

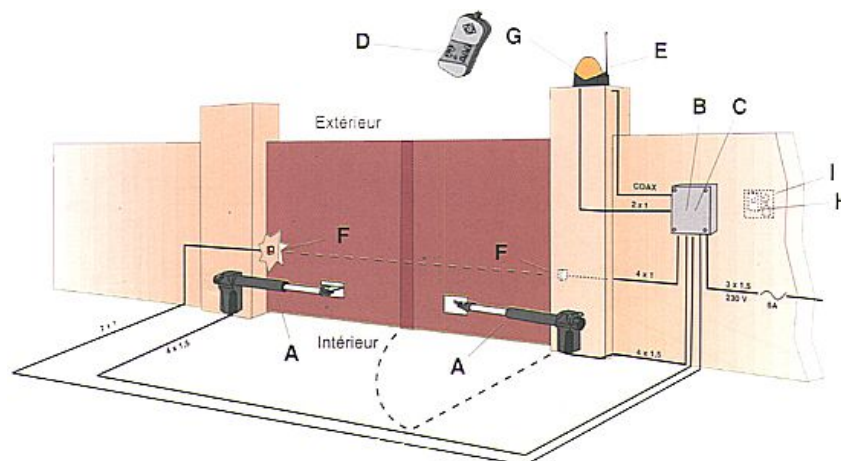
## Feuille de travaux pratiques n° 2 Projet OutDoorGate

### Etude de cas

Ce document présente le cas d'étude et une première analyse en UML. On s'intéresse à un système de portail automatique de propriété individuelle.

## 1 Le cas d'étude en bref

Le cas porte sur un équipement domotique simplifié, la gestion d'une portail d'entrée d'une propriété individuelle à deux battants. Le système en fonctionnement comprend des dispositifs matériels et logiciels. On considère deux dispositifs, un *client* sous forme de télécommande ou de smartphone et un *serveur*, le logiciel de contrôle de la portail, qui pilote les différents dispositifs physique. Une illustration est donnée en Figure ??.



source : <http://www.cabasvanessabruno.fr/electrification-portail-moteur-pour-portillon-17246/>

Figure 1 : Portail automatisé à deux battants

Dans un souci de simplification et d'évolution, nous donnons une description plus abstraite que celle du modèle d'implantation mais moins complexe que celle d'un vrai dispositif.

Il s'agit de concevoir et mettre en œuvre un logiciel de gestion de portails, dont l'analyse du domaine et les spécifications des exigences logicielles (fonctionnelles et non fonctionnelles) sont fournies.

Le prototypage se fera en utilisant des automates LEGO MINDSTORMS EV3, un ensemble d'outil de construction, programmation et de commande de robots LEGO.

## 2 Description du fonctionnement et des composants

Ce système comprend deux battants qui s'ouvrent et se ferment, un moteur pour actionner les battants (ou portes) (tirer, pousser) et des capteurs pour recueillir des informations (contact battant ouvert, contact portail fermé,

détecteur de présence). Les capteurs sont similaires en ce sens qu'ils signalent à leur contrôleur qu'un événement (contact, présence) a eu lieu. L'utilisateur dispose d'une télécommande pour piloter la portail avec simplement deux boutons : ouverture partielle, ouverture totale. Ces boutons déclenchent la fermeture.

Le principe de fonctionnement du système est le suivant. Supposons le portail fermé. L'utilisateur enclenche l'ouverture de la portail en appuyant sur le bouton `OuvertureP` ou le bouton `OuvertureT` de sa télécommande. Il peut arrêter l'ouverture en réappuyant sur un de ces boutons, le moteur s'arrête. Dans le cas contraire le portail s'ouvre sur un battant ou complètement sur deux battants et déclenche les capteurs `porte ouverte` qui entraîne l'arrêt du moteur. Appuyer sur le bouton `OuvertureP` le bouton `OuvertureT` entraîne la fermeture de la portail si elle est ouverte (partiellement ou complètement).

