

Projet BÉA

Bracelet Électronique pour l'Autonomie

Ce rapport de stage présente le travail effectué côté technique lors du stage à Atos Origin Intégration de mars à août 2009 dans le cadre de la double formation (INSA-IAE). Il comprend notamment les comptes rendus de l'ensemble des grammes développés dans le cadre du projet BEA.

**Pascal MIETLICKI – Master 2 management de l'innovation –
5^{ème} année génie informatique INSA**

29/08/2009



Table des matières

Table des matières	3
Executive summary	4
Introduction	5
A. Étude sur les trajets GPS	7
A.1. Le format NMEA	8
A.2. Compression des trames.....	9
A.3. Interface graphique en VBA.....	15
B. Le programme sur PDA	18
B.1. Les modules.....	19
B.2. GPS Intermediate Driver.....	19
B.3. GPS Tracker	22
C. Analyse des résultats.....	33
C.1. Outil de visualisation et analyse des trajets	34
Analyse et détermination du seuil « idéal ».....	37
D. Bilan	Error! Bookmark not defined.
D.1. Enseignements de l'INSA et IAE.....	Error! Bookmark not defined.
D.2. Bilan personnel	Error! Bookmark not defined.
Glossaire et mots clefs	40
Références bibliographiques	41
Références webographiques	42
Illustrations	43
Projet BÉA - Annexes	45

Executive summary

Mission

Analyse du marché de la téléassistance en Europe et principalement en France pour le développement d'un **Bracelet Électronique pour l'Autonomie** (BEA).

Étude, conception et développement des programmes préalable de ce projet de Recherche et Développement subventionné par la DGE (Direction Générale des Entreprises).

Au niveau technique, ma problématique principale fut d'optimiser les trajets GPS de manière à renvoyer les points les plus significatifs d'un parcours piéton.

Utilisateurs cibles

La solution s'adresse à **toutes les personnes dépendantes** qui puissent être confronté à des situations critiques nécessitant une aide médicale ou une prise en charge spécifique.

Le marché

Le **marché de la téléassistance est en pleine croissance** (10% par an) **en France mais aussi dans toute l'Europe¹**. **D'ici à 2020, environ 30% des européens auront plus de 60 ans**. En France, **225 000 nouveaux cas de personnes par an** sont touchés par la maladie d'Alzheimer ou des maladies apparentés². **La population de personnes dépendantes augmentera de 40 à 70% d'ici à 2040³**.

Outils

Projet développé sous Visual Studio en .NET à l'aide du langage C# pour une plateforme cible tournant sous Windows Mobile 6 avec le Compact Framework .NET version 3.5.

¹ Source la téléassistance en chiffres de l'AFRATA (Association Française de TéléAssistance)

² Source : « Le marché de la dépendance des personnes âgées », rapport stratégique mai 2007, Candestic

³ Source étude Xerfi : Accueil des personnes âgées

Introduction

Le nombre de personnes dépendantes dans le monde est en constante augmentation (1 nouveaux cas toutes les 7 secondes dans le monde). En Europe, plus particulièrement, la population âgée augmente dans tous les pays et la France, notamment, ne fait pas exception. Les changements de la société comme la réduction du nombre des aidants, l'allongement de la durée de vie, la diminution du nombre des professionnels de santé font que la téléassistance va prendre une place de plus en plus importante dans la vie de chacun, à la condition qu'elle réussisse à répondre aux défis du monde d'aujourd'hui⁴.

Ainsi, le marché de la téléassistance est en pleine croissance. Dans ce contexte, des solutions basées sur les technologies de géolocalisation précises pourront permettre d'assurer la sécurité, que ce soit des personnes autonomes présentant une fragilité ou des personnes à forte dépendance.

C'est pourquoi, Atos Origin ayant détecté cette opportunité souhaite développer un bracelet électronique qui puisse répondre aux besoins des malades et des aidants afin de prévenir les situations critiques (fugues, errances, chutes...).

C'est ainsi que le projet BÉA (Bracelet Electronique pour l'Autonomie) est né. Projet labellisé par 2 pôles de compétitivité Aerospace Valley et Cancer Bio Santé et subventionné dans le cadre du 7^{ème} appel à projet du FUI (Fonds Unique Interministériel).

C'est au sein de ce projet que j'ai évolué lors de mon stage de 6 mois à Atos Origin Integration qui a débuté le 01 mars 2009, stage s'intégrant bien à ma double formation (INSA – IAE) en nécessitant des compétences transversales à la fois en informatique et en management.

La première partie de mon stage exploitait les compétences acquises au cours de la formation en management de l'innovation à l'IAE, j'ai évalué les opportunités de marché pour la sureté des personnes âgées dépendantes. J'ai réalisé une étude de marché complète qui met en relief les débouchés éventuels et les recommandations à suivre. J'ai notamment du réaliser un dossier complet pour une demande de subventions auprès de la **DGE**.

Par la suite, j'ai réalisé les spécifications de la plateforme à développer (disponibles en annexe) ainsi qu'un programme sur PDA fonctionnant sous Windows Mobile. Ce programme permet de :

- récupérer les informations provenant de la puce GPS et de les compresser au maximum notamment en diminuant le nombre de points contenu dans un trajet pour limiter la bande passante utilisée ainsi qu'optimiser l'autonomie de la batterie

⁴ Source : « Le marché de la dépendance des personnes âgées », rapport stratégique mai 2007, Candestic
Pascal MIETLICKI – Projet BÉA – Rapport technique

- Enregistrer les fichiers dans différents formats (NMEA, GPX, NMEA Optimisé)
- Afficher la cartographie correspondante au trajet en cours
- Afficher la constellation satellite (nombre de satellites, puissance du signal et positionnement)
- Afficher la boussole indiquant la direction actuelle

C'est donc cette 2^{ème} partie technique représentative de la formation d'ingénieur INSA que se propose d'illustrer ce rapport.

A. Étude sur les trajets GPS

A.1.	Le format NMEA	8
1.	Type de trames	8
A.2.	Compression des trames	9
1.	Minimisation du nombre de points	9
2.	Algorithme d'optimisation	12
A.3.	Interface graphique en VBA.....	15

La présentation du projet BÉA met en relief un enjeu majeur : la maîtrise de l'autonomie du bracelet ainsi que de la bande passante pour l'envoi des données.

C'est ainsi que j'ai été chargé d'étudier les données GPS afin de minimiser le nombre de points significatifs d'un trajet. Travail essentiel pour la version 2 du bracelet où un programme complet sera développé par Atos Origin et devra prendre en compte l'ensemble des contraintes indiquées dans la partie précédente⁵.

Dans ce contexte, j'ai d'abord choisi de réaliser un programme sous VBA à l'aide de Microsoft Excel afin d'analyser les trajets GPS. J'ai choisi de commencer sous Excel car je ne disposais pas encore de l'IDE Visual Studio permettant de développer en .NET (étant donné que la demande d'installation sur mon poste n'avait pas encore aboutie).

A.1. Le format NMEA

Les traces brutes GPS sont sauvegardées au format NMEA. La norme NMEA 0183 est une spécification pour la communication entre équipements marins dont les équipements GPS. Elle est définie et contrôlée par la National Marine Electronics Association (NMEA)⁶.

1. Type de trames

Il existe plus d'une trentaine de trames GPS différentes. Le type d'équipement est défini par les deux caractères qui suivent le \$. Le type de trame est défini par les caractères suivants jusqu'à la virgule. Par exemple :

```
$GPRMC,180432,A,4027.027912,N,08704.857070,W,000.04,181.9,131000,1.8,W,D*25
```

est une trame GPS de type RMC. Chaque trame a sa syntaxe propre, mais selon le cas elles peuvent ou doivent se terminer, après le *, par une somme de contrôle qui permet de vérifier que la trame n'a pas été endommagée avant sa réception⁷.

La trame RMC est une trame très courante qui donne les informations essentielles (RMC veut dire Recommended Minimum sentence C) :

Champ	Valeur	Signification
1	<u>180432</u>	UTC au format hhmmss.ss (18 heures, 4 minutes et 32 secondes)
2	A	Statut (A – donnée valide, V - Mauvaise)
3	<u>4027.027912</u>	Latitude géographique au format ddmm.mmmmmm (40 degrés et 27.027912 minutes)
4	N	Direction de la latitude (N - Nort, S - Sud)
5	<u>08704.85707</u> <u>0</u>	Longitude géographique au format ddmm.mmmmmm (87 degrés et 4.85707 minutes)

⁵ Source Annexe technique DGE – Projet BÉA

⁶ Source site web wikipedia (<http://fr.wikipedia.org/wiki/NMEA>)

⁷ Source site web wikipedia (<http://fr.wikipedia.org/wiki/NMEA>)

6	W	Direction de la longitude (E - Est, W - Ouest)
7	000.04	Vitesse fond en nœuds (0.04 nœuds)
8	181.9	Cap vrai en degrés (181.9°)
9	131000	Date au format ddmmyy (13 Octobre, 2000)
10	1.8	Variation magnétique (1.8°)
11	W	Direction de la variation magnétique (E - Est, W - Ouest)
12	D	Indication du mode de fonctionnement (A - autonome, D - différentiel, N - donné non valide)

Ce type de trame RMC donne les informations minimum recommandées dont les informations essentielles dont j'ai besoin pour l'étude des traces GPS.

A.2. Compression des trames

Le fait de ne prendre en compte que les trames RMC permet déjà d'optimiser la taille des fichiers NMEA : un fichier NMEA de 2621 lignes contient 306 lignes de trames RMC soit une optimisation de l'ordre de 88%.

Cependant, le programme final lira les informations de la puce GPS, où on récupère directement chaque information (latitude, longitude, altitude...), le but est alors d'optimiser le nombre de coordonnées (latitude, longitude) pour ne renvoyer que celles significatives.

1. Minimisation du nombre de points

Afin de renvoyer un minimum de données, il a fallu réfléchir à un algorithme permettant de minimiser le nombre de points. La solution retenue est simple mais néanmoins efficace : si la personne se dirige dans la même direction (par exemple 0° nord) pendant une certaine durée, l'algorithme devra prendre le premier et le dernier point et supprimer tous les points intermédiaires. Un GPS renvoyant généralement les coordonnées toutes les secondes (fréquence d'1 hertz), cette solution permet de supprimer un nombre non négligeable de points.

Pour l'illustrer, prenons un exemple :

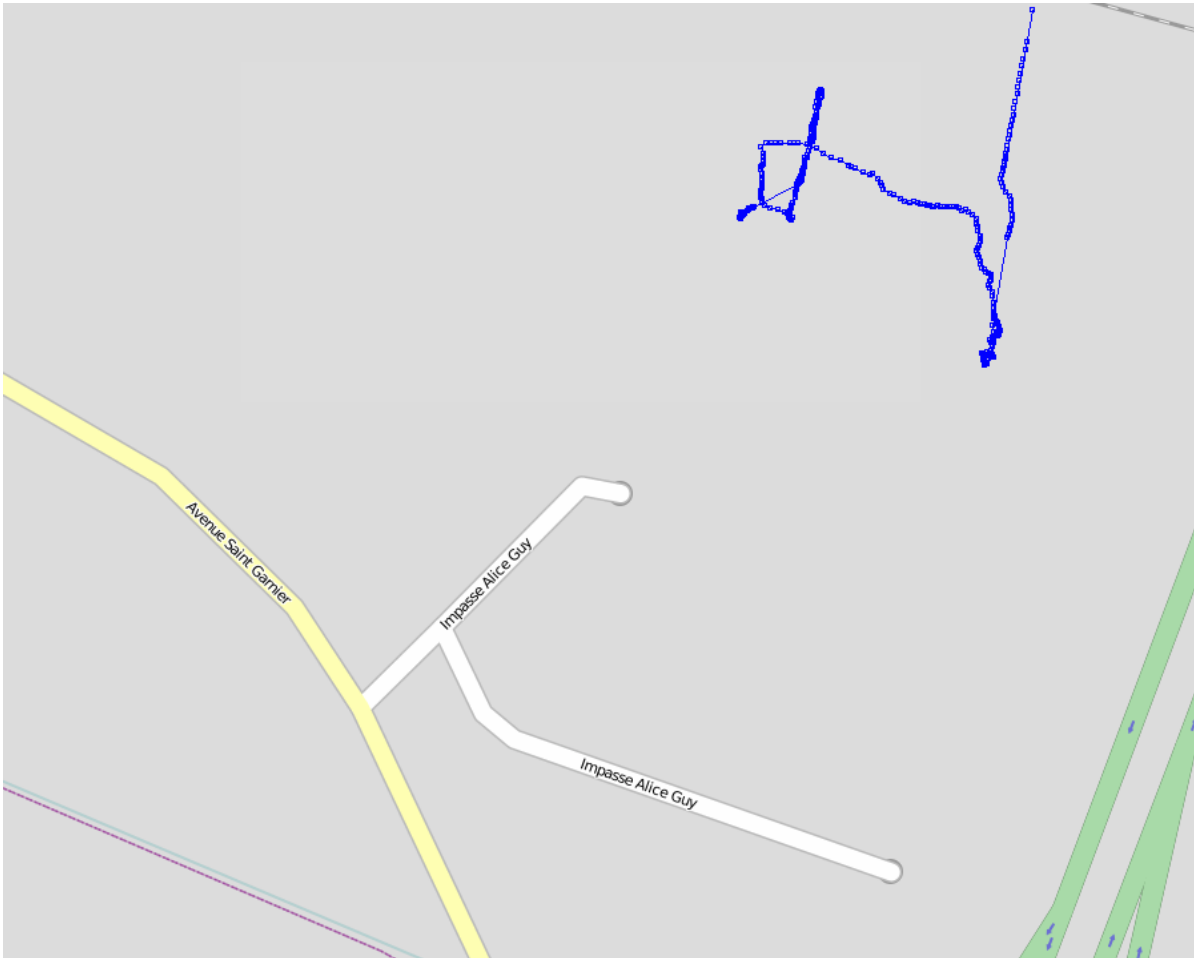


Illustration 1 : exemple de trajet GPS piéton

Si on zoome un peu en haut à droite, on remarque toute une série de points alignés qui indique que l'appareil contenant la puce GPS se dirigeait dans la même direction pendant quelques secondes. Sur le morceau de chemin ci-dessous, les points en rouge sont ceux à supprimer ou plutôt à ignorer par le programme lors de la capture en temps réel des points et les points en bleu sont ceux à garder.

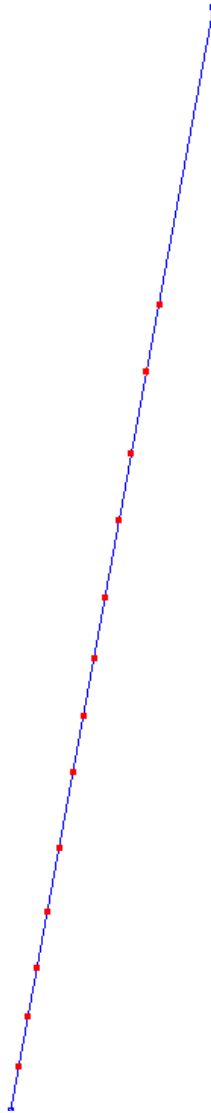


Illustration 2 : illustration de l'algorithme de suppression des points GPS

Malgré sa simplicité, cet algorithme permet une optimisation de l'ordre de 19%. En effet, il a été testé sur 18 trajets routiers dont les fichiers NMEA ont été récupérés et la moyenne du pourcentage d'optimisation global est de 19,11%.

Nom du trajet	Nombre de trames RMC	Nombre optimisé de trames RMC	Pourcentage d'optimisation
Auch Labège	5188	3741	27,89%
Balma Labège	688	588	14,53%
Blagnac CNES	898	748	16,70%
CNES Blagnac	1337	1043	21,99%
CNES Labège	413	369	10,65%
CNES Rabastens	2263	1791	20,86%
Cugnaux Labège	1220	876	28,20%
Labastide Beauvoir Labège	1201	886	26,23%
Labège Auch	5383	3923	27,12%
Labège Balma	570	500	12,28%

Labège CNES	406	342	15,76%
Labège Cugnaux	1188	858	27,78%
Labège Esquirol	3464	3104	10,39%
Labège Labastide Beauvoir	3086	2588	16,14%
Labège Médiathèque	3122	2860	8,39%
Rabastens CNES	2180	1726	20,83%
		TOTAL	19,11%

Illustration 3 : Résultats trajets routiers optimisés

Cependant, à terme cet algorithme sera mis à contribution sur des trajets piétonniers, et son efficacité dépend du type de trajet, si l'utilisateur a un comportement balbutiant alors l'optimisation ne sera pas très efficace.

Nous avons donc testé l'algorithme sur un trajet piétonnier arbitraire qui contient 306 trames RMC, on obtient alors 293 trames RMC dans le fichier optimisé soit une optimisation de l'ordre de 4,25%. Toutefois ce pourcentage est relatif car ce trajet possède un nombre relativement faible de coordonnées GPS et cela s'explique par le fait qu'il ne dure que 5 minutes. On peut faire l'hypothèse que sur un trajet beaucoup plus long et donc plus représentatif, le nombre de points sera beaucoup plus conséquent et le pourcentage d'optimisation aussi.

Néanmoins, ce trajet piétonnier a permis de mettre en évidence le fait qu'on pouvait encore améliorer l'optimisation, j'ai donc rajouté dans le programme un paramètre appelé « seuil ». Ce paramètre est expliqué et illustré par la suite.

