



Institut Supérieur  
d'Informatique, de  
Modélisation et de  
leurs Applications

Complexe des Cézeaux  
BP 125  
63173 Aubière Cedex



Institut National de la  
Recherche Agronomique

Unité de Nutrition Humaine,  
UMR1019 INRA-Université  
d'Auvergne

Laboratoire de Nutrition  
Humaine  
58 rue Montalembert, BP321  
63009 Clermont-Ferrand

**Rapport d'ingénieur**  
**Projet de 3<sup>ème</sup> année**

**Filière 3** : Informatique des Systèmes d'Information,  
de Production et Aide à la décision

---

# **TRAITEMENT MATHÉMATIQUE DE DONNÉES CARDIAQUES : ÉTAPE PRÉLIMINAIRE DE LA PREDICTION DE LA DÉPENSE ÉNERGÉTIQUE**

---

---

**Présenté par** : Clara SPRIET  
Anthony WALTER

**Responsables ISIMA** : Philippe LACOMME  
Gérard FLEURY

**Responsable INRA** : Sylvie ROUSSET

---

**Durée du projet** : 160 heures  
Octobre 2009 - Mars 2010



Institut Supérieur  
d'Informatique, de  
Modélisation et de  
leurs Applications

Complexe des Cézeaux  
BP 125  
63173 Aubière Cedex



Institut National de la  
Recherche Agronomique

Unité de Nutrition Humaine,  
UMR1019 INRA-Université  
d'Auvergne

Laboratoire de Nutrition  
Humaine  
58 rue Montalembert, BP321  
63009 Clermont-Ferrand

**Rapport d'ingénieur**  
**Projet de 3<sup>ème</sup> année**

**Filière 3 : Informatique des Systèmes d'Information,  
de Production et Aide à la décision**

---

# **TRAITEMENT MATHÉMATIQUE DE DONNÉES CARDIAQUES : ÉTAPE PRÉLIMINAIRE DE LA PREDICTION DE LA DÉPENSE ÉNERGÉTIQUE**

---

---

**Présenté par :** Clara SPRIET  
Anthony WALTER

**Responsables ISIMA :** Philippe LACOMME  
Gérard FLEURY

**Responsable INRA :** Sylvie ROUSSET

---

**Durée du projet :** 160 heures  
Octobre 2009 - Mars 2010

## REMERCIEMENTS

Nous souhaitons tout d'abord remercier Madame **Sylvie ROUSSET**. Sa passion et son dévouement concernant le projet nous ont permis d'avancer en toutes circonstances.

Nous remercions également Messieurs **Philippe LACOMME** et **Gérard FLEURY** pour le suivi de notre projet et leurs précieux conseils.

CONFIDENTIEL

## RESUME

Le projet Modelheart de modélisation de la **dépense énergétique** a été conçu et réalisé par l'équipe de Métabolisme énergétique et lipidique de l'Unité de Nutrition Humaine (INRA Clermont-Ferrand). Cette étude inclut 120 volontaires pour la détermination de plusieurs paramètres les concernant, notamment, leur dépense énergétique, leur rythme cardiaque, leurs mouvements, leur température et flux de chaleur corporelle. La moitié des volontaires a séjourné dans une chambre calorimétrique pendant 24h alors que l'autre moitié a été étudiée en conditions habituelles de vie pendant 11 jours. Les données recueillies proviennent de différentes sources (différents capteurs, différents volontaires et différentes conditions expérimentales). Les résultats sont enregistrés dans des **fichiers Excel** différents.

Notre travail a consisté en la création d'un logiciel permettant de traiter ses fichiers. Il a été écrit en langage JAVA grâce à la librairie **JExcelApi** qui gère les fichiers Excel. Le graphisme a été réalisé sous **NetBeans**.

A la fin de notre projet, les scientifiques de l'INRA sont désormais en mesure de standardiser les fichiers obtenus par deux capteurs et surtout, de remplacer leurs **valeurs manquantes et aberrantes** par des données correctes. Ces deux étapes étaient essentielles pour la suite de leurs réflexions concernant le traitement mathématique des fichiers.

Les premiers tests informatiques sont concluants et répondent aux exigences de notre **analyse** préalable. Un stage va débuter en Avril pour que notre travail soit complété et finalisé.

**Mots-clés** : Dépense énergétique, fichiers Excel, JExcelApi, NetBeans, analyse, valeurs manquantes et aberrantes

## ABSTRACT

The Modelheart project has been realized by the INRA on 120 volunteers in order to determine lots of parameters like their **energy expenditure**, their heart rate and their temperature. Some have lived in a calorimetric chamber during 24 hours and some have stayed at home normally during 11 days. Data come from different sources (different sensors, different volunteers...). Results are recorded in **Excel files**.

Our work has been created software which can process the files. It is designed in the programming language JAVA especially thanks to the **JExcelApi** library which is able to read, write and modify Excel spreadsheets dynamically. The graphic design has been programmed with **NetBeans**.

At the end, the INRA scientists are now able to make the standardization of the files of two sensors and above all, to replace their **missing and aberrant values**. Those two steps are essentials for the mathematical process of the files.

The first verifications of the software gave good results and complied with our preliminary **analyze**. A placement will begin on April in order to complete our work and finalize it.

**Keywords** : Energy expenditure, Excel files, JExcelApi, NetBeans, analyze, missing and aberrant values

# TABLE DES MATIERES

REMERCIEMENTS .....	
RESUME.....	
ABSTRACT.....	
TABLE DES MATIERES .....	
TABLE DES FIGURES.....	
INTRODUCTION .....	1
1. Contexte.....	2
1.1. L'Institut National de Recherche Agronomique .....	2
1.2. Chambres calorimétriques au CRNH .....	2
1.3. Le Projet Modelheart .....	3
2. Analyse.....	4
2.1. Différentes informations.....	4
2.1.1. Le Capteur ActiHeart.....	4
2.1.2. Le Capteur ArmBand.....	5
2.1.3. L'Electrocardiogramme.....	5
2.1.4. Paramètres capturés dans la chambre.....	6
2.1.5. Données concernant les volontaires.....	6
2.1.6. Mesure de la dépense énergétique par la technique de l'eau doublement marquée .....	7
2.1.7. Planning des activités menées en chambre calorimétrique.....	7
2.2. Diagramme de classe .....	8
2.3. Etapes à suivre .....	10
2.3.1. Etape 1 : Enregistrement d'une expérience .....	10
2.3.1.1. Traitement des données provenant des capteurs .....	11
2.3.1.2. Traitement des données provenant de la chambre calorimétrique .....	12
2.3.2. Etape 2 : Traitement mathématiques .....	12
2.3.3. Etape 3 : Détermination de la fonction finale de prédiction de la dépense énergétique .....	13
3. Modifications des fichiers Excel.....	13
3.1. Standardisation .....	13
3.1.1. Librairie JExcelApi .....	13
3.1.2. Standardisation ActiHeart.....	14
3.1.3. Standardisation ArmBand .....	15
3.2. Valeurs manquantes .....	17
3.2.1. ArmBand .....	17
3.2.2. ActiHeart.....	18
4. Réalisation graphique.....	21
4.1. Choix technique .....	21
4.2. Présentation générale du logiciel.....	22
4.3. Quelques aspects du développement .....	29
4.3.1. Arborescence .....	29
4.3.2. Saisie des Groupes .....	34
4.3.3. Parcours des fichiers.....	38
5. Validation et prolongements .....	40
5.1. Tests et Validation.....	40
5.2. Prolongements.....	43
CONCLUSION .....	44
GLOSSAIRE.....	
BIBLIOGRAPHIE .....	

## TABLE DES FIGURES

Figure 1 : Photo d'une partie d'une des deux chambres calorimétriques du CRNH .....	3
Figure 2 : Tableau représentant un exemple de données .....	5
de fréquence cardiaque relevées par le capteur ActiHeart .....	5
Figure 3 : Planning théorique .....	7
Figure 4 : Diagramme de classe du projet .....	9
Figure 5 : Actigramme montrant quelles données doivent être traitées.....	10
Figure 6 : Actigramme A0 montrant la différence de traitement .....	10
Figure 7 : Actigramme A2 - Traiter les données provenant des capteurs .....	11
Figure 8 : Actigramme A1 - Traiter les données de la chambre calorimétrique .....	12
Figure 9 : Extrait du code de la standardisation ActiHeart .....	14
Figure 10 : Fichier ArmBand d'origine .....	15
Figure 11 : Fichier ArmBand corrigé .....	15
Figure 12 : Cause du problème de jour dans le fichier ArmBand .....	16
Figure 13 : Problème du jour dans le fichier ArmBand .....	16
Figure 14 : Traitement des valeurs manquantes d'un fichier ArmBand .....	17
Figure 15 : Algorithme des valeurs manquantes des fichiers ActiHeart .....	18
Figure 16 : Interpolation pour moins de 6 valeurs manquantes.....	18
Figure 17 : Graphique représentant la fréquence cardiaque brute en fonction du temps .....	20
Figure 18 : Graphique représentant la fréquence cardiaque brute .....	20
en fonction du temps mais sur un fichier traité .....	20
Figure 19 : Fenêtre d'accueil du logiciel .....	22
Figure 20 : Fenêtre « Nouveau Projet ».....	23
Figure 21 : Fenêtre « Saisie des Groupes ».....	24
Figure 22 : Fenêtre « Etapes » .....	25
Figure 23 : Fenêtre « Création d'une Expérience 1/3 » avec la liste déroulante « Groupe » ouverte ..	26
Figure 24 : Fenêtre « Création d'une Expérience 2/3 » .....	27
et boîte de dialogue « La standardisation a réussi ! » .....	27
Figure 25 : Fenêtre « Création d'une Expérience » .....	28
Figure 26 : Extrait du code pour créer un dossier.....	29
Figure 27 : Extrait du code pour créer un fichier .....	30
Figure 28 : Fichier de configuration « config.txt » .....	30
Figure 29 : Fichier de configuration « config.txt » mis à jour .....	31
Figure 30 : Extrait du code pour sauvegarder le fichier ArmBand d'origine .....	31
Figure 31 : Fichier de configuration « configExperience.txt » .....	32
Figure 32 : Extrait du code pour supprimer l'arborescence (dossiers et fichier) d'un projet .....	32
Figure 33 : Arborescence du logiciel.....	33
Figure 34 : Fenêtre « Saisie des Groupes » avec un groupe activé .....	34
Figure 35 : Fenêtre « Saisie des Groupes » avec trois groupes activés .....	35
Figure 36 : Fichier de configuration « config.txt » avec le groupe 4 inutilisé (valeurs nulles) .....	36
Figure 37 : Liste déroulante des groupes d'expériences .....	37
Figure 38 : Extrait de code pour créer la liste déroulante des noms des groupes d'expériences .....	37
Figure 39 : Une des deux fenêtres « Parcourir » du logiciel .....	38
Figure 40 : Extrait du code pour sélectionner plus rapidement les fichiers Excel.....	39
Figure 41 : Extrait du code pour tester la bonne saisie du nom du projet et du nombre de groupes ..	40
Figure 42 : Boîte de dialogue d'avertissement .....	41
Figure 43 : Boîte de dialogue d'information .....	41
Figure 44 : Boîte de dialogue de confirmation .....	41
Figure 45 : Calendrier pour aider l'utilisateur à saisir la date de réalisation d'une expérience .....	42
Figure 46 : Exemple d'une comparaison entre l'ECG et la fréquence cardiaque corrigée .....	42

## INTRODUCTION

L'obésité représente un facteur de risque dans le développement de nombreuses maladies incluant les maladies cardio-vasculaires, le diabète de type 2 et certaines formes de cancer pour ne citer que les plus importantes. En France, comme dans la plupart des pays industrialisés, le nombre d'enfants obèses commence à atteindre des chiffres alarmants : selon des estimations, 16% des enfants seraient en surpoids ou obèses dans l'hexagone. Ce phénomène inquiétant touche de plus en plus tôt, indifféremment les garçons et les filles mais pourrait être stoppé par une politique de prévention active.

La dépense énergétique représente la quantité d'énergie consommée par l'organisme au cours d'une unité de temps pour réaliser une action ou remplir une fonction déterminée. L'alimentation assure les besoins en énergie de l'organisme. Pour qu'un individu conserve un poids stable il est nécessaire que la quantité d'énergie apportée par les aliments soit égale à la quantité d'énergie dépensée par l'individu.

Le but du travail développé par les chercheurs de l'INRA est de mettre au point des équations standardisées, prédictives de la dépense énergétique des 24h à partir de l'enregistrement de la fréquence cardiaque ainsi que d'autres paramètres physiologiques. L'intérêt majeur de cette nouvelle approche est de faire disparaître la nécessité d'un étalonnage individuel de la relation fréquence cardiaque/dépense énergétique en introduisant une modélisation n'utilisant que des paramètres facilement accessibles pour chaque individu. Cette connaissance plus fine des besoins énergétiques devrait permettre de mieux contrôler le poids avec, à terme, une spécification plus efficace des caractéristiques souhaitables des aliments.

Lors de notre projet de troisième année, le travail qui nous a été confié a été de créer un logiciel mettant en œuvre des équations de prédiction de la dépense énergétique des 24h, utilisant comme critère principal la fréquence cardiaque, les paramètres de mouvement, la température corporelle et le flux de chaleur enregistrés en continu minute par minute, et des critères simples comme le poids, la taille, l'âge ou le sexe.

Pour mieux comprendre le travail que nous avons effectué, il est tout d'abord nécessaire de présenter brièvement l'institut pour lequel nous avons travaillé, puis le projet qui a nécessité notre collaboration et enfin les outils qui ont servis à recueillir les données du projet. Dans un second temps, nous expliciterons en détail notre implication qui concerne la préparation des données de prédiction de la dépense énergétique. Puis nous exposerons le travail réalisé en expliquant le fonctionnement du logiciel à partir de quelques points de développement. Enfin, nous terminerons par la présentation des principaux résultats et tests, et discuterons des améliorations à apporter pour la continuité du projet.

## 1. Contexte

### 1.1. L'Institut National de Recherche Agronomique

Créé en 1946 dans le contexte de la reconstruction nationale d'après-guerre et du projet de modernisation de l'agriculture française, l'INRA [INRA] (Institut National de Recherche Agronomique) a accompagné depuis les mutations du monde agricole, des filières alimentaires et des territoires avec l'objectif de répondre aux attentes exprimées par la société, notamment celle de la suffisance alimentaire de la nation. Les défis scientifiques et sociétaux sont aujourd'hui bien différents et ont une dimension mondiale : profondes évolutions de l'alimentation, érosion de la biodiversité, développement des maladies émergentes, progrès de la chimie verte... L'INRA a profondément renouvelé ses approches pour y répondre.

Aujourd'hui, premier institut de recherche agronomique en Europe, deuxième dans le monde, l'INRA mène des recherches finalisées pour une alimentation saine et de qualité, pour une agriculture compétitive et durable, et pour un environnement préservé et valorisé.

Ses recherches sont guidées par l'évolution des questionnements scientifiques et orientées par les défis planétaires posés par l'alimentation, l'environnement et la valorisation des territoires, que l'agriculture et l'agronomie ont à relever. Changement climatique, nutrition humaine, compétition entre cultures alimentaires et non alimentaires, épuisement des ressources fossiles, équilibre dans la gestion des territoires sont autant d'enjeux qui positionnent l'agronomie comme fondatrice d'un développement harmonieux sur les plans économique, social et environnemental. L'INRA produit des connaissances fondamentales et construit grâce à elles des innovations et des savoir-faire pour la société. Il met son expertise au service de la décision publique. La complémentarité des thématiques étudiées, celle des approches utilisées, et la diversité de ses partenariats donnent à l'INRA une grande capacité et pertinence d'action au service de la société.

### 1.2. Chambres calorimétriques au CRNH

Le Centre de Recherche en Nutrition Humaine d'Auvergne (CRNH Auvergne) rassemble dans un Groupement d'Intérêt Public (GIP), 18 unités de recherche appartenant à 6 organismes différents dont l'INRA. Ce centre est équipé de deux chambres calorimétriques identiques qui permettent la mesure en continu de la dépense énergétique de l'Homme. Elles sont basées sur le principe de la calorimétrie indirecte en circuit ouvert. Cela consiste à mesurer les échanges gazeux respiratoires des individus (consommation d'oxygène et production de CO<sub>2</sub>).

Ces chambres possèdent la superficie (10 m<sup>2</sup>) et le confort (lit, lavabo, toilettes, four micro-ondes, TV, lecteur VHS et DVD) nécessaires à un séjour prolongé en conditions normales de vie (figure 1). Elles sont également équipées de tapis roulants, de vélos d'appartement, ou de steppers selon les besoins de l'étude, pour simuler une activité physique proche de celle que l'on peut avoir au quotidien. Ces installations permettent de mesurer les cinétiques de dépense énergétique, de fréquence cardiaque et d'accélérométrie sur des périodes qui peuvent varier de 24h à plusieurs jours selon le protocole d'étude. Il est ainsi possible de déterminer l'énergie dépensée par une personne durant son sommeil, au cours des repas, lors d'activités calmes ou bien lors d'efforts contrôlés (comme la marche ou le vélo).



Figure 1 : Photo d'une partie d'une des deux chambres calorimétriques du CRNH

### 1.3. Le Projet Modelheart

Le projet Modelheart a été réalisé sur 120 volontaires afin de déterminer plusieurs paramètres les concernant, notamment, leur dépense énergétique, leur rythme cardiaque, leurs mouvements, leur température et flux de chaleur corporelle. Ces volontaires ont été séparés en deux populations :

Les sujets appartenant à la première population (population P1) ont séjourné dans une chambre calorimétrique pendant une durée de 24h. Ils ont porté deux capteurs (ActiHeart et ArmBand) ainsi qu'un électrocardiogramme (ECG). Plusieurs mesures ont donc été effectuées toutes les minutes. Ils ont réalisé des activités diverses qui ont été notées dans un planning.

