



Simulação de Integrações de HL7 num Prestador de Saúde

RITA ISABEL SILVA COSTA

julho de 2024

Simulação de Integrações de HL7 num Prestador de Saúde

Rita Isabel Silva Costa

**Dissertação para obtenção do Grau de Mestre em
Engenharia Biomédica**

Supervisor: Engenheiro António Martins

Orientador: Professora Doutora Natércia Lima

Coorientador: Professor Doutor Joaquim Alves

Júri:

Presidente:

Joaquim Alves

Vogais:

António Constantino Martins, Arguente

Natércia Lima

Resumo

Health Level 7 (HL7) é um conjunto de *standards* amplamente utilizados em todo o mundo para a troca de informações na área da saúde, promovendo a interoperabilidade entre os diferentes sistemas. A simulação é uma técnica essencial em muitas áreas, incluindo a saúde, onde é permitido testar cenários clínicos, desde os mais simples aos mais complexos, em ambiente controlado, sem contribuir com nenhum risco para pacientes ou profissionais. A integração de sistemas de saúde utilizando *standards*, como os HL7, oferece benefícios significativos nas tecnologias de saúde, entre os quais a partilha segura e eficaz de informações, a redução da redundância de dados e a ajuda na tomada de decisão devido às informações em tempo-real partilhadas entre os mais diversos sistemas de um provedor de saúde.

Este projeto teve como objetivo a criação e desenvolvimento de um *software* capaz de simular, de forma realista, integrações de HL7 em prestadores de saúde. O simulador deve permitir a execução de testes de *workflow* específicos de um hospital, centro de saúde entre outros, utilizando informações realistas do ambiente e dos pacientes envolvidos, sem afetar o sistema de produção. Este foi concebido para gerar *outputs* das interações da simulação, tendo em conta a validação de integrações em sistemas de informação hospitalares.

A escolha da linguagem de programação recaiu sobre a linguagem em Java, devido à robustez e documentação da biblioteca HAPI, que suporta a criação das mensagens HL7 utilizadas na simulação. A interface gráfica foi desenvolvida utilizando JavaFX, sendo a estrutura gráfica definida por um ficheiro CSS, tornando o programa intuitivo e prático para o utilizador. Na estrutura do *software* criado inclui-se a definição de estados, transições e eventos, configurando igualmente a frequência com que as transições ocorrem e a probabilidade de as mesmas ocorrerem para criação de cenários realistas.

O simulador desenvolvido permite criar e modificar configurações da simulação, visualizar o diagrama da máquina de estados e executar simulações em tempo-real ou com o dobro da velocidade. Os resultados são guardados em ficheiros de texto que incluem as mensagens HL7 geradas e os movimentos dos pacientes no decorrer da simulação. São apresentados três casos de estudo que demonstram a flexibilidade e a eficácia do simulador em diferentes cenários, validando a sua capacidade de gerar resultados coerentes e realistas.

O desenvolvimento deste simulador de integrações HL7 mostrou-se uma ferramenta valiosa para a área da saúde, facilitando o teste e a validação de integrações sem comprometer os sistemas de produção. Futuras melhorias podem incluir a introdução dos mais diferentes tipos de mensagens HL7, a inclusão de diferentes tipos de profissionais na simulação, envolvidos num *workflow* hospital e, conseqüentemente, incluídos nas mensagens HL7. Posteriormente, a intenção é partilhar o *software* criado e melhorado, como ferramenta *open-source* para entajuda da comunidade científica.

Palavras-chave: HL7, Interoperabilidade, Saúde, Simulador, HAPI, máquina de estados

Abstract

Health Level 7 (HL7) is a set of standards widely used worldwide for exchanging information in the health sector, promoting interoperability between different systems. Simulation is an essential technique in many areas, including healthcare, where it is possible to test clinical scenarios, from the simplest to the most complex, in controlled environments that do not pose any risks to patients or professionals. Integrating healthcare systems using standards, such as HL7, offers significant benefits in healthcare technology, including the reliable and effective sharing of information, the elimination of data redundancy and the assistance in decision-making due to the real-time information shared between a healthcare provider's various systems.

This project aimed to create and develop software capable of realistically simulating HL7 integrations in healthcare providers. The simulator should allow specific workflow tests to be carried out in a hospital, health centre, etc., using realistic information about the environment and the patients involved, without affecting the production system. It is designed to generate outputs from the simulation interactions, considering the validation of integrations in hospital information systems.

The programming language chosen was Java, due to the solidity and documentation of the HAPI library, which supports the development of the HL7 messages used in the simulation. The GUI was developed using JavaFX, with the design defined by a CSS file, making the program intuitive and practical for the user. The structure of the software created includes the definition of states, transitions, and events, as well as configuring the frequency with which transitions occur and the probability of them occurring to create realistic scenarios.

The simulator allows the user to create and modify the simulation settings, view the state machine diagram, and run simulations in real-time or in duplicate. The results are kept in text files including the HL7 messages generated and the patient's movements during the simulation. Three case studies are presented which demonstrate the simulator's flexibility and effectiveness in different scenarios, validating its ability to generate consistent and realistic results.

The development of this HL7 integration simulator has proved to be a valuable tool for the healthcare sector, making it easier to test and validate integrations without affecting production systems. Future improvements could include the introduction of different types of HL7 messages, the integration of different professional roles in the simulation, involved in a hospital workflow and consequently included in the HL7 messages, and then the distribution of the software created and improved as an open-source tool to help the scientific community.

