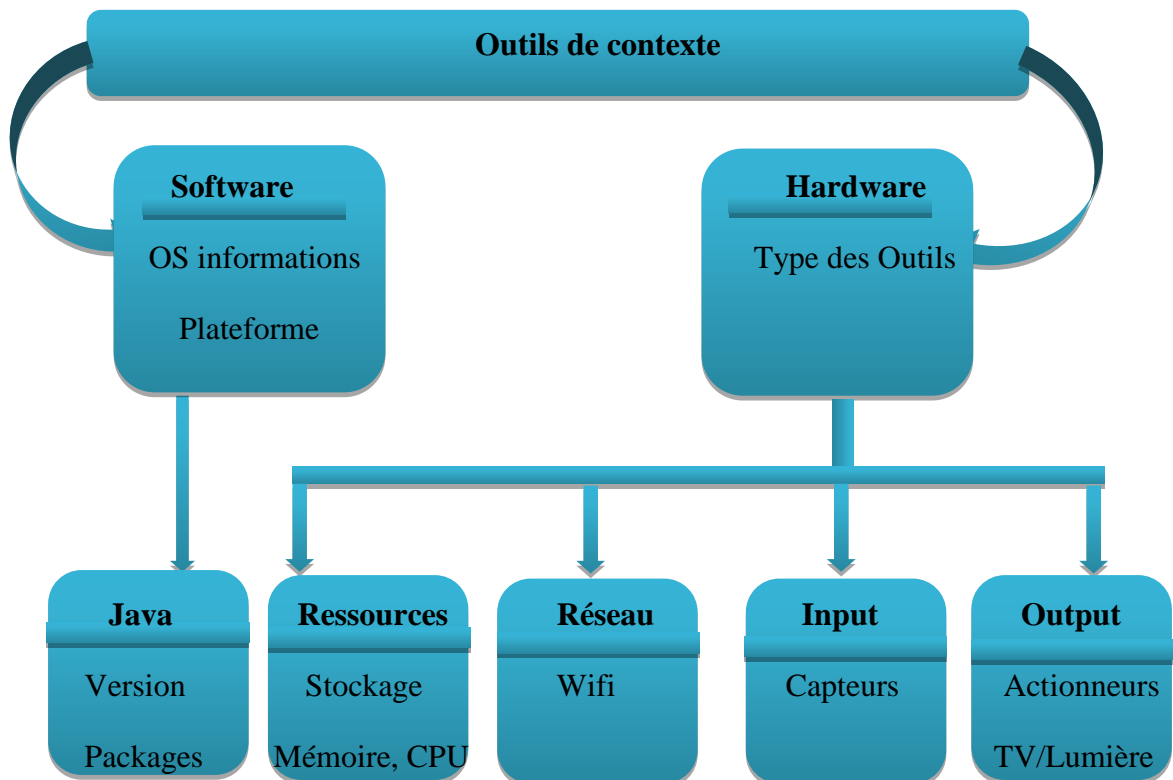


Figure 2.4. Contexte de développement de l'application.

En examinant le software, Java présente plusieurs avantages dont on cite essentiellement sa puissance à la gestion des multithreads qui lui ont conféré une célébrité dans le développement des applications dédiées au transport et son utilisation est devenue très répandue.

En ce qui concerne le Hardware, on a besoin de :

- ✚ Unités de stockage telles que les mémoires pour sauvegarder les informations utiles.
- ✚ CPU pour faire des comparaisons et prendre des décisions
- ✚ Capteurs pour extraire les informations de l'environnement.
- ✚ Les messages relatifs aux modes de fonctionnement, de réglementation etc. ainsi que les effets sonores appropriés.

- ✚ Puisqu'on va utiliser des réseaux Wifi, le standard 802.11 est alors proposé pour supporter l'échange très rapide des données entre les différents composants.

## 5. MODELISATION DU SYSTEME

---

A ce stade de « modélisation du système », on a opté pour la méthode ACCORD/UML [16]. Cette méthodologie conçue au CEA-LIST pour le développement des systèmes temps réels embarqués sert de base pour une modélisation conforme de modélisation UML. Elle s'appuie sur la définition d'un profil UML et définit un ensemble d'artefacts et de règles de modélisation permettant la construction des modèles d'une application [17].

### 5.1. POURQUOI ACCORD/UML ?

Au départ, UML est un langage de modélisation objet généraliste. Il vise le développement de systèmes logiciels en tout genre, de la base de données, aux applications Internet en passant par les systèmes temps-réel embarqués. Cependant pour être utilisé au mieux dans un domaine d'application précis, UML doit être quelque peu adapté. Or cela est possible car UML contient des mécanismes internes permettant de construire des spécialisations de UML pour ses besoins : les stéréotypes, les valeurs marquées (tagged values) et les contraintes. Ce qui a créé un besoin de structuration déterminé par la notion de profil en UML 1.3.

L'intérêt issu de la modélisation par la méthodologie ACCORD/UML basé sur la notion de profil est de :

- ✚ Guider, contrôler et automatiser le développement des modèles.
- ✚ Amélioration de la maîtrise de développement d'un produit.
- ✚ Gain en qualité, sûreté et en fiabilité des produits issus d'un tel développement.

### 5.2. MODELE D'ANALYSE PRELIMINAIRE (PAM)

Cette étape joue un rôle important dans le cycle de développement du projet. Elle vise à préciser les fonctions générales de l'application, en termes très généraux, ainsi que ses interactions avec l'environnement. Au cours de cette étape, les exigences du produit peuvent être reformatées sous forme de texte et de graphiques facilement accessibles, même pour les utilisateurs inexpérimentés. Le PAM est constitué d'un dictionnaire, d'un modèle de cas d'utilisation et d'un modèle de scénario de haut niveau.

- ✚ **Constitution du dictionnaire :** Le dictionnaire produit est une compilation de tous les concepts clés extraits du document initial des exigences qui sont pertinentes pour le domaine et l'intérêt de l'application. Il prend la forme d'une table en différentes colonnes

définissant les catégories de l'analyste de système utilisées pour classer les concepts clés du domaine du document initial des exigences.

Le tableau 1 représente le dictionnaire compilé à partir du cahier des charges du gestionnaire TV et lumière :

<b>Nom</b> <b>(acteurs ou classes)</b>	<b>Qualificatifs</b> <b>(Attributs ou relations)</b>	<b>Verbes</b> <b>(opérations)</b>
Capteur de présence/lumière	Relation avec accessoire pour signalisation sonore Et avec l'actionneur de la lumière	Mettre en service Arrêter service Interrompre système Réactiver système
Caméra	Dormant Eveillé	Eteindre TV Allumer TV
Wifi		Envoi information
Capteur de son		Ordre vocal Arrêter Ordre vocal Activer
Bouton ON/OFF		Activer le système Désactiver le système
Equation de contrôle		Calculer le poids personne

*Tableau 2.1. Dictionnaire pour notre application.*

\* **Description des cas d'utilisation :** Décrire les exigences fonctionnelles via des cas d'utilisation et les éléments de l'environnement qui interagissent avec le système par les acteurs.

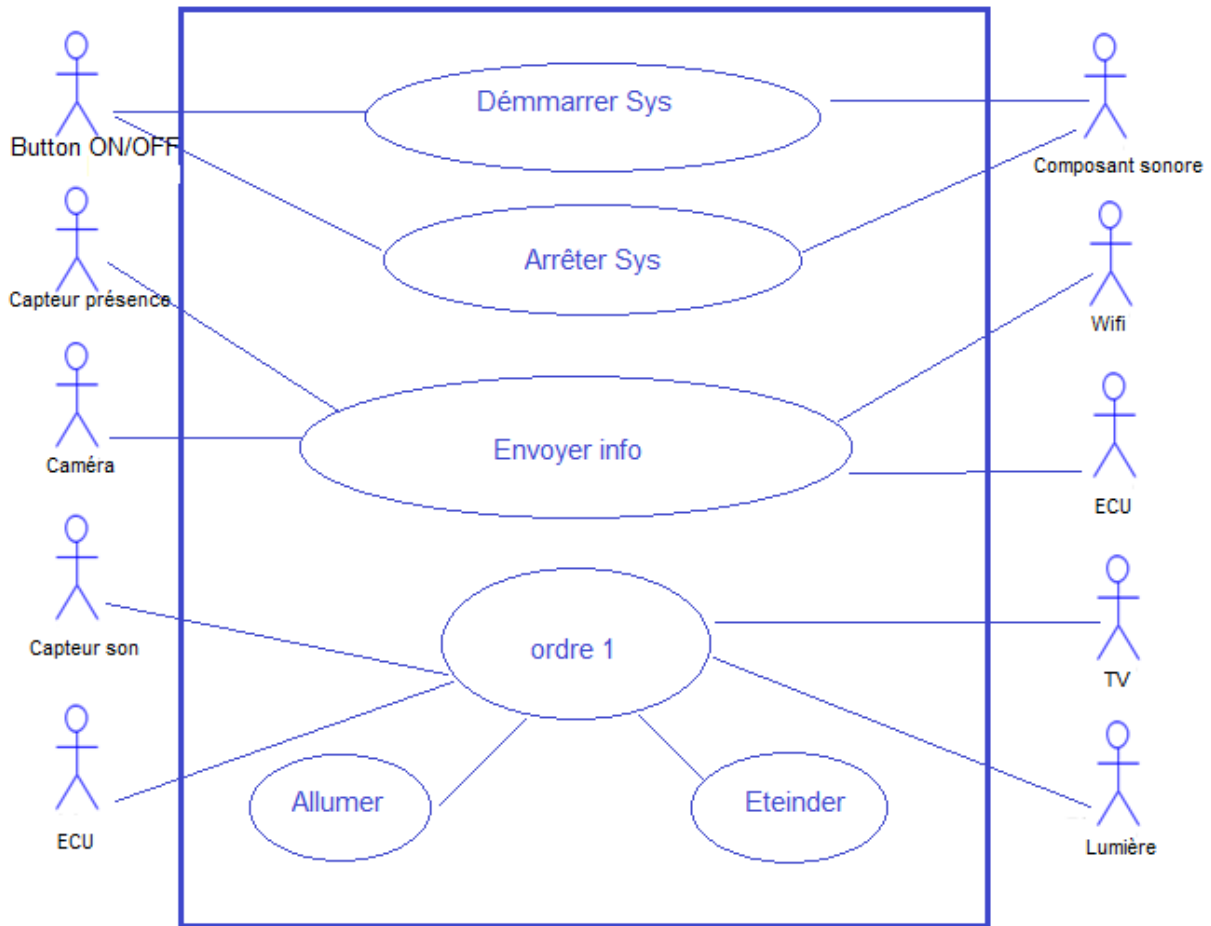


Figure 2.5. Diagramme cas d'utilisation

\* **Les scénarios de haut niveau** : sont des diagrammes de séquence qui décrivent pour chaque cas d'utilisation identifié dans le diagramme de cas d'utilisation un ensemble de scénarios possibles.