Les sujets appartenant à la deuxième population (population P2), quant à eux, ont vécu de façon normale (conditions habituelles de vie). Ils ont porté seulement les deux capteurs. De plus, au début de l'expérience, ils ont bu une eau marquée par deux isotopes  $H2^{18}O$  et  $2H2O$  dont la décroissance dans l'eau corporelle au cours du temps est révélatrice de la dépense énergétique. Les enregistrements des mesures ont duré 11 jours.

Les données recueillies proviennent donc de différentes sources (différents capteurs, différents volontaires et différentes conditions expérimentales). Les résultats sont enregistrés dans des fichiers Excel différents pour chaque capteur.

Pour commencer à travailler sur ce projet, nous avons d'abord dû analyser les données du problème et les besoins des futurs utilisateurs du programme informatique qu'on nous demandait de réaliser.

## 2. Analyse

### 2.1. Différentes informations

La première étape de cette analyse a été de rassembler toutes les informations que nous avons pour une expérience. Pour cela, le plus simple était de collecter les informations en fonction de leurs sources.

#### 2.1.1. Le Capteur ActiHeart

C'est un capteur qui a été posé sur le thorax des volontaires des deux populations. Il a mesuré chaque minute :

- la fréquence cardiaque brute (raw BPM) et nettoyée (cleaned BPM),
- l'accélérométrie selon l'axe longitudinal (Activity),
- la dépense énergétique liée à l'activité physique.

Certaines valeurs de fréquence cardiaque peuvent être aberrantes :

- celles inférieures à 30 battements/minute,
- celles qui sont supérieures à 30 battements/minute et qui augmentent ou diminuent avec un écart de 100 battements en l'espace d'une minute,
- celles qui sont supérieures à 1.75 fois la moyenne des fréquences cardiaques mesurées pendant les 4 minutes précédant le point analysé.

Dans ces 3 cas, le logiciel du capteur ActiHeart leur affecte arbitrairement une valeur brute égale à 0. Puis il propose des corrections (cleaning). Le logiciel détermine une autre valeur de la fréquence cardiaque en examinant le minimum et le maximum d'un intervalle interbattement (il y en a 64 par minute). Cette nouvelle valeur est comparée à la valeur correcte précédente (notée OK). Si cette nouvelle valeur ne diffère pas de plus de 30 battements de la valeur correcte précédente, alors elle remplace la valeur nulle et le terme « Recovered » signale la correction. Sinon la valeur reste égale à 0. Après ce « nettoyage » et en présence de valeurs résiduelles nulles (5 consécutives au maximum), le logiciel fait une interpolation. S'il y a plus de 5 valeurs nulles consécutives, les valeurs « corrigées » sont affectées de la valeur 0 (Lost). Toutefois, ces corrections ne nous ont pas paru satisfaisantes dans le cas des données manquantes aux extrémités de fichiers et lorsqu'il y en a beaucoup. C'est pourquoi nous avons testé d'autres algorithmes de remplacement des données manquantes.

Un exemple de données qui pourraient être obtenues par ce capteur est le suivant :

Cleaned BPM	Raw BPM	
66	66	OK
67	67	OK
71	71	OK
61	61	OK
62	62	OK
64	64	OK
63	58	OK
62	0	Interpolated
61	0	Interpolated
61	75	Recovered
0	0	Lost
60	0	Recovered

Figure 2 : Tableau représentant un exemple de données de fréquence cardiaque relevées par le capteur ActiHeart

### 2.1.2. Le Capteur ArmBand

C'est un capteur posé sur le bras droit des volontaires des deux populations. Il mesure à chaque minute :

- l'accélérométrie selon les axes : transversal et longitudinal,
- la température corporelle (°C),
- la température proximale (°C),
- le flux de chaleur : perte de chaleur par convection ( $W/m^2$ ),
- la sudation : perte de chaleur par évaporation (mesure d'impédance exprimée en micro Siemens),
- le nombre de pas,
- la position (debout/couché et endormi/éveillé),
- la dépense énergétique totale (liée à l'activité physique, au métabolisme de base et à la thermogénèse induite par la digestion) (Kcal/minute).

### 2.1.3. L'Electrocardiogramme

C'est un ensemble de 3 électrodes qui n'a été porté que par les volontaires de la population P1 (en chambre). Il renseigne sur la fréquence cardiaque.

#### 2.1.4. Paramètres capturés dans la chambre

Ils sont au nombre de six :

- la concentration en oxygène de l'air de la chambre,
- la concentration en CO<sub>2</sub> de l'air de la chambre,
- la température de la pièce,
- l'humidité relative de l'air de la chambre,
- la pression atmosphérique,
- le débit d'arrivée d'air.

Des calculs sont d'abord effectués sur la concentration en O<sub>2</sub> et en CO<sub>2</sub> puisqu'il existe un problème de dérive des capteurs. Puis, c'est à partir de ces six données acquises minute par minute que le calcul de la dépense énergétique est effectué selon le principe de la calorimétrie indirecte. Ce calcul de la dépense énergétique nous servira de mesure de référence.

#### 2.1.5. Données concernant les volontaires

Seuls des volontaires normo pondéraux en bonne santé ont été inclus dans cette étude. Une visite médicale préliminaire comprenant un examen cardiaque au repos et à l'effort a permis de vérifier l'état de santé des candidats.

Les données concernant les volontaires sont les suivantes :

- le sexe (autant d'hommes que de femmes ont été recrutés),
- l'âge (36-55 ans),
- la taille,
- le poids,
- l'indice de masse corporelle (BMI) calculé à partir du poids et de la taille (poids (kg)/taille<sup>2</sup> (m<sup>2</sup>)). Celui-ci devait être normal (entre 18 et 25).
- droitier/gaucher,
- fumeur/non fumeur (n'ont été recrutés que des non fumeurs ou ≤ 5 cigarettes par jour),
- pourcentage de masse maigre (masse musculaire et osseuse) mesuré par radiographie aux rayons X,
- capacité pulmonaire maximum (VO<sub>2</sub> max) (uniquement chez les volontaires de la population P1). Cette donnée est un résultat du test d'effort.
- fréquence cardiaque maximum (F<sub>cmax</sub>) qui est égale en théorie à 220 moins l'âge,
- apports énergétiques (24h),
- indice sportif (sédentaire à très actif) calculé à partir d'un questionnaire d'activité (MAQ). Cette donnée est aussi un résultat du test d'effort.

D'autre part, l'ensemble des volontaires a passé une visite médicale avant de commencer l'étude pour s'assurer de leur bon état de santé.

### 2.1.6. Mesure de la dépense énergétique par la technique de l'eau doublement marquée

Cette méthode a été utilisée pour les volontaires de la population P2. C'est une technique biochimique qui permet d'évaluer leur dépense énergétique globale sur les 11 jours en condition habituelle de vie. Au commencement de l'expérience, les volontaires boivent un flacon d'eau marquée par deux isotopes ( $H_2^{18}O$  et  $^2H_2O$ ). La mesure de la concentration isotopique dans les urines à la fin de l'expérience nous renseigne sur la dépense énergétique globale. A la différence des volontaires de la population P1, celle-ci est donc réduite à une seule valeur.

### 2.1.7. Planning des activités menées en chambre calorimétrique

Un planning de 24 activités a été établi pour la population P1 (figure 3).

	activité	heure-début théorique	Durée théorique
1	sommeil	0:00	6:00
2	Métabolisme de Base	7:00	1:00
3	questionnaire1	8:00	0:15
4	questionnaire2	9:00	0:10
5	écriture	9:15	0:15
6	debout-assis	9:30	0 :05
7	debout statique	9:35	0 :15
8	déambulation	9:50	0 :15
9	repos1	10:05	0 :25
10	marche3kmh	10:30	0 :20
11	repos2	10:50	0 :10
12	marche4kmh	11:00	0 :20
13	repos3	11:20	0 :30
14	questionnaire3	11:50	0 :10
15	déjeuner	12:00	0 :20
16	questionnaire4	13:00	0 :10
17	marche5kmh	14:00	0 :20
18	repos4	14:20	0 :30
19	stepper	14:50	0 :20
20	repos5	15:10	0 :30
21	marche6kmh	15:40	0 :20
22	repos6	16:00	0 :30
23	ménage	16:30	0 :20
24	questionnaire5	16:50	0 :10

*Figure 3 : Planning théorique*

Celui-ci correspond au planning théorique. Cependant, pour chaque volontaire, le planning réel a été enregistré. Par exemple, le volontaire x aura pu commencer son déjeuner à 12:03 au lieu de 12:00. Chaque planning est donc unique. La durée de chaque activité est enregistrée.

Les volontaires de la population P2, quant à eux, n'ont pas fait de liste des activités qu'ils ont réalisées. Cependant, ils ont noté l'heure à laquelle ils enlevaient leurs capteurs.

## 2.2. Diagramme de classe

Nous avons récolté ces informations au fil de plusieurs réunions que nous avons menées en présence de Madame Sylvie ROUSSET et de nos responsables ISIMA, Messieurs Philippe LACOMME et Gérard FLEURY. Durant les premiers mois, nous nous sommes rencontrés une fois par semaine.

Lors de chaque réunion, nous présentions un diagramme de classe qui décrivait l'organisation des données telle que nous l'avions perçue et surtout telle que nous allions l'implémenter.

Les retouches ont été nombreuses. En effet, à chaque réunion nous parlions d'éléments nouveaux, de fichiers Excel qui n'avaient pas été mentionnés. De plus, nos premières esquisses ont été invalidées puisque nous n'avions pas pris en compte le fait que le logiciel devait rester valable pour de futurs projets.

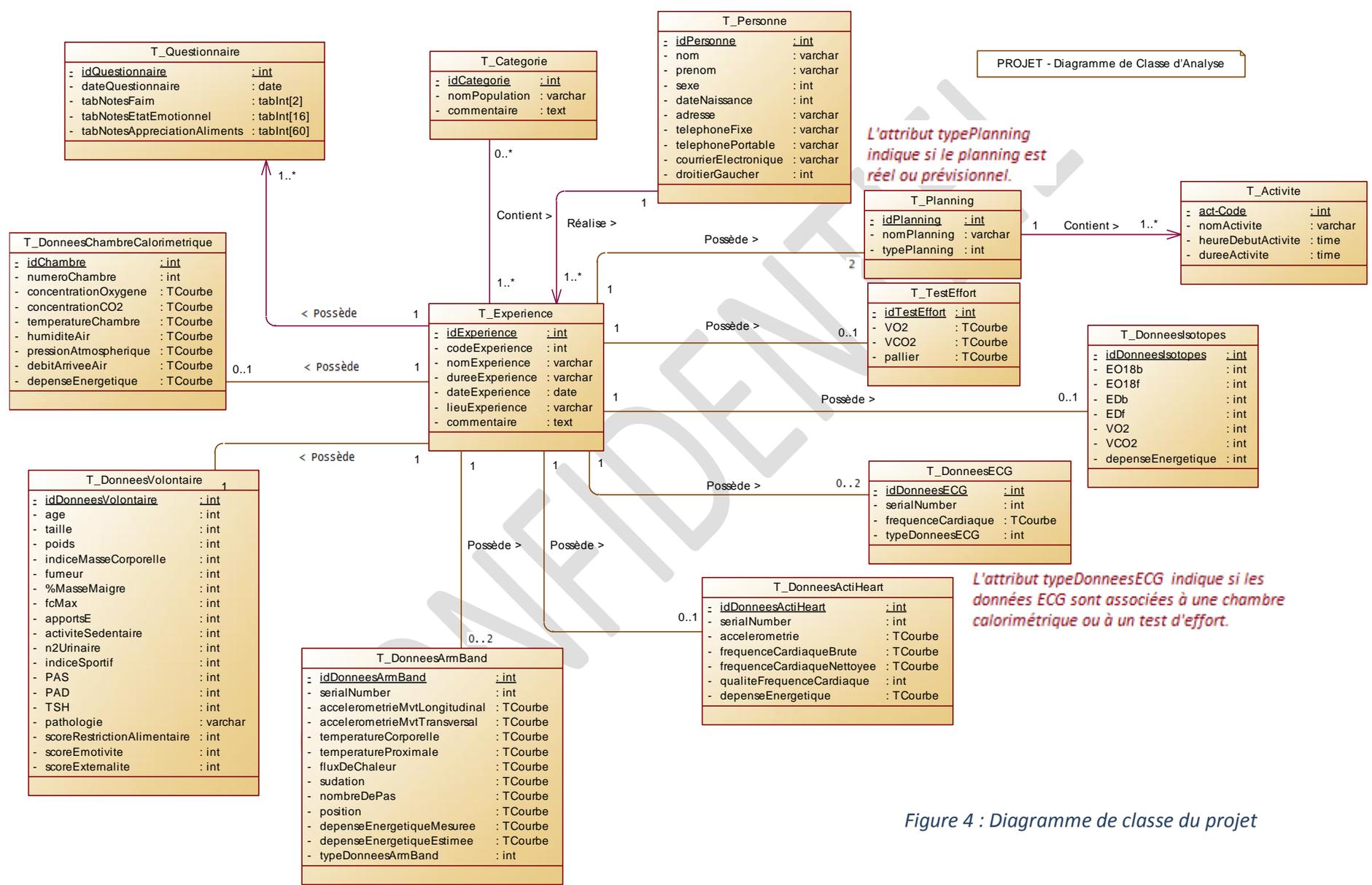
C'est pour cette raison que nous avons choisi de placer au centre du diagramme une classe *T\_Expérience*. Ce sont elles qui sont au centre de chaque projet. Celui-ci ne sera constitué que d'expériences qui auront des particularités. Par exemple, nous avons créé une table *T\_DonnéesChambreCalorimétrique* qui sera absente pour une expérience où le volontaire sera étudié à domicile.

Une expérience (*T\_Expérience*) comprend donc des informations permanente sur la personne qui la passe (*T\_Personne*), sur ses caractéristiques physiologiques et psychologiques (*T\_Volontaire*), ainsi que des données provenant des capteurs (*T\_DonneesArmBand*, *T\_DonneesActiHeart*, *T\_DonneesECG*), de la chambre (*T\_DonnéesChambreCalorimétrique*), du planning (*T\_Planning*), du test d'effort (*T\_TestEffort*), des isotopes (*T\_DonneesIsotopes*) et des différents questionnaires que le volontaire a rempli à des moments précis du planning (*T\_Questionnaire*).

De plus, une table *T\_Catégorie* a été ajoutée puisque, comme nous le verrons plus tard, l'utilisateur va avoir besoin de déterminer des catégories de personnes pour réaliser des calculs. La catégorie peut être par exemple celle des hommes sédentaires.

Chaque table sera représentée par un fichier Excel gardé en mémoire dans le logiciel.

Le diagramme de classe en question est présenté figure 4.



L'attribut typePlanning indique si le planning est réel ou prévisionnel.

L'attribut typeDonneesECG indique si les données ECG sont associées à une chambre calorimétrique ou à un test d'effort.

L'attribut typeDonneesArmBand indique si les données ArmBand sont associées à une chambre calorimétrique ou à un test d'effort.

Figure 4 : Diagramme de classe du projet

## 2.3. Étapes à suivre

### 2.3.1. Étape 1 : Enregistrement d'une expérience

La première étape à réaliser devait être l'enregistrement complet d'une expérience. Cela implique que tous les fichiers soient entrés dans le logiciel et qu'ils soient traités en même temps. Toutefois, certains types de fichiers tel que le planning des activités qui n'ont pas de données manquantes, n'ont pas besoin d'être traités (figure 5). Seuls ceux concernant les données de la chambre calorimétrique et ceux des capteurs doivent être modifiés. De plus, leurs traitements seront différents (figure 6).