Keywords: HL7, Interoperability, Healthcare, Simulator, HAPI, State Machine

Agradecimentos

Com a minha jornada académica a terminar, é com enorme gratidão que agradeço a todos aqueles que, de alguma forma, contribuíram para a minha vida académica e que me ajudaram a terminar esta dissertação da melhor forma.

Agradeço aos meus orientadores, Natércia Lima e Joaquim Alves, pelo apoio e disponibilidade durante todo este processo, visto terem sido fundamentais para que este projeto fosse realizado da melhor forma.

Agradeço ao António Martins a sua incansável paciência, dedicação e suporte oferecido nestes últimos meses. A sua experiência e partilha de conhecimentos nesta área tão complexa foi crucial para dar o melhor no desenvolvimento deste projeto.

À minha família, especialmente, ao meu pai, mãe e irmã, agradeço todo o apoio e compreensão durante todos os desafios gerados neste processo.

Finalmente, agradeço ao meu namorado, amigos e colegas de trabalho por estarem sempre lá nos momentos altos e baixos, tentando me manter motivada e focada no verdadeiro objetivo, sem nunca duvidar do meu verdadeiro valor.

A todos, o meu sincero obrigada!

Índice

| | | |
|----------|-----------------------------------|-----------|
| 1 | Introdução | 1 |
| 1.1 | Enquadramento e pertinência | 1 |
| 1.2 | Objetivos | 2 |
| 1.3 | Apresentação da Empresa | 3 |
| 1.4 | Estrutura do Documento | 4 |
| 2 | Estado da Arte | 5 |
| 2.1 | Simuladores | 5 |
| 2.2 | Máquinas de estados | 10 |
| 2.3 | Interoperabilidade | 11 |
| 2.4 | IHE | 14 |
| 2.5 | HL7 | 15 |
| 2.5.1 | Tipos do Standard HL7 | 15 |
| 2.5.2 | Estrutura | 16 |
| 2.5.3 | Eventos HL7 v2.0 | 17 |
| 2.6 | Conclusão do Capítulo 2 | 17 |
| 3 | Métodos e Aplicação | 19 |
| 3.1 | Tecnologias utilizadas | 19 |
| 3.2 | Estrutura do Projeto Desenvolvido | 21 |
| 3.3 | Conclusão do Capítulo 3 | 28 |
| 4 | Resultados e Discussão | 29 |
| 4.1 | Apresentação de Resultados | 29 |
| 4.1.1 | Configurações da Simulação | 29 |
| 4.1.2 | Simulação | 40 |
| 4.2 | Discussão de Resultados | 44 |
| 4.2.1 | Caso 1 | 44 |
| 4.2.2 | Caso 2 | 49 |
| 4.2.3 | Caso 3 | 54 |
| 4.3 | Conclusão do Capítulo 4 | 60 |
| 5 | Conclusão | 61 |
| 5.1 | Conclusão Finais | 61 |
| 5.2 | Limitações e Investigação Futura | 62 |

Lista de Figuras

| | |
|---|----|
| Figura 1 - Sede da empresa SECTRA, Suécia [5]. | 3 |
| Figura 3 - Simulador <i>Link Trainer</i> [11]. | 6 |
| Figura 4 - <i>Kriegsspiel</i> , jogo de guerra desenvolvido pelo Exército Prussiano [12]. | 6 |
| Figura 5 - <i>Swift Playgrounds</i> , jogo sério desenvolvido por Cheng, G. e Chen, C. [15]. | 8 |
| Figura 6 - Sistema <i>open-source Patient Manager</i> [19]. | 9 |
| Figura 7 - Order Manager [21]. | 10 |
| Figura 8 - Máquina de estados de um torniquete com recolha de moedas (Adaptado de [24]). | 11 |
| Figura 9 - Níveis de Interoperabilidade (Adaptado de [27]). | 12 |
| Figura 10 - Filme de Raio-X [32]. | 13 |
| Figura 11 - <i>Framework</i> desenvolvida pelo IHE (Adaptado de [33]). | 14 |
| Figura 12 - Desenvolvimento dos <i>standards</i> HL7 ao longo do tempo [37]. | 15 |
| Figura 17 - Estrutura do Projeto Desenvolvido. | 21 |
| Figura 18 - Diagrama de Classes do Projeto Desenvolvido. | 22 |
| Figura 19 - Classe "Main.java" | 23 |
| Figura 20 - Classe "Configuration.java" | 23 |
| Figura 21 - Classe "Simulation.java" | 24 |
| Figura 22 - Classe "State.java". | 24 |
| Figura 23 - Classe "Transition.java". | 25 |
| Figura 24 - Classe "Event.java". | 25 |
| Figura 25 - Classes "Person.java" e "PatientInfo.java" | 26 |
| Figura 26 - Classe "StateMachine.java". | 27 |
| Figura 27 - Classes (a) "TimeSimulator.java" e (b) "StateMachineVisualizer.java". | 27 |
| Figura 28 - Listagem de configurações já existentes. | 30 |
| Figura 29 - Aviso caso não existam ficheiros de configuração. | 30 |
| Figura 30 - Menu principal da aplicação. | 31 |
| Figura 31 - Escolha do nome da Simulação. | 31 |
| Figura 32 - Criação dos estados da Simulação. | 32 |
| Figura 33 - Definição dos estados de início e de fim da Simulação (a), sendo apresentadas as opções possíveis de estados (b). | 32 |
| Figura 34 - Definição das condições da Transição de início e de fim da simulação. | 33 |
| Figura 35 - Opções disponíveis para os Eventos. | 34 |
| Figura 36 - Definição das transições entre estados, assim como os eventos. | 34 |
| Figura 37 - Listagem das transições criadas. | 35 |
| Figura 38 - Aviso apresentado caso as configurações das transições não sejam coerentes. | 35 |
| Figura 39 - Detalhes da Configuração da Simulação. | 36 |
| Figura 40 - Aviso apresentado aquando da criação da configuração. | 36 |
| Figura 41 - Seleção da configuração a modificar. | 38 |
| Figura 42 - Detalhes da configuração após modificação. | 39 |
| Figura 43 - Aviso apresentado aquando da modificação da configuração. | 39 |