2. Algorithme d'optimisation

Comme illustré précédemment, on se base sur la « direction »⁸ (en anglais « bearing ») entre 2 coordonnées GPS⁹ ce qui permet de supprimer les points intermédiaires qui auraient la même direction que les points précédents. Le code ci-dessous montre une partie de l'algorithme d'optimisation :

```

Do While (i <= total_ligne And Not IsEmpty(Cells(i, 4)) And Not IsEmpty(Cells(j, 4)))

    'on recupere les infos de latitudes et longitudes des 2 premiers points qu'on converti en degres decimaux
    first_lat1 = parseDeg(Cells(i, 4).Value & CStr(Cells(i, 5).Value), Visu)
    first_lat2 = parseDeg(Cells(i + 1, 4).Value & CStr(Cells(i + 1, 5).Value), Visu)
    first_lon1 = parseDeg(Cells(i, 6).Value & CStr(Cells(i, 7).Value), Visu)
    first_lon2 = parseDeg(Cells(i + 1, 6).Value & CStr(Cells(i + 1, 7).Value), Visu)
    TimeInSec = TimeInSec + (2 / PerEch)
    'On calcule le bearing de ces 2 points
    AncBear = Bearing(first_lat1, first_lon1, first_lat2, first_lon2)
    'on prend ce bearing comme reference
    j = i + 2
    Do
        'on boucle jusqu'a trouver un bearing different
        'cela permet de minimiser le nombre de points à sauvegarder
        second_lat1 = parseDeg(Cells(j, 4).Value & CStr(Cells(j, 5).Value), Visu)
    
```

⁸ Aviation formulary : ensemble de calculs pour l'aviation (<http://williams.best.vwh.net/avform.html>)

⁹ Une coordonnée GPS est un point qui contient une latitude et une longitude

```

second_lat2 = parseDeg(Cells(j + 1, 4).Value & CStr(Cells(j + 1, 5).Value), Visu)
second_lon1 = parseDeg(Cells(j, 6).Value & CStr(Cells(j, 7).Value), Visu)
second_lon2 = parseDeg(Cells(j + 1, 6).Value & CStr(Cells(j + 1, 7).Value), Visu)
NewBear = Bearing(second_lat1, second_lon1, second_lat2, second_lon2)
j = j + 2
TimeInSec = TimeInSec + (2 / PerEch)
If (j > total_ligne) Then
    Exit Do
End If

Loop While ((NewBear >= (AncBear * (1 - (Seuil / 100)))) And (NewBear <= (AncBear *
(1 + (Seuil / 100))))

```

Illustration 4: Portion de l'algorithme d'optimisation en VBA

On remarque que le programme prend une valeur appelée « Seuil », cette valeur permet d'être plus « indulgent » concernant les points n'ayant pas la même direction. Cela permet de supprimer des points que l'on ne considère pas significatifs. Par exemple, pour une personne se déplaçant vers 180° soit le sud, on pourra considérer, par exemple, que tous les points entre 170° et 190° ne représentent pas un changement significatif de direction. Dans le code en VBA, ce seuil est exprimé en pourcentage (pour 10% avec une valeur de référence de 180°, on ignore les points entre 162° et 198°).

D'autre part, le programme calcule le « bearing » (ou direction)¹⁰ entre chaque paire de points. J'ai choisi de le calculer manuellement bien qu'il soit présent normalement dans les trames RMC pour la simple et bonne raison que, parfois, le GPS ne le calcule pas ou mal :

Type	Temps	Statut	Latitude	DirLat	Longitude	DirLon	Vitesse	Bearing	Date	Chk
\$GPRMC	82713	A	4336,055087	N	121,790524	E	1,1	20,4	260509	A*5D
\$GPRMC	82714	A	4336,05511	N	121,790916	E			260509	A*6E
\$GPRMC	82715	A	4336,055306	N	121,791354	E	1,2	38	260509	A*5F
\$GPRMC	82716	A	4336,055593	N	121,791686	E	1,1	31,5	260509	A*53
\$GPRMC	82717	A	4336,055862	N	121,791821	E			260509	A*65
\$GPRMC	82718	A	4336,056118	N	121,791852	E			260509	A*69
\$GPRMC	82719	A	4336,056468	N	121,791842	E			260509	A*6B
\$GPRMC	82720	A	4336,056699	N	121,792013	E	1,1	356,8	260509	A*6A

Illustration 5 : Bearing incorrect dans les trames RMC

On remarque que, parfois, le calcul ne se fait tout simplement pas, il m'a donc paru plus fiable de le recalculer. D'autre part, face aux nombreux types de puces GPS existantes n'ayant pas le même format de trame, on ne peut pas forcément savoir avec certitude où se trouve le « bearing » correct.

```

Public Function Bearing(ByVal lat1 As Double, ByVal lon1 As Double, ByVal lat2 As Double,
ByVal lon2 As Double)
lat1 = Radians(lat1)
lat2 = Radians(lat2)

Dim dLon As Double

```

¹⁰ Ensemble de scripts pour les calculs GPS (<http://www.movable-type.co.uk/scripts/latlong.html>)

```
dLon = Radians((lon2 - lon1))
```

```
Dim y As Double
```

```
Dim x As Double
```

```
Dim notDeg As Double
```

```
y = Sin(dLon) * Cos(lat2)
```

```
x = Cos(lat1) * Sin(lat2) - Sin(lat1) * Cos(lat2) * Cos(dLon)
```

```
If x = 0 And y = 0 Then
```

```
Bearing = 0
```

```
Else
```

```
Bearing = Brng(Application.WorksheetFunction.Atan2(x, y))
```

```
End If
```

```
End Function
```

Illustration 6 : Calcul du “bearing” (ou direction) en VBA

Après avoir créé les macros Visual Basic, j’ai testé le programme afin de vérifier que les coordonnées optimisées étaient cohérentes. Pour faire la comparaison, j’ai utilisé un outil de visualisation disponible sur Internet appelé GPS visualizer (<http://www.gpsvisualizer.com>). Cet outil m’a permis de constater que le trajet optimisé est similaire au trajet initial.



Illustration 7 : Comparaison avec GPS visualizer (gauche : initial, droite : optimisé)

On remarque tout de suite les limites de cet outil, on ne peut pas voir précisément les points ni le nombre de points ce qui ne permet pas de faire une comparaison efficace. Par la suite, je me suis servi d’un outil libre écrit en Java beaucoup plus précis que j’ai modifié pour les besoins de l’analyse: Prune GPS.

A.3. Interface graphique en VBA

Le programme sous VBA contient plusieurs modules :

- **CalculBearing** qui permet le calcul de la direction mais aussi des fonctions utiles comme passer de degrés à radians, passer du format degrés-minutes vers le format décimal¹¹, etc.
- **EcrireFicGPRMC** qui permet à partir d'une feuille de calcul, d'écrire l'ensemble des trames RMC dans un fichier
- **EcrireLatLon** qui renvoie la latitude et longitude décimale dans un fichier
- **EcireOptimLatLon** qui écrit les latitudes et longitudes optimisés dans un fichier
- **LireFicNMEA** qui lit un fichier NMEA et qui parse les trames RMC en remplissant les feuilles de calcul

Une fois toutes ces fonctions effectuées et validées, j'ai décidé de créer une interface graphique de manière à faciliter l'utilisation des macros.

L'interface graphique se lance dès que le fichier Excel est ouvert.

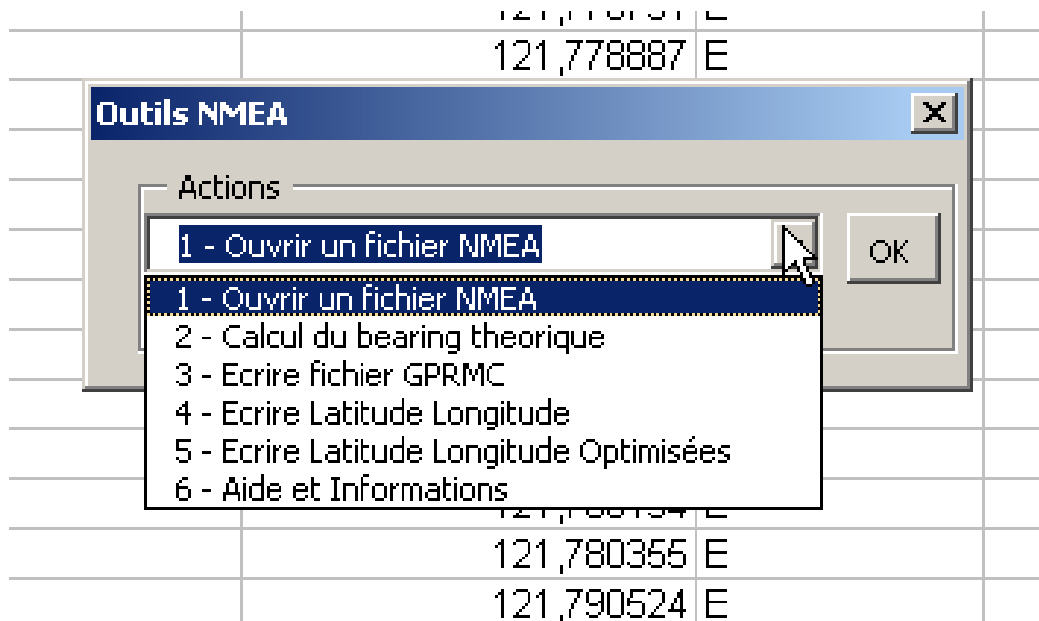


Illustration 8 : Interface graphique sous VBA

L'interface donne l'accès à six fonctionnalités, la plus importante est sans doute l'écriture dans un fichier des latitudes et longitudes optimisées. Lorsque l'utilisateur sélectionne cette dernière option, d'autres paramètres s'affichent :

¹¹ Pour récupérer une valeur décimale standard, on passe du format ddm. mmmm vers dd. dddd, on récupère la partie mm. mmmm de la valeur initiale qu'on divise par 60.

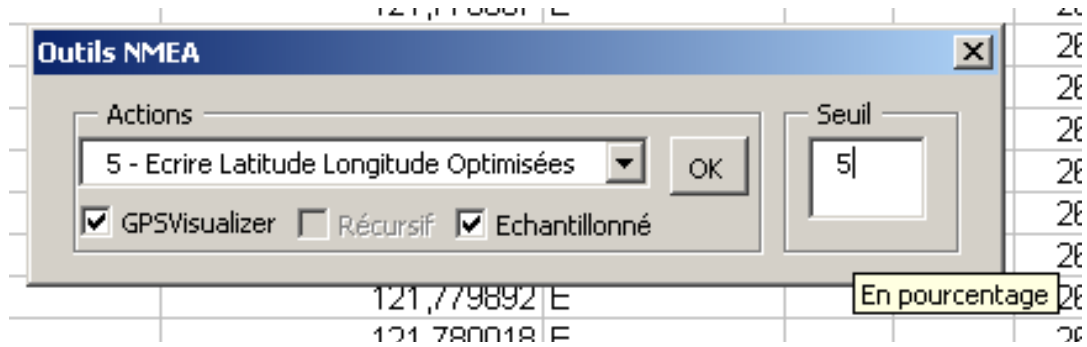


Illustration 9 : Menu graphique pour l'optimisation en VBA

L'utilisateur peut indiquer divers paramètres :

- **GPS Visualizer** permet de formater les données de manière à ce qu'elles soient directement lisibles sur le site (<http://www.gpsvisualizer.com>)
- **Récursif** : par défaut le calcul est non récursif (itératif), une fonction récursive a été réalisée de manière à supprimer les bugs, on peut considérer que si les résultats du calcul itératif et récursif sont égaux alors le calcul est « correct » car les algorithmes sont indépendants en théorie. Il faut savoir que, si l'option Récursif est sélectionnée, alors l'option échantillonnée et la sélection du seuil ne sont plus possibles (le seuil par défaut est à 0) car la fonction récursive est dépourvue de ces fonctionnalités.
- **Échantillonné** simule un échantillonnage des données : si pas de changement de cap alors on écrit une position toutes les 5 secondes, si changement de cap on écrit toutes les secondes. Ceci afin de simuler l'utilisation finale réelle car on peut très bien imaginer une personne immobile depuis un certain temps et, dans ce cas, le programme ne renverrait jamais sa position.
- **« Seuil »** permet d'indiquer le seuil de tolérance en pourcentage utilisé lors de la comparaison de la direction des différentes paires de points

Un autre menu, le menu 6 est un menu d'aide permettant notamment de récapituler les différentes fonctionnalités offertes par le programme à l'utilisateur :

Outils NMEA

AIDE

L'option 1 permet d'ouvrir n'importe quel fichier NMEA et de le charger dans Excel sous la feuille « NMEA » ainsi que de récupérer l'ensemble des trames RMC (Recommended Minimum Data) sous la feuille « GPRMC »

L'option 2 permet de recalculer la direction (« bearing » en anglais) manuellement en utilisant la formule :

$$\text{Bearing} = \text{ATAN2}(\text{COS}(\text{lat1}) * \text{SIN}(\text{lat2}) - \text{SIN}(\text{lat1}) * \text{COS}(\text{lat2}) * \text{COS}(\text{lon2} - \text{lon1}), \text{SIN}(\text{lon2} - \text{lon1}) * \text{COS}(\text{lat2}))$$

Comme atan2 retourne des valeurs entre $-\pi$ et $+\pi$ (soit -180° et $+180^\circ$). Afin de normaliser le résultat en tant sous un angle complet de 360° (où 0° indique le nord, 90° l'est, 180° le sud et 270° l'ouest), on convertit en degrés puis on utilise la formule : $(\theta + 360) \text{ Mod } 360$. Ce qui donne la direction initiale orthodromique entre le point initial et le point final. Une nouvelle colonne intitulé « Initial Bearing Théorique » apparaîtra dans la feuille GPRMC avec les résultats du calcul pour chaque paire de points.

L'option 3 permet de générer le fichier GPRMC (données suffisantes pour une visualisation)

L'option 4 permet d'écrire la latitude et la longitude dans un fichier (exploitable sous GPS Visualizer – <http://www.gpsvisualizer.com/>)

L'option 5 permet d'écrire dans un fichier les latitudes et longitudes « optimisées » c'est-à-dire le point initial et le point final des intervalles ayant le même cap (bearing ou direction). Quelques options supplémentaires sont disponibles :

Pour GPS Visualizer permet de formater les données afin de les afficher sous GPS Visualizer

Echantillonné simule un échantillonnage des données : si pas de changement de cap alors on écrit une position toutes les 5 secondes, si changement de cap on écrit toutes les secondes

Récurusif par défaut le calcul est non récursif (itératif), une fonction récursive a été réalisée de manière à supprimer les bugs, on peut considérer que si les résultats du calcul itératif et récursif sont égaux alors le calcul est « correct » car les algorithmes sont indépendants en théorie. Il faut savoir que, si l'option Récursif est sélectionnée, alors l'option échantillonnée et la sélection du seuil ne sont plus possibles (le seuil par défaut est à 0) car la fonction récursive est dépourvue de ces fonctionnalités.

Quelques explications sur NMEA (<http://fr.wikipedia.org/wiki/Nmea>) :

La norme NMEA 0183 est une spécification pour la communication entre équipements marins dont les équipements GPS. Elle est définie et contrôlée par la National Marine Electronics Association. Il existe plus d'une trentaine de trames GPS différentes. Le type d'équipement est défini par les deux caractères qui suivent le \$. Le type de trame est défini par les caractères suivants jusqu'à la virgule. Les plus connus et répandus sont :

Informations minimums recommandées

\$GPRMC,180432,A,4027.027912,N,08704.857070,W,000.04,181.9,131000,1.8,W,D*25

Champ	Valeur	Signification
1	<u>180432</u>	UTC au format hhmmss.ss (18 heures, 4 minutes et 32 secondes)
2	A	Statut (A – donnée valide, V - Mauvaise)
3	<u>4027.027912</u>	Latitude géographique au format ddmn.mmmmm (40 degrés et 27.027912 minutes)

OK

Illustration 10 : Menu d'aide de l'interface graphique en VBA

B. Le programme sur PDA

B.1.	Les modules.....	19
B.2.	GPS Intermediate Driver	19
1.	Sans ce type d’outil	19
2.	Avantages de ce type d’outil.....	20
3.	Diagramme de classes	21
B.3.	GPS Tracker	22
1.	Informations	22
2.	Logs	23
3.	Seuil	25
4.	Signal.....	26
5.	Boussole.....	27
6.	Route	28
7.	SharpMap	30
8.	Diagramme de classes	31
9.	Transmission des données en ASN.1	32

Le travail effectué sous VBA m'a permis de valider les différentes fonctions utilisées afin de faciliter au maximum la transition sous le langage .NET. J'ai donc réalisé un « prototype » sur PDA pour l'analyse et la compression des trajets GPS ainsi que l'affichage de ces trajets. Ce prototype est présenté dans cette partie.

B.1. Les modules

Le programme sur PDA est basé sur 2 modules :

- Le module **Location** qui gère le dialogue avec la puce GPS en utilisant l'outil « GPS Intermediate Driver ». C'est en fait une couche d'abstraction matérielle qui permet d'accéder à n'importe quel GPS (interne ou lié à un port de communication sur Smartphone).
- Le module **GPS Tracker** qui contient la classe pour l'optimisation, l'interface graphique, la classe pour l'exportation vers le format GPX, la classe gérant le tableau pour l'affichage des coordonnées GPS

B.2. GPS Intermediate Driver

« GPS Intermediate Driver » fournit des fonctionnalités aux applications qui utilisent des appareils GPS. Il le fait en fournissant une couche intermédiaire pour l'interrogation du GPS. Cet outil permet aux développeurs d'écrire du code qui fonctionne avec tout type de matériel GPS.

1. Sans ce type d'outil

Sans des outils comme « GPS Intermediate Driver », les applications accèdent généralement au matériel GPS par l'intermédiaire d'un port COM. Selon le type de matériel, ce port COM peut être un véritable port série, auquel le matériel GPS est connecté par l'intermédiaire d'un câble série. Mais cela peut aussi être un port série « virtuel » rattachés au matériel GPS grâce à un périphérique, SDIO, Bluetooth, ou autres.

Avec la plupart des dispositifs GPS, une fois la connexion effectuée directement sur ce port, celle-ci est exclusive, de sorte qu'une seule application à la fois peut interagir avec le matériel GPS.

Ce type de connexion permet d'accéder aux données NMEA brutes qu'il faut ensuite « parser » (ou filtrer) afin de récupérer les informations voulues.

2. Avantages de ce type d'outil

Avec un outil comme « GPS Intermediate Driver », les applications ne dialoguent plus directement avec le matériel GPS. Le pilote se charge de le faire. Ainsi « GPS Intermediate Driver » offre deux avantages principaux:

- Il permet à plusieurs applications d'utiliser le matériel GPS en même temps.
- Il n'est plus nécessaire que chaque application parse les données NMEA provenant du GPS car l'API fournit par « GPS Intermediate Driver » analyse elle-même les chaînes NMEA du GPS et renvoie directement les informations parsées.

