Document d’avancement

Partie Simon HERLIN

**Introduction**

Depuis la dernière réunion plus remarque ont été faite par rapport au diagramme de classe et de séquence. Donc dans un premier temps les différentes modifications effectuaient pour répondre aux exigences nécessaires au bon fonction du logiciel seront montré puis une fois ces modifications montraient, vous pourrez observé les différents IHM développées ainsi qu’à la suite les classes du logiciel avec leurs explications.

La première modification majeure qui a été faite est les diagrammes UML, ceux de séquence pour la circulation de l’information (commande des modules) du site HERCOME WEB à HERCOME Center vers les modules.

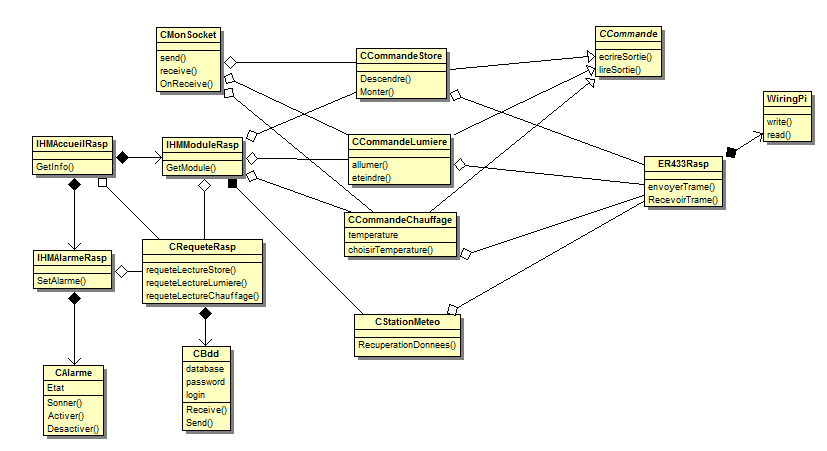
Pour cela les anciens diagrammes de séquences doivent être séparée en deux diagrammes pour bien séparer le côté site web de l’application lourde.

La deuxième modification majeure est la plus grosse est le diagramme de classe de HERCOME Center, en effet plusieurs classes ont été ajouter, certainne ont été suprimer et surtout une classe de gestion qui s’occupera de gérer entierememnt le trafic des informations fût ajouté, cela donne une meilleur organisation, visibilité et cohérence au diagramme de classe, de ces changements les diagrammes de séquence de HECOME Center ont tous d^’ut subir unléger changement.

La troisième modification majeure est la gestion de l’alarme, en effet le mot de passe ainsi que l’état de l’alarme seront enregistrer dans un fichier et non sur la base de données. Cela résoudra plusieurs problèmes déjà un sur la sécurité, le mot de passe sera plus difficile à hacker. Mais aussi sur le fonctionnement de l’alarme après ue coupure de courant elle recherchera diractement l’état sur le fichier et non sur la bdd quelle ne peut pas trouver avant un certain temps (le temps que tout ce réniatilise convenablement).

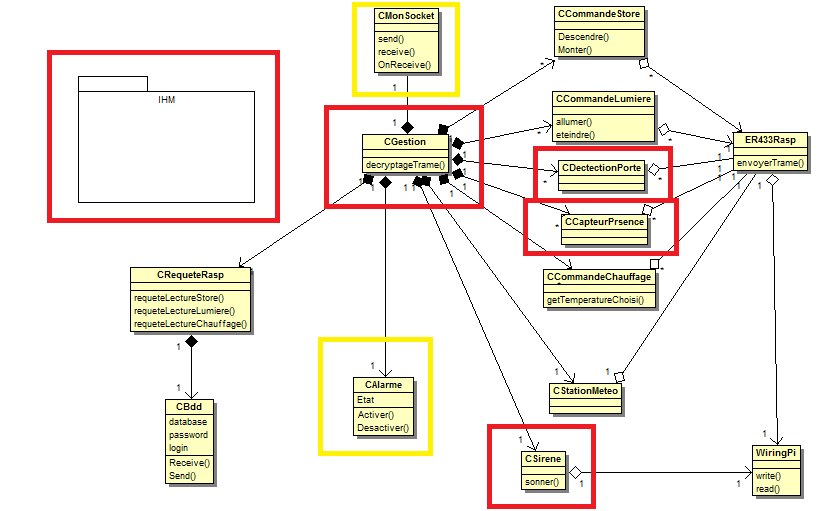
**Modification des diagrammes UML**

Ancien diagramme de classe HERCOME Center



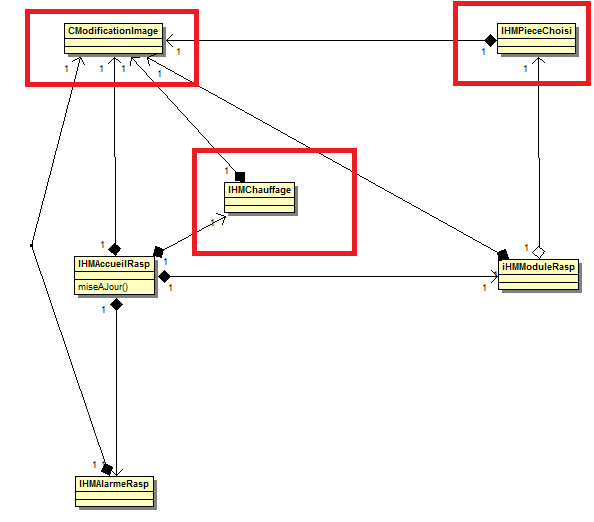
Nouveau diagramme de classe HERCOME Center :

En rouge les classes qui ont été rajouté et en jaune les modifications de postion/ liaison au niveau des classes.

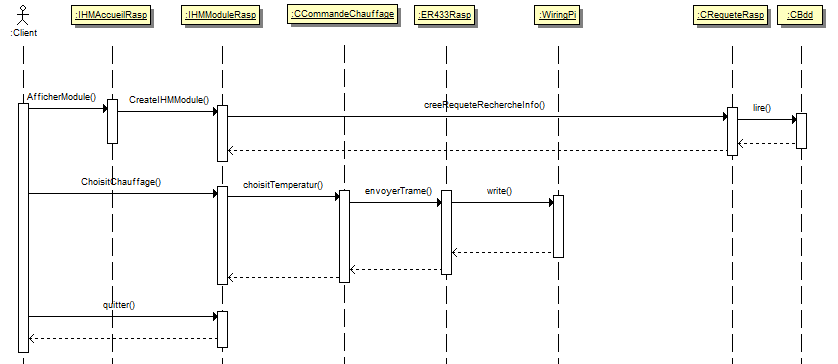


Comme nous pouvons le voir la classe CCommand a été suprimer dput a sont inutilité, la classe CMonSocket quant à elle a changer ses liaisons, en effet elle n’est plus liée diretement aux classes de command mais à la principale classe de ce diagramme de classe CGetion qui est une nouvelle classe. Cette classe gére le flux dans le logiciel. Toutes le flux transite par cette classe pour savoir quelle sont les actions à effectuer, que ce soit en command ou en reception de données.

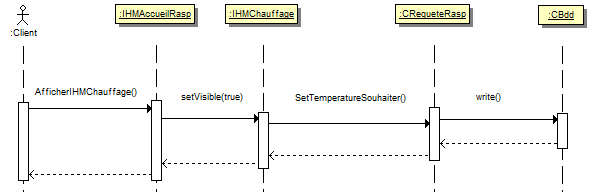
Partie package IHM :



Ancien diagramme de séquence Gestion chauffage center :

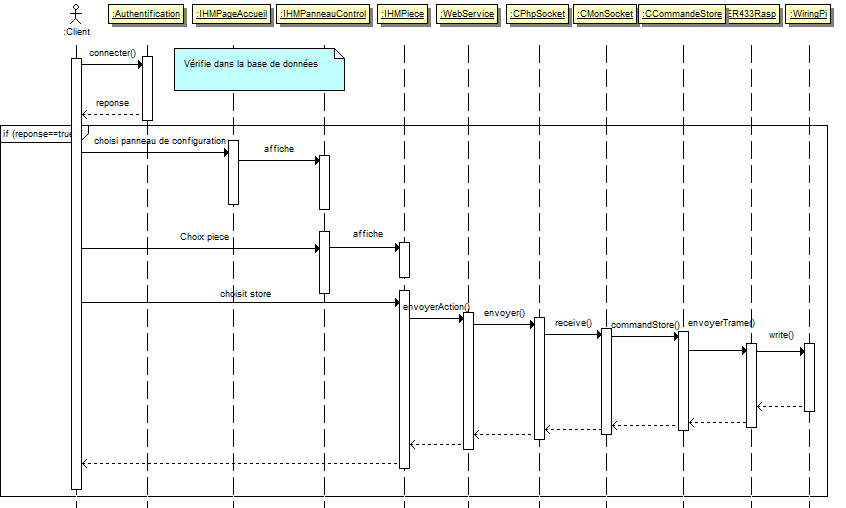


Nouveau diagramme de séquence Gestion de chauffage center :



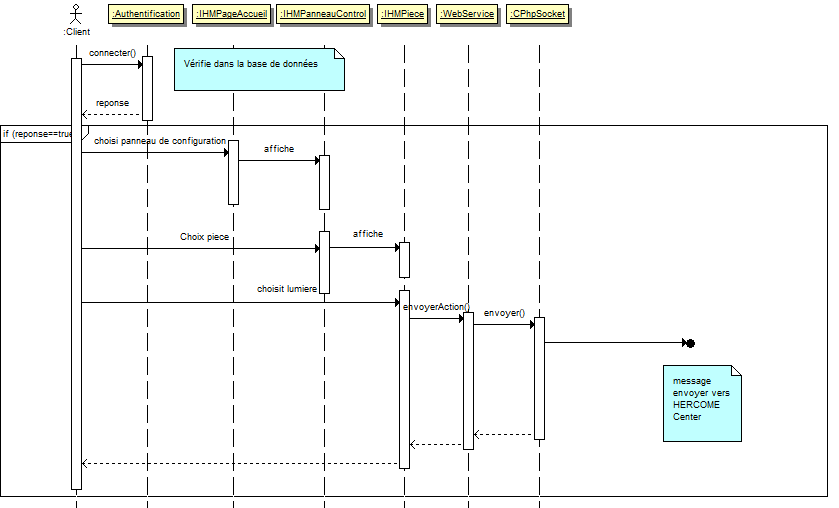
Grosse modification sur ce diagramme de séquence. En effet l’enregistrement de la valeur souhaiter de la valeur se fait uniquement sur la base de données.

Ancien diagramme de séquence Gestion composant (store) WEB :

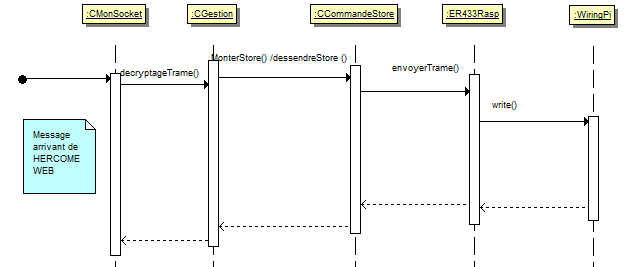


Nouveaux diagrammes de séquence composant (store) web vers center.

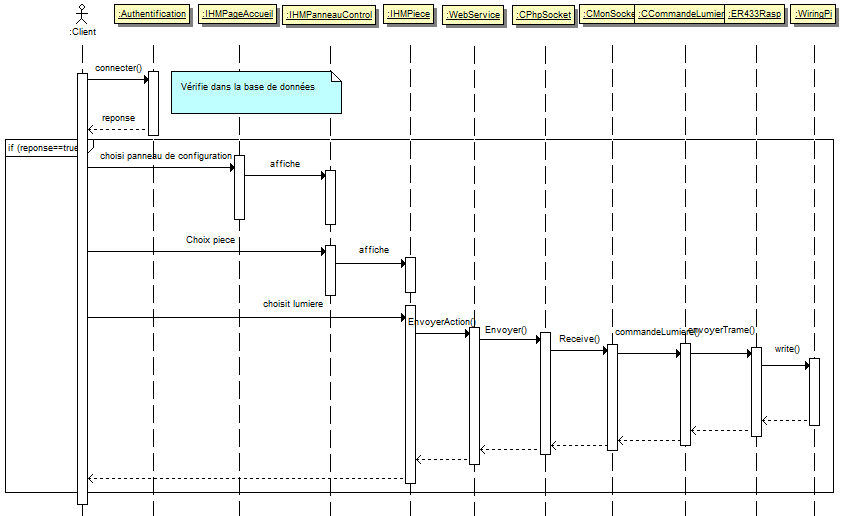
Départ Web



Arrivé center :

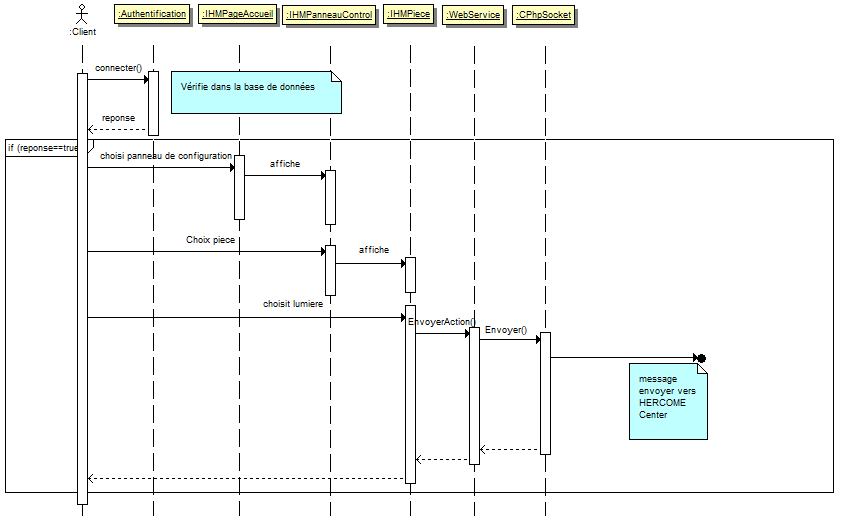


Ancien diagramme de séquence Gestion composant (lumière) WEB :

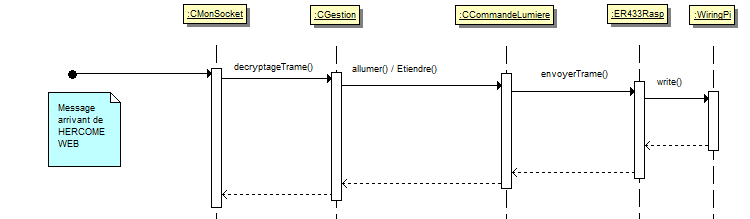


Nouveaux diagrammes de séquence composant (lumière) web vers center.