On peut arrêter la fermeture en réappuyant sur les boutons qui relancent une ouverture, le moteur s'arrête. Dans le cas contraire la portail se ferme complètement et déclenche un capteur `portail fermé` qui entraîne l'arrêt du moteur. Si on détecte une présence lors de la fermeture alors la portail se bloque puis se remet en ouverture.

## 2.1 Télécommande

La télécommande dispose de deux boutons, `OuvertureT` et `OuvertureP`, sur lesquels l'utilisateur peut appuyer à tout moment.

- Lorsque le portail est fermé, une pression sur le bouton `OuvertureP` déclenchera l'ouverture partielle (le battant gauche) du portail tandis qu'une pression sur le bouton `OuvertureT` déclenchera l'ouverture totale (les deux battants) du portail.
- En ouverture partielle, une pression sur le bouton `OuvertureP` arrête la manœuvre. En ouverture ou à l'arrêt, une pression sur le bouton `OuvertureT` déclenchera l'ouverture totale du portail.
- En ouverture totale, une pression sur le bouton `OuvertureT` arrête la manœuvre. A l'arrêt, une pression sur le bouton `OuvertureT` déclenchera la fermeture totale du portail.
- Lorsque le portail est ouvert, une pression sur l'un des boutons le referme.
- En fermeture, une pression sur l'un des boutons arrête la fermeture et déclenche l'ouverture totale.

Le choix entre ces activités sera fait par le contrôleur en fonction de l'état de la portail. La télécommande, seule, ne peut pas décider de la marche à suivre. La télécommande, lorsqu'elle est activée, réagit à deux événements (appui sur le bouton d'ouverture partielle ou appui sur le bouton de ouverture totale) et, à chaque fois, se *contente* de signaler au contrôleur qu'il y a eu pression sur le bouton. Son comportement est modélisé dans la figure 2.

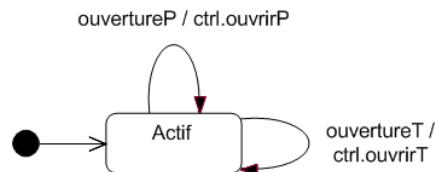


Figure 2 : Diagramme Etats-transitions de la télécommande - portail

La variable `ctrl` représente le contrôleur.

Les événements `ouvertureP` et `ouvertureT` induisent l'exécution, par la télécommande, d'une action du contrôleur. Cela se traduit par l'envoi d'un message au contrôleur pour lui demander d'exécuter son action `ouvrir`.

## 2.2 Capteur de contact

Le capteur signale à son contrôleur lorsque le contact a lieu. Il se comporte donc de façon comparable à la télécommande et nous pouvons modéliser son comportement par la figure 3. Sachant qu'il y a plusieurs capteurs de contacts, on les distingue par leur identifiant, fourni au contrôleur par la variable `self`.

## 2.3 Capteur de présence

Le capteur signale à son contrôleur lorsque le contact a lieu. Il se comporte donc de façon comparable à la télécommande et nous pouvons modéliser son comportement par la figure 4.

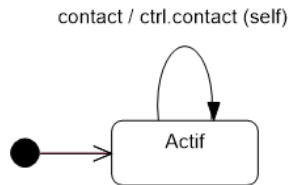


Figure 3 : Diagramme Etats-transitions de la télécommande - portail - Capteur de contact

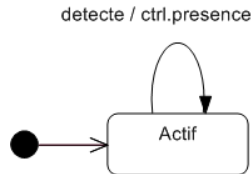


Figure 4 : Diagramme Etats-transitions de la télécommande - portail - Capteur de présence

## 2.4 Moteur

Le moteur actionne la porte (le battant). Soit il pousse la porte, soit il la tire... soit il est à l'arrêt. Nous avons donc trois états, En poussée, En tirée, Arrêt. Le résultat de ces actions est mentionné en filigrane dans le texte. Lorsque le moteur pousse, la porte se referme; lorsqu'il tire, la porte s'ouvre. Nous supposons que le moteur réalise trois actions : pousser, tirer et arrêter. Ces actions sont exécutées lorsque le contrôleur lui demande de le faire. Son comportement est alors celui de la figure 5.