D'autre part, ces atouts font qu'il n'est plus nécessaire que l'utilisateur indique à chaque application à quel port est connecté le matériel GPS. Car « GPS Intermediate Driver » récupère directement l'information par l'intermédiaire du panneau de configuration du terminal.

3. Diagramme de classes

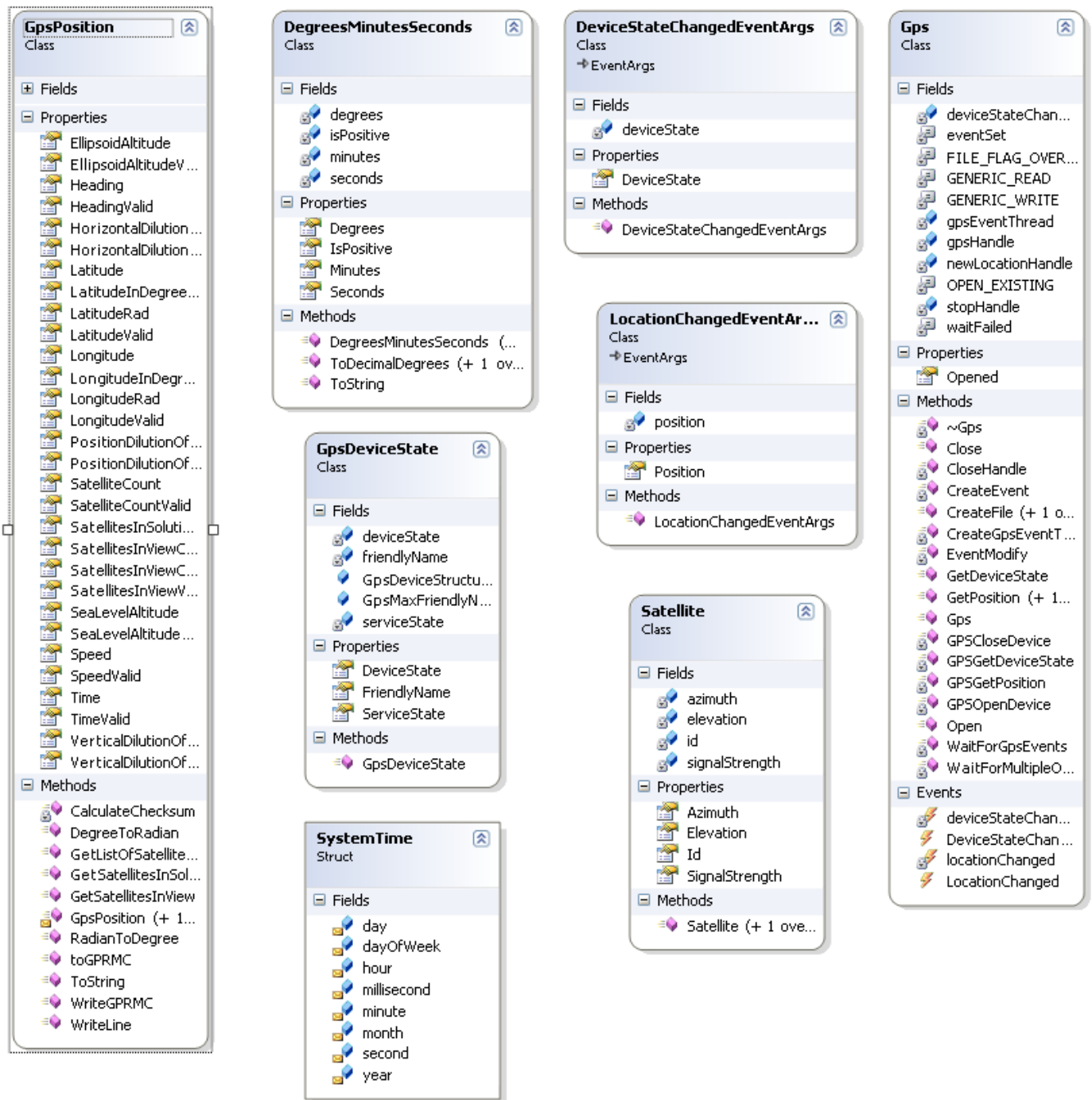


Illustration 11 : Diagramme de classes du module “Location”

Le module « Location » est un module d'exemple de l'utilisation du pilote « GPS Intermediate Driver ». Il permet d'accéder aux données GPS et de déclencher un événement dès que des données sont reçues qui sont ensuite traitées par l'application « GPS Tracker » pour mettre à jour les données et effectuer les traitements.

B.3. GPS Tracker

Le projet BÉA a nécessité le développement d'un programme sur PDA afin d'exploiter les données récupérées du GPS intégré. Ce programme sur PDA fonctionne sur *Windows Mobile* avec le Compact Framework version 3.5 de *.NET*. L'environnement de développement utilisé fut *Visual Studio 2009*. La suite de ce chapitre présente les principaux composants de ce programme.

1. Informations

L'onglet "Infos" permet de voir, en temps réel, différentes informations :

- nom de la puce GPS (constructeur) ;
- activation ou non du GPS (On ou Off) ;
- la latitude décimale ;
- la longitude décimale ;
- un tableau récapitulatif des coordonnées au format degrés-minutes (ddmm.mmmm) ainsi que le temps pour chacune de ces coordonnées.

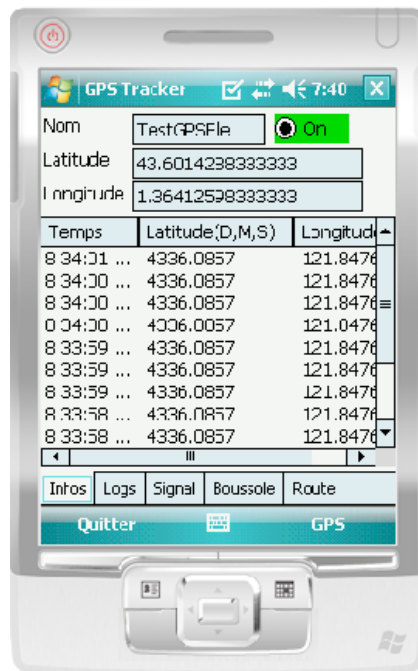


Illustration 12 : Onglet "Informations" du programme PDA

```

if ((position.LatitudeValid) && (position.Latitude!=0))
{
    // On récupère la position courante
    GpsPosition currentpos = position;

    // On met à jour la latitude
    this.txtLatitude.Text = currentpos.Latitude.ToString();

    if ((position.LongitudeValid) && (position.Longitude!=0))
    {

```

```
// On met à jour la longitude  
this.txtLongitude.Text = currentpos.Longitude.ToString();
```

Illustration 13 : Code pour la mise à jour des « TextBox » de latitude et longitude en C#

A chaque nouvelle position, si celle-ci est valide et cohérente, on l'ajoute au tableau « eventLog » de type « ListView ». Cela permet à l'utilisateur de revoir l'ensemble des coordonnées GPS déjà traitées.

```
if (position.TimeValid)  
{  
    evtLog.Add(position.Time, position);  
}
```

Illustration 14 : Portion de code pour la mise à jour de la liste des positions GPS en C#

2. Logs

L'onglet "logs" permet, comme son nom l'indique, de gérer les logs ou fichiers de sortie générés par le programme. Il existe différents types de sorties :

- **GPX** qui est un fichier XML spécifique pour les coordonnées GPS, ouvert et compatible avec la plupart des programmes ;
- **GPRMC** qui est une des trames d'un fichier NMEA standard, c'est la "Recommended minimum specific GPS/Transit data" qui contient notamment la longitude et la latitude au format degrés-minutes ainsi que la direction (Nord, Sud, Est, Ouest) ;
- **un fichier "optimisé"** qui permet de renvoyer une liste de latitude et longitude soit au format GPRMC (si la case "visu" est cochée) soit une liste brute séparée par des virgules.

Le fichier de sortie "optimisé" se base sur un seuil. Il suffit de renseigner la case "Seuil" pour le spécifier. Il y a 2 types de seuil :

- en pourcentage si la valeur rentrée est inférieur à 1 ou qu'elle contient le symbole '%' (exemple : 5%, 10%, 15% ou 0.05, 0.1, 0.15...etc) ;
- en degrés si la valeur rentrée est supérieur ou égale à 1.

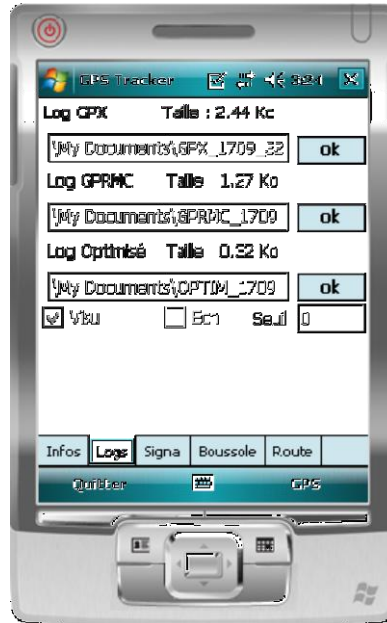


Illustration 15 : Onglet "Logs" du programme PDA

```

private void Update_Size(SaveFileDialog fd, String type)
{
    long sizeOctets = new FileInfo(fd.FileName).Length;
    Double sizeKo = Math.Round((Double)sizeOctets / 1000, 2);

    if (type.Equals("GPRMC"))
    {
        labTailleGPRMC.Text = string.Format("Taille : {0} Ko", sizeKo);
    }
    else
    {
        if (type.Equals("GPX"))
        {
            this.labTailleGPX.Text = string.Format("Taille : {0} Ko", sizeKo);
        }
        else
        {
            if (type.Equals("OPTIM"))
            {
                this.labTailleOptim.Text = string.Format("Taille : {0} Ko", sizeKo);
            }
        }
    }
}

```

Illustration 16 : Portion de code pour la mise à jour de la taille des fichiers de « Logs » en C#

3. Seuil

Pour mieux comprendre, l'algorithme permettant de minimiser le nombre de points d'un trajet se base sur une logique simple : Si entre plusieurs points la "direction" reste constante, alors le programme garde le premier et le dernier point de l'ensemble des points composant cette portion de trajet. Le seuil va permettre d'être plus "indulgent" sur cette direction de manière à supprimer encore plus de points. Si un seuil est spécifié alors la direction de référence sera modulée en fonction de ce seuil :

```

this.newBear = (Double) this.Bearing();
if (this.seuil < 0.9999999)
{
    //Seuil en pourcentage
    if (newBear <= ancBear * (1 - this.seuil) || newBear >= ancBear * (1 + this.seuil))
    {
        this.keep = true;
        //On écrit dans le fichier
        pos1.WriteGPRMC(output);
        // On prend la nouvelle valeur comme référence
        ancBear = newBear;
    }
    else
    {
        this.keep = false;
    }
}
else
{
    //Seuil en degrees
    if (newBear <= (ancBear - this.seuil) || newBear >= (ancBear + this.seuil))
    {
        this.keep = true;
        //On écrit dans le fichier
        pos1.WriteGPRMC(output);
    }
    else
    {
        this.keep = false;
    }
}
}

```

Illustration 17 : Optimisation en temps réel en C# sous .NET

La principale difficulté de cette partie fut de « convertir » le code VBA d'Excel qui prenait en entrée un ensemble de coordonnées GPS déjà fournies dans un tableau vers un code C# qui puisse traiter les coordonnées au fur et à mesure en temps réel. Une des solutions fut d'utiliser une variable statique pour stocker la direction précédente.

```

if (lastpos != null)
{
    // On est sur une nouvelle paire
    Optim opt = new Optim(lastpos, currentpos, seuil);
}

```

```

opt.WriteGPRMC(Optimfile, this.visu);
Optimfile.Flush();
Update_Size(savOptim, "OPTIM");
}

```

Illustration 18 : Utilisation de la classe "Optim" en C#

4. Signal

Cette partie permet de visualiser les informations des satellites captés par le GPS :

- La partie haute montre le positionnement des satellites, le point du milieu indique le zénith ;
- La partie basse montre la puissance du signal pour chacun des satellites.

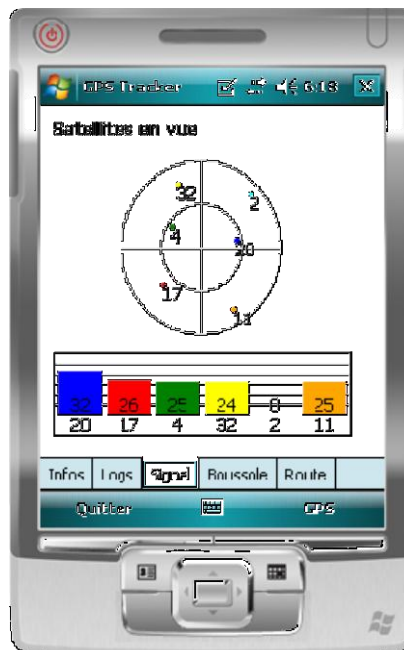


Illustration 19 : Onglet "Signal" du programme PDA

```

private void DrawGSV()
{
    //Génération des niveaux des différents signaux
    int SatCount = position.SatelliteCount;
    Bitmap imgSignals = new Bitmap(picGSVSignals.Width, picGSVSignals.Height);
    Graphics g = Graphics.FromImage(imgSignals);
    g.Clear(Color.White);
    Pen penBlack = new Pen(Color.Black, 1);
    Pen penGray = new Pen(Color.LightGray, 1);
    int iMargin = 4; //Distance vers la bordure de l'image
    int iPadding = 4; //Distance entre les graphes de signaux
    g.DrawRectangle(penBlack, 0, 0, imgSignals.Width - 1, imgSignals.Height - 1);
    StringFormat sFormat = new StringFormat();
    int barWidth = 1;
    if (SatCount > 0)
        barWidth = (imgSignals.Width - 2 * iMargin - iPadding * (SatCount - 1)) / SatCount;
}

```

```

//Pour dessiner les lignes horizontales
for (int i = imgSignals.Height - 15; i > iMargin; i -= (imgSignals.Height - 15 - iMargin) /
5)
    g.DrawLine(penGray, 1, i, imgSignals.Width - 2, i);
sFormat.Alignment = StringAlignment.Center;
//Pour dessiner les satellites
Satellite[] sat = position.GetSatellitesInView();
for (int i = 0; i < SatCount; i++)
{
    int startx = i * (barWidth + iPadding) + iMargin;
    int starty = imgSignals.Height - 15;
    int height = (imgSignals.Height - 15 - iMargin) / 50 * sat[i].SignalStrength;
    g.FillRectangle(new System.Drawing.SolidBrush(Colors[i]), startx, starty, barWidth, -
(sat[i].SignalStrength));
    if ((position.GetListOfSatellitesInSolution()).Contains(sat[i].Id))
    {
        g.DrawRectangle(penBlack, startx, starty, barWidth, -height);
    }
}

```

Illustration 20 : Portion de code pour la generation des signaux satellites en C#

5. Boussole

Comme son nom l'indique, affiche la direction actuelle du PDA en se basant sur les données GPS. L'utilisation du driver « GPS Intermediate Driver » fait que la récupération de la direction est extrêmement simple. Ce qui n'aurait pas été le cas si j'avais travaillé directement avec les données NMEA. L'image pour la boussole a été récupérée sur Internet, elle n'est pas forcément libre de droit mais il est très simple de la changer dans le programme.



Illustration 21 : Onglet "Boussole" du programme PDA

```

if (position.HeadingValid)
{
    textBoxHeading.Text = position.Heading.ToString();
    //Pour la boussole
}

```

```
pictureBoxPoint.Left = (int)(116 + 100 * Math.Sin(position.Heading * Math.PI / 180));  
pictureBoxPoint.Top = (int)(116 - 100 * Math.Cos(position.Heading * Math.PI / 180));  
}
```

Illustration 22 : Affichage de la boussole et positionnement du point de déplacement en C#

6. Route

Cette partie fut une des plus longues à implémenter, elle affiche une carte de la France et dessine les points GPS au fur et à mesure sur celle-ci. Pour se faire, le programme utilise une librairie appelée « *SharpMap* » qui a été « portée » sur le Compact Framework 3.5 de .NET. Cette carte permet plusieurs choses :

- de zoomer (ZoomIn et ZoomOut)
- de se déplacer sur la carte (Pan)

Le menu, en bas à gauche, permet aussi de :

- centrer automatiquement la vue sur l'ensemble du chemin parcouru grâce au bouton « centrer »
- charger une carte au format NMEA
- sauvegarder une carte au format GPX contenant les points optimisés (en fonction du seuil indiqué dans la partie « Logs »)

D'autre part, si l'utilisateur clique sur un point de la carte sans avoir sélectionné les options de « Zoom » ou de « Pan », alors un label s'affiche en bas de la carte et renvoie les coordonnées du point sélectionné.

Le choix de l'utilisation de la librairie **SharpMap** a été longuement réfléchi, j'ai d'abord tenté d'effectuer l'affichage des coordonnées GPS sans aucune librairie. J'ai malheureusement été confronté à différents problèmes et notamment celui de la transformation de la latitude et longitude décimales pour qu'elle s'affiche dans un cadre graphique en C#.

Par la suite, pour des raisons de pérennité du programme, j'ai décidé de faire appel à une librairie libre. Mon choix s'est porté sur SharpMap car cette librairie est librement utilisable même dans un programme payant (grâce à sa licence **LGPL**). D'autre part, une partie du code avait déjà été portée sur le Compact Framework .NET pour les applications embarquées.

J'ai utilisé une carte « shapefile » de la France pour l'affichage du trajet. La possibilité de concaténer différentes couches de « shapefile » fait que l'utilisateur peut avoir accès à des informations complémentaires visibles sur la carte (tel que les routes, les fleuves, etc). Pour le prototype, j'ai choisi de garder une carte très simple donc de taille restreinte. Celle-ci n'affiche que les régions. J'ai aussi ajouté une carte mondiale pour une utilisation faite en dehors de la France, à terme.

