



UNIVERSIDADE DE ÉVORA

ESCOLA DE CIÊNCIAS E TECNOLOGIA

DEPARTAMENTO DE INFORMÁTICA

A Serious Game para Reabilitação Cardíaca

Jorge Miguel Candeias Mota

Orientação: Miguel José Simões Barão

Co-orientação: Jorge Duarte dos Santos Bravo

Mestrado em Engenharia Informática

Dissertação

Évora, 2013

Mestrado em Engenharia Informática

A Serious Game para Reabilitação Cardíaca

Jorge Miguel Candeias Mota

Orientador

Miguel José Simões Barão

Co-Orientador

Jorge Duarte dos Santos Bravo

Sumário

Os acidentes cardiovasculares são uma das principais causas de morte em todo o mundo. Quando uma pessoa sobrevive a um acidente destes, ou lhe é detetado indícios da doença, executa um plano de reabilitação cardíaca, o que nem sempre é tão bem sucedido quanto poderia ser devido nomeadamente ao custo do tratamento e à falta de meios físicos e humanos. Neste trabalho propõe-se uma solução para controlo da carga de esforço em reabilitação cardíaca, que se baseia na informação obtida dos sensores VitalJacket e Kinect, para determinar o nível de dificuldade de um Serious Game, de modo a elevar o nível de esforço do paciente a uma Zona Alvo de Treino (ZAT) prescrita clinicamente. Toda a configuração do plano de exercícios é ajustável às necessidades e personalizável para cada paciente.

Palavras Chave: Serious Game, Reabilitação Cardíaca, Sensores

A Serious Game for Cardiac Rehabilitation

Abstract

Myocardial infarctions are one of the leading causes of death worldwide. Individuals who are victims of such events or are diagnosed as being at risk of experiencing one will generally be prescribed a cardiac rehabilitation regimen, which may not be as effective as planned due to treatment costs and the lack of available human and material resources. This thesis proposes a framework to control cardio-respiratory load during the rehabilitation process based on data gathered from VitalJacket and Kinect sensors in order to determine the difficulty level that a "serious game" should have so as to regulate a patient's physical exertion and reach a clinically designated "target practice area". Exercise regimens can be customized on-demand and can be personalized for each individual patient.

Keywords: Serious Game, Cardiac Rehabilitation, Sensors

Agradecimentos

Este trabalho marca o final de uma etapa importante na minha vida, a qual não poderia ter percorrido sozinho, pelo que quero agradecer a todos aqueles que nela foram importantes.

Agradeço ao meu orientador, Professor Doutor Miguel José Simões Barão e co-orientador, Professor Jorge Duarte dos Santos Bravo, por me terem confiado este trabalho e pela sua dedicação, orientação, boa disposição e perspectivas que me permitiram adquirir sobre o tema desta dissertação e não só.

À Microsoft e à BioDevices por terem cedido todo o material e apoio necessário para este trabalho.

Aos meus pais e irmã, os quais possibilitaram e incentivaram a minha formação.

À Kemet Electronics e Closer por me terem apoiado durante esta caminhada.

Uma grande obrigado à Joana Vicente, Hélder Rodrigues, Mundo, Fuse e Deau por todo o apoio e inspiração.

Por último, agradeço a todos os meus amigos que fazem parte da minha vida.

Acrónimos

2D	Bidimensional
3D	Tridimensional
API	Application Programming Interface
BD	Base de Dados
bpm	Batimentos por minuto
BPMrep	Batimentos por minuto do paciente em repouso
DLL	Dynamic-link library
ECG	Eletrocardiograma
FC	Frequência Cardíaca
FPS	frames per second
FR	Fatores de risco
IA	Inteligência Artificial
IR	Infrared
PT	Personal Trainer
SDK	Software Development Kit
SG	Serious Game
SQL	Structured Query Language
SSRS	SQL Server Reporting Services
ST	Skeleton Tracking

viii

VJ VitalJacket

ZAT Zona Alvo Treino

Conteúdo

Sumário	i
Abstract	iii
Lista de Conteúdo	x
Lista de Figuras	xii
Lista de Tabelas	xiii
1 Introdução	3
1.1 Motivação	3
1.2 Objetivos	4
1.3 Estrutura da Dissertação	5
2 Estado da arte	7
2.1 <i>Serious Games</i>	8
2.2 <i>Estado da arte de Serious Games para reabilitação cardíaca</i>	9
2.2.1 EA SPORTS Active 2	9
2.2.2 Esoma Exercise System	9
3 Sensores	11
3.1 Motor de jogo	11
3.2 Kinect	11
3.2.1 Funcionamento	12
3.2.2 Características	13
3.3 VitalJacket	15

3.3.1	<i>Wearable Technology</i>	16
3.3.2	Funcionamento	16
3.3.3	Características	17
4	Desenvolvimento	19
4.1	Conceitos e estrutura das aplicações e dados	19
4.1.1	Modelo e armazenamento de dados	22
4.2	Painel de controlo	24
4.3	Concepção do <i>Serious Game</i> para Reabilitação Cardíaca	28
4.4	Motor de jogo	29
4.4.1	XNA Game Studio	30
4.4.2	Unity 3D	31
4.4.3	Decisão sobre o motor de jogo a utilizar	31
4.5	Implementação do Serious Game para Reabilitação Cardíaca	33
4.5.1	Aquisição dos dados provenientes do VitalJacket	36
4.5.2	Sistemas de controlo de esforço	36
4.5.3	Instalação e execução do jogo	37
4.5.4	Jogabilidade	38
5	Modelo e ajuste de carga	39
6	Resultados experimentais	43
7	Conclusões e trabalho futuro	47
7.1	Conclusões	47
7.2	Trabalho futuro	48
	Referências bibliográficas	49

Lista de Figuras

3.1	Sensor de movimento Kinect	12
3.2	Feixe de raios IR projetados no objeto (após difração).	12
3.3	Efeito “sombra” do Kinect	14
3.4	Nomenclatura, oficial do Kinect, e respetiva posição das articulações detetadas	15
3.5	VitalJacket	16
3.6	Triângulo de Einthoven - cada elétrodo corresponde a uma das três derivações unipolares aumentadas dos membros: RA (braço direito), LA (braço esquerdo) e LL (perna esquerda)	17
4.1	Gráfico de Gantt que ilustra duas sessões de treino	20
4.2	Esquema representativo da relação entre sistemas e dispositivos respeitantes a esta solução	21
4.3	Diagrama da base de dados	24
4.4	Painel de controlo especificando parte do plano de exercício da reabilitação do paciente	25
4.5	Fluxograma referente ao pedido de alterações de dados de um determinado utilizador (paciente), por parte de um administrador (profissional de saúde). Ao Cliente corresponderá ao que é efetuado de modo direto pelo administrador no Painel de Controlo enquanto as ações do Servidor corresponderão às tomadas de forma autónoma pela aplicação ou BD.	27
4.6	Fluxo de funcionamento geral do jogo	30
4.7	Interface de desenvolvimento Unity 3D	31
4.8	Estrutura do jogo no Unity	33
4.9	Interface de ligação entre o modelo e o jogador - via OpenNI SDK	34
4.10	Menu principal do jogo	35
4.11	Ambiente do modo de jogo 3D	35

4.12	Esta seta indica que o jogador está com pelo menos 3 bpm abaixo da prescrição do exercício	37
4.13	Ambiente do modo de jogo 2D	38
5.1	Ciclo representativo dos ajustes de carga no jogo	41
5.2	Esta figura mostra uma simulação estocástica da resposta de um paciente com o sistema de controlo	41
5.3	Esta figura mostra uma simulação estocástica da resposta de um paciente com o sistema de controlo sem o fator de amplificação ($K = 0$)	42
6.1	Experimentação do sistema desenvolvido	43
6.2	Sessão de exercício em execução	45
6.3	Comportamento do ritmo cardíaco do jogador (linha azul) face ao objetivo (ZATref - linha laranja) durante o exercício	45
6.4	Evolução do fator carga aplicado em ordem ao tempo durante o exercício	46

Lista de Tabelas

2.1	Comparação entre o jogo EA Sports Active 2 e Objectivo deste trabalho . . .	9
3.1	Comparação entre Cardiófrequencímetro e VitalJacket	17
4.1	Tabela <i>Registo Valores</i>	23
4.2	Resultado da aplicação da função Pivot do Microsoft SQL Server	23
4.3	Comparativo das médias do desvio padrão OpenNi e Kinect SDK por co-ordenada	32
6.1	plano de exercício da sessão de teste	44
6.2	Resultado do teste <i>frames per second</i> realizado por FRAPS	44

Capítulo 1

Introdução

Este capítulo apresenta o tema desta dissertação, possibilitando ao leitor entender o contexto deste trabalho. São descritas as motivações e os objetivos deste trabalho. Por fim, é apresentada a estrutura da dissertação.

1.1 Motivação

As doenças cardiovasculares são uma das principais ameaças à vida do ser humano. Em Portugal, este tipo de doença é mesmo a principal causa de morte [9].

Algumas das doenças cardiovasculares mais comuns são: Ataque Cardíaco, Arritmias, Anginas e Insuficiência Cardíaca. Hoje em dia já existem vários métodos clínicos, comprovados, que ajudam os portadores destas doenças no processo de reabilitação [4].

A reabilitação cardíaca é recomendada não só a pessoas com problemas já detetados, mas também a pessoas em risco de sofrerem algum destes problemas. Os problemas cardiovasculares são na sua maioria hereditários - o que permite o aumento da eficácia do diagnóstico e respetiva prevenção.

Os benefícios obtidos com a prática de uma reabilitação cardíaca bem efetuada são notórios em diversos fatores entre os quais[6]:

- **Fatores de risco** – são reduzidos controlando o que os altera de forma direta: hipertensão, obesidade, tabagismo, *stress* e sedentarismo
- **Anatómicos** - as lesões nas artérias tendem a diminuir com o processo de reabilitação.

- **Sintomáticos** - o aumento da condição física do paciente faz com que este melhore na questão da falta de ar (dispneia) e inevitavelmente a fadiga constante na prática dos exercícios do cotidiano.
- **Psicológicos** - aumentar a auto-estima e o bem-estar fazem com que seja provocado um aumento de confiança no paciente, ao mesmo tempo que ajuda no controle da depressão e ansiedade, o que auxilia o paciente a conhecer os seus limites.
- **Econômicos** - melhorar as capacidades cardiovasculares do utente, fazem com que diminua a probabilidade de internamento e a despesa com medicação.

Infelizmente, os métodos de diagnóstico disponíveis não são acessíveis a todos os interessados devido nomeadamente a problemas económicos, humanos e de infraestruturas. Pelo fator risco, que neste caso é transversal a todos nós, a saúde é um dos setores onde sempre se investiu recursos e esforços para que possamos ser o mais saudáveis possível.

A tecnologia, sendo um dos principais setores da indústria atual tem atuado de forma persistente no desenvolvimento de sensores dos mais variados tipos visando dar resposta às necessidades. Devido à constante aposta no setor da saúde, esta é também uma área que não foge à regra e que tem sido congratulada com um aumento constante de dispositivos/aplicações que dão resposta a diversas necessidades. Por exemplo essa evolução deu origem a que um tradicional ECG, que recolhe a informação do comportamento cardiovascular de um paciente interpretando sinais elétricos através de um osciloscópio, pudesse ser substituído por um eletrocardiógrafo digital.

Equipamentos tecnológicos automáticos geralmente têm como vantagem a precisão na execução de uma tarefa, e a possibilidade de a repetir em condições iguais com a mesma taxa de sucesso - eliminando o famoso “erro humano”. Além disso, processos automáticos permitem ser replicados a uma grande escala. Assim melhorar todo o processo de reabilitação cardíaca utilizando a tecnologia atual é um desafio.

1.2 Objetivos

Este projeto tem como objetivo o desenvolvimento de uma aplicação, na forma de um *Serious Game*, que permita que a reabilitação cardíaca de pacientes de baixo risco seja efetuada na própria casa do paciente, com recurso a hardware de baixo custo.

A aplicação tem que ser motivadora para o paciente, encorajando-o a praticar determinados exercícios de reabilitação. Na reabilitação cardíaca, os conjunto de exercícios físicos deverão ser agrupados por sessões de exercícios. Por sua vez, estas sessões terão que possibilitar alterações conforme prescrição do profissional de saúde. Neste caso, a aplicação será na forma de um jogo, mais concretamente um *Serious Game* tornando possível o cumprimento de um plano de treino, e ao mesmo tempo assegurar uma motivação positiva no utilizador.

O sensor a ser utilizado neste projeto para avaliar os movimentos do paciente será o Microsoft Kinect e o sensor de ritmo cardíaco será o VitalJacket. Estes dois sensores serão o meio através do qual o paciente interagirá com o jogo. O objetivo principal que estará presente em todas as etapas do jogo e consiste em fazer com que o ritmo cardíaco do utente atinja a Zona Alvo de Treino (*ZAT*) estabelecida por prescrição clínica.

O ajuste do nível de esforço do paciente é efetuado mudando o grau de dificuldade do jogo, este ajuste é efetuado em função dos objetivos especificados pelo médico e da informação obtida pelos sensores, formando um sistema de controlo realimentado. Propõe-se um modelo dinâmico para a resposta do ritmo cardíaco ao esforço físico e é desenvolvido um controlador para este sistema.