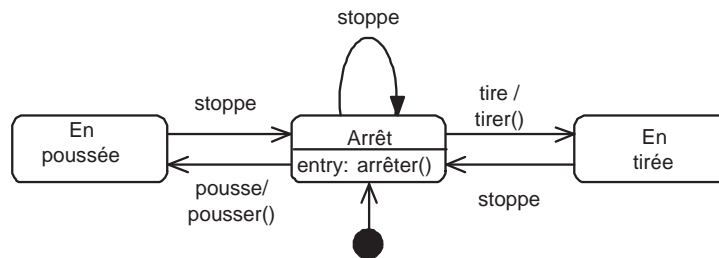


Figure 5 : Diagramme Etats-transitions du moteur - portail

## 2.5 Battant (porte)

La modélisation du comportement d'un battant va faire appel à plusieurs notions. Le comportement de la porte peut être décrit par la figure 6. La variable `mt` correspond au moteur qui commande la portail.

Au niveau macroscopique, la porte peut être dans un des trois états suivants : fermée, ouverte ou en mouvement. Elle se bloque dès que le contrôleur le décide. Cet arrêt dure jusqu'à ce qu'une seconde pression sur la télécommande soit détectée, ce qui aura pour effet de relancer le mouvement dans le sens demandé. La portail peut donc être dans cinq états distincts, `porteOuverte`, `porteFermée`, `Attente`, `Ouverture` et `Fermeture`. L'état `Mouvement` est en fait un super-état (il regroupe `Ouverture`, `Attente` et `Fermeture`) et le diagramme final est un automate hiérarchique.

## 2.6 Contrôleur

Le contrôleur reçoit les signaux des capteurs et de la télécommande et commande le portail. Son rôle est d'analyser ces signaux et de décider, en fonction de ceux-ci, ce que doit faire le portail. Pour décrire nous pourrions nous baser sur un produit des automates des comportement des battants mais nous avons choisi une description plus spécifique. Les états du contrôleur correspondent, en fait, à la coordination des deux battants qu'il faut abstraire au niveau du portail, aux différentes étapes dans son fonctionnement.

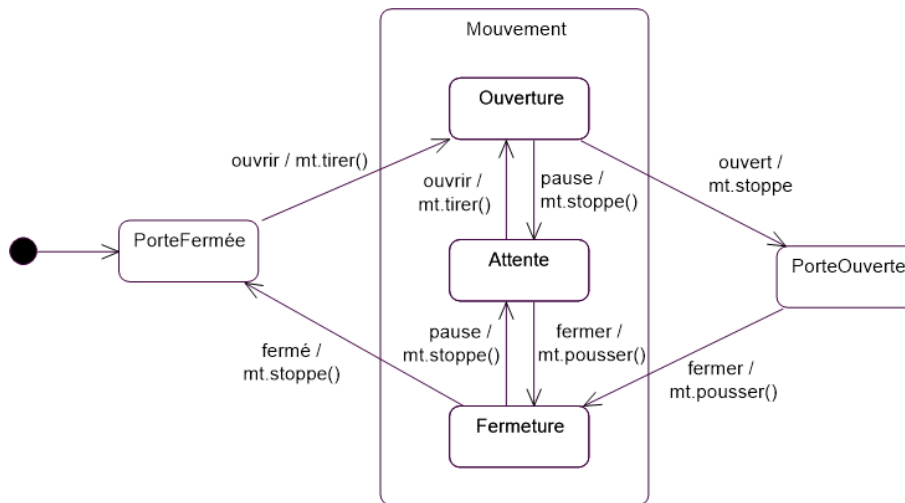


Figure 6 : Diagramme Etats-transitions de la porte - portail

Au niveau macroscopique, le portail porte peut être dans un des quatre états suivants : fermé, ouvert partiellement, ouvert totalement ou en mouvement. Contrairement à l'automate d'un battant, les mouvements sont plus complexes car on tient compte des capteurs en plus des ordres de la télécommande.