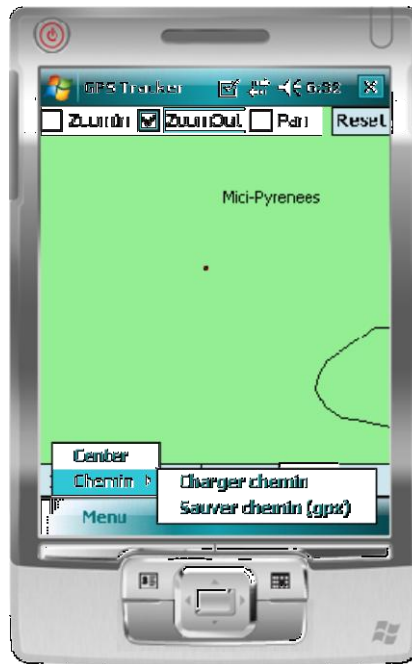


Illustration 23 : Onglet “Route” du programme PDA

```

if (_points != null)
{
    //Calculate bounding rectangle
    double maxLat = -90, minLat = 90, maxLon = -180, minLon = 180;

    //Calculate bounding box for array of locations
    for (int i = 0; i < _points.Count; i++)
    {
        //get latitudes
        if (_points[i].Y > maxLat)
            maxLat = _points[i].Y;

        if (_points[i].Y < minLat)
            minLat = _points[i].Y;

        //get longitudes
        if (_points[i].X > maxLon)
            maxLon = _points[i].X;

        if (_points[i].X < minLon)
            minLon = _points[i].X;
    }

    BoundingBox bbox = new BoundingBox(minLon, minLat, maxLon, maxLat);
    myMap.ZoomToBox(bbox);

    this.imgMap.Refresh();
}

```

Illustration 24 : Partie du code pour “centrer” la carte au niveau du parcours en C#

7. SharpMap

SharpMap est un moteur de rendu cartographique qui permet à partir de différentes sources de données SIG d'en obtenir le rendu graphique dans la projection souhaitée. SharpMap est écrit en C#, basé sur le framework .Net 2.0 et distribué sous la licence « GNU Lesser General Public License »¹². Le moteur de rendu peut être utilisé pour les clients légers, riches et même mobiles avec une librairie allégée.

Les bases de l'utilisation de SharpMap consistent à définir les différentes couches de données, les styles qui leurs correspondent, la projection et le niveau d'affichage de la carte. La génération de la carte utilise l'API GDI+ pour afficher les éléments cartographiques et laisse un large choix pour la personnalisation du style de chaque couche. SharpMap permet de faire la conversion entre les coordonnées graphiques et les coordonnées spatiales afin de pouvoir interagir avec la carte.

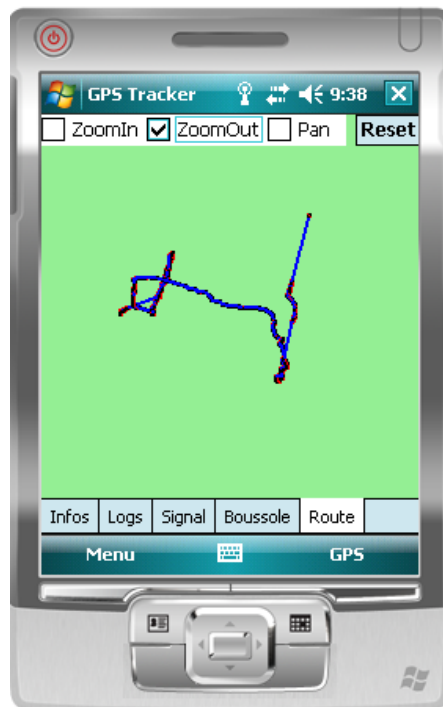


Illustration 25 : Affichage du trajet piéton sur PDA avec SharpMap

¹² <http://www.atolcd.com/btn-ii/systeme-dinformation-geographique-web-sig/sharpmap.html>

8. Diagramme de classes

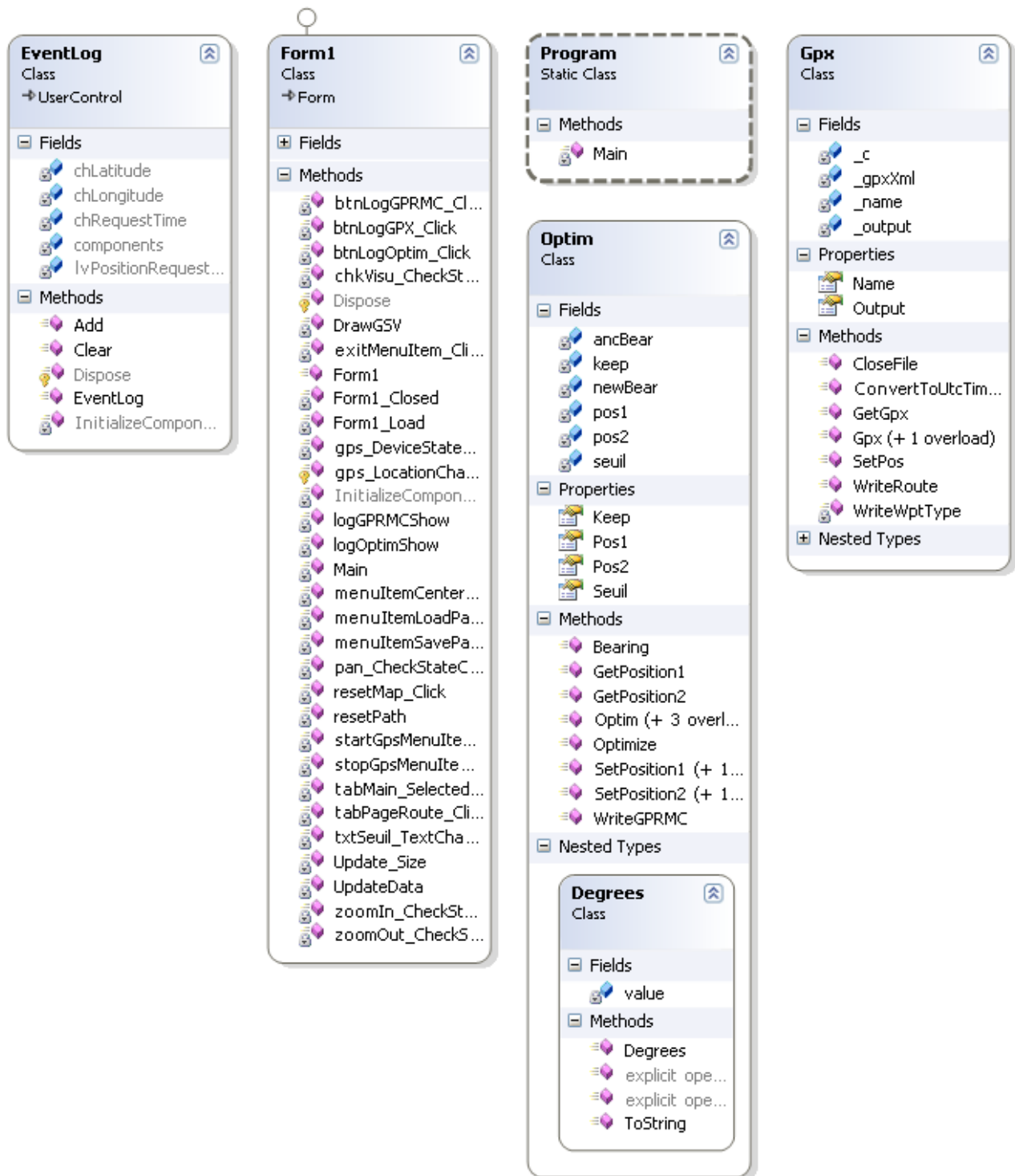


Illustration 26 : Diagramme de classes du module "GPS Tracker"

La classe sans doute la plus importante de ce diagramme est « Optim » c'est elle qui est chargée de déterminer, en temps réel, les coordonnées GPS à sauvegarder ou à supprimer en fonction de la direction de référence. Elle calcule le « cap » (direction ou bearing en anglais) pour déterminer, en fonction du seuil, si les nouvelles coordonnées GPS sont significatives ou non.

9. Transmission des données en ASN.1

Suite au développement des différentes parties du programme, j'ai intégré une solution de transmission des données au format ASN.1 (toujours dans un souci d'optimisation) développé par un collègue stagiaire. J'ai donc modifié le programme en sauvegardant l'ensemble des données optimisés dans un tableau qui est ensuite encodé au format ASN.1 et envoyé à un serveur écrit en Java. Ce serveur décode les données ASN.1 transmises et écrit dans un fichier les latitudes et longitudes des coordonnées GPS.

J'ai comparé les données en entrée et en sortie afin de valider la transmission des données ASN.1. Nous avons ensuite fait une démonstration à notre responsable : Mr Pascal PEDIRODA qui s'est déroulée avec succès.

C. Analyse des résultats

C.1.	Outil de visualisation et analyse des trajets.....	34
1.	Prune GPS	34
2.	Modification de Prune GPS	35
	<i>Choix des couleurs pour chaque trajet</i>	<i>35</i>
	<i>Calcul automatique du pourcentage de discordance.....</i>	<i>36</i>
C.2.	Analyse et détermination du seuil « idéal ».....	37

Suite au développement du programme sur PDA, il semblait nécessaire de trouver un seuil pertinent afin de modéliser un trajet piéton avec un minimum de points. Le but étant d'optimiser la bande passante et le trafic généré par le PDA.

C.1. Outil de visualisation et analyse des trajets

Comme évoqué précédemment, l'outil Web GPS visualizer n'est pas très approprié pour l'analyse des points. En effet, il n'affiche que les lignes et ne montre pas les détails de chaque point. La carte du programme sur PDA répond en partie à ce problème en affichant les points d'une couleur différente mais n'offre pas beaucoup de fonctionnalités en ce qui concerne les détails de ces points.

Ainsi, après de multiples recherches, j'ai finalement trouvé un outil libre appelé **Prune GPS**¹³. Il affiche le détail de chaque point et constitue un outil très complet.

1. Prune GPS

Prune est une application pour la visualisation, l'édition et la conversion des données GPS.

Prune est capable de charger des données textuelles (par exemple, séparées par des tabulations ou des virgules) ou XML. Cette application permet d'afficher, de modifier (par exemple supprimer des points) et sauvegarder les données GPS. Elle peut également exporter les données au format GPX ou KML. Prune est écrite en Java et est publiée sous licence GNU GPL.

¹³ <http://activityworkshop.net/software/prune/>

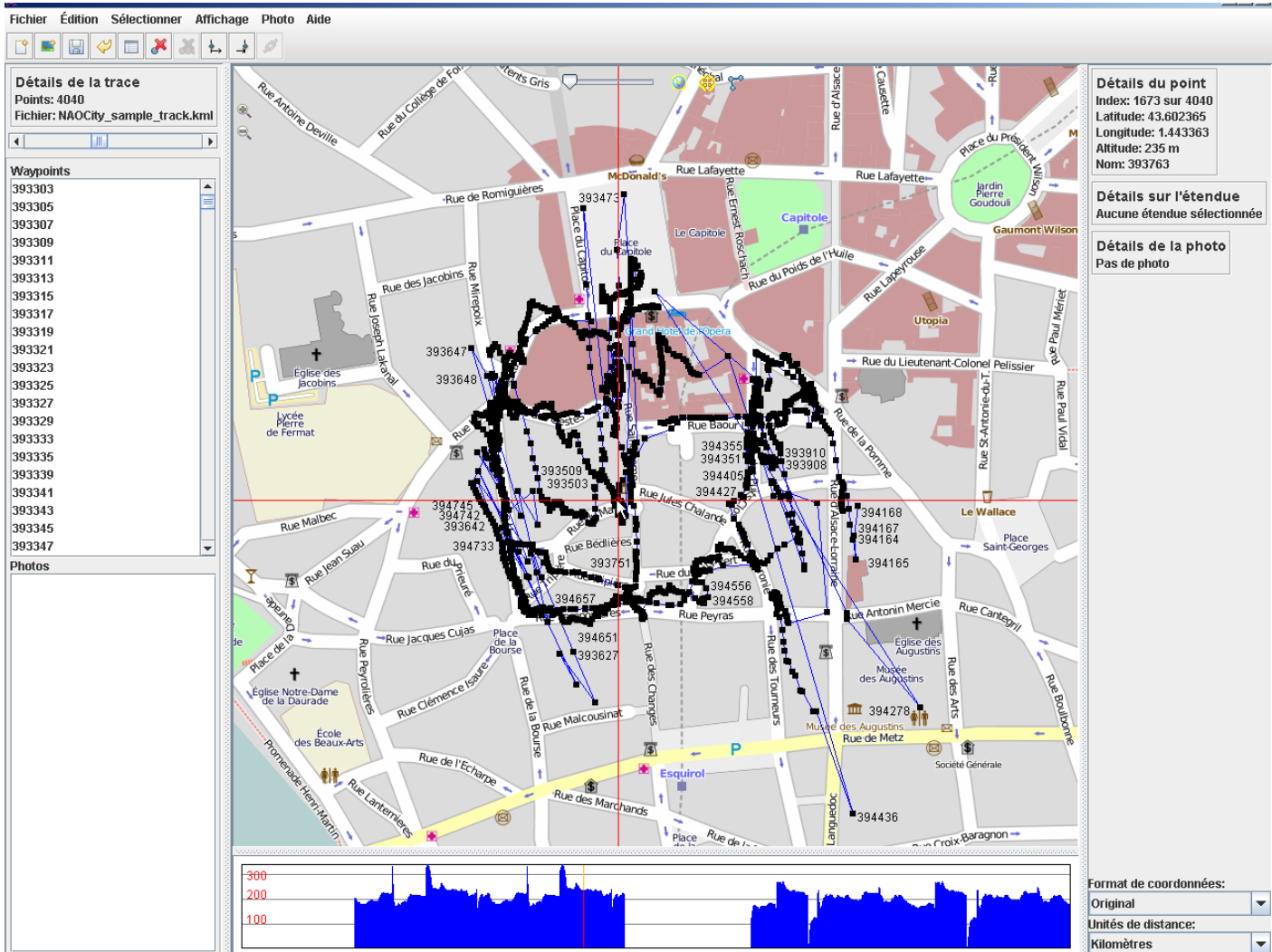


Illustration 27 : Analyse d'un trajet GPS avec Prune GPS sous Windows XP

2. Modification de Prune GPS

Pour les besoins du projet et de l'analyse, il a fallu que je compare les chemins optimisés avec le chemin initial. Or Prune n'a pas vraiment été développée dans ce but. J'ai donc fait plusieurs modifications afin de pouvoir faire une comparaison plus efficace.

Choix des couleurs pour chaque trajet

La première difficulté est que Prune ne permettait pas de choisir différentes couleurs pour différents trajets. J'ai donc rajouté cette fonctionnalité de manière à pouvoir comparer plus facilement 2 trajets différents.

```

if (answer == JOptionPane.YES_OPTION)
{
    // Combiner les données avec le parcours courant
    _undoStack.add(new UndoLoad(_track.getNumPoints(), loadedTrack.getNumPoints()));
}

```

```

//Pour la sélection de la couleur du parcours
while(newColor==null) {
    newColor = JColorChooser.showDialog(
        _frame,
        I18nManager.getText("dialog.color"),
        Color.RED);
}
//Comparaison des points discordants et calcul du pourcentage de discordance
_trackInfo.set_discord(loaderTrack.outsidePoints(_track));
//Mise à jour du nombre de points
_trackInfo.setNumPointsLoaded(loaderTrack.getNumPoints());
//Activation de la couleur choisie pour le parcours ajouté
loaderTrack.setColor(newColor);
_track.combine(loaderTrack);

```

Illustration 28 : Choix de la couleur du trajet et récupération des points discordants en Java

Calcul automatique du pourcentage de discordance

Afin d'éviter de vérifier à l'œil nu pour chaque trajet si la discordance est trop importante ou non, j'ai décidé de faire un calcul automatique du pourcentage de discordance basé à la fois sur le nombre de points « discordants » c'est-à-dire en dehors du trajet de référence mais aussi sur l'éloignement de ces points par rapport aux points de référence.

Sur l'image ci-dessous, on voit la version modifiée de Prune, en rouge le trajet optimisé et en bleu le trajet de référence. On voit alors s'afficher un pourcentage de compression, le nombre de points en dehors du trajet de référence et un pourcentage de discordance pour ce trajet optimisé.

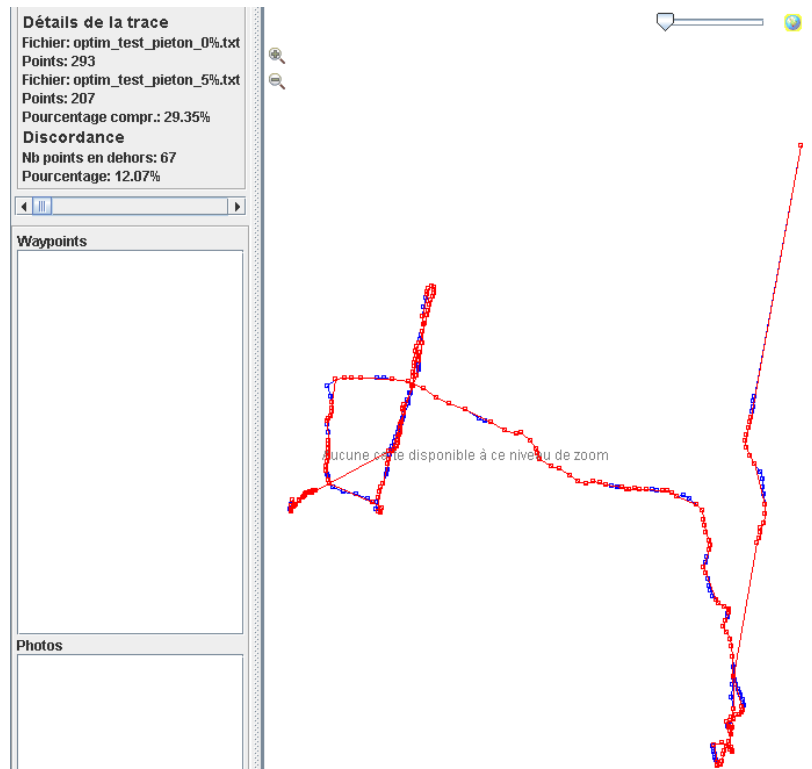


Illustration 29 : Affichage de 2 trajets avec Prune modifié (en bleu le trajet de référence)

Pour le calcul des points en dehors, je récupère l'équation de la droite entre 2 points GPS du trajet optimisé et je vérifie que les points du trajet initial vérifient l'équation de cette droite de manière à être sûr que la droite optimisée passe par ces points.