O software desenvolvido deverá ser modular e direcionado a uma plataforma Microsoft. Deverá ser ainda garantida a diferença nas permissões de acesso por diferentes intervenientes da aplicação. Os acessos relativos aos utentes deste jogo deverão ser restringidos à respetiva autenticação no jogo para garantir a sua identificação.

O sistema tem que possibilitar a recolha de dados durante o jogo, armazenados num servidor remoto, de forma invisível ao utilizador. A configuração do jogo, que é assegurada pelos profissionais de saúde, tem que estar também localizada de forma remota em relação ao servidor *BD* e ao jogo. Deverão poder ser executadas várias instâncias do jogo, por diversos pacientes ao mesmo tempo, independentemente da sua localização geográfica. O número de instâncias do jogo deverá ser independente do número de servidores/nós *BD*.

A aplicação utilizará um acesso à internet para possibilitar a troca de informação entre o jogo e o servidor remoto que suportará a *BD*. Para que a aplicação possa ser utilizada por pessoas de várias faixas etárias e nível de conhecimentos informáticos, esta terá que ser *user friendly*.

Contudo, o produto resultante do final deste trabalho não visa substituir os métodos equivalentes utilizados na medicina atual, mas sim ser um suporte aos mesmos podendo funcionar como uma ferramenta extra.

1.3 Estrutura da Dissertação

Esta dissertação está organizada em 7 capítulos. No Capítulo 1 foi apresentada a motivação e os objetivos para este trabalho.

O Capítulo 2 apresenta o conceito e estado da arte em *Serious Games* direcionados à saúde, alguns exemplos de jogos que recorrem a este conceito.

No Capítulo 3 são introduzidos os sensores utilizados neste projeto: Kinect e VitalJacket.

O desenvolvimento do jogo, as suas características, opções tomadas e funcionalidades, bem como o detalhe dos componentes que o constituem, estão descritos no Capítulo 4.

No Capítulo 5 é explicado o modelo de aplicação de carga que atua sobre a dificuldade do jogo.

O Capítulo 6 apresenta os testes realizados e os respectivos resultados obtidos com o intuito de validar os objetivos propostos.

No Capítulo 7 é feita uma análise crítica sobre o trabalho desenvolvido e os resultados obtidos, e são apresentadas sugestões de trabalho que poderá ser realizado no futuro para melhorar esta solução.

Capítulo 2

Estado da arte

Com todo o desenvolvimento tecnológico destes últimos anos, os videojogos tomaram uma posição bem definida e cimentada no quotidiano das crianças, jovens, adultos e alguns idosos. Este é um dos setores que mais cresce todos os anos na indústria do entretenimento.

Um videojogo é uma aplicação que recorre à computação gráfica para promover entretenimento à pessoa em questão - o jogador. Ao longo do jogo, o jogador tem que executar ações para que atinja os objetivos:

“A game is a form of art in which participants, termed players, make decisions in order to manage resources through game tokens in the pursuit of a goal.” (Greg Costikyan)

Para jogar um videojogo, é necessário uma plataforma (*hardware/software*) compatível com o jogo em questão. Com o objetivo de acompanhar o crescimento desta indústria, existe constantemente a necessidade de melhorar a capacidade e características das plataformas que suportam os jogos. Por essa razão, todos os anos são feitas novas apostas em consolas, lançando para o mercado de consumo novos dispositivos e acessórios que suportam novos jogos, e despertem novas emoções e desafios no jogador. Neste momento as plataformas mais conhecidas no mercado são o PC, Playstation, XBOX, Wii e dispositivos móveis (Android/iOS/Windows Phone).

Os videojogos podem ser categorizados conforme o seu tipo, como por exemplo: estratégia, simuladores, ação, entre outros. Por vezes, devido à multidisciplinaridade dos videojogos, torna-se complicada a sua categorização. Este trabalho envolve um outro género de jogo: o *Serious Game*.

2.1 *Serious Games*

Desde o início dos videojogos que existe a ideia de que um jogo eletrónico pode não ter como único objetivo, gerar entretenimento para o jogador. Pretendia-se que os videojogos pudessem ser sinónimo de algo mais, como por exemplo proporcionar ao jogador a aquisição de novos conhecimentos durante a prática destes. Durante a história dos videojogos, foram feitas algumas tentativas de desenvolvimento de aplicações deste género principalmente na área da educação. Essas primeiras aplicações desenvolvidas, não tiveram sucesso pois os jogos, na sua maioria, ou eram demasiado fáceis ou demasiado difíceis - não se adaptavam ao jogador e dessa forma não contribuíam na aprendizagem do jogador.

O primeiro jogo a atingir o sucesso, surgiu na década de 80 com o nome *Army Battlezone* - produto da Atari (marca conceituada no mundo dos videojogos). O objetivo desse jogo era o treino de forças militares (em ambiente de tanques de combate), sendo plataforma de jogo *arcade*¹. As duas características principais que distinguiram este jogo dos que existiam nesse tempo foi o facto deste ser um FPS (First Person Shooter) e de toda a visão do jogo ser feita através de um óculo semelhante ao que existe nos tanques de combate. Desse modo, com a influência também do realismo proporcionado pela mecânica de jogo, o jogador/militar é “integrado” no jogo, e de forma quase inconsciente treina num ambiente simulado semelhante à realidade.

Com a prática deste jogo os militares mostraram que este novo tipo de videojogos era uma nova forma de aprendizagem. Uma vantagem notória neste jogo era o facto de ser uma alternativa económica e viável ao treino dos militares, pois a reprodução deste jogo apenas requeria mais plataformas, enquanto o treino em tanques de guerra reais teria que suportar armamento e combustível, para além do risco de vida eminente durante o treino.

O *Serious Game* é um modo de jogo que permite ao jogador aprofundar o conhecimento sobre determinado assunto ou explorar novas experiências que seriam difíceis, ou até mesmo impossíveis, de alcançar no mundo real, devido por exemplo ao elevado custo, segurança ou limitações próprias do jogador. Desta forma, num **SG** o entretenimento nunca é o objetivo principal, embora seja uma componente importante na motivação do jogador.

“serious game-based systems whose goal is to engender communication opportunities for players to learn about their strengths and weaknesses, receive real-time ingame assessment feedback on their performance, and share diverse solutions and strategies during, between, and after game play in order to update and adapt their understanding” (Raybourn, 2007).

¹“A video game arcade cabinet, also known as a video arcade machine or video coin-op, is the housing within which a video arcade game’s hardware resides” Wikipedia

2.2 Estado da arte de Serious Games para reabilitação cardíaca

Nos últimos anos, também devido à evolução das consolas, os Serious Games para reabilitação cardíaca têm surgido no mercado tanto na forma de produtos comerciais como de investigação e desenvolvimento.

2.2.1 EA SPORTS Active 2

Como exemplo de produtos comerciais, temos o jogo EA Sports Active 2, que permite ao utilizador praticar desporto dentro de casa, com recurso à consola Wii da Nintendo e ao auxílio de um cardiofrequencímetro próprio e dos acelerómetros dos controlos da Wii. Embora não seja um produto etiquetado como **SG** para reabilitação cardíaca, pode funcionar como tal. Na tabela 2.1, são comparadas as características do EA Sports Active 2 com o objetivo deste trabalho.

	EA SPORTS Active 2	Objetivo
<i>Personal Trainer</i>	Sim	Sim
Correcção de postura	Sim	Não
Detalhe do comportamento cardíaco	Não	Sim
Total configuração de sessão	Não	Sim
Instruções específicas por utilizador	Não	Sim
Histórico de resultados	Sim	Sim

Tabela 2.1: Comparação entre o jogo EA Sports Active 2 e Objectivo deste trabalho

2.2.2 Esoma Exercise System

Um bom exemplo de uma investigação sobre este tema é o Esoma Exercise System desenvolvido pelo Instituto de Tecnologia de Massachusetts (MIT). O protótipo recorre ao sensor de movimentos Kinect, da Microsoft e além do ritmo cardíaco monitoriza também a saturação de oxigénio do paciente durante o exercício. Envia também os dados relativos ao treino do paciente, em tempo real, para um monitor/profissional de saúde.

Capítulo 3

Sensores

3.1 Motor de jogo

”A sensor (also called detector) is a converter that measures a physical quantity and converts it into a signal which can be read by an observer or by an (today mostly electronic) instrument.” (Definição de “Sensor” em Wikipedia)

Os sensores são cada vez mais utilizados por todo o tipo de aplicações devido à sua utilidade e facilidade de integração. Existem variados tipos de sensores para diversas finalidades. Neste trabalho, para ser possível atingir os objetivos, seria necessário utilizar um sensor de movimento, para captar o comportamento físico do utilizador e um sensor que providencie uma leitura do ritmo cardíaco em tempo real, do utilizador, em modo ambulatório via *wireless*. Um requisito de ambos os sensores é transmitir a informação lida para a aplicação a desenvolver em tempo real.

Entre os sensores que cumprem estes requisitos pretendidos, optou-se pela utilização do Kinect e VitalJacket pela sua mediatização e disponibilidade para a plataforma alvo.

3.2 Kinect

O Kinect (Figura 3.1) é um sensor de movimentos da marca Microsoft, lançado em Novembro de 2010 para a plataforma de jogos Xbox 360. Foi desenvolvido pela Microsoft em parceria com a empresa *Prime Sense*. Embora o conceito deste sensor não fosse original, foi o primeiro sensor de movimentos, direcionado aos jogos, a ser comercializado em larga escala.



Figura 3.1: Sensor de movimento Kinect

3.2.1 Funcionamento

A tecnologia que permitiu à Prime Sense desenvolver este sensor, tem como base a utilização de um laser que emite um feixe de raios **IR** através de uma lente difrativa sendo dessa forma transformado o feixe numa organizada matriz de pontos **IR**. Essa matriz, de dimensão fixa, é projetada no objeto, e logo de seguida esses pontos projetados no objeto são recolhidos pelo receptor **IR**. Após serem comparadas as intensidades dos pontos na sua origem (quando são o feixe sofre a difração) com a intensidade do ponto projetado e tendo em conta a natureza dos raios **IR** (Figura 3.2), é possível saber com alguma precisão, a distância entre o ponto que está projetado no objeto e o sensor Kinect. Quando dois pontos projetados possuem diferentes intensidades, significa que não se encontram à mesma distancia do sensor. Pode-se concluir que o ponto com mais intensidade encontra-se geralmente mais perto do sensor, expecto se o material em questão (onde estão a ser projetados os feixes) não refletir o feixe **IR** de forma normal, absorvendo-o. Com esta técnica é possível saber a forma, posição e tamanho de um objeto. Executando este método várias vezes em determinado intervalo de tempo Δt , tendo em conta a diferença da posição Δs onde se encontra o objeto é possível saber a velocidade (v) com que se move determinado objeto (equação 3.1).

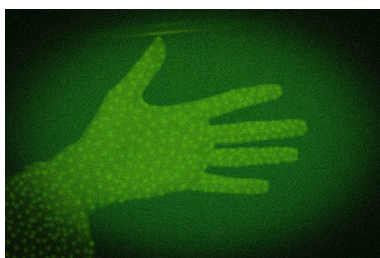


Figura 3.2: Feixe de raios IR projetados no objeto (após difração).

$$v = \frac{\Delta s}{\Delta t} \quad (3.1)$$

Embora esta seja a principal funcionalidade, o Kinect permite ainda que seja captado som, através do seu microfone integrado, e ainda vídeo através da sua câmara. O microfone serve, principalmente, para possibilitar à aplicação que faz uso do Kinect reconhecer comandos de voz e possibilitar ao Kinect uma perceção mais concreta do ambiente no qual está inserido como por exemplo saber o número de pessoas presentes na sala (através do

ruido). A câmara, como todos os sensores de imagem, devolve uma matriz de pixeis com a respetiva cor de cada um (RGB) - esta informação é por exemplo utilizada pelo Kinect para assimilar com a informação do sensor IR aumentando a precisão dos resultados.

Os dados fornecidos pelo Kinect SDK respeitantes à posição do utilizador são:

- **Pontos 3D** – cada ponto simboliza neste caso, o eixo de ligação entre dois segmentos/“ossos”. É utilizado um sistema de coordenadas tridimensionais (x,y,z) sendo que x e y são a localização de determinado ponto numa tela 2D fictícia, equivalente às coordenadas de um determinado objeto numa fotografia sendo que o ponto $(0;0)$ se situa no centro geométrico da mesma. A coordenada z é a distância entre esse ponto e o sensor;
- **Matriz rotação** – para determinar o ângulo que é feito entre cada ligação e o vector de coordenadas do sensor, é utilizada uma matriz. Esta informação é essencial para distinguir se por exemplo o paciente se encontra de frente ou de costas para o sensor [2];

3.2.2 Características

Nos primeiros tempos de existência do Kinect, além de não existir quaisquer concorrência no mercado para este género de produto, este estava apenas disponível para a consola de jogos Xbox 360. Foi preciso esperar um ano para que, de forma oficial, a Microsoft disponibilizasse o SDK de desenvolvimento do sensor, compatível com sistemas operativos Windows.

Para ser realizada a configuração do sensor, tendo em conta o utilizador, é necessário que a pessoa em questão esteja à frente do sensor e a uma distancia mínima de 2 metros. O Kinect permite que várias pessoas sejam controladas pelo sensor ao mesmo tempo. Para calibrar a posição da pessoa relativamente ao sensor (que está fixo), é necessário que esta se mantenha estática enquanto é feita a assimilação entre a câmara RGB e o sensor IR.