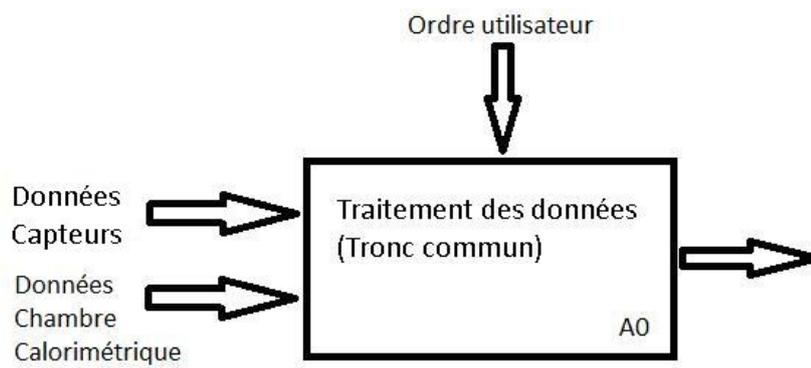


Figure 5 : Actigramme montrant quelles données doivent être traitées

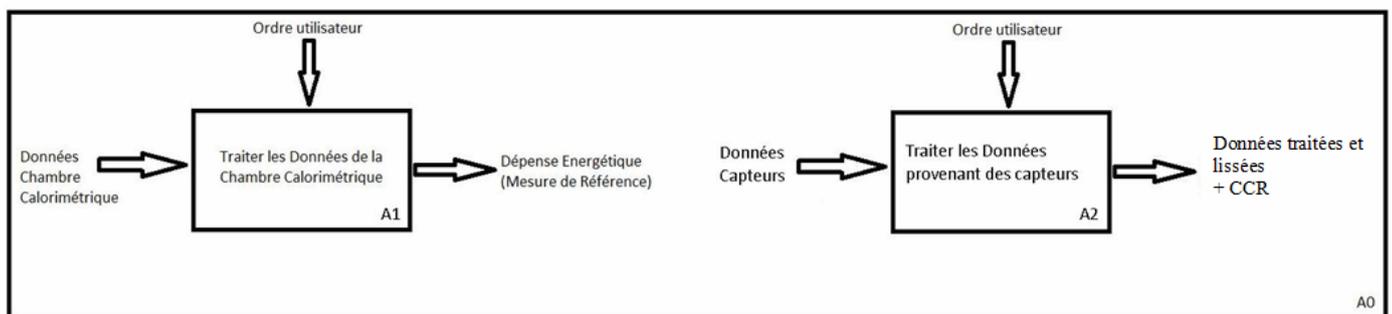


Figure 6 : Actigramme A0 montrant la différence de traitement

### 2.3.1.1. Traitement des données provenant des capteurs

Les différentes démarches qu'il faut adopter pour traiter les données provenant des capteurs ActiHeart et ArmBand ainsi que leur ordre sont présentées dans l'actigramme ci-dessous (figure 7) :

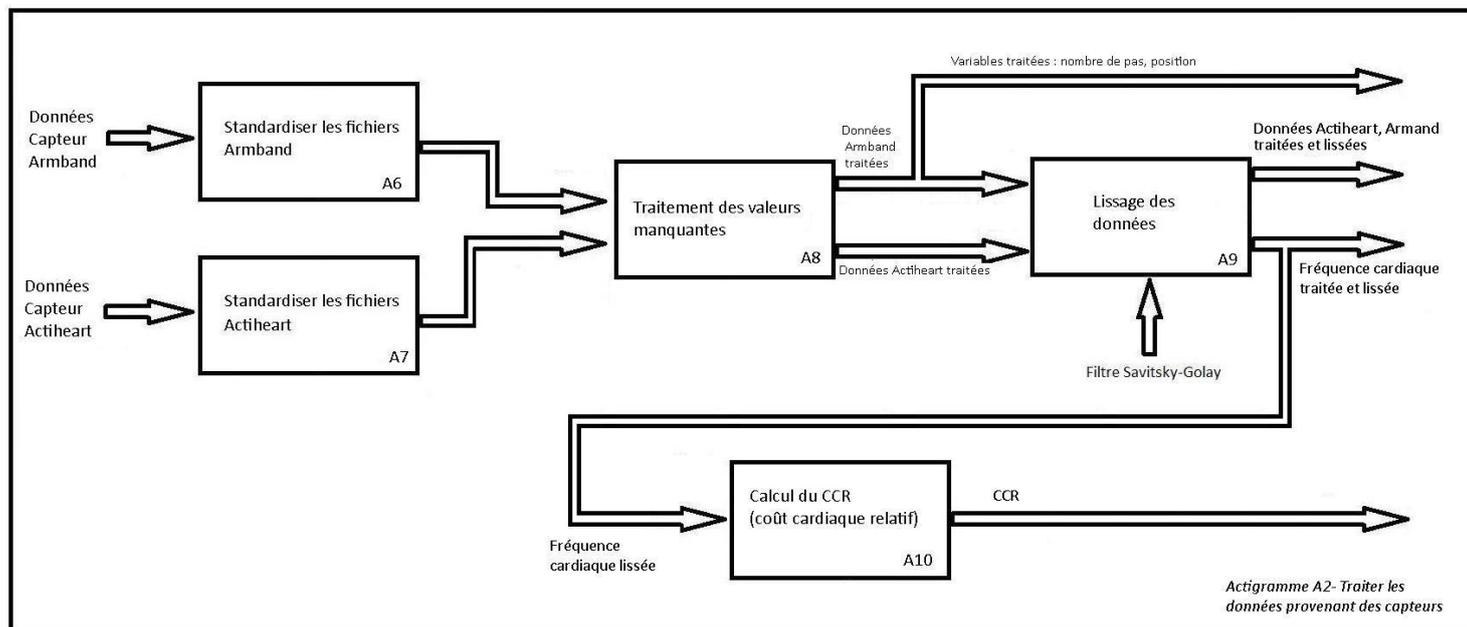


Figure 7 : Actigramme A2 - Traiter les données provenant des capteurs

Comme nous pouvons le voir, il faut tout d'abord standardiser les fichiers ArmBand et ActiHeart. Cela va se faire de manière automatique. Pour les fichiers ActiHeart, il suffit de tronquer le fichier à une heure bien précise de début mais aussi de fin. En effet, parfois l'enregistrement a débuté avant que la personne ne porte le capteur. Pour les fichiers ArmBand, après l'avoir coupé aussi, il faut supprimer des lignes blanches et retrouver les lignes manquantes. Cela sera expliqué dans le paragraphe 3.1..

Ensuite, il va falloir traiter les valeurs manquantes. Les résultats recueillis jusqu'à présent laissent apparaître un grand nombre de données manquantes. Celles-ci sont pour l'instant notées arbitrairement 0 pour l'ActiHeart, ou par rien (blanc) pour l'ArmBand. Cela peut-être dû à plusieurs causes : soit le capteur se décolle de la peau, soit les sujets ont retiré ponctuellement les capteurs (douches)... Un des objectifs est donc d'établir une analyse mathématique résolvant ce problème. Par exemple, si les données manquantes successivement sont peu nombreuses ( $\leq 5$ ), on peut envisager de faire de l'interpolation linéaire simple. Ce traitement sera explicité dans le paragraphe 3.2..

Les données seront alors lissées grâce au filtre de Savitsky-Golay, dont le principe est fondé sur la méthode des moindres carrés. En effet, les utilisateurs ont souhaité lisser les données car elles sont généralement bruitées : elles montrent des artéfacts ou des évolutions brusques et instantanés des signaux qui n'ont pas raison d'être. Cette méthode de lissage fera disparaître ces artéfacts.

Enfin, les utilisateurs ont souhaité disposer de variables transformées pour limiter la variabilité individuelle. Ainsi à partir de la fréquence cardiaque, un simple calcul permettra de calculer le coût cardiaque (CCR) relatif à l'instant t :  $CCR = (F_c(t) - F_{c\min}) / (F_{c\max} - F_{c\min})$ .

### 2.3.1.2. Traitement des données provenant de la chambre calorimétrique

Les différentes démarches qu'il faut adopter pour traiter les données provenant de la chambre calorimétrique ainsi que leur ordre sont présentés dans l'actigramme ci-dessous (figure 8) :

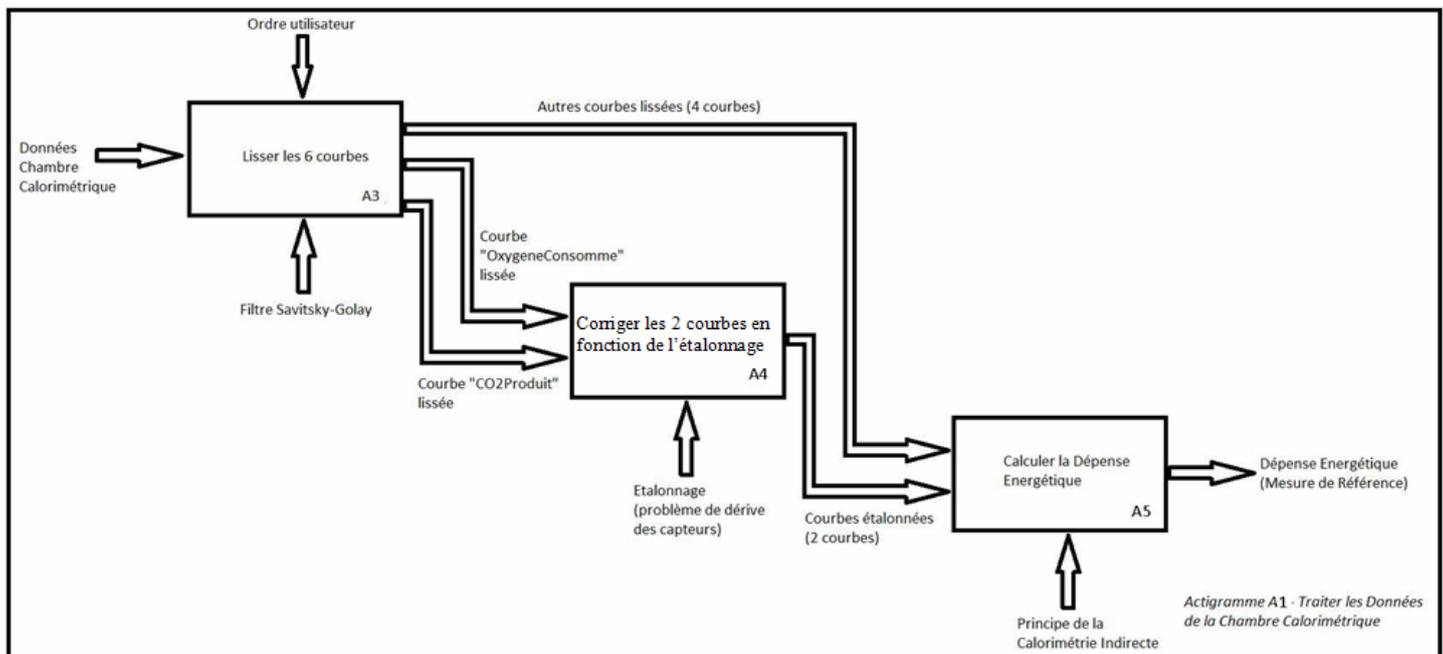


Figure 8 : Actigramme A1 - Traiter les données de la chambre calorimétrique

Cette figure montre que le lissage des données est la première étape. Ensuite, il faudra corriger les concentrations en O<sub>2</sub> et CO<sub>2</sub> en raison de la dérive de l'appareil de mesure et des changements de la pression atmosphérique. Enfin, la dépense énergétique sera calculée à partir de l'équation élaborée par Weir (1949).

### 2.3.2. Etape 2 : Traitement mathématiques

Une fois les expériences enregistrées, il sera désormais possible de réaliser différents traitements mathématiques. Les objectifs de cette partie sont de pouvoir :

- calculer des fonctions dérivées première et seconde,
- rechercher la valeur minimale, maximale ainsi que la moyenne des paramètres mesurés et de leurs dérivées,
- détecter la périodicité de l'activité physique des sujets,
- visualiser l'évolution de tous les paramètres au cours du temps, pris deux à deux,
- générer des graphiques tels que les boîtes à moustaches pour estimer la répartition des valeurs de Coût Cardiaque Relatif, d'accélérométrie et de dépense énergétique (faible, moyenne, forte) au cours de la journée ou pendant les 11 jours d'enregistrement...

D'autres modes graphiques seraient intéressants à élaborer pour mettre en évidence des cycles d'activités.

---

Toutes ces évaluations pourront être effectuées sur 24h ou sur des périodes d'activité contrôlées (en chambre calorimétrique) ou encore sur la période de 11 jours (en conditions habituelles de vie), pour un seul individu ou pour une famille d'individus (par exemple : hommes versus femmes, sédentaires versus sportifs).

Il est important de mettre l'accent sur la transparence des options et des calculs. L'utilisateur devra être totalement autonome. Il choisit la fonction mathématique et les fichiers Excel qu'il veut traiter. D'autres options lui seront aussi proposées telles que la notion de familles d'individus ou de période d'évaluation... De cette façon, l'outil pourra être largement réutilisable dans le cadre d'autres projets.

### 2.3.3. Etape 3 : Détermination de la fonction finale de prédiction de la dépense énergétique

Le but général de ce projet est de modéliser la dépense énergétique des volontaires en fonction de leur fréquence cardiaque, de leur paramètre de mouvement, de leur température corporelle, du flux de chaleur dissipé par leur peau et la conductance de leur peau.

Le scientifique devra pouvoir choisir des paramètres présents dans la base de données (accélérométrie du capteur ArmBand, âge du volontaire...), puis rechercher les coefficients associés à ces paramètres qui permettent d'expliquer le plus fidèlement possible la dépense énergétique mesurée soit par calorimétrie indirecte soit par la technique de l'eau doublement marquée.

On peut imaginer cette étape sous la forme d'une calculatrice où les éléments de la base de données seront les touches de cette dernière.

Ce projet aborde donc de multiples aspects. Après l'analyse, notre tâche a été de réaliser le début de l'étape 1, c'est-à-dire la standardisation des fichiers et la gestion des valeurs manquantes.

## 3. Modifications des fichiers Excel

Dans cette partie, nous allons présenter les techniques qui ont été choisies, les problèmes qu'elles ont engendrés et quels algorithmes ont été utilisés.

### 3.1. Standardisation

#### 3.1.1. Librairie JExcelApi

Avant de commencer à standardiser les fichiers, il a fallu réfléchir aux technologies que nous allions utiliser. Pour des raisons de portabilité, le langage JAVA a été choisi. Nous avons alors fait des recherches pour savoir comment manipuler des fichiers Excel dans un logiciel écrit en JAVA.

Une librairie a semblé parfaitement convenir : la librairie JExcelApi [JEXCELAPI]. C'est une java API open source permettant aux développeurs de lire, d'écrire et de modifier dynamiquement des feuilles de calculs Excel.

Nous avons donc inclus la librairie dans le programme et appris à nous en servir grâce aux tutoriaux. Sa particularité vient du fait qu'il est impossible de modifier un fichier directement. Il faut ouvrir le fichier en mode lecture et en créer un nouveau en mode écriture. Le logiciel va faire la copie du premier fichier dans le second et c'est ce dernier qui sera modifié. Il n'y a donc aucun danger pour le fichier d'origine.

### 3.1.2. Standardisation ActiHeart

La première étape, comme nous l'avons dit précédemment consistait à standardiser les fichiers provenant des capteurs. Ceux concernant l'ActiHeart n'ont pas posé de grandes difficultés.

La fonction StandardisationActiHeart, qui a été codée, commence par s'assurer que le fichier correspond bien à des données ActiHeart puis elle va « couper » le fichier à une heure précise. Celle-ci sera donnée par l'utilisateur. En effet, bien souvent l'enregistrement a commencé avant que le capteur ne soit porté. Le programme parcourt le fichier et, tant qu'il ne trouve pas l'heure dite, il supprime la ligne. Le logiciel conserve ensuite les lignes correspondant à la durée de l'expérience indiquée par l'utilisateur. Enfin, il supprime les lignes au-delà de l'expérience. Ci-dessous vous est présenté l'extrait du code qui illustre le fonctionnement de JExcelApi pour des opérations simples (figure 9).

```
//Tant que ce n'est pas l'heure du début
while((hour != heureDeb)|| (min != minuteDeb))
{
    //on supprime la ligne
    sheet.removeRow(i);
    cell = sheet.getWritableCell(1,i);
    dt = (DateTime) cell;
    hour = dt.getDate().getHours();
    min = dt.getDate().getMinutes();
}

// On avance de durée
if (typejour) //durée en jour
    i = i+duree*24*60 +1;
else // durée en minute
    i = i+duree*60+1;

// On supprime les dernières lignes
cell = sheet.getWritableCell(1,i);
while (cell.getType() != CellType.EMPTY)
{
    sheet.removeRow(i);
    cell = sheet.getWritableCell(1,i);
}
```

Figure 9 : Extrait du code de la standardisation ActiHeart

Le programme va aussi supprimer les données aberrantes, c'est-à-dire les valeurs de fréquence cardiaque supérieures à 150 qui sont associées à la mention « Interpolated », « Recovered », « not worn » et « lost ».

### 3.1.3. Standardisation ArmBand

La standardisation des fichiers du capteur ArmBand a posé plus de problèmes. Elle est présente dans la fonction StandardisationArmband. Le programme commence par supprimer les lignes blanches présentes au milieu des fichiers. Ensuite, et comme pour les fichiers ActiHeart, le fichier est tronqué en fonction de l'heure de début et de la durée de l'expérience. De plus, une colonne est ajoutée : la colonne Accélérométrie globale qui résulte du calcul de la racine carrée de la somme des accélérométries transversale et longitudinale élevées au carré.

Un traitement spécifique supplémentaire est nécessaire pour rechercher les lignes et valeurs manquantes. En effet le logiciel du ArmBand n'inscrit tout simplement aucune ligne ni valeur quand elles sont manquantes. Il faut donc retrouver les « lignes absentes » dans les horaires et faire figurer leur contenu en tant que valeurs manquantes pour qu'elles puissent être traitées. Ceci est illustré sur les figures 10 et 11 ci-dessous.

	A	B	C
1	Heure CC	Accel.Transversale - pics	Accel Longitudina
797	2009-01-29 21:29:00,000		264
798	2009-01-29 21:30:00,000		338
799	2009-01-29 21:31:00,000		398
800	2009-01-29 21:32:00,000		335
801	2009-01-29 21:33:00,000		306
802	2009-01-29 21:40:00,000		346
803	2009-01-29 21:41:00,000		337
804	2009-01-29 21:42:00,000		334

*Figure 10 : Fichier ArmBand d'origine*

	A	B	C	D
1	Heure CC	Accéléromé	Accel.Transversale - pics	Accel Longitud
783	2009-01-29 21:31:00,000	585,188004	398	
784	2009-01-29 21:32:00,000	505,831988	335	
785	2009-01-29 21:33:00,000	480,141646	306	
786	2009-01-29 21:34:00,000	-1	0	
787	2009-01-29 21:35:00,000	-1	0	
788	2009-01-29 21:36:00,000	-1	0	
789	2009-01-29 21:37:00,000	-1	0	
790	2009-01-29 21:38:00,000	-1	0	
791	2009-01-29 21:39:00,000	-1	0	
792	2009-01-29 21:40:00,000	546,484218	346	
793	2009-01-29 21:41:00,000	516,18795	337	
794	2009-01-29 21:42:00,000	495,491675	334	

*Figure 11 : Fichier ArmBand corrigé*

Le problème majeur de ce travail est lié aux dates. Il est très vite apparu que les dates qui apparaissaient dans le fichier n'étaient pas celles lues par JExcelApi. Par exemple, lorsque 21h34 était affiché, le logiciel lisait 22h34. En effet, elles étaient lues en heure GMT. C'est le temps solaire moyen au méridien de Greenwich. Cela a créé des difficultés jusqu'à ce que nous nous en apercevions, pour

**TRAITEMENT MATHÉMATIQUE DE DONNÉES CARDIAQUES :  
ETAPE PRELIMINAIRE DE LA PREDICTION DE LA DEPENSE ÉNERGETIQUE**

Octobre 2009 - Mars 2010

couper les fichiers par exemple. Cela a rapidement été résolu en soustrayant 1 à l'heure et en faisant attention pour le passage à minuit.