En conséquent, plus le seuil est grand et forcément, plus il y a de points du trajet de référence qui se trouvent en dehors.

Analyse et détermination du seuil « idéal »

Une fois les modifications faites de Prune, j'ai été chargé de déterminer un seuil « idéal » qui puisse être un bon compromis entre un nombre minimal de points tout en restant fidèle au trajet initial. J'ai commencé par exécuter le programme avec différents seuils dont voici les résultats :

Seuil	Nombre points optimisés	Pourcentage de compression	Nombre points discordants	Pourcentage discordance	Ratio	Différence
0%	293	4,25%	0	0%	N.C	N.C
5%	207	32,35%	67	11,59%	2,79	0,21
10%	165	46,08%	100	17,83%	2,58	0,28
15%	147	51,96%	118	21,82%	2,38	0,30
5°	228	25,49%	17	2,63%	9,69	0,23
6°	217	29,80%	21	3,50%	8,51	0,26
10°	171	44,12%	29	5,20%	8,48	0,39
12°	143	53,27%	37	6,55%	8,13	0,47

On remarque que les seuils en degrés donnent de meilleur résultat étant donné qu'ils sont constants et ne dépendent pas du « cap » de référence précédent. Les résultats démontrent que, si l'on souhaite optimiser la compression et minimiser la discordance alors le seuil de 5° semble être le meilleur compromis.

Cependant, si la compression s'avère être un critère un peu plus important, la meilleure solution semble être le seuil de 12° :

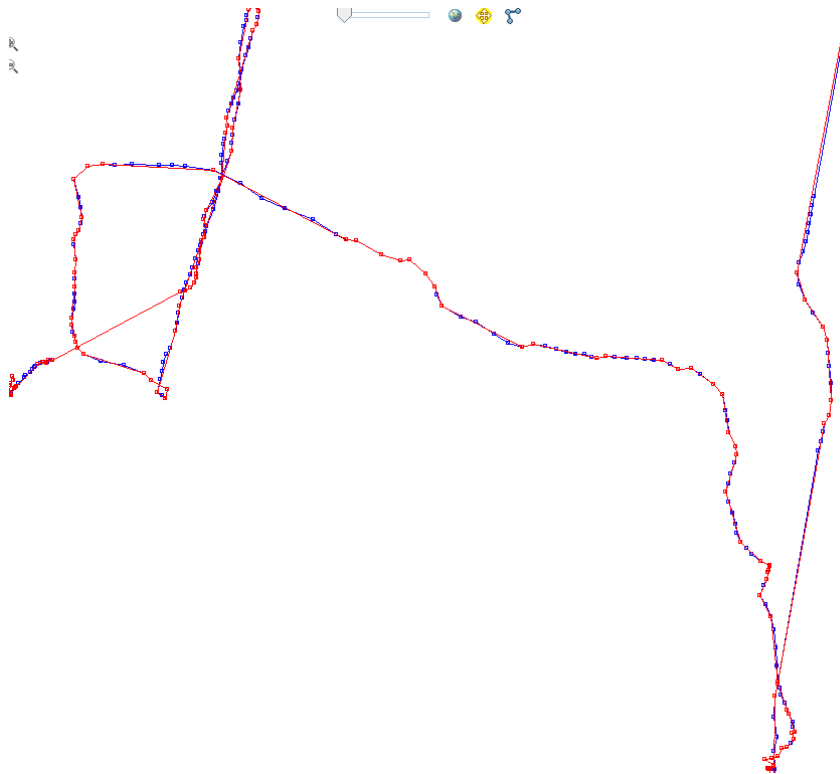


Illustration 30 : Présence d'une bonne concordance pour un seuil de 12°

En effet, graphiquement, on remarque une très bonne concordance avec le trajet initial (ici en bleu) et un très bon taux de compression (plus de 50%) avec 143 points pour un trajet initial de 306 points. D'autre part, si l'on fait la différence entre ce taux de compression et le taux de discordance, c'est la valeur pour laquelle on obtient le meilleur résultat.

Dès que l'on augmente le taux de 1° soit 13° la concordance graphique n'est plus présente bien que le taux de discordance soit encore convenable (7.05%), cela se vérifie graphiquement :



Illustration 31 : Discordance pour un seuil de 13°

On remarque que les lignes du graphique optimisé (en rouge) ne suivent pas parfaitement celle du graphique initial (en bleu) ce qui est particulièrement vrai dans les boucles.

Afin de valider l'analyse du programme et les hypothèses précédentes, j'ai effectué des tests sur un deuxième parcours constitué de 1433 points dont voici les résultats :

Seuil	Nombre points optimisés	Pourcentage de compression	Nombre points discordants	Pourcentage discordance	Ratio
0°	852	40,54%	0	0%	N.C
5°	700	51,15%	147	11,03%	4,64
12°	603	57,92%	238	17,66%	3,27

On constate que le pourcentage de compression est relativement élevé et que le ratio entre le pourcentage de compression et celui de discordance reste le plus élevé pour un seuil de 5°.

Glossaire et mots clefs

AFRATA	Association Française de TéléAssistance
APA	Allocation Personnalisée d'Autonomie : prestation nationale distribuée par les départements
DREES	Direction de la Recherche, des Études, de l'Évaluation et des Statistiques
GIR	divers degré de dépendance, mesurés selon une nomenclature officielle, les GIR (groupe iso-ressources) de GIR 1 (le plus dépendant) à GIR 6. GIR1 à 4 sont les niveaux des personnes dépendantes pouvant prétendre à l'APA
Prévalence	Mesure de l'état de santé d'une population à un instant donné
SIG	Un système d'information géographique est un outil permettant d'organiser et présenter des données spatialement référencées
API	Une interface de programmation (<i>Application Programming Interface</i>) est un ensemble de fonctions, procédures ou classes mises à disposition des programmes informatiques
NMEA	La norme NMEA est une spécification pour la communication entre équipements marins. Elle est définie et contrôlée par la National Marine Electronics Association (NMEA)
VBA	Visual Basic for Applications (VBA) est une implémentation de Microsoft Visual Basic qui est intégrée dans toutes les applications de Microsoft Office
GPS	Le Global Positioning System que l'on peut traduire en français par « Géo-Positionnement par Satellite »
RMC	Recommend Minimum sentence C est une des trames principales d'un fichier NMEA
UTC	Le Temps universel coordonné est une échelle de temps adoptée comme base du temps civil international
Visual Studio	Microsoft Visual Studio est une suite de logiciels de développement pour Windows conçu par Microsoft
.NET	Nom d'un ensemble de produits et de technologies informatiques de Microsoft pour rendre ses applications portables ou facilement accessibles par Internet.
Checksum (ou somme de contrôle)	Concept de la théorie des codes utilisé pour les codes correcteurs
PDA	Personal Digital Assistant est un assistant personnel ou ordinateur de poche soit un appareil numérique portable
Port COM	Port informatique particulier destiné à la communication

Références bibliographiques

1 INSEE

Titre Enquête et études
 démographiques de
 l'INSEE

*Voir p. **Error! Bookmark not defined.***

2 DREES

Titre Études et Résultats

*Voir p. **Error! Bookmark not defined.***

3 LA PRISE EN CHARGE DES PERSONNES AGÉES DÉPENDANTES

Titre La prise en charge
 des personnes âgées
 dépendantes dans
 leur dernière
 période de vie

Auteurs Commissariat général
 au Plan (2005)

4 LE MARCHÉ DE LA DÉPENDANCE

Titre Le marché de la
 dépendance des
 personnes âgées

Auteurs Candesic (2007)
Voir p. 2 - 4

5 ANNEXE TECHNIQUE DGE

Titre Bracelet Électronique
 pour l'Autonomie –
 Annexe technique

Auteurs P.PEDIRODA
 P.MIETLICKI (2009)
*Voir p. **Error!***
Bookmark not defined.

5

6 ACCUEIL DE PERSONNES ÂGÉES

Titre Accueil de personnes
 âgées

Auteurs Étude Xerfi (2008)
Voir p. 3

Références webographiques

1 GPSINFORMATION.ORG

Adresse du site web :

<http://www.gpsinformation.org/dale/nmea.htm>

2 WIKIPEDIA

Adresse du site web :

<http://fr.wikipedia.org/wiki/NMEA>

Voir p. 6-7

3 CODEPEDIA

Adresse du site web :

<http://www.codepedia.com/1/The+GPRMC+Sentence>

4 AVIATION FORMULARY

Adresse du site web :

<http://williams.best.vwh.net/avform.html>

5 MOVABLE TYPE

Adresse du site web :

<http://www.movable-type.co.uk/scripts/latlong.html>

6 DIFFERENT DEGREE FORMATS

Adresse du site web :

http://home.online.no/~sigurdhu/Deg_formats.htm

7 TMA.DK

Adresse du site web :

<http://www.tma.dk/gps/>

8 NMEA OUTPUT STRUCTURE

Adresse du site web :

http://bse.unl.edu/adamchuk/web_ssm/

9 CODEPROJECT – LOCATION FROM CELL ID

Adresse du site web :

<http://www.codeproject.com/KB/mobile/DeepCast.aspx#5>

10 GPS VISUALIZER

Adresse du site web :

<http://www.gpsvisualizer.com/>

11 PRUNE GPS

Adresse du site web :

<http://activityworkshop.net/software/prune/>

Illustrations

Illustration 7 : exemple de trajet GPS piéton	10
Illustration 8 : illustration de l’algorithme de suppression des points GPS	11
Illustration 9 : Résultats trajets routiers optimisés	12
Illustration 10: Portion de l'algorithme d'optimisation en VBA	13
Illustration 11 : Bearing incorrect dans les trames RMC.....	13
Illustration 12 : Calcul du “bearing” (ou direction) en VBA	14
Illustration 13 : Comparaison avec GPS visualizer (gauche : initial, droite : optimisé).....	14
Illustration 14 : Interface graphique sous VBA	15
Illustration 15 : Menu graphique pour l’optimisation en VBA.....	16
Illustration 16 : Menu d’aide de l’interface graphique en VBA	17
Illustration 17 : Diagramme de classes du module “Location”	21
Illustration 18 : Onglet “Informations” du programme PDA	22
Illustration 19 : Code pour la mise à jour des « TextBox » de latitude et longitude en C#.....	23
Illustration 20 : Portion de code pour la mise à jour de la liste des positions GPS en C#.....	23
Illustration 21 : Onglet “Logs” du programme PDA	24
Illustration 22 : Portion de code pour la mise à jour de la taille des fichiers de « Logs » en C#.....	24
Illustration 23 : Optimisation en temps réel en C# sous .NET	25
Illustration 24 : Utilisation de la classe “Optim” en C#.....	26
Illustration 25 : Onglet “Signal” du programme PDA	26
Illustration 26 : Portion de code pour la generation des signaux satellites en C#.....	27
Illustration 27 : Onglet “Boussole” du programme PDA	27
Illustration 28 : Affichage de la boussole et positionnement du point de déplacement en C#	28
Illustration 29 : Onglet “Route” du programme PDA.....	29
Illustration 30 : Partie du code pour “centrer” la carte au niveau du parcours en C#.....	29
Illustration 31 : Affichage du trajet piéton sur PDA avec SharpMap	30
Illustration 32 : Diagramme de classes du module “GPS Tracker”	31
Illustration 33 : Analyse d’un trajet GPS avec Prune GPS sous Windows XP	35
Illustration 34 : Choix de la couleur du trajet et récupération des points discordants en Java	36
Illustration 35 : Affichage de 2 trajets avec Prune modifié (en bleu le trajet de référence)	36
Illustration 36 : Présence d’une bonne concordance pour un seuil de 12°	38
Illustration 37 : Discordance pour un seuil de 13°	39
Illustration 38 : Environnement	53
Illustration 39 : Fonctions	54

BéA



Projet BÉA - Annexes

Bracelet Électronique pour l'Autonomie

Ce rapport de stage présente le travail effectué côté technique lors du stage à Atos Origin Intégration de mars à août 2009 dans le cadre de la double formation (INSA-IAE). Il comprend notamment les comptes rendus de l'ensemble des gammes développés dans le cadre du projet BEA.

**Pascal MIETLICKI – Master 2 management de l'innovation –
5^{ème} année génie informatique INSA**

29/08/2009



Table des matières

A.	Spécification pour la PAM	48
1.	introduction	50
1.1	Objet du document	50
1.2	Champ d'application	51
1.3	Lectorat	51
2.	documentation, terminologie et sigles	52
2.1	Documents applicables	52
2.2	Documents de référence	52
2.3	Terminologie et sigles	52
3.	Description générale	52
3.1	Vue d'ensemble des fonctionnalités du produit	53
3.2	Méthodes et outils	54
3.3	Description du système PAM	54
3.4	Caractéristiques des utilisateurs	55
3.5	Contraintes	55
3.6	Description de l'utilisation du système	56
3.7	Scénarios optionnels	57
3.8	Environnement d'exploitation	61
4.	Exigences	61
4.1	Répartition des exigences	62
4.2	Exigences fonctionnelles	62
4.3	Exigences de performance	64
4.4	Contraintes de conception	64
B.	Documents de présentation	66
1.	Présentation globale pour SWATCH	66
2.	Architecture globale	67
3.	Projet BEA version 1	68
4.	Projet BEA version 2	68

A. Spécification pour la PAM



- >> AMELIORER LA PERFORMANCE
- >> AUGMENTER LA SOUPLESSE
- >> ASSURER LA TRANSPARENCE
- >> REDUIRE LES COUTS
- >> AMELIORER LA RELATION CLIENT
- >> ACCELERER LA MISE SUR LE MARCHE
- >> INNOVER
- >> AMELIORER L'EFFICACITE
- >> ACCROITRE L'ADAPTABILITE
- >> GARANTIR LA CONFORMITE

BEA

Document d'exigences pour la PAM

Client	:	None
Affaire/Projet	:	BEA
Référence	:	GNSS-BEA-RQ-ATOS
Révision	:	1.0
Etat	:	Draft
Date	:	21/04/09
Classification	:	Unclassified

Nombre d'annexes: 1

Circuit de validation

	NOM	FONCTION (SOCIETE)	DATE	VISA
AUTEUR		Equipe BEA	21/04/09	
VERIFICATEUR				
APPROBATEUR				

Diffusion

NOM	SOCIETE	FONCTION

Historique des révisions

REVISION	DATE	MODIFICATION (NATURE ET CHAPITRES CONCERNES)	ORIGINE	AUTEUR
1.0	29/04/09	Première version pour DDKP	AOI	P.MIETLICKI
1.1	26/05/09	Deuxième version après réunion	AOI	P.MIETLICKI

1. INTRODUCTION

1.1 Objet du document

Le présent document spécifie les exigences logicielles pour la PAM (Plateforme d'Assistance Médicalisée), vue comme une boîte noire munie d'interfaces externes. Il exclut, cependant, les exigences relatives au protocole de communication de ces interfaces (format et synchronisation d'échanges des données) ; ces exigences sont plutôt couvertes par des « spécifications d'exigences d'interface ».

La PAM (Plateforme d'Assistance Médicalisée) est un des composants critiques du système BEA (Bracelet Electronique pour l'Autonomie).

Dans la suite de ce document, on emploiera indistinctement les termes « PAM », « plateforme » ou « La PAM » pour désigner la Plateforme d'Assistance Médicalisée nécessaire au bracelet BEA.

L'objectif de ce document est de traduire la spécification de besoin en exigences logicielles permettant de conduire à une phase de conception informatique. L'ensemble des aspects relatifs à ce composant logiciel est décrit (directement dans ce document ou via des pointeurs vers d'autres documents) au travers des aspects fonctionnels (cas d'utilisation), matériels, logiciels, sécurité ou encore contraintes de développement.

Au delà de la présentation générale du composant et de son rôle dans BEA (Bracelet Electronique pour l'Autonomie), ces spécifications logicielles sont déclinées en :

- Spécifications fonctionnelles : définition des fonctions (cas d'utilisation) que doit assurer le composant,
- Exigences d'architecture : description de l'environnement matériel et logiciel dans lequel le composant devra s'intégrer et identification des éléments majeurs d'architecture logicielle du composant (processus, bibliothèques, COTS),
- Spécifications opérationnelles : définition des conditions de mise en œuvre opérationnelle du composant,
- Contraintes de sécurité : identification des contraintes logicielles induites par la sécurité et que doit supporter le composant.
- Contraintes de performance : identification des contraintes de performances liées au composant.
- Contraintes de développement : contraintes particulières de développement agissant sur le composant.

Ce document s'appuie sur le descriptif détaillé présenté notamment au 7^{ème} FUI.

Le présent document s'adresse au client, au concepteur et au responsable des essais du système.

NOTES :

1) Le client doit particulièrement vérifier que le présent document couvre bien tout son besoin pour l'exploitation et l'entretien du système : la validité du système livré repose uniquement sur les exigences contenues dans les « spécifications d'exigences ».

2) Ce type de document fait abstraction de l'architecture et du fonctionnement interne du système ; ces aspects sont plutôt couverts par les « spécifications de conception ».

3) Dans le présent document, les verbes « devoir », « convenir » et « pouvoir » ainsi que le futur sont utilisés dans les sens très précis suivants : le verbe « devoir » est utilisé pour exprimer une obligation qui lie une partie contractante à une ou plusieurs autres ; le futur exprime une déclaration d'intention faite par une partie ou ce qu'elle se propose de faire ; l'expression « il convient de » exprime une recommandation parmi d'autres possibilités ; le verbe « pouvoir » exprime une liberté d'action dans le cadre du présent document.