O sensor IR revela dificuldades e não apresenta resultados credíveis quando está exposto a demasiada luz solar ou quando projeta o feixe que sofre difração em objetos transparentes, refletores ou absorsores deste tipo de radiação. No caso em que não há luz visível presente na sala, a câmara RGB não deteta o utilizador e isso dificulta o processo de calibração do Kinect.

Não é possível identificar um objeto que esteja escondido atrás de outro que esteja na “sombra” de outro objeto mais próximo do sensor. Este problema deve-se principalmente ao facto do emissor de raios infravermelhos estar localizado a uma distancia de 8 centímetros do recetor. O esquema da Figura 3.3 representa um possível caso de “sombra e é apresentado seguindo uma visão de topo/cima-baixo. Os dois objetos que estão na imagem são representados por polígonos de cor preta. As linhas vermelhas representam os feixes

de **IR** que tocam a extremidade do *Objeto1*. Como foi referido, para que algo seja reconhecido pelo Kinect, é necessário que esteja ao alcance tanto do emissor como do receptor **IR**. Nesta figura, esses pontos visíveis por ambos estão representados em cor verde. Os pontos invisíveis para o Kinect, são representados pela cor amarela e são esses os pontos “sombra”. Neste caso, o *Objeto1* é totalmente visível pelo sensor, enquanto que o *Objeto2* apenas é parcialmente visível.

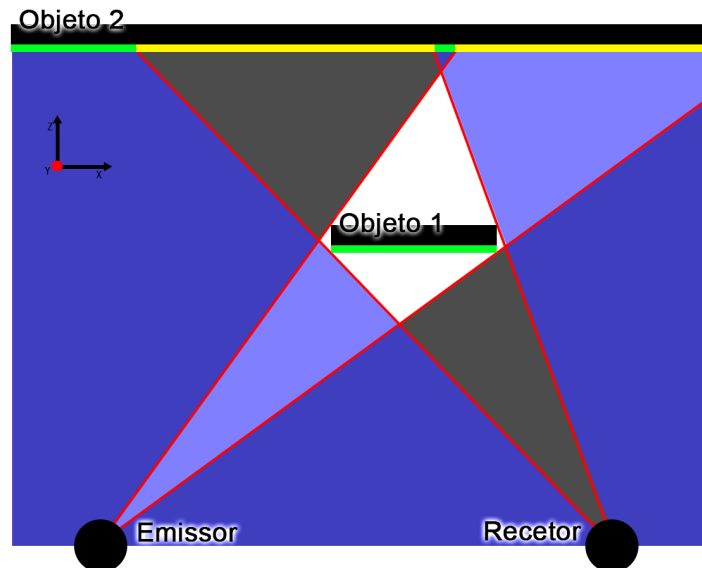


Figura 3.3: Efeito “sombra” do Kinect

O Kinect **SDK** trabalha os dados provenientes do sensor IR e disponibiliza em tempo real a representação das 20 articulações principais do corpo humano (Figura 3.4), que podem ser posteriormente utilizadas pelas aplicações para captar movimentos ou representar a pessoa no jogo através de um modelo.

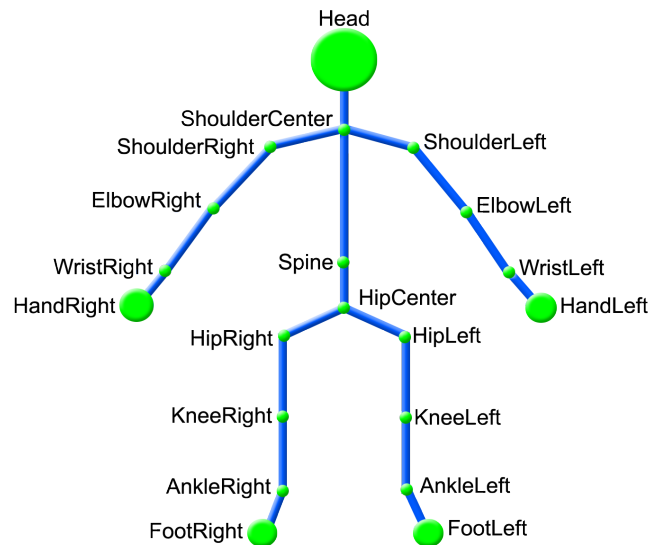


Figura 3.4: Nomenclatura, oficial do Kinect, e respetiva posição das articulações detetadas

3.3 VitalJacket

VitalJacket is the first ambulatory ECG system that combines wearable technology with a mainstream biomedical engineering solutions, aiming to provide a more reliable and comfortable Holter system. VitalJacket is an easy-to-wear t-shirt that collects over 72 hours of continuous data. Access to increased periods of such clinical information allows doctors to make a more accurate assessment of their patient's cardiac problems in a more natural, daily environment.

[1]

Também a medicina de diagnóstico com monitorização de sinais vitais dependem da evolução tecnológica. A BioDevices germinou com a criação do VitalJacket que é um exemplo de sucesso desta evolução. O VitalJacket é um produto que foi desenvolvido na Universidade de Aveiro e testado com sucesso no Hospital S. Sebastião em Aveiro.

Consiste num dispositivo na forma de colete, capaz de medir e registar o comportamento cardíaco de quem o está a utilizar (Figura 3.5). A BioDevices está a trabalhar para que esta se torne uma alternativa viável face ao habitual ECG, com o objetivo do VitalJacket ser menos dispendioso e permitir que sejam recolhidos os dados em pleno exercício físico, sem que a pessoa tenha que estar ligada a uma máquina de forma imóvel através de fios. O VitalJacket possibilita o armazenamento interno dos dados recolhidos pelos seus sensores, bem como o envio desses mesmos dados, em tempo real, para outros dispositivos através de Bluetooth.



Figura 3.5: VitalJacket

3.3.1 *Wearable Technology*

O conceito *Wearable Technology* (em português “tecnologia vestível”) é bastante recente, e aplica-se a todo o tipo de dispositivos que o nosso corpo tem a capacidade de “integrar”, de alguma forma, e que nos acrescentam novas características.

Este novo tipo de tecnologia é uma consequência da crescente massificação de utilização tecnológica refletida em dispositivos móveis (por exemplo telemóveis, leitores MP3, computadores portáteis entre outros) com o melhor da nanotecnologia [5]

Algumas evidências práticas deste conceito são os relógios inteligentes, óculos e até mesmo as próteses ortopédicas.

3.3.2 **Funcionamento**

O VitalJacket, à semelhança do **ECG**, utiliza elétrodos para efetuar a comunicação entre o corpo do utilizador e o sensor. Esses elétrodos são colocados no paciente com a disposição do Triângulo de Einthoven (Figura 3.6) à semelhança do que acontece no **ECG**.

São necessários os seguintes passos para dar início à atividade do **VJ**:

1. limpar com álcool as zonas onde vão ser colocados os elétrodos;
2. colocar os elétrodos nas respetivas posições;
3. vestir o colete e ligar os elétrodos aos condutores que estão incorporados no VitalJacket;
4. ligar o dispositivo eletrónico que está incorporado no computador e efetuar a configuração apertando um botão de forma a que o dispositivo grave a informação no cartão SD ou a envie diretamente por *Bluetooth*.

Toda operação de preparação do **VJ** operacional é feita em aproximadamente 15 minutos, enquanto a remoção do colete e elétrodos do corpo do utilizador demora cerca de 5 minutos[7].

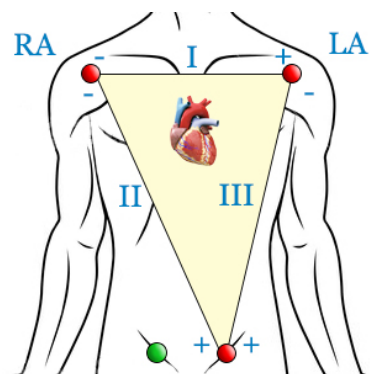


Figura 3.6: Triângulo de Einthoven - cada elétrodo corresponde a uma das três derivações unipolares aumentadas dos membros: RA (braço direito), LA (braço esquerdo) e LL (perna esquerda)

A leitura dos dados por *Bluetooth* pode ser feita através de uma aplicação própria da Biodevices (disponível para PC e dispositivos móveis). Neste trabalho, o ideal seria ligar o **VJ** de forma direta à aplicação a desenvolver, para minimizar o tempo de resposta da aplicação e facilitar a sua instalação e desenvolvimento.

3.3.3 Características

Existem dois modelos do VitalJacket: o 1L e o 5L. A versão mais simples - 1L - utiliza três elétrodos que possibilitam a leitura da frequência cardíaca enquanto que a versão mais complexa (5L) utiliza cinco e desse modo tem a capacidade de ler os sinais de um **ECG**, temperatura corporal e também o nível de saturação no sangue.

O VitalJacket 5L apresenta diversas vantagens face a um cardiofrequencímetro (Tabela 3.1).

	Cardiofrequencímetro	VitalJacket
Monitor de ritmo cardíaco	Sim	Sim
Monitor de ondas cardíacas	Não	Sim
Qualidade clínica	Não	Sim
Relatório de eventos	Não	Sim
Deteção de arritmias	Não	Sim

Tabela 3.1: Comparação entre Cardiofrequencímetro e VitalJacket

Comparado ao **ECG**, o **VJ** possibilita uma recolha dos dados em ambulatório pois a transmissão (ou armazenamento no cartão de memória) dos dados é realizada em modo sem fios.

Capítulo 4

Desenvolvimento

O intuito deste projeto, é ajudar o paciente no processo da sua reabilitação cardíaca, recorrendo não só ao auxílio dos sensores Kinect e VitalJacket, mas também a *software* específico. Tal como acontece nos processos de reabilitação cardíaca da medicina convencional, neste jogo cada paciente deverá ter um programa personalizado no que diz respeito aos exercícios a executar, duração e frequência do respetivo treino, ritmo cardíaco¹ alvo em cada fase do treino entre outros.

Durante a etapa de criação do jogo, a fase de desenvolvimento deve ser estruturada para facilitar futuras alterações relacionadas com o desenvolvimento do próprio jogo. Entre os principais detalhes que devem ser definidos destacam-se: a decisão sobre a plataforma que suportará o jogo e as ferramentas que serão utilizadas durante o seu desenvolvimento.

Deste modo a solução proposta terá como plataforma de execução computadores com sistema operativo Windows. Esta opção justifica-se pelo facto dessa combinação ser a mais predominante nos hospitais como nas casas dos pacientes, pois seria mais difícil dispor de uma consola específica num hospital ou mesmo em casa dos utilizadores.

4.1 Conceitos e estrutura das aplicações e dados

Para permitir o controlo da frequência cardíaca é necessário definir os limites que serão aplicados ao paciente em determinado exercício - *ZAT* (Zona Alvo Treino). A *ZAT* Referência é o valor ideal a atingir e apesar de nem sempre ser equidistante da *ZAT* Mínima

¹O ritmo cardíaco ou frequência cardíaca, pode ser definido como o número de batimentos do coração por unidade de tempo, normalmente por minuto (bpm). O ritmo cardíaco pode ser medido através de um ECG, cardiofrequencímetro ou de forma manual.

e Máxima, tem que se situar obrigatoriamente entre estas. A **ZAT** Mínima corresponde ao número mínimo de **bpm** que o paciente atingir durante o exercício enquanto a **ZAT** Máxima corresponde ao máximo de **bpm** do paciente durante todo o exercício.

O processo de reabilitação posto em prática neste trabalho será orientado por sessões de treino. Cada dia de treino neste processo corresponderá a uma sessão sendo que cada sessão é formada por sub-sessões (geralmente três: aquecimento, exercício principal e retorno à calma/repouso). Por fim, cada uma das sub-sessões será composta por vários tipos de exercício, proporcionado por diversos tipos de jogos. A relação: sessão - sub-sessão - exercício terá obrigatoriamente uma duração definida, e as respectivas **ZAT**. A figura 4.1 é um exemplo do plano de reabilitação de um paciente com duas sessões de treino.

Sessão	Data	Subsessão	Sequência	Exercício	Zat Min	Zat Max	Zat ref	Duração (m)	Tempo (m)										
									0	5	10	15	20	25	30	35	40		
1	15/03/2013	0-Aquecimento	1	Caminhada	80	140	120	5	■										
		1-Exercício principal	1	Corrida	110	160	140	15		■	■	■							
			2	Saltos	110	150	130	5				■							
			3	Agachamentos	100	150	130	5					■						
			4	Corrida	110	160	140	10						■	■				
		2-Retorno à calma	1	Alongamentos	90	150	130	5											■
2	18/03/2013	0-Aquecimento	1	Caminhada	80	140	120	5	■										
		1-Exercício principal	1	Corrida	110	160	140	10		■	■								
			2	Saltos	110	150	130	5				■							
			3	Corrida	110	165	150	10					■	■					
			4	Agachamentos	100	150	130	5							■				
			5	Corrida	110	160	140	5									■		
2-Retorno à calma	1	Alongamentos	90	150	130	5											■		

Figura 4.1: Gráfico de Gantt que ilustra duas sessões de treino

Este trabalho disponibilizará modos de jogo em ambiente **2D** e **3D** os quais apresentarão diferentes características. Desses dois modos de jogo, derivarão vários tipos de jogos que refletir-se-ão em diferentes tipos de exercícios físicos do paciente.