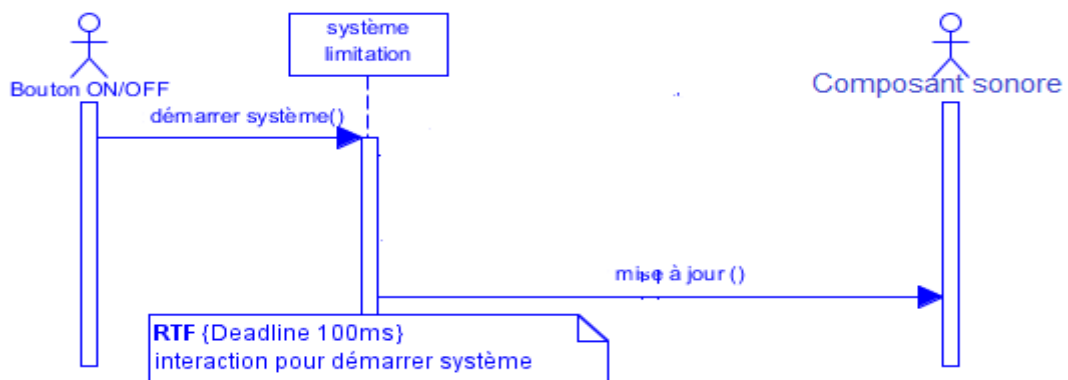


Figure 2.6 diagramme de séquence démarrer le système

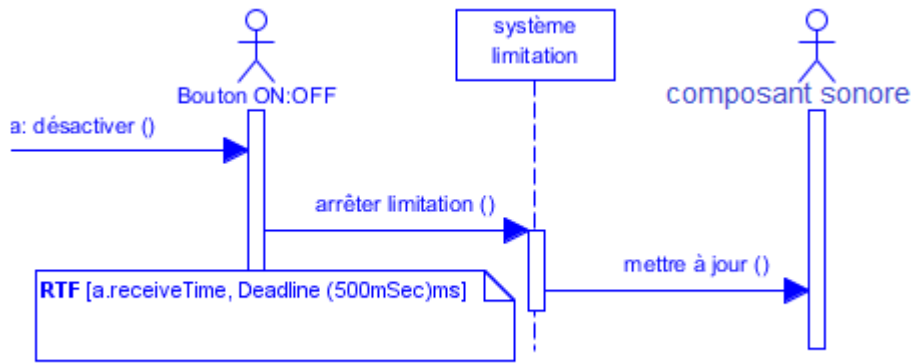


Figure 2.7 diagramme de séquence arrêter le système

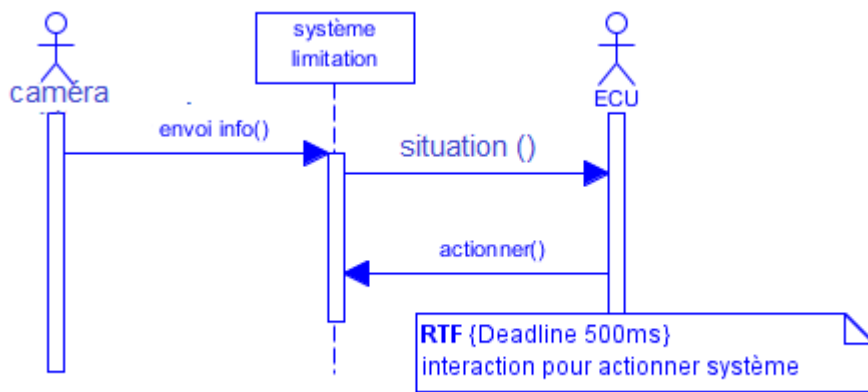


Figure 2.8 diagramme de séquence envoyer informations sur la présence

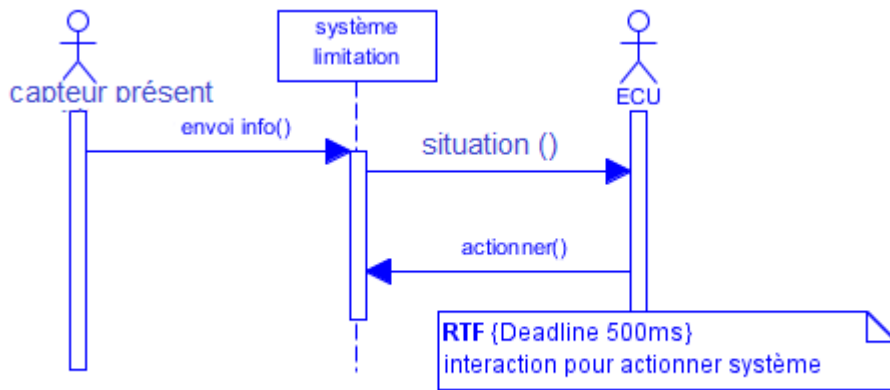


Figure 2.9 diagramme de séquence envoyer informations sur la présence 2

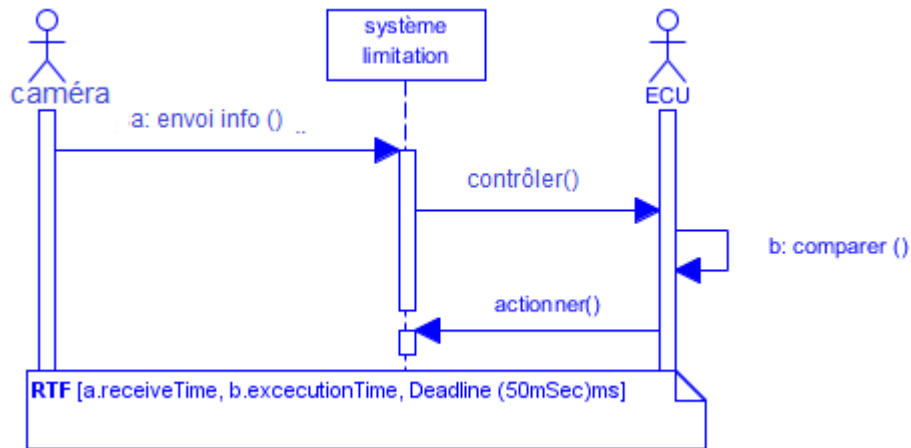


Figure 2.10 diagramme de séquence contrôler le fonctionnement 1

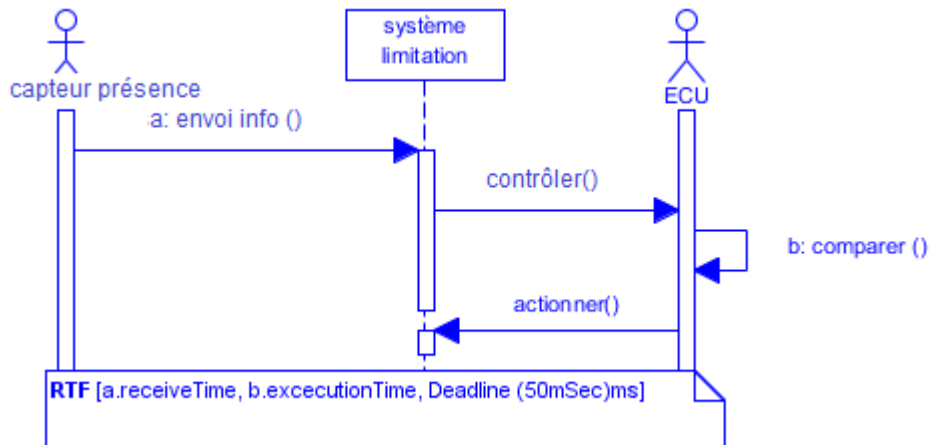


Figure 2.11 diagramme de séquence contrôler le fonctionnement 2

\* **Classification des acteurs (actif et passif) :** le produit du travail de l'activité précédente, les diagrammes de séquence de haut niveau, offrent la possibilité de spécifier le type de rôle joué par les acteurs - passifs ou actifs - par rapport au système. Cette information sera utilisée par la suite dans la phase de modélisation pour construire automatiquement une architecture de composants génériques et aussi construire automatiquement à partir des modèles d'analyse préliminaire des bases des modèles d'analyse détaillée.

