

Sistema de Monitorização de Atividade Física

Fase B

Grupo 3



Índice




Constituição do Grupo	4
Introdução	5
Diagrama de Gantt	6
Arquitetura da Fase B	7
Diagrama de blocos da Arquitetura	8
Gestor Serviço	9
Interface Utilizador (UI)	10
Sensor Simulado	11
Protocolo de comunicação	12
Tipos de mensagens	14
Trama Data	14
Trama ERROR	14
Trama START	15
Trama STOP	15
Trama Id/Area	16
Reconhecimento da atividade	17
Análise de Testes	21
Gestor de Edifício	Erro! Marcador não definido.



Índice de Figuras

Figura 1 - Diagrama de Gantt	6
Figura 2 - Sistema sensor simulado, concentrador e gestor de serviço	7
Figura 3 - Protocolo TCP/IP	12
Figura 4 - Acelerómetro sem forças aplicadas.....	17
Figura 5 - Acelerómetro apenas com força de movimento	18
Figura 6 - Acelerómetro em estado normal	18
Figura 7 - Acelerómetro com inclinação de 45 graus	19
Figura 8 - Acelerómetro com eixos fixados	19
Figura 9 - Posicionamento do sensor	21
Figura 10 - Gráfico paciente deitado	21
Figura 11 - Gráfico paciente sentado	22
Figura 12 - Gráfico paciente andar	23
Figura 13 - Gráfico paciente correr.....	23

Constituição do Grupo

	Augusto Mota (a76563) a76563@halunos.uminho.pt
	Hugo Machado (a80362) a80362@alunos.uminho.pt
	Miguel Moreira (a77314) a77314@alunos.uminho.pt



Introdução

Nesta segunda fase do projeto, Fase B, o grupo teve como objetivos principais dar um passo em frente no projeto da cadeira de LTI II (Laboratórios de Telecomunicações 2) “Sistema de Monitorização de Atividade Física”.

Nesta fase podemos verificar todos os passos que demos na evolução do nosso projeto. Irá ser apresentado o nosso desenvolvimento do Gestor de Serviço onde fizemos uma recolha para uma Base de Dados local com toda a informação enviada por todos os concentradores da rede local do serviço e de seguida processá-la de modo a que fosse possível apresentar todo o conteúdo importante da monitorização da atividade física de cada sujeito de uma determinada área. Para tal, face às dificuldades que tivemos na realização da Fase A, em primeiro lugar aprimoramos todo o nosso projeto relativo a fase anterior de modo a que fosse possível a realização desta segunda fase de uma forma muito mais eficaz e concisa.

Construímos estes Gestores de forma a que fossem capazes de interpretar os dados enviados pelos concentradores e determinar as condições dos pacientes em tempo real.

Foi desenvolvido também nesta fase uma interface de utilização do gestor de serviço onde é apresentado o comportamento físico de um respetivo sujeito e ainda a sua área onde se encontra.

Nesta fase do projeto, quanto à ligação entre concentradores e gestor de área, o destaque é dado à comunicação por mensagens aplicacionais TCP/IP.



Diagrama de Gantt

Apresentamos de seguida na Figura 1 a planificação temporal do grupo relativa à Fase B.

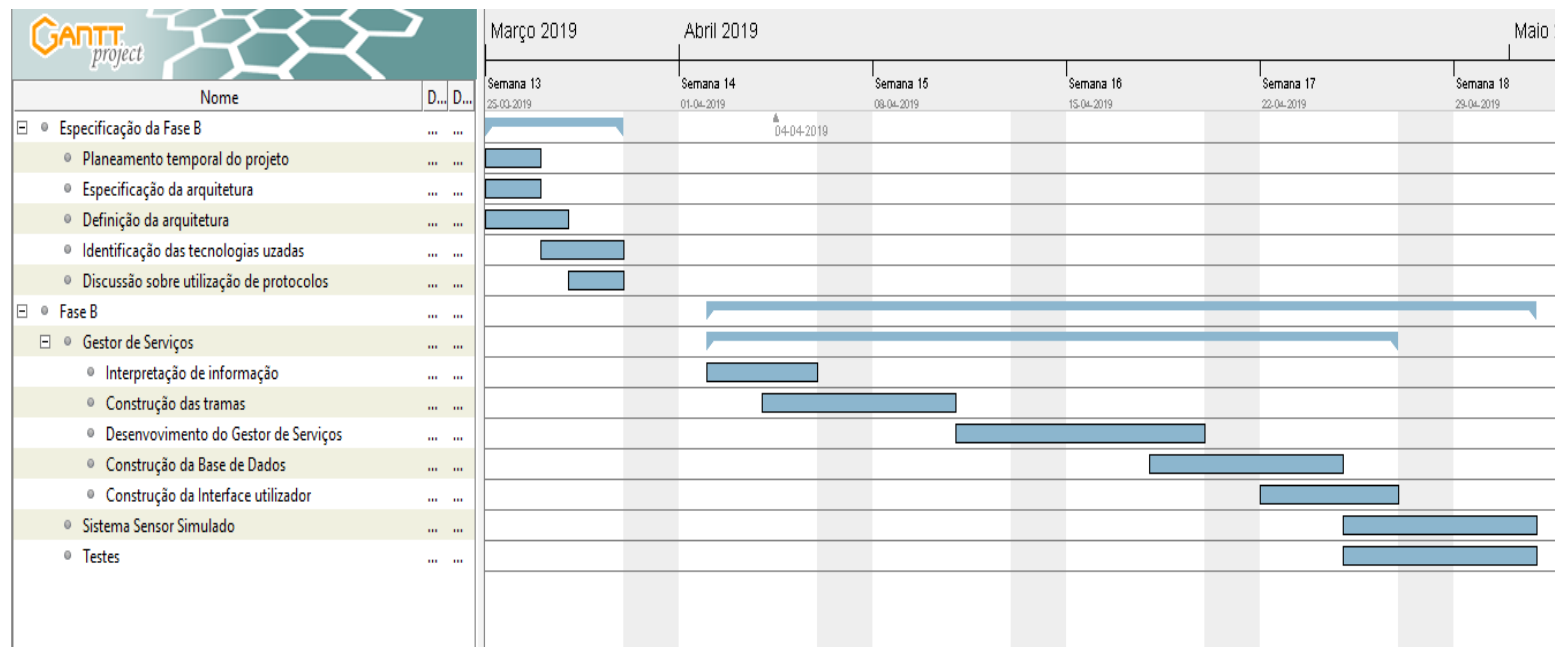


Figura 1 - Diagrama de Gantt

Arquitetura da Fase B

Para esta segunda fase, Fase B, o grupo terá como objetivo principal a implementação de uma primeira versão dos Gestores de Serviço incluído de tal forma todos os processos de comunicação com os concentradores, conforme se pode visualizar na figura 2. Os grupos têm ainda a opção de implementar sistemas sensores simulados.