1.2 Champ d'application

Le bracelet BEA a pour objectif principal d'améliorer le quotidien des personnes âgées dépendantes en les protégeant dans leurs déplacements quotidiens et en étant capables de prévenir les personnes compétentes dans tous types de situations dangereuses (accident, chute, panique...).

Le produit à développer est une plateforme permettant de traiter et de qualifier les alertes provenant du bracelet BEA. Cette plateforme doit aussi produire des rapports détaillés afin d'anticiper des situations potentiellement critiques (chutes, fugues...) et renseigner le personnel soignant pour qu'ils puissent établir un diagnostic précis.

Une description générale de l'application et de ses fonctions est décrite dans le document de référence /1/.

1.3 Lectorat

Le présent document se compose des chapitres suivants :

1. INTRODUCTION :

Ce chapitre fournit une vue d'ensemble du présent document. Il en décrit :

L'objet ;

La portée ;

2. DOCUMENTATION, TERMINOLOGIE, SIGLES

Les définitions, acronymes et abréviations ;

Les références ;

3. DESCRIPTION GÉNÉRALE :

Ce chapitre décrit les facteurs généraux qui affectent le système et ses exigences. Il n'énonce aucune exigence spécifique. Il fournit plutôt le contexte de ces exigences, définies en détail au chapitre 3, pour en faciliter la compréhension. Il décrit sommairement :

L'environnement du système ;

Les fonctions du système ;

Les caractéristiques des utilisateurs ;

Les contraintes ;

Les hypothèses et dépendances.

4. EXIGENCES SPÉCIFIQUES :

Ce chapitre contient les exigences spécifiques au système et ce, à un niveau suffisamment détaillé permettant aux concepteurs de concevoir un système répondant à ces exigences, et aux vérificateurs de vérifier que le système répond à ces exigences.

Il formule les exigences d'un point de vue extérieur au système, d'un point de vue des utilisateurs ou des systèmes externes.

Au minimum, il décrit tous les intrants du système, tous les extrants du système et toutes les fonctions exécutées par le système en réaction aux intrants, pour soutenir ses extrants.

Il décrit :

Les exigences fonctionnelles ;

Les exigences de performance ;

Les contraintes de conception ;

Les attributs (disponibilité, sécurité, ...)

2. DOCUMENTATION, TERMINOLOGIE ET SIGLES

2.1 Documents applicables

Les documents applicables sont ceux qui contiennent des exigences imposables au domaine auquel se réfère le présent document et dont la satisfaction doit pouvoir être vérifiée.

REFERENCE, REVISION	TITRE	EMETTEUR	DATE

2.2 Documents de référence

Les documents de référence sont les documents qui ont été utilisés pour la rédaction du présent document, sans toutefois contenir d'exigences imposables au domaine auquel se réfère le présent document.

REFERENCE, REVISION	TITRE	EMETTEUR	DATE
2.3	Annexe technique présentée à la DGE	PM	31/03/09

2.3 Terminologie et sigles

TERME / SIGLE	DEFINITION

3. DESCRIPTION GENERALE

Le projet de recherche BÉA se propose de donner les moyens pour avvertir les aidants dès l'amorce d'une situation critique, et de sécuriser ainsi l'environnement du malade de façon responsable, sans le contraindre ou l'isoler.

Un bracelet montre, porté par la personne dépendante, peut détecter des situations critiques et émettre des alertes à une Plateforme d'Assistance Médicalisé 24h/24, 7 jours/7. En fonction des protocoles opérationnels définis, les aidants (famille, service d'aide à domicile, voisins) ou les services professionnels de gestion de l'urgence (SAMU, Pompiers, Police) sont avvertis afin de traiter au plus tôt la situation critique.

La PAM est un composant présentant un couplage fort au sein du système. Elle est conçue pour pouvoir fonctionner de manière autonome du reste du système à condition d'être alimenté en données d'entrée.

La principale mission de la PAM est de traiter et de qualifier les données envoyées par le bracelet. Pour répondre à cet objectif, la PAM assurera :

- la réception et l'analyse des informations provenant du bracelet (notamment traitement des alertes en provenance du bracelet pour indiquer un niveau d'alerte s'il y a lieu)
- le stockage des données reçues
- le calcul et traitement des données stockées afin de générer des alarmes vers les personnes compétentes (aidants, médecins, opérateurs...)
- le traitement et la mise à jour des zones de geofencing en provenance du et vers le bracelet

Une des principales missions de la PAM est de superviser le bracelet, elle doit assurer qu'un bracelet fonctionne correctement, que la liaison est correcte, que le bracelet est allumé et répond aux requêtes. Elle doit pouvoir générer des alertes en cas de problème de liaison avec le bracelet.

La PAM doit notamment mettre à jour les zones de geofencing en analysant les données provenant du bracelet (position, données physiologiques...). Elle doit être en mesure d'éviter les fausses alertes. Cependant, la PAM ne doit absolument pas omettre une alerte véritable car les conséquences pourraient être dramatiques aussi bien au niveau humain qu'au niveau économique.

La PAM doit fonctionner 24 heures par jour et représente un composant critique du système BEA. Dans ce contexte la PAM doit être disponible et remplir ces fonctions à tout moment. Elle doit aussi être au sein d'une architecture répliquée permettant de basculer sur un relais en cas de problème. Toutes les fonctions de la PAM doivent être interchangeables sans interruption de services, la moindre seconde perdue pouvant avoir des effets dramatiques et des conséquences désastreuses. Les performances doivent être les meilleurs possibles et, ainsi, la PAM doit pouvoir traiter les données avec une grande efficacité.

La PAM va traiter des données sensibles et il est donc indispensable d'assurer la confidentialité et le respect de ces informations. Toutes communications entre la PAM et un composant extérieur doit être cryptée de manière fiable.

D'autre part, les données de la PAM doivent pouvoir être affichées par une IHM afin d'aider au diagnostic du personnel soignant et aussi, si nécessaire, pour localiser une personne en fugue sur une carte.

La figure suivante illustre l'environnement du système, mettant en évidence ses interfaces externes.

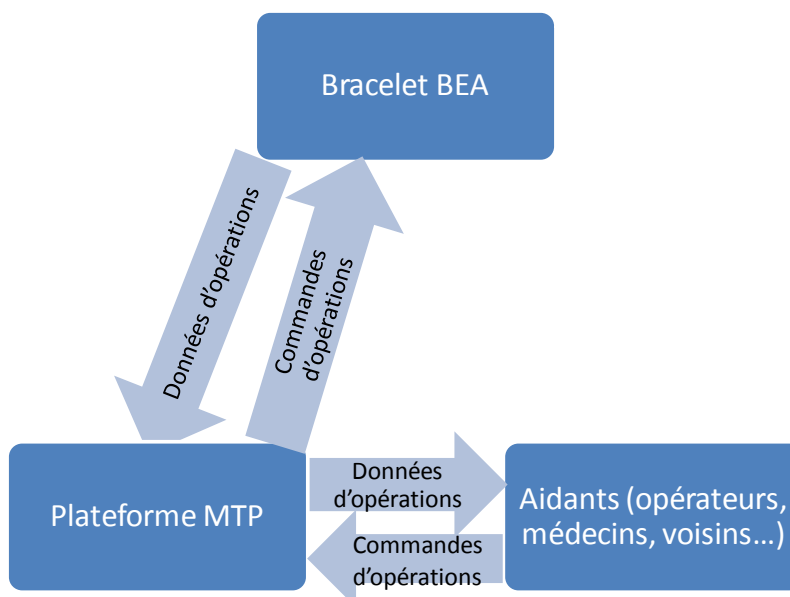


Illustration 32 : Environnement

3.1 Vue d'ensemble des fonctionnalités du produit

La figure suivante illustre les fonctions du système. Dans ce document, nous nous focalisons sur les fonctionnalités offertes par la PAM qui sont détaillées ci-après.

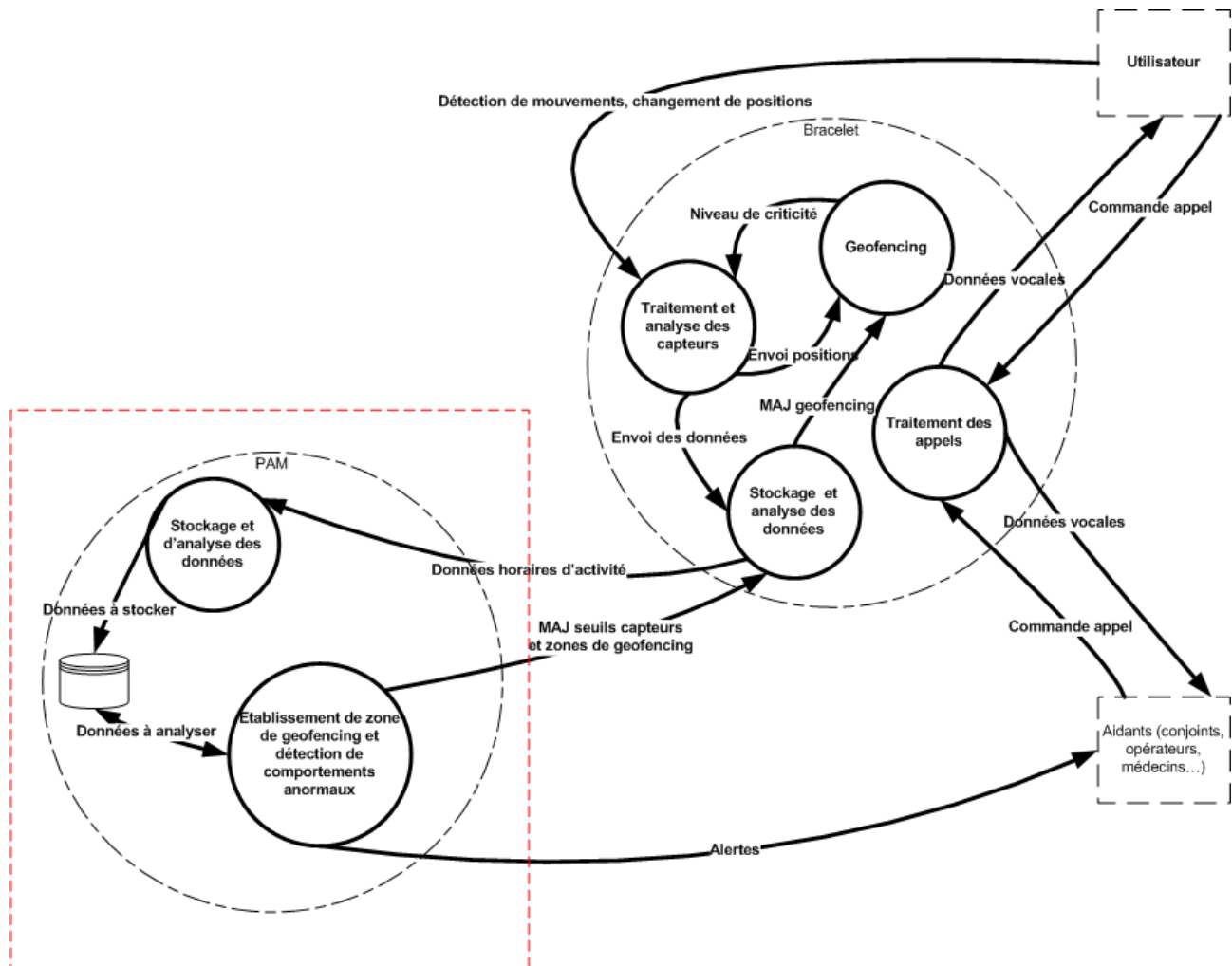


Illustration 33 : Fonctions

3.2 Méthodes et outils

La conception de la PAM sera faite en UML. L'outil UML utilisé pour ce projet est Microsoft Visio.

TBC : Le langage imposé et utilisé pour la conception est le langage Java.

3.3 Description du système PAM

Une des principales missions de la PAM est de vérifier l'intégrité des alertes générées par le bracelet. Elle doit qualifier ces alertes en indiquant un niveau d'urgence afin que l'opérateur puisse prendre les mesures nécessaires.

Elle doit aussi pouvoir mettre à jour les zones de geofencing et prévenir des situations potentiellement dramatiques (chute, fugues...) pour la personne portant le bracelet.

Tous les événements seront écrit dans une base de données et un rapport détaillé pourra être généré afin de visualiser facilement les informations à destination des médecins, de la famille...

Le format d'échange des données entre le bracelet et la PAM sera décrit dans un fichier XML.

3.3.1 Stockage et analyse des données

L'objectif de cette fonction est de stocker et traiter les données reçues. Cette fonction du système a donc un aspect critique. Elle doit être capable de stocker l'ensemble des données reçues par le bracelet permettant de sécuriser et de protéger la personne bénéficiaire.

Elle doit aussi sauvegarder toutes les données annexes de connexions des opérateurs et des aidants telles que les données de connexions, les données d'opération comme les requêtes pour remonter la position. En définitive, tous les logs du système.

Plus précisément, elle vise les objectifs suivants :

- 1) Opérations normales : analyse et stockage des données.
- 2) Opérations exceptionnelles : gérer les opérations autres que normales relatives à la présente fonction, telles que dépassement de la mémoire ou l'absence de message.

3.3.2 Etablissement de zones de geofencing et détection de comportements anormaux

Les objectifs de cette fonction est de traiter et qualifier les alertes en provenance du bracelet. En fonction du type d'alerte, un niveau de criticité est associé à cette alerte. Par exemple, une qualification ascendante de 0 à 5 (0 : la moins critique et 5 la plus critique).

D'autre part, cette fonction gère les zones de geofencing, en fonction des données reçues du bracelet (alertes et positions), elle va déterminer de nouvelles zones mieux adaptées (BEA v2). C'est une fonction critique du système car les zones de geofencing doivent rester pertinente et être établies avec un maximum de précautions.

Cette fonction va notamment pouvoir, dans la mesure du possible, détecter les comportements anormaux en analysant continuellement les données envoyées par le bracelet et présentes dans la base de données.

Les alertes qualifiées sont ensuite renvoyées à l'opérateur. **TBC** : toutes les alertes quelque soit leur niveau de criticité sont envoyées à l'opérateur.

- 1) Opérations normales : Sur envoi d'une alerte, analyse de cette alerte et ajout d'un niveau de criticité puis envoi vers l'opérateur. *Analyse continue des zones de geofencing pour établir des GAA intelligent en s'appuyant sur les données physiologiques. Analyse des données physiologiques afin d'essayer d'anticiper des situations dangereuses (BEA v2).*
- 2) Opérations exceptionnelles : gérer les opérations autres que normales relatives à la présente fonction, telles que des données incohérentes ainsi qu'un résultat anormal.

3.4 Caractéristiques des utilisateurs

3.4.1 Utilisateur

L'utilisateur ne possède :

- 1) Aucune formation particulière ;
- 2) Aucune expérience particulière ;
- 3) Aucune connaissance technique particulière.

3.5 Contraintes

3.5.1 Performance

L'objectif de ces contraintes est de satisfaire des exigences numériques. Plus précisément, elle vise les objectifs suivants, pour le système :

- 1) Statiques : supporter plusieurs centaines de messages ;
- 2) Dynamiques : offrir des temps de réponse de l'ordre de la seconde. Car, dans les cas les plus critiques, plus les secours arriveront vite sur place plus les chances de survie seront élevées.

3.5.2 Conception

L'objectif de ces contraintes est de satisfaire des contraintes de conception. Plus précisément, elle vise les objectifs suivants, pour le système :

- 1) Contraintes de conformité : respecter des normes de rédaction et de programmation ;
- 2) Contraintes techniques : être de dimension analogue à une montre standard et donc trouver les composants les plus réduits possibles, avoir une consommation d'énergie minimale;
- 3) Contraintes d'intégrité : renvoyer des données cohérentes ;
- 4) Contraintes écologiques : limiter la consommation d'énergie ;
- 5) Contraintes de légalité : établir des clauses concernant la responsabilité en cas de la non génération d'alertes (faux négatif), étudier les clauses de confidentialité des données afin de préserver l'intégrité et la dignité de l'utilisateur.

3.5.3 Attributs

L'objectif de ces contraintes est de satisfaire des exigences d'attribut : disponibilité, sécurité, maintenabilité, ... Plus précisément, elles visent les objectifs suivants, pour le système :

- 1) Disponibilité : démarrer automatiquement suite à la mise sous tension ;
- 2) Sécurité : générer des données véridiques et cohérentes, limiter l'accès à l'interface et la possibilité d'enlever le bracelet au personnel de support ;
- 3) Maintenabilité : être mis sous gestion de la configuration ;
- 4) Fiabilité : supporter le traitement de plusieurs centaines de messages en quelques dizaines de secondes sans défaillances ;
- 5) Transférabilité : isoler les parties du code logiciel lié à des parties pouvant changer, de l'environnement ou de la plate-forme, dans des modules dédiés.