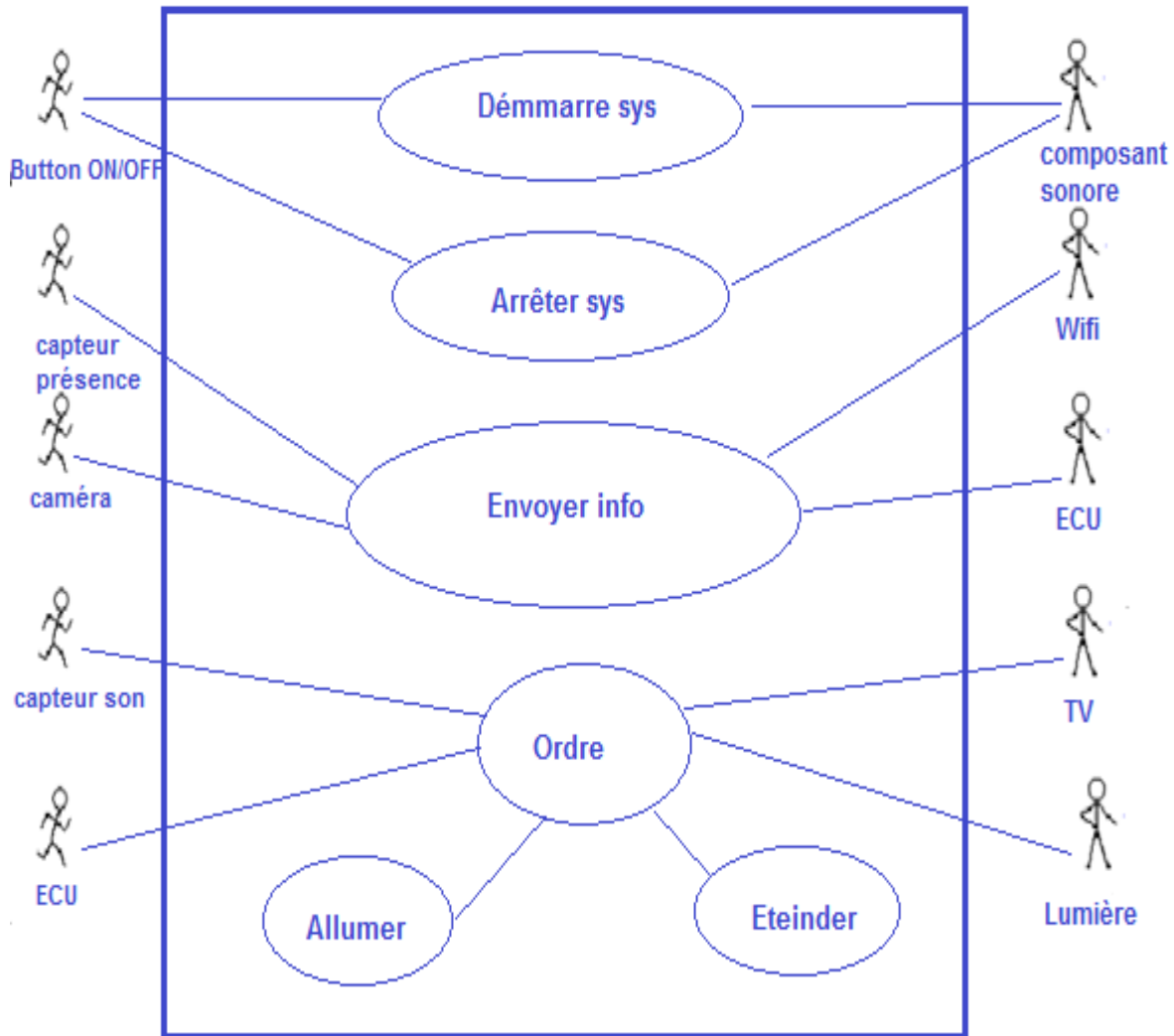


Figure 2.12 Classification des acteurs 'actif ou passif' dans le diagramme cas d'utilisation

### 5.3. MODELE D'ANALYSE DETAILLEE (DAM)

Le modèle clair et sans ambiguïté des besoins des utilisateurs qui résulte de l'analyse préliminaire reflète principalement une approche fonctionnelle. L'objectif de la phase suivante consiste à construire un modèle global de l'application en fonction des besoins des utilisateurs exprimés essentiellement comme des cas d'utilisation et des diagrammes de séquence. Cette phase se préoccupe de trouver des réponses à la question suivante: « Qu'est-ce que mon système est en mesure de faire ? ».

Pour répondre à cette question, on va organiser un modèle global tout en suivant les étapes ci-dessous :

\* **Construction d'une base structurale :** cette activité vise à construire à partir du modèle d'analyse préliminaire (PAM) une première version du modèle structurel du DAM. Ce

premier projet est également détaillée / raffiné jusqu'à ce que le modèle atteigne un détail satisfaisant l'expert de domaine.

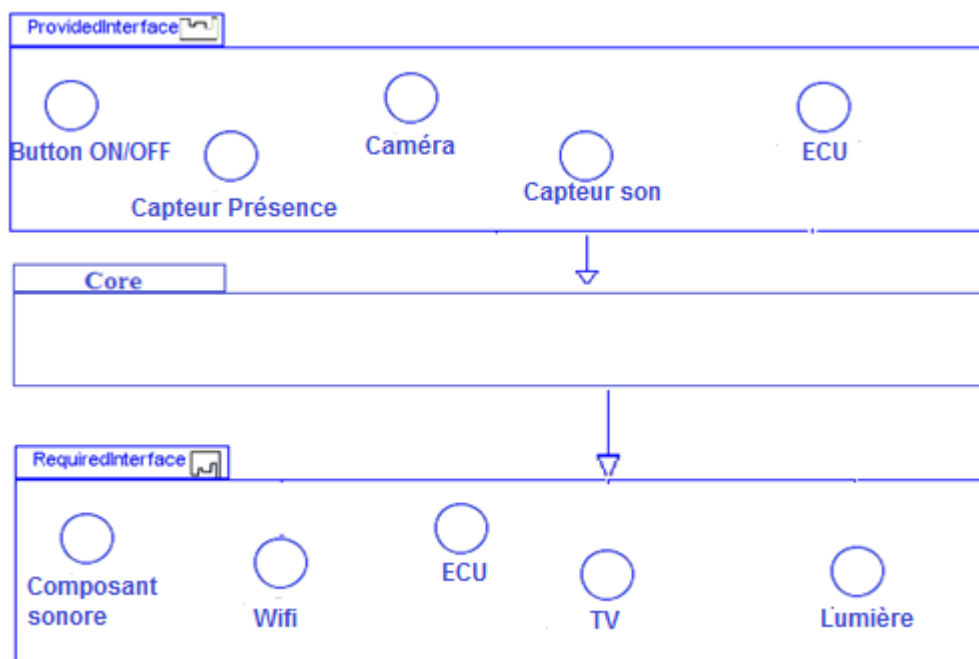


Figure 2.13 Interface packages population

\* **Décrire la vue de la structure** : L'analyse globale de ces liens fournit le modèle structurel global indiqué dans la figure 2.14 Dans ce diagramme, les classes d'interface qui, jusqu'à présent, ont été distingués des autres classes en utilisant le stéréotype «*Interface*», apparaissent maintenant comme les milieux marqués par leurs noms de classe. Cette nouvelle représentation est liée à l'icône spécifique attaché au stéréotype.

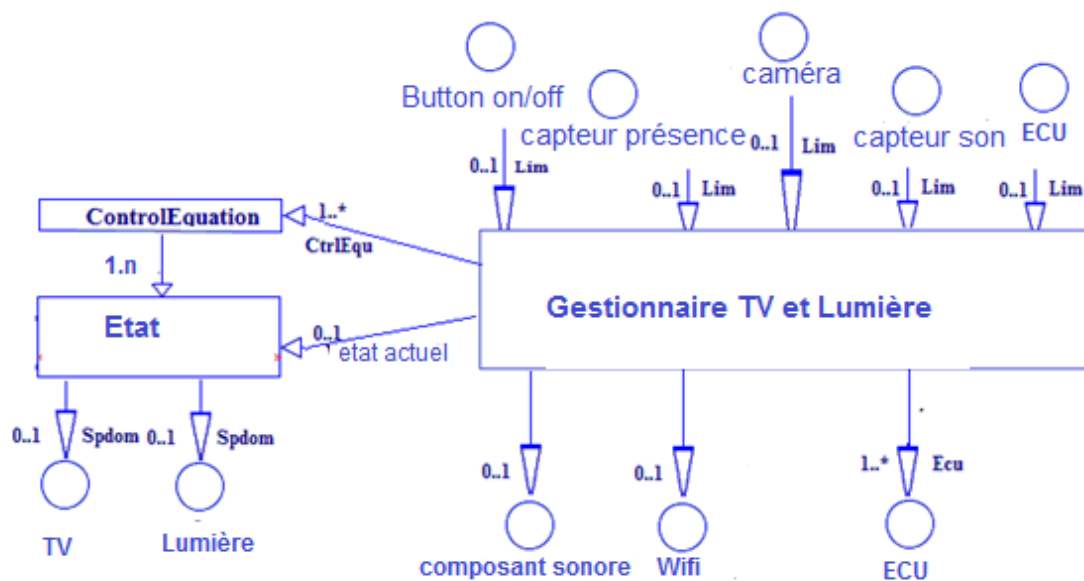


Figure 2.14 Modèle global de structure du gestionnaire TV et lumière avec les relations entre classes.

\* **Spécification de communication:** cette étape consiste à modéliser les signaux reçus et envoyés des différentes classes d'applications, et identifier leurs sources potentielles. La figure 2.15 illustre tous les signaux du modèle global du gestionnaire TV et lumière.