| | |
|---|----|
| Figura 44 - Listagem das configurações existentes (a) e detalhes da configuração escolhida pelo utilizador (b). | 40 |
| Figura 45 - Início da Simulação escolhida..... | 41 |
| Figura 46 - Simulação a decorrer..... | 42 |
| Figura 47 - Aviso aquando do término da simulação..... | 42 |
| Figura 48 - Exemplo das informações apresentadas no ficheiro de texto após a simulação. | 43 |
| Figura 49 - Término do programa. | 43 |
| Figura 50 - Simulação exemplo com 2 estados. | 44 |
| Figura 51 - Estado da simulação de 2 estados. | 45 |
| Figura 52 - Simulação exemplo com 3 estados. | 49 |
| Figura 53 - Estado da Simulação de 3 estados. | 50 |
| Figura 54 - Simulação exemplo com 5 estados. | 54 |
| Figura 55 - Estado da simulação de 5 estados. | 56 |

Acrónimos e Símbolos

Lista de Acrónimos

| | |
|--------------|--|
| ACR | <i>American College of Radiology</i> |
| CDA | <i>Clinical Document Architecture</i> |
| CSS | <i>Cascading Style Sheets</i> |
| DICOM | <i>Digital Imaging and Communication in Medicine</i> |
| EHR | <i>Electronic Health Record</i> |
| FHIR | <i>Fast Healthcare Interoperability Resources</i> |
| GUI | <i>Graphical User Interface</i> |
| HAPI | <i>HL7 Application Programming Interface</i> |
| HIMSS | <i>Healthcare Information and Management Systems Society</i> |
| HL7 | <i>Health Level 7</i> |
| IDE | <i>Integrated Development Environment</i> |
| IHE | <i>Integrating the Healthcare Enterprise</i> |
| NEMA | <i>National Electrical Manufacturers Association</i> |
| RIM | <i>Reference Information Model</i> |
| REST | <i>Representational State Transfer</i> |
| UE | <i>União Europeia</i> |

1 Introdução

Neste capítulo é apresentado o projeto desenvolvido no âmbito da unidade curricular Tese/Dissertação, explicando o contexto e enquadramento assim como a pertinência do mesmo. São igualmente apresentados os objetivos definidos para este trabalho e feita uma apresentação sumária da empresa na qual foi desenvolvido. A última secção deste capítulo explana a estrutura deste documento.

1.1 Enquadramento e pertinência

A área da saúde tem sido uma das áreas com a maior e mais constante evolução da história moderna. Acompanhando estes desenvolvimentos, junta-se a complexidade dos sistemas criados pelo ser humano como forma de prestar o melhor cuidado de saúde possível. Para que esse objetivo fosse alcançado, o desenvolvimento da tecnologia foi, e ainda o é nos dias de hoje, o principal pilar de desenvolvimento de novas e melhores ferramentas com vista a aumentar a qualidade de vida e longevidade dos pacientes [1].

A simulação é, desde os tempos mais remotos, uma das mais seguras e versáteis formas de teste de situações reais, podendo ser utilizada também de forma lúdica ou educativa. Assim, a simulação em áreas complexas e críticas como a Saúde é uma técnica cada vez mais utilizada. São exigidos certos critérios de eficiência dos processos e tecnologias utilizados. A grande utilização de dados, a necessidade de integração e interoperabilidade entre sistemas torna-se imprescindível para melhorar a qualidade do atendimento, prestação dos cuidados e segurança do processamento dos dados. Para que estes objetivos sejam concretizados são, por vezes, utilizadas simulações de cenários clínicos em ambientes controlados que permitem a análise e teste de várias variáveis, sem que existam riscos impactantes para os pacientes e profissionais de saúde [2], [3].

Health Level 7, ou HL7, é uma organização, fundada em 1987, que desenvolve *standards* adotados internacionalmente com o objetivo de troca de informações na área da saúde. Estes *standards* são amplamente utilizados por organizações de saúde de forma a facilitar a

comunicação entre os mais diversos sistemas de informação, garantindo que existe a troca de dados críticos de forma segura e clara. A integração de sistemas de saúde utilizando *standards* HL7 oferece vários benefícios entre os quais se destacam: o compartilhamento das informações dos pacientes entre os vários sistemas de forma precisa e rápida; a diminuição da redundância dos dados, devido à reduzida entrada manual dos mesmos; e a tomada de decisão de forma segura por parte dos prestadores de cuidados de saúde, pois proporciona uma base sólida para a análise e respetiva tomada de decisão [2], [3].

1.2 Objetivos

Considerando que os sistemas de informação na saúde são considerados sistemas críticos, o principal objetivo deste trabalho foca-se na implementação de testes do *workflow* específicos e realistas de um Hospital, centro de saúde ou outro provedor de saúde, com o objetivo de simular a integração de HL7 sem afetar o sistema de produção, visto que os testes são feitos no *software* desenvolvido, utilizando informações reais do ambiente e pessoas envolvidas.