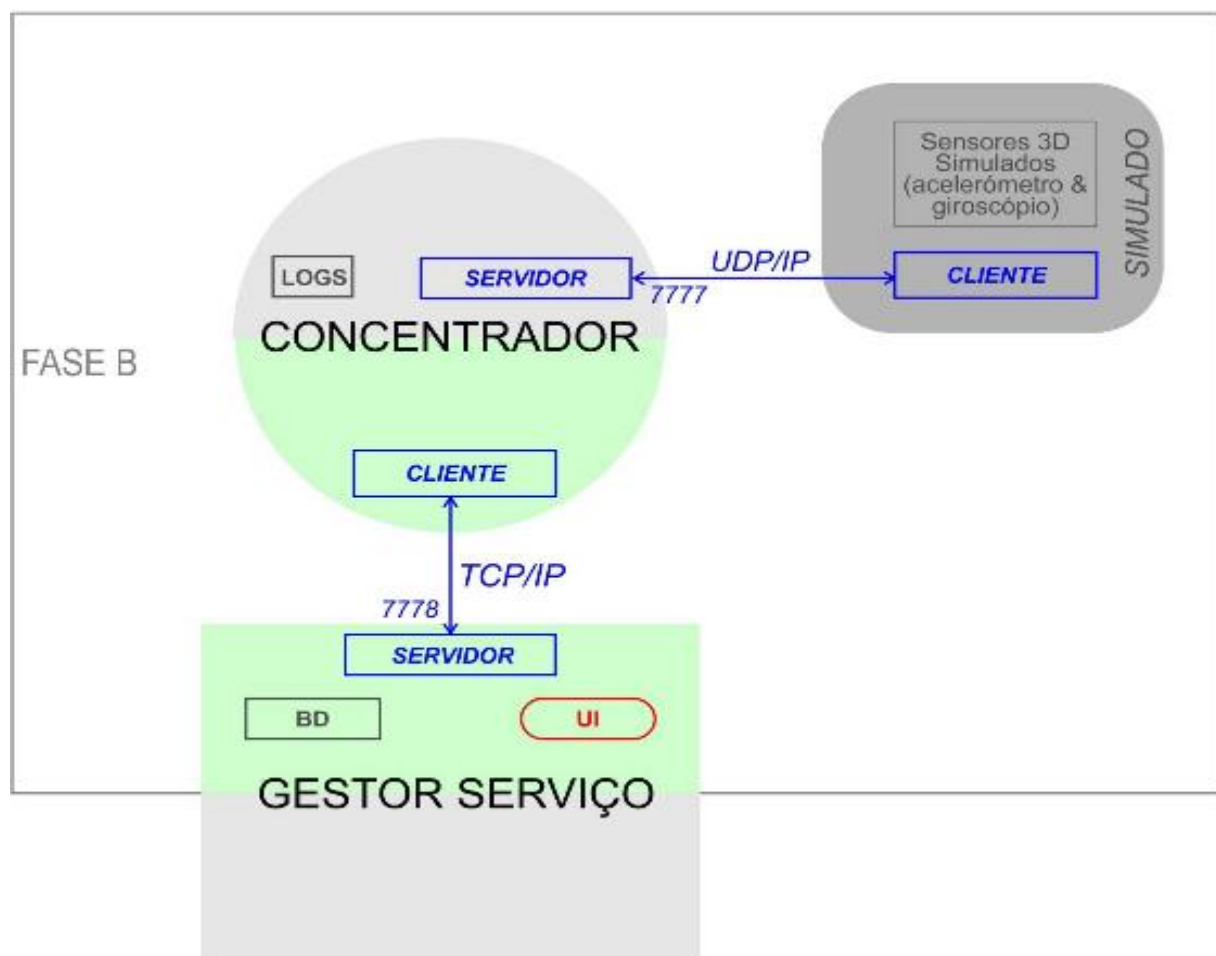
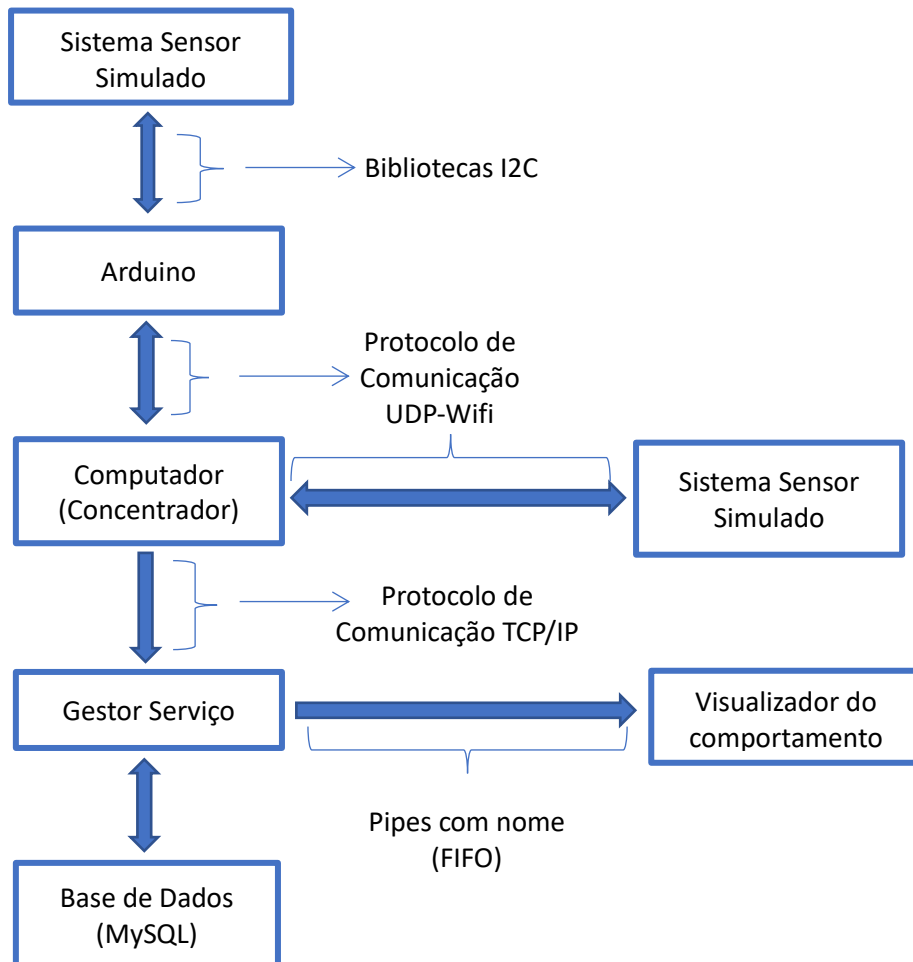


Figura 2 - Sistema sensor simulado, concentrador e gestor de serviço

Diagrama de blocos da Arquitetura

O sistema que implementamos tem como objetivo monitorizar em tempo-real um conjunto de variáveis que o sistema sensor recolhe ao longo do tempo. Para isso encontrasse em baixo um diagrama com o modelo de arquitetura do nosso sistema.



No nosso sistema sensor, temos várias variáveis que têm de ser interpretadas para monitorizar o paciente em tempo real, essas variáveis são:

- Acelerómetro eixo x;
- Acelerómetro eixo y;
- Acelerómetro eixo z;
- Osciloscópio eixo x;
- Osciloscópio eixo y;
- Osciloscópio eixo z;
- Temperatura;

As variáveis de acelerómetro e de osciloscópio, que são geradas pelo sistema sensor, são medidas com recurso a um circuito eletrónico desenhado de propósito para este projeto.