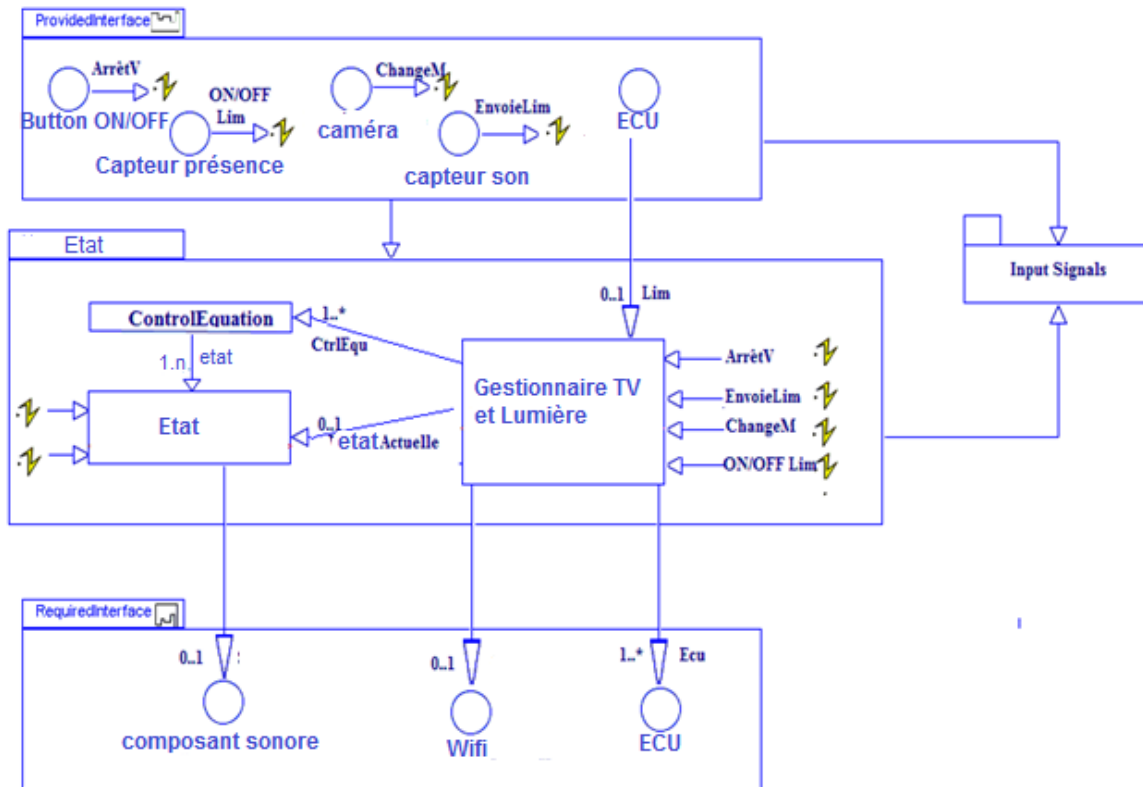


Figure 2.15 Modèle global de structure du gestionnaire TV et lumière sans ambiguïtés

\* **Déclarer les ressources actives:** Dans l'approche ACCORD/UML, la concurrence est gérée par la spécification des objets temps réel dont le fonctionnement est similaire à celle des objets actifs définis dans les langages de programmation concurrents. La figure 2.16 présente la spécification des objets temps réel identifier par le stéréotype « RealTimeObject ».

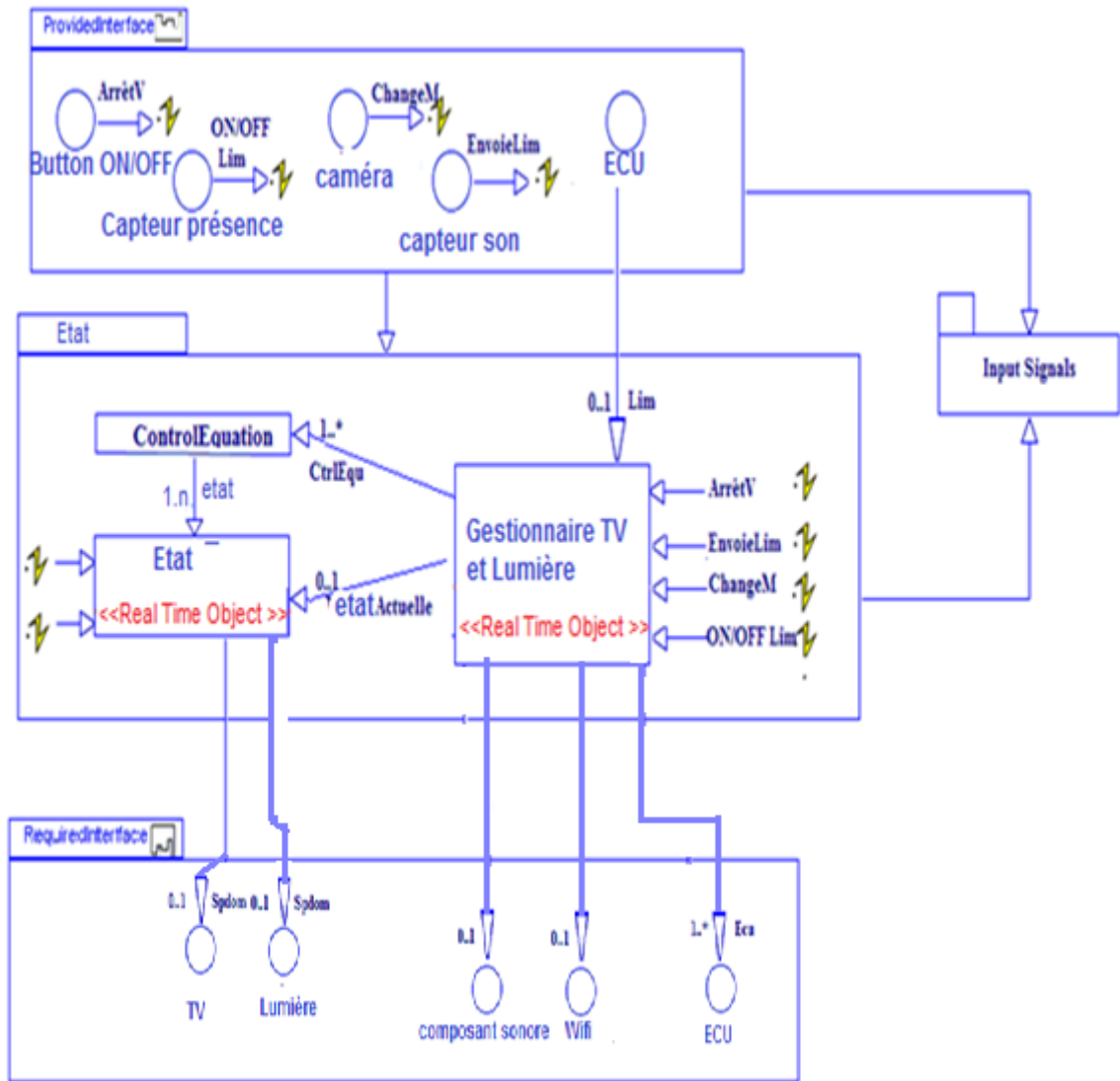


Figure 2.16 Modèle global de structure avec identification du «RealTimeObject ».

## 6. CONCLUSION

Dans ce chapitre, une architecture globale du système a été proposée, avec une description de son fonctionnement. Une modélisation avec ACCORD/UML a été fourni.

# CHAPITRE 3 : REALISATION

---

## 1. INTRODUCTION

---

Après l'achèvement de la conception de système, on suit maintenant une succession d'étapes afin de réaliser un prototype qui enrichisse la vision abstraite du système.

Pour cela, et dans ce qui suit, on va présenter les outils matériels et logiciels utiliser pour la réalisation de prototype, et à la fin on va présenter le montage de notre système.

## 2. LES OUTILS MATERIELS UTILISES

---

### 2.1. PRESENTATION D'ARDUINO :

Arduino est une plate-forme de prototypage d'objets interactifs à usage créatif constituée d'une carte électronique et d'un environnement de programmation.

L'Arduino est une plaquette électronique que l'on peut connecter par USB à un ordinateur afin de téléviser du code sur le microcontrôleur qui s'y trouve. Un microcontrôleur, c'est un processeur de petite taille qui fournit des entrées et sorties analogiques et qui fonctionne selon le programme que l'on aura enregistré dessus. Cet appareil est autonome. Après avoir reçu le programme qu'on a conçu, on peut le déconnecter de l'ordinateur et il fonctionnera tout seul (sous réserve de l'alimenter en énergie électrique avec une pile ou mieux une cellule photovoltaïque). Le circuit Arduino est libre. On peut donc étudier ses plans, et créer des dérivés. Plusieurs constructeurs proposent ainsi différent modèle de circuits [18] .

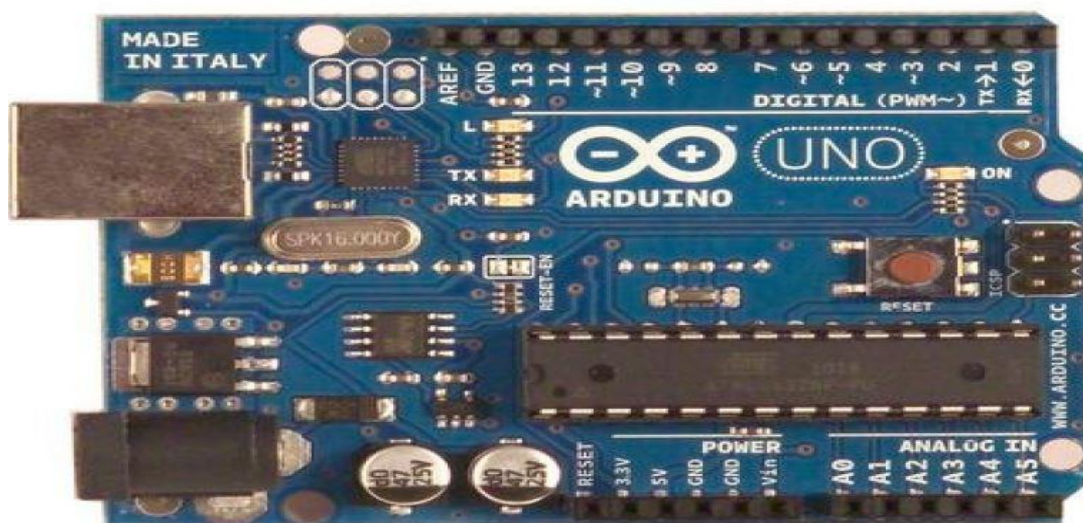


Figure 3.1 Une carte Arduino de type UNO

## 2.2. POUR QUOI ARDUINO ?

L'intérêt des systèmes embarqués est de concevoir des systèmes automatiques puissants et de permettre le contrôle de systèmes complexes.