Départ Web

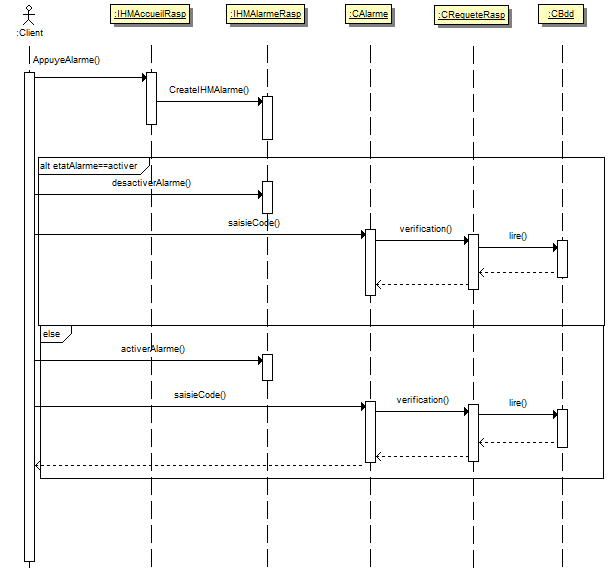


Arrivé web

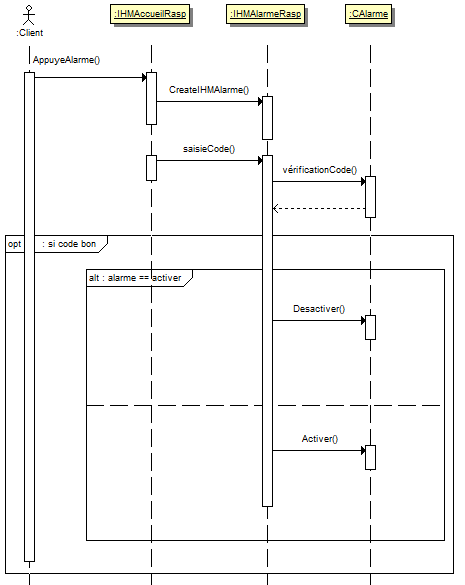


Pour toutes ces parties on remarque bien qu’il faut bien différencier les deux parties. En effet une fois sur CMonSocket on est arriver sur le programme HERCOME Center sur la classe CMonSocket. Qui est une fois reçut envoie la trame a CGestion qui redirigera le message vers la trame concerner.

Ancien diagramme de sequence gérer alarme HERCOME Center :

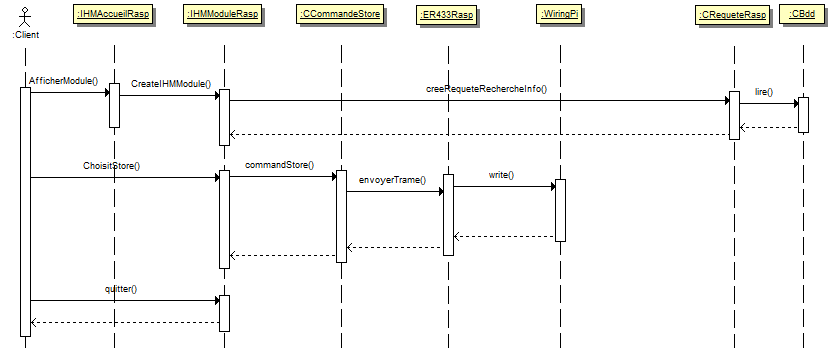


Nouveau diagramme de séquence gestion alarme HERCOME Center :

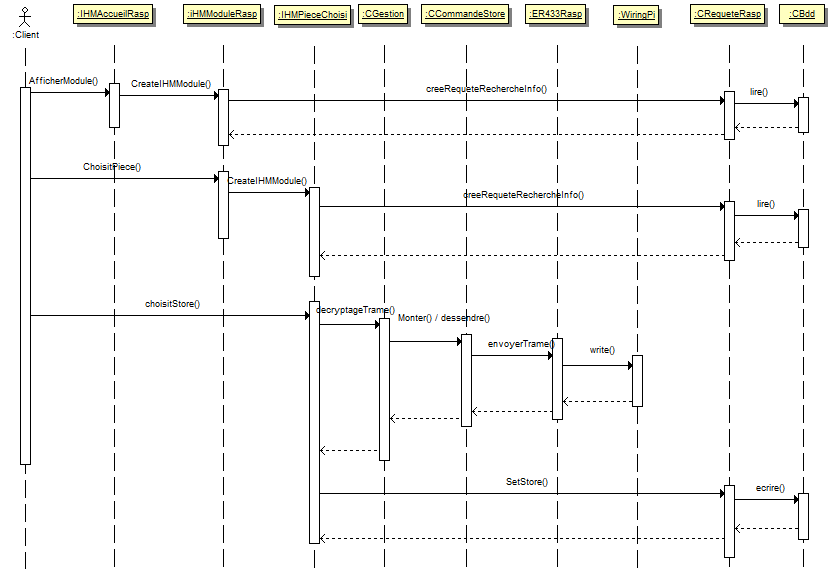


Le principal changement que nous remarquons dans le diagramme de séquence et la gestion de l’activation/ désactivation de l’alarme. Maintenant pour vérifier si le code est correct la classe CAlarme le fait grâce à un fichier quell lira. Pour l’activation/ désactivation de l’alarme l’état de l’alarme sera aussi modifier dans le fichier. Cela aura pour effet que s’il y a une eventuelle panne de courant que l’état soit bien rénitialiser. Et non que l’alarme se désactive après une petite panne de courant mettant les maisons à un risque de cambriolage.

Ancien diagramme de séquence gestion composant (store) HERCOME Center :

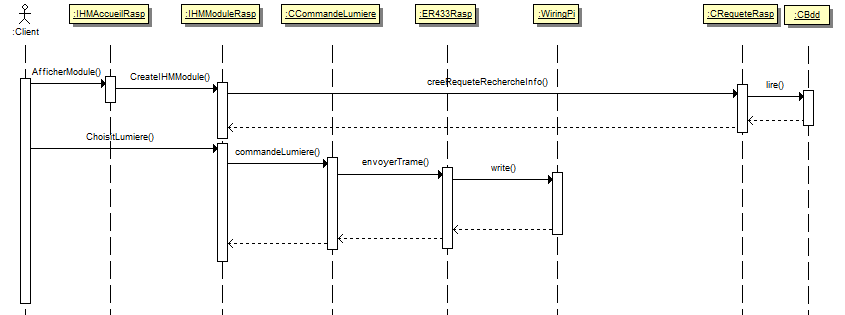


Nouveau diagramme de séquence gestion composant (store) HERCOME Center :

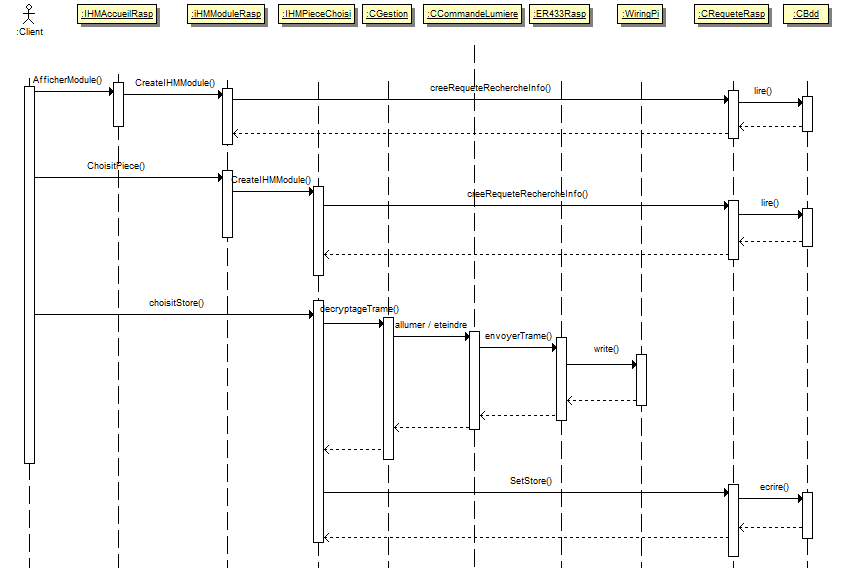


Le principal changement dans ces diagrammes de séquences sont l’ajouts des deux classes IHMPieceChoisi et CGestion. En effet maintenant pour choisir le store qui convient faudra choisir la piece que l’on veut (ici le nom IHMModuleRsp n’est donc plus approprié le changement sera fait), une fois la piéce choisit, IHMPieceChoisi répertorie l’ensemble des modules et donc de choisir le store sur lequel on veut intérrargir.

Ancien diagramme de séquence gestion composant (lumiere) HERCOME Center :



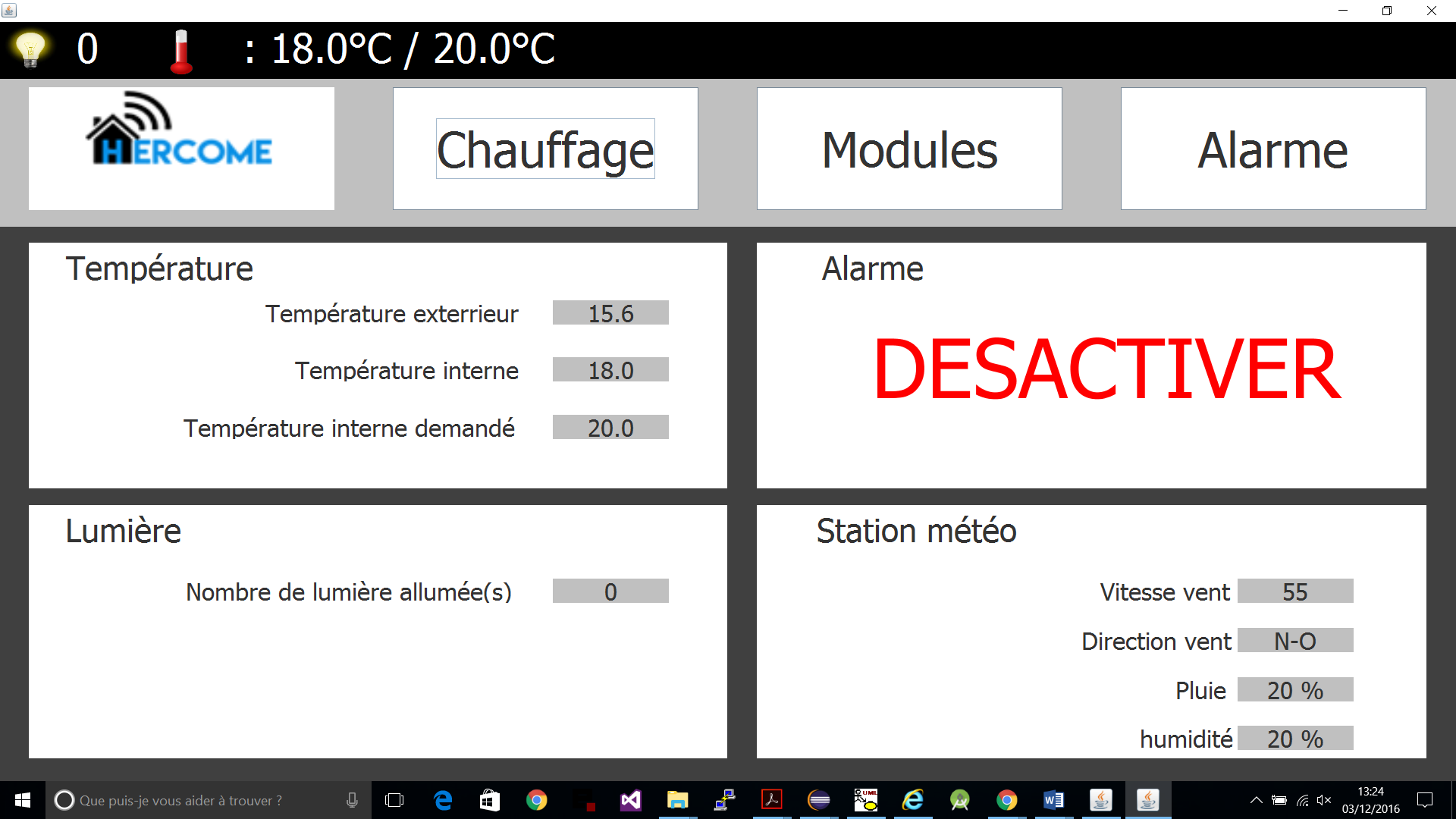
Nouveau diagramme de séquence gestion composant (lumiere) HERCOME Center :



Le principal changement dans ces diagrammes de séquences sont l’ajouts des deux classes IHMPieceChoisi et CGestion. En effet maintenant pour choisir le store qui convient faudra choisir la piece que l’on veut (ici le nom IHMModuleRsp n’est donc plus approprié le changement sera fait), une fois la piéce choisit, IHMPieceChoisi répertorie l’ensemble des modules et donc de choisir la lumière sur laquelle on veut intérrargir.

**Représentations des IHM**

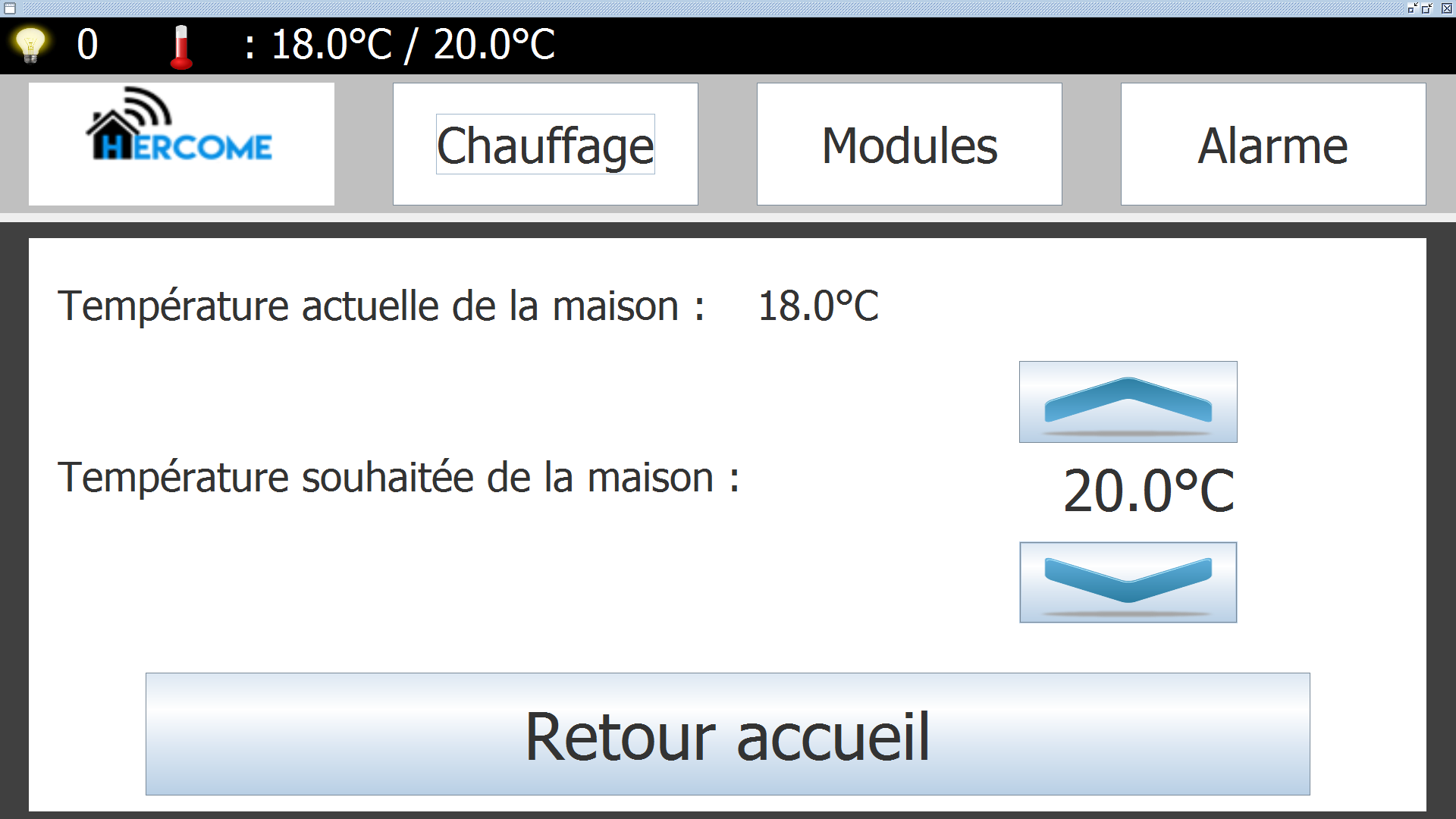
Voici les IHM du logiciel HERCOME Center :

Accueil :

Le code représentatif de cette IHM se trouve dans la partie annexe dans code HERCOME center Accueil.