Cependant, le problème persistait sur certains fichiers. Il y avait encore une heure de décalage. La raison était que ces expériences avaient été réalisées en heure d'été. Il y avait donc cette fois deux heures de différence. Nous avons donc mis en place un code permettant de déterminer si l'expérience s'est déroulée en heure d'été ou d'hiver, et qui adapte l'ajustement en fonction de cette information.

Au moment où nous écrivons ce rapport, un problème de dates encore différent des précédents est en voie d'être résolu. Il est cette fois lié au jour. Lorsque une ligne blanche se situe à l'endroit où se trouvent des valeurs manquantes, le programme réalise la standardisation mais se trompe dans le jour. Voici un exemple (figures 12 et 13) :

	A	B	
1	Heure CC	Accel.Transversale - pics	Accel Lon
13727	2009-02-07 23:07:00,000	181	
13728	2009-02-07 23:08:00,000	204	
13729	2009-02-07 23:09:00,000	361	
13730	2009-02-07 23:09:34,590		
13731	2009-02-07 23:09:50,030		
13732	2009-02-07 23:12:00,000	269	
13733	2009-02-07 23:13:00,000	243	
13734	2009-02-07 23:14:00,000	200	

*Figure 12 : Cause du problème de jour dans le fichier ArmBand*

	A	B	C	
1	Heure CC	Accéléromé	Accel.Transversale - pics	Ac
13840	2009-02-07 23:08:00,000	270,739727		204
13841	2009-02-07 23:09:00,000	504,905932		361
13842	2009-02-08 23:10:00,000	480,348643		-1
13843	2009-02-08 23:11:00,000	431,234065		-1
13844	2009-02-07 23:12:00,000	406,676776		269
13845	2009-02-07 23:13:00,000	381,42496		243
13846	2009-02-07 23:14:00,000	320,51521		209

*Figure 13 : Problème du jour dans le fichier ArmBand*

Or, le programme ne modifie à aucun moment le jour. Il réagit comme s'il gardait en mémoire la ligne qui avait été supprimée. Nous avons essayé de forcer le Garbage Collector mais cela n'y a rien fait. Nous sommes donc, à l'heure actuelle, en train d'essayer de résoudre ce problème.

## 3.2. Valeurs manquantes

### 3.2.1. ArmBand

Les fichiers ArmBand contiennent des valeurs manquantes lorsque les volontaires en conditions habituelles de vie ont enlevé le capteur quand ils prenaient une douche ou qu'ils allaient à la piscine. Le laps de temps pour une douche est d'à peu près 10 à 15 minutes. Le cas de la piscine est beaucoup moins fréquent mais quand il se présente, une heure d'acquisition de données au moins peut manquer. Ces intervalles étaient trop grands pour pouvoir faire des prédictions convenables à l'aide de techniques mathématiques poussées. Nous avons donc réfléchi à une solution avec Monsieur FLEURY, qui nous a conseillés de prendre simplement la moyenne sur les 5 valeurs qui précèdent et les 5 qui suivent (en ignorant les 0 qui pourraient survenir) pour remplacer les valeurs manquantes. La moyenne des valeurs prochaine encadrant celles qui sont manquantes peut être les plus pertinentes. Ceci concerne les variables d'Accélérométrie, de Flux de chaleur, de Température de la peau, de Température proximale, d'Impédance moyenne et de Dépense énergétique. Les variables discrètes telles que le nombre de pas ainsi que celles qui ne serviront pas à la fonction finale n'ont pas été traitées. Un exemple de résultat est le suivant (figure 14) :

Heure CC	Accéléromé	Accel.Transv	Accel Longitu	Flux de ch
2009-10-01 07:05:00,000	434,728651	283	330	101,80077
2009-10-01 07:06:00,000	395,773925	251	306	98,375434
2009-10-01 07:07:00,000	309,012945	217	220	90,710212
2009-10-01 07:08:00,000	383,345797	265	277	87,034400
2009-10-01 07:09:00,000	175,855054	138	109	82,773811
2009-10-01 07:10:00,000	435,965595	279	335	97,257881
2009-10-01 07:11:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:12:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:13:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:14:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:15:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:16:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:17:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:18:00,000	381,229655	-1	-1	135,73574
2009-10-01 07:19:00,000	448,325774	314	320	195,19790
2009-10-01 07:20:00,000	449,027839	299	335	187,21476
2009-10-01 07:21:00,000	368,781778	240	280	158,20065
2009-10-01 07:22:00,000	412,237795	274	308	174,38641
2009-10-01 07:23:00,000	433,970045	283	329	186,20596
2009-10-01 07:24:00,000	466,647619	292	364	179,70031

Figure 14 : Traitement des valeurs manquantes d'un fichier ArmBand

### 3.2.2. ActiHeart

Le traitement des valeurs manquantes ne concerne ici que la fréquence cardiaque présente dans les fichiers ActiHeart. Il a été plus difficile car les données manquantes sont beaucoup plus nombreuses que celles des fichiers ArmBand. Dans ce cas, prendre les moyennes de valeurs précédentes et suivantes n'aurait pas eu de sens. Il a fallu inventer un algorithme de remplissage qui respecterait la logique de l'expérience. C'est Madame ROUSSET qui nous a fourni à chaque séance les pistes de réalisation. Nous codions alors par étapes : si les résultats étaient corrects nous validions, sinon nous affinions le modèle.

L'algorithme suivant a été retenu (figure 15) et a été programmé dans le logiciel.

Calcul de la fréquence de sommeil :  
Moyenne des données respectant les contraintes suivantes :  
(0<heure<6) et (sommeil = 1) et (mention = "OK") et (Allongé = 1) et (Accélérométrie <= 5)

Calcul de la fréquence de repos :  
Moyenne des données respectant les contraintes suivantes :  
(heure > 6) et (Accélérométrie < 35) et (sommeil = 0) et (mention = "OK") et (Allongé = 1)

Remplacement des valeurs nulles :  
Si nb données manquantes < 5 alors  
interpolation linéaire

Sinon  
Si (0<heure<6) et (nb de données utilisées dans la fréquence de sommeil >= 60) alors  
Fréquence de sommeil

Sinon  
Si (heure > 6) et (Accélérométrie < 35) et (nb de données utilisées dans la fréquence de repos > 30) alors  
Fréquence de repos

Sinon  
Régression linéaire

fsi

fsi

*Figure 15 : Algorithme des valeurs manquantes des fichiers ActiHeart*

Tout d'abord, s'il y a moins de 6 valeurs manquantes successivement, le logiciel réalise une interpolation linéaire (c'est comme si on traçait une droite entre le point d'avant et le point d'après). Un exemple se trouve figure 16.

Date	Time	Edited Activit	Raw Activity	Cleaned BPM	Raw BPM	
24-sept-09	18:34:00	21	21	94		94 OK
24-sept-09	18:35:00	67	67	87		86,75 Interpol:
24-sept-09	18:36:00	148	148	80		79,5 Interpol:
24-sept-09	18:37:00	27	27	72		72,25 Interpol:
24-sept-09	18:38:00	8	8	65		65 OK

*Figure 16 : Interpolation pour moins de 6 valeurs manquantes*

Ensuite, les fréquences de sommeil et de repos sont calculées et le logiciel remplace les valeurs comme indiquées dans l'algorithme. La difficulté a été de vérifier pour chaque valeur si le volontaire était allongé ou non, et endormi ou non. En effet, les variables Allongé et Endormi se trouvent dans le fichier ArmBand. Il a donc fallu ouvrir le fichier ArmBand traité et s'en servir. Pour ne pas rencontrer de problèmes avec les heures, nous nous sommes servis des numéros de lignes. Puisque les deux fichiers étaient tronqués à la même heure, la première ligne de chacun d'eux correspondait au même horaire. La deuxième aussi et ainsi de suite.

Enfin, pour remplacer les quelques valeurs manquantes résiduelles de fréquence cardiaque, nous avons envisagé de déterminer les valeurs de remplacement à partir de la régression linéaire [REGRESSION] entre les valeurs d'accélérométrie et de fréquence cardiaque. Il s'agit de proposer un modèle linéaire, c'est-à-dire de rechercher la droite dont l'équation est  $y_i = ax_i + b$  où  $x_i$  représentent les points de la fréquence cardiaque et  $y_i$  les points de l'accélérométrie. Cette droite devra passer au plus près des points du graphe. Passer au plus près, selon la méthode des moindres carrés, c'est

rendre minimale la somme des carrés des écarts des points à la droite 
$$\sum_{i=1}^n (y_i - ax_i - b)^2$$
.

Pour résoudre cela, on utilise :

$$a = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2} \text{ et } b = \frac{\sum_{i=1}^n y_i \sum_{i=1}^n x_i^2 - \sum_{i=1}^n x_i \sum_{i=1}^n x_i y_i}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2}.$$

Nous avons testé deux types de régression : soit sur toutes les valeurs, soit sur les 5 données précédentes et les 5 données suivantes la plage de données manquantes. Il est apparu des résultats aberrants avec la méthode des 2 fois cinq points. Nous avons donc gardé la régression sur l'ensemble des données.

Pour finir, le programme écrit, à la fin du fichier, la valeur calculée de la fréquence de repos et de sommeil ainsi que le nombre de valeurs encore manquantes. Cela a été très utile au fur et à mesure que nous codions les étapes pour savoir combien de valeurs avaient été modifiées par l'interpolation linéaire puis par chacun des autres algorithmes.

Les résultats finaux sont facilement appréciables comme nous pouvons le voir sur les figures 17 et 18 où la première figure montre la fréquence cardiaque brute en l'absence de traitement et la deuxième est le résultat fourni par le programme pour un fichier donné.

TRAITEMENT MATHÉMATIQUE DE DONNÉES CARDIAQUES :  
ÉTAPE PRÉLIMINAIRE DE LA PRÉDICTION DE LA DÉPENSE ÉNERGÉTIQUE

Octobre 2009 - Mars 2010

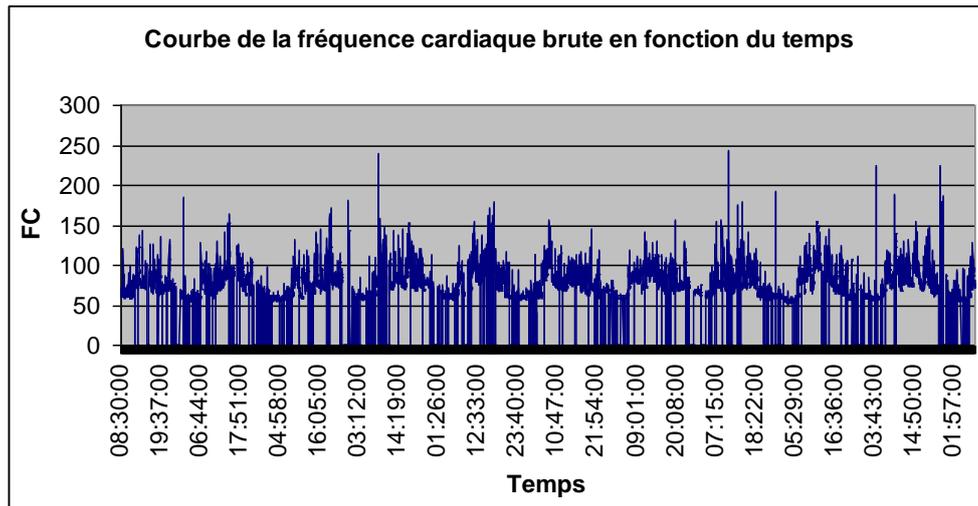


Figure 17 : Graphique représentant la fréquence cardiaque brute en fonction du temps

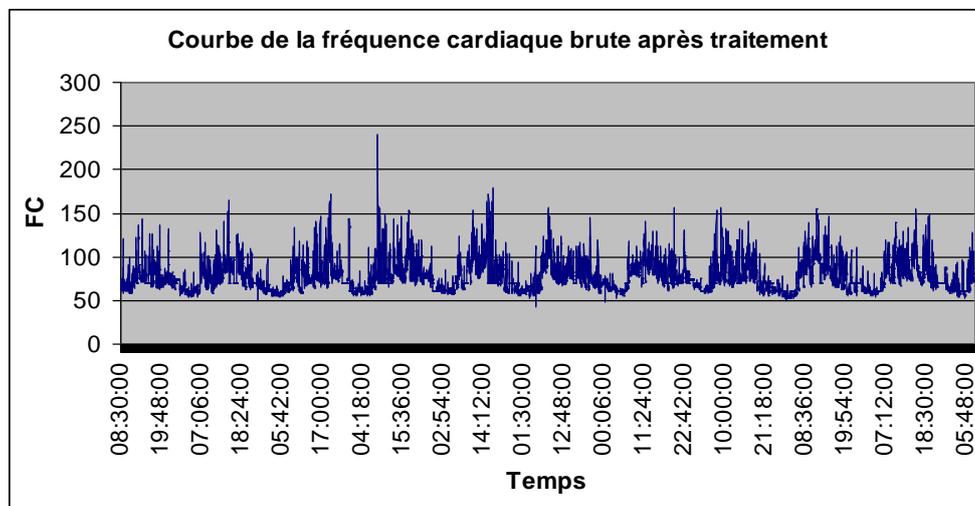


Figure 18 : Graphique représentant la fréquence cardiaque brute en fonction du temps mais sur un fichier traité

On peut remarquer qu'il ne reste aucune valeur nulle. Par contre, des données aberrantes sont encore présentes et devraient être supprimées au lissage.

Pour pouvoir tester ces traitements et pour que Madame ROUSSET puisse faire des validations, il fallait lui fournir au fur et à mesure un outil qui soit simple d'utilisation.

## 4. Réalisation graphique

### 4.1. Choix technique

Le logiciel a été développé à l'aide d'un Environnement de Développement Intégré (EDI), ou en anglais Integrated Development Environment (IDE). Il s'agit d'un logiciel qui regroupe, au minimum, à la fois un éditeur et un compilateur. Mais dans la plupart des cas, un EDI est beaucoup plus que ces deux outils et peut en comprendre un très grand nombre dont le plus courant est le débogueur qui permet de suivre pas à pas l'exécution du programme. Le but d'un tel logiciel est de faciliter la programmation en n'ayant à utiliser qu'un programme pour toutes les phases de la conception d'un logiciel : écriture du code, compilation, débogage et tests.

L'EDI utilisé pour le développement du logiciel dans le cadre de notre projet est NetBeans [NETBEANS] avec la version 6.8.. Cet EDI, essentiellement pour le langage JAVA, possède un éditeur, un compilateur et un débogueur très complet qui est très utile lors de la recherche d'erreurs de programmation. De plus, NetBeans incorpore un environnement graphique de développement qui permet de créer des fenêtres beaucoup plus facilement. Il suffit de positionner les éléments que l'on souhaite ajouter directement dans la fenêtre à l'aide de la souris. Par ailleurs, une fois les éléments positionnés, il est possible de modifier leurs propriétés directement à l'aide de menus du logiciel, évitant ainsi de devoir aller programmer l'ajout et les propriétés de chaque élément. NetBeans englobe aussi d'autres outils utiles et performants, comme la coloration syntaxique du code.

## 4.2. Présentation générale du logiciel

Afin de faciliter le développement du logiciel, deux modes de fonctionnement ont été mis en place : le « mode automatique » et le « mode manuel ». Le « mode automatique » est accessible à l'aide des boutons « Nouveau Projet » (figure 19), pour démarrer la saisie des groupes et des expériences d'un projet, et « Ouvrir Projet » sur la fenêtre d'accueil. Ce dernier est désactivé car la fonction « Ouvrir un projet » n'a pas été installée. Le « mode automatique » sera expliqué tout au long de cette partie du rapport. Le « mode manuel », situé sur la première fenêtre, permet à l'utilisateur de tester les améliorations nouvellement créées mais non intégrées dans le fonctionnement automatique. Lorsqu'elles auront été approuvées par Madame ROUSSET, elles seront accessibles dans le premier mode. Toutes les fonctionnalités développées pendant notre stage ayant été validées, le bouton « Mode Manuel » a été désactivé sur la fenêtre d'accueil.