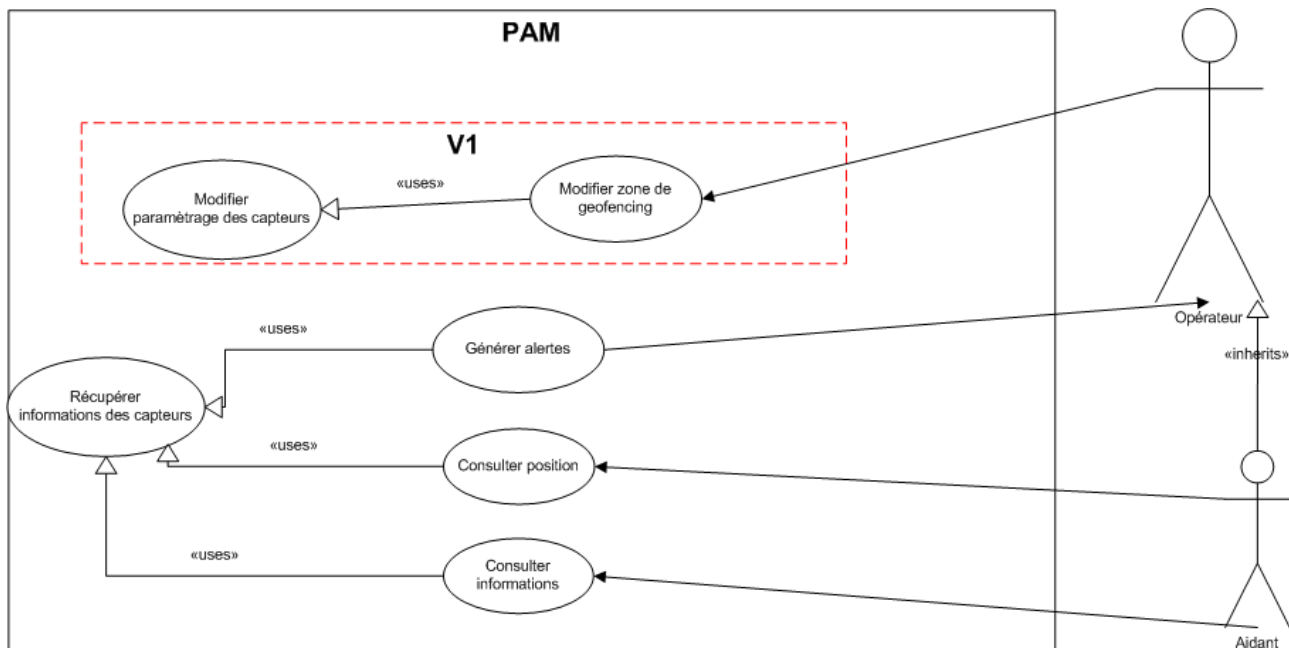
3.6 Description de l'utilisation du système

Les acteurs impliqués dans l'utilisation sont :

- Un opérateur chargé de récupérer les alertes et d'appliquer des procédures en fonction du niveau de criticité.
- Un aidant qui peut venir en aide à la personne bénéficiaire du bracelet, ça peut être un voisin, un conjoint, un médecin ou une infirmière par exemple.
- Le bracelet qui génère des alertes (par détection d'une situation anormale pour la personne)

Il est important de préciser qu'un aidant aura différents droits en fonction de son statut, s'il est juste un voisin, il n'aura nullement le droit de pouvoir accéder aux informations confidentielles concernant la personne ce qui n'est pas le cas s'il est de la famille ou si c'est un médecin.

Les cas d'utilisation identifiés sont :



3.7 Scénarios optionnels

Les cas d'utilisations sont listés ci-dessous :

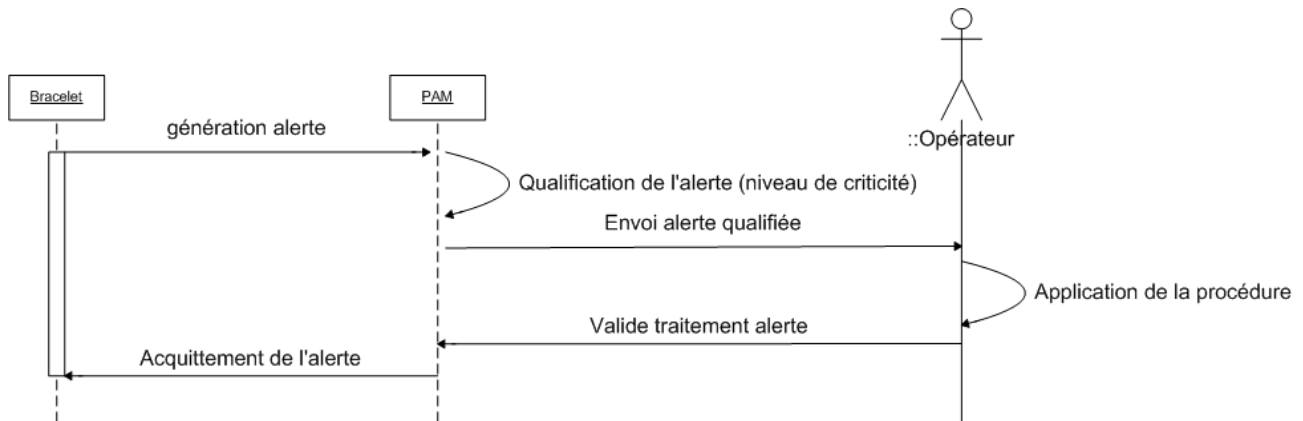
Id	Label	Acteur principal	Complexité
BEA-QT-UC-0	Génération d'alertes	Bracelet	Haute
BEA-QT-UC-1	Consulter position	Aidant	Moyenne
BEA-QT-UC-2	Consulter informations	Aidant	Moyenne
BEA-QT-UC-3	Modifier zones de geofencing	Opérateur	Moyenne

3.7.1 BEA-QT-UC-0: Génération d'alertes

Nom :	BEA-QT-UC-0 : Génération d'alertes
Description :	Une alerte émise par le bracelet vers la PAM est ensuite qualifiée et transmise vers l'opérateur
Acteurs :	Bracelet, Opérateur
Pré-condition :	Bracelet BEA installé et fonctionnel. Les composants de géolocalisation sont présents et fonctionnent correctement
Description de la sequence :	<ul style="list-style-type: none"> Le bracelet détecte une situation dangereuse (chute, sortie de la zone de geofencing...) La plateforme reçoit l'alerte La plateforme traite l'alerte en la qualifiant Si l'alerte n'est pas grave (données physiologiques normales par exemple), l'alerte est qualifiée au niveau le moins grave <i>Si l'alerte n'est pas grave et provient d'une sortie de zone, la zone de geofencing est</i>

	<p><i>mise à jour grâce au système GAA (BEA v2)</i></p> <ul style="list-style-type: none"> L'opérateur reçoit l'alerte et doit exécuter la procédure nécessaire pour la traiter (appel pour s'assurer que la personne va bien...)
Post-conditions :	L'alerte a été vérifié. Un rapport est produit. La base de données est mise à jour avec les données provenant de l'alerte.

Diagramme de séquence du cas d'utilisation BEA-QT-UC-0 :



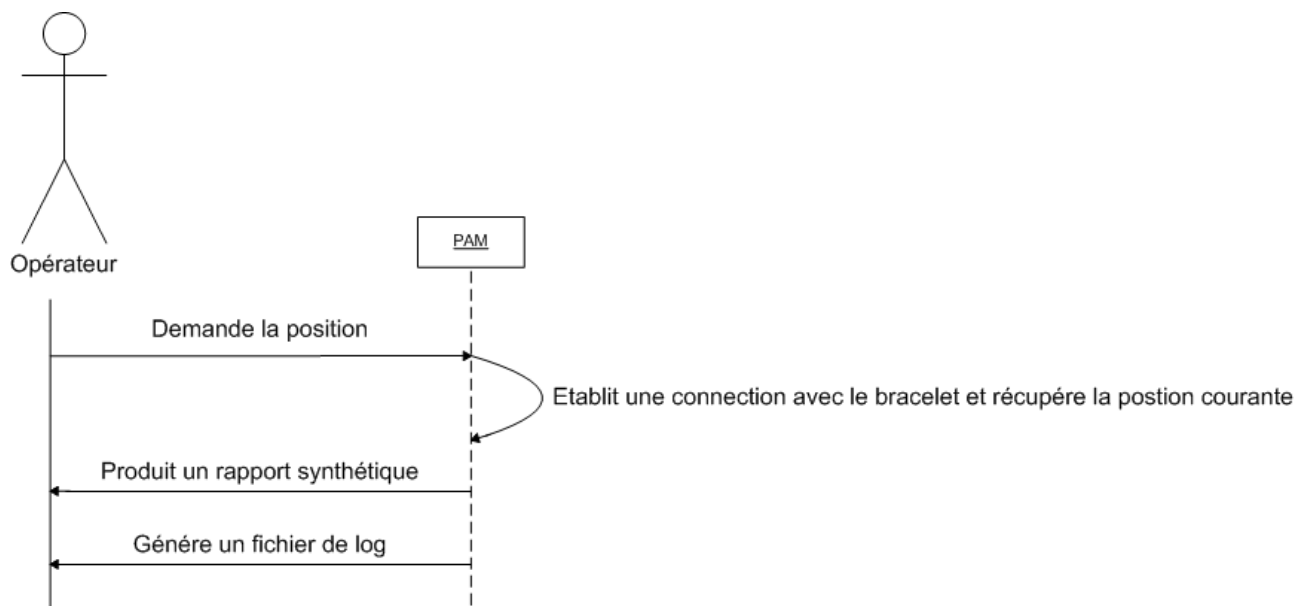
TBC : Si l'alerte n'est pas grave, envoi des informations au système GAA qui renvoie une nouvelle zone de geofencing et mise à jour de cette zone sur le PDA (BEA v2)

TBC : Envoie de l'alerte à l'opérateur dans tous les cas (même si l'alerte n'est pas d'une grande criticité)

3.7.2 BEA-QT-UC-1: Consulter position

Nom :	BEA-QT-UC-1 : Récupérer la position de la personne
Description :	L'aidant effectue une demande afin de récupérer la position de la personne.
Acteur :	Aidant
Pre-conditions :	Bracelet BEA installé et fonctionnel. Les composants de géolocalisation sont présents et fonctionnent correctement.
Description de la séquence :	<ul style="list-style-type: none"> L'aidant s'identifie Si l'aidant a les droits suffisants, l'aidant peut demander la position de la personne à la PAM Le bracelet reçoit l'ordre et envoie l'information à la PAM La PAM renvoie la position de la personne sur une carte
Post-conditions :	Un rapport synthétique est produit. Les logs sont sauvegardés dans la base de données et contiennent les opérations effectuées.

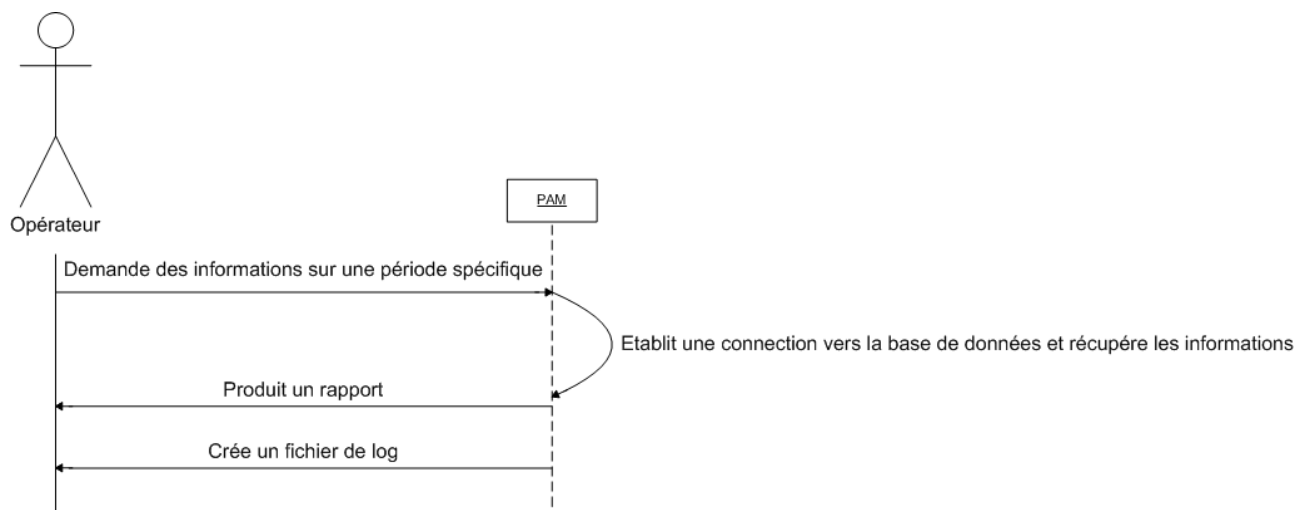
Diagramme de séquence du cas d'utilisation BEA-QT-UC-1 :



3.7.3 BEA-QT-UC-2: Consulter informations

Name :	BEA-QT-UC-2 : Consulter informations
Description :	L'aidant demande un rapport détaillé sur la personne (par heure, jours, mois ou année) dans le but d'établir un diagnostic précis
Acteur :	Aidant
Pre-conditions :	Bracelet BEA installé et fonctionnel. Les composants de géolocalisation sont présents et fonctionnent correctement.
Description de la séquence :	<ul style="list-style-type: none"> • L'aidant s'identifie • Si l'aidant a les droits suffisants, l'aidant peut demander des informations détaillées de la personne à la PAM • La PAM récupère et vérifie les données dans la base de données • La PAM génère un rapport avec les données récupérées
Post-conditions :	Un rapport est produit. Les logs sont sauvegardés dans la base de données et contiennent les opérations effectuées.

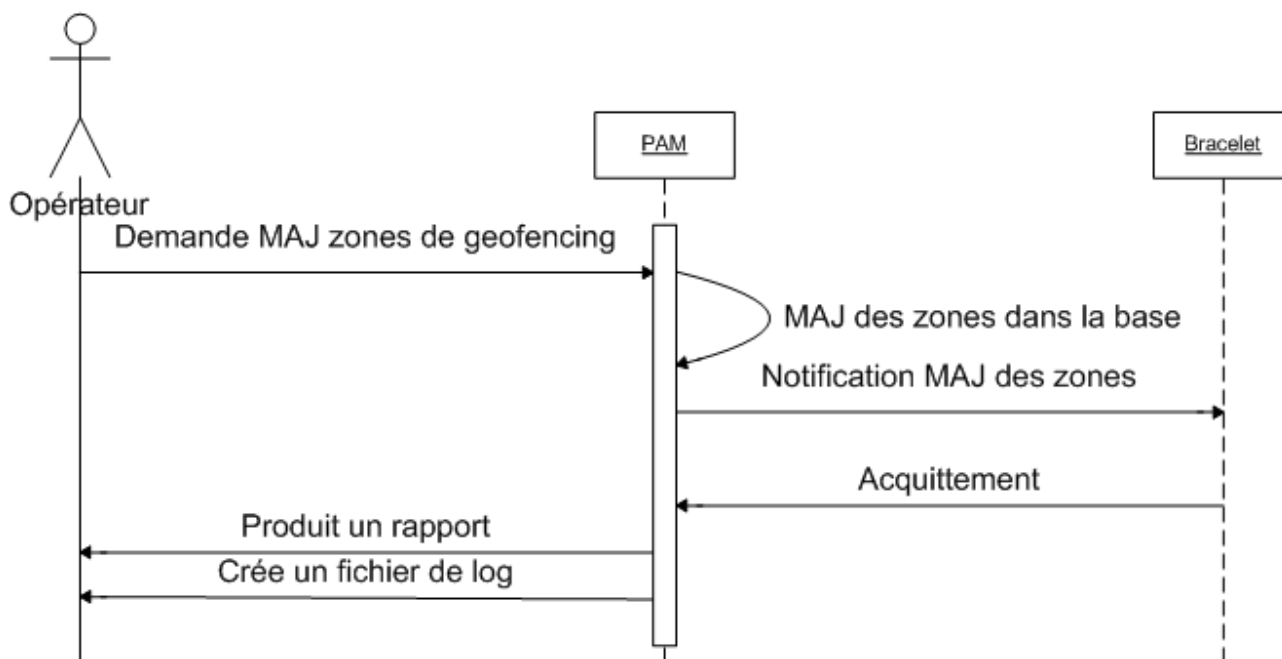
Diagramme de séquence du cas d'utilisation BEA-QT-UC-2 :



3.7.4 BEA-QT-UC-3: Modifier zones de geofencing

Name :	BEA-QT-UC-2 : Modifier zones de geofencing
Description :	L'opérateur en accord avec le personnel soignant et les aidants peut mettre à jour les zones de geofencing de la personne bénéficiaire du bracelet.
Acteurs :	Opérateurs et aidants
Pre-conditions :	Bracelet BEA installé et fonctionnel. Les composants de géolocalisation sont présents et fonctionnent correctement.
Description de la séquence :	<ul style="list-style-type: none"> • L'opérateur s'identifie • Si l'opérateur a été invité par un aidant (médecin ou autres) à mettre à jour la zone de geofencing sur une plage horaire spécifique, l'opérateur devra alors confirmer cette mise à jour. • La PAM récupère et met à jour les données dans la base de données • La PAM envoie une requête au bracelet pour le notifier de la mise à jour des nouvelles zones de geofencing
Post-conditions :	Un rapport est produit. Les logs sont sauvegardés dans la base de données et contiennent les opérations effectuées.

Diagramme de séquence du cas d'utilisation BEA-QT-UC-3 :



TBC : Confirmation nécessaire au préalable par un médecin et/ou un aidant (conjoint, famille...)

3.8 Environnement d'exploitation

3.8.1 Environnement matériel

Configuration pour le développement :

Processeur « x86 » avec Windows ou Linux

Operational configuration :

Accès au serveur MTP de Thales Alenia Space

3.8.2 Environnement logiciel

Eclipse et les plugins pour faire des Web Services (Spring) et les plugins pour la gestion des dépendances telles que Maven

Serveur Tomcat

3.8.3 Dépendances

La PAM utilisera les services fournis par la plateforme MTP notamment concernant l'accès à la base de données (hibernate...)

L'accès aux données de la plateforme et le développement des outils principaux sera fait par l'intermédiaire de Web services

Pour que la plateforme puisse exploiter des données concrètes et correctes, le bracelet doit fonctionner correctement et envoyer les données nécessaires.

4. EXIGENCES

Ce chapitre liste les exigences spécifiques. Les sous-sections sont regroupées autour de fonctions haut niveaux.

4.1 Répartition des exigences

Les fonctionnalités primaires concernent le fait que le système doit être capable de renvoyer des alertes avérées en cas de problème et aussi n'en omettre aucune. Ces alertes étant renvoyées dans un temps fini de l'ordre de quelques secondes.

4.2 Exigences fonctionnelles

E_BEAFUNC_0

Tout échange de données doit être fait au sein d'une connexion sécurisée et donc cryptées.

E_BEAFUNC_1

La PAM, en tant que composant critique, doit être opérationnel 24h/24.

4.2.1 BEA-QT-UC-0: Génération d'alertes

E_BEAFUNC_010

Le système PAM devra traiter toutes les alertes en provenance du bracelet. Aucune alerte ne doit être ignorée ou mal traitée.