O responsável pela escolha do jogo que o paciente irá executar, em determinada sessão/exercício, e definição das respectivas **ZAT** de controlo será o profissional de saúde através de uma aplicação independente do jogo. Essa aplicação, que sustentará toda a personalização do jogo para cada paciente de forma singular, será um Painel de Controlo.

Ao paciente competirá autenticar-se no jogo, e seguir as recomendações prescritas previamente pelo profissional de saúde, para qualquer exercício específico, ou instruções genéricas, ao longo dos exercícios propostos.

O **Jogo** será outra aplicação e terá como principal função, através da via do entretenimento incentivar o paciente constantemente para que este consiga atingir os objetivos previamente estabelecidos no Painel de Controlo. O VitalJacket permitirá ao jogo ter conhecimento da frequência cardíaca do paciente em tempo real, enquanto o Kinect transmitirá a posição do paciente em relação ao sensor - serão estas as interfaces de comunicação entre o jogador e o jogo.

Como serão conhecidas as diferentes **ZAT** para determinado exercício/paciente, o jogo

terá a capacidade de saber a cada momento se o paciente está com a frequência cardíaca desejada de deste modo atingir os objetivos. Desse modo, o jogo exigirá mais, ou menos esforço do paciente, tendo em conta o seu comportamento e os seus objetivos. A esse ajuste de dificuldade física, dar-se-á o nome de **carga**.

Nesta aplicação, a **carga** será representada por um número decimal entre 0 (repouso máximo) e 1 (esforço máximo). A carga modificará o estado do jogo alterando variáveis tais como a velocidade, direção e distância de objetos virtuais (que fazem parte do jogo), em relação à posição do paciente no jogo.

Todos os dados considerados relevantes para o processo de reabilitação como por exemplo as definições de treino para cada paciente e os seus resultados em jogo análise comportamental do paciente e definição de treino serão guardados numa **BD SQL Server**.

Para minimizar recursos exigidos e ao mesmo tempo assegurar o funcionamento em larga escala, da solução apresentada, terá que ser garantido que apenas um servidor **BD** e uma instância Painel de Controlo serão suficientes para controlar e gerir o trabalho de múltiplos utilizadores. A disponibilização e comunicação dos dispositivos neste sistema pode ser visto no esquema da Figura 4.2. Neste esquema, à esquerda estão representadas três instâncias do Jogo, com as respetivas ligações ao Kinect e VitalJacket. No centro da imagem, ligado a todos os outros dispositivos, encontra-se representada a Base de Dados. À direita da imagem encontra-se uma instância do Painel de Controlo, operada pelo profissional de saúde, que se liga também à Base de Dados. A ligação da Base de Dados com o Jogo e com o Painel de Controlo é efetuada via *webservice* que retornam *JSON*.



Figura 4.2: Esquema representativo da relação entre sistemas e dispositivos respeitantes a esta solução

4.1.1 Modelo e armazenamento de dados

Para armazenar todos os dados resultantes da parametrização do painel de controlo, registar os acessos ao jogo bem como registos efetuados durante o próprio jogo, utilizar-se-á uma instância de Microsoft SQL Server 2012 Express. Um ponto a favor deste tipo de armazenamento fornecido pela Microsoft, é a possibilidade aliada à facilidade de no futuro a base de dados ser migrada de um servidor físico para a Cloud - Windows Azure.

O servidor Microsoft SQL Server 2012 será configurado pelo protocolo TCP/IP, o que neste caso permitirá a criação de uma ligação remota entre o jogo, painel de controlo e base de dados. No mesmo servidor, será criada uma instância de **SSRS** que servirá para suportar todos os relatórios criados (através de *queries* à base de dados e fórmulas matemáticas para tratamento dos respetivos dados) [8].

Cada paciente na base de dados terá a designação de *Jogador* e será registado com um identificador único (*ID*). Ficará também registado o seu nome, sexo, nome de utilizador, senha de acesso ao jogo, data de nascimento e ainda comentários acerca do mesmo (inseridos pelo profissional de saúde) que poderão ser utilizados para guardar por exemplo um breve diagnóstico do paciente.

O conceito de um *exercício* será a prática de um determinado *jogo*. A acompanhar cada *exercício* poderão estar comentários, por exemplo alertas, e instruções (que aparecerão em forma de pop-up no jogo).

As sub-sessões consistirão na atribuição de *exercícios* ao *jogador*, num determinado período temporal. Cada sub-sessão estará anexada à sessão e ordenada através de um número sequencial. Será também declarada a duração de cada sub-sessão, as **ZAT** que irão controlar o esforço cardíaco do jogador bem como instruções e comentários particulares.

Assim é possível serem apresentados comentários e instruções comuns a todos os utilizadores do jogo (na definição de exercício) em simultâneo com os comentários e instruções particulares do treino de determinado paciente.

Como todos os dados possíveis de serem obtidos beneficiarão correções posteriores neste trabalho, serão registados numa tabela com o nome [*Registo Valores*].

Porque com a evolução deste trabalho surgirão novas métricas interessantes para serem registadas durante o processo de reabilitação, aquando o jogo, optou-se pela estrutura dessa mesma tabela não ser flexível mas sim fixa - ou seja, neste caso será uma tabela com o conceito do modelo de dados Entidade-Atributo-Valor. Com a criação de novas métricas/variáveis, com este modelo não é necessário alterar a estrutura da tabela. Operações de escrita e leitura da tabela já implementadas não requerirão de alterações no momento em que forem adicionadas novas métricas à tabela.

De maneira que este modelo de dados dificulta a leitura e associação de correlações entre variáveis, utilizou-se o método *Pivot* do SQL Server para transformar os resultados numa

forma mais legível.

A Tabela 4.1 é o resultado da *query*:

```
SELECT [IDJogador], [DataRegistro], [VarNome], [VarValor] FROM RegistoValores
```

IDJogador	DataRegistro	VarNome	VarValor
1	2012-10-03 18:19:18	carga	0.89
1	2012-10-03 18:19:18	pulso	130
1	2012-10-03 18:19:21	carga	0.95
...

Tabela 4.1: Tabela *RegistoValores*

Executando a função Pivot do SQL Server, obteremos os resultados visíveis na Tabela 4.2:

```
select
    pvt.[IDJogador], max(pvt.[pulso]) Pulso, max(pvt.[carga]) Carga,
    max(pvt.[DataRegistro]) Data
from RegistoValores
    PIVOT (
        min(VarValor)
        for VarNome in ([pulso],[carga])
    ) pvt
group by
    pvt.[IDJogador], pvt.[DataRegistro]
```

IDJogador	Pulso	Carga	Data
1	130	0.89	2012-10-03 18:19:18
1	NULL	0.95	2012-10-03 18:19:21
...

Tabela 4.2: Resultado da aplicação da função Pivot do Microsoft SQL Server

O diagrama da base de dados será o apresentado na Figura 4.3. **Nota:** Existirão outras tabelas auxiliares aos programas como por exemplo tabelas que garantam ao Painel de Controlo, segurança durante a sua utilização.

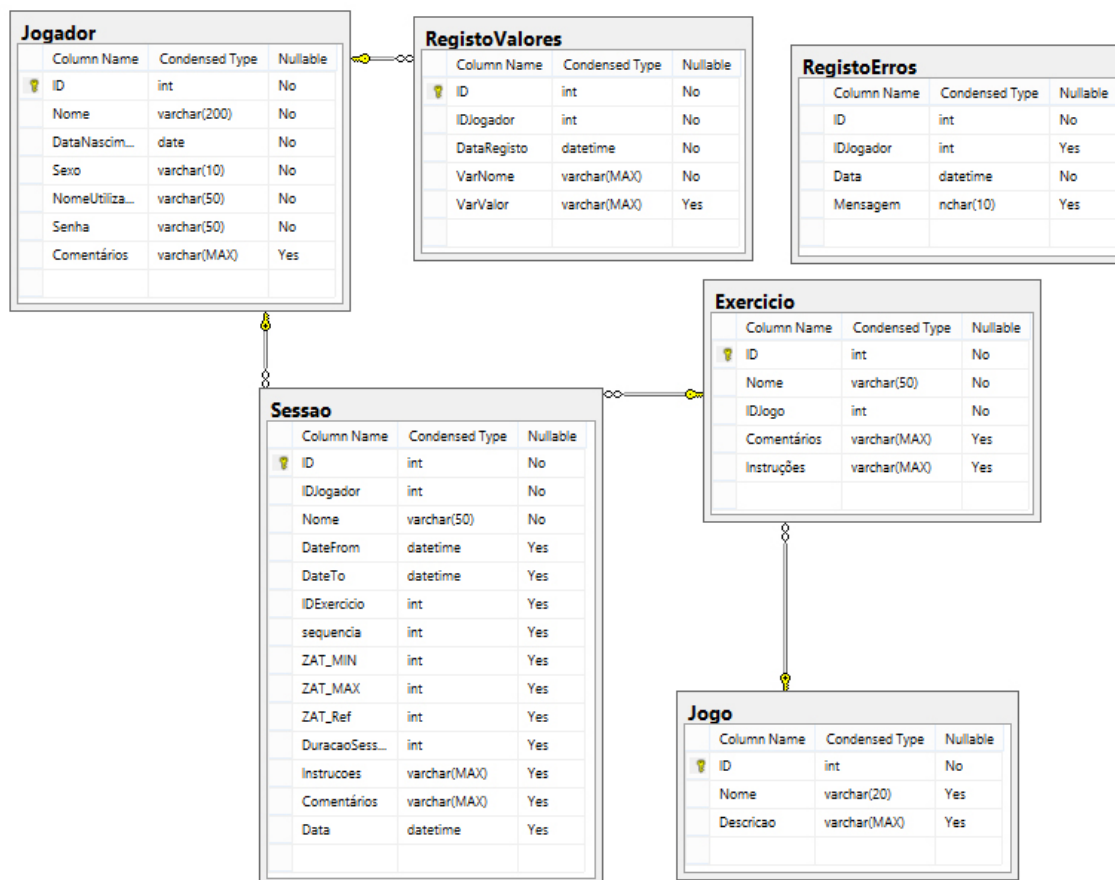


Figura 4.3: Diagrama da base de dados

4.2 Painel de controlo

Para preparar a configuração de cada sessão de reabilitação e aceder a relatórios de desempenho do paciente, o responsável de saúde utilizará o Painel de Controlo. Esta aplicação permitirá ainda que sejam criados perfis de utilizador (que representarão os pacientes no sistema) e parametrizar o plano de exercícios de cada um dos pacientes.

O Painel de Controlo será implementado em tecnologia .Net (Figura 4.4). Será escolhida esta tecnologia entre as demais devido às suas propriedades de integração com o tipo de servidor de base de dados escolhido e sistema operativo alvo, Microsoft Windows, bem como a facilidade de desenvolvimento.

Figura 4.4: Painel de controlo especificando parte do plano de exercício da reabilitação do paciente

Esta aplicação será como um front-end da **BD** uma vez que o Painel de Controlo é apenas uma interface de leitura e escrita da/na **BD**.

As funcionalidades desta aplicação serão:

- **apresentar informação geral relativamente ao jogo ou definições:** será apresentada informação referente ao número de utilizadores registados, número de horas acumuladas de treino dos pacientes, número de exercícios criados no sistema e últimas atualizações da aplicação (na forma de mensagem de texto) - o que funcionará como um *feed* de notícias;
- **criar e manter de contas de utilizador:** será nesta divisão que o profissional de saúde registará o paciente no sistema. Para tal será necessário especificar alguma informação sobre o paciente tais como nome, data de nascimento, *nickname* e respetiva palavra chave (estes dois últimos campos servirão para que o paciente se autentique no jogo antes de iniciar o treino);
- **aceder a relatórios de atividade:** existirão relatórios que possibilitarão ao profissional de saúde analisar o desempenho de qualquer um dos pacientes através de dados expressos em forma de tabelas ou gráficos que mostrem a evolução da frequência

cardíaca em determinado exercício em ordem ao tempo, e a comparação com o desejado. Será também possível verificar o cumprimento do plano de treino.

- **definir exercícios no sistema:** um exercício será a combinação de um determinado jogo com um nome e um texto que servirá de descrição ao mesmo. O mesmo tipo de jogo poderá estar associado a mais do que um exercício para por exemplo despertar diferentes comportamentos no jogador. Nesse caso o jogo iria ser o mesmo mas como as instruções ou mesmo comentários inerentes ao jogo diferentes. Desta forma, dois pacientes a executarem dois exercícios diferentes, embora estejam a jogar o mesmo jogo, poderão ter objetivos de treino diferentes. Os jogos que serão associados aos exercícios, irão conter uma descrição que tal como o respetivo nome, apenas poderá ser criada e editável pelo programador. Com o aparecimento de novos tipos de jogo, poderão ser criados novos tipos de exercício.
- **definir sub-sessões de treino:** é essencial no processo de reabilitação de um paciente que as sub-sessões de exercícios possam ser únicas e adaptadas às características desse paciente. Sendo uma sessão um conjunto de sub-sessões, uma sub-sessão pode ser um treino relativo ao "retorno à calma".

Cada sub-sessão terá uma validade temporal e estará associada a uma série de exercícios (ordenada por uma sequência numérica). Por sua vez, cada exercício sequencial terá as respetivas *ZAT* de treino e duração. A funcionalidade da validade temporal é para assegurar a possibilidade do profissional de saúde por exemplo manter o plano de treino "retorno à calma" durante determinado tempo.