Gestor Serviço

Para a implementação do nosso Gestor de Serviço teremos com foco os seguintes pontos:

- Recolher numa simples Base de Dados local a informação enviada por todos os concentradores da rede local e processar esses resultados;
- Os dados presentes na Base de Dados devem determinar as condições dos sujeitos em tempo real sendo apresentados como exemplo:

Normal → parado;

Normal → andamento;

Alarme → agitado;

Alarme → queda;

Alarme → inativo.

Em suma a estrutura dos dados será a seguinte:

$X \rightarrow Y$

Onde x corresponde ao estado do paciente e o Y corresponde ao comportamento físico do mesmo.

As condições do paciente serão disponibilizadas aos utilizadores através de uma interface de utilizador local.



Interface Utilizador (UI)

Para cada cliente existirá:

- Uma interface adequada;
- Visualizador do comportamento físico de cada sujeito;
- Configurador de associação de um sujeito a uma área;

Para o gestor do serviço ou administrador:

- Que é único terá acesso a todas as funcionalidades disponíveis;
- A identificação do serviço deve ser configurada através da interface de utilizadores;
- Conjunto de áreas disponíveis para cada serviço vai estar indicada num ficheiro de configuração;
- Num ficheiro de configuração vamos conter a lista de sujeitos monitorizados em cada serviço e a sua área respetiva;
- O gestor vai alertar quando um sujeito deixou de ser monitorizado;
- A associação entre a identificação dum determinado sujeito a ser monitorizado e a identificação desse sistema vai ser feita nos concentradores.



Sensor Simulado

A implementação do Sensor Simulado terá como objetivo principal fazer com que as funcionalidades dos concentradores e dos gestores serviços possam ser testadas num ambiente com múltiplos sistemas sensores a funcionar em simultâneo.

Estes serão implementados inteiramente numa aplicação software com seguintes componentes:

1. Um ou mais ficheiros com valores sensoriais armazenados e que podem ter resultado da monitorização do funcionamento dum sistema sensor real;
2. No Sensor simulado, as mensagens serão encapsuladas em datagramas UDP (aplicação cliente UDP/IP);
3. As mensagens serão enviadas para a porta 7777 do servidor UDP (concentrador).

Protocolo de comunicação

O protocolo que vai ser utilizado nesta fase para realizar uma comunicação entre o concentrador e o Gestor Serviço é o protocolo TCP/IP, protocolo que é atualmente o mais usado em redes locais visto que foi criado para ser usado na Internet, sendo a Internet um sistema muito popular. Este protocolo deve ser construído à base de um pequeno conjunto de mensagens que irão servir para os concentradores enviarem informações básicas sobre o comportamento físico dos pacientes. Este conjunto de dados será enviado através do protocolo transporte TCP/IP. Este protocolo vai incluir uma forma de os gestores saberem quando a comunicação com um determinado concentrador é terminada brutaamente.

A grande vantagem deste protocolo é que ele é roteável, basicamente foi criado para grandes redes e de longa distância, onde podemos criar vários caminhos para o pacote atingir o computador recetor.

Este tipo de protocolo, possui uma arquitetura aberta onde qualquer pessoa poderá adapta-lo para o seu sistema operacional, possibilitando que todos os sistemas comuniquem entre si sem qualquer tipo de dificuldade. Na figura 3 podemos ver que este protocolo se encontra organizado em quatro camadas.

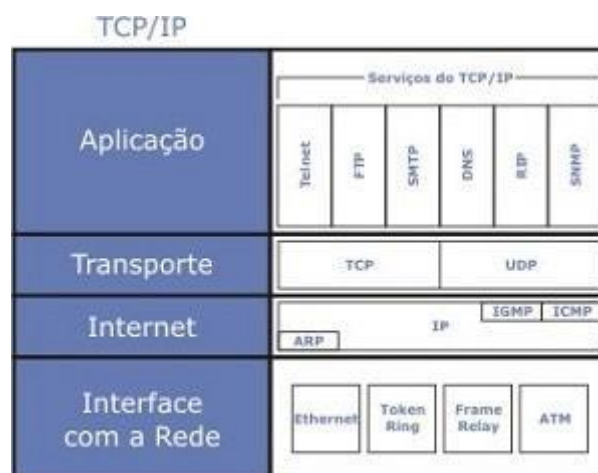


Figura 3 - Protocolo TCP/IP

Como podemos ver na figura 3, este protocolo encontrasse organizado em quatro camadas do protocolo TCP/IP:

- Camada de Aplicação: responsável pela comunicação entre os aplicativos e o protocolo de transporte. Nesta camada operam vários protocolos, sendo um dos mais conhecidos o HTTP, SMTP, FTP, SNMP, DNS e entre outros. Por exemplo, quando entramos num endereço WWW no browser para podermos visualizar uma página internet, o browser vai comunicar com a camada de aplicação TCP/IP, onde depois será atendido pelo protocolo HTTP.



A camada de aplicação comunica com a camada de transporte através da definição de uma certa porta. As portas são numeradas e as aplicações padrão usar sempre a mesma porta. Ao utilizar um número de uma porta, permite que o protocolo de transporte saiba qual é o tipo de conteúdo do pacote de dados, e o recetor saberá qual o protocolo de aplicação à qual ele terá que entregar os pacotes de dados. Através deste método, quem estiver a receber o pacote destinado a certa porta específica, o protocolo TCP irá entregar ao protocolo que estiver conectado a esta porta.

- Camada de transporte: camada responsável por receber os dados enviados pela camada de aplicação e transformar esses dados em pacotes que irão ser enviados para a camada de internet. No protocolo TCP/IP a camada de transporte utiliza um método de multiplexação, método que permite transmitir simultaneamente dados das diferentes aplicações. Nesta camada vão operar dois protocolos distintos, denominados por TCP (Transmission Control Protocol) e UDP (User Datagram Protocol). Ao contrário do TCP, o protocolo UDP não verifica se os dados chegaram ou não ao destino. Esta é uma das razões pelo qual o protocolo mais usado na transmissão de dados é o TCP. Ao receber os dados, a camada de transporte vai captar os pacotes que são transmitidos pela camada Internet e coloca esses dados em ordem e verifica se os dados chegaram corretamente. O protocolo IP, que é um dos protocolos mais conhecidos na camada Internet, não verifica se os dados chegam ao destino. Quem vai remontar a ordem dos pacotes recebidos, verificar se está a faltar algum e ainda retransmissão do pacote que não chegou é o protocolo de transporte TCP.
- Camada de Internet: Existem vários protocolos que podemos utilizar nesta camada, como por exemplo o protocolo IP, ICMP e ARP. Na realização da transmissão de um dado, o pacote de dados que será recebido na camada TCP vai ser dividido em pacotes chamados datagramas. Estes datagramas são enviados para uma camada de interface com a rede, onde serão transmitidos através do cabeçalho da rede. Esta camada não vai verificar se estes datagramas chegam ao destino corretamente, quem faz essa verificação é o protocolo TCP. Esta camada é apenas responsável pelo roteamento de pacotes, basicamente adiciona aos datagramas informações sobre o seu percurso.
- Camada de Interface com a Rede: esta camada vai ser responsável por enviar o datagrama recebido pela camada de internet através da rede.