Ici toutes les valeurs sont bien entendu des valeurs qui viennent de la classe CRequeteRasp, ainsi que l’état de l’alarme.

IHM chauffage :

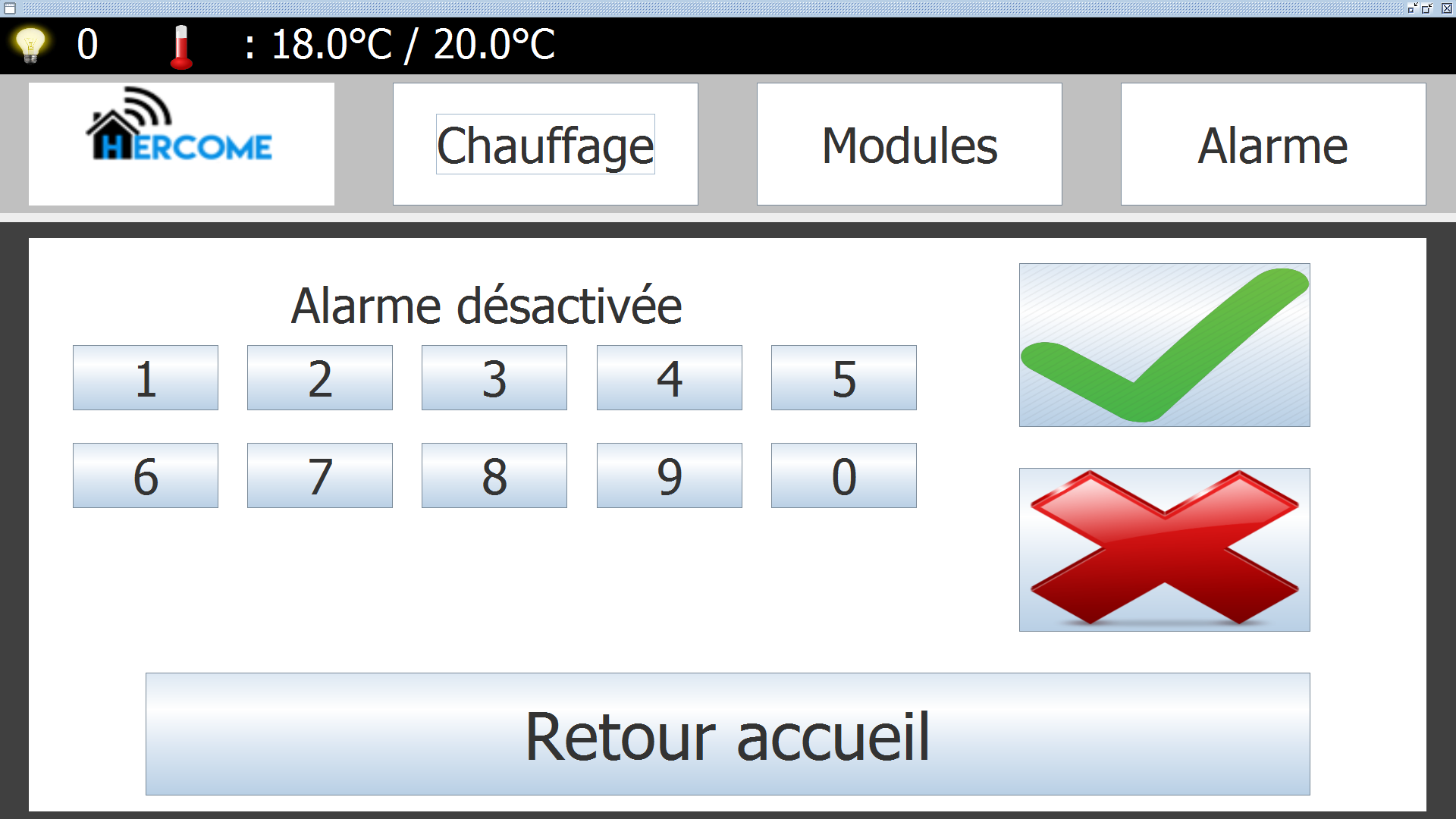
Une fois que l’on clique sur l’onglet chauffage en haut de l’application on arrive directement sur cette page.

Le code représentatif de cette IHM se trouve dans la partie annexe dans code HERCOME center Chauffage.

Encore une fois les valeurs misent sont celle qui viennent de CRequetRasp. Par contre toutes les valeurs et modifications sont pour l’instant enregistrer dans des attributs pour simuler le comportement des différentes classes. En effet la base de données n’était pas encore finie lors du développement et des tests.

Lorsque que l’on appuie sur les flèches du haut ou du bas cela monte ou descend la valeur de la température que l’on souhaite dans la maison.

IHM Alarme :

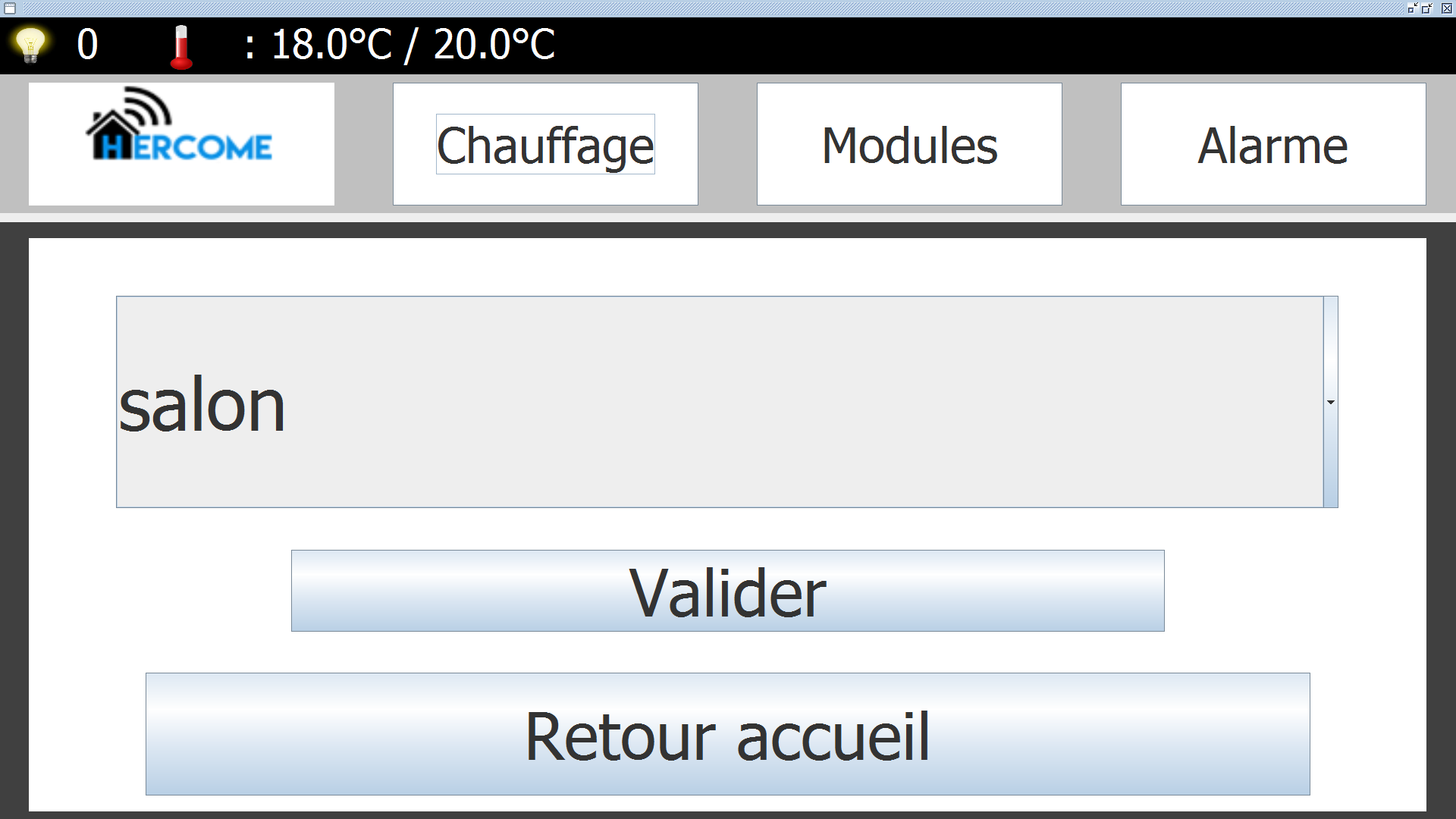
Une fois que l’on clique sur l’onglet Alarme en haut de l’application on arrive directement sur cette page.

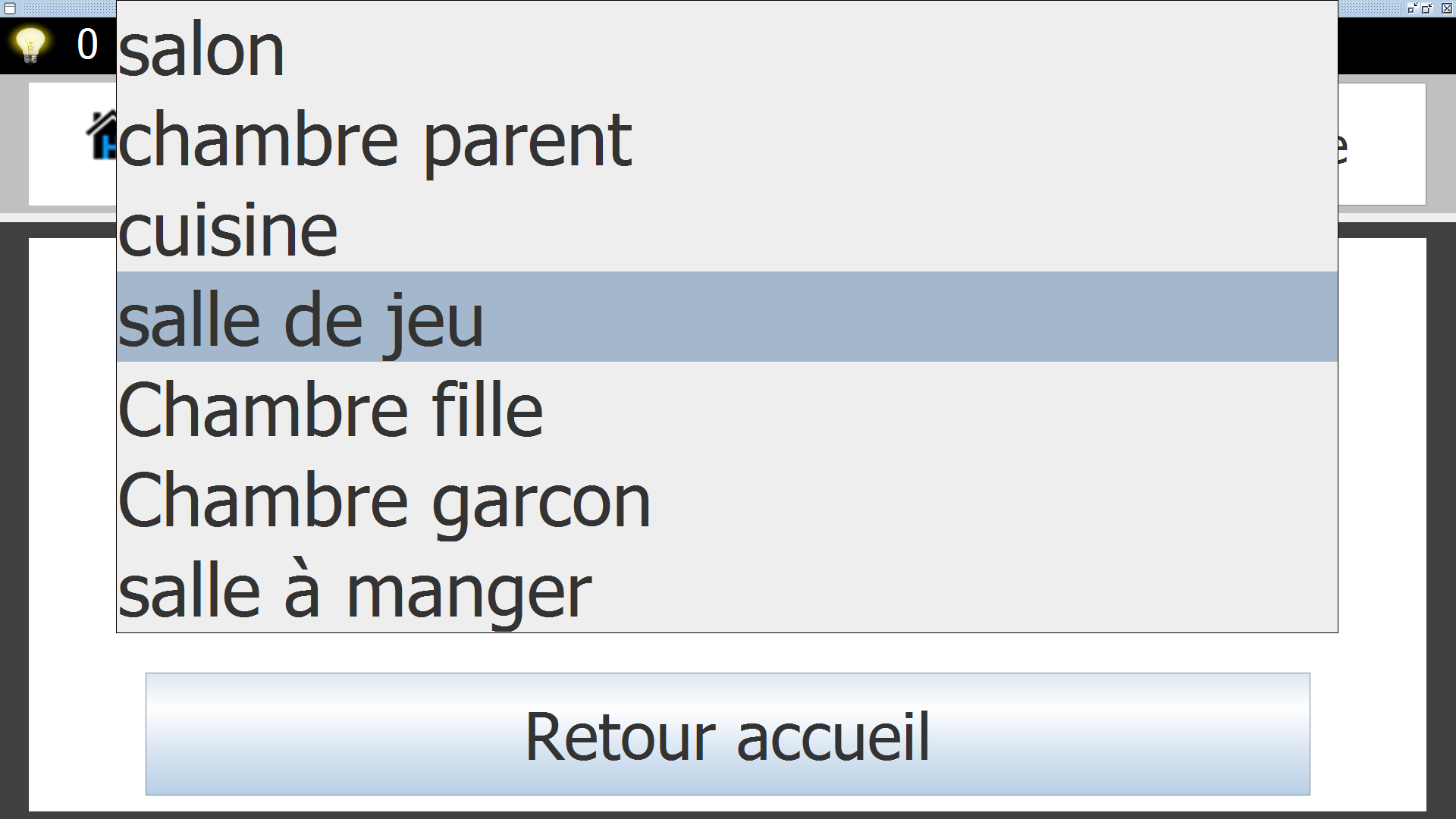
Le code représentatif de cette IHM se trouve dans la partie annexe dans code HERCOME center Alarme.

Cette IHM subira des changements en effet pour l’instant un code a été en dur est il n’y a qu’une saisie pour activer ou non l’alarme. La prochaine modification sera le rajout pour la modification du code tout en gardant l’activation ou non de l’alarme. La forme de l’IHM sera totalement modifiée également.

IHM ChoixPiece :

Une fois que l’on clique sur l’onglet Modules en haut de l’application on arrive directement sur cette page.





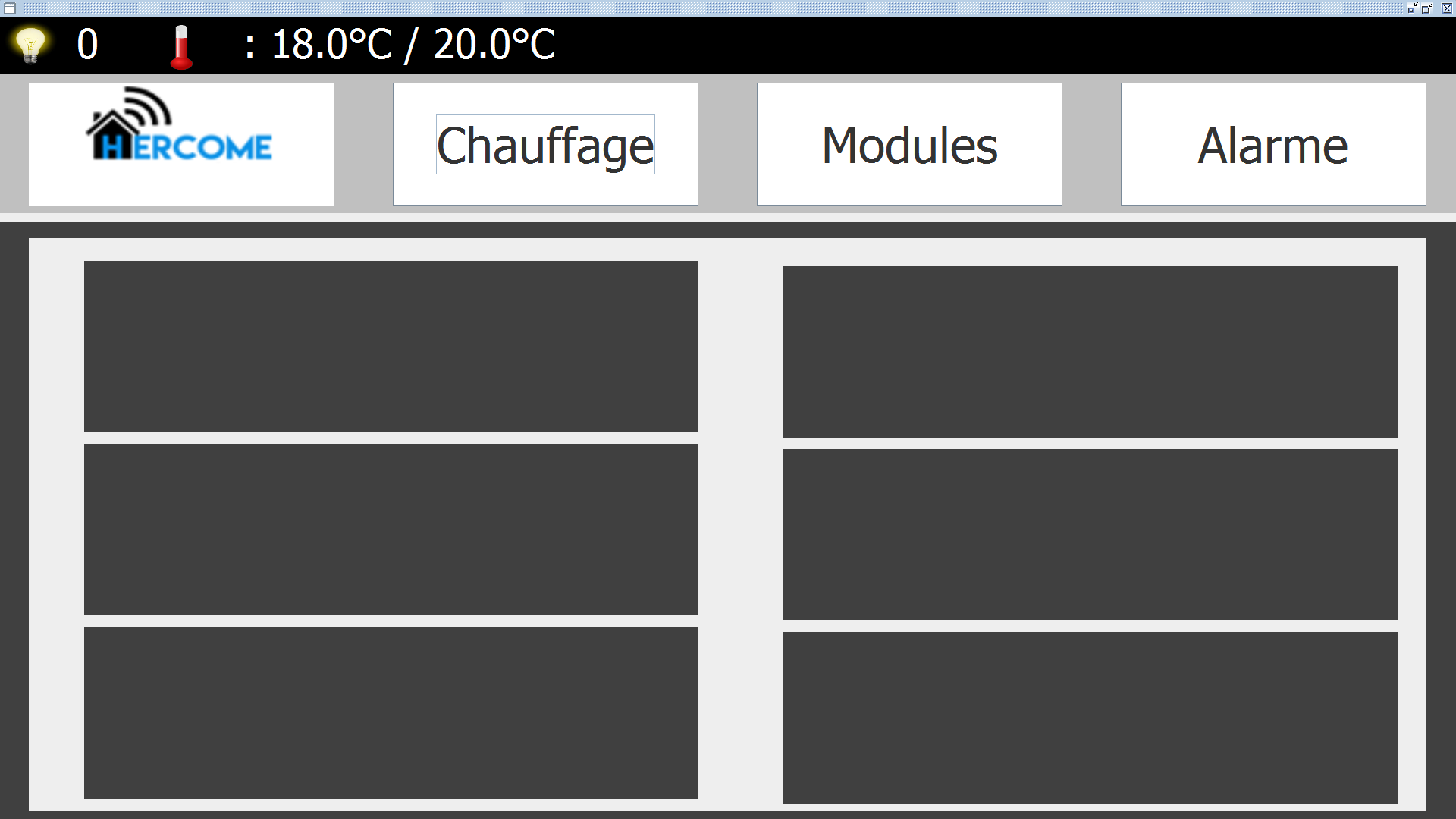
Le code représentatif de cette IHM se trouve dans la partie annexe dans code HERCOME center CHoixModule.