Por fim, é ainda possível definir instruções e comentários específicos para cada combinação exercício - sequência - sessão como é visível na Figura 4.4.

Durante a inserção de dados sobre o treino de determinado paciente, a aplicação confirma a validade das *ZAT* Máxima e Mínima inseridas. Para tal, a aplicação valida essa informação através da idade do paciente (derivável da data de nascimento inserida no registo) - Equação 4.1 - e da fórmula de Karvonen - Equação 4.2, sendo que a variável i corresponderá à intensidade do exercício. Neste caso, como é um Serious Game para pacientes cardíacos, a intensidade será entre 40% a 60%. [3].

$$FC_{max} = 220 - idade \quad (4.1)$$

$$FC_{treino} = FC_{repouso} + i * (FC_{max} - FC_{repouso}) \quad (4.2)$$

Toda e qualquer operação executada no Painel de Controlo que terá consequências na base de dados sobre a forma de comandos de edição (*insert*, *update* ou *delete*) assim como *queries* mais complexas que envolvam precedência de ações na BD (por exemplo um *select* imediatamente após uma operação *insert*) estarão protegidos por *Rollback Transactions* ²

²"Rolls back an explicit or implicit transaction to the beginning of the transaction, or to a savepoint

em caso de falha de uma delas.

Outra funcionalidade do Painel de Controlo é a possibilidade deste ser utilizado em várias instâncias em simultâneo.

No entanto, essa utilização poderá ser limitada pela aplicação pois será utilizado um sistema de isolamento da transação de parametrização (pelo profissional de saúde), recorrendo a semáforos em caso de acesso concorrente. Esse problema de concorrência a dados poderia fazer com que os dados que um administrador estaria a alterar já teriam sido alterados por outro. Um *timeout* de 20 minutos/sessão servirá para corrigir eventuais saídas inesperadas no decorrer da aplicação e até mesmo expirar a sessão de alguém que se tenha esquecido de fechar a sessão. Esse fluxo de controlo sobre as alterações está descrito na Figura 4.5.

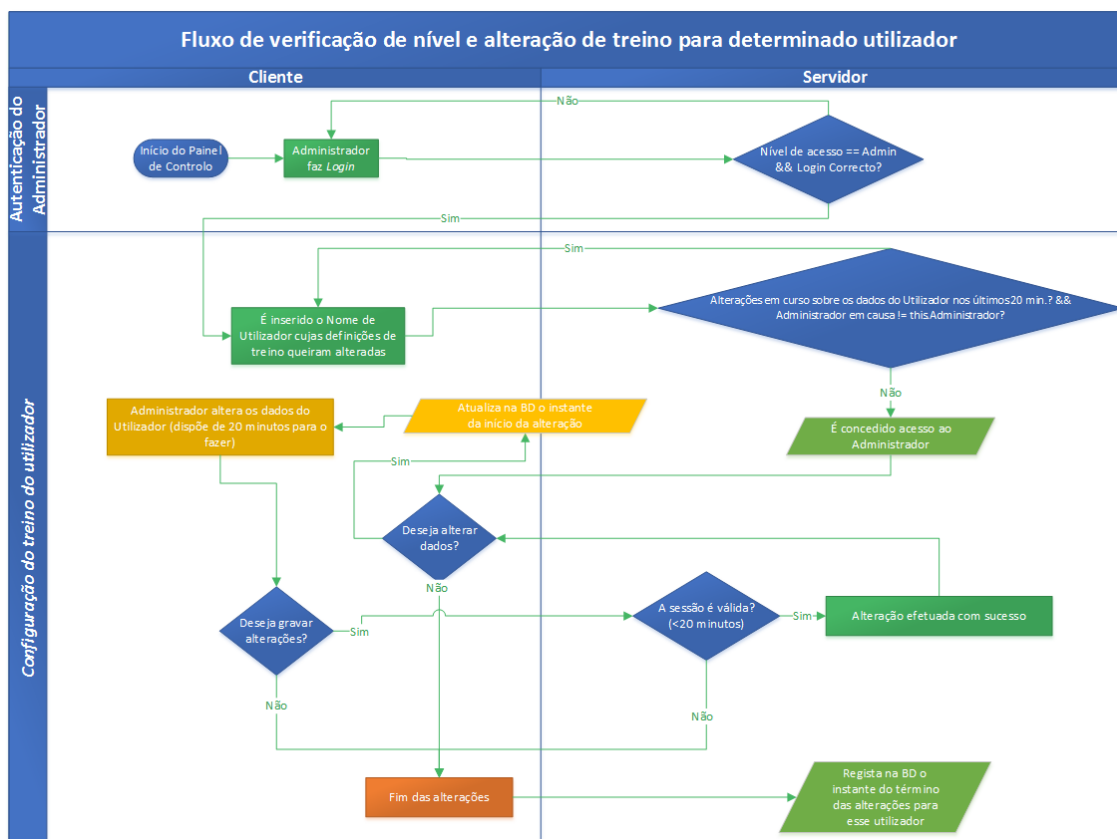


Figura 4.5: Fluxograma referente ao pedido de alterações de dados de um determinado utilizador (paciente), por parte de um administrador (profissional de saúde). Ao Cliente corresponderá ao que é efetuado de modo direto pelo administrador no Painel de Controlo enquanto as ações do Servidor corresponderão às tomadas de forma autónoma pela aplicação ou BD.

inside the transaction. You can use ROLLBACK TRANSACTION to erase all data modifications made from the start of the transaction or to a savepoint. It also frees resources held by the transaction.
 (Microsoft Transact-SQL Reference (Database Engine))

4.3 Concepção do *Serious Game* para Reabilitação Cardíaca

”A video game is an electronic game that involves human interaction with a user interface to generate visual feedback on a video device.”

(Definição de “Video Game” em Wikipedia)

Independentemente da área de contexto e metodologia utilizada, é necessário definir dois fatores em qualquer jogo, inclusive num **SG**: o objetivo e o público alvo. Desta forma, este trabalho definiu os pacientes com distúrbios cardiovasculares como público alvo, e como principal objetivo fazer com que esses mesmos utentes atinjam os pré-determinadas frequências cardíacas alvo durante todas as fases do treino. Para cumprir estes objetivos, o jogo contará com o auxílio dos sensores VitalJacket e Kinect.

Tendo em conta os objetivos e público alvo deste jogo, e antes de serem escolhidas as ferramentas e metodologias de desenvolvimento, serão definidos os principais conteúdos de um jogo tais como: **história**, **grafismo**, metodologia de **IA** e **fluxo** do jogo.

Embora a **história** de um jogo seja sempre um ponto importante para o jogador, pois é algo que o motiva a voltar ao jogo para descobrir como acaba, neste trabalho é algo não fulcral (tendo em conta as outras partes deste jogo). Uma narrativa quando é criada, tem que ter um fim declarado, e neste caso como não é certo o número de sessões que o jogador irá executar, à partida seria complicado criar uma narrativa completa. Mas para “substituir” a história no processo de incentivo do jogador a voltar ao jogo e a fazer cada vez melhor (para além dos benefícios próprios diretos no mesmo) opta-se por se criar uma comparação/avaliação automática (por parte do jogo) face à prestação do paciente (no final de cada sessão) comparando com as prestações anteriores.

O **grafismo** que cobrirá toda a parte visual do jogo, bem como o áudio que devolverá *feedback* ao utente, terão que ser de simples compreensão e aceitação pelas diferentes faixas etárias. O ideal seria criar um personagem e um pacote sonoro para crianças, outro para adultos e outro para idosos, mas não será feito dessa forma devido aos recursos que essa opção acarretaria. Irão ser utilizados gráficos e sons minimalistas que possibilitem aos utilizadores uma fácil identificação.

Quanto à Inteligência Artificial (**IA**) aplicada a este jogo, terá que ser sempre relacionada com os resultados da frequência cardíaca do paciente face ao esperado. O jogo terá que ser suficientemente inteligente para incentivar o utente a esforçar-se mais ou menos consoante o seu estado atual e o objetivo a atingir. Para que tal possa acontecer desenvolver-se-á um algoritmo que calcule a **carga** de esforço a aplicar ao paciente para que este se aproxime dos objetivos, sempre tendo em conta os limites das **ZAT**.

Tal como todos os procedimentos, todos os jogos necessitam de um **fluxo** que funcionará como uma máquina de estados. Como o jogo terá a possibilidade de ser utilizado por diversos utentes, ao mesmo tempo, com diversas definições, o ideal para o programador é que este seja apenas uma aplicação genérica e não uma aplicação única, atribuída a cada

paciente após a configuração da mesma por parte do programador. À semelhança do que acontece em todos os jogos, nomeadamente os jogos multi jogador, o jogo a distribuir pelos jogadores será sempre o mesmo, e o que criará diferença na jogabilidade, para além do jogador, serão as definições atribuídas, durante o jogo, a cada jogador. Neste jogo acontecerá o mesmo e as definições de jogo (sessões de exercícios com instruções e comentários) serão “carregadas” para o jogo através de uma inevitável autenticação no jogo por parte do paciente. Tendo em vista a utilização do jogo por pessoas de todas as faixas etárias, o fluxo de jogo deverá ser simplista ao máximo, ao ponto de exigir ao paciente o mínimo número de *clicks* ou teclas pressionadas.

Qualquer registo, ou consulta de dados, ou cálculos efetuados pela aplicação deverão ser transparentes à vista do utilizador para que este se concentre o máximo possível na sua prestação física durante o jogo. É importante que o jogo consiga registar dados sobre o jogador ao longo da sessão de treino, para que os resultados possam ser mais tarde estudados e dessa forma, otimizar a configuração do plano de treino do jogador, ou até mesmo otimizar o SG. Quanto à leitura dos dados, se a aplicação for autónoma (será o pretendido), esta deverá ser capaz de fazer a leitura do plano de exercícios na base de dados e sozinha parametrizar todas as componentes de jogo de acordo com esse mesmo plano principalmente tendo em conta a sequência e duração de cada exercício/jogo em determinada sessão.

A Figura 4.6 ilustra o fluxo de jogo desejável para este SG cumprindo os requisitos apresentados.

Para implementar o *Serious Game para Reabilitação Cardíaca* o ideal seria utilizar um programa que servisse para criar e desenvolver um videojogo, ou seja recorrer a um motor de jogo.

4.4 Motor de jogo

”A game engine is a system designed for the creation and development of video games. The leading game engines provide a software framework that developers use to create games for video game consoles, mobile devices and personal computers. The core functionality typically provided by a game engine includes a rendering engine (“renderer”) for 2D or 3D graphics, a physics engine or collision detection (and collision response), sound, scripting, animation, artificial intelligence, networking, streaming, memory management, threading, localization support, and a scene graph.”

(Definição de “Game Engine” em Wikipedia)

Durante a concepção do projecto foram testados dois motores de jogo distintos com o objectivo de se escolher o que melhor se adaptaria à solução do problema. Os motores de jogo testados são o XNA Game Studio (Microsoft) (secção 4.4.1) e o Unity 3D (secção 4.4.2). Na secção 4.4.3 são apresentadas diferenças entre os dois motores de jogo.