O protocolo TCP/IP é um conjunto de protocolos que irão trabalhar em conjunto para proporcionar a garantia de entrega nos serviços de rede de computadores. Este protocolo garante que cada camada consiga comunicar com a sua camada inferior ou superior.



Tipos de mensagens

De seguida apresentamos os diferentes tipos de mensagem que teremos.

Trama Data

Mensagem que será enviada num certo concentrador para um gestor de serviço com os valores das amostras recolhidas pelo sistema sensores que estão conectados.

- TP - tipo de mensagem;
- ISS - identificação do sistema sensor;
- ISu - identificação do sujeito;
- NS - número de amostras presentes na trama;
- TS - timestamp do momento em que recolhe o valor da primeira amostra que vai estar presente na mensagem;
- Amostras - conteúdo do acelerómetro, giroscópio e temperatura retirado no respetivo tempo.

Estrutura:

4 Bytes	4 Bytes	4 Bytes	4 Bytes	4 Bytes	NS * 28 Bytes
TP	ISS	ISu	NS	TS	Amostras

Trama ERROR

Mensagem que será enviada num certo concentrador para um gestor de serviço a indicar a condição de erro.

- TP - tipo de mensagem;
- ISS - identificação do sistema sensor;
- ISu - identificação do sujeito;
- TS - timestamp do momento em que recolhe o valor da primeira amostra que vai estar presente na mensagem.

Estrutura:

4 Bytes	4 Bytes	4 Bytes	4 Bytes
TP	ISS	ISu	TS



Trama START

A mensagem que será enviada num certo concentrador para um gestor de serviço a indicar a que recolha de amostrar num sistema sensor foi iniciada.

- TP - tipo de mensagem;
- ISS - identificação do sistema sensor;
- ISu - identificação do sujeito.

Estrutura:

4 Bytes	4 Bytes	4 Bytes
TP	ISS	ISu

Trama STOP

Mensagem que será enviada num certo concentrador para um gestor serviço a indicar que recolha de amostras num sistema sensor foi interrompida.

- TP - tipo de mensagem;
- ISS - identificação do sistema sensor;
- ISu - identificação do sujeito.

Estrutura:

4 Bytes	4 Bytes	4 Bytes
TP	ISS	ISu



Trama Id/Area

Mensagem que será enviada num certo concentrador para um gestor serviço com a identificação da área do sujeito.

- TP - tipo de mensagem;
- ISS - identificação do sistema sensor;
- ISu - identificação do sujeito.

Estrutura:

4 Bytes	4 Bytes	4 Bytes
TP	ISS	ISu

Reconhecimento da atividade

Atualmente, vivemos num mundo onde o bem-estar de uma pessoa é umas das preocupações primordiais, sendo necessário manter em observação os movimentos do paciente.

A deteção do movimento dos pacientes é muito importante, visto que é necessário monitorizar constantemente a sua rotina diária, por exemplo, se um paciente não se encontra ativo, ou se este se encontra muito agitado e etc. Este projeto tem como propósito a deteção da atividade física dos pacientes através de um acelerómetro e giroscópio. Num hospital, por exemplo, é necessário monitorizar constantemente os pacientes, e quando é detetado quaisquer movimentos irregulares é essencial seguir a atividade física do paciente. Muitas vezes o zelador do paciente necessita de se ausentar em busca de auxílio de um enfermeiro ou de um médico deixando o paciente sozinho o que poderia resultar em fatalidade. Para resolver isso, o nosso programa deteta a atividade do paciente, e caso ocorra alguma irregularidade, as autoridades médicas são informadas no preciso momento. Para classificarmos a atividade física de um certo paciente é necessário usar um triaxial acelerómetro e também de um triaxial giroscópio.

Para percebermos a análise de resultados, vamos na secção em baixo explicar como o acelerómetro e o giroscópio funcionam para conseguirmos interpretar os valores que recebemos ao longo do tempo.

Para tal, vamos imaginar uma caixa com uma bola dentro dela, figura 4.

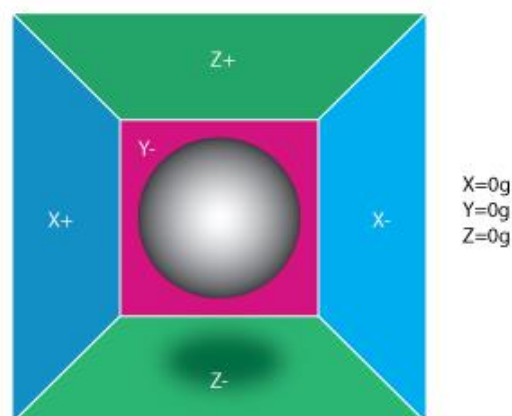


Figura 4 - Acelerómetro sem forças aplicadas

Se assumirmos que a caixa não sofre qualquer tipo de efeitos de forças gravitacionais ou nenhum outro tipo de força que possa afetar a posição da bola, a bola simplesmente vai encontrar-se a flutuar no meio da caixa. Através da imagem em cima, podemos visualizar que atribuímos a cada eixo um par de paredes, estando oculto o Y+ de modo a ser possível visualizar o interior da caixa.

Se movermos subitamente a caixa para o lado esquerdo (acelerar com valor $1g = 9,8 \text{ m/s}^2$), a bola vai atingir a parede X-. Atingida esta, é medida uma força de pressão que a bola aplica na parede e emitimos um valor de $-1g$ no eixo dos X. Podemos visualizar esta pequena explicação na figura 5.

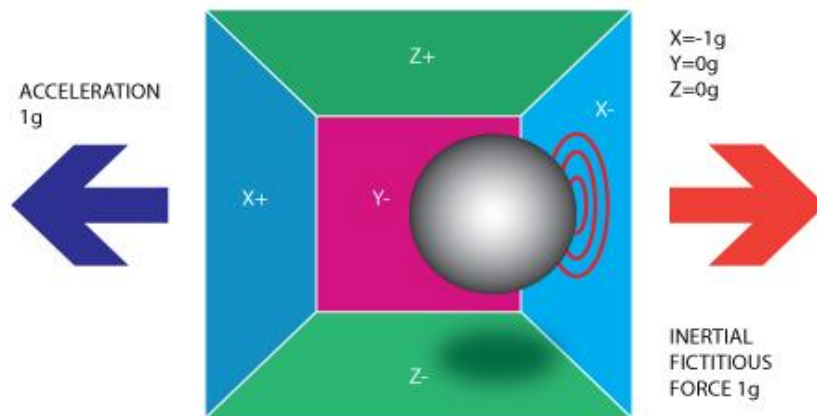


Figura 5 - Acelerómetro apenas com força de movimento

O acelerómetro vai detetar uma força na direção oposta do vetor de aceleração, força essa que é denominada de Força de Inércia. O acelerómetro mede a aceleração indiretamente através de uma força que é aplicada numa das suas paredes. Essa força pode ser o resultado da aceleração, mas nem sempre é causada por ela.