E_BEAFUNC_020

TBC : Toute alerte, quelque soit son niveau de criticité, doit être indiquée à l'opérateur.

E_BEAFUNC_030

Tout échange de données doit être sécurisé. Les alertes émises par la PAM vers l'opérateur doivent rester confidentielles et être cryptées.

E_BEAFUNC_040

L'algorithme permettant d'affecter un niveau de criticité sur chaque alerte doit être fiable et ne doit pas considérer une alerte potentiellement dangereuse pour la personne comme une alerte non critique.

E_BEAFUNC_050

TBC : Si l'alerte n'est pas traitée par l'opérateur dans un temps fini, l'alerte sera transmise automatiquement aux aidants et/ou, selon la criticité, au personnel soignant.

E_BEAFUNC_060

TBC : L'alerte doit automatiquement déclencher une procédure de tracking de la personne pour s'assurer de son emplacement et donc de sa sécurité à tout moment.

4.2.2 BEA-QT-UC-1: Consulter position

E_BEAFUNC_070

Tout accès aux données de la personne doit garantir, au maximum, la confidentialité et le respect de la vie privée des personnes. Toute connexion doit être sécurisée et seules les personnes habilitées pourront avoir accès aux informations personnelles.

E_BEAFUNC_080

La personne bénéficiaire ou un responsable légal aura d'abord indiqué qu'elle accepte que l'on puisse voir sa position.

E_BEAFUNC_090

Tout échange de données doit être sécurisé. Les données émises par la PAM vers l'opérateur doivent rester confidentielles et être encryptées.

E_BEAFUNC_100

L'opérateur, sauf en cas de force majeure tel qu'une situation potentiellement dramatique, doit avoir l'accord préalable des aidants pour accéder aux informations confidentielles tel que la position.

E_BEAFUNC_110

La position renvoyée par la PAM doit être la plus précise possible et être directement renvoyée par le bracelet dans la mesure du possible.

4.2.3 BEA-QT-UC-2: Consulter informations

E_BEAFUNC_70

Tout accès aux données de la personne doit garantir, au maximum, la confidentialité et le respect de la vie privée des personnes. Toute connexion doit être sécurisée et seules les personnes habilitées pourront avoir accès aux informations personnelles.

E_BEAFUNC_80

La personne bénéficiaire ou un responsable légal aura d'abord indiqué qu'elle accepte que l'on puisse accéder à ses informations.

E_BEAFUNC_90

L'opérateur, sauf en cas de force majeure tel qu'une situation potentiellement dramatique, doit avoir l'accord préalable des aidants pour accéder aux informations confidentielles.

E_BEAFUNC_120

Les informations renvoyées par la PAM sont récupérées de la base de données et un rapport détaillé doit être renvoyé. **TBC** : Plusieurs niveaux de détails sont possibles en fonction du statut du demandeur (médecin, aidant...)

4.2.4 BEA-QT-UC-3: Mise à jour des zones de geofencing

E_BEAFUNC_70

Tout accès aux données de la personne doit garantir, au maximum, la confidentialité et le respect de la vie privée des personnes. Toute connexion doit être sécurisée et seules les personnes habilitées pourront avoir accès aux informations personnelles.

E_BEAFUNC_130

TBC : Toute mise à jour des zones de geofencing doit être, au préalable, validée par une personne compétente (médecin, infirmier...).

E_BEAFUNC_140

Une mise à jour des zones ne peut être faite que par une personne disposant des droits et des habilitations suffisantes. En aucun cas cette mise à jour ne doit être faite arbitrairement.

E_BEAFUNC_150

Une mise à jour des zones doit être prise en compte le plus rapidement possible par le bracelet. Le bracelet doit être notifié de chaque mise à jour directement par la PAM.

E_BEAFUNC_160

Chaque modification doit être sauvegardée dans la base de données. Les zones de geofencing doivent être mises à jour en parallèle dans la base de données et dans le bracelet.

4.2.5 Interface avec l'utilisateur

La PAM doit permettre d'empêcher les situations critiques en générant des alertes en cas de situations critiques sans n'en omettre aucune. Elle doit aussi permettre d'aider au diagnostic des médecins en agrégeant des informations sur le bénéficiaire et en les présentant au personnel soignant ou aux aidants lorsqu'ils se connectent. *D'autre part, les données collectées doivent permettre d'anticiper des situations critiques et générer des zones de geofencing auto-adaptatives (BEAv2).*

4.2.6 Données

Un protocole doit être défini afin de répondre à un transfert de données fiables et cryptés entre la plateforme et le bracelet.

4.2.7 Les scénarios de transaction

Le système doit renvoyer, à l'aide des informations fournies par le bracelet, les informations concernant la position, les données physiologiques provenant des capteurs (sudation, pouls, position)... ainsi que détecter et renvoyer les alertes.

4.3 Exigences de performance

La PAM doit satisfaire des exigences numériques.

4.3.1 Performances dynamiques

La PAM doit satisfaire des exigences numériques dynamiques.

4.3.1.1 Temps (dynamiques)

4.3.1.1.1 Temps de réponse général

La PAM doit offrir un temps de réponse général inférieur à 1 milliseconde, concernant la fourniture de tout extrant, sauf exigence contraire.

4.4 Contraintes de conception

La PAM doit satisfaire des contraintes de conception.

4.4.1 Contraintes de conformité

La PAM doit satisfaire des contraintes de conformité.

4.4.1.1 Rédaction

4.4.1.1.1 Norme de rédaction de l'entreprise

La PAM doit être documentée selon la norme de rédaction technique en vigueur dans l'unité administrative responsable de sa conception.

4.4.1.1.2 Gabarit de rédaction de l'entreprise

La PAM doit être documentée selon le gabarit de rédaction en vigueur dans l'unité administrative responsable de sa conception.

4.4.1.2 Programmation

4.4.1.2.1 Norme de programmation de l'entreprise

La PAM doit, si l'un de ses composants est logiciel, être programmée selon la norme de programmation en vigueur dans l'unité administrative responsable de sa conception.

4.4.2 Contraintes de légalité

La PAM doit satisfaire des contraintes de légalité.

4.4.2.1 Législation

Être conforme aux norms et lois en vigueur et avoir des échanges cryptés avec la plateforme de manière à respecter la vie privée et la dignité de l'utilisateur.

4.4.3 Fiabilité

La PAM doit satisfaire des exigences de fiabilité.

4.4.3.1 Erreurs / Défaillances

4.4.3.1.1 Fournitures sans défaillances

La PAM doit supporter la fourniture des informations sans défaillances.

4.4.3.2 Sauvegarde et recuperation

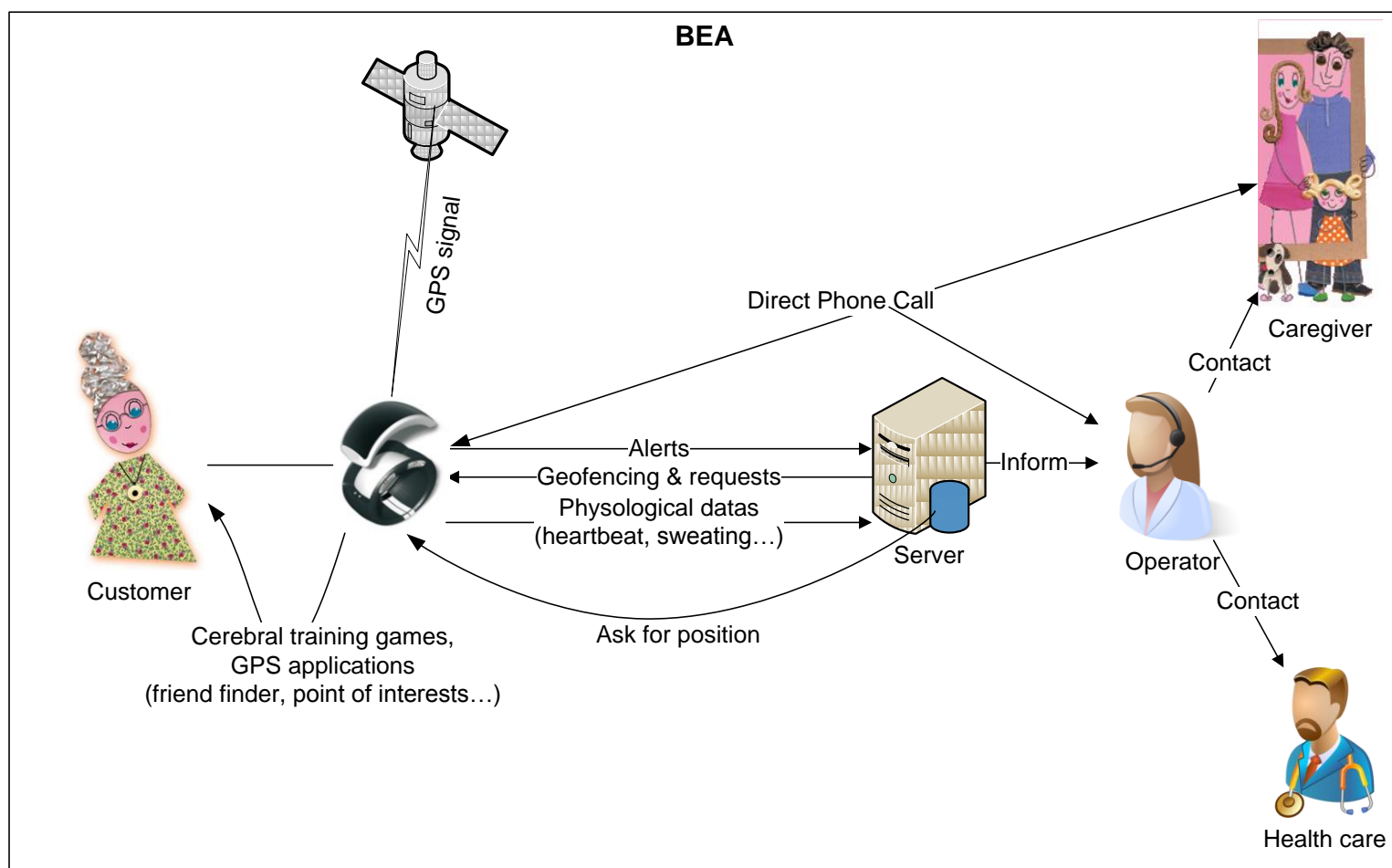
La PAM doit pouvoir sauvegarder et récupérer les informations notamment à des fins légales.

4.4.4 Exigences d'implémentations

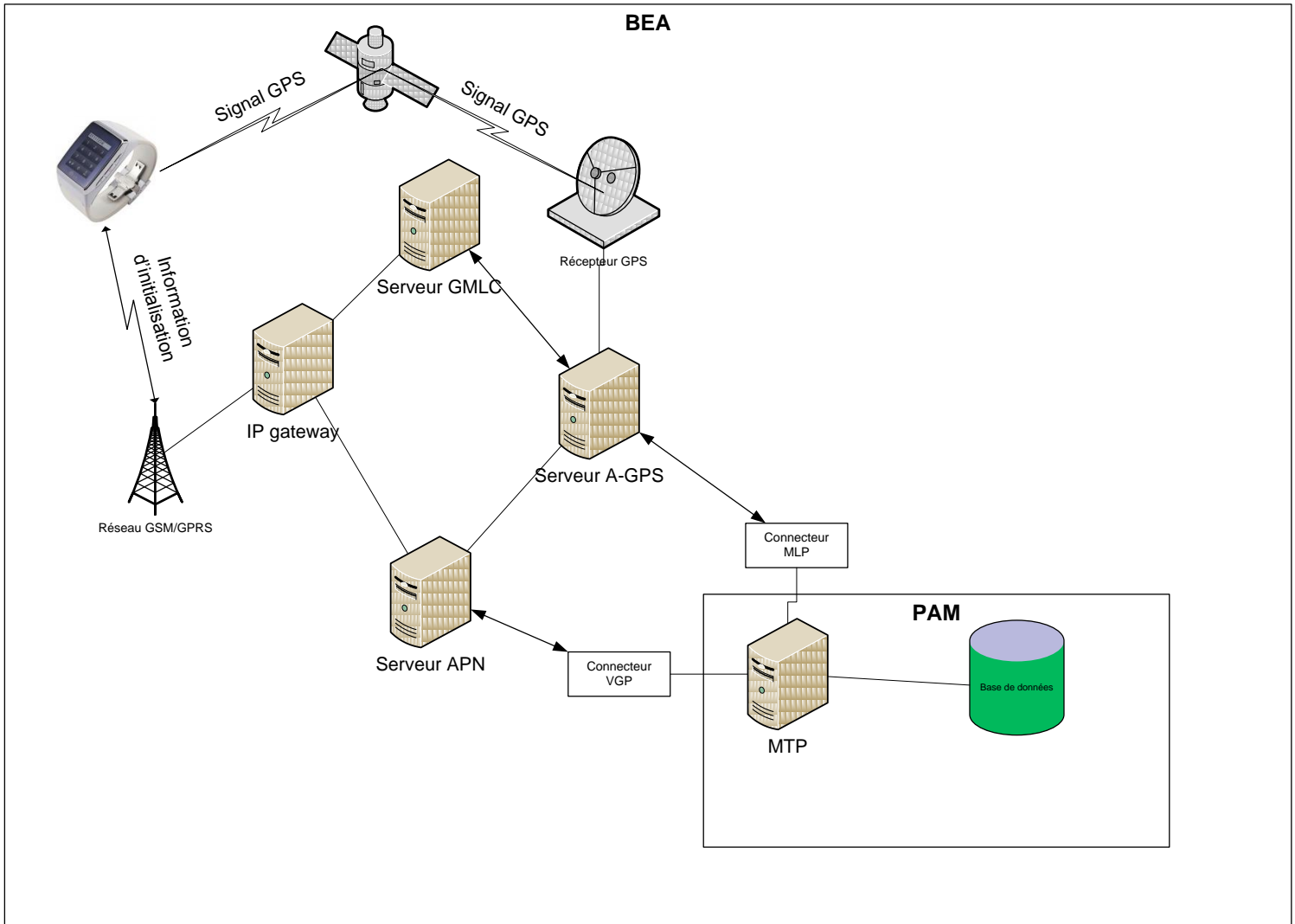
La partie logicielle sera développée grâce à des outils d'aide au développement tel qu'Eclipse qui est un Environnement de Développement Intégré ainsi que Junit pour les tests. Ce développement s'appuie sur des patrons de conception (ou design patterns) afin d'augmenter la qualité du résultat final et d'accélérer le développement.

Documents de présentation

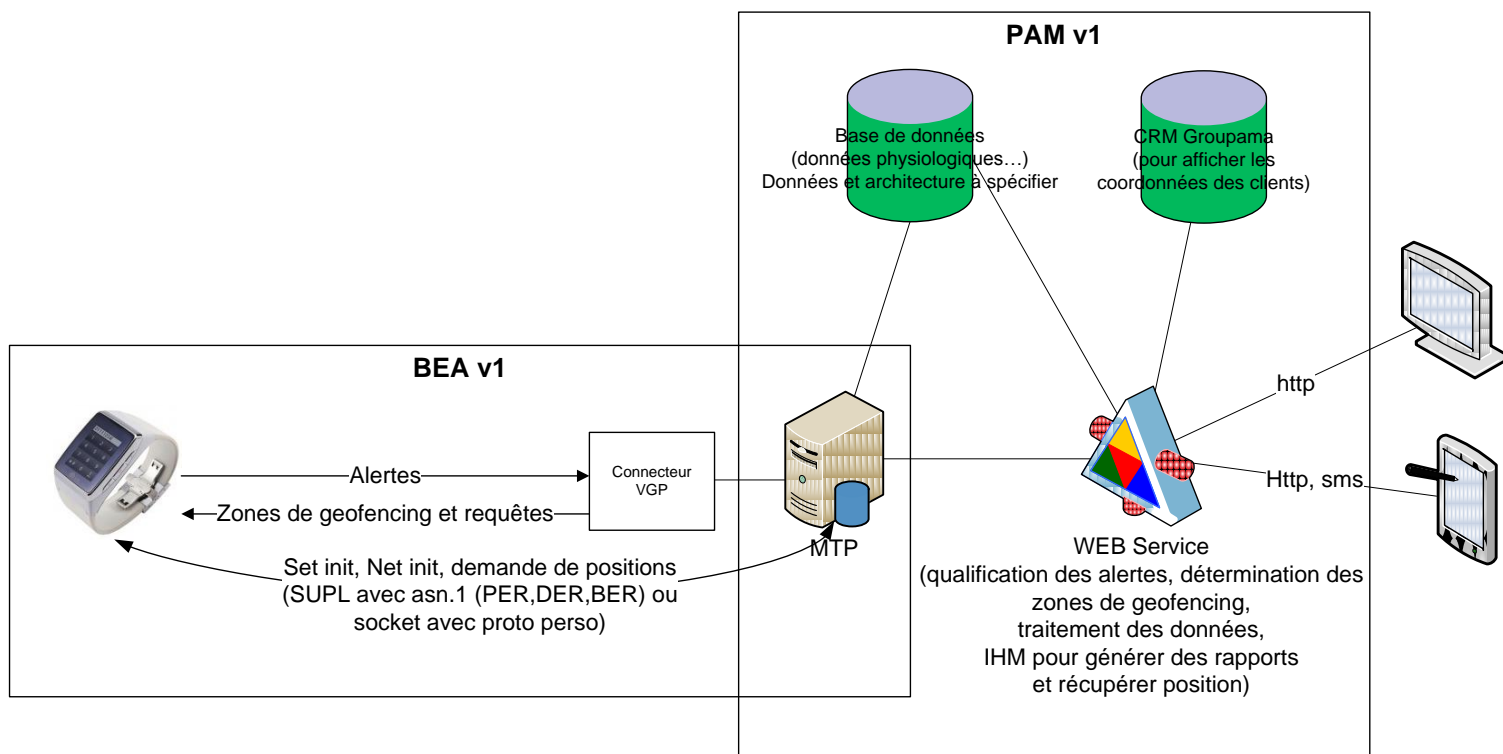
A.1. Présentation globale



A.2. Architecture globale



A.3. Projet BEA version 1



A.4. Projet BEA version 2