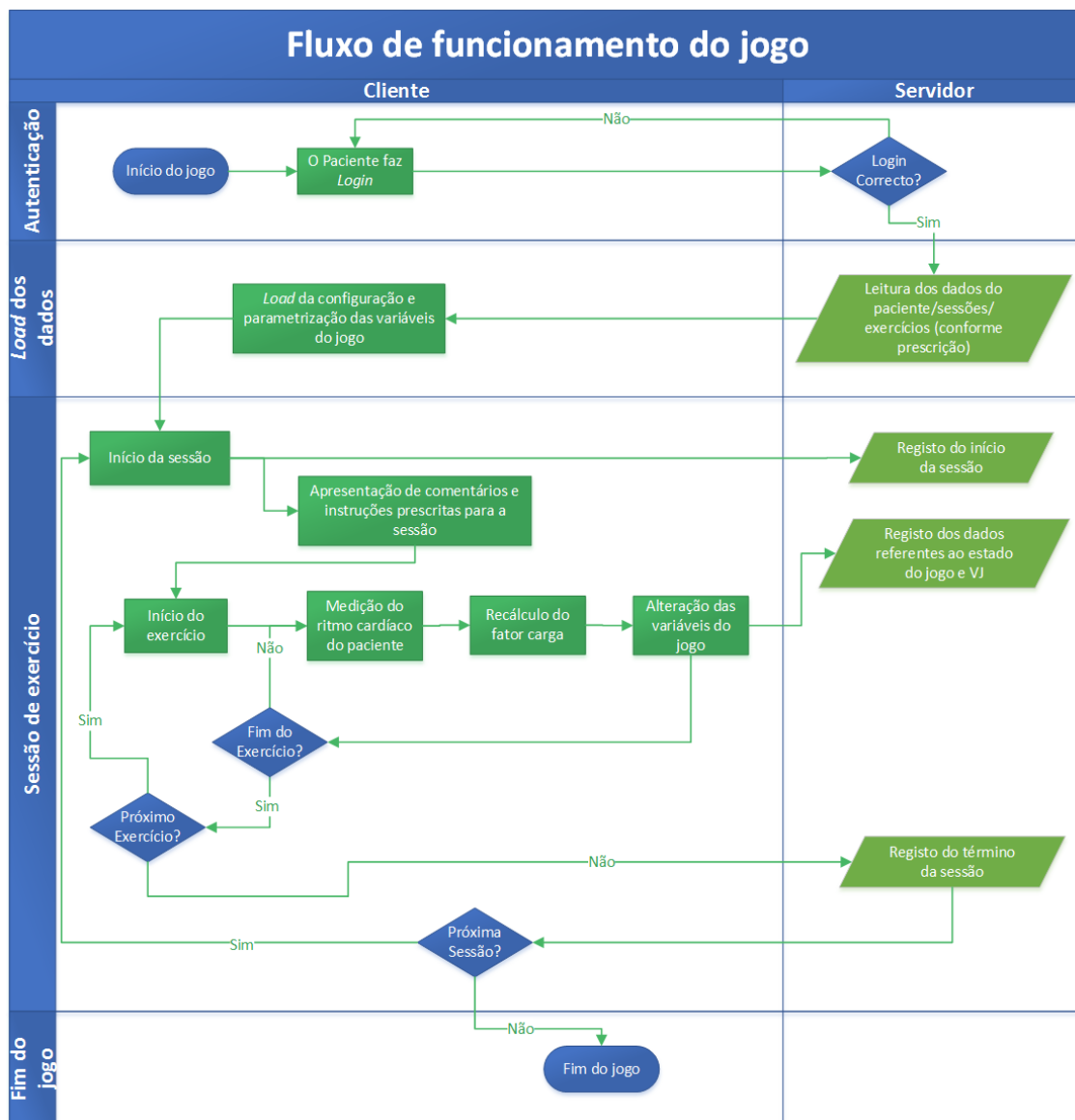


Figura 4.6: Fluxo de funcionamento geral do jogo

4.4.1 XNA Game Studio

O motor de jogo XNA Game Studio é uma *framework* desenvolvida pela Microsoft com o objetivo de servir o programador, na criação de jogos para as plataformas Windows, Windows Phone e Xbox 360. A linguagem de programação é C# e funciona de forma directa sobre a plataforma .Net Framework (também da Microsoft). O IDE de desenvolvimento é o Visual Studio e a versão XNA Game Studio Express é gratuita para fins académicos.

4.4.2 Unity 3D

O Unity 3D possibilita o desenvolvimento de aplicações 2D e 3D para a maioria das plataformas existentes no mercado (PSX, Windows, Mac OS, Wii, Web, iOS). As linguagens de programação suportadas pelo Unity 3D são Boo, C# e JavaScript. Este motor de jogo oferece ainda um editor gráfico que possibilita ao programador um controlo total sobre o ambiente gráfico da aplicação. Esta funcionalidade permite que ajustes de ponto de vista sobre o jogo, luzes, dimensões de objetos, entre outros, possam ser modificados de forma visível e intuitiva pelo programador (Figura 4.7). Os *scripts* de execução (em qualquer uma das três linguagens suportadas) são associados de forma direta a esses objetos. Os três métodos principais do Unity 3D são o Start(), Awake() e Update().

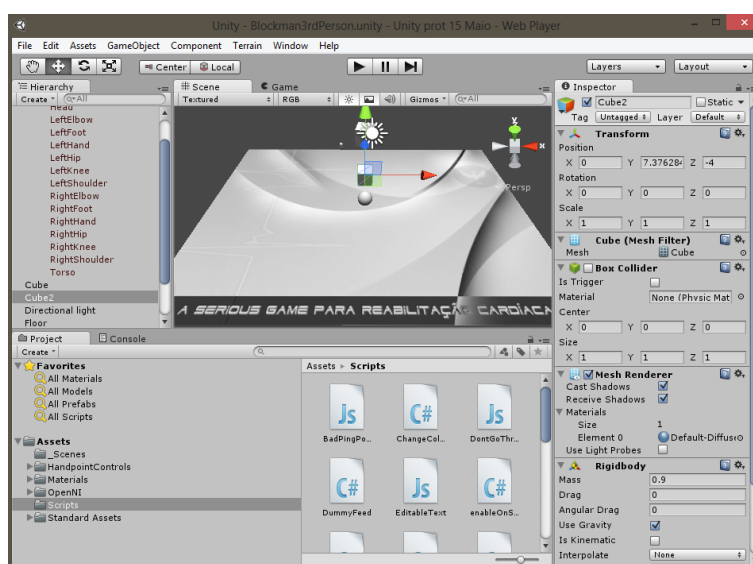


Figura 4.7: Interface de desenvolvimento Unity 3D

4.4.3 Decisão sobre o motor de jogo a utilizar

O XNA Game Studio dispõe de bibliotecas que facilitam a comunicação com o Kinect, e é totalmente compatível com o Kinect por intermédio do Kinect SDK.

Por sua vez, para permitir que o Kinect pudesse ser utilizado como uma interface háptica no Unity, seria fundamental recorrer a bibliotecas externas. A solução seria recorrer a um SDK para Kinect, que fosse também compatível com o Unity, e a principal alternativa ao Kinect SDK encontrada foi o OpenNI SDK.

Este SDK alternativo dispõe de todas as funcionalidades oferecidas pelo Kinect SDK da Microsoft. Tem como vantagem o facto de ser compatível com multi-plataforma e de algumas das funcionalidades estarem mais aperfeiçoadas comparativamente ao Kinect SDK. O OpenNI é um produto *Open Source* disponibilizado pela PrimeSense, e conta com a colaboração da comunidade aberta formada por utilizadores e programadores que tratam

Coordenada	Kinect SDK	OpenNi SDK
X	0,060212	0.030242
Y	0,057867	0.031427
Z	0,061247	0,028786

Tabela 4.3: Comparativo das médias do desvio padrão OpenNi e Kinect SDK por coordenada

de melhorar a qualidade deste SDK. Para facilitar a utilização do Kinect no Unity via OpenNI, também existem bibliotecas disponíveis.

A biblioteca gratuita mais popular é o Zigfu e consiste num conjunto de scripts C# que permite ao programador obter um controlo direto dos “*joints*” de um modelo 3D em tempo real.

Na fase inicial de desenvolvimento deste SG foram criados um protótipo de conectividade entre cada um dos motores de jogo e o Kinect.

Realizaram-se testes nas mesmas condições (luminosidade, modelo de teste, computador e distância ao sensor) sobre os dois protótipos.

Embora em qualquer um dos protótipos se verificasse que seria possível atingir os objetivos, notou-se que o jogo de tecnologia XNA apresentava um maior *delay* entre os movimentos do utilizador e a assimilação por parte da aplicação comparativamente ao que acontecia no Unity 3D. Durante os testes notou-se também que a posição dos pontos do *skeleton tracking* (ST) na versão do Kinect SDK apresentava uma maior dispersão na precisão das coordenadas face à versão com OpenNI + Zigfu (Unity).

Utilizou-se um modelo humano em repouso para se aferir o ruído de informação lida pelo Kinect e trabalhada via Kinect SDK e OpenNI. Registaram-se os valores lidos pelo Kinect diretamente num ficheiro de texto, a partir das duas versões de software desenvolvidas. Como para cada ponto do ST existem três coordenadas (ponto 3D), calculou-se o desvio padrão sobre a amostra de dados sobre cada coordenada/ponto. No fim, calculou-se a média dos desvios padrões agrupadas por coordenada (Tabela 4.3).

Desta forma é possível aferir que o OpenNI apresenta uma maior capacidade de lidar com o ruído durante o processo de leitura por parte do Kinect, nas condições em que foram realizados os testes.

Considerando estas diferenças e ainda o facto do Unity 3D possibilitar o desenvolvimento através do editor gráfico, fez com que o motor de jogo adotado para o implementação deste SG fosse o Unity 3D.

4.5 Implementação do Serious Game para Reabilitação Cardíaca

Após ter sido escolhido o motor de jogo, para definir a estrutura da solução foi necessário reforçar a análise às necessidades do jogo, presentes no fluxograma da Figura 4.6.

Seria necessário criar uma estrutura/hierarquia para organizar os recursos (*Assets*) do jogo (Figura 4.8). Seguindo os princípios do Unity, cada vista sobre o jogo seria uma *Scene*, nas quais objetos assumiriam diversos tipos de comportamentos e propriedades.

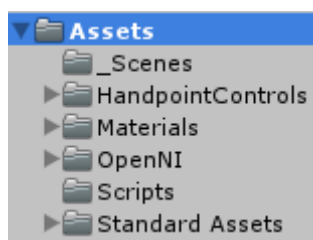


Figura 4.8: Estrutura do jogo no Unity

Todas as ações assumidas pelos objetos (a representação do paciente no jogo é também um objeto) são definidas por *scripts* programados em Javascript (versão Unity).

Cada objeto de formato tridimensional utilizado no jogo que não seja um sólido platônico, seria criado no Blender e a respectiva textura no Adobe Photoshop.

A relação entre o modelo tridimensional que representará o jogador no jogo e os "joints" do Kinect é feito de forma direta no script de programação fornecido pelo OpenNi SDK (Figura 4.9).



Figura 4.9: Interface de ligação entre o modelo e o jogador - via OpenNI SDK

Durante o desenvolvimento do jogo, foram criados diferentes ambientes físicos e multi-dimensionais (2D - Figura 4.10 e 3D - Figura 4.11) com o objetivo de despertar diferentes reações físicas no paciente. No caso do jogo 2D é necessária uma abstração da coordenada y o que implicou ter que modificar a projeção das coordenadas x e z .

4.5. IMPLEMENTAÇÃO DO SERIOUS GAME PARA REABILITAÇÃO CARDÍACA³⁵



Figura 4.10: Menu principal do jogo

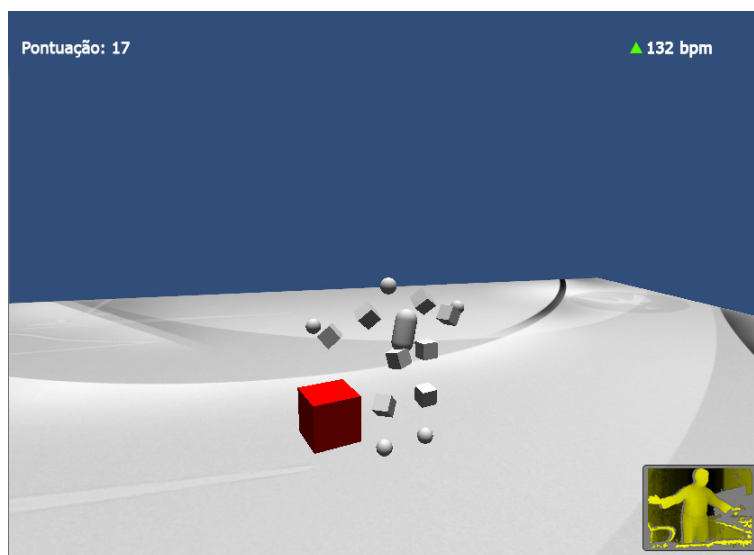


Figura 4.11: Ambiente do modo de jogo 3D

Uma função do tipo *insert* ou *update* executada na **BD** por parte do Unity 3D será:

```
function executeQuery(query2execute: String)
{
    dbcon = new SqlConnection(connectionString);
    dbcon.Open();
    var dbcmd: IDbCommand = dbcon.CreateCommand();
    dbcmd.CommandText = query2execute;
    dbcmd.ExecuteReader();
    dbcon.Close();
}
```

Para alterar a posição de um objeto no decorrer do jogo, de forma aleatória atribui-se ao mesmo um *script* deste género:

```
var target : Vector3;
var direction = Vector3(-1,0,0);

function Start() {
    direction = Vector3(Random.Range(-1.0,1.0),0,Random.Range(-1.0,1.0));
    transform.position.z = Random.Range(-5.5,5.5);
    transform.position.x = Random.Range(-7,7);
    target = direction;
}
```

4.5.1 Aquisição dos dados provenientes do VitalJacket

Para que fosse possível interpretar os dados provenientes do VitalJacket, foi necessário contactar a BioDevices. A empresa responsável pelo produto, forneceu a informação necessária, para interpretar os dados lidos pelo protocolo Bluetooth (quando ligado o VitalJacket ao computador). Essa informação consiste numa **DLL** pertencente à BioDevices (criada para interpolar valores do **VJ**) e uma matriz que funciona como um dicionário que traduz a informação interpelada em informação prática ao programa.

A ligação do VitalJacket ao computador, é efetuada pelo intermédio de uma porta COM no computador.

Infelizmente a versão do VitalJacket utilizada neste sistema é a mais simples: 1L. Desta forma não é possível que a aplicação assegure na totalidade a segurança do paciente uma vez que não é possível detetar risco de ataque cardíaco por exemplo, pois só é possível ler informação relativa à **FC**.