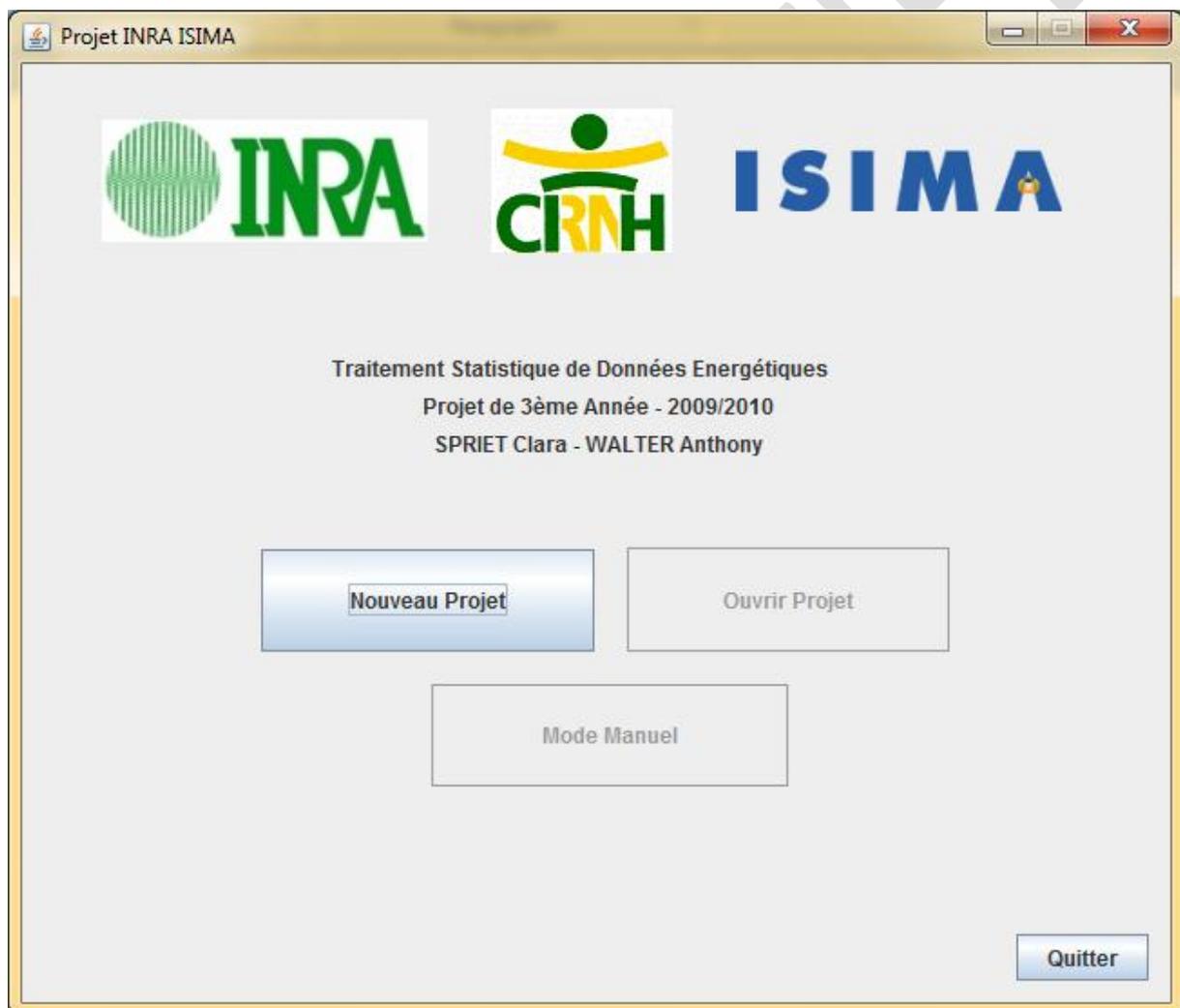


Figure 19 : Fenêtre d'accueil du logiciel

**TRAITEMENT MATHEMATIQUE DE DONNEES CARDIAQUES :  
ETAPE PRELIMINAIRE DE LA PREDICTION DE LA DEPENSE ENERGETIQUE**

Octobre 2009 - Mars 2010

Lorsque l'utilisateur choisit le « mode automatique » en appuyant sur le bouton « Nouveau Projet », une fenêtre s'ouvre. Pour démarrer un nouveau projet, il faut saisir le nom et le nombre de groupes d'expériences (figure 20).

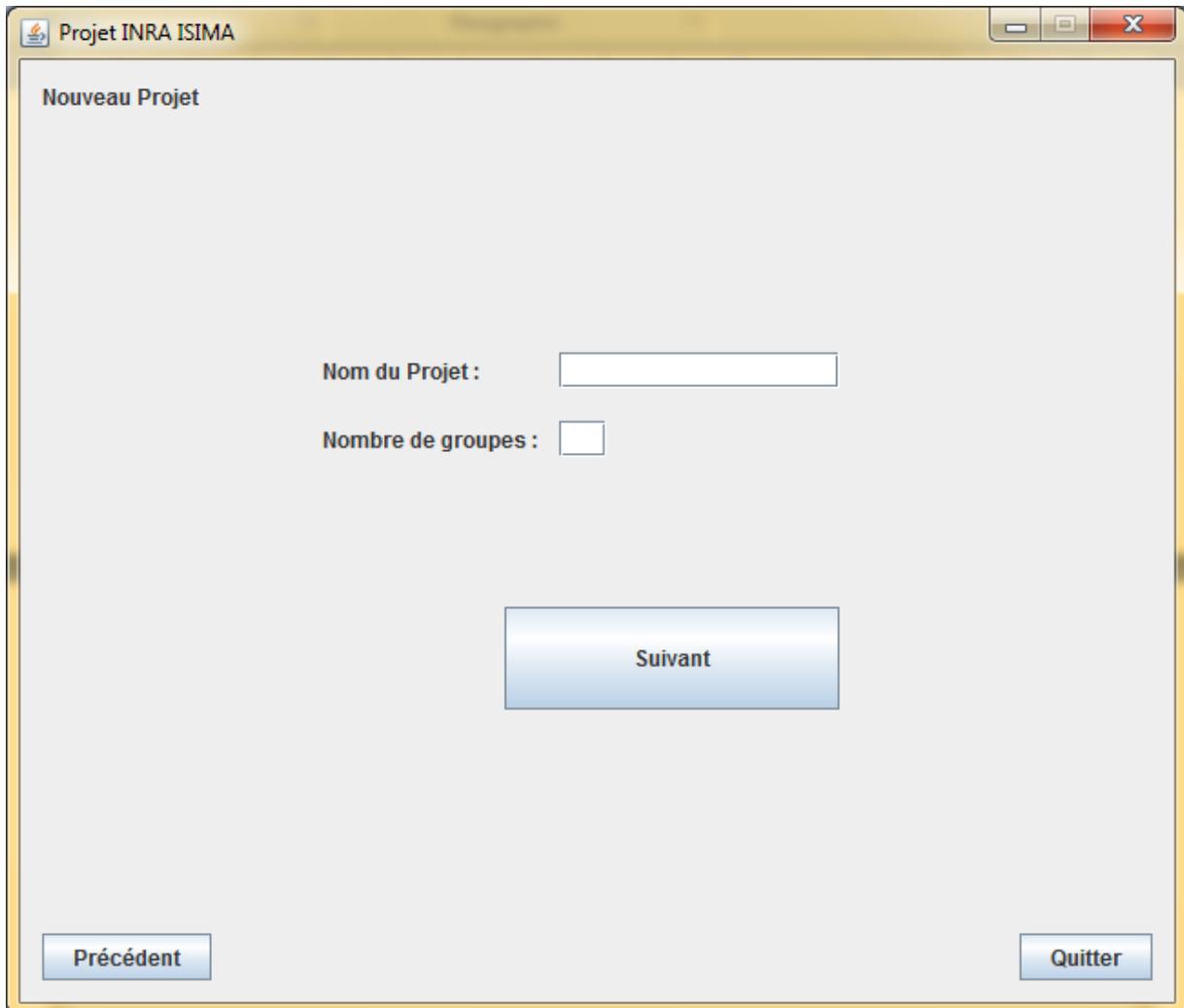


Figure 20 : Fenêtre « Nouveau Projet »

**TRAITEMENT MATHÉMATIQUE DE DONNÉES CARDIAQUES :**  
**ÉTAPE PRÉLIMINAIRE DE LA PRÉDICTION DE LA DÉPENSE ÉNERGÉTIQUE**

Octobre 2009 - Mars 2010

La notion de groupes est un concept essentiel dans la création d'un projet. Elle permet de classer les expériences dans un ou plusieurs groupes qui sont définis selon deux paramètres : l'heure du début de l'expérience et la durée de l'expérience (figure 21). Par exemple, pour le projet Modelheart, l'utilisateur pourrait renseigner deux groupes : les groupes « Chambre Calorimétrique » et « Domicile ». Pour les chambres calorimétriques, les expériences débuteront à minuit pour une durée de 17 heures. En conditions habituelles de vie (groupe « Domicile »), les expériences commenceront à 8h30 et s'étendront sur une période de 11 jours. Cette astuce évite à l'utilisateur de remplir pour chaque expérience son heure de début et sa durée et optimise donc la saisie des expériences.

**Projet INRA ISIMA**

**Saisie des Groupes**

Groupe 1 : Nom du Groupe :

Paramètres : Heure du début de l'expérience :  heures  minutes

Durée de l'expérience :  heures ▼

Groupe 2 : Nom du Groupe :

Paramètres : Heure du début de l'expérience :  heures  minutes

Durée de l'expérience :  jours ▼

Groupe 3 : Nom du Groupe :

Paramètres : Heure du début de l'expérience :  heures  minutes

Durée de l'expérience :  jours ▼

Groupe 4 : Nom du Groupe :

Paramètres : Heure du début de l'expérience :  heures  minutes

Durée de l'expérience :  jours ▼

*Figure 21 : Fenêtre « Saisie des Groupes »*

Après avoir saisi les paramètres des groupes du projet, l'utilisateur accède à la fenêtre « Etapes » à l'aide du bouton « Créer le projet ». Seul un bouton sur les trois est activé dans cette fenêtre. Il s'agit du bouton « Création d'une expérience » (figure 22). Les deux autres, « Etape 2 » et « Etape 3 », sont désactivés car ces étapes n'ont pas été traitées pendant notre stage.

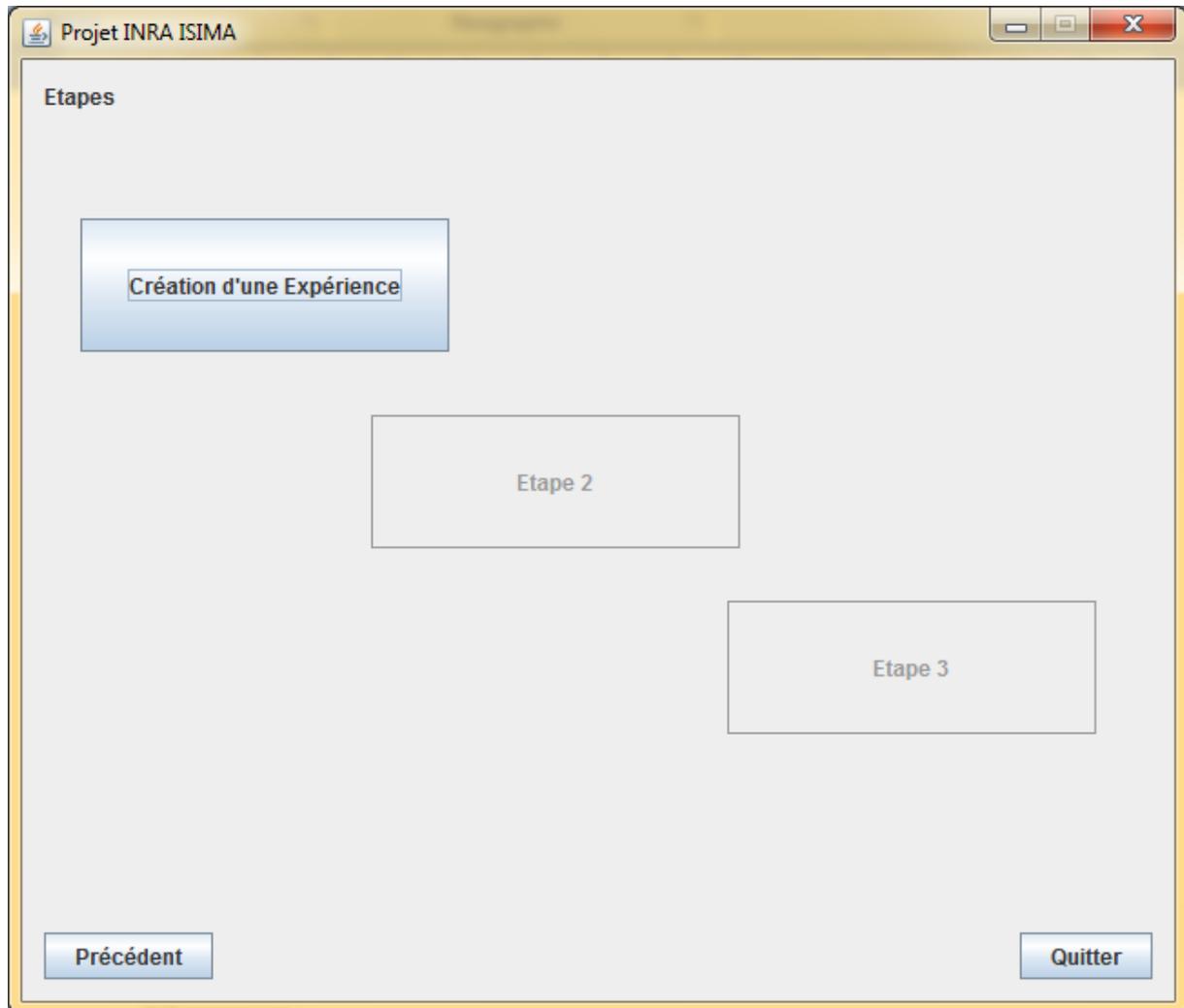


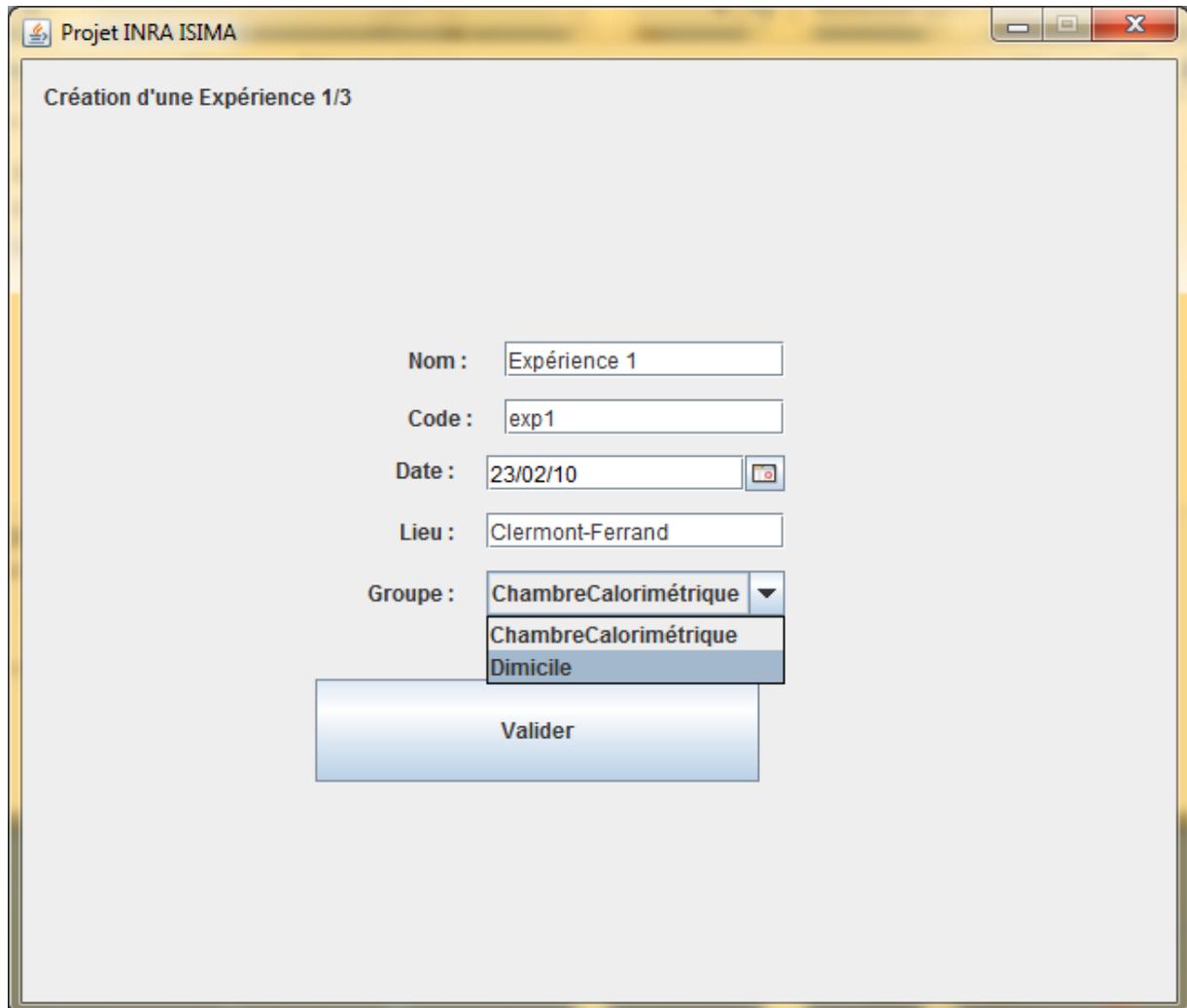
Figure 22 : Fenêtre « Etapes »

C'est donc à partir de cette fenêtre que l'utilisateur a la possibilité d'enregistrer une ou plusieurs expériences, mais seulement les unes après les autres. Cette procédure reste fastidieuse, mais elle n'est réalisée qu'une seule fois pour chaque expérience. Tous les renseignements demandés lors de cette étape sont sauvegardés dans le logiciel et recherchés automatiquement lorsqu'ils sont utiles à son bon fonctionnement.

**TRAITEMENT MATHEMATIQUE DE DONNEES CARDIAQUES :**  
**ETAPE PRELIMINAIRE DE LA PREDICTION DE LA DEPENSE ENERGETIQUE**

Octobre 2009 - Mars 2010

Lors de la création d'une expérience, l'utilisateur saisit son nom, son code, sa date et son lieu. Le code et le lieu sont des champs facultatifs. Il faut aussi associer l'expérience à un groupe (figure 23).