Pensando de uma forma mais real, e aceitando todas as forças existentes na terra, se a caixa estiver pousada numa superfície, a bola irá cair na parede Z- e será aplicada uma força de 1g na parede inferior como demonstra na figura 6:

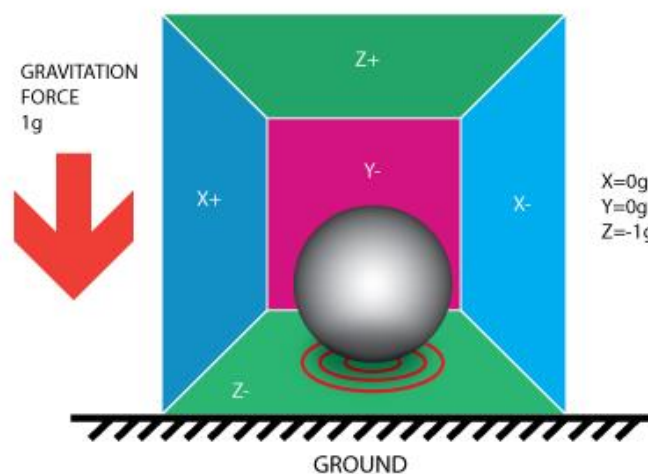


Figura 6 - Acelerómetro em estado normal

Neste caso a caixa não se vai mover, mas vai assumir um valor de -1g no eixo dos Z. A pressão que a bola vai exercer na caixa é denominada por força gravitacional. A aceleração causa uma força de inercia que é capturada pelo mecanismo de deteção de força do acelerómetro.

No exemplo seguinte vamos imaginar que inclinamos a caixa num ângulo de 45° para a direita. A bola irá tocar em duas paredes, a -Z e a -X. Vejamos o exemplo na figura 7.

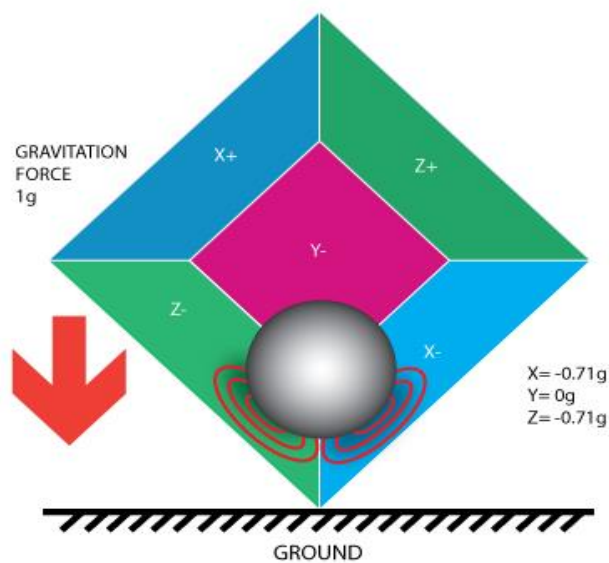


Figura 7 - Acelerómetro com inclinação de 45 graus

Nos exemplos que mostramos em cima, fixamos a força gravitacional e foi realizada uma rotação da caixa imaginária. Nos últimos 2 exemplos onde exemplificamos o vetor força gravitacional esta força manteve-se constante. Estes exemplos foram úteis para entender como funciona o acelerómetro e como ele interage com as forças externas, mas para entender melhor o acelerómetro será mais prático realizar cálculos se fixarmos o sistema de coordenadas do acelerómetro e imaginar que o vetor de força gira em torno de nós. Vejamos a figura 8 de modo a entendermos melhor.

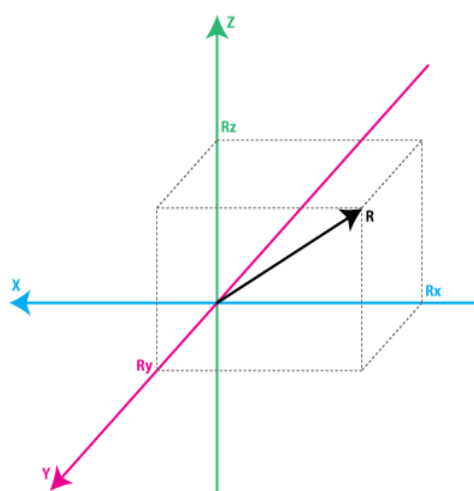


Figura 8 - Acelerómetro com eixos fixados



Na imagem a cima podemos visualizar um sistema de eixos de modo a conseguirmos explicar os modelos anteriores. Vamos imaginar que cada eixo do novo modelo seja perpendicular às respectivas faces da caixa imaginária que usamos nos exemplos anteriores. O vetor R é denominado por vetor força que o acelerómetro está a medir (sendo a força de gravitação a força de inércia ou a combinação de ambas as forças). Os valores de R_x , R_y e R_z são projeções do vetor R nos 3 eixos.

Para calcular o módulo do nosso vetor basta usar a seguinte expressão:

$$|R| = \sqrt{a_x^2 + a_y^2 + a_z^2}$$

Analise de Testes

Neste texto analisaremos alguns testes realizados ao nosso MPU-6050 em vários estados do nosso paciente, entre eles deitado, de pé, sentado, andar e ainda quando este se encontra a correr. Para ajudar na análise, o grupo, através dos valores do acelerómetro do nosso sensor (x, y, z) e com o apoio de um software online chamado de Plotly, construímos os gráficos que nos permitirão visualizar as variações instantâneas da aceleração nos três eixos. Para uma obtenção dos valores o mais corretos possível, na realização dos testes, colocamos o nosso sistema sensor encostado ao peito, local que o grupo considerou o mais correto. Como exemplificação apresentamos a seguinte imagem:

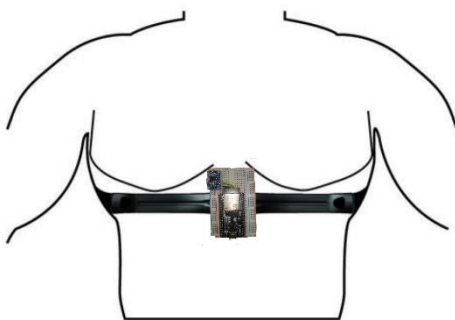


Figura 9 - Posicionamento do sensor

Na figura 10 podemos verificar o gráfico obtido quando o paciente se encontra deitado.