4.5.2 Sistemas de controlo de esforço

Para controlar o esforço do jogador durante o exercício, existem dois métodos:

- Modelo de carga - Consiste num algoritmo de aplicação de um esforço ao jogador (carga) através das variáveis do jogo (objetos) incentivando o jogador a alterar o seu estado de reação ao jogo. Se o esforço do paciente tiver de aumentar, o jogo torna-se mais rápido, há mais projéteis, e por exemplo no modo de jogo 3D os cubos tornam-se mais pesados e por isso mais difíceis de destruir. O algoritmo que modela este método será explicado no capítulo seguinte.
- Indicação visual - Durante o decorrer do jogo, a cada 3 segundos, após o a **FC** do jogador ser lida, é comparada com a **ZAT** de referência. Sempre que a distância entre essas duas medidas é superior a 3 **bpm**, o jogo alertará o paciente através de uma

4.5. IMPLEMENTAÇÃO DO SERIOUS GAME PARA REABILITAÇÃO CARDÍACA 37

seta com orientação vertical, acerca do seu comportamento. Caso essa seta tenha sentido cima-baixo, significará que o jogador terá que diminuir a sua frequência cardíaca, caso a seta tenha sentido baixo-cima, significa o contrário (Figura 4.12). Esse intervalo de 3 bpm funcionará como uma margem de conforto para o jogador. Caso não existisse essa margem o jogador teria que constantemente ajustar a sua FC mesmo estando muito próximo da ZAT de referência.



Figura 4.12: Esta seta indica que o jogador está com pelo menos 3 bpm abaixo da prescrição do exercício

O modelo de carga, detalhado no Capítulo 5, é implementado no jogo através da função *modelo* presente no código:

```
static var constanteInt:double = 1000000.0;

function modelo(pulso:int , a:double , b:double , ymin:double , ymax:double ,
xref:double , k:double)
{
    y = pulso * constanteInt; //frequencia cardiaca lida
    r = xref * constanteInt; //ritmo de referencia
    xr = (-a * (ymin - r)) / b;
    xr = xr * constanteInt;
    x = saturacao(k * (r - y) + xr);
    Debug.Log(" Pulso:~" + pulso + "~~Carga:" + x);
    return x;
}

function saturacao(sat:double)
{
    sat = sat / constanteInt;
    if (sat > 1) {
        sat = 1;
    } else if (sat < 0) {
        sat = 0;
    } else
        sat = sat;
    return sat;
}
```

4.5.3 Instalação e execução do jogo

A instalação do jogo é efetuada a partir de um ficheiro apropriado: *.msi* e foi gerada pelo Intel AppUp com o auxílio do Visual Studio. Desta forma, será apenas necessário que a pessoa que instalar o jogo em determinado computador, não tenha que instalar componentes à parte como por exemplo o OpenNI SDK.

Ao executar o jogo, o jogador terá que confirmar a porta COM que estará emparelhada com o VitalJacket e autenticar-se na aplicação.

4.5.4 Jogabilidade

Para iniciar a sessão de treino, basta emparelhar o VitalJacket com o computador, executar o jogo e iniciar sessão no mesmo. Após o jogo começar, será pedido ao jogador que calibre a sua posição relativamente ao Kinect.

A posição gráfica do paciente no modo de jogo 2D (Figura 4.13) é representada pelo círculo branco. Para pontuar é necessário apanhar o objeto cor-de-rosa e escapar aos objetos verdes. Quando o paciente alcança o objeto cor-de-rosa, é instanciado um novo objeto no mapa, noutra local. A representação gráfica do jogo equivale a uma vista aérea de uma sala na qual o paciente se move.

No modo de jogo 3D (Figura 4.11) o paciente tem que intersestar e/ou escapar a objetos (dependendo do tipo). Nos diversos modos de jogo, um aumento de carga refletir-se-á num incremento de velocidade, número e/ou direção dos objetos.

Em jogos mais elaborados, modo 3D, o jogador terá que mexer determinadas partes do seu corpo para intersestar os objetos.

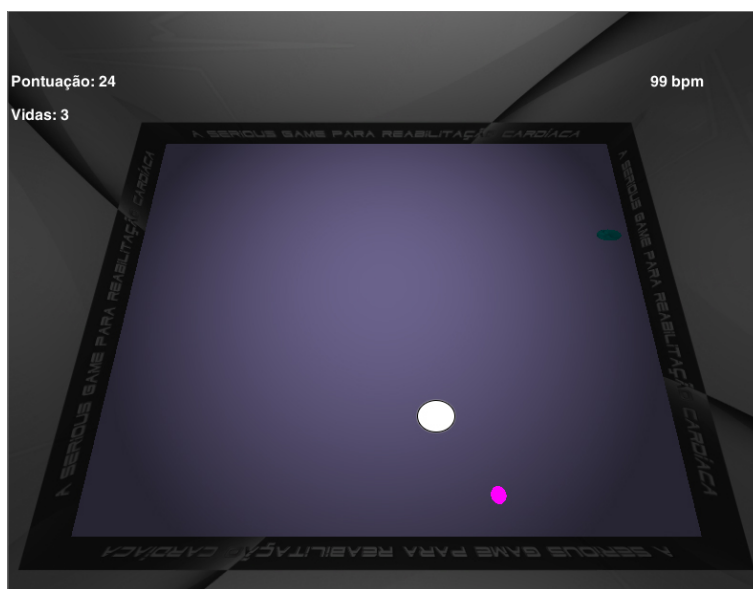


Figura 4.13: Ambiente do modo de jogo 2D

Capítulo 5

Modelo e ajuste de carga

O estudo do metabolismo do ser humano é bastante complexo devido ao elevado número de variáveis que o constituem e influenciam. Para fazer com que o ritmo cardíaco do paciente x atinja a Zona Alvo Treino de referência em determinado exercício, o jogo aplica uma carga no esforço do paciente. Essa carga aplicada traduz-se na exigência física que o jogo exige ao paciente, para o elevar/descer/manter em determinado ritmo.

O comportamento da frequência cardíaca x é modelado por uma equação diferencial de primeira ordem, onde o estado x representa o ritmo cardíaco e u é uma variável que define o nível de esforço ou dificuldade que o jogo apresenta ao paciente - Equação 5.1. A variável $bpm_{repouso}$ corresponde ao número de bpm do paciente em estado de repouso.

$$\frac{dx}{dt} = a(x - bpm_{repouso}) + bu \quad (5.1)$$

Nesta Equação 5.1, a é um número inferior a zero enquanto b é um fator a definir no jogo para amplificar a carga u . Sendo esta equação apenas um modelo, a e b servem para moldar o modelo fisiológico do paciente através de fatores no jogo como por exemplo, o tempo que o paciente deverá demorar até que atinja a carga de referência. Como nas ciências da computação não é possível trabalhar equações com variáveis contínuas, recorreu-se à discretização na Equação 5.3. Como a discretização consiste em dividir um intervalo de tempo num número limitado de “instantes” temporais de curta duração, é possível obter uma aproximação à derivada como mostra a Equação 5.2.

$$\frac{dx}{dt} \approx \frac{x(t + \Delta t) - x(t)}{\Delta t} \quad (5.2)$$

Substituindo na Equação 5.1 a derivada ao tempo pela aproximação à mesma (Equação 5.2) obtém-se a Equação 5.3:

$$\frac{x(t + \Delta t) - x(t)}{\Delta t} = a(x(t) - bpm_{repouso}) + bu(t) \quad (5.3)$$

$$\Leftrightarrow x(t + \Delta t) = x(t) + \Delta t a x(t) + \Delta t (-abpm_{repouso} + bu(t))$$

Obter a carga de referência (u_{ref}) implica que o ritmo cardíaco obtido seja igual ao ritmo cardíaco objetivo (x_{obj}) - Equação 5.4.

$$\frac{dx}{dt} = 0 \Leftrightarrow a(x_{obj} - bpm_{repouso}) + bu_{ref} = 0 \quad (5.4)$$

Resolvendo em ordem a u :

$$u_{ref} = -\frac{a(x_{obj} - bpm_{repouso})}{b} \quad (5.5)$$

Como o modelo de carga será aplicado a um modelo humano, e tendo em conta que este não reagirá de forma diretamente proporcional ao fator carga de referência aplicado pelo jogo, para obtermos a carga a aplicar é ainda necessário acrescentar uma oscilação à carga de referência que reflita o comportamento do modelo fisiológico em questão.

Desta forma é necessário um amplificador do fator carga, variável K , e ao mesmo tempo influenciar esse amplificador tendo em conta a distância entre a frequência lida em determinado instante (x) e a frequência cardíaca objetivo (x_{obj}). Modificar a variável K resultará, em obter um ajuste de carga mais ou menos sensível. Podemos desta forma definir a carga u como mostra a Equação 5.6.

$$u = u_{ref} + K(x_{obj} - x) \quad (5.6)$$

Por sua vez, u atua sobre a pessoa e é então obtido (lido) uma novo ritmo cardíaco x (Figura 5.1).

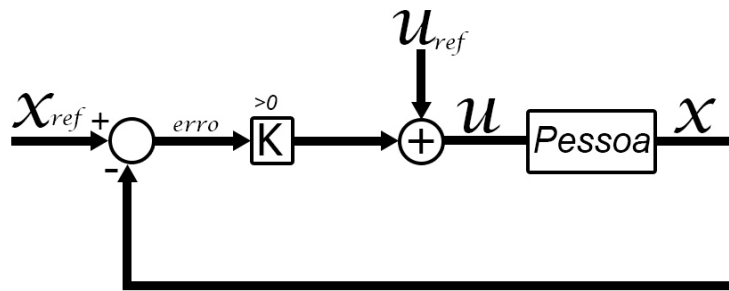


Figura 5.1: Ciclo representativo dos ajustes de carga no jogo

Uma simulação do comportamento da carga e ritmo cardíaco seguindo este algoritmo é o exemplo da Figura 5.2.

No gráfico de posição superior da Figura 5.2 repara-se que o paciente iniciou o exercício com um ritmo cardíaco de 60 bpm. Ao longo do tempo foi aplicada uma carga (gráfico de baixo) que aproximou o ritmo cardíaco do paciente, ao objetivo/ZATref de 140 bpm. Note-se que como o metabolismo do paciente é algo irregular, aconteceram quebras e aumentos no ritmo, o que refletiu variações na carga aplicada.

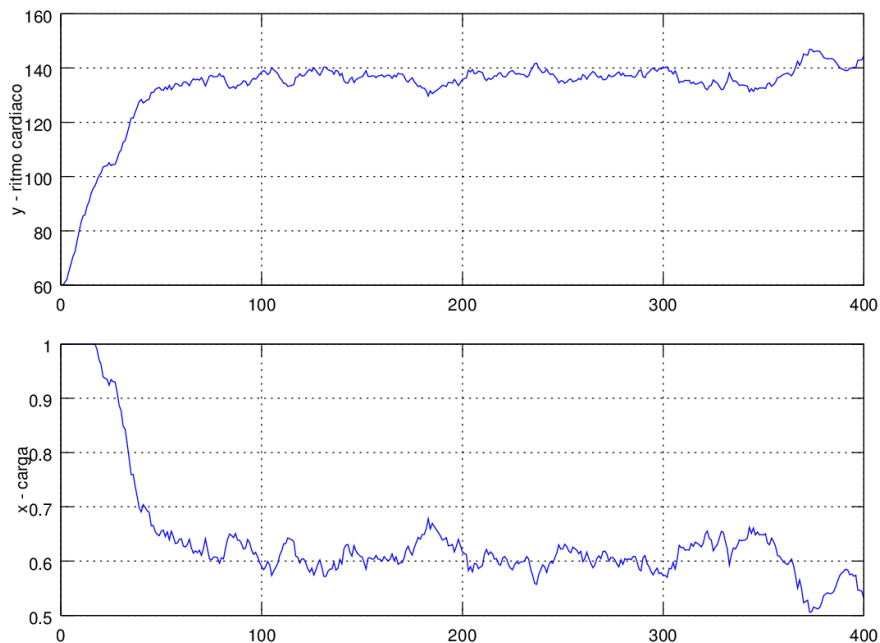


Figura 5.2: Esta figura mostra uma simulação estocástica da resposta de um paciente com o sistema de controlo

Sem a amplificação da carga (fator K) uma simulação do comportamento da carga e ritmo cardíaco, é o exemplo da Figura 5.3. Nessa simulação, comparando com a da Figura 5.2,

é visível que o paciente demora mais tempo a atingir o objetivo, e ainda que após atingir o objetivo, não se mantém tão próximo deste com regularidade.

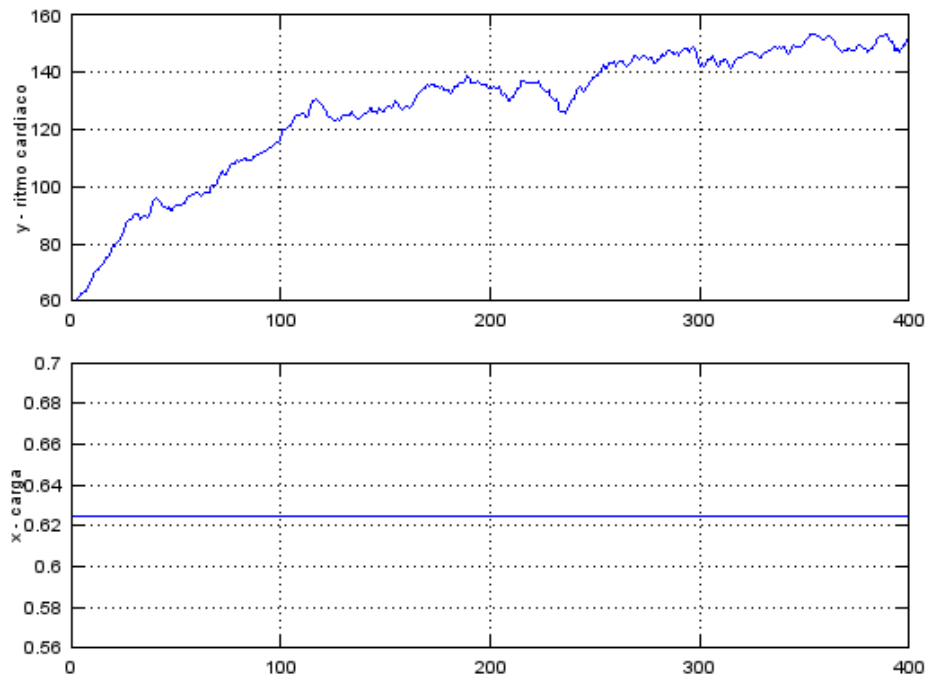


Figura 5.3: Esta figura mostra uma simulação estocástica da resposta de um paciente com o sistema de controlo sem o fator de amplificação ($K = 0$)

Capítulo 6

Resultados experimentais

De forma a verificar o funcionamento do jogo e testar o comportamento deste, realizou-se uma simulação para avaliar o comportamento deste a nível de performance tanto do *software* como do modelo de ajuste de carga implementado (Figura 6.1).