The screenshot shows a software window titled "Projet INRA ISIMA" with a subtitle "Création d'une Expérience 1/3". The window contains a form with the following fields and values:

- Nom : Expérience 1
- Code : exp1
- Date : 23/02/10
- Lieu : Clermont-Ferrand
- Groupe : ChambreCalorimétrique (dropdown menu is open, showing options: ChambreCalorimétrique, Domicile)

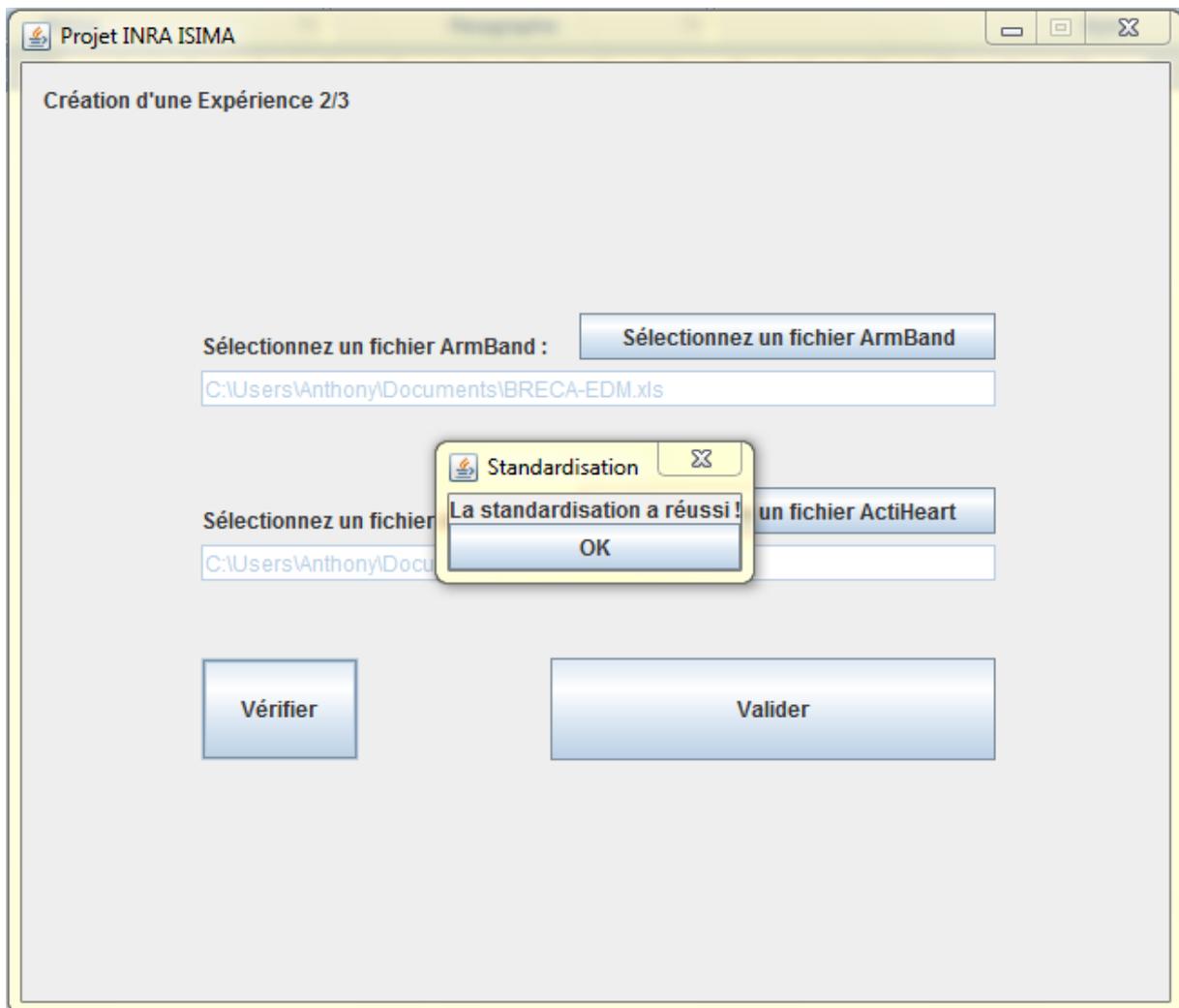
A "Valider" button is located below the form.

Figure 23 : Fenêtre « Création d'une Expérience 1/3 » avec la liste déroulante « Groupe » ouverte

**TRAITEMENT MATHEMATIQUE DE DONNEES CARDIAQUES :  
ETAPE PRELIMINAIRE DE LA PREDICTION DE LA DEPENSE ENERGETIQUE**

Octobre 2009 - Mars 2010

Après avoir choisi le groupe d'expériences et cliqué sur le bouton « Valider », l'utilisateur, par l'intermédiaire de la fenêtre « Création d'une Expérience 2/3 », est invité à sélectionner les fichiers Excel ArmBand et ActiHeart à partir de ceux présents sur son ordinateur puis à appuyer sur le bouton « Vérifier ». Lorsque la boîte de dialogue « La standardisation a réussi ! » apparaît à l'écran, le bouton « Valider » est activé (figure 24).



*Figure 24 : Fenêtre « Création d'une Expérience 2/3 »  
et boîte de dialogue « La standardisation a réussi ! »*

L'utilisateur, en cliquant sur le bouton « Valider », accède à la dernière fenêtre pour finaliser la création de l'expérience (figure 25). Il a la possibilité de saisir des informations supplémentaires à l'aide du champ « Commentaires ». Il lui suffit de cliquer sur le bouton « Valider pour terminer la Création de l'Expérience » pour mettre fin à la procédure.

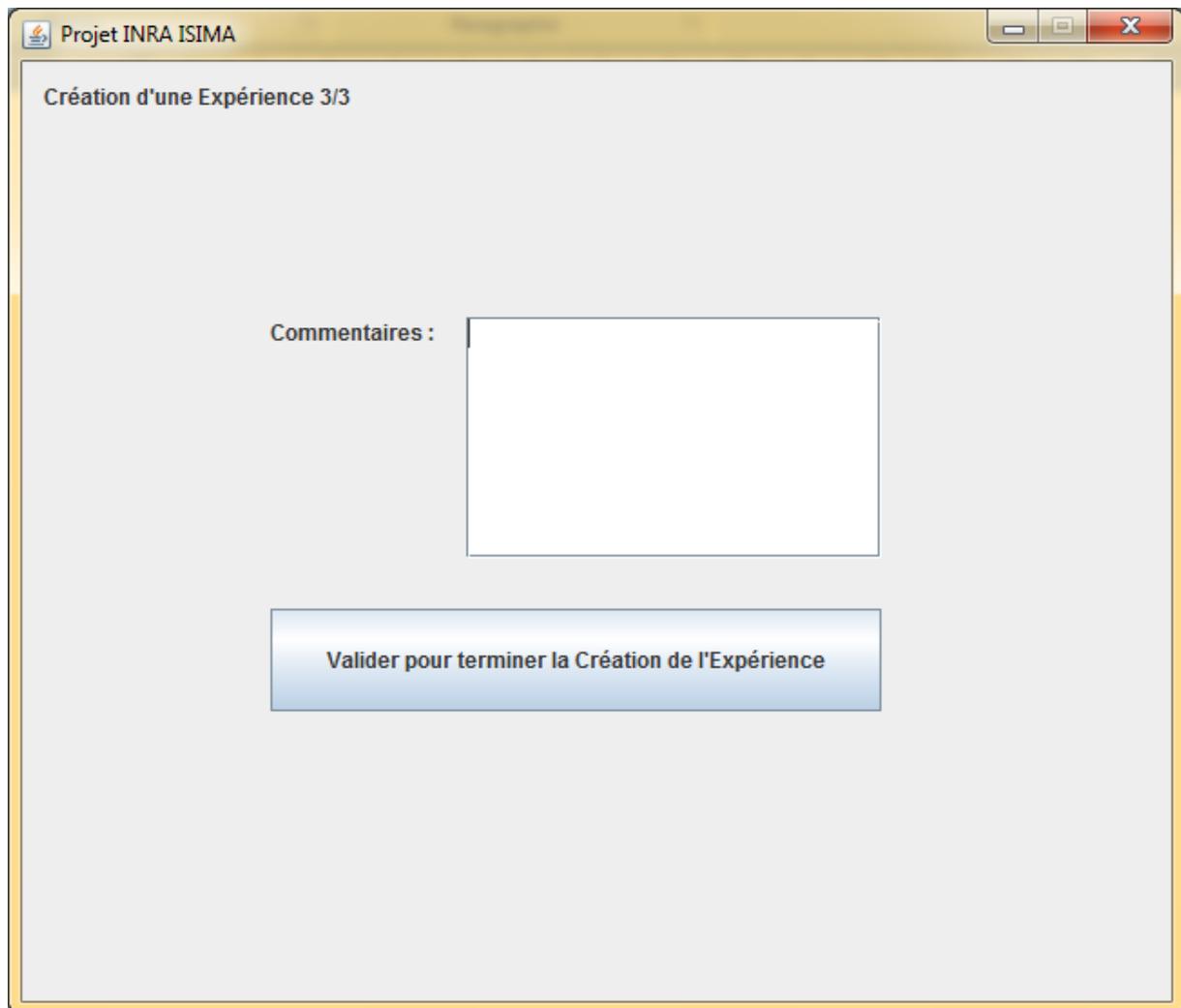


Figure 25 : Fenêtre « Création d'une Expérience »

La fenêtre « Etapes » réapparaît et l'utilisateur a le choix de ressaisir une nouvelle expérience, ou bien de fermer le logiciel à l'aide du bouton « Quitter » ou de la croix rouge en haut à droite de la fenêtre.

### 4.3. Quelques aspects du développement

#### 4.3.1. Arborescence

Une des étapes importantes du développement a été la mise en place de l'arborescence des fichiers et des dossiers constituant le logiciel. La copie, et non leur déplacement, des fichiers Excel à partir de leur emplacement d'origine vers l'emplacement final dans le logiciel était un point très important aux yeux de Madame ROUSSET. Lors de la création d'une expérience, les fichiers Excel ArmBand et ActiHeart sont copiés en lecture seule, c'est-à-dire sans pouvoir être modifiés par l'intermédiaire du logiciel, dans un dossier spécifique. Après standardisation, les fichiers Excel modifiés sont copiés dans un autre dossier dans le logiciel. L'utilisateur peut donc avoir accès aux fichiers bruts mais aussi à ceux qui ont été corrigés. Il faut aussi ajouter que le fait d'enregistrer les fichiers Excel ArmBand et ActiHeart pour chaque expérience ne supprime pas les fichiers sources stockés sur l'ordinateur de l'utilisateur. Les fichiers d'origine ne sont donc pas déplacés.

Plus concrètement, nous allons vous expliquer l'arborescence que nous avons installée en vous détaillant les différents points de code. Prenons par exemple la création d'un projet « Modelheart » qui comporte deux groupes d'expériences « Chambre Calorimétrique » et « Domicile ». Tous deux contiendront plusieurs expériences. Un dossier intitulé du nom du projet, c'est-à-dire « Projet\_Modelheart », est créé dans le dossier « Projets » (situé à la racine du logiciel) lors de la saisie d'un nouveau projet (figure 26).

```
// Dossier intitulé du nom du projet
// jTextField1.getText() récupère le nom du projet saisi dans la fenêtre « Nouveau Projet »
File dossier = new File("Projets/Projet_" + jTextField1.getText());
// Création du dossier
// Si test est égal à faux, alors la création ne s'est pas déroulée correctement
// (à cause d'un même nom de dossier par exemple)
boolean test = dossier.mkdirs();
```

Figure 26 : Extrait du code pour créer un dossier

Dans le nouveau dossier « Projet\_Modelheart », deux dossiers, « Categories » et « Groupes », et un fichier de configuration sont créés automatiquement (figure 27). Le dossier « Categories » n'est pas utilisé par la version que nous avons conçue mais servira au développement futur du logiciel. Le dossier « Groupes » regroupe les différents groupes d'expériences. Le fichier de configuration « config.txt » sauvegarde tous les paramètres nécessaires au projet (figure 28).

```
// Pour créer un fichier config.txt et y insérer des données
FileWriter writer = null;
// Première chaîne à insérer : "Nom du projet : ModelHeart"
// L'instruction System.getProperty("line.separator") désigne un saut de ligne
String texte1 = "Nom du projet : " + jTextField1.getText() +
System.getProperty("line.separator") + System.getProperty("line.separator");
// Deuxième chaîne à insérer : "Nombre de groupes : 2"
String texte2 = "Nombre de groupes : " + jTextField2.getText() +
System.getProperty("line.separator");
try {
    // Création du fichier config.txt dans l'arborescence Projets/Projet_ModelHeart/
    writer = new FileWriter("Projets/Projet_" + jTextField1.getText() + "/config.txt", true);
    // Insertion de la première chaîne
    writer.write(texte1,0,texte1.length());
    // Insertion de la deuxième chaîne
    writer.write(texte2,0,texte2.length());
    [...]
    writer.close();
}
```

Figure 27 : Extrait du code pour créer un fichier

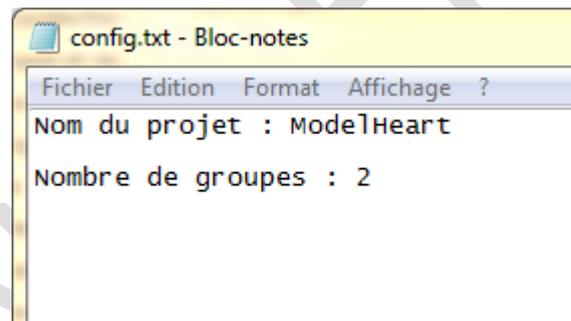


Figure 28 : Fichier de configuration « config.txt »

Lorsque l'utilisateur renseigne les paramètres des deux groupes d'expériences « Chambre Calorimétrique » et « Domicile », le fichier de configuration est mis à jour en sauvant les caractéristiques des expériences (figure 29). Deux dossiers « Groupe\_ChambreCalorimétrique » et « Groupe\_Domicile » sont, dans le même temps, créés.

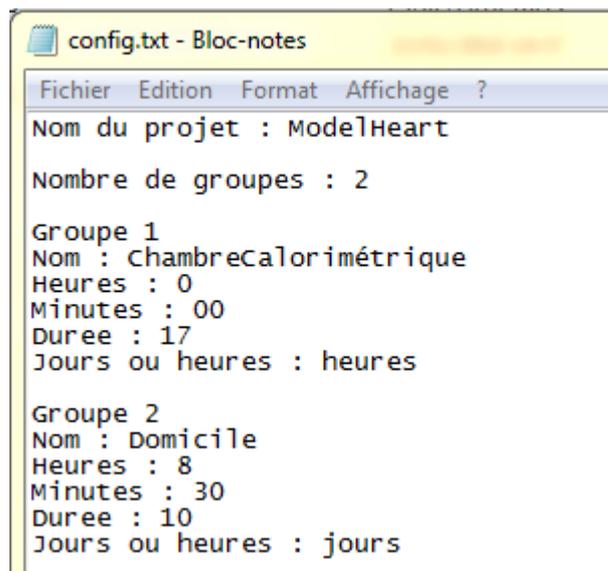


Figure 29 : Fichier de configuration « config.txt » mis à jour

La phase de création du projet est donc terminée, et celle de l'enregistrement des expériences débute lorsque l'utilisateur clique sur le bouton « Création d'une Expérience » à partir de la fenêtre « Etapes ». Un dossier intitulé du nom de l'expérience est créé dans le dossier du groupe d'expériences à qui il est associé. Il y a un dossier pour chaque expérience enregistrée. Par exemple, pour une expérience réalisée en Chambre Calorimétrique, un dossier « Experience\_Exp1 » est créé dans le dossier « Groupe\_ChambreCalorimétrique ». Il contiendra deux sous-dossiers : « DonneesBrutes » pour la sauvegarde des fichiers Excel d'origine (figure 30) et « DonneesTraitees » pour la sauvegarde des fichiers Excel modifiés.

```
// Copie du fichier ArmBand
// Fichier source
FileChannel in = null;
// Fichier destination
FileChannel out = null;
try {
    // Initialisation du fichier source
    in = new FileInputStream(cheminFichierArmBand).getChannel();
    // Initialisation du fichier destination
    out = new FileOutputStream(cheminExperience + "/DonneesBrutes/" +
nomArmBand).getChannel();

    // Copie depuis le in vers le out
    in.transferTo(0, in.size(), out);
    in.close();
    out.close();
    [...]
}
[...]
```

Figure 30 : Extrait du code pour sauvegarder le fichier ArmBand d'origine

Dans chaque dossier de groupes d'expériences, après l'ajout d'une expérience, un fichier de configuration « configExperience.txt » est créé puis mis à jour afin de sauvegarder la liste des noms des expériences avec leur date de réalisation (figure 31).

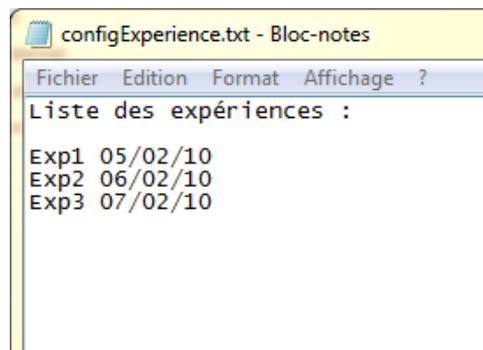


Figure 31 : Fichier de configuration « configExperience.txt »

La procédure d'enregistrement d'une expérience ne peut pas être interrompue, c'est-à-dire que l'utilisateur n'a pas la possibilité de revenir en arrière (par exemple pour corriger une erreur) ou de fermer le logiciel. C'est bien sûr contraignant et cela fera l'objet d'une amélioration du logiciel. Nous avons interdit cette manipulation à l'utilisateur car nous n'avons pas eu le temps de gérer correctement la modification ou la suppression d'un enregistrement d'une expérience. L'objectif étant de garder un logiciel « propre » pour éviter d'éventuelles erreurs de fonctionnement. Concernant les projets, la suppression n'a pas été développée et seule la modification avant validation de la création est possible. Pour ce faire, l'utilisateur a la possibilité de revenir à l'étape précédente (« Nouveau Projet ») en utilisant le bouton « Précédent » lors de la saisie des groupes. Le dossier du projet ainsi que ses sous-dossiers et son fichier de configuration seront ainsi supprimés (figure 32).

```
// Chemin du dossier du projet à supprimer
File path = new File(cheminProjet);
// Si le chemin existe
if (path.exists() ) {
    // Lister les fichiers et les dossiers du projet
    File[] files = path.listFiles();
    for (int i=0; i<files.length; i++) {
        // Si c'est un dossier à supprimer
        if (files[i].isDirectory()) {
            resultat &= deleteDirectory(files[i]);
        }

        // Sinon c'est un fichier à supprimer
        else {
            resultat &= files[i].delete();
        }
    }
}
// Suppression définitive du chemin du projet
resultat &= path.delete();
```

Figure 32 : Extrait du code pour supprimer l'arborescence (dossiers et fichier) d'un projet

Le schéma suivant résume l'arborescence mise en place dans le logiciel (figure 33).