Alors comment faire des montages électroniques simplement en utilisant un langage de programmation ? La réponse, c'est le projet Arduino qui l'apporte. Celui-ci a été conçu pour être accessible à tous par sa simplicité. Mais il peut également être d'usage professionnel, tant les possibilités d'applications sont nombreuses [19].

### A. Avantages Arduino :

- **Economique** : les cartes microcontrôleur sont bon marché et l'éditeur est gratuit.
- **Ce programme facilement** : le langage Arduino est trop simple à apprendre.
- **Multi plateformes (Windows, Mac, Linux)** : peut s'installer sur des différents OS.
- **Logicielle et matériel open-source** : le schéma de la carte est disponible pour tous et le langage est libre de droit.
- **Nombreuses bibliothèques** : ce qui représente un grand nombre de fonctions faciles à programmer.
- **Nombreuses extensions matérielles** : ces cartes qu'on appelle bouclier (shields) sont facilement branchables sur le microcontrôleur.
- **Communauté très active** : de nombreux blogs et sites web permettent de retrouver de nombreuses ressources en ligne.

### B. Inconvénients Arduino :

- Espace mémoire très limité.
- Prix des shields assez élevé en général.
- Puissance du µc bridée par les fonctions simples d'emploi.
- Les pièces sont indisponibles par fois.

## 2.3. LES DIFFÉRENTES CARTES ARDUINO [20]:

Plusieurs cartes Arduino existent et qui se différencient par la puissance du microcontrôleur ou par la taille et la consommation de la carte. Le choix du type de carte Arduino s'effectue en fonction des besoins de votre projet.

### 1) Arduino UNO :

La carte Arduino UNO est la carte la plus couramment utilisée qui constitue un bon choix pour les débutants. Cette carte convient pour tous les petits projets et la plupart des tutoriels disponibles sont faits avec une carte UNO. Elle présentera cependant ses limites lorsqu'on aura besoin de brancher un nombre plus conséquent de périphérique externe.

## 2) Arduino MEGA :

L'Arduino méga est une Arduino UNO mais avec plus d'entrées et sorties (54 contre 14), plus de mémoire également ce qui peut être utile pour les programmes un peu complexes tout comme son processeur, il est plus puissant.

## 3) Arduino MEGA ADK :

Les caractéristiques techniques restent les mêmes que l'Arduino MEGA mais elle permet d'interagir avec un smartphone Android grâce à un port USB ouvrant de nouvelles horizons en vue de la capacité des smartphones et leurs nombreux capteurs.

## 4) Arduino Yun :

La carte Arduino Yun est la première carte Arduino avec wifi intégré qui combine à la fois la puissance de linux avec la facilité d'utilisation d'Arduino. Cette carte est dérivée de l'Arduino Leonardo et permet toujours plus de perspectives.

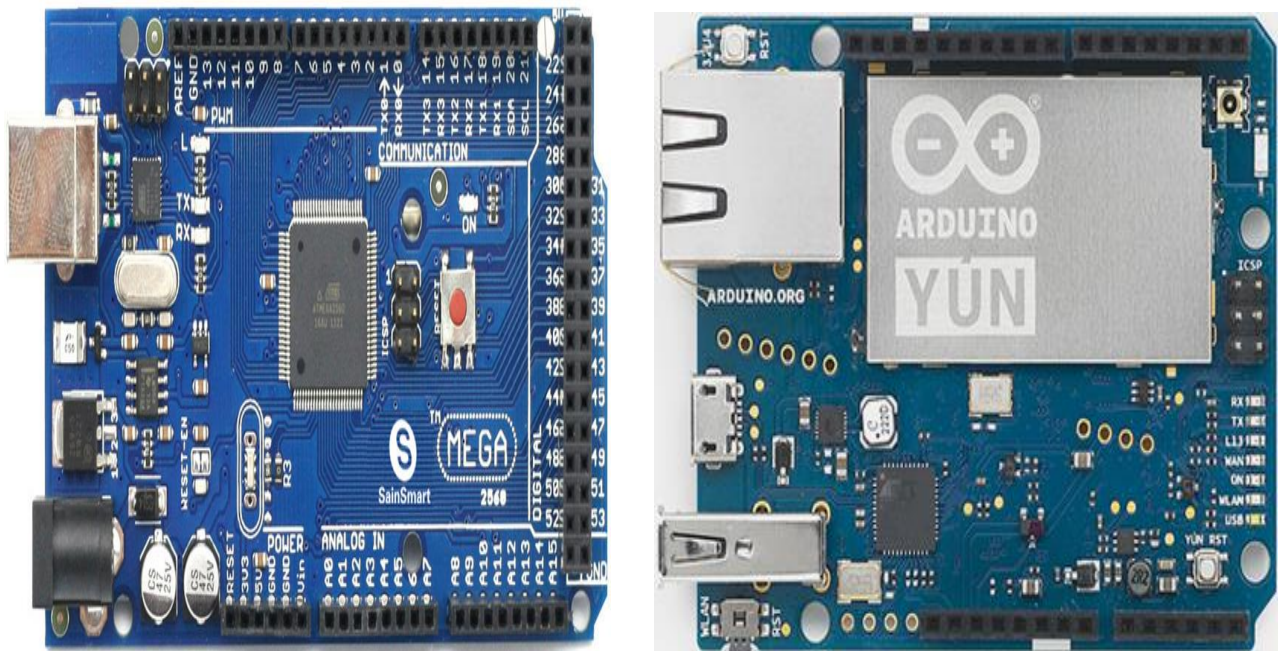


Figure 3.2. Arduino MEGA et Arduino YUN respectivement.

### 2.3.1 CHOIX DE LA CARTE ARDUINO A UTILISER :

Vu les avantages et les inconvénients, le domaine d'utilisation de chaque carte, ainsi que le coût et la disponibilité sur le marché, et bien sûr sans omettre de mentionner que c'est la première expérience qu'on utilise le montage électronique d'un système on a opté pour la carte Arduino UNO.

### 2.3.2. ARCHITECTURE DE LA CARTE ARDUINO « UNO » :

L'Arduino est une carte basée sur un microcontrôleur (mini-ordinateur) Atmel ATMEGA8 ou ATMEGA168. Elle dispose dans sa version de base de 1 Ko de mémoire vive, et 8Ko de mémoire flash pour stocker ses programmes. Elle peut être connectée à 13 entrées ou sorties numériques, dont 3 PWM (pouvant donner 6 entrées et 3 sorties analogiques convertissant en 10 bits sur lesquelles on peut brancher différents appareils :