No modelo de carga integrado na aplicação, afetaram-se as variáveis K, a e b com os valores 0.01, 0.16 e 2.0 respetivamente. Esta combinação de valores foi a melhor encontrada para o modelo humano que testaria este sistema - resultado de experimentação.

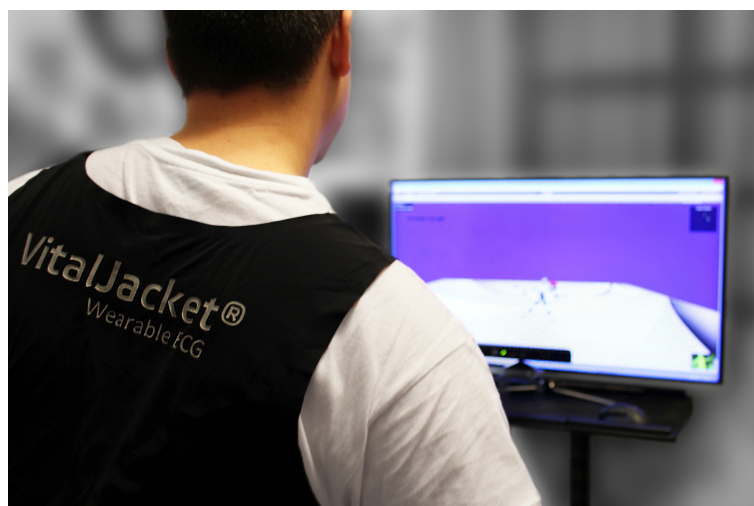


Figura 6.1: Experimentação do sistema desenvolvido

O sistema no qual o jogo foi preparado tinha como características:

- servidor **BD** Web Edition SQL Azure database (recorreu-se à *cloud* para emular um servidor externo)
- computador *desktop*:
 - CPU: E8400
 - Memória RAM: 4GB DDR2
 - GPU: Nvidia 8800GT
 - OS: Microsoft Windows 7 x64
- programa FRAPS para avaliar a performance em *frames per second* do jogo
- Microsoft Kinect v1.1
- VitalJacket 1L

Foi configurada no Painel de Controlo uma sessão de treino, recorrendo a uma instância do jogo em modo 2D durante os primeiros 24 minutos de treino e modo 3D no resto do tempo e com as características presentes na tabela 6.1.

Sequência	Tipo de exercício	Duração (minutos)	ZATRef
1	Aquecimento	10	100
2	Aeróbico	14	140
3	Força Resistente	6	100
4	Retorno à calma	10	80

Tabela 6.1: plano de exercício da sessão de teste

No final da sessão os dados registados sobre os resultados dessa sessão foram traduzidos num gráfico que mostra compara os **bpm** efetivos do jogador face ao objetivo (ZATref) - Figura 6.3. O fator carga de esforço em ordem ao tempo de duração da sessão aplicada pelo jogo, está expressa na Figura 6.4.

Relativamente ao teste do software FRAPS sobre a fluidez de jogo, os resultados obtidos foram os apresentados na Tabela 6.2.

	MIN	MAX	AVG
FPS	54	67	62

Tabela 6.2: Resultado do teste *frames per second* realizado por FRAPS



Figura 6.2: Sessão de exercício em execução

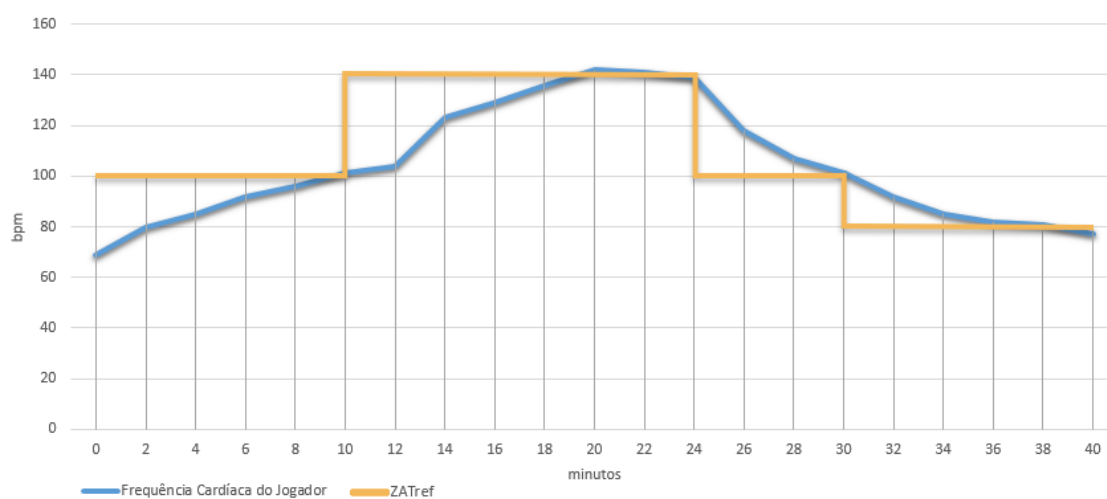


Figura 6.3: Comportamento do ritmo cardíaco do jogador (linha azul) face ao objetivo (ZATref - linha laranja) durante o exercício

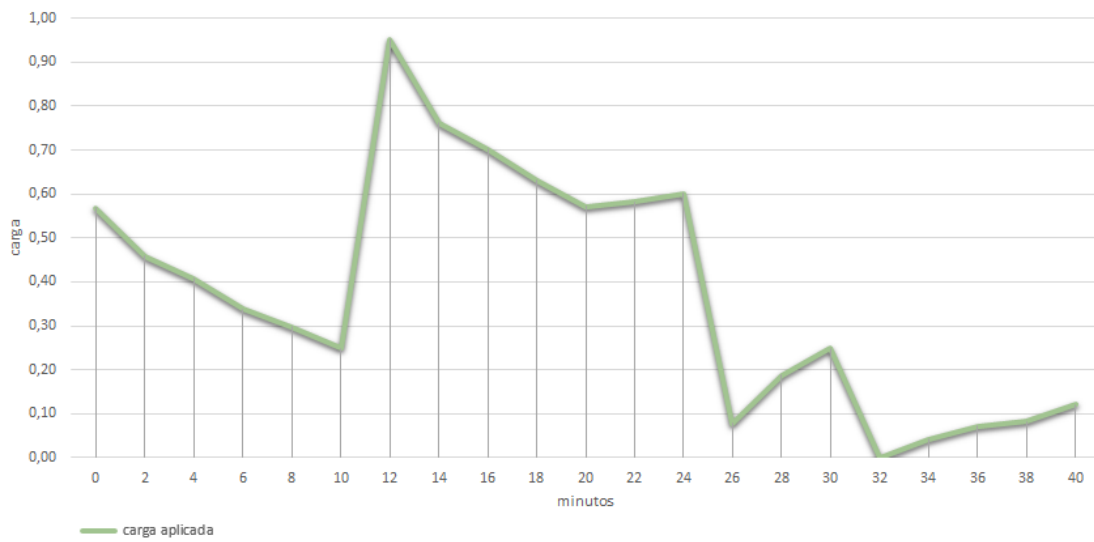


Figura 6.4: Evolução do fator carga aplicado em ordem ao tempo durante o exercício

Capítulo 7

Conclusões e trabalho futuro

7.1 Conclusões

Os sensores utilizados juntamente com a restante tecnologia envolvida neste projeto revelam-se compatíveis, tornando possível o desenvolvimento de um sistema completo.

O objetivo de se obter um sistema semi-autônomo está cumprido, pois esta solução apresentada apenas necessita que sejam parametrizados alguns valores referentes ao jogo e ao paciente pelo profissional de saúde. Toda essa parametrização poderá ser feita à distância tal como a análise dos resultados do paciente. O sistema desenvolvido permite ainda verificar a validade de alguns desses parâmetros essenciais ao treino do paciente como por exemplo as Zonas Alvo Treino.

É possível definir uma sessão de treino única para cada paciente, incluindo também avisos e comentários personalizados durante o decorrer do treino.

Infelizmente, com o VitalJacket 1L não foi possível desenvolver uma melhor monitorização do comportamento cardíaco do paciente enquanto decorre o jogo, pois este apenas disponibiliza a frequência cardíaca do paciente.

O algoritmo de balanceamento de cargas aplicado no jogo, mostra que é possível dirigir a frequência cardíaca do paciente à **ZAT** de referência. Em contrapartida, uma vez atingida a **ZAT** de referência, é difícil de manter a frequência cardíaca estável nesse valor, pois a forma como a carga aplicada, através de fatores no jogo como a velocidade dos objetos, não atua diretamente no jogador. Pacientes diferentes podem reagir de diferentes formas à mesma carga aplicada. O que faz com que a carga se altere constantemente.

A utilização do jogo embora só requeira a autenticação do paciente, poderá não ser fácil

para determinados utilizadores (nomeadamente os mais idosos) pois é ainda necessário um emparelhamento/ligação do VitalJacket ao computador e uma calibração do sensor Kinect.

O jogo não pôde ser testado em doentes cardíacos por razões de segurança e não-monotorização da informação disponível em ECG.

Relativamente ao teste de desempenho do jogo efetuado pelo FRAPS, conclui-se que o jogo é bastante fluido apresentando uma média de 62 FPS.

Combinando estas tecnologias é possível serem criadas novas aplicações de interesse na área da saúde.

7.2 Trabalho futuro

No futuro seria interessante que a aplicação fizesse uso do sensor de movimento de forma a interpretar e corrigir em tempo real a correção à postura do paciente, rentabilizando assim o esforço durante todo o exercício. Ao mesmo tempo preveniria o aparecimento de problemas físicos derivados a uma eventual postura errada durante a execução dos exercícios.

Para complementar o Painel de Controlo poder-se-ia implementar uma componente *web* que apresentasse o historial dos treinos dos pacientes entre outros.

O reconhecimento de alguns gestos por parte do Kinect, seria uma mais valia para esta aplicação para por exemplo possibilitar ao utilizador parar ou sair do jogo sem ter que se dirigir ao computador - o que se poderia melhorar com a utilização da mais recente versão do Kinect.

A autenticação poderia ser feita utilizando por exemplo o sensor RGB do Kinect para interpretar um *QR code*¹ que tivesse, de forma cifrada, a identificação do paciente.

Seria interessante utilizar e tratar a informação disponibilizada pelo VitalJacket versão 5L. Desta forma seria possível explorar as características do ECG dinamizando a utilidade deste SG de reabilitação.

¹"QR code (abbreviated from Quick Response Code) is the trademark for a type of matrix barcode (or two-dimensional barcode) first designed for the automotive industry in Japan" - Definição de QR Code em Wikipedia

Bibliografia

- [1] BioDevices. Documentation. www.biodevices.pt/. visitado 06-2012.
- [2] Greg Borenstein. *3D vision with Kinect, Processing, Arduino, and MakerBot*. O'Reilly, 2012.
- [3] Carla Marisa Maia Moreira (2007). Prescrição do exercício físico. <http://obesidade.info/prescricaoaf.htm>. visitado 08-2013.
- [4] Dr. Rui Cruz Ferreira. Reabilitação cardíaca: Realidade nacional e recomendações clínicas. <http://www.portaldasaude.pt/>. visitado 12-2012.
- [5] Freitas, Sara and Levene. Evaluating the development of wearable devices, personal data assistants and the use of other mobile devices in further and higher education institutions. 2005.
- [6] Instituto do Coração. Reabilitação cardíaca. <http://www.institutodocoracao.pt/pt/pagina/9/reabilitacao-cardiaca/>. visitado 12-2012.
- [7] S.A. Life Beat Centro de Diagnóstico Avançado. Registo ambulatório electrocardiograma 24 horas - vitaljacket. http://lifebeat.pt/exames/servicos_cardiovasculares/registo_ambulatorio_electrocardiograma_24_horas_vitaljacket/. visitado 01-2013.
- [8] Microsoft. SQL Server Documentation. [http://msdn.microsoft.com/en-us/library/ee229559\(v=sql.10\).aspx](http://msdn.microsoft.com/en-us/library/ee229559(v=sql.10).aspx). visitado 09-2012.
- [9] Ricardo Almendra. *Geografia da Doença Cardiovascular em Portugal Continental: enfarte agudo do miocárdio - padrões e sazonalidade*. 2012.