Cette IHM sert à choisir la pièce dans laquelle on veut interagir et gérer les modules.

Toutes les pièces indiquer seront celle enregistrer dans la base de données. Pour l’instant de même que pour les autres IHM les valeurs viennent des attributs de la classe CRequeteRasp.

IHM PieceChoisi :

Une fois que l’on clique sur l’onglet module et que l’on choisit sa pièce on arrive directement sur cette page.

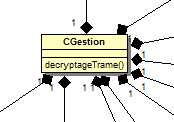


Le code représentatif de cette IHM se trouve dans la partie annexe dans code HERCOME center pieceChoisi.

Cette IHM n’est pas fini (problème de scrollPane). Quand cette IHM sera fini le rectangle en gris représentera les modules présents dans la pièce. Le nombre grandiront celons le nombre de modules présents (d’où l’utilité de la scroll bar). Une fois le moules indiquer on pourra interagir donc soit monter/ descendre le store, éteindre/ allumer la lumière.

**Explication des classes spécifiques**

La classe CGestion :



La classe CGestion est la classe qui dirigera le flux d’information vers les commande de module et lumière.

Comme il a été indiqué au début du rapport la trame a une certaine composition.

Le but de cette classe avec la méthode decryptageTrame est de recherche le caractère dans la trame qui indique vers quelle module la trame doit être transmise.

**public** **float** decryptageTrame(**char** [] trame){

**float** retour= (**float**) 0.0;

**switch** (trame[4]){

Ici on recherche le caractère qui correspond au module (ici en position 4 sur le tableau de caractère).

Ensuite une fois la que l’on sait vers quel module on doit envoyer la trame la classe décrypte aussi les données dedans.

**case** '3' :

CCommandLumiere mesLumieres = **new** CCommandLumiere();

**if** (trame[8]=='0'){

mesLumieres.allumerLumiere(trame);

maRequete.setEtatLamp(trame[1]\*100+trame[2]\*10+trame[3], **true**);

}

**else**{

mesLumieres.eteindreLumiere(trame);

maRequete.setEtatLamp(trame[1]\*100+trame[2]\*10+trame[3], **false**);

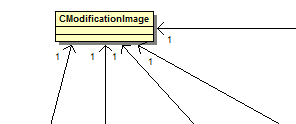
}

**break**;

Sur cet exemple on montre une fois la trame arriver on retire l’info du type de trame que cela est. Puis une fois arriver dans le type quelle corespond et on retire les données pour savoir quel ordre a été donnée comme montre ci-dessus soit allumer, soit éteindre.

La classe CModificationImage :

La classes CModificationImages est ici pour mettre les différentes images sur l’interfaces graphique à la taille qui convient.



Cette classe prend l’image la modifie à ma taille choisi est renvoie l’image sur les IHM.

**public** Image scaleImage(Image source, **int** width, **int** height) {

BufferedImage image = **new** BufferedImage(width, height, BufferedImage.***TYPE\_INT\_ARGB***);

Graphics2D g = (Graphics2D) image.getGraphics();

g.setRenderingHint(RenderingHints.***KEY\_INTERPOLATION***, RenderingHints.***VALUE\_INTERPOLATION\_BILINEAR***);

g.drawImage(source, 0, 0, width, height, **null**);

g.dispose();

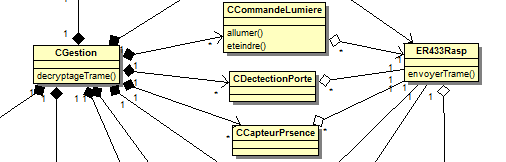
**return** image;

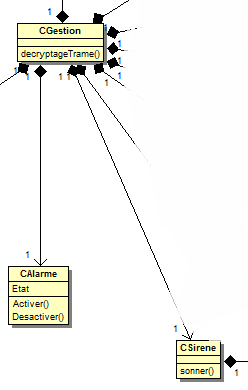
}

Voici la méthode principale de cette classe elle nécéssite l’image ainsi que la largeur, hauteur de l’image. Puis elle retourne l’image modifiée.

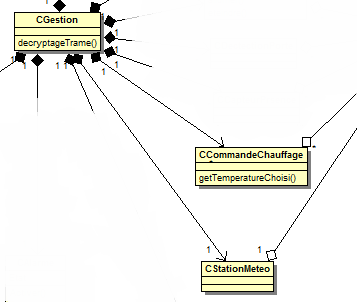
Les classes CDetectionPorte/ CCapteurPresence et CAlarme / CSirene

Les classes CDetectionPorte CCapteurPresence sont des classes qui seront solicité régulierement par les modules externes à travers les classes de communication (WiringPI ER433Rasp). Une fois solicité si une présence/ porte ouverte est détecter cette classe communiquera avec la classe CGestion.



La classe CGestion comme vu précédement est ici pour décrypter une fois décrypter, elle regardera sur la classe CAlarme si l’alarme est activée. Si elle est activée la classes CGestion actionnera la méthode sonner de la classe CSirene.

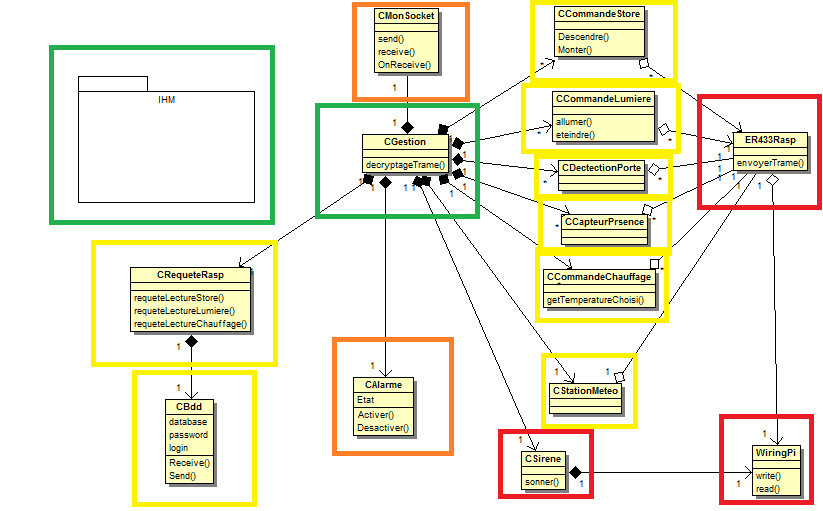
Les classes CStationMeteo et CChauffage :

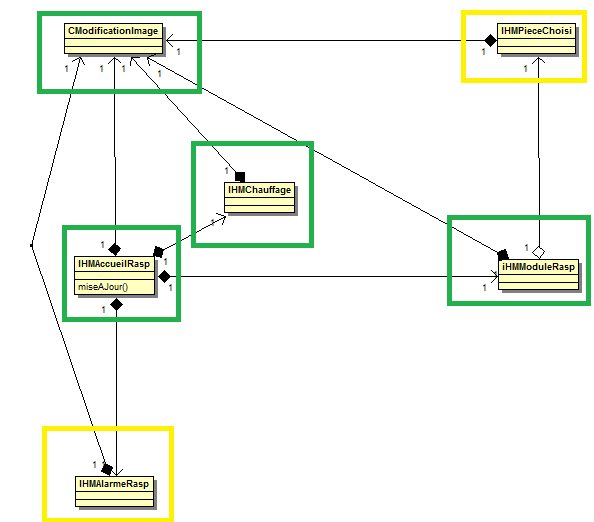
Les classes CChauffages et CStationMeteo sont ici pour faire la remontée des informations des différents modules vers la base de données (température, humidité etc).

Une fois l’information arrivée vers la classe ER433Rasp, elle fait parvenir l’information vers la classe correspondant CCommandChauffage ou CStation. Puis vers CGestion qui fait fait le dernier cheminement vers la base de données en passant au préalable par la classes CRequeteRasp pour contruire la requete SQL dédiée.

**Suivit de l’avancement par rapports aux diagrammes de classe**

En vert les classes qui sont normalement fini (sauf changement à prévoir). En jaune les classes qui sont pas totalement fini d’être développé mais dont le travail dessus n’est plus très conséquent, et enfin les rouges se dont les classes doivent encore avoir pas mal de travail ou pas encore développer. Et les classes avec aucune couleur non pas encore été développer.





L’ensemble du code des différentes classes sont mis dans les documents annexe.

**Conclusion**

En guise de conclusion je peux dire que l’avancement du projet est bon. Lors de la derniere reunion il a été dit que nous avions un peu de retard. Un retard qui a été nécessaire pour la bonne finision des différents diagrammes UML cela a vraiment permit de pouvoir avancer plus rapidement sur le code, tout en ayant une très bonne compréhension et une très bonne visibilité de l’avancement.

Grâce à se sacrifice de temps par le passé, aujourd’hui le projet avance beaucoups plus vite. Nous ne sommes ni en retard ni en avance. Le retard que nous avins a donc été ratrappé.

Sur les semaine avenir l’IHM sur les piéces choisi sera fini (une légere perte de temps dût à un problème de scroll bar). Ensuite la connexion à la base de données sera faite pour ensuite par la suite finir les différentes classes de commandes et de reception des modules.

Une fois HERCOME Center développer il ne restera plus qu’a développer le logiciel gestion de barriere. Ce logiciel sera développé en C++.

**Annexe**

IHM HERCOME Center Accueil :

package view;

//import java.awt.BorderLayout;

import java.awt.EventQueue;

import java.awt.Image;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JLabel;

import javax.swing.SwingConstants;

import java.awt.Font;

import java.awt.Frame;

import javax.swing.JButton;

import java.awt.Color;

//import javax.swing.Box;

import java.awt.Dimension;

//importation classe besoin

import model.CRequeteRasp;

import model.CModificationImage;

import view.\*;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class IHMAccueilRasp extends JFrame {

/\*\*

\*

\*/

private static final long serialVersionUID = 8308205125237951841L;

private JPanel contentPane;

private CModificationImage modifImg;

private static CRequeteRasp maRequete=new CRequeteRasp();

private String temp;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

IHMAccueilRasp frame = new IHMAccueilRasp(maRequete);

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public IHMAccueilRasp(CRequeteRasp req) {

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 543, 416);

contentPane = new JPanel();

contentPane.setBackground(Color.LIGHT\_GRAY);

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

maRequete = req;

modifImg = new CModificationImage();

this.pack();

JFrame.setDefaultLookAndFeelDecorated(true);

setExtendedState(Frame.MAXIMIZED\_BOTH);

Dimension dimension = java.awt.Toolkit.getDefaultToolkit().getScreenSize();

int height = (int)dimension.getHeight();

int width = (int)dimension.getWidth();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie menu de l'application evc le logo

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblNewLabel = new JLabel("");

lblNewLabel.setBounds(2\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

lblNewLabel.setOpaque(true);

lblNewLabel.setBackground(Color.WHITE);

Image img = new ImageIcon(this.getClass().getResource("/logoHercome.jpg")).getImage();

contentPane.setLayout(null);

lblNewLabel.setIcon(new ImageIcon(this.modifImg.scaleImage(img, 21\*width/100, 15\*height/100)));

contentPane.add(lblNewLabel);

JButton btnChauffage = new JButton("Chauffage");

btnChauffage.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

IHMChauffageRasp chauffage = new IHMChauffageRasp(maRequete);

chauffage.setVisible(true);

dispose();

}

});

btnChauffage.setBounds(27\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnChauffage.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnChauffage.setBackground(Color.WHITE);

contentPane.add(btnChauffage);

JButton btnModules = new JButton("Modules");

btnModules.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

IHMModuleRasp module = new IHMModuleRasp(maRequete);

module.setVisible(true);

dispose();

}

});

btnModules.setBounds(52\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnModules.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnModules.setBackground(Color.WHITE);

contentPane.add(btnModules);

JButton btnAlarme = new JButton("Alarme");

btnAlarme.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

IHMAlarmeRasp alarme = new IHMAlarmeRasp(maRequete);

alarme.setVisible(true);

dispose();

}

});

btnAlarme.setBounds(77\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnAlarme.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnAlarme.setBackground(Color.WHITE);

contentPane.add(btnAlarme);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie texte information de l'application

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie bande noir

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblInfoImgAmp = new JLabel ("");

lblInfoImgAmp.setBounds(1, 1, 4\*width/100, 6\*height/100);

lblInfoImgAmp.setOpaque(true);

lblInfoImgAmp.setBackground(Color.WHITE);

Image iconAmpoule = new ImageIcon(this.getClass().getResource("/iconAmpoule.jpg")).getImage();

contentPane.setLayout(null);

lblInfoImgAmp.setIcon(new ImageIcon(this.modifImg.scaleImage(iconAmpoule, 4\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgAmp);

this.temp = Integer.toString(maRequete.getNombreDeLum());

JLabel lblNbLum = new JLabel(this.temp);

lblNbLum.setBounds(5\*width/100, 1, 2\*width/100, 6\*height/100);

lblNbLum.setHorizontalAlignment(SwingConstants.CENTER);

lblNbLum.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

lblNbLum.setForeground(Color.WHITE);

contentPane.add(lblNbLum);

JLabel lblInfoImgTemp = new JLabel ("");

lblInfoImgTemp.setBounds(11\*width/100, 1\*height/100, 3\*width/100, 6\*height/100);

lblInfoImgTemp.setOpaque(true);

lblInfoImgTemp.setBackground(Color.BLACK);

Image iconTemp = new ImageIcon(this.getClass().getResource("/iconTemp.png")).getImage();

contentPane.setLayout(null);

lblInfoImgTemp.setIcon(new ImageIcon(this.modifImg.scaleImage(iconTemp, 3\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgTemp);

this.temp = " : "+Float.toString(maRequete.getTemperatureIntern())+"°C";

this.temp += " / "+Float.toString(maRequete.getTemperatureSouhaiter())+"°C";

JLabel lblTemp = new JLabel(this.temp);

lblTemp.setBounds(14\*width/100, 1, 26\*width/100, 6\*height/100);

lblTemp.setHorizontalAlignment(SwingConstants.CENTER);

lblTemp.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

lblTemp.setForeground(Color.WHITE);

contentPane.add(lblTemp);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie temperature

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblTemperature = new JLabel("Temp\u00E9rature");

lblTemperature.setBounds(4\*width/100, 28\*height/100, 14\*width/100, 4\*height/100);

lblTemperature.setHorizontalAlignment(SwingConstants.CENTER);

lblTemperature.setFont(new Font("Tahoma", Font.PLAIN, 4\*height/100));