Assim, foi pensada uma metodologia para a criação de um *software*, desenvolvido com a base lógica de numa máquina de estados capaz de, através do *input* do utilizador, gerar o *output* realístico e esperado de uma interação de HL7 num prestador de cuidados de saúde.

Com o desenvolvimento deste projeto é esperado um impacto positivo no desenvolvimento das integrações de HL7 em prestadores de cuidados de saúde. Pretende-se um *software* fácil e prático para testar este tipo de integrações nos mais diversos ambientes hospitalares, desde os mais complexos aos mais simples. De igual forma, reduz-se o impacto negativo da utilização dos sistemas utilizados para produção, para o teste destas integrações.

Assim, o desenvolvimento desta dissertação tem como principais objetivos a criação de um simulador com a capacidade de simular realisticamente diversas integrações de HL7 num prestador de saúde sem afetar os sistemas de informação.

Este simulador deve ser capaz de simular eventos básicos utilizando informações básicas por exemplo, dados de pacientes e instalações, sendo que cada um destes eventos pode ter uma mensagem HL7 v2.x associada.

De forma a obtermos uma simulação realista, foram tidos em consideração alguns fatores como paralelismo de eventos, a escalabilidade dos eventos consoante a dimensão do prestador de saúde, eventos esses desencadeados numa sequência de máquina de estados. De igual forma, o simulador deve ter em consideração as circunstâncias dos eventos no mundo real, como adição de eventos básicos, como admissão e alta, assim como alguns mais complexos como por exemplo reagendamento, atualização de dados demográficos, e cancelamento, sendo que nestes eventos deve sempre utilizar informações pessoais e espaços realísticos.

O futuro do projeto desenvolvido conta com a disponibilização do *software* para a comunidade das tecnologias de saúde, apoiado por diversas bibliotecas já existentes, e criado com o intuito de se tornar *open-source*.

1.3 Apresentação da Empresa

A SECTRA é uma empresa dedicada à tecnologia médica e sistemas de comunicação seguros, sendo esta última a sua primeira área de negócio (*SECure TRANsmissions*). Fundada em 1978, na cidade de Linköping na Suécia, por uma equipa de investigadores da Universidade de Linköping, dá a sua entrada no mercado com a implementação de soluções seguras destinadas a bancos. Mais tarde, o seu negócio é redirecionado para a criação de comunicações seguras, sendo ainda hoje utilizada por departamentos governamentais em todo o mundo, incluindo a maioria dos chefes de estado presentes na União Europeia (UE). Em 1989, o seu negócio foi expandido para a área de tecnologia médica pelo atual diretor executivo (CEO), Torbjörn Kronander, devido ao seu interesse pela área. A partir desse momento iniciou-se a transação para sistemas de gestão de imagens digitais para os departamentos de radiologia sendo que, em 1993, foi conseguido o primeiro departamento de radiologia *film-free* em Mjölby [4].

Atualmente, a SECTRA conta com vários modelos de negócio adicionais aos mencionados anteriormente, estando presente em mais de 60 países e contabilizando mais de 1100 colaboradores mundialmente. A sede da empresa mantém-se na cidade de Linköping na Suécia (Figura 1).



Figura 1 - Sede da empresa SECTRA, Suécia [5].

A Sectra Iberia foi o local de desenvolvimento do projeto referido neste documento. Em Portugal, mais especificamente Matosinhos a única área de atuação é a área médica sendo que, até ao momento, estão em atividade cerca de 50 trabalhadores [6].

1.4 Estrutura do Documento

A presente dissertação está organizada e dividida em 5 capítulos.

Além do presente capítulo, Introdução, onde se apresenta um breve contexto do desenvolvimento desta dissertação, bem como os objetivos e a apresentação da empresa onde foi desenvolvida, é também referida a estrutura deste documento.

No capítulo do Estado da Arte são apresentados os conceitos base para a compreensão do desenvolvimento do *software*, assim como uma listagem dos projetos existentes e relacionados com o tema.

De seguida, encontra-se o capítulo Métodos e Aplicação que descreve as tecnologias utilizadas, a organização do *software* e qual a relação entre todos os conceitos criados.

Em continuidade, o capítulo de Resultados e Discussão apresenta as funcionalidades do programa numa forma generalizada, sendo que posteriormente são discutidos casos práticos de simulações feitas com o uso do simulador.

Por fim, é apresentada a Conclusão final mencionando os aspetos a melhorar no futuro desenvolvimento contínuo do projeto.

2 Estado da Arte

Neste capítulo é feita uma breve revisão bibliográfica dos tópicos necessários à compreensão do tema desenvolvido nesta dissertação. Assim, este capítulo foca-se na explanação dos conceitos de simulação, em geral e aplicada à área da Saúde, e também a simulação realística e a definição e funcionamento de uma máquina de estados. De igual forma é feita a menção a tópicos cruciais para o desenvolvimento deste projeto como Interoperabilidade, *Integrating the Healthcare Enterprise* (IHE) e *Health Level 7* (HL7).

2.1 Simuladores

A palavra simulador é definida como sendo um “aparelho de instrução e treino que simula o funcionamento e ambiente de outro aparelho (este para ação prática) como, por exemplo, o simulador de voo, o simulador de radar, etc.”, desta forma, pode ser considerado como uma técnica de praticar e aprender, transversal às mais diversas áreas [7], [8].