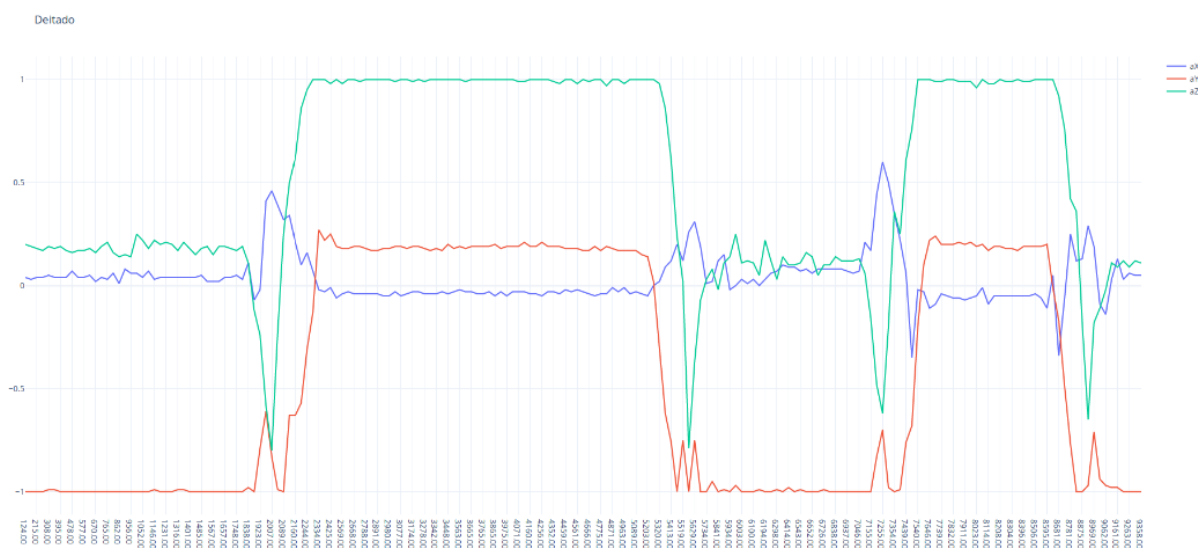


Figura 10 - Gráfico paciente deitado

Para analisar o sistema quando o nosso paciente se encontra deitado, alternamos o estado do mesmo entre de pé e deitado para melhor análise das variações instantâneas dos eixos relativos ao estado em que se encontra o sujeito. Começamos com um pequeno intervalo em que o paciente se encontra de pé passando depois para um estado deitado. Podemos verificar que tanto num estado como no outro as variações dos eixos do acelerómetro são insignificantes em ambos, existindo apenas alterações acentuadas quando este se levantava ou se deitava. Valores que consideramos corretos pois em ambos os estados não existe movimento algum por parte do nosso sistema sensor.

De seguida, na figura 11, apresentamos o gráfico correspondente aos valores obtidos quando o nosso paciente se encontra sentado alternando com momentos em que se encontra de pé.



Figura 11 - Gráfico paciente sentado

Para este teste, o paciente teve de se sentar de duas formas diferentes de modo a que fosse possível demonstrar as diferenças de quando ele se encontrava de pé e quando se sentava. Inicialmente o nosso sujeito encontrava-se de pé, e numa primeira alteração de estado este sentou-se de uma forma correta, o que a nível de valores de acelerómetro não houve grandes alterações, como esperado, pois o posicionamento do nosso sistema sensor não sofria alterações. O nosso paciente volta-se a por a pé e na terceira alteração de estado este senta-se de uma forma mais inclinada para trás, podendo assim, como demonstrado com os valores do acelerómetro no gráfico, ver as diferenças de quando este se encontra de pé e sentado.

No gráfico 12 conseguimos visualizar as alterações de valores do nosso acelerómetro quando este se encontra a andar.

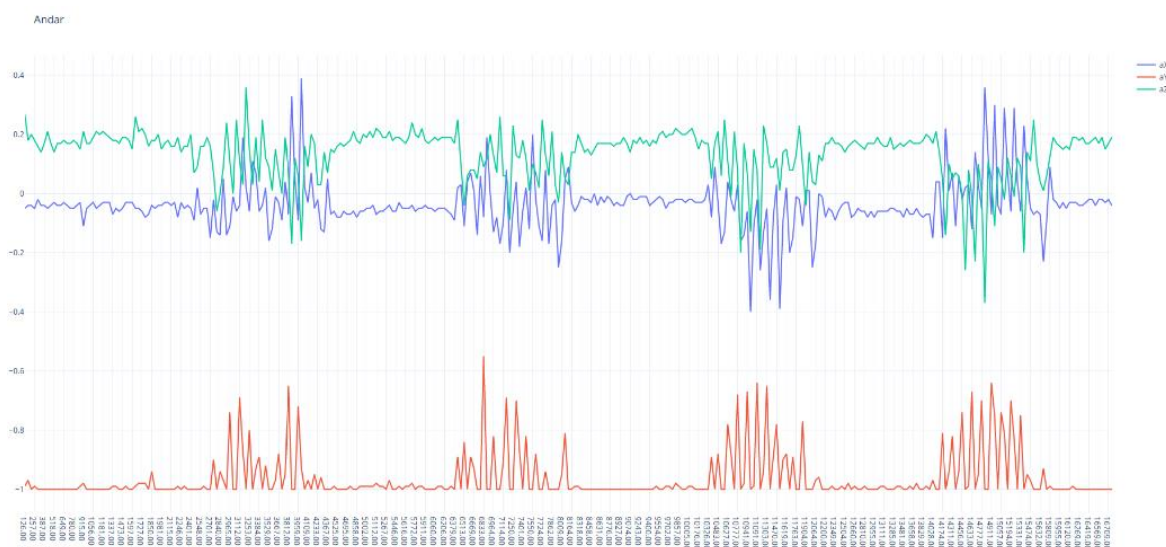


Figura 12 - Gráfico paciente andar

De forma a ser possível a análise dos valores de acelerómetro quando o nosso paciente se encontra a andar, realizamos os testes com os estados parado e andar intercalados, iniciando cada teste com o sujeito parado durante alguns segundos. Após a realização do gráfico através dos valores obtidos tornou-se mais fácil verificar quando o nosso sujeito iniciava a sua caminhada e as variações dos nossos eixos do acelerómetro nesta atividade, havendo, já com pequenas movimentações, grandes picos de variações dos nossos eixos.

Por último, na figura 13, podemos verificar o gráfico que demonstra as variações de valores dos eixos do acelerómetro quando o nosso paciente se encontra a correr.