//lblTemperature.setBackground(Color.WHITE);

contentPane.add(lblTemperature);

JLabel lblTemperatureExterne = new JLabel("Temp\u00E9rature exterrieur");

lblTemperatureExterne.setBounds(18\*width/100, 34\*height/100, 18\*width/100, 3\*height/100);

lblTemperatureExterne.setHorizontalAlignment(SwingConstants.CENTER);

lblTemperatureExterne.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblTemperatureExterne);

JLabel lblResultatTempExt = new JLabel(Float.toString(maRequete.getTemperatureExtern()));

lblResultatTempExt.setBounds(38\*width/100, 34\*height/100, 8\*width/100, 3\*height/100);

lblResultatTempExt.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatTempExt.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatTempExt.setOpaque(true);

lblResultatTempExt.setBackground(Color.LIGHT\_GRAY);

contentPane.add(lblResultatTempExt);

JLabel lblTemperatureInterne = new JLabel("Temp\u00E9rature interne");

lblTemperatureInterne.setBounds(19\*width/100, 41\*height/100, 18\*width/100, 3\*height/100);

lblTemperatureInterne.setHorizontalAlignment(SwingConstants.CENTER);

lblTemperatureInterne.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblTemperatureInterne);

JLabel lblResultatTempInt = new JLabel(Float.toString(maRequete.getTemperatureIntern()));

lblResultatTempInt.setBounds(38\*width/100, 41\*height/100, 8\*width/100, 3\*height/100);

lblResultatTempInt.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatTempInt.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatTempInt.setOpaque(true);

lblResultatTempInt.setBackground(Color.LIGHT\_GRAY);

contentPane.add(lblResultatTempInt);

JLabel lblTemperatureInterneDemande = new JLabel("Temp\u00E9rature interne demandé");

lblTemperatureInterneDemande.setBounds(12\*width/100, 48\*height/100, 24\*width/100, 3\*height/100);

lblTemperatureInterneDemande.setHorizontalAlignment(SwingConstants.CENTER);

lblTemperatureInterneDemande.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblTemperatureInterneDemande);

JLabel lblResultatTempIntDem = new JLabel(Float.toString(maRequete.getTemperatureSouhaiter()));

lblResultatTempIntDem.setBounds(38\*width/100, 48\*height/100, 8\*width/100, 3\*height/100);

lblResultatTempIntDem.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatTempIntDem.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatTempIntDem.setOpaque(true);

lblResultatTempIntDem.setBackground(Color.LIGHT\_GRAY);

contentPane.add(lblResultatTempIntDem);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie Alarme

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblAlarme = new JLabel("Alarme");

lblAlarme.setBounds(53\*width/100, 28\*height/100, 14\*width/100, 4\*height/100);

lblAlarme.setHorizontalAlignment(SwingConstants.CENTER);

lblAlarme.setFont(new Font("Tahoma", Font.PLAIN, 4\*height/100));

//lblTemperature.setBackground(Color.WHITE);

contentPane.add(lblAlarme);

if (maRequete.etatAlarme()==true){

JLabel lblEtatAlarme = new JLabel("ACTIVER");

lblEtatAlarme.setBounds(58\*width/100, 34\*height/100, 36\*width/100, 16\*height/100);

lblEtatAlarme.setHorizontalAlignment(SwingConstants.CENTER);

lblEtatAlarme.setFont(new Font("Tahoma", Font.PLAIN, 16\*height/100));

lblEtatAlarme.setForeground(Color.RED);

//lblTemperature.setBackground(Color.WHITE);

contentPane.add(lblEtatAlarme);

}

else{

JLabel lblEtatAlarme = new JLabel("DESACTIVER");

lblEtatAlarme.setBounds(58\*width/100, 34\*height/100, 36\*width/100, 16\*height/100);

lblEtatAlarme.setHorizontalAlignment(SwingConstants.CENTER);

lblEtatAlarme.setFont(new Font("Tahoma", Font.PLAIN, 10\*height/100));

lblEtatAlarme.setForeground(Color.RED);

//lblTemperature.setBackground(Color.WHITE);

contentPane.add(lblEtatAlarme);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie Lumiere

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblLumiere = new JLabel("Lumière");

lblLumiere.setBounds(4\*width/100, 60\*height/100, 9\*width/100, 4\*height/100);

lblLumiere.setHorizontalAlignment(SwingConstants.CENTER);

lblLumiere.setFont(new Font("Tahoma", Font.PLAIN, 4\*height/100));

//lblTemperature.setBackground(Color.WHITE);

contentPane.add(lblLumiere);

JLabel lblLumiereAllume = new JLabel("Nombre de lumière allumée(s)");

lblLumiereAllume.setBounds(12\*width/100, 68\*height/100, 24\*width/100, 3\*height/100);

lblLumiereAllume.setHorizontalAlignment(SwingConstants.CENTER);

lblLumiereAllume.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblLumiereAllume);

JLabel lblResultatLum = new JLabel("");

lblResultatLum.setBounds(38\*width/100, 68\*height/100, 8\*width/100, 3\*height/100);

lblResultatLum.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatLum.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatLum.setOpaque(true);

lblResultatLum.setBackground(Color.LIGHT\_GRAY);

this.temp = Integer.toString(maRequete.getNombreDeLum());

lblResultatLum.setText(this.temp);

contentPane.add(lblResultatLum);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie station meteo

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblStationMeteo = new JLabel("Station météo");

lblStationMeteo.setBounds(53\*width/100, 60\*height/100, 20\*width/100, 4\*height/100);

lblStationMeteo.setHorizontalAlignment(SwingConstants.CENTER);

lblStationMeteo.setFont(new Font("Tahoma", Font.PLAIN, 4\*height/100));

//lblTemperature.setBackground(Color.WHITE);

contentPane.add(lblStationMeteo);

JLabel lblVitesseVent = new JLabel("Vitesse vent");

lblVitesseVent.setBounds(75\*width/100, 68\*height/100, 10\*width/100, 3\*height/100);

lblVitesseVent.setHorizontalAlignment(SwingConstants.CENTER);

lblVitesseVent.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblVitesseVent);

this.temp = Integer.toString(maRequete.getVitesseVent());

JLabel lblResultatVitVent = new JLabel(this.temp);

lblResultatVitVent.setBounds(85\*width/100, 68\*height/100, 8\*width/100, 3\*height/100);

lblResultatVitVent.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatVitVent.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatVitVent.setOpaque(true);

lblResultatVitVent.setBackground(Color.LIGHT\_GRAY);

//lblResultatVitVent.setText(temp);

contentPane.add(lblResultatVitVent);

JLabel lblDirectionVent = new JLabel("Direction vent");

lblDirectionVent.setBounds(73\*width/100, 74\*height/100, 13\*width/100, 3\*height/100);

lblDirectionVent.setHorizontalAlignment(SwingConstants.CENTER);

lblDirectionVent.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblDirectionVent);

this.temp = maRequete.getDirectionVent();

JLabel lblResultatDirecVent = new JLabel(this.temp);

lblResultatDirecVent.setBounds(85\*width/100, 74\*height/100, 8\*width/100, 3\*height/100);

lblResultatDirecVent.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatDirecVent.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatDirecVent.setOpaque(true);

lblResultatDirecVent.setBackground(Color.LIGHT\_GRAY);

//lblResultatDirecVent.setText(temp);

contentPane.add(lblResultatDirecVent);

JLabel lblPluie = new JLabel("Pluie");

lblPluie.setBounds(76\*width/100, 80\*height/100, 13\*width/100, 3\*height/100);

lblPluie.setHorizontalAlignment(SwingConstants.CENTER);

lblPluie.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblPluie);

this.temp = Integer.toString(maRequete.getPluie());

JLabel lblResultatPluie = new JLabel(this.temp+" %");

lblResultatPluie.setBounds(85\*width/100, 80\*height/100, 8\*width/100, 3\*height/100);

lblResultatPluie.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatPluie.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatPluie.setOpaque(true);

lblResultatPluie.setBackground(Color.LIGHT\_GRAY);

//lblResultatDirecVent.setText(temp);

contentPane.add(lblResultatPluie);

JLabel lblHumidite = new JLabel("humidité");

lblHumidite.setBounds(75\*width/100, 86\*height/100, 13\*width/100, 3\*height/100);

lblHumidite.setHorizontalAlignment(SwingConstants.CENTER);

lblHumidite.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

//lblTemperatureExterne.setBackground(Color.WHITE);

contentPane.add(lblHumidite);

this.temp = Integer.toString(maRequete.getHumidite());

JLabel lblResultatHum = new JLabel(this.temp+" %");

lblResultatHum.setBounds(85\*width/100, 86\*height/100, 8\*width/100, 3\*height/100);

lblResultatHum.setHorizontalAlignment(SwingConstants.CENTER);

lblResultatHum.setFont(new Font("Tahoma", Font.PLAIN, 3\*height/100));

lblResultatHum.setOpaque(true);

lblResultatHum.setBackground(Color.LIGHT\_GRAY);

//lblResultatDirecVent.setText(temp);

contentPane.add(lblResultatHum);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie des differant rectangle de couleur de l'application pour separer les differante partie

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//bande noir au dessus

JLabel lblBandeNoir = new JLabel("");

lblBandeNoir.setBounds(0,0,width,7\*height/100);

lblBandeNoir.setOpaque(true);

lblBandeNoir.setBackground(Color.BLACK);

//contentPane.setLayout(null);

contentPane.add(lblBandeNoir);

// rectangle gris claire sur les bouton menu

JLabel lblRecGris = new JLabel("");

lblRecGris.setBounds(0,7\*height/100,width,17\*height/100);

lblRecGris.setOpaque(true);

lblRecGris.setBackground(Color.LIGHT\_GRAY);

//contentPane.setLayout(null);

contentPane.add(lblRecGris);

// rectangle blanc des differente parti permettant le recap de la maison

JLabel lblRecTemp = new JLabel("");

lblRecTemp.setBounds(2\*width/100,27\*height/100,48\*width/100,30\*height/100);

lblRecTemp.setOpaque(true);

lblRecTemp.setBackground(Color.WHITE);

//contentPane.setLayout(null);

contentPane.add(lblRecTemp);

JLabel lblRecAlarme = new JLabel("");

lblRecAlarme.setBounds(52\*width/100,27\*height/100,46\*width/100,30\*height/100);

lblRecAlarme.setOpaque(true);

lblRecAlarme.setBackground(Color.WHITE);

//contentPane.setLayout(null);

contentPane.add(lblRecAlarme);

JLabel lblRecLum = new JLabel("");

lblRecLum.setBounds(2\*width/100,59\*height/100,48\*width/100,31\*height/100);

lblRecLum.setOpaque(true);

lblRecLum.setBackground(Color.WHITE);

//contentPane.setLayout(null);

contentPane.add(lblRecLum);

JLabel lblRecStationMeteo = new JLabel("");

lblRecStationMeteo.setBounds(52\*width/100,59\*height/100,46\*width/100,31\*height/100);

lblRecStationMeteo.setOpaque(true);

lblRecStationMeteo.setBackground(Color.WHITE);

//contentPane.setLayout(null);

contentPane.add(lblRecStationMeteo);

//rectangle gris foncer du reste de l'application

JLabel lblMiddleDarkGray = new JLabel("");

lblMiddleDarkGray.setBounds(0,25\*height/100,width,75\*height/100);

lblMiddleDarkGray.setOpaque(true);

lblMiddleDarkGray.setBackground(Color.DARK\_GRAY);

//contentPane.setLayout(null);

contentPane.add(lblMiddleDarkGray);

}

}

IHM HERCOME Center Chauffage :

package view;

import java.awt.BorderLayout;

import java.awt.Color;

import java.awt.Dimension;

import java.awt.EventQueue;

import java.awt.Font;

import java.awt.Frame;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.SwingConstants;

import javax.swing.border.EmptyBorder;

import model.CModificationImage;

import model.CRequeteRasp;

public class IHMChauffageRasp extends JFrame {

private JPanel contentPane;

private CModificationImage modifImg;

private CRequeteRasp maRequete;

private String temp;

/\*\*

\* Launch the application.

\*/

/\*\*

\* Create the frame.

\*/

public IHMChauffageRasp(CRequeteRasp req) {

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 450, 300);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

contentPane.setLayout(new BorderLayout(0, 0));

setContentPane(contentPane);

maRequete = req;

modifImg = new CModificationImage();

this.pack();

JFrame.setDefaultLookAndFeelDecorated(true);

setExtendedState(Frame.MAXIMIZED\_BOTH);

Dimension dimension = java.awt.Toolkit.getDefaultToolkit().getScreenSize();

int height = (int)dimension.getHeight();

int width = (int)dimension.getWidth();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie menu de l'application evc le logo

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblNewLabel = new JLabel("");

lblNewLabel.setBounds(2\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

lblNewLabel.setOpaque(true);

lblNewLabel.setBackground(Color.WHITE);

Image img = new ImageIcon(this.getClass().getResource("/logoHercome.jpg")).getImage();

contentPane.setLayout(null);

lblNewLabel.setIcon(new ImageIcon(this.modifImg.scaleImage(img, 21\*width/100, 15\*height/100)));

contentPane.add(lblNewLabel);

JButton btnChauffage = new JButton("Chauffage");

btnChauffage.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

//IHMChauffageRasp chauffage = new IHMChauffageRasp();

//chauffage.setVisible(true);

}

});

btnChauffage.setBounds(27\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnChauffage.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnChauffage.setBackground(Color.WHITE);

contentPane.add(btnChauffage);

JButton btnModules = new JButton("Modules");

btnModules.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

IHMModuleRasp module = new IHMModuleRasp(maRequete);

module.setVisible(true);

dispose();

}

});

btnModules.setBounds(52\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnModules.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnModules.setBackground(Color.WHITE);

contentPane.add(btnModules);

JButton btnAlarme = new JButton("Alarme");

btnAlarme.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

IHMAlarmeRasp alarme = new IHMAlarmeRasp(maRequete);

alarme.setVisible(true);

dispose();

}

});

btnAlarme.setBounds(77\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnAlarme.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnAlarme.setBackground(Color.WHITE);

contentPane.add(btnAlarme);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie texte information de l'application

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie bande noir

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblInfoImgAmp = new JLabel ("");

lblInfoImgAmp.setBounds(1, 1, 4\*width/100, 6\*height/100);

lblInfoImgAmp.setOpaque(true);

lblInfoImgAmp.setBackground(Color.WHITE);

Image iconAmpoule = new ImageIcon(this.getClass().getResource("/iconAmpoule.jpg")).getImage();

contentPane.setLayout(null);

lblInfoImgAmp.setIcon(new ImageIcon(this.modifImg.scaleImage(iconAmpoule, 4\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgAmp);

this.temp = Integer.toString(maRequete.getNombreDeLum());

JLabel lblNbLum = new JLabel(this.temp);

lblNbLum.setBounds(5\*width/100, 1, 2\*width/100, 6\*height/100);

lblNbLum.setHorizontalAlignment(SwingConstants.CENTER);

lblNbLum.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

lblNbLum.setForeground(Color.WHITE);

contentPane.add(lblNbLum);

JLabel lblInfoImgTemp = new JLabel ("");

lblInfoImgTemp.setBounds(11\*width/100, 1\*height/100, 3\*width/100, 6\*height/100);

lblInfoImgTemp.setOpaque(true);

lblInfoImgTemp.setBackground(Color.BLACK);

Image iconTemp = new ImageIcon(this.getClass().getResource("/iconTemp.png")).getImage();

contentPane.setLayout(null);

lblInfoImgTemp.setIcon(new ImageIcon(this.modifImg.scaleImage(iconTemp, 3\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgTemp);

this.temp = " : "+Float.toString(maRequete.getTemperatureIntern())+"°C";

this.temp += " / "+Float.toString(maRequete.getTemperatureSouhaiter())+"°C";