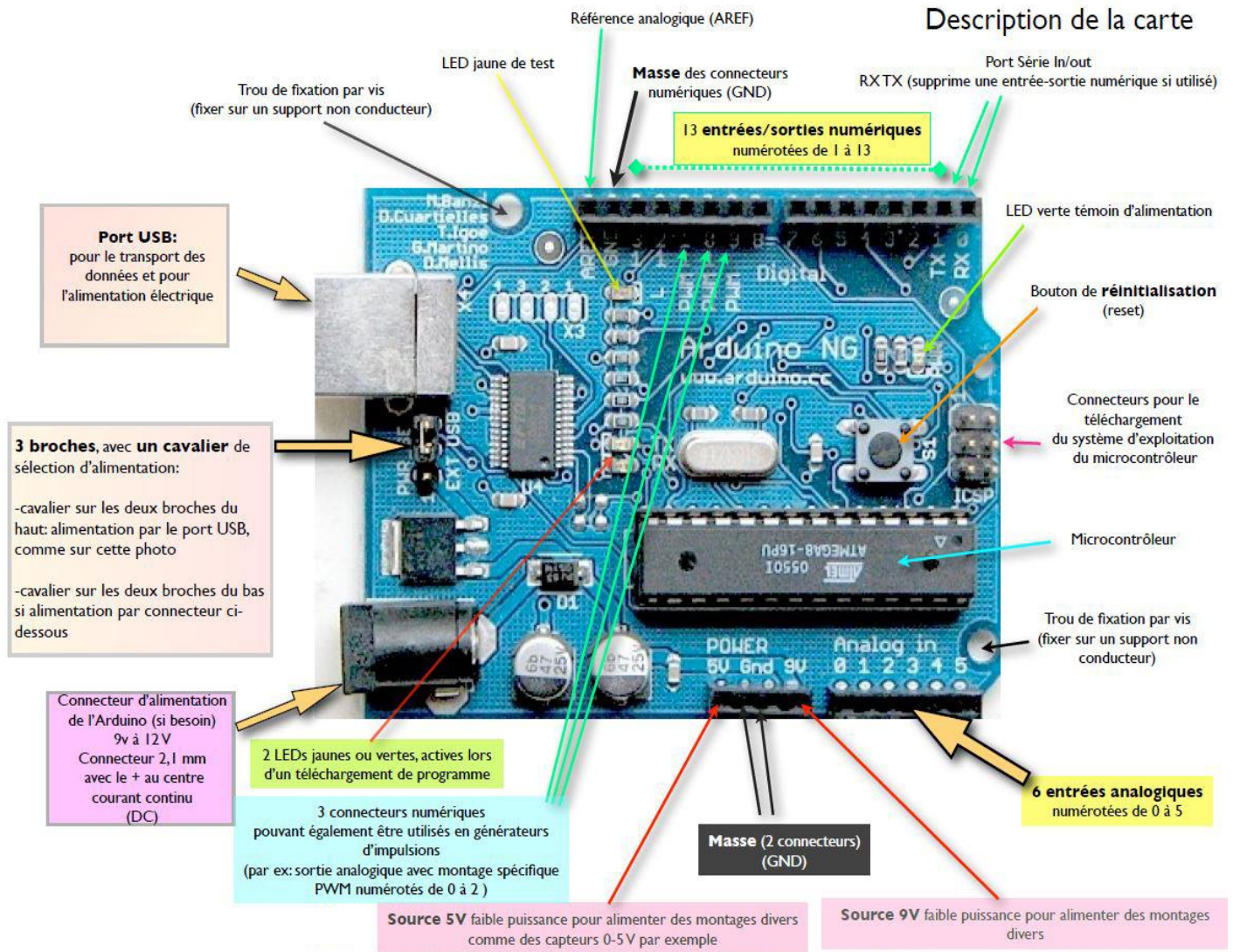


Figure 3.3 Schéma descriptive de la carte Arduino UNO

- ✚ **Côté des entrées**, des capteurs, appareils qui collectent des informations sur leur environnement comme la variation de température via une sonde thermique, le mouvement via un détecteur de présence, le contact via un bouton poussoir, etc.,
- ✚ **Côté des sortis**, des actuateurs, des appareils qui agissent sur le monde physique, telle une petite lampe qui produit de la lumière, un moteur qui actionne un bras articulé, etc.

Dans la version la plus courante, la communication avec l'ordinateur se fait par un port USB. Il existe plusieurs versions de l'Arduino, dont une version miniaturisée, et d'autres projets

sont également en gestation. La carte dispose d'un logiciel système interne (modifiable) et des programmes utilisateur.

La figure 3.3 en dessus décrit bien la carte Arduino UNO.

## 2.4. LES DIFFERENTS ACCESSOIRES ET CAPTEURS UTILISES :

### 2.4.1. DRIVER MOTEUR L293D :

Le composant **L293D** est un pont de puissance composé de plusieurs transistors et relais qui permet d'activer la rotation d'un moteur. Le **L293D** est un double pont-H, ce qui signifie qu'il est possible de l'utiliser pour commander quatre moteurs distincts (dans un seul sens) grâce à ses 4 canaux. Il est également possible de constituer deux pont-h afin de piloter deux moteurs distincts, dans les deux sens et indépendamment l'un de l'autre. Il est important de noter que le **L293D** peut délivrer au maximum 600mA, on doit donc choisir les moteurs en conséquence.

#### ✚ Caractéristiques techniques du L293D

Voici les caractéristiques techniques du composant **L293D** :

- ✚ Nbre de pont-H : 2
- ✚ Courant Max Régime continu : 600mA (x2)
- ✚ Courant de pointe Max < 2ms : 1200mA
- ✚ VS Max Alim moteur : 36v
- ✚ VSS Max Alim logique : 7v
- ✚ Nbre de Broches : 16 DIP
- ✚ Perte de tension : 1.3v



Figure 3.4 Driver moteur

### 2.4.2. MOTEUR A COURANT CONTINU :

Un **moteur DC** est un convertisseur électromécanique permettant la conversion bidirectionnelle d'énergie entre une installation électrique parcourue par un courant continu et un dispositif mécanique ; selon la source d'énergie. Pour faire simple, cela signifie qu'un moteur à

courant continu va pouvoir convertir de l'électricité en énergie mécanique. **Les moteurs DC** ont ainsi la particularité de pouvoir fonctionner dans les 2 sens, suivant la manière dont le courant lui est soumis.



Figure 3.5 Moteur DC

#### **2.4.3. MODULE BLUETOOTH HC-05 :**

Le module HC-05 est un module Bluetooth SPP (Serial Port Protocol) facile à utiliser, conçu pour une configuration de connexion série sans fil transparente. Le module Bluetooth HC-05 peut être utilisé dans une configuration Master ou Slave, ce qui en fait une excellente solution pour la communication sans fil. Ce module Bluetooth de port série est entièrement équipé de la modulation Bluetooth V2.0 + EDR (Enhanced Data Rate) de 3Mbps avec un émetteur-récepteur radio de 2,4 GHz et une bande de base. Il utilise CSR Bluecore 04 - Système externe Bluetooth à puce unique avec technologie CMOS et avec AFH (Adaptive Frequency Hopping Feature).

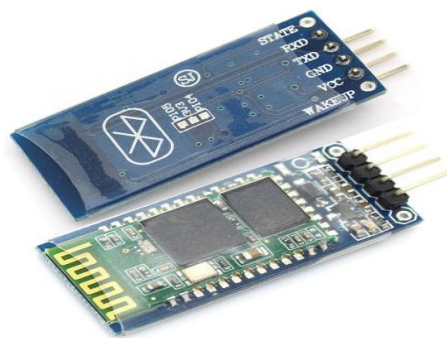


Figure 3.6 Module Bluetooth HC-05

#### **2.4.4. RECEPTEUR INFRAROUGE :**

Les signaux de télécommande sont souvent des octets de données codées et transmises par pulsation (allumage et désactivation de la LED IR à une fréquence spécifique) La plupart des télécommandes fonctionnent à une fréquence de 32 à 40 KHz et la plupart des récepteurs peuvent recevoir cette gamme.



Figure 3.7 Récepteur infrarouge.

Les capteurs de type TSOP permettent de capter ces signaux infrarouges. Ce n'est donc pas seulement la présence d'une lumière infrarouge qui les fait réagir il faut que cette lumière clignote dans une certaine fréquence pour qu'ils réagissent.

#### **2.4.5. AFFICHEUR LCD 16\*2 :**

Les écrans LCD existent depuis 1971. Ils n'ont pas cessé de se développer depuis, et équipent maintenant bien des appareils à affichage embarqué (appareils photo, digicodes, montres, téléphones...). LCD est l'abréviation anglaise de "*Liquid Crystal Display*" qui veut dire : afficheur à cristaux liquides. Cette technologie permet de créer des écrans plats qui consomment peu d'énergie. Les afficheurs LCD, sont des modules compacts intelligents et nécessitent peu de composants externes pour un bon fonctionnement. Ils sont relativement bons marchés et s'utilisent avec beaucoup de facilité.

L'écran LCD qui nous avons travaillé avec est un écran permettant l'affichage de 16x2 caractères, c'est-à-dire deux lignes de 16 caractères.

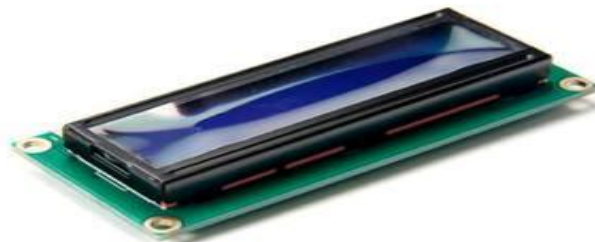


Figure 3.8 Afficheur LCD

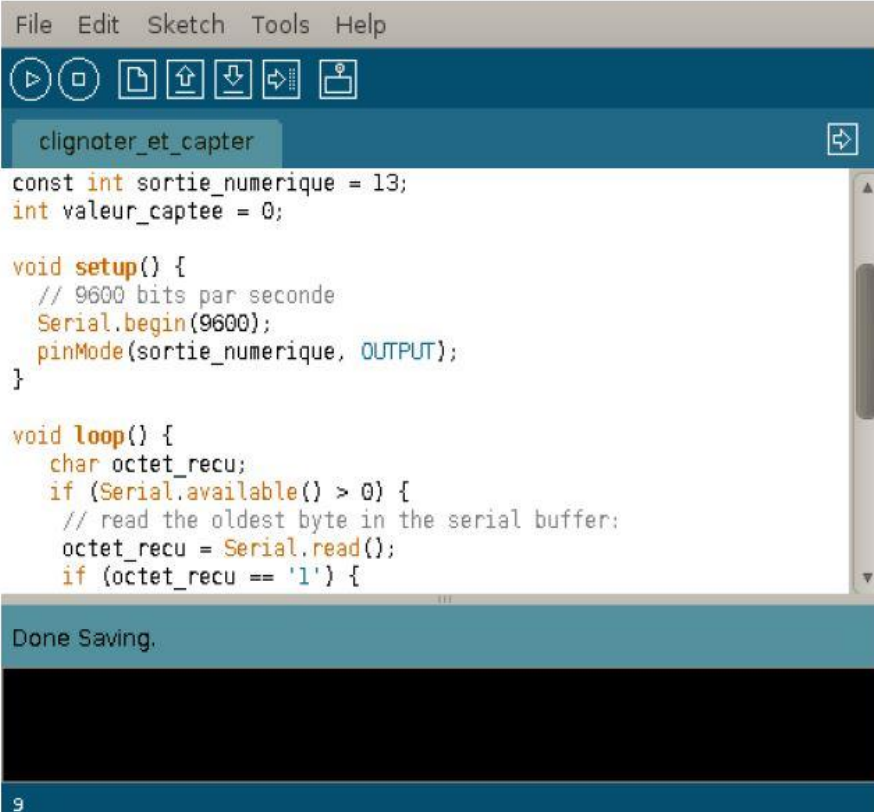
## 3. LES LOGICIELS UTILISES :

### 3.1. ENVIRONNEMENT DE DEVELOPPEMENT [20] :

L'environnement de programmation Arduino (IDE en anglais) est une application écrite en Java inspirée du langage Processing. L'IDE permet d'écrire, de modifier un programme et de le convertir en une série d'instructions compréhensibles pour la carte.