Un état `Mouvement` n'a pas vraiment de sens ici du fait de la séparation des cas entre une ouverture partielle et une ouverture totale et le tissage dense entre les cas. Le diagramme d'activités de la figure 7 décrit ce comportement. La variable `pg` (resp. `pd`, `lum`) représente la porte gauche (resp. droite, lumière). L'événement `ouvrirT`

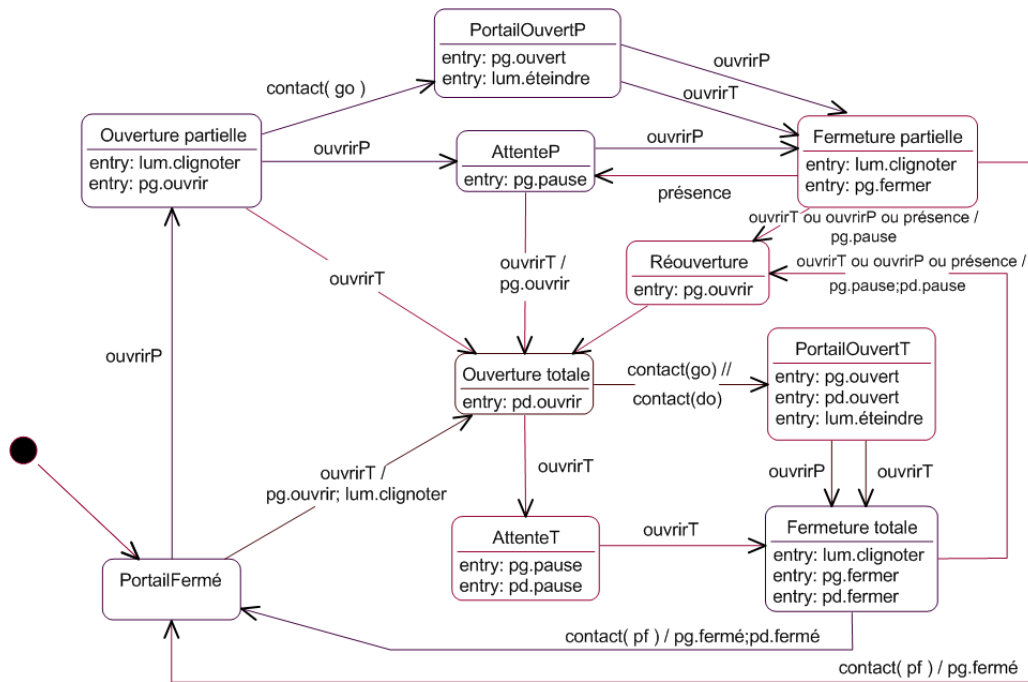


Figure 7 : Diagramme Etats-transitions du contrôleur - portail

ou `ouvrirP` ou `présence` est un raccourci syntaxique pour trois événements ayant un effet similaire. Les actions sur les portes sont transférées aux moteurs par la porte elle-même.

Le comportement global s'explique comme suit :

- Lorsque le portail est fermé, une pression sur le bouton `OuvertureP` de la télécommande déclenchera l'ouverture partielle (le battant gauche) du portail tandis qu'une pression sur le bouton `OuvertureT` de la télécommande déclenchera l'ouverture totale (les deux battants) du portail. Les ouvertures s'arrêtent

lorsque les capteurs de contact associés gauche *go* ou droit *do* signalent l'arrivée en butée d'ouverture.