JLabel lblTemp = new JLabel(this.temp);

lblTemp.setBounds(14\*width/100, 1, 26\*width/100, 6\*height/100);

lblTemp.setHorizontalAlignment(SwingConstants.CENTER);

lblTemp.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

lblTemp.setForeground(Color.WHITE);

contentPane.add(lblTemp);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie centrale

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblTextTemp = new JLabel("Température actuelle de la maison :");

lblTextTemp.setBounds(4\*width/100, 30\*height/100, 46\*width/100, 10\*height/100);

//lblTextTemp.setHorizontalAlignment(SwingConstants.CENTER);

lblTextTemp.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

contentPane.add(lblTextTemp);

JLabel lblResultatTemp = new JLabel (Float.toString(maRequete.getTemperatureIntern())+"°C");

lblResultatTemp.setBounds(52\*width/100, 30\*height/100, 10\*width/100, 10\*height/100);

lblResultatTemp.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

contentPane.add(lblResultatTemp);

JLabel lblTextTempSouhait = new JLabel("Température souhaiter de la maison :");

lblTextTempSouhait.setBounds(4\*width/100, 51\*height/100, 50\*width/100, 10\*height/100);

//lblTextTemp.setHorizontalAlignment(SwingConstants.CENTER);

lblTextTempSouhait.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

contentPane.add(lblTextTempSouhait);

JLabel lblResultatTempSouhait = new JLabel(Float.toString(maRequete.getTemperatureSouhaiter())+"°C");

lblResultatTempSouhait.setBounds(73\*width/100, 50\*height/100, 15\*width/100, 15\*height/100);

lblResultatTempSouhait.setFont(new Font("Tahoma", Font.PLAIN, 7\*height/100));

contentPane.add(lblResultatTempSouhait);

Image imgUp = new ImageIcon(this.getClass().getResource("/iconUp.png")).getImage();

JButton btnTempUp = new JButton(new ImageIcon(this.modifImg.scaleImage(imgUp, 15\*width/100, 10\*height/100)));

btnTempUp.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

maRequete.setTemperatureSouhaiterPlus();

lblResultatTempSouhait.setText(Float.toString(maRequete.getTemperatureSouhaiter())+"°C");

lblTemp.setText(": "+Float.toString(maRequete.getTemperatureIntern())+"°C / "+Float.toString(maRequete.getTemperatureSouhaiter())+"°C");

//contentPane.add(lblResultatTempSouhait);

}

});

btnTempUp.setBounds(70\*width/100,42\*height/100, 15\*width/100, 10\*height/100);

btnTempUp.setFont(new Font("Tahoma", Font.PLAIN, 8\*height/100));

contentPane.add(btnTempUp);

Image imgDown = new ImageIcon(this.getClass().getResource("/iconDown.png")).getImage();

JButton btnTempDown = new JButton(new ImageIcon(this.modifImg.scaleImage(imgDown, 15\*width/100, 10\*height/100)));

btnTempDown.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

maRequete.setTemperatureSouhaiterMoin();

lblResultatTempSouhait.setText(Float.toString(maRequete.getTemperatureSouhaiter())+"°C");

lblTemp.setText(": "+Float.toString(maRequete.getTemperatureIntern())+"°C / "+Float.toString(maRequete.getTemperatureSouhaiter())+"°C");

//contentPane.add(lblResultatTempSouhait);

}

});

btnTempDown.setBounds(70\*width/100,64\*height/100, 15\*width/100, 10\*height/100);

btnTempDown.setFont(new Font("Tahoma", Font.PLAIN, 8\*height/100));

contentPane.add(btnTempDown);

JButton btnQuitter = new JButton("Retour accueil");

btnQuitter.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

IHMAccueilRasp accueil = new IHMAccueilRasp(maRequete);

accueil.setVisible(true);

dispose();

}

});

btnQuitter.setBounds(10\*width/100,80\*height/100, 80\*width/100, 15\*height/100);

btnQuitter.setFont(new Font("Tahoma", Font.PLAIN, 8\*height/100));

contentPane.add(btnQuitter);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie des differant rectangle de couleur de l'application pour separer les differante partie

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//bande noir au dessus

JLabel lblBandeNoir = new JLabel("");

lblBandeNoir.setBounds(0,0,width,7\*height/100);

lblBandeNoir.setOpaque(true);

lblBandeNoir.setBackground(Color.BLACK);

//contentPane.setLayout(null);

contentPane.add(lblBandeNoir);

// rectangle gris claire sur les bouton menu

JLabel lblRecGris = new JLabel("");

lblRecGris.setBounds(0,7\*height/100,width,17\*height/100);

lblRecGris.setOpaque(true);

lblRecGris.setBackground(Color.LIGHT\_GRAY);

//contentPane.setLayout(null);

contentPane.add(lblRecGris);

// rectangle blanc des differente parti permettant le recap de la maison

JLabel lblRecTemp = new JLabel("");

lblRecTemp.setBounds(2\*width/100,27\*height/100,96\*width/100,70\*height/100);

lblRecTemp.setOpaque(true);

lblRecTemp.setBackground(Color.WHITE);

//contentPane.setLayout(null);

contentPane.add(lblRecTemp);

//rectangle gris foncer du reste de l'application

JLabel lblMiddleDarkGray = new JLabel("");

lblMiddleDarkGray.setBounds(0,25\*height/100,width,75\*height/100);

lblMiddleDarkGray.setOpaque(true);

lblMiddleDarkGray.setBackground(Color.DARK\_GRAY);

//contentPane.setLayout(null);

contentPane.add(lblMiddleDarkGray);

}

}

IHM HERCOME Center Alarme :

**package** view;

**import** java.awt.BorderLayout;

**import** java.awt.Color;

**import** java.awt.Dimension;

**import** java.awt.Font;

**import** java.awt.Frame;

**import** java.awt.Image;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.ImageIcon;

**import** javax.swing.JButton;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JPanel;

**import** javax.swing.SwingConstants;

**import** javax.swing.border.EmptyBorder;

**import** model.CModificationImage;

**import** model.CRequeteRasp;

**public** **class** IHMAlarmeRasp **extends** JFrame {

/\*\*

\*

\*/

**private** **static** **final** **long** ***serialVersionUID*** = 3935263116957716522L;

**private** JPanel contentPane;

**private** CModificationImage modifImg;

**private** **static** CRequeteRasp *maRequete*=**new** CRequeteRasp();

**private** String temp,temp2="";

/\*\*

\* Launch the application.

\*/

/\*\*

\* Create the frame.

\*/

**public** IHMAlarmeRasp(CRequeteRasp req) {

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

setBounds(100, 100, 450, 300);

contentPane = **new** JPanel();

contentPane.setBorder(**new** EmptyBorder(5, 5, 5, 5));

contentPane.setLayout(**new** BorderLayout(0, 0));

setContentPane(contentPane);

*maRequete* = req;

modifImg = **new** CModificationImage();

**this**.pack();

JFrame.*setDefaultLookAndFeelDecorated*(**true**);

setExtendedState(Frame.***MAXIMIZED\_BOTH***);

Dimension dimension = java.awt.Toolkit.*getDefaultToolkit*().getScreenSize();

**int** height = (**int**)dimension.getHeight();

**int** width = (**int**)dimension.getWidth();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie menu de l'application evc le logo

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblNewLabel = **new** JLabel("");

lblNewLabel.setBounds(2\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

lblNewLabel.setOpaque(**true**);

lblNewLabel.setBackground(Color.***WHITE***);

Image img = **new** ImageIcon(**this**.getClass().getResource("/logoHercome.jpg")).getImage();

contentPane.setLayout(**null**);

lblNewLabel.setIcon(**new** ImageIcon(**this**.modifImg.scaleImage(img, 21\*width/100, 15\*height/100)));

contentPane.add(lblNewLabel);

JButton btnChauffage = **new** JButton("Chauffage");

btnChauffage.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

IHMChauffageRasp chauffage = **new** IHMChauffageRasp(*maRequete*);

chauffage.setVisible(**true**);

dispose();

}

});

btnChauffage.setBounds(27\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnChauffage.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

btnChauffage.setBackground(Color.***WHITE***);

contentPane.add(btnChauffage);

JButton btnModules = **new** JButton("Modules");

btnModules.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent arg0) {

IHMModuleRasp module = **new** IHMModuleRasp(*maRequete*);

module.setVisible(**true**);

dispose();

}

});

btnModules.setBounds(52\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnModules.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

btnModules.setBackground(Color.***WHITE***);

contentPane.add(btnModules);

JButton btnAlarme = **new** JButton("Alarme");

btnAlarme.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

}

});

btnAlarme.setBounds(77\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnAlarme.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

btnAlarme.setBackground(Color.***WHITE***);

contentPane.add(btnAlarme);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie texte information de l'application

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie bande noir

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblInfoImgAmp = **new** JLabel ("");

lblInfoImgAmp.setBounds(1, 1, 4\*width/100, 6\*height/100);

lblInfoImgAmp.setOpaque(**true**);

lblInfoImgAmp.setBackground(Color.***WHITE***);

Image iconAmpoule = **new** ImageIcon(**this**.getClass().getResource("/iconAmpoule.jpg")).getImage();

contentPane.setLayout(**null**);

lblInfoImgAmp.setIcon(**new** ImageIcon(**this**.modifImg.scaleImage(iconAmpoule, 4\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgAmp);

**this**.temp = Integer.*toString*(*maRequete*.getNombreDeLum());

JLabel lblNbLum = **new** JLabel(**this**.temp);

lblNbLum.setBounds(5\*width/100, 1, 2\*width/100, 6\*height/100);

lblNbLum.setHorizontalAlignment(SwingConstants.***CENTER***);

lblNbLum.setFont(**new** Font("Tahoma", Font.***PLAIN***, 5\*height/100));

lblNbLum.setForeground(Color.***WHITE***);

contentPane.add(lblNbLum);

JLabel lblInfoImgTemp = **new** JLabel ("");

lblInfoImgTemp.setBounds(11\*width/100, 1\*height/100, 3\*width/100, 6\*height/100);

lblInfoImgTemp.setOpaque(**true**);

lblInfoImgTemp.setBackground(Color.***BLACK***);

Image iconTemp = **new** ImageIcon(**this**.getClass().getResource("/iconTemp.png")).getImage();

contentPane.setLayout(**null**);

lblInfoImgTemp.setIcon(**new** ImageIcon(**this**.modifImg.scaleImage(iconTemp, 3\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgTemp);

**this**.temp = " : "+Float.*toString*(*maRequete*.getTemperatureIntern())+"°C";

**this**.temp += " / "+Float.*toString*(*maRequete*.getTemperatureSouhaiter())+"°C";

JLabel lblTemp = **new** JLabel(**this**.temp);

lblTemp.setBounds(14\*width/100, 1, 26\*width/100, 6\*height/100);

lblTemp.setHorizontalAlignment(SwingConstants.***CENTER***);

lblTemp.setFont(**new** Font("Tahoma", Font.***PLAIN***, 5\*height/100));

lblTemp.setForeground(Color.***WHITE***);

contentPane.add(lblTemp);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie alarme

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblTextAlarme = **new** JLabel("");

lblTextAlarme.setBounds(20\*width/100, 30\*height/100, 46\*width/100, 10\*height/100);

lblTextAlarme.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(lblTextAlarme);

Image iconValider = **new** ImageIcon(**this**.getClass().getResource("/iconValider.png")).getImage();

JButton btnValider = **new** JButton(**new** ImageIcon(**this**.modifImg.scaleImage(iconValider, 20\*width/100, 20\*height/100)));

btnValider.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**if** (Integer.*parseInt*(temp2.toString())==*maRequete*.getCodeAlarme()){

**if** (*maRequete*.etatAlarme()==**true**){

*maRequete*.SetEtatAlarme(**false**);

lblTextAlarme.setText("Alarme desactiver");

}

**else**{

*maRequete*.SetEtatAlarme(**true**);

lblTextAlarme.setText("Alarme activer");

}

}

}

});

btnValider.setBounds(70\*width/100,30\*height/100, 20\*width/100, 20\*height/100);

contentPane.add(btnValider);

JLabel resultat = **new** JLabel("");

resultat.setBounds(10\*width/100, 70\*height/100, 50\*width/100, 10\*height/100);

resultat.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(resultat);

Image iconCorriger = **new** ImageIcon(**this**.getClass().getResource("/iconCorriger.png")).getImage();

JButton btnCorriger = **new** JButton(**new** ImageIcon(**this**.modifImg.scaleImage(iconCorriger, 20\*width/100, 20\*height/100)));

btnCorriger.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2="";

resultat.setText(temp2);

}

});

btnCorriger.setBounds(70\*width/100,55\*height/100, 20\*width/100, 20\*height/100);

contentPane.add(btnCorriger);

JButton btnUn = **new** JButton("1");

btnUn.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent arg0) {

temp2 +="1";

resultat.setText(temp2);

}

});

btnUn.setBounds(5\*width/100, 40\*height/100, 10\*width/100, 8\*height/100);

btnUn.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnUn);

JButton btnDeux = **new** JButton("2");

btnDeux.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="2";

resultat.setText(temp2);

}

});

btnDeux.setBounds(17\*width/100, 40\*height/100, 10\*width/100, 8\*height/100);

btnDeux.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnDeux);

JButton btnTrois = **new** JButton("3");

btnTrois.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="3";

resultat.setText(temp2);

}

});

btnTrois.setBounds(29\*width/100, 40\*height/100, 10\*width/100, 8\*height/100);

btnTrois.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnTrois);

JButton btnQuatre = **new** JButton("4");

btnQuatre.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="4";

resultat.setText(temp2);

}

});

btnQuatre.setBounds(41\*width/100, 40\*height/100, 10\*width/100, 8\*height/100);

btnQuatre.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnQuatre);

JButton btnCinq = **new** JButton("5");

btnCinq.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="5";

resultat.setText(temp2);

}

});

btnCinq.setBounds(53\*width/100, 40\*height/100, 10\*width/100, 8\*height/100);

btnCinq.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnCinq);

JButton btnSix = **new** JButton("6");

btnSix.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="6";

resultat.setText(temp2);

}

});

btnSix.setBounds(5\*width/100, 52\*height/100, 10\*width/100, 8\*height/100);

btnSix.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnSix);

JButton btnSept = **new** JButton("7");

btnSept.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="7";

resultat.setText(temp2);

}

});

btnSept.setBounds(17\*width/100, 52\*height/100, 10\*width/100, 8\*height/100);

btnSept.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnSept);

JButton btnHuit = **new** JButton("8");

btnHuit.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="8";

resultat.setText(temp2);

}

});

btnHuit.setBounds(29\*width/100, 52\*height/100, 10\*width/100, 8\*height/100);

btnHuit.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnHuit);

JButton btnNeuf = **new** JButton("9");

btnNeuf.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="9";

resultat.setText(temp2);

}

});

btnNeuf.setBounds(41\*width/100, 52\*height/100, 10\*width/100, 8\*height/100);

btnNeuf.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnNeuf);

JButton btnZero = **new** JButton("0");

btnZero.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

temp2 +="0";

resultat.setText(temp2);

}

});

btnZero.setBounds(53\*width/100, 52\*height/100, 10\*width/100, 8\*height/100);

btnZero.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

contentPane.add(btnZero);

**if** (*maRequete*.etatAlarme()==**true**){

lblTextAlarme.setText("Alarme activer");

}

**else**{

lblTextAlarme.setText("Alarme désactiver");

}

JButton btnQuitter = **new** JButton("Retour accueil");

btnQuitter.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent arg0) {

IHMAccueilRasp accueil = **new** IHMAccueilRasp(*maRequete*);

accueil.setVisible(**true**);

dispose();