Historicamente, a simulação encontra-se presente em diversas áreas, tendo sido aperfeiçoada ao longo dos anos. Algumas das principais áreas onde a simulação tem sido fulcral no desenvolvimento de novas e melhoradas tecnologias são as áreas militares, aviação, educação e saúde [9].

O primeiro caso de sucesso no uso de um simulador foi concretizado na aviação no final de 1920 com o desenvolvimento do *Link Trainer* por Edwin Albert Link (Figura 2). Esta tecnologia tinha como finalidade o treino dos pilotos durante a Segunda Guerra Mundial, tendo sido treinados mais de 500 000 pilotos com este tipo de equipamentos. Este aparelho consistia num avião de pequenas dimensões, de madeira e com asas articuladas, onde se integram pequenas partes feitas de materiais utilizados nos aviões de grandes dimensões utilizados na altura. No interior, os pilotos teriam acesso aos controlos que, devido a uma bomba elétrica presente na estrutura,

dava a sensação realística do movimento de voo quando estes eram acionados. Anteriormente à invenção do *Link Trainer*, os pilotos enfrentavam dificuldades a pilotar em condições meteorológicas adversas, pois até então era feito com o uso da vista da superfície da terra. Desta forma, esta invenção foi uma das mais importantes na história do avião [9], [10].

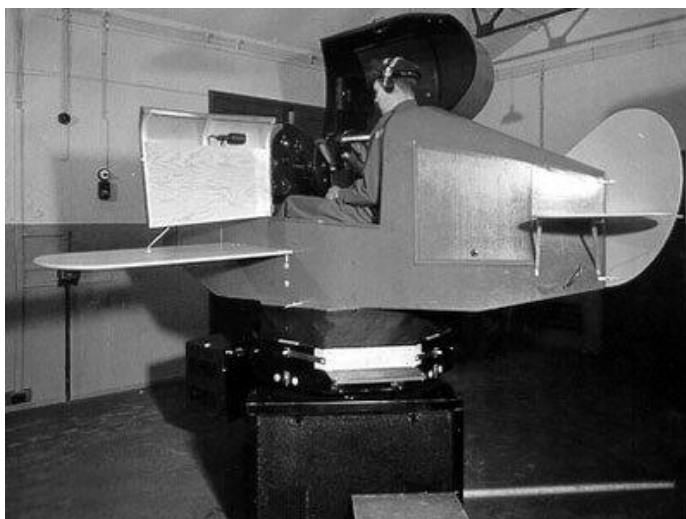


Figura 2 - Simulador *Link Trainer* [11].

A simulação de uma batalha utilizando peões e tabuleiros, amplamente utilizada na época medieval, foi a origem dos jogos de guerra como os conhecemos hoje. Um dos exemplos das primeiras simulações para fins militares é mencionado na história romana, em que os comandantes utilizavam mesas de madeira e peões, de forma a planejar as estratégias da batalha. Mais tarde, o desenvolvimento de novas simulações para treino militar deu origem a jogos como o *Kriegsspiel*, Figura 3, um jogo desenvolvido pelo exército prussiano no século XIX como forma de treinar o pensamento estratégico [9], [12].



Figura 3 - *Kriegsspiel*, jogo de guerra desenvolvido pelo Exército Prussiano [12].

Atualmente, com o desenvolvimento da tecnologia, os jogos passaram dos tabuleiros para os computadores. No início da era digital, os jogos eram criados para uso exclusivo do treino militar. No entanto, esta área tendo vindo a expandir para o uso de entretenimento [9].

Todos estes simuladores militares têm por base os mesmos elementos: objetivos que desejam alcançar, o cenário em que se encontram, as regras que devem seguir, os jogadores e uma análise ponderada [9].

A educação é uma área que necessita de constante desenvolvimento nas mais diversas vertentes de forma a melhorar a aprendizagem e teste das temáticas envolvidas. O objetivo que se pretende concretizar com o uso de simulações nesta área é que seja desenvolvido o treino técnico e funcional do caso de estudo. Além disso, as aptidões de pensamento crítico e toma de decisão podem ser aperfeiçoadas com o uso de simuladores, bem como a colaboração e comunicação em equipa, contando que cada individuo trabalhe em equipa de forma a alcançar o objetivo de aprendizagem pretendido [7].

A utilização de outras técnicas de simulação, como jogos sérios e as mais variadas tecnologias que utilizam realidade virtual, têm tido um crescimento exponencial devido à evolução tecnológica que enfrentamos [9], [13].

A definição de jogos sérios tem várias perspetivas no sentido em que depende da indústria, no entanto, é comum defini-los como sendo “jogos que não têm entretenimento, prazer ou diversão como o seu objetivo principal”. Assim, estes podem ser diferenciados dos tradicionais jogos como sendo desenvolvidos com propósitos lúdicos e não de puro entretenimento [13], [14].

Um exemplo de jogos sérios é o desenvolvimento do *Swift Playgrounds* (Figura 4) por Guo-Ming Cheng e Chia-Pin Chen. Tendo como base a importância do pensamento computacional, programação e resolução de problemas principalmente nas classes mais jovens, desenvolveram uma aplicação lúdica que, sem qualquer conhecimento prévio de programação, ensina aos seus utilizadores os tópicos básicos de programação. Desta forma, com a ajuda do “Byte”, personagem do jogo, os seus utilizadores conseguem arrecadar conhecimentos de funções e ciclos frequentemente utilizados na programação [13], [14].