- En ouverture partielle, une pression sur le bouton *OuvertureP* arrête la manœuvre. En ouverture partielle ou à l'arrêt (état *AttenteP*), une pression sur le bouton *OuvertureT* déclenchera l'ouverture totale du portail.
- En ouverture totale, une pression sur le bouton *OuvertureT* arrête la manœuvre. A l'arrêt (état *AttenteT*), une pression sur le bouton *OuvertureT* déclenchera la fermeture totale du portail.
- Lorsque le portail est ouvert, une pression sur l'un des boutons le referme.
- En fermeture, si une présence est détectée, la fermeture s'interrompt et l'ouverture totale est relancée. De même, une pression sur l'un des boutons de la télécommande arrête la fermeture et déclenche l'ouverture totale.

## 2.7 Système global

Le système en fonctionnement comprend des dispositifs matériels et logiciels.

## 2.8 Structure

La structure matérielle est constituée d'un contrôleur, de deux portes (battant) équipées chacune d'un moteur, d'une lumière clignotante, d'une télécommande, d'un capteur de présence et de trois capteurs de contact (un pour détecter que les battants gauche et droits sont ouverts et un pour détecter la portail fermée<sup>1</sup>). Dans une version abstraite du diagramme de classes, figure 8, nous avons uniquement des opérations.

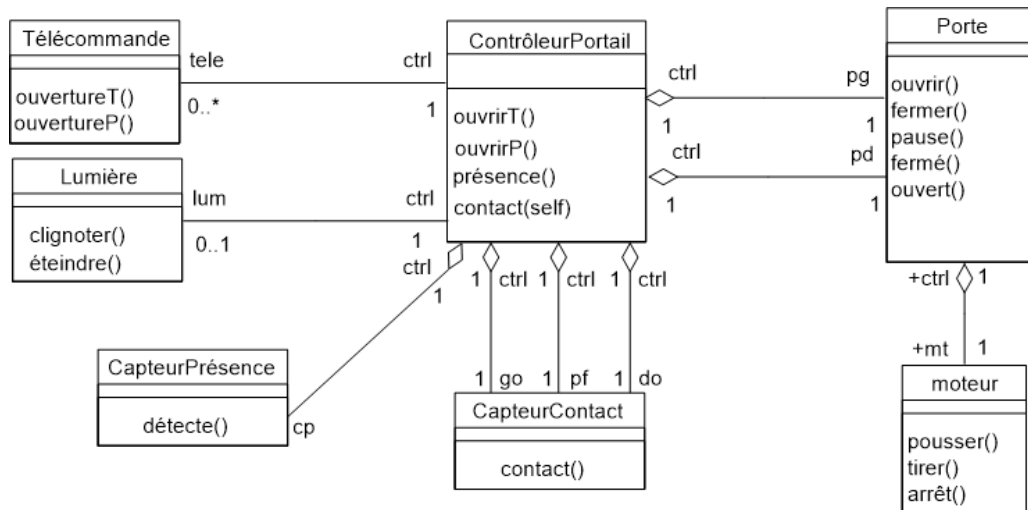


Figure 8 : Diagramme de classes - portail

Nous pourrions, par la suite (durant la phase de conception) ajouter éventuellement un attribut *état* dans chacune des classes ayant un comportement dynamique (i.e. représenté par un diagramme états-transitions), cet attribut servant de support d'enregistrement de l'état courant. Cela risque toutefois de faire double emploi avec un attribut du métamodèle.

## 2.9 Protocole de communication

Les dispositifs physique échangent directement avec le contrôleur. On suppose donc un système simple d'échanges de messages en circuit fermé.

La télécommande communique à distance avec le contrôleur pour commander l'ouverture et la fermeture des portails. En pratique, la communication avec LEGO MINDSTORMS EV3 se fait en bluetooth ou Wifi. En extension, un protocole sécurisé vérifie l'authenticité de la télécommande pour ces échanges.

1. Nous aurions pu en avoir deux, mais l'EV3 n'ayant que 4 ports, nous avons dès le début intégré cette contrainte d'implantation.

### 3 Exigences, contraintes et hypothèses

Nous précisons ici quelques exigences du système.

#### 3.1 Hypothèses et restrictions

On ne traite que de la partie logicielle. On ne tient pas compte du fonctionnement en mode manuel, lorsque le contrôleur est déconnecté.