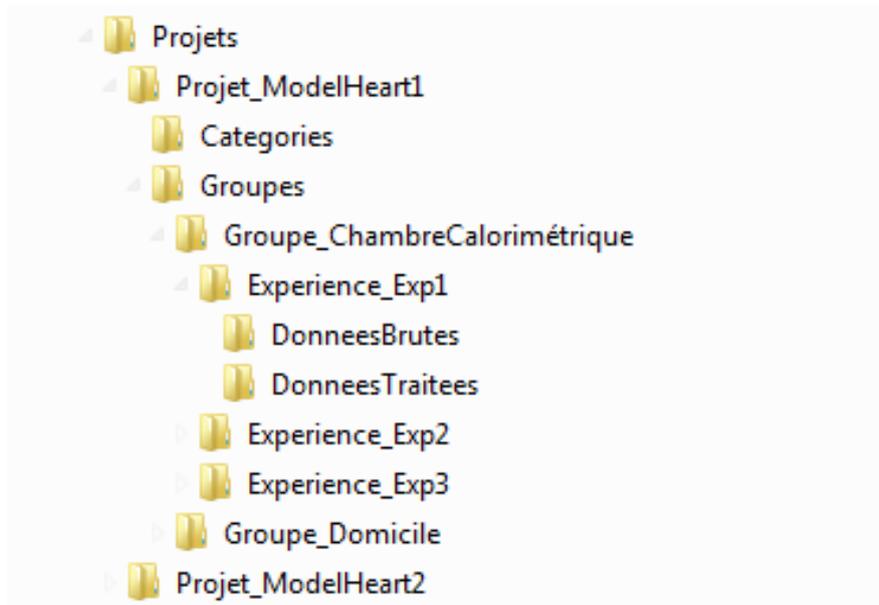


Figure 33 : Arborescence du logiciel

### 4.3.2. Saisie des Groupes

Un autre aspect du développement, qui a été difficile à mettre en œuvre, est la notion de groupes d'expériences. Comme nous l'avons expliquée précédemment, la saisie de groupes d'expériences est un concept essentiel dans la création d'un projet. A l'aide de la fenêtre « Nouveau projet », l'utilisateur renseigne le nombre de groupes qui seront utilisés pendant le projet. Le nombre est limité à quatre pour éviter de surcharger la fenêtre « Saisie des Groupes » (figure 34).

The screenshot shows a window titled "Projet INRA ISIMA" with a sub-header "Saisie des Groupes". It contains four identical sections for group entry:

- Groupe 1 :** Nom du Groupe: [text box]  
Paramètres: Heure du début de l'expérience: [ ] heures [ ] minutes  
Durée de l'expérience: [ ] jours [dropdown arrow]
- Groupe 2 :** Nom du Groupe: [text box]  
Paramètres: Heure du début de l'expérience: [ ] heures [ ] minutes  
Durée de l'expérience: [ ] jours [dropdown arrow]
- Groupe 3 :** Nom du Groupe: [text box]  
Paramètres: Heure du début de l'expérience: [ ] heures [ ] minutes  
Durée de l'expérience: [ ] jours [dropdown arrow]
- Groupe 4 :** Nom du Groupe: [text box]  
Paramètres: Heure du début de l'expérience: [ ] heures [ ] minutes  
Durée de l'expérience: [ ] jours [dropdown arrow]

Buttons: "Créer le projet" (bottom right), "Précédent" (bottom left).

Figure 34 : Fenêtre « Saisie des Groupes » avec un groupe activé

**TRAITEMENT MATHEMATIQUE DE DONNEES CARDIAQUES :**  
**ETAPE PRELIMINAIRE DE LA PREDICTION DE LA DEPENSE ENERGETIQUE**  
Octobre 2009 - Mars 2010

Selon le nombre de groupes choisi par l'utilisateur, les informations nécessaires sont activées sur la fenêtre « Saisie des Groupes » (figure 35). Les autres sont alors désactivées.

Nom du Projet :

Nombre de groupes :

**Suivant**

**Projet INRA ISIMA**

**Saisie des Groupes**

Groupe 1 : Nom du Groupe :   
Paramètres : Heure du début de l'expérience :  heures  minutes  
Durée de l'expérience :  jours ▼

Groupe 2 : Nom du Groupe :   
Paramètres : Heure du début de l'expérience :  heures  minutes  
Durée de l'expérience :  heures ▼

Groupe 3 : Nom du Groupe :   
Paramètres : Heure du début de l'expérience :  heures  minutes  
Durée de l'expérience :  jours ▼

Groupe 4 : Nom du Groupe :   
Paramètres : Heure du début de l'expérience :  heures  minutes  
Durée de l'expérience :  jours ▼

**Créer le projet**

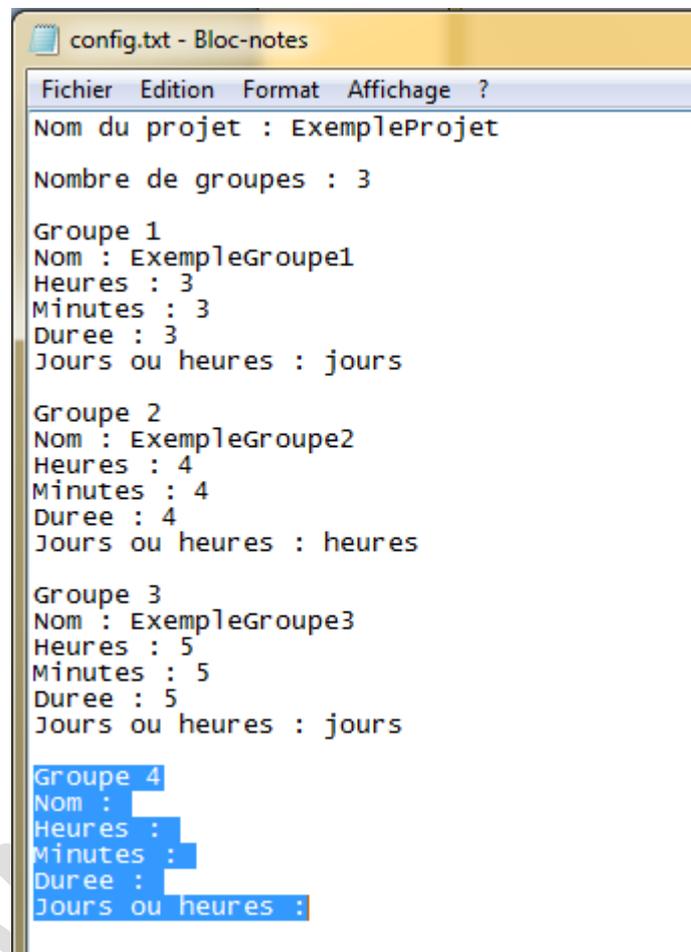
**Précédent**

Figure 35 : Fenêtre « Saisie des Groupes » avec trois groupes activés

**TRAITEMENT MATHEMATIQUE DE DONNEES CARDIAQUES :  
ETAPE PRELIMINAIRE DE LA PREDICTION DE LA DEPENSE ENERGETIQUE**

Octobre 2009 - Mars 2010

Le principe est simple à comprendre mais plus compliqué à mettre en place. Le logiciel utilise le fichier de configuration « config.txt » pour sauvegarder les informations des quatre groupes. Dans notre exemple, le groupe 4 est désactivé, il ne va donc pas être utilisé, mais il sera présent dans le fichier de configuration avec des valeurs nulles (figure 36).



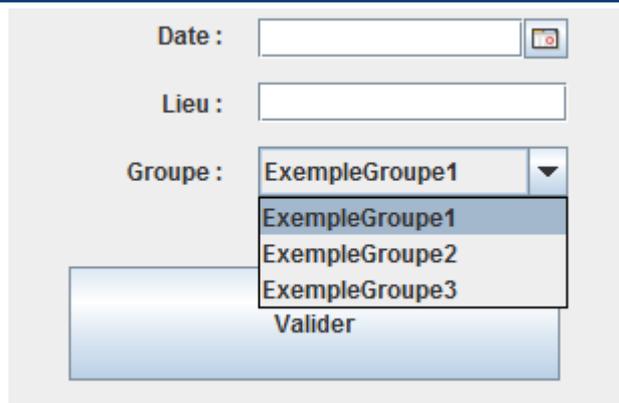
```
config.txt - Bloc-notes
Fichier  Edition  Format  Affichage  ?
Nom du projet : ExempleProjet
Nombre de groupes : 3
Groupe 1
Nom : ExempleGroupe1
Heures : 3
Minutes : 3
Duree : 3
Jours ou heures : jours
Groupe 2
Nom : ExempleGroupe2
Heures : 4
Minutes : 4
Duree : 4
Jours ou heures : heures
Groupe 3
Nom : ExempleGroupe3
Heures : 5
Minutes : 5
Duree : 5
Jours ou heures : jours
Groupe 4
Nom : 
Heures : 
Minutes : 
Duree : 
Jours ou heures :
```

Figure 36 : Fichier de configuration « config.txt » avec le groupe 4 inutilisé (valeurs nulles)

Ce fichier de configuration servira aussi lors de la première étape de chaque création d'une expérience. Chacune est associée à son groupe d'expériences, c'est pourquoi nous avons développé une liste déroulante qui se met à jour automatiquement à l'aide de « config.txt » et qui énumère les noms des groupes (figure 37).

**TRAITEMENT MATHEMATIQUE DE DONNEES CARDIAQUES :  
ETAPE PRELIMINAIRE DE LA PREDICTION DE LA DEPENSE ENERGETIQUE**

Octobre 2009 - Mars 2010



The screenshot shows a graphical user interface with the following elements:

- A 'Date' field with a calendar icon.
- A 'Lieu' (Location) text input field.
- A 'Groupe' dropdown menu currently showing 'ExempleGroupe1'. The dropdown list is open, showing three options: 'ExempleGroupe1', 'ExempleGroupe2', and 'ExempleGroupe3'.
- A 'Valider' (Validate) button below the dropdown.

Figure 37 : Liste déroulante des groupes d'expériences

En respectant l'arborescence, un dossier du nom de l'expérience est créé dans le dossier du groupe qui lui correspond. Cette technique de la liste déroulante est détaillée à la figure 38.

```
// Création de la liste déroulante
private JComboBox1;
jComboBox1 = new JComboBox();
// Lecture du fichier de configuration config.txt
BufferedReader buff = new BufferedReader(new FileReader([...]config.txt));
try {
    // Ligne du fichier
    String line;
    // Pour compter les lignes du fichier
    int compteur = 0;
    // Lecture du fichier ligne par ligne
    // Cette boucle se termine quand la méthode retourne la valeur null
    while ((line = buff.readLine()) != null) {
        compteur ++;
        // Pour trouver la bonne ligne, c'est-à-dire les lignes comportant le nom
        // des groupes d'expériences
        if ((compteur == 6 || compteur == 13 || compteur == 20 || compteur == 27))
        {
            // Dans le fichier, chaque ligne de nom de groupe se compose de
            // Nom : leNomDuGroupe
            // Il faut donc couper cette ligne en deux à partir du séparateur « : »
            nomGroupe = line.split(":");
            // Il faut prendre la deuxième partie de la ligne (soit nomGroupe[1])
            // La première étant nomGroupe[0]
            nom = nomGroupe[1];
            // Si le nom du groupe n'est pas vide
            // Cela signifie que le groupe est utilisé (actif)
            if (!nom.equals(""))
                // Alors ce nom est ajouté dans la liste déroulante
                jComboBox1.addItem(nom);
        }
    }
}
[...]
```

Figure 38 : Extrait de code pour créer la liste déroulante des noms des groupes d'expériences

### 4.3.3. Parcours des fichiers

L'étape d'enregistrement des fichiers Excel ArmBand et ActiHeart étant longue et fastidieuse, nous avons mis en place deux astuces pour gagner en rapidité de saisie. Le problème était que le chemin utilisé pour chercher les fichiers dans l'ordinateur n'était pas enregistré, et l'utilisateur était à chaque fois obligé de parcourir ses dossiers personnels.

La première solution a été simplement d'ajouter une ligne dans le code des deux éléments « Parcourir » du fichier JAVA (figure 39).

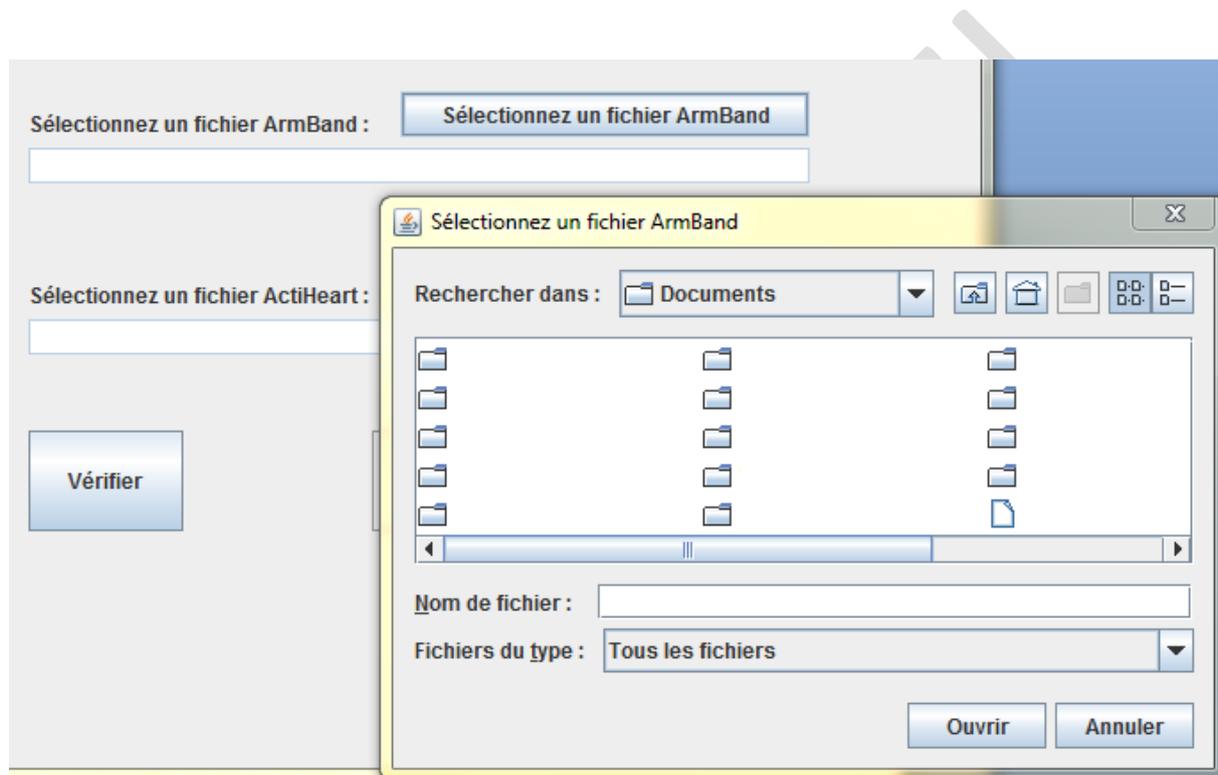


Figure 39 : Une des deux fenêtres « Parcourir » du logiciel

Le code rajouté est le suivant : `« jFileChooserParcourir.rescanCurrentDirectory(); »`. « jFileChooserParcourir » désigne l'élément « Parcourir », c'est-à-dire la fenêtre « Parcourir » pour rechercher des fichiers sur l'ordinateur. La méthode « `rescanCurrentDirectory()` » va permettre de mémoriser le dernier dossier visité par l'utilisateur lors de la recherche de son fichier à partir de l'élément « Parcourir ». Par exemple, l'utilisateur va tout d'abord parcourir ses fichiers personnels pour trouver le fichier ArmBand à l'aide de la première fenêtre « Parcourir » (en appuyant sur le bouton « Sélectionnez un fichier ArmBand »). Et dans un second temps, il va rechercher le fichier ActiHeart associé, à l'aide de la deuxième fenêtre « Parcourir », en cliquant sur le bouton « Sélectionner un fichier ActiHeart ». L'astuce est de créer un et un seul élément commun aux deux fenêtres « Parcourir » pour avoir la mémorisation du dernier répertoire visité. Comme les deux fichiers ArmBand et ActiHeart sont le plus souvent dans le même dossier, l'utilisateur n'aura pas besoin de faire des manipulations supplémentaires pour rechercher le fichier Excel ActiHeart (il aura déjà visité le répertoire pour sélectionner le fichier ArmBand).