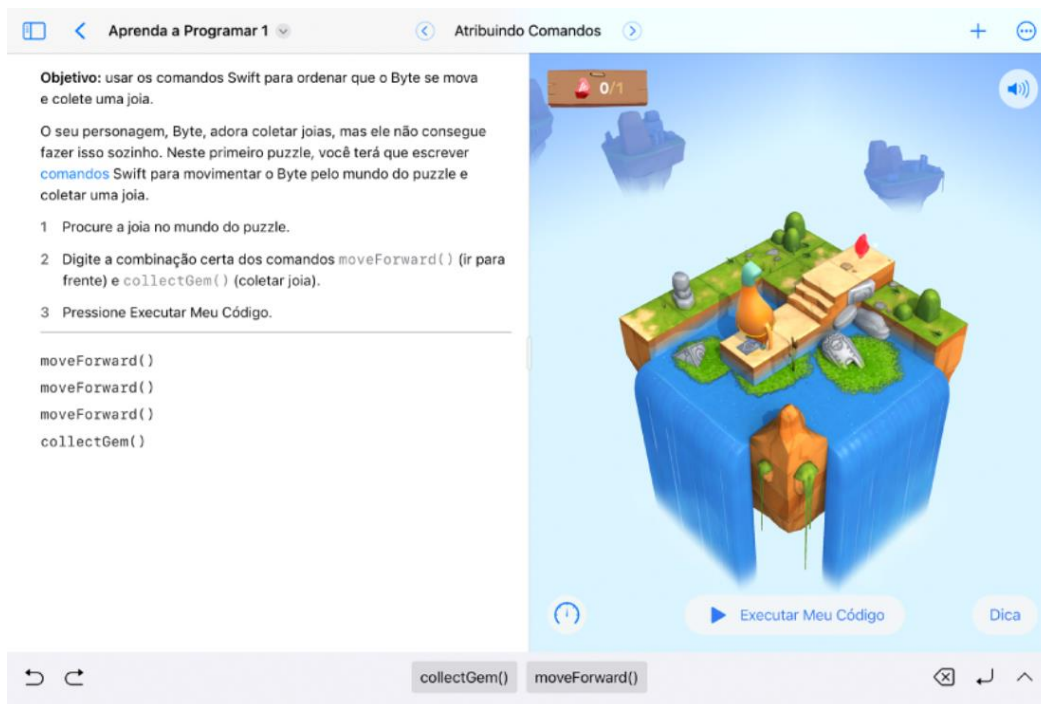


Figura 4 - *Swift Playgrounds*, jogo sério desenvolvido por Cheng, G. e Chen, C. [15].

A simulação realista é utilizada como forma de preparação e teste dos mais diversos cenários aquando de uma implementação no contexto real. Este tipo de metodologia é amplamente utilizado em sistemas críticos, como a saúde, pois o seu maior benefício é a avaliação das possíveis reações adversas sem impactar o sistema e os envolvidos no mesmo.

Assim, na área da saúde é evidente o uso de simuladores, como por exemplo os jogos sérios, de forma a educar nas mais diversas vertentes, seja na melhoria do estilo de vida ou no estudo de medicina, sendo esta última considerada uma das mais importantes áreas de aplicação de simuladores [9].

A educação em medicina sempre utilizou a técnica de “ver, fazer, ensinar”, de acordo com William Hasted, cirurgião do século XX e pioneiro na criação de um modelo de aprendizagem nesta área [16]. Assim, a aprendizagem na saúde com o uso de simuladores é uma prática antiga e até hoje utilizada, pois permite que o aprendiz cometa erros e aprenda com os mesmos, sem comprometer a vida humana. São muitos os benefícios do uso de simulações entre os quais a melhoria da técnica, o pensamento crítico e de tomada de decisão e a comunicação intrapessoal [16], [17].

De igual forma, na área da saúde são cada vez mais utilizados simuladores, não com propósito lúdico, mas com o objetivo de testar situações reais sem correr os riscos inerentes aos mesmos e, de igual forma, reduzir custos e o stress já existente neste tipo de instituições. Um dos muitos exemplos é o uso de simulação discreta de eventos em que o comportamento de eventos padrão reais é simulado. Esta abordagem é particularmente interessante pois utiliza a informação recolhida como forma de tomar conhecimento de cenários nunca pensados.

Permite, por exemplo, prever e simular onde é que um paciente que entrou no hospital deve ir e quanto tempo estará neste estabelecimento [18].

Numa outra área, o *Patient Manager* é um sistema *open-source* desenvolvido em conformidade com a *framework* do IHE, com o principal objetivo de ajudar os programadores de sistemas informáticos a testar os seus sistemas, visto que o ideal na interoperabilidade é que estes sejam desenvolvidos em conformidade com o IHE de forma a serem facilmente integrados com outros sistemas de informação existentes no provedor de saúde.

O *Patient Manager* é capaz de testar o envio de mensagens HL7 dos mais variados tipos e receber avisos dos mesmos, de forma a testar integrações de HL7 (Figura 5). O utilizador tem o poder de decidir as informações que deseja enviar para o *Patient Manager* a partir do seu sistema, como por exemplo, informações para criar um paciente. Assim, também pode enviar mensagens HL7 para alterar estas informações ou até mesmo o cancelamento de um evento [19].