La modélisation proposée ici reste une esquisse, tous les éléments n'ont pas été analysés en détail (notamment les interfaces des dispositifs matériels, les communications et protocoles associés...). Malgré tout, le modèle proposé permet de mieux comprendre le système à implanter et d'en faire une simulation, non seulement logicielle (le *twin digital*) mais aussi matérielle avec le prototype Lego.

#### 3.2 Exigences fonctionnelles

Le système doit permettre de manipuler la portail via son contrôleur. Ces exigences fonctionnelles sont classées par priorité.

1. Gestion de configuration : lancement, arrêt.
  - (a) Initialiser le mode automatique, paramétrer
    - i. Vérifier l'état des dispositifs physique (capteurs et actionneurs).
    - ii. Initialiser chaque dispositif.
    - iii. Initialiser le mode automatique du contrôleur.
    - iv. Vérifier la cohérence globale (assertions).
2. Fonctionnement de base - Gérer les mouvements
  - (a) ouvrir partiellement
  - (b) fermer
  - (c) ouvrir totalement
3. Fonctionnement sécurisé de niveau 1
  - (a) test de la lumière clignotante
  - (b) test de présence en fermeture
  - (c) test de fonctionnement aléatoire de la télécommande
  - (d) test des capteurs d'ouverture, fermeture
4. Monitoring
  - (a) Afficher l'état du système. Etat du contrôleur Etat des dispositifs
  - (b) Gérer les mouvements Ouverture partielle ou totale, Fermeture partielle ou totale
  - (c) Gestion d'un historique (Log)
    - i. Liste des commandes et des états du contrôleur
    - ii. Liste des anomalies et changements de modes (urgence/normal)
5. Gestion des incidents
  - (a) Dysfonctionnement des capteurs
  - (b) Prise en compte des interruptions de mouvements et des opérations de reprise.
  - (c) Suivi des pannes
6. Gestion des accès
  - (a) Enregistrer un usager pour l'administration du système
  - (b) Modifier, supprimer les d'informations d'un usager (il y a au moins un administrateur du système).
  - (c) Enregistrer une télécommande , Supprimer une télécommande
7. Divers.

### 3.3 Extra-Fonctionnelle

Le système doit permettre de manipuler la portail en toute sécurité.

- Le portail ne peut-être à la fois ouvert complètement (que ce soit pour un ou deux battants) et fermé complètement.
- L'état des dispositifs est cohérent avec celui du contrôleur (cohérence des capteurs).
  - Si le portail est ouvert, le contrôleur la perçoit comme tel.
  - Si le portail est ouvert ou fermé, le moteur est à l'arrêt.
  - Si le portail est en attente, le moteur est à l'arrêt.
  - Si le portail est en ouverture, le ou les moteurs sont en tirage.
  - Si le portail est en fermeture, le ou les moteurs sont en poussée . . .
- Lorsqu'une présence est détectée en fermeture, la portail se bloque. Lorsqu'il reprend son mouvement, aucun obstacle ne doit être détecté.

### 3.4 Exigences techniques

La mise en œuvre du prototype se fera avec les outils Java pour Lego Mindstorm, la plate forme **Lejos**.

### 3.5 Evolutions à prévoir

On peut imaginer des facilités supplémentaires paramétrables.

- Un détecteur de présence empêche le portail de se refermer en cas d'obstruction.
- Lorsqu'un obstacle est détecté en mouvement, le portail se bloque. Lorsqu'elle reprend son mouvement, aucun obstacle ne doit être détecté.
- Ajout de temporisations (contraintes temporelles). Par exemple, le portail "en attente" se referme au bout 2 minutes...
- Le portail se met en demi-ouverture.
- En cas de rupture de communication avec la télécommande (pas de nouvelle action au bout d'un certain temps), le portail se met automatiquement en mode manuel.
- En cas de dysfonctionnement d'un dispositif, le portail se met automatiquement en mode manuel, par exemple si un capteur n'est plus actif (ne répond plus).