}

});

btnQuitter.setBounds(10\*width/100,80\*height/100, 80\*width/100, 15\*height/100);

btnQuitter.setFont(**new** Font("Tahoma", Font.***PLAIN***, 8\*height/100));

contentPane.add(btnQuitter);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie des differant rectangle de couleur de l'application pour separer les differante partie

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//bande noir au dessus

JLabel lblBandeNoir = **new** JLabel("");

lblBandeNoir.setBounds(0,0,width,7\*height/100);

lblBandeNoir.setOpaque(**true**);

lblBandeNoir.setBackground(Color.***BLACK***);

//contentPane.setLayout(null);

contentPane.add(lblBandeNoir);

// rectangle gris claire sur les bouton menu

JLabel lblRecGris = **new** JLabel("");

lblRecGris.setBounds(0,7\*height/100,width,17\*height/100);

lblRecGris.setOpaque(**true**);

lblRecGris.setBackground(Color.***LIGHT\_GRAY***);

//contentPane.setLayout(null);

contentPane.add(lblRecGris);

// rectangle blanc

JLabel lblRecChoix = **new** JLabel("");

lblRecChoix.setBounds(2\*width/100,27\*height/100,96\*width/100,70\*height/100);

lblRecChoix.setOpaque(**true**);

lblRecChoix.setBackground(Color.***WHITE***);

//contentPane.setLayout(null);

contentPane.add(lblRecChoix);

//rectangle gris foncer du reste de l'application

JLabel lblMiddleDarkGray = **new** JLabel("");

lblMiddleDarkGray.setBounds(0,25\*height/100,width,75\*height/100);

lblMiddleDarkGray.setOpaque(**true**);

lblMiddleDarkGray.setBackground(Color.***DARK\_GRAY***);

//contentPane.setLayout(null);

contentPane.add(lblMiddleDarkGray);

}

}

IHM HERCOME Center Module :

package view;

import java.awt.Color;

import java.awt.Dimension;

import java.awt.EventQueue;

import java.awt.Font;

import java.awt.Frame;

import java.awt.Image;

import javax.swing.DefaultComboBoxModel;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JComboBox;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.SwingConstants;

import javax.swing.border.EmptyBorder;

//importation classe besoin

import model.CRequeteRasp;

import model.CModificationImage;

import java.awt.event.ActionListener;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

public class IHMModuleRasp extends JFrame {

private static final long serialVersionUID = 3095235317386885468L;

private JPanel contentPane;

private CModificationImage modifImg;

private static CRequeteRasp maRequete;

private JComboBox combo = new JComboBox();

private String temp;

/\*public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

IHMModuleRasp frame = new IHMModuleRasp();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

\*/

/\*\*

\* Create the frame.

\*/

public IHMModuleRasp(CRequeteRasp req) {

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 450, 300);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

//this.maRequete = req;

maRequete = req;

modifImg = new CModificationImage();

this.pack();

JFrame.setDefaultLookAndFeelDecorated(true);

setExtendedState(Frame.MAXIMIZED\_BOTH);

Dimension dimension = java.awt.Toolkit.getDefaultToolkit().getScreenSize();

int height = (int)dimension.getHeight();

int width = (int)dimension.getWidth();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie menu de l'application evc le logo

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblNewLabel = new JLabel("");

lblNewLabel.setBounds(2\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

lblNewLabel.setOpaque(true);

lblNewLabel.setBackground(Color.WHITE);

Image img = new ImageIcon(this.getClass().getResource("/logoHercome.jpg")).getImage();

contentPane.setLayout(null);

lblNewLabel.setIcon(new ImageIcon(this.modifImg.scaleImage(img, 21\*width/100, 15\*height/100)));

contentPane.add(lblNewLabel);

JButton btnChauffage = new JButton("Chauffage");

btnChauffage.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

IHMChauffageRasp chauffage = new IHMChauffageRasp(maRequete);

chauffage.setVisible(true);

dispose();

}

});

btnChauffage.setBounds(27\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnChauffage.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnChauffage.setBackground(Color.WHITE);

contentPane.add(btnChauffage);

JButton btnModules = new JButton("Modules");

btnModules.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

}

});

btnModules.setBounds(52\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnModules.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnModules.setBackground(Color.WHITE);

contentPane.add(btnModules);

JButton btnAlarme = new JButton("Alarme");

btnAlarme.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

IHMAlarmeRasp alarme = new IHMAlarmeRasp(maRequete);

alarme.setVisible(true);

dispose();

}

});

btnAlarme.setBounds(77\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnAlarme.setFont(new Font("Tahoma", Font.PLAIN, 6\*height/100));

btnAlarme.setBackground(Color.WHITE);

contentPane.add(btnAlarme);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie texte information de l'application

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie bande noir

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblInfoImgAmp = new JLabel ("");

lblInfoImgAmp.setBounds(1, 1, 4\*width/100, 6\*height/100);

lblInfoImgAmp.setOpaque(true);

lblInfoImgAmp.setBackground(Color.WHITE);

Image iconAmpoule = new ImageIcon(this.getClass().getResource("/iconAmpoule.jpg")).getImage();

contentPane.setLayout(null);

lblInfoImgAmp.setIcon(new ImageIcon(this.modifImg.scaleImage(iconAmpoule, 4\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgAmp);

this.temp = Integer.toString(maRequete.getNombreDeLum());

JLabel lblNbLum = new JLabel(this.temp);

lblNbLum.setBounds(5\*width/100, 1, 2\*width/100, 6\*height/100);

lblNbLum.setHorizontalAlignment(SwingConstants.CENTER);

lblNbLum.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

lblNbLum.setForeground(Color.WHITE);

contentPane.add(lblNbLum);

JLabel lblInfoImgTemp = new JLabel ("");

lblInfoImgTemp.setBounds(11\*width/100, 1\*height/100, 3\*width/100, 6\*height/100);

lblInfoImgTemp.setOpaque(true);

lblInfoImgTemp.setBackground(Color.BLACK);

Image iconTemp = new ImageIcon(this.getClass().getResource("/iconTemp.png")).getImage();

contentPane.setLayout(null);

lblInfoImgTemp.setIcon(new ImageIcon(this.modifImg.scaleImage(iconTemp, 3\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgTemp);

this.temp = " : "+Float.toString(maRequete.getTemperatureIntern())+"°C";

this.temp += " / "+Float.toString(maRequete.getTemperatureSouhaiter())+"°C";

JLabel lblTemp = new JLabel(this.temp);

lblTemp.setBounds(14\*width/100, 1, 26\*width/100, 6\*height/100);

lblTemp.setHorizontalAlignment(SwingConstants.CENTER);

lblTemp.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

lblTemp.setForeground(Color.WHITE);

contentPane.add(lblTemp);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie bouton module

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*JButton btnLumiere = new JButton("Module Lumière");

btnLumiere.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

IHMModuleLumRasp lum = new IHMModuleLumRasp(maRequete);

lum.setVisible(true);

dispose();

}

});

btnLumiere.setBounds(8\*width/100,34\*height/100, 36\*width/100, 20\*height/100);

btnLumiere.setFont(new Font("Tahoma", Font.PLAIN, 8\*height/100));

contentPane.add(btnLumiere);

JButton btnStore = new JButton("Module Store");

btnStore.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

IHMModuleStoreRasp store = new IHMModuleStoreRasp(maRequete);

store.setVisible(true);

dispose();

}

});

btnStore.setBounds(58\*width/100,34\*height/100, 36\*width/100, 20\*height/100);

btnStore.setFont(new Font("Tahoma", Font.PLAIN, 8\*height/100));

contentPane.add(btnStore);

\*/

ArrayList<String> mesPieces = new ArrayList<String>();

mesPieces = this.maRequete.getLesPieces();

String[] test = mesPieces.toArray(new String[0]);

combo.setPreferredSize(new Dimension(20\*width/100, 20\*height/100));

combo.setModel(new DefaultComboBoxModel(test));

combo.setBounds(8\*width/100,34\*height/100, 84\*width/100, 26\*height/100);

combo.setFont(new Font("Tahoma", Font.PLAIN, 9\*height/100));

//combo.setSize(200, combo.getPreferredSize().height);

contentPane.add(combo);

JButton btnSaisie = new JButton("Valider");

btnSaisie.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent arg0){

IHMPieceChoisi maPieceChoisi = new IHMPieceChoisi ((String) combo.getSelectedItem(),maRequete);

maPieceChoisi.setVisible(true);

dispose();

}

});

btnSaisie.setBounds(20\*width/100,65\*height/100, 60\*width/100, 10\*height/100);

btnSaisie.setFont(new Font("Tahoma", Font.PLAIN, 8\*height/100));

contentPane.add(btnSaisie);

/\*JLabel testTest = new JLabel ((String) combo.getSelectedItem());

testTest.setBounds(8\*width/100, 65\*height/100, 84\*width/100, 8\*height/100);

testTest.setHorizontalAlignment(SwingConstants.CENTER);

testTest.setFont(new Font("Tahoma", Font.PLAIN, 5\*height/100));

contentPane.add(testTest);

\*/

JButton btnQuitter = new JButton("Retour accueil");

btnQuitter.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

IHMAccueilRasp accueil = new IHMAccueilRasp(maRequete);

accueil.setVisible(true);

dispose();

}

});

btnQuitter.setBounds(10\*width/100,80\*height/100, 80\*width/100, 15\*height/100);

btnQuitter.setFont(new Font("Tahoma", Font.PLAIN, 8\*height/100));

contentPane.add(btnQuitter);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie des differant rectangle de couleur de l'application pour separer les differentes partie

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//bande noir au dessus

JLabel lblBandeNoir = new JLabel("");

lblBandeNoir.setBounds(0,0,width,7\*height/100);

lblBandeNoir.setOpaque(true);

lblBandeNoir.setBackground(Color.BLACK);

//contentPane.setLayout(null);

contentPane.add(lblBandeNoir);

// rectangle gris claire sur les bouton menu

JLabel lblRecGris = new JLabel("");

lblRecGris.setBounds(0,7\*height/100,width,17\*height/100);

lblRecGris.setOpaque(true);

lblRecGris.setBackground(Color.LIGHT\_GRAY);

//contentPane.setLayout(null);

contentPane.add(lblRecGris);

// rectangle blanc

JLabel lblRecChoix = new JLabel("");

lblRecChoix.setBounds(2\*width/100,27\*height/100,96\*width/100,70\*height/100);

lblRecChoix.setOpaque(true);

lblRecChoix.setBackground(Color.WHITE);

//contentPane.setLayout(null);

contentPane.add(lblRecChoix);

//rectangle gris foncer du reste de l'application

JLabel lblMiddleDarkGray = new JLabel("");

lblMiddleDarkGray.setBounds(0,25\*height/100,width,75\*height/100);

lblMiddleDarkGray.setOpaque(true);

lblMiddleDarkGray.setBackground(Color.DARK\_GRAY);

//contentPane.setLayout(null);

contentPane.add(lblMiddleDarkGray);

}

}

IHM HERCOME Center pièce choisi :

**package** view;

**import** java.awt.BorderLayout;

**import** java.awt.Color;

**import** java.awt.Dimension;

**import** java.awt.Font;

**import** java.awt.Frame;

**import** java.awt.Image;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.util.ArrayList;

**import** javax.swing.ImageIcon;

**import** javax.swing.JButton;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JPanel;

**import** javax.swing.JScrollPane;

**import** javax.swing.ScrollPaneConstants;

**import** javax.swing.SwingConstants;

**import** javax.swing.border.EmptyBorder;

**import** model.CModificationImage;

**import** model.CRequeteRasp;

**import** model.CGestion;

**public** **class** IHMPieceChoisi **extends** JFrame {

/\*\*

\*

\*/

**private** **static** **final** **long** ***serialVersionUID*** = -7351129140236878668L;

**private** JPanel contentPane;

**private** CModificationImage modifImg;

**private** **static** CRequeteRasp *maRequete*;

**private** String temp;

/\*\*

\* Create the frame.

\*/

**public** IHMPieceChoisi(String maPiece,CRequeteRasp req) {

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

setBounds(100, 100, 450, 300);

contentPane = **new** JPanel();

contentPane.setBorder(**new** EmptyBorder(5, 5, 5, 5));

contentPane.setLayout(**new** BorderLayout(0, 0));

setContentPane(contentPane);

**this**.*maRequete* = req;

//maRequete = req;

modifImg = **new** CModificationImage();

**this**.pack();

JFrame.*setDefaultLookAndFeelDecorated*(**true**);

setExtendedState(Frame.***MAXIMIZED\_BOTH***);

Dimension dimension = java.awt.Toolkit.*getDefaultToolkit*().getScreenSize();

**int** height = (**int**)dimension.getHeight();

**int** width = (**int**)dimension.getWidth();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie menu de l'application evc le logo

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblNewLabel = **new** JLabel("");

lblNewLabel.setBounds(2\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

lblNewLabel.setOpaque(**true**);

lblNewLabel.setBackground(Color.***WHITE***);

Image img = **new** ImageIcon(**this**.getClass().getResource("/logoHercome.jpg")).getImage();

contentPane.setLayout(**null**);

lblNewLabel.setIcon(**new** ImageIcon(**this**.modifImg.scaleImage(img, 21\*width/100, 15\*height/100)));

contentPane.add(lblNewLabel);

JButton btnChauffage = **new** JButton("Chauffage");

btnChauffage.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent arg0) {

IHMChauffageRasp chauffage = **new** IHMChauffageRasp(*maRequete*);

chauffage.setVisible(**true**);

dispose();

}

});

btnChauffage.setBounds(27\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnChauffage.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

btnChauffage.setBackground(Color.***WHITE***);

contentPane.add(btnChauffage);

JButton btnModules = **new** JButton("Modules");

btnModules.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

IHMModuleRasp module = **new** IHMModuleRasp(*maRequete*);

module.setVisible(**true**);

dispose();

}

});

btnModules.setBounds(52\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnModules.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

btnModules.setBackground(Color.***WHITE***);

contentPane.add(btnModules);

JButton btnAlarme = **new** JButton("Alarme");

btnAlarme.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

IHMAlarmeRasp alarme = **new** IHMAlarmeRasp(*maRequete*);

alarme.setVisible(**true**);

dispose();

}

});

btnAlarme.setBounds(77\*width/100, 8\*height/100, 21\*width/100, 15\*height/100);

btnAlarme.setFont(**new** Font("Tahoma", Font.***PLAIN***, 6\*height/100));

btnAlarme.setBackground(Color.***WHITE***);

contentPane.add(btnAlarme);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie texte information de l'application

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie bande noir

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JLabel lblInfoImgAmp = **new** JLabel ("");

lblInfoImgAmp.setBounds(1, 1, 4\*width/100, 6\*height/100);

lblInfoImgAmp.setOpaque(**true**);

lblInfoImgAmp.setBackground(Color.***WHITE***);

Image iconAmpoule = **new** ImageIcon(**this**.getClass().getResource("/iconAmpoule.jpg")).getImage();

contentPane.setLayout(**null**);

lblInfoImgAmp.setIcon(**new** ImageIcon(**this**.modifImg.scaleImage(iconAmpoule, 4\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgAmp);

**this**.temp = Integer.*toString*(*maRequete*.getNombreDeLum());

JLabel lblNbLum = **new** JLabel(**this**.temp);

lblNbLum.setBounds(5\*width/100, 1, 2\*width/100, 6\*height/100);

lblNbLum.setHorizontalAlignment(SwingConstants.***CENTER***);

lblNbLum.setFont(**new** Font("Tahoma", Font.***PLAIN***, 5\*height/100));

lblNbLum.setForeground(Color.***WHITE***);

contentPane.add(lblNbLum);