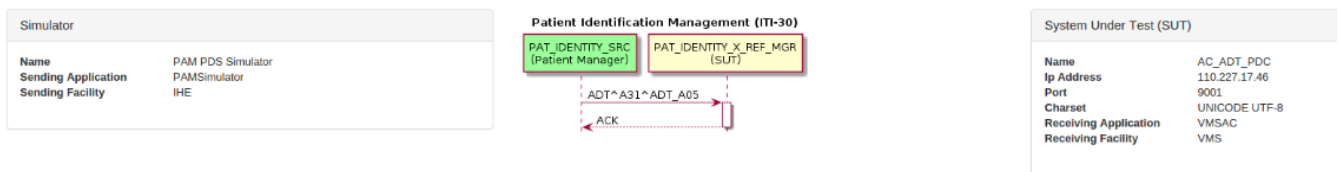


Figura 5 - Sistema *open-source Patient Manager* [19].

Da mesma forma, o *Order Manager* é utilizado como forma de simular os vários perfis de integração relacionados com a imagiologia e requisição laboratorial. O principal objetivo, em primeira instância, é ajudar os técnicos de equipamentos médicos a consultar as listas de trabalho sem a necessidade do apoio de qualquer outro sistema usado *Order Placer* e/ou *Order Filler*, utilizados para produzir as listas de *Order*, e, simultaneamente, ajudar no teste e validação do envio e receção de mensagens de HL7 exigidas para estas transações (Figura 6). Este simulador permite ainda a reutilização de informações utilizadas em integrações no *Patient Manager*, anteriormente mencionado [21].

| Order Selection | | | | | | | |
|---|--|--------------|---------------------|---------------|----------------------|-------------------------------------|-----------------------|
| Search criteria | | | | | | | |
| Creator | Show All | x | Last modified | Any | x | | |
| Fist name | Show All | x | Last name | Show All | x | | |
| Patient Identifier | | | Visit number | | | | |
| Placer order number | | | Filler order number | | | | |
| Order numbers | Encounter information | Control code | Status | Result status | Universal Service Id | Last changed | Creator |
| 1*IHE_OM_OP*1.3.6.1.4.1.12559.11.1.2.2.4.2*ISO | Rosine Bartoli | DC | IP | | 10750 | 9/2/16 11:18:56 AM (CEST GMT+0200) | OM_RAD_OP_IHE_Gazelle |
| 1*IHE_OM_OF*1.3.6.1.4.1.12559.11.1.2.2.4.3*ISO | 3***IHEPAM&1.3.6.1.4.1.12559.11.1.2.2.5&ISO*VN | | | | | | |
| 9257*IHE_OM_OP*1.3.6.1.4.1.12559.11.1.2.2.4.2*ISO | Yvo Schmidt | NW | | | 10410 | 8/17/16 11:30:40 AM (CEST GMT+0200) | OM_RAD_OP_IHE |
| 8826*IHE_OM_OF*1.3.6.1.4.1.12559.11.1.2.2.4.3*ISO | 11340***IHEPAM&1.3.6.1.4.1.12559.11.1.2.2.5&ISO*VN | | | | | | |
| 8694*IHE_OM_OF*1.3.6.1.4.1.12559.11.1.2.2.4.3*ISO | Kondragunta Shyamsundar | SN | | | 10250 | 7/13/16 1:54:10 PM (CEST GMT+0200) | |
| | 11487***IHEPAM&1.3.6.1.4.1.12559.11.1.2.2.5&ISO*VN | | | | | | |
| 8712*IHE_OM_OF*1.3.6.1.4.1.12559.11.1.2.2.4.3*ISO | Kasandra Leannon | SN | | C | OC-SL-AS-OD | 7/13/16 11:45:23 AM (CEST GMT+0200) | |
| | 141164410***MIPSN | | | | | | |
| 9313*IHE_OM_OP*1.3.6.1.4.1.12559.11.1.2.2.4.2*ISO | Ellen Adam | NA | | | OC-IOL-ASCAN-OD | 7/13/16 11:45:14 AM (CEST GMT+0200) | |
| 8723*IHE_OM_OF*1.3.6.1.4.1.12559.11.1.2.2.4.3*ISO | 137***TEST_ABE&1.2.3.4.3.6666&ISO*VN | | | | | | |

Figura 6 - Order Manager [21].

2.2 Máquinas de estados

Uma máquina de estados é um modelo matemático abstrato utilizado para desenvolver algoritmos, amplamente utilizado em diversas áreas como engenharia, ciências e informática. Descreve o comportamento do sistema, constituído por um conjunto de estados num determinado espaço de tempo, variando consoante os eventos de *input* e *output* entre estados, designados de transições [22], [23].

Matematicamente, $M = [S, I, O, f_s, f_o]$ define-se como Máquina de Estados se S for um conjunto de estados, I um conjunto de *Inputs*, O um conjunto de *Outputs*, considerando que a máquina de estados se inicia sempre num estado fixo S_0 . A função f_s , representa o próximo estado, isto é, atribui a um estado o seu *input* sendo que para qualquer input/estado corresponde um estado seguinte, $f_s: S \times I \rightarrow S$. Já a função f_o representa a função do output, sendo representada por $f_o: S \rightarrow O$ [24].

Geralmente, estas incluem vários elementos: estado atual, condição, ação e estado secundário. Todos estes termos estão interligados, ou seja, quando uma condição é atingida, é acionada uma ação que pode fazer a transição de um estado para o seguinte. Neste cenário, estado atual e condição são a causa, e ação e estado secundário o resultado. Desta forma, as máquinas de estados ditam uma forma sistemática de controlar e gerir um sistema com comportamento sequencial [22], [25].