La spécificité d'Arduino se situe au niveau de la sauvegarde du code qui s'enregistre et s'exécute habituellement depuis le microcontrôleur et non pas sur votre ordinateur. Le programme se lance dès que l'appareil Arduino est mis sous tension.

Lorsque l'Arduino est connecté à un ordinateur, il est capable de communiquer avec diverses applications, notamment « Processing »



```
File Edit Sketch Tools Help
clignoter_et_capter
const int sortie_numerique = 13;
int valeur_captee = 0;

void setup() {
  // 9600 bits par seconde
  Serial.begin(9600);
  pinMode(sortie_numerique, OUTPUT);
}

void loop() {
  char octet_recu;
  if (Serial.available() > 0) {
    // read the oldest byte in the serial buffer:
    octet_recu = Serial.read();
    if (octet_recu == '1') {
```

Done Saving.

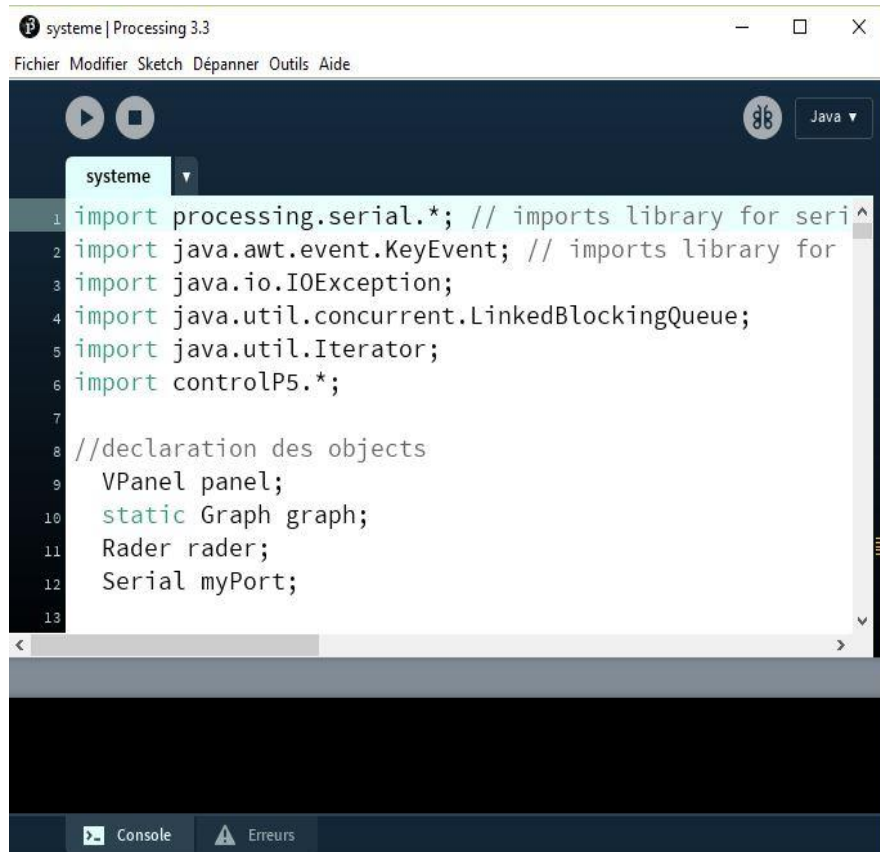
9

Figure 3.9. IDE d'Arduino.

### 3.2. PROCESSING :

Conçu par des artistes, pour des artistes, Processing est un environnement de création fréquemment utilisé pour générer des œuvres multimédias à partir d'un code informatique sur ordinateur. L'attrait de ce logiciel réside dans sa simplicité d'utilisation et dans la diversité de ses applications : image, son, applications sur Internet et sur téléphones mobiles, conception d'objets électroniques interactifs.

Processing fédère une forte communauté d'utilisateurs professionnels et amateurs : artistes, graphistes, vidéastes, typographes, architectes, web designers et designers en général. Il est également utilisé par des enseignants en arts qui souhaitent familiariser leurs étudiants avec les potentialités artistiques de la programmation, les concepteurs du logiciel l'ayant pensé dès l'origine comme un outil d'apprentissage.



```
systeme
1 import processing.serial.*; // imports library for seri
2 import java.awt.event.KeyEvent; // imports library for
3 import java.io.IOException;
4 import java.util.concurrent.LinkedBlockingQueue;
5 import java.util.Iterator;
6 import controlP5.*;
7
8 //declaration des objects
9 VPanel panel;
10 static Graph graph;
11 Rader rader;
12 Serial myPort;
13
```

Figure 3. 10. IDE Processing

### 3.3. LES LANGAGES DE PROGRAMMATION UTILISES :

#### 3.3.1. C++

Le C++ est un **langage** de programmation dit "haut-niveau" créé en 1983 par Bjarne Stroustrup des laboratoires Bell et normalisé en 1998. Il se distingue de son prédécesseur (le **langage C**) en proposant une programmation orientée objet. Ce **langage** est particulièrement utilisé dans les applications demandant de hautes performances.



Figure 3.11. C++

### 3.3.2. PYTHON

Python est le langage de programmation open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages.

Il reste aussi accessible pour les débutants, à condition de lui consacrer un peu de temps pour la prise en main. De nombreux tutoriels sont d'ailleurs disponibles pour l'étudier sur des sites Internet spécialisés ou sur des comptes Youtube. Sur les forums d'informatique, il est toujours possible de trouver des réponses à ses questions, puisque beaucoup de professionnels l'utilisent.



Figure 3.12 Python

### 3.3.3. FLASK

Flask est un micro Framework open-source de développement web en Python. Il est classé comme micro Framework car il est très léger. Flask a pour objectif de garder un noyau simple mais extensible. Il n'intègre pas de système d'authentification, pas de couche d'abstraction de base de données, ni d'outil de validation de formulaires. Cependant, de nombreuses extensions permettent d'ajouter facilement des fonctionnalités.



Figure 3.13 Flask

### 3.3.4. FLUTTER

Flutter est un nouveau Software Development Kit créé par Google, construit en C, C++ et Dart, dont l'objectif est de nous proposer une nouvelle approche «cross-platform» de développement mobile. En se basant uniquement sur le langage Dart, il permet de construire des applications natives Android et iOS. • Flutter se situe au même niveau que React, Native ou Xamarin, ce qui signifie qu'à partir du code écrit en Dart va ensuite être généré du code Android et iOS.



Figure 3.14 Flutter

## 4. IMPLEMENTATION

---

### 4.1. EXTRAITS DE CODE

Dans notre code il y a trois parties essentielles qui sont : la déclaration, la fonction listen qui contient le code, et qui s'exécute une seule fois et la fonction loop qui contient le code récursif.

#### a) LA DECLARATION :

Dans la partie déclaration on a effectué des appels des bibliothèques ainsi que l'initialisation des variables. La figure (3.15) montre un extrait de la partie déclaration).

A screenshot of an IDE window titled "Fichier Édition Croquis Outils Aide". The code editor shows the following declarations:

```
#include<Servo.h>
#include <IRremote.h>
#include<LiquidCrystal.h>
int pinServo = 7;
int pinTrigger = 8;
int pinEcho = 9;
int sens1Moteur1 = 12;
int sens2Moteur1 = 8;
int vitesseMoteur1 = 11;
int sens1Moteur2 = 2;
int sens2Moteur2 = 4;
int vitesseMoteur2 = 5;
int maxD = 40;
int maxV = 180;
int FPS = 45;
int keyCode [10] = {0xffff, 0x4393, 0xffff, 0x4393, 0xffff, 0x4
char DIN_RECEPTEUR_INFRAROUGE = 2;
```

Figure 3.15 Partie déclaration

## b) LA FONCTION LISTEN :

Dans la partie listen, nous avons fait l'initialisation de la communication série définie le mode des pins d'Arduino (entrée/sortie).

```
void _listen() async {
  // _socket = await Socket.connect("192.168.43.251", 8000);

  if (!_isListening) {
    bool available = await _speech.initialize(
      onStatus: (val) {print('onStatus: $val');
        if (val == 'listening'){
          setState(() {
            T = true;
          });
        }else setState(() {
          T= false;
        });} ,
      onError: (val) => print('onError: $val'),
    );
    if (available) {

      setState(() => _isListening = true);
      _speech.listen(
        onResult: (val) => setState(() {
          _text = val.recognizedWords;
        });
      );
    }
  }
}
```

Figure 3.16 un extrait de code de la fonction listen.