JLabel lblInfoImgTemp = **new** JLabel ("");

lblInfoImgTemp.setBounds(11\*width/100, 1\*height/100, 3\*width/100, 6\*height/100);

lblInfoImgTemp.setOpaque(**true**);

lblInfoImgTemp.setBackground(Color.***BLACK***);

Image iconTemp = **new** ImageIcon(**this**.getClass().getResource("/iconTemp.png")).getImage();

contentPane.setLayout(**null**);

lblInfoImgTemp.setIcon(**new** ImageIcon(**this**.modifImg.scaleImage(iconTemp, 3\*width/100, 6\*height/100)));

contentPane.add(lblInfoImgTemp);

**this**.temp = " : "+Float.*toString*(*maRequete*.getTemperatureIntern())+"°C";

**this**.temp += " / "+Float.*toString*(*maRequete*.getTemperatureSouhaiter())+"°C";

JLabel lblTemp = **new** JLabel(**this**.temp);

lblTemp.setBounds(14\*width/100, 1, 26\*width/100, 6\*height/100);

lblTemp.setHorizontalAlignment(SwingConstants.***CENTER***);

lblTemp.setFont(**new** Font("Tahoma", Font.***PLAIN***, 5\*height/100));

lblTemp.setForeground(Color.***WHITE***);

contentPane.add(lblTemp);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

partie app

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

JPanel central = **new** JPanel ();

//JScrollPane scrollPane = new JScrollPane(central, ScrollPaneConstants.VERTICAL\_SCROLLBAR\_AS\_NEEDED, ScrollPaneConstants.HORIZONTAL\_SCROLLBAR\_AS\_NEEDED);

central.setLayout(**null**);

central.setBounds(2\*width/100,27\*height/100,96\*width/100,70\*height/100);

central.setPreferredSize(**new** Dimension(96\*width/100,70\*height/100));

**final** JScrollPane scrollPane = **new** JScrollPane(central, JScrollPane.***VERTICAL\_SCROLLBAR\_AS\_NEEDED***, JScrollPane.***HORIZONTAL\_SCROLLBAR\_NEVER***);

ArrayList<JPanel> mesModules = **new** ArrayList<JPanel>();

**int** largeurPanel = 96\*width/100;

**int** hauteurPanel = 70\*height/100;

//JScrollPane scrollPane = new JScrollPane(central);

//scrollPane.setLayout(null);

//scrollPane.setBounds(0,0,96\*width/100,70\*height/100);

//scrollPane.setBackground(Color.green);

**for** (**int** i=0;i<10;i++){

JPanel temporairePanel = **new** JPanel();

**if** (i%2==0){

temporairePanel.setBounds(largeurPanel\*4/100, hauteurPanel\*4/100+i/2\*hauteurPanel\*30/100+i\*hauteurPanel/100, largeurPanel\*44/100, hauteurPanel\*30/100);

//temporairePanel.setLocation(largeurPanel\*4/100, hauteurPanel\*30/100+i\*hauteurPanel\*20/100);

//temporairePanel.setSize(largeurPanel\*44/100, hauteurPanel\*20/100);

}

**else**{

temporairePanel.setBounds(largeurPanel\*54/100, hauteurPanel\*4/100+i/2\*hauteurPanel\*30/100+i\*hauteurPanel/100, largeurPanel\*44/100, hauteurPanel\*30/100);

//temporairePanel.setBounds(200, 400, 100, 500);

}

temporairePanel.setBackground(Color.***darkGray***);

mesModules.add(temporairePanel);

central.add(mesModules.get(i));

//central.setViewportView(mesModules.get(i));

//scrollPane.setViewportView(mesModules.get(i));

}

//central.add(scrollPane);

//JScrollPane scrollPane = new JScrollPane(central);

//scrollPane.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL\_SCROLLBAR\_NEVER); //SETTING SCHEME FOR HORIZONTAL BAR

//scrollPane.setVerticalScrollBarPolicy(JScrollPane.VERTICAL\_SCROLLBAR\_AS\_NEEDED);

//this.add(scrollPane);

contentPane.add(central);

//contentPane.add(scrollPane);

//contentPane.add(central);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Partie des differant rectangle de couleur de l'application pour separer les differentes partie

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//bande noir au dessus

JLabel lblBandeNoir = **new** JLabel("");

lblBandeNoir.setBounds(0,0,width,7\*height/100);

lblBandeNoir.setOpaque(**true**);

lblBandeNoir.setBackground(Color.***BLACK***);

//contentPane.setLayout(null);

contentPane.add(lblBandeNoir);

// rectangle gris claire sur les bouton menu

JLabel lblRecGris = **new** JLabel("");

lblRecGris.setBounds(0,7\*height/100,width,17\*height/100);

lblRecGris.setOpaque(**true**);

lblRecGris.setBackground(Color.***LIGHT\_GRAY***);

//contentPane.setLayout(null);

contentPane.add(lblRecGris);

// rectangle blanc

JLabel lblRecChoix = **new** JLabel("");

lblRecChoix.setBounds(2\*width/100,27\*height/100,96\*width/100,70\*height/100);

lblRecChoix.setOpaque(**true**);

lblRecChoix.setBackground(Color.***WHITE***);

//contentPane.setLayout(null);

contentPane.add(lblRecChoix);

//rectangle gris foncer du reste de l'application

JLabel lblMiddleDarkGray = **new** JLabel("");

lblMiddleDarkGray.setBounds(0,25\*height/100,width,75\*height/100);

lblMiddleDarkGray.setOpaque(**true**);

lblMiddleDarkGray.setBackground(Color.***DARK\_GRAY***);

//contentPane.setLayout(null);

contentPane.add(lblMiddleDarkGray);

}

}

HERCOME Center classe Alarme :

package model;

import model.CCapteurPresence;

import model.CRequeteRasp;

public class CAlarme {

private CCapteurPresence monCapteur;

private CRequeteRasp maRequete;

private boolean etat=false;

public CAlarme(){

}

public void activer(){

this.etat = true;

this.maRequete.activerAlarme();

this.monCapteur.activer();

}

public void desactiver(){

this.etat = false;

this.monCapteur.desactiver();

}

}

HERCOME Center classe bdd :

package model;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.ResultSetMetaData;

import java.sql.SQLException;

import java.sql.Statement;

public class CBdd

{

private String url = "192.168.1.253";

private String user = "root";

private String passwd = "";

private Connection conn;

private Statement state;

private ResultSet result;

private ResultSetMetaData resultMeta;

public CBdd(){

}

public void connection(){

try

{

this.conn = DriverManager.getConnection(this.url, this.user, this.passwd);

this.state = conn.createStatement();

}

catch (Exception e)

{

e.printStackTrace();

}

}

public int retourDonneInt(String requete,String donnee){

int resultat = 0;

try {

this.result = state.executeQuery(requete);

this.resultMeta = result.getMetaData();

resultat = result.getInt(donnee);

} catch (SQLException e) {

// TODO Bloc catch généré automatiquement

e.printStackTrace();

}

return resultat;

}

public String retourDonneString(String requete,String donnee){

String resultat = "";

try {

this.result = state.executeQuery(requete);

this.resultMeta = result.getMetaData();

resultat = result.getString(donnee);

} catch (SQLException e) {

// TODO Bloc catch généré automatiquement

e.printStackTrace();

}

return resultat;

}

}

HERCOME Center classe CCommandStore :

**package** model;

**public** **class** CCommandStore {

**public** **void** monterStore(**char**[] trame){

// ajouter dans la trame le code correspondant à l'information de monter

}

**public** **void** dessendreStore(**char**[] trame){

// ajouter dans la trame le code correspondant à l'information de dessendre

}

**public** **void** stoperStore(**char**[] trame){

// ajouter dans la trame le code correspondant à l'information à stoper l'action en court

}

}

HERCOME Center classe CCommandLumiere :

**package** model;

**public** **class** CCommandLumiere {

**public** **void** allumerLumiere(**char**[] trame){

// ajouter dans la trame le code correspondant à l'information d'allumer

}

**public** **void** eteindreLumiere(**char**[] trame){

// ajouter dans la trame le code correspondant à l'information d'eteindre

}

}

HERCOME Center classe CGestion :

**package** model;

**public** **class** CGestion {

**private** CRequeteRasp maRequete;

**public** CGestion(CRequeteRasp req){

**this**.maRequete = req;

}

**public** **float** decryptageTrame(**char** [] trame){

**float** retour= (**float**) 0.0;

**switch** (trame[4]){

**case** '1' :

CCommandStore mesStores = **new** CCommandStore();

**switch** (trame[8]){

**case** '0' :

mesStores.monterStore(trame);

**break**;

**case** '1':

mesStores.dessendreStore(trame);

**break**;

**case** '2':

mesStores.stoperStore(trame);

**break**;

}

**break**;

**case** '2' :

retour = maRequete.getTemperatureSouhaiter();

**break**;

**case** '3' :

CCommandLumiere mesLumieres = **new** CCommandLumiere();

**if** (trame[8]=='0'){

mesLumieres.allumerLumiere(trame);

maRequete.setEtatLamp(trame[1]\*100+trame[2]\*10+trame[3], **true**);

}

**else**{

mesLumieres.eteindreLumiere(trame);

maRequete.setEtatLamp(trame[1]\*100+trame[2]\*10+trame[3], **false**);

}

**break**;

**default** :

**break**;

}

**return** retour;

}

}

HERCOME Center classe CModificationImage :

package model;

import java.awt.Graphics2D;

import java.awt.Image;

import java.awt.RenderingHints;

import java.awt.image.BufferedImage;

public class CModificationImage {

public CModificationImage(){

}

// methode pour modifier taille image

public Image scaleImage(Image source, int width, int height) {

BufferedImage image = new BufferedImage(width, height, BufferedImage.TYPE\_INT\_ARGB);

Graphics2D g = (Graphics2D) image.getGraphics();

g.setRenderingHint(RenderingHints.KEY\_INTERPOLATION, RenderingHints.VALUE\_INTERPOLATION\_BILINEAR);

g.drawImage(source, 0, 0, width, height, null);

g.dispose();

return image;

}

}

HERCOME Center classe CRequeteRasp (classe qui devra être modifier pour réellement communiquer avec CBdd et surrpimer tous les attributs )

**package** model;

**import** java.util.ArrayList;

**import** model.CBdd;

**public** **class** CRequeteRasp {

CBdd maBdd;

**float** testTempSouhaiter=20;

**float** testTempintern=18;

**float** testTempExtern = (**float**) 15.6;

**int** codeAlarme=123456789;

**boolean** etatAlarme = **false**;

**int** nbLumAllumer = 0;

**int** nbPiece;

ArrayList<String> nomPiece = **new** ArrayList<String>();

ArrayList<Boolean> etatLampePiece = **new** ArrayList<Boolean>();

ArrayList<ArrayList<Boolean>> pieceMaisonLum =**new** ArrayList<ArrayList<Boolean>>();

**public** CRequeteRasp (){

maBdd = **new** CBdd ();

**this**.nomPiece.add("salon");

**this**.nomPiece.add("chambre parent");

**this**.nomPiece.add("cuisine");

**this**.nomPiece.add("salle de jeu");

**this**.nomPiece.add("Chambre fille");

**this**.nomPiece.add("Chambre garcon");

**this**.nomPiece.add("salle à manger");

**this**.etatLampePiece.add(**false**);

**this**.etatLampePiece.add(**false**);

**this**.pieceMaisonLum.add(**this**.etatLampePiece);

**this**.etatLampePiece.clear();

**this**.etatLampePiece.add(**false**);

**this**.pieceMaisonLum.add(**this**.etatLampePiece);

**this**.etatLampePiece.clear();

**this**.etatLampePiece.add(**false**);

**this**.pieceMaisonLum.add(**this**.etatLampePiece);

**this**.etatLampePiece.clear();

**this**.etatLampePiece.add(**false**);

**this**.etatLampePiece.add(**false**);

**this**.pieceMaisonLum.add(**this**.etatLampePiece);

**this**.etatLampePiece.clear();

**this**.etatLampePiece.add(**false**);

**this**.pieceMaisonLum.add(**this**.etatLampePiece);

**this**.etatLampePiece.clear();

**this**.etatLampePiece.add(**false**);

**this**.pieceMaisonLum.add(**this**.etatLampePiece);

**this**.etatLampePiece.clear();

**this**.etatLampePiece.add(**false**);

**this**.etatLampePiece.add(**false**);

**this**.etatLampePiece.add(**false**);

**this**.pieceMaisonLum.add(**this**.etatLampePiece);

**this**.nbPiece = nomPiece.size();

**for** (**int** i=0;i<nbPiece;i++){

**this**.etatLampePiece.add(**false**);

}

}

// a prog les differente requete pour la base de donnée

**public** **int** recupNumeroArduino (){

**this**.maBdd.connection();

String query = "SELECT ARD\_Num FROM ARDUINO";

**return** maBdd.retourDonneInt(query,"ARD\_Num");

}

**public** **void** activerAlarme(){

**this**.maBdd.connection();

String query = "";

}

**public** **boolean** etatAlarme(){

**this**.maBdd.connection();

**return** **this**.etatAlarme;

}

**public** **void** SetEtatAlarme(**boolean** etat){

**this**.etatAlarme = etat;

}

**public** **int** getNombreDeLum(){

**this**.maBdd.connection();

**return** **this**.nbLumAllumer;

}

**public** **void** setNombreDeLumPlus(){

**this**.nbLumAllumer++;

}

**public** **void** setNombreDeLumMoin(){

**this**.nbLumAllumer--;

}

**public** **int** getVitesseVent(){

**this**.maBdd.connection();

**return** 55;

}

**public** String getDirectionVent(){

**this**.maBdd.connection();

**return** "N-O";

}

**public** **int** getPluie(){

**this**.maBdd.connection();

**return** 20;

}

**public** **int** getHumidite(){

**this**.maBdd.connection();

**return** 20;

}

**public** **float** getTemperatureExtern(){

**this**.maBdd.connection();

**return** **this**.testTempExtern;

}

**public** **float** getTemperatureIntern(){

**this**.maBdd.connection();

**return** testTempintern;

}

**public** **float** getTemperatureSouhaiter(){

**this**.maBdd.connection();

**return** testTempSouhaiter;

}

**public** **void** setTemperatureSouhaiterPlus(){

**this**.maBdd.connection();

testTempSouhaiter+=0.5;

}

**public** **void** setTemperatureSouhaiterMoin(){

**this**.maBdd.connection();

testTempSouhaiter-=0.5;

}

**public** **int** getCodeAlarme(){

**return** **this**.codeAlarme;

}

**public** **void** setCodeAlarme(**int** code){

**this**.codeAlarme=code;

}

**public** **int** getNbPiece(){

**return** **this**.nbPiece;

}

**public** String nomPieceIndexer(**int** id){

**return** **this**.nomPiece.get(id);

}

**public** ArrayList<String> getLesPieces(){

**return** **this**.nomPiece;

}

**public** **boolean** getEtatLampe(**int** id){

**return** **this**.etatLampePiece.get(id);

}

**public** **void** setEtatLamp(**int** id, **boolean** etat){

**this**.etatLampePiece.set(id, etat);

}

**public** ArrayList<Boolean> getEtatStoreParPiece(**int** id){

**return** **this**.pieceMaisonLum.get(id);

}

**public** **void** setEtatStoreParPiece(**int** id,ArrayList<Boolean> piece){

**this**.pieceMaisonLum.set(id, piece);

}

**public** ArrayList<Boolean> getEtatLumParPiece(**int** id){

**return** **this**.pieceMaisonLum.get(id);

}

**public** **void** setEtatLumParPiece(**int** id,ArrayList<Boolean> piece){

**this**.pieceMaisonLum.set(id, piece);

}

}