Cette première solution n'est pas totalement satisfaisante parce qu'à chaque fois que la création d'une expérience se termine, l'élément « Parcourir » est détruit (au sens JAVA). Il n'existe plus et par conséquent, la mémorisation du dernier répertoire est effacée. Nous avons donc mis en œuvre une deuxième solution complémentaire. A la première création d'une expérience d'un projet, le chemin du dernier répertoire visité et validé lors de la sélection du fichier ArmBand (c'est-à-dire que ce fichier a été choisi dans ce dernier répertoire) est sauvegardé dans un fichier « configChemin.txt » à la racine du projet. Ce fichier sera dorénavant lu à chaque fois que l'utilisateur cliquera sur le bouton « Sélectionnez un fichier ArmBand » ou sur « Sélectionnez un fichier ActiHeart » et permettra d'initialiser la fenêtre « Parcourir » avec le chemin sauvegardé. Cette astuce ne peut être efficace que si la plupart des fichiers ArmBand et ActiHeart sont placés dans le même dossier. Elle peut être très utile et permettre un gain de temps considérable dans la phase de création des expériences. Cette technique est détaillée à la figure 40.

```
// Initialisation du fichier configChemin.txt
File fichierConfigChemin = new File(cheminProjet + "/configChemin.txt");
// Contient le chemin du dernier répertoire visité
String stringConfigChemin = null;
// Si le fichier configChemin.txt existe
// Nous sommes dans le cas où il y a déjà eu au moins une création d'une expérience
if (fichierConfigChemin.exists())
{
    // Lecture du fichier configChemin.txt
    // pour rechercher le chemin du dernier répertoire visité
    [...]
    // Initialisation de cette chaîne avec le chemin du dernier répertoire visité
    stringConfigChemin = [...];
    // Initialisation de l'élément « Parcourir » avec le chemin sauvegardé
    jFileChooserParcourir = new JFileChooser(stringConfigChemin);
}
// Sinon si le fichier configChemin.txt n'existe pas
// Nous sommes dans le cas où c'est la première création d'une expérience
else
{
    // Utilisation « normale » de l'élément « Parcourir »
    jFileChooserParcourir = new JFileChooser();
    [...]
    // Lorsque le fichier ArmBand a été sélectionné, le logiciel va pouvoir
    // créer le fichier configChemin.txt en sauvegardant le dernier répertoire visité
    // Ce fichier sera maintenant utilisé à chaque sélection d'un fichier Excel
}
[...]
```

Figure 40 : Extrait du code pour sélectionner plus rapidement les fichiers Excel

## 5. Validation et prolongements

### 5.1. Tests et Validation

Selon le cahier des charges, le logiciel doit posséder une interface simple et intuitive. La mise en main du logiciel doit être aisée et rapide car les utilisateurs, non informaticiens, souhaitent pouvoir travailler efficacement avec cet outil de travail. Nous avons donc privilégié la mise en place de champs à renseigner simples et compréhensibles par tous.

Pour tous les champs à saisir, des tests informatiques sont codés pour éviter les erreurs et ainsi empêcher un fonctionnement anormal du logiciel. Ils permettent aussi une uniformisation des données du logiciel. Par exemple, lors de la création d'un nouveau projet, dans la fenêtre « Nouveau projet », un test vérifie si les champs « Nom du Projet » et « Nombre de Groupes » sont bien renseignés. « Nom du Projet » et « Nombre de Groupes » ne doivent pas être vides, et « Nombre de Groupes » doit contenir un entier positif compris entre 1 et 4 (un projet peut contenir un à quatre groupes d'expériences) (figure 41).

```
// jTextField1.getText() correspond au nom du projet
// jTextField2.getText() correspond au nombre de groupes d'expériences
// Test pour vérifier si les deux champs ne sont pas vides
if (jTextField1.getText().equals("") || jTextField2.getText().equals(""))
{
    // Affichage d'une boîte de dialogue avertissant un ou plusieurs champs vides
}
else
{
    // Test pour vérifier si « Nombre de Groupes » est un entier
    if (!IsValidInt(jTextField2.getText()))
    {
        // Affichage d'une boîte de dialogue avertissant que ce n'est pas un entier
    }
    else
    {
        // Récupération de la donnée au format String...
        c1 = jTextField2.getText();
        // Conversion en int
        e1 = Integer.parseInt(c1);
        // Test pour vérifier si « Nombre de Groupes » est compris entre 1 et 4
        if (e1 <= 0 || e1 > 4)
        {
            // Affichage d'un avertissement car non compris entre 1 et 4
        }
        else
        {
            // Tous les critères sont réunis, il peut accéder à l'étape suivante
            [...]
        }
    }
}
```

Figure 41 : Extrait du code pour tester la bonne saisie du nom du projet et du nombre de groupes

Si ces critères ne sont pas respectés, l'utilisateur est averti avec l'affichage d'une boîte de dialogue d'avertissement et il ne peut pas accéder à l'étape suivante (figure 42).



Figure 42 : Boîte de dialogue d'avertissement

D'autres boîtes de dialogue, dites d'information, sont présentes dans le logiciel, notamment pour annoncer à l'utilisateur le bon déroulement de la standardisation des fichiers Excel ArmBand et ActiHeart (figure 43).

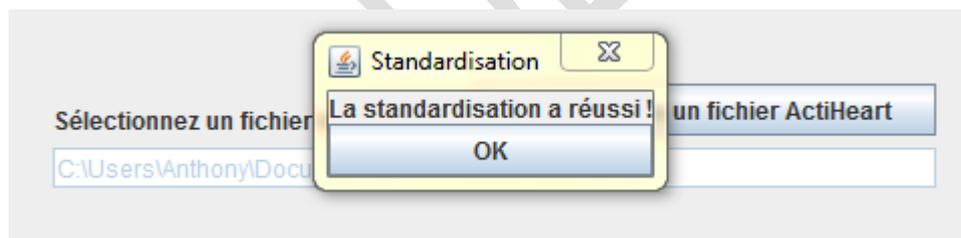


Figure 43 : Boîte de dialogue d'information

La dernière boîte de dialogue, appelée boîte de dialogue de confirmation, s'affiche lorsque l'utilisateur souhaite fermer le logiciel soit à l'aide du bouton « Quitter » situé en bas à droite, soit en exécutant la croix rouge en haut à droite de chaque fenêtre (figure 44).

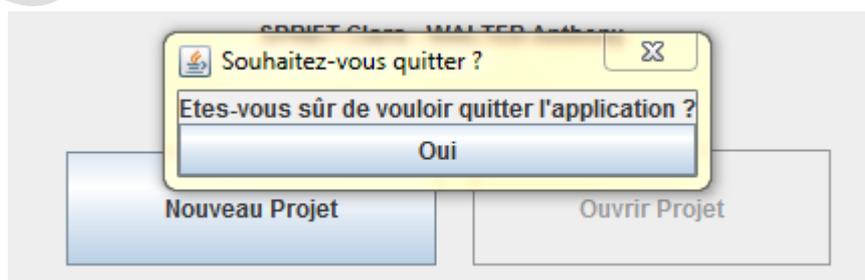


Figure 44 : Boîte de dialogue de confirmation

**TRAITEMENT MATHÉMATIQUE DE DONNÉES CARDIAQUES :**  
**ÉTAPE PRÉLIMINAIRE DE LA PRÉDICTION DE LA DÉPENSE ÉNERGÉTIQUE**

Octobre 2009 - Mars 2010

Pour aider les utilisateurs à remplir plus facilement, plus rapidement et sans erreur la date de réalisation de chaque expérience, un calendrier a été mis en place. En appuyant sur l'icône « calendrier », l'utilisateur n'a plus qu'à sélectionner le jour, ainsi que le mois et l'année à partir des listes déroulantes. La date est alors remplie automatiquement au bon format dans le champ qui lui est réservé (figure 45).

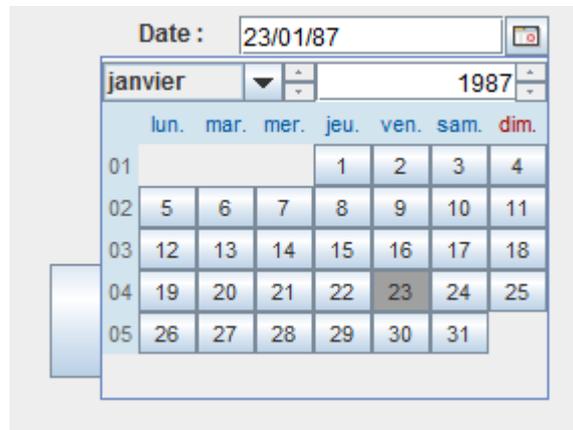


Figure 45 : Calendrier pour aider l'utilisateur à saisir la date de réalisation d'une expérience

Le logiciel a aussi été testé plusieurs fois en effectuant tous les scénarios de fonctionnement et en vérifiant les algorithmes mis en œuvre. Les résultats issus de la standardisation ont été comparés avec les données de l'électrocardiogramme. Ils sont très concluants (figure 46).

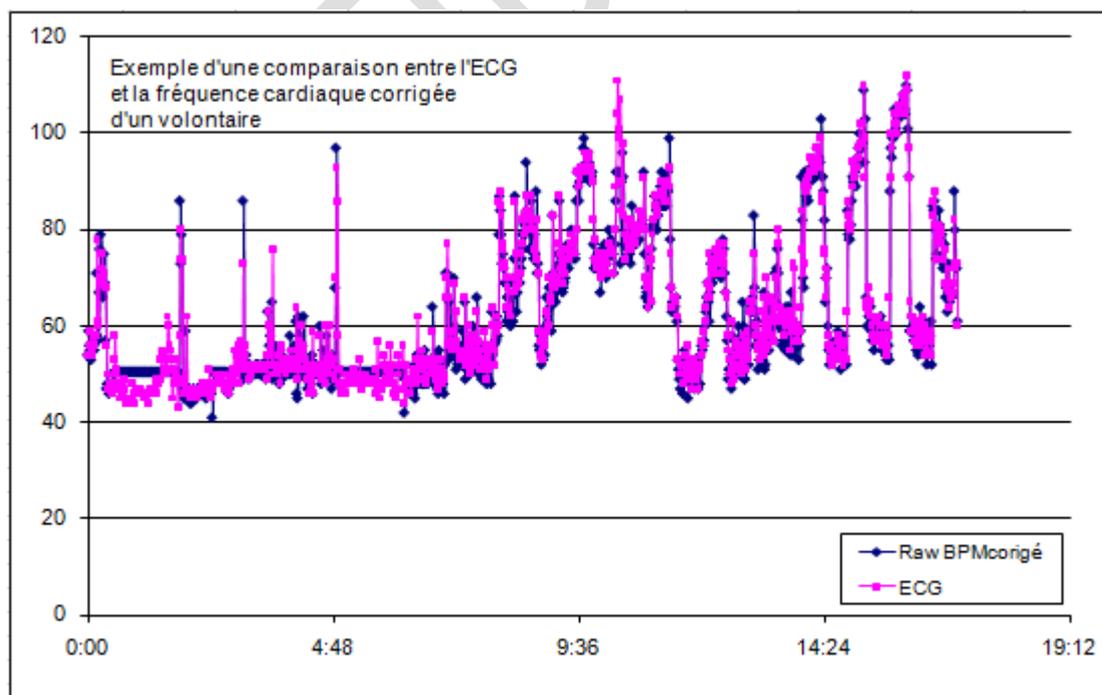


Figure 46 : Exemple d'une comparaison entre l'ECG et la fréquence cardiaque corrigée

A chaque modification du logiciel, même mineure, nous procédions à de nouveaux tests à l'aide des fichiers Excel ArmBand et ActiHeart afin de la valider. Avant chaque réunion, nous faisons en sorte de répertorier tous les problèmes rencontrés et non résolus afin de les présenter à Madame ROUSSET et à Messieurs LACOMME et FLEURY pour essayer de trouver ensemble une solution. Nous avons, par exemple, résolu une difficulté lors des dernières entrevues hebdomadaires. Le code du logiciel était compilé sur l'ordinateur d'Anthony, et testé et validé sur celui de Clara. Pendant plusieurs semaines, le logiciel ne se comportait pas de la même manière sur l'ordinateur de Clara et sur celui d'Anthony. Madame ROUSSET avait aussi rencontré les mêmes problèmes à partir de son ordinateur. Les standardisations se déroulaient correctement sur celui d'Anthony, mais bloquaient sur ceux de Clara et de Madame ROUSSET. Ces erreurs nous contrariaient grandement parce que le langage JAVA est un langage multiplateforme et il devrait fonctionner sur tous les ordinateurs et sur tous les systèmes d'exploitation (Madame ROUSSET travaille sur le système Windows XP, Clara développe sous Windows Vista et Anthony sous Windows Seven). Monsieur LACOMME nous a donc aidés à trouver une solution et il s'est avéré qu'il y avait un problème de compatibilité entre nos environnements JAVA (versions plus anciennes). Pour respecter la portabilité du logiciel, l'ordinateur doit posséder la version 6 (ou une version supérieure) du JRE (JAVA Runtime Environment) qui désigne un ensemble d'outils permettant l'exécution de programmes JAVA sur toutes les plateformes. Le code du logiciel a été compilé à partir de la version 1.6.0.18 du JDK (JAVA Development Kit). Nous avons donc mis à jour l'environnement JAVA sur ces deux ordinateurs et la standardisation des fichiers Excel ArmBand et ActiHeart n'a plus posé de problème. Nous avons essayé tout au long de notre stage de communiquer régulièrement l'avancement de nos travaux à Madame ROUSSET en lui envoyant une nouvelle version du logiciel pour qu'elle puisse avoir le temps de la tester et de nous apporter ses suggestions et ses remarques avant le déroulement de la réunion.

## 5.2. Prolongements

Pendant notre projet, nous avons rédigé un cahier des charges très complet et avons commencé à répondre aux nombreux problèmes posés. Notre travail s'est arrêté à la deuxième partie de la première étape : le traitement des valeurs manquantes des fichiers Excel ArmBand et ActiHeart. Nous avons construit les bases du projet en développant intégralement un logiciel qui permet actuellement de standardiser correctement et très rapidement les deux fichiers Excel les plus difficiles à traiter : ArmBand et ActiHeart ainsi que d'éliminer leurs valeurs manquantes.

Notre travail va être repris et continué par un stagiaire à partir du mois d'avril. En respectant le cahier des charges, il continuera le développement du logiciel en commençant par réfléchir à l'étape de lissage des courbes des fichiers Excel. Un document sera rédigé lors de notre dernière réunion afin de répertorier les nombreuses tâches à réaliser.

Outre le fait de lisser les courbes, d'étalonner les courbes d'Oxygène consommé et de CO<sub>2</sub> produit, de calculer la dépense énergétique, et de calculer le coût cardiaque relatif, l'étudiant aura aussi d'autres objectifs. Le premier étant de réaliser l'étape 2, c'est-à-dire la génération et la visualisation de courbes à partir des données traitées, mais aussi effectuer des calculs sur les courbes (dérivées, minimum, maximum, moyenne, ...) selon une catégorie d'expériences donnée. Le dernier objectif, qui est en réalité l'objectif principal de ce projet, est de modéliser la dépense énergétique des volontaires en fonction de leur fréquence cardiaque, de leur paramètre de mouvement, de leur température corporelle, du flux de chaleur dissipé par leur peau et la conductance de leur peau, etc. (étape 3).

Le stagiaire devra aussi améliorer le fonctionnement du logiciel en lui rajoutant de nouvelles fonctions. Par exemple, la phase d'ouverture d'un projet, et peut-être aussi celle de modification d'un projet, seraient à développer. Il faudrait également travailler sur la modification et la fermeture d'une expérience qui n'a pas été terminée.

## CONCLUSION

Au moment de la rédaction de ce rapport, les derniers tests ont été effectués. Seul un problème de dates lié au jour sur les fichiers ArmBand persiste, mais le logiciel peut tout de même être utilisé. Il peut aider Madame Sylvie ROUSSET à ordonner ses fichiers, et surtout à les standardiser et à remplacer les valeurs manquantes. Pour ces dernières, les tests ont été validés grâce à la redondance des capteurs pour les fichiers en chambre calorimétrique. Une opération comme celle-ci se comptait en heures avant le développement du logiciel. Elle est maintenant réduite à quelques secondes.

Ce sont donc les deux premiers traitements envisagés par le cahier des charges qui ont été réalisés. L'analyse nous a demandé beaucoup de temps, de réunions et de travail, et c'est ce point tout particulièrement qui a été intéressant.

Pour la première fois, nous avons pu commencer un projet depuis le début, c'est-à-dire depuis l'identification des besoins. Ce projet a été complet puisque à partir de notre analyse et de la rédaction d'un cahier des charges, nous avons imaginé une conception et passé à la réalisation.

## GLOSSAIRE

### Dépense énergétique :

Quantité d'énergie consommée par l'organisme au cours d'une unité de temps pour réaliser une action ou remplir une fonction déterminée.

### ECG :

Electrocardiogramme.

### Garbage Collector :

Récupération d'emplacements en mémoire qui ne sont plus utilisés par le système, mise à jour des tables d'allocation de mémoire.

### Interpolation linéaire :

Soit  $f$  une fonction définie sur  $\mathbb{R}$ ,  $[a; b]$  un intervalle de  $\mathbb{R}$  et  $c$  un nombre réel. Quand il n'est pas possible de calculer l'image de  $c$  par  $f$ , on utilise une interpolation linéaire, cela consiste à remplacer  $f(c)$  par  $g(c)$  où  $g$  est la fonction affine telle que :  $g(a) = f(a)$  et  $g(b) = f(b)$ .

### Isotopes :

Chacun des différents types d'atomes d'un même élément, différant par leur nombre de neutrons mais ayant le même nombre de protons et d'électrons, et possédant donc les mêmes propriétés chimiques.

### Langage JAVA :

C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut-être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (jsp).

## BIBLIOGRAPHIE

[JEXCELAPI] <http://www.andykhan.com/jexcelapi/tutorial.html> : le tutorial de la librairie JExcelApi.

[INRA] <http://www.inra.fr/> : le site web de l'INRA.

[REGRESSION] <http://www.bibmath.net/dico/index.php3?action=affiche&quoi=/r/reglin.html> : informations sur la régression linéaire.

[NETBEANS] <http://netbeans.developpez.com/cours/> : le tutorial de NetBeans.

CONFIDENTIEL