## c) LA FONCTION LOOP :

La fonction LOOP sert à chercher en continue la présence d'une personne ou la réception d'une commande vocale.

```
void loop() {
  // put your main code here, to run repeatedly:

  if(Serial.available() >0) {

    command = Serial.read();
    if(command == 'A'){
      digitalWrite(DOOR_PIN, HIGH);
    }
    if(command == 'B'){
      digitalWrite(DOOR_PIN, LOW);
    }
    if(command == 'C'){
      digitalWrite(LIGHT_PIN, HIGH);
    }
    if(command == 'D'){
      digitalWrite(LIGHT_PIN, LOW);
    }
    if(command == 'E'){
      digitalWrite(TV_PIN, HIGH);
      Serial.write("TV on");
    }
  }
}
```

Figure 3.17. Extrait de code de la fonction LOOP

## 4.2. LE PROTOTYPE

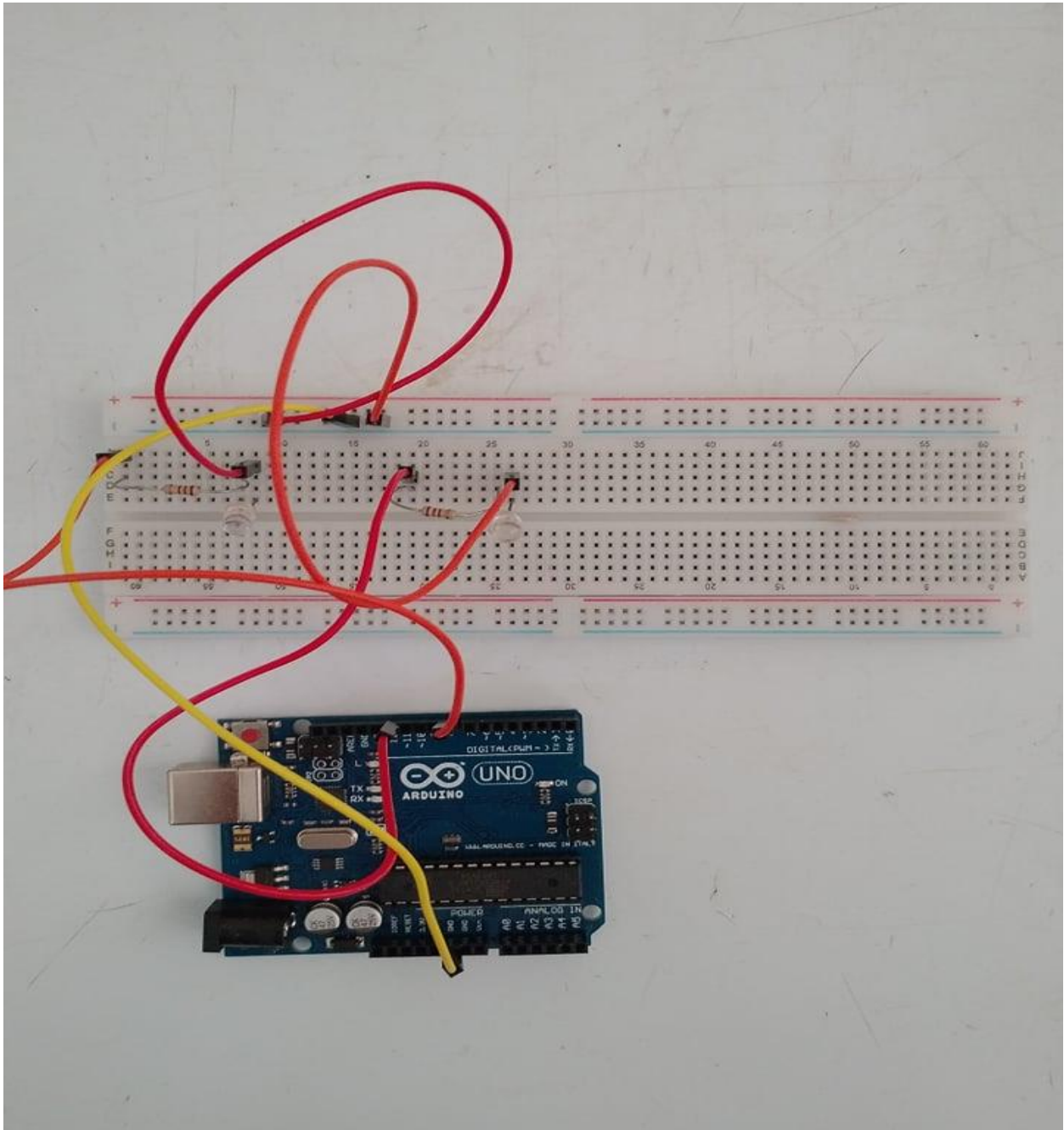


Figure 3.18 Le prototype raccordé.

La figure ci-dessus, montre bien, le montage physique de notre système. Le raccordement de chaque composant, ainsi que son fonctionnement (figure 3.18).

## 5. CONCLUSION

---

En arrivant à ce stade, on a achevé la réalisation de notre système. Et tous ce qui reste sera discuter par la suite dans la conclusion et perspectives.

# CONCLUSION ET PERSPECTIVES

---

Ce travail avait pour but d'étudier la contribution de la sensibilité au contexte sur l'amélioration de la vie des personnes à mobilité réduite. L'état de l'art a permis de mieux comprendre le domaine, de lister les principales méthodes et approches utilisées à mettre en évidence les problèmes des handicaps pour solutionner les problèmes posés par la notion de la sensibilité au contexte.

Nous nous sommes principalement intéressés au problème rencontré à la maison (éteindre/ allumer la lumière ou bien la télévision) afin de se sentir mieux, de bien s'adapter et s'intégrer dans la société. Ce travail pose la première pierre à l'édifice dans l'amélioration de la vie quotidienne des personnes à mobilité réduite mieux encore la réduction significative de leurs dépendances des autres.

Plusieurs perspectives pourraient être envisagées pour poursuivre ce travail. Tout d'abord, la réalisation d'un prototype du système, qui va mettre en valeur nos espérances et estimations.

- [1] Dey, K. A., Abowd, D. G. (2000). Towards a better understanding of context and context-awareness. In Computer Human Interactions (CHI2000) Workshop on the What, Who, Where and How of Context-Awareness.
- [2] Lieberman, H., Selcker, T. (2000). Out of context : Computer systems that adapt to, and learn from, context. In IBM System Journal. Juillet 2000.
- [3] Want, R., Hopper, A., Falcao, V., Gibbons, J. (1992). The active Badge Location System. In ACM Transaction on Information Systems. Octobre 1992. v. 10, p. 91102.
- [4] Schmidt, A., Beigl M., Gellersen H. W. (1998). There is more to context than location. In Proceedings of the International Workshop on Mobile Computing (IMC'98). Rostock, Allemagne, novembre 1998.
- [5] Schilit B., Adams N., Want R. «Context-Aware Computing Applications». 1st International Workshop on Mobile Computing Systems and Applications. pp. 85-90, 1994.
- [6] Ranganathan A., Al-Muhtadi J., Biehl J., Ziebart B., Campbell R., Bailey B. « Towards a pervasive computing benchmark, PerWare '05 Workshop on Support for Pervasive Computing ». Third IEEE International Conference on Pervasive Computing and Communications (PerCom 2005), pp. 194–198, 2005.
- [7] Lombardi S. « Context-awareness and context modeling». Ubiquitous Computing Seminar FS2014, Institute for Pervasive Computing, Zurich, June 10th 2014.
- [8] Privat G., Ramparany F. « les interfaces contextuelles ». Publications France Télécoms R&D, 2007.
- [9] Brown, P. J. (1998). Triggering information by context. In personal Technologies. v. 2, p. 19.
- [10] Ryan N., Pascoe J., Morse D. «Enhanced Reality Fieldwork: the Context-Aware Archaeological Assistant». Gaffney, V., van Leusen, M., Exxon, S. (eds.) Computer Applications in Archaeology, 1997.
- [11] Schilit B., Theimer M. «Disseminating Active Map Information to Mobile Hosts». IEEE Network, 8(5) (1994) 22-32, 1994.
- [12] Salber D., Dey A.K., Abowd G.D. «Ubiquitous Computing: Defining an HCI Research Agenda for an Emerging Interaction Paradigm». Georgia Tech GVU Technical Report GIT-GVU-98-01 (1998).
- [13] Brown, P. J. 1995. « The stick-e document: a framework for creating context-aware applications ». Electronic Publishing-Chichester-.p, 259-272.

- [14] Lieberman, H., Selcker, T. (2000). Out of context : Computer systems that adapt to, and learn from, context. In IBM System Journal. Juillet 2000.
- [15] Miraoui, M., Tadj, C., & Amar, C. B. 2009, October.« Dynamic context-aware service adaptation in a pervasive computing system ». In Mobile Ubiquitous Computing, Systems, Services and Technologies, Third International Conference on IEEE,p. 77-82.
- [16] CEA, I-Logix, Uppsala, OFFIS, PSA, MECEL, ICOM, “Methodology for developing real time embedded systems”, Project IST 10069 AIT-WOODDES, février 2001.
- [17] Sébastien Gérard, Arnaud Cuccuru and Frédéric Loiret, “Accord-UML: a methodological approach for model based development and validation of RT/E systems”, Artist2 workshop: MoCC - Models of Computation and Communication, November 16-17, 2006
- [18] <http://arduino-guides.blogspot.com/2012/05/tsop-ir-receiver.html>
- [19] <http://www.zem.fr/arduino-controler-des-moteurs-dc-avec-lcomposantl293d/>
- [20] <https://www.carnetdumaker.net/articles/mesurer-une-distance-avec-un-capteur-ultrason-hc-sr04-et-une-carte-arduino-genuino/>

## Résumé

L'émergence de la sensibilité au contexte ont favorisé le déploiement des services dans les environnements dédiés aux personnes dépendantes (âgées ou déficientes). Les applications sensibles au contexte contribuent significativement à l'amélioration de la qualité de vie de ces personnes en termes d'autonomie et de sécurité.

Dans ce travail nous allons proposer une application sensible au contexte aux personnes dépendantes.

**Mots clés :** Les applications sensibles au contexte, temps réel, personnes à mobilité réduite.

## Abstract

The emergence of context sensitivity has encouraged the deployment of services in environments dedicated to dependent people (elderly or disabled). Context-sensitive applications significantly contribute to improving the quality of life of these people in terms of autonomy and safety.

In this work we will propose a context sensitive application to dependent people....

**Keyword:** Context sensitive applications, real time, people with reduced mobility.

## ملخص

شجع ظهور حساسية السياق على نشر الخدمات في بيئات مخصصة للأشخاص المعالين (كبار السن أو المعوقين). تساهم التطبيقات الحساسة للسياق بشكل كبير في تحسين نوعية حياة هؤلاء الأشخاص من حيث الاستقلالية والسلامة في هذا العمل ، سنقترح تطبيقًا حساسًا للسياق للأشخاص المعالين.

**الكلمات المفتاحية:** تطبيقات حساسة للسياق ، في الوقت الفعلي ، للأشخاص ذوي القدرة المحدودة على الحركة.