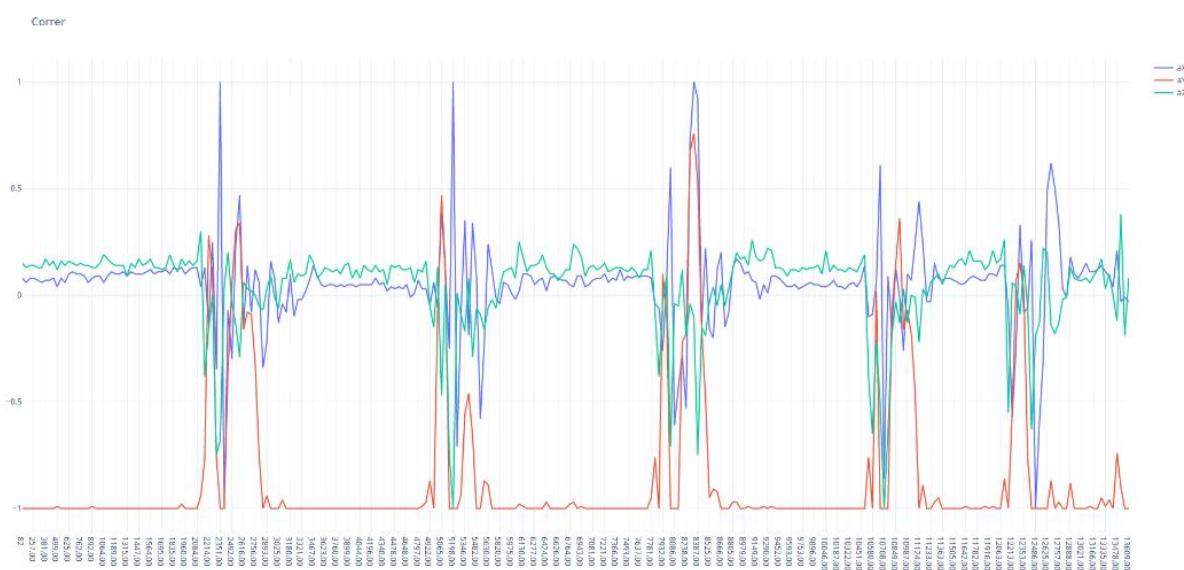


Figura 13 - Gráfico paciente correr



Do mesmo modo como aconteceu nos testes do paciente a andar, nos testes de corrida, o grupo decidiu seguir o mesmo esquema. Iniciamos o teste com um pequeno intervalo de segundos em que o nosso sujeito se encontra parado intercalando depois com intervalos em que este corria. Com os valores obtidos pelo acelerómetro construímos o gráfico e deste modo conseguimos visualizar as acentuadas alterações de valores dos nossos eixos quando o nosso sujeito se encontra a correr, existindo assim grandes picos de acelerações neste estado.

Algoritmo de comportamento

Para interpretação do comportamento recorreremos apenas aos valores dos eixos do acelerómetro, pois os valores do giroscópio eram menos precisos e a partir da fórmula do vetor dada pelo docente somos capazes de calcular o movimento e posição do sujeito em análise.

Com os testes feitos e análise de amostras, tiradas por nós, percebemos que o movimento pode ser interpretado através da variação dos eixos e através do cálculo do vetor, percebemos que o vetor toma valores mais altos ou tem picos quando está em movimento, mas isto não foi suficiente pois se o sujeito estivesse de pé parado, um dos eixos estava a 1 e daí o vetor também era aproximadamente 1.

A nossa solução foi recorrer à diferença dos vetores, e assim já conseguimos diferenciar se o sujeito está parado ou em movimento.

Após mais alguns testes e amostras, reparamos que o resultado nem sempre era o mais certo e ainda se demonstrava obter alguns erros, pois pequenas variações com pequenos movimentos poderiam resultar num movimento diferente do sucedido. Para obtermos valores mais precisos recorreremos a fazer uma média das diferenças, e assim conseguimos obter um resultado mais perto da realidade.

Para determinar se a posição do sujeito, usamos os eixos, quando não está em movimento. E por fim para determinar uma queda não conseguimos obter, um algoritmo constante, por exemplo travagens súbitas após uma corrida poderiam resultar numa queda, por isso resolvemos fazer uma condição que após uma paragem súbita se o sujeito se encontrar deitado no chão dá aviso de queda.

(Para o funcionamento ótimo do algoritmo os parâmetros deverão ser: $ns=8/pa=50/pm=400$)

O funcionamento real do algoritmo é feito da seguinte forma:

- Os valores de cada amostra são recolhidos da trama;
- é calculado o vetor dos eixos $accX$, $accY$ e $accZ$;
- é feita a diferença com o vetor anterior (caso seja o primeiro vetor não é feita a diferença);
- a diferença é somada;
- quando todas as diferenças (de um número limitado) forem calculadas e somadas é efetuada a média;
- com a media determinamos se o sujeito está em movimento, e com os eixos a sua posição;



Base de Dados

De forma a haver uma recolha e organização de todos os dados obtidos de cada sujeito, os docentes da cadeira propuseram a criação de uma base de dados para o efeito.

Inicialmente o grupo, a par da informação obtida pelo docente da cadeira, tinha em mente criar uma simples base de dados no formato de um ficheiro. Contudo, após encontradas bibliotecas que nos facilitavam a realização da tarefa, que nos interligava código C com MySQL (tecnologia para criação de base de dados)[1], optamos pela realização desta neste formato.

MySQL é um sistema de gestão de base de dados que utiliza como interface uma linguagem de consulta estruturada (SQL – *Structured Query Language*). É atualmente um dos sistemas de gestão de base de dados mais usados.

A nossa base de dados foi dividida em várias tabelas (ou entidades) que se distinguem pelas áreas em questão de cada concentrador. Cada uma destas tabelas conterá os valores obtidos através do acelerómetro, giroscópio e temperatura provenientes do sensor e correspondentes a um determinado sujeito assim como os seus comportamentos. Por exemplo, os valores correspondentes à Área X, estão identificados na tabela por Area_X_Values e os seus comportamentos por Area_X_Behaviour, onde X é o número da Área.

De modo a que fosse possível a inserção e retorno da informação na base de dados recorremos a *Query's*. Uma *query* é uma função que executa um pedido/consulta de informação ou de um determinado dado a uma base de dados.

Contudo, no que toca a tecnologias de base de dados a nossa solução para o problema não é a mais eficiente pois o nosso foco foi apenas guardar valores.

De seguida apresentamos o formato das nossas tabelas relativas aos valores e aos comportamentos de cada sujeito.

Area_X_Values:

ISS	ISu	Data	Tempo	aX	aY	aZ	gX	gY	gZ	temperatura
-----	-----	------	-------	----	----	----	----	----	----	-------------

Area_X_Behaviour:

ISS	ISu	Data	Tempo	Comportamento
-----	-----	------	-------	---------------

Arquitetura de Software do Gestor

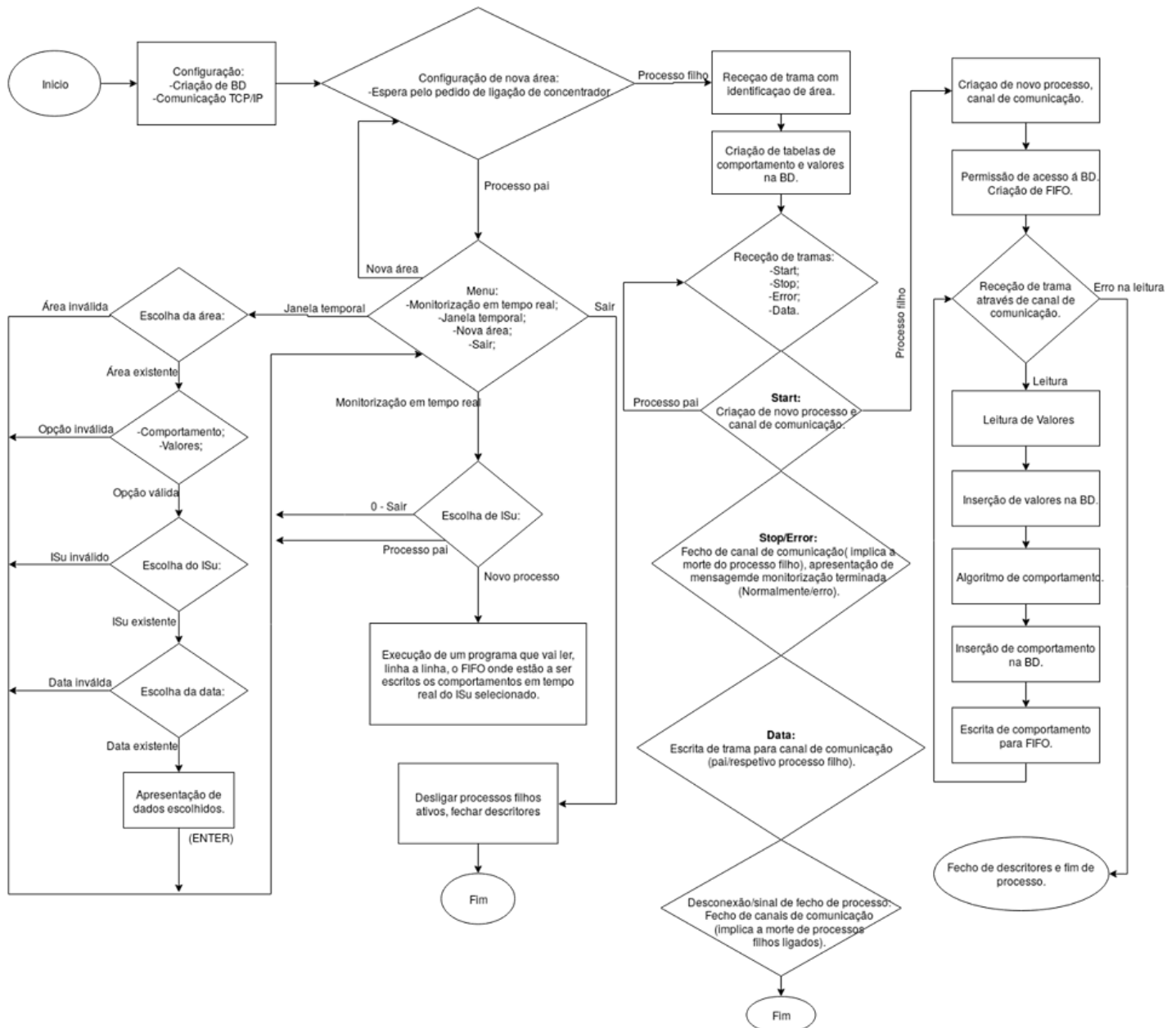


Figura 14 - Diagrama de blocos do concentrador (em linguagem natural)

Na inicialização do gestor(o servidor da arquitetura gestor/concentrador) a primeira fase consiste em criar um ficheiro (`setupTCPserver.txt`) para guardar as varias etapas cruciais da configuração do gestor, ler o ficheiro de configuração com a porta TCP/IP[2] a ser usada e é inicializada e configurada a SOCKET TCP/IP, por onde a comunicação entre concentrador e gestor se vai fazer e também é criada a base de dados, onde os valores e comportamentos serão armazenados.



Na seguinte fase vamos estabelecer contacto com o concentrador, a ligação é efetuada através de um `listen()`(no servidor), que aguarda que um concentrador estabeleça contacto, através de um `connect()`, e de seguida o gestor faz um `accept()`, e daí é criado um novo descritor (SOCKET) usado para a comunicação entre concentrador e gestor, após a ligação estabelecida, é criado um novo processo que está encarregue de lidar com o concentrador a ele ligado.

Após estas fases iniciais, o processo pai, entra no menu, onde existem as opções Monitorização em tempo real, janela temporal e nova área. Na Monitorização em tempo real, executo dois comandos da BASH que imprimem os FIFO's existentes na diretoria `"/Real"`, são os canais para onde os comportamentos associados a um sujeito vão ser escritos, após sabermos quais os sujeitos que estão a ser monitorizados, escolhemos um e é executado outro comando da BASH que abre um novo código numa nova janela. Este novo código apenas lê linha a linha o FIFO e imprime o resultado no ecrã, caso o sujeito eixe de ser monitorizado esta janela é fechada automaticamente.

Na opção janela temporal, apenas são executados alguns comandos, também conhecidos como QUERYS em MySQL, respetivos às seleções efetuadas pelo utilizador, tais como área, valores ou comportamento, sujeito e data.

Na última opção, nova área do menu, é efetuada uma nova conexão a um novo concentrador, o processo de conexão é o mesmo descrito em cima.

Voltando ao processo filho, este é o encarregado pela receção de mensagem do concentrador. Esta troca de mensagens começa receção da trama `Id/Area` onde o gestor é informado da área que está a monitorizar, e com o conhecimento da área são criadas duas tabelas na base de dados para os valores e para o comportamento, de seguida vamos para função de receção das tramas `Start`, `Stop`, `Error` e `Data`.

Quando é recebida uma trama de `Start`, é inicializado um canal de comunicação e é criado um novo processo para a função inserção de dados na base de dados.

Na receção da trama `Stop` e `Error`, o processo é o mesmo, consiste em fechar o canal de comunicação específico ao sujeito contido na trama, que leva á morte segura do processo responsável pela inserção de dados, no final é imprimida uma mensagem que indica se a monitorização foi desligada normalmente ou por algum erro.

Por último na receção da trama `Data`, a própria trama é enviada através do canal de comunicação para o processo responsável por aquele sujeito.

No processo filho (ou neto do processo principal) responsável pela inserção de dados, quando entra na função de inserção, inicialmente obtém a permissão de acesso há base de dados e cria o FIFO específico ao sujeito, e de seguida fica á espera das tramas `Data`.

Quando recebe uma trama, a informação é lida e inserida na tabela de valores da base de dados, seguidamente é efetuado o algoritmo de interpretação de comportamento, descrito anteriormente, e é devolvido o correspondente comportamento e inserido novamente na base de dados, na tabela de comportamento e caso o FIFO esteja aberto (ou seja a janela de monitorização em tempo real deste sujeito aberta), o resultado do comportamento também é escrito no FIFO.

Quando este processo se apercebe que o canal de comunicação foi fechado, ele procede para o fecho dos descritores abertos, e "suicida-se" de forma segura.



Conclusão

Em suma, com o finalizar desta fase e apesar de todas as dificuldades, conseguimos obter um resultado bastante superior em comparação com a fase A.

No início desta fase, visto os problemas encontrados na fase anterior, tais como construção, leitura e tempos das tramas, mais instabilidades e vulnerabilidades do código em si, resolvemos reescrever todo o nosso projeto, tanto a nível do concentrador como do sistema sensor de forma que fosse possível a realização desta nova fase com segurança e eficiência no seu alicerce (Fase A).

Após a conclusão desta primeira etapa iniciamos então a realização da fase B. Nesta, deparamo-nos com algumas dificuldades no que toca à base de dados e na interpretação dos comportamentos do sujeito, contudo conseguimos ultrapassar todos estes contratempos e concluir esta fase, na opinião do grupo, com sucesso.

Visto a construção do sistema sensor simulado não ser um ponto fundamental para a realização desta fase, o grupo decidiu deixar para o fim a sua realização, pois com o atraso inicial obtido com a reconstrução da fase A, achamos melhor dar mais importância aos pontos fundamentais do projeto, e no final, na possibilidade de ainda termos tempo, aí construía-mos o sistema sensor simulado. Contudo a sua realização não foi possível pois esgotamos todo o nosso tempo com os pontos mais importantes.

Por fim, embora não tenhamos efetuado o sensor simulado, achamos que a fase B no seu todo ficou pronta e funcional.



Referências Bibliográficas

[1] - <http://zetcode.com/db/mysqlc/>

[2] - <https://www.geeksforgeeks.org/tcp-server-client-implementation-in-c/>