Um exemplo de uma máquina de estado simples é o torniquete com um sistema de recolha de moedas. A Figura 7 descreve a máquina de estados necessária para que este sistema seja funcional [24].

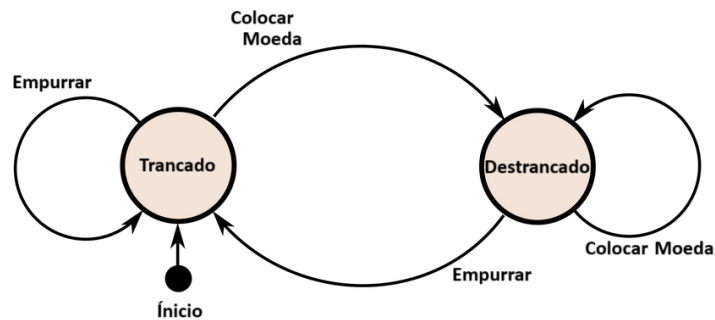


Figura 7 - Máquina de estados de um torniquete com recolha de moedas (Adaptado de [24]).

Analisando a máquina de estados, representada na Figura 7, podemos concluir que a máquina possui dois estados *Trancado* e *Destrancado*. Cada um destes estados possui várias transições, neste caso Empurrar ou Colocar moeda. O estado de início é o estado *Trancado*, sendo que colocar moeda, faz a máquina transitar para o estado seguinte, *Destrancado*, e Empurrar mantém a máquina no mesmo estado. Na mesma ótica, no estado *Destrancado*, colocar moeda não faz a máquina transitar para o estado seguinte, no entanto a ação Empurrar faz.

2.3 Interoperabilidade

Interoperabilidade é considerada pela *Healthcare Information and Management Systems Society* (HIMSS) como a “capacidade de diferentes sistemas de informação, equipamentos e aplicações terem acesso, troca, integração e cooperação do uso dos dados de uma maneira coordenada, dentro e fora de uma organização, região e fronteira, para proporcionar uma transmissão rápida e sem discontinuidades da informação, otimizando os serviços de saúde dos indivíduos e das populações a nível mundial” [26].

A interoperabilidade pode ser classificada em quatro níveis, como apresentado na Figura 8:



Figura 8 - Níveis de Interoperabilidade (Adaptado de [27]).

- O nível mais baixo, denominado de **Fundamental**, estabelece os requisitos necessários para a conexão de um sistema ou aplicação, de forma a que ocorra uma transferência de dados segura;
- O nível 2, ou **Estrutural**, define o formato, a sintaxe e a forma como a informação é organizada na sua transferência entre sistemas;
- O nível **Semântico**, nível 3, fornece modelos e codificação dos dados, incluindo a utilização de elementos dos dados seguindo normas publicadas, proporcionando uma melhor compreensão para o utilizador;
- No último nível, o **Organizacional**, são incluídas todas as organizações que possam facilitar da comunicação e utilização segura dos dados, tanto no interior das organizações como entre as mesmas e outras entidades, ou até mesmo indivíduos. Nesta categoria são tidas em consideração as opiniões de governos, a jurisdição dos países envolvidos, entre outras entidades sociais e políticas [26].

Independentemente de permitir transferência de uma grande quantidade de dados e de ser utilizada por diversas plataformas, nem todos os casos de interoperabilidade são bem-sucedidos. Desta forma, estes desafios foram estudados por investigadores e comprovados, tanto teoricamente como empiricamente [28], [29].

O futuro da Interoperabilidade na saúde prende-se com o desenvolvimento de formas que permitam que a troca de informações médicas entre sistemas proporcione um melhor cuidado dos pacientes, o desenvolvimento de políticas para melhorar a saúde de uma população, o desenvolvimento de ferramentas para melhorar o acesso das suas próprias informações de saúde e o estudo do impacto dos problemas de saúde a grande escala, podendo ser estas instituições estabelecer parcerias com instituições académicas [30].

De forma a atingir a interoperabilidade “perfeita”, todos os sistemas de um determinado provedor de saúde têm de estar completamente conectados, seguindo as normas existentes e utilizando ferramentas que permitam atingir esse objetivo. Entre esses sistemas inclui-se o

Electronic Health Record (EHR), assim como programas para a prescrição de medicamentos ou registo de dados clínicos [30].

O EHR foi criado como forma de integrar com outros sistemas de informação os registos médicos eletrónicos de um determinado indivíduo, gerados pelo médico, assim como o registo de saúde pessoal, gerido pelo paciente. A premissa do EHR é de facilitar a disponibilidade das informações acerca da saúde dos pacientes, independentemente do local de prestação de cuidados de saúde. Desta forma, é garantida a prestação de melhores cuidados de saúde. Os sistemas de registo de dados devem estar interligados de forma segura e sem falhas para a utilização por parte dos médicos e prestadores de serviços de saúde. Desta forma, o corpo clínico e os fornecedores necessitam de formação para entender o *workflow* e os elementos dos dados, visto que a qualidade da informação guardada nestes registos depende diretamente da qualidade da informação introduzida nestes sistemas [28], [29].

O DICOM (*Digital Imaging and Communication in Medicine*) é outro exemplo de *standard* com o objetivo de atingir a máxima interoperabilidade entre sistemas de saúde. Este foi desenvolvido no início de 1980 pela *National Electrical Manufacturers Association (NEMA)* em colaboração com *American College of Radiology (ACR)* para que seja um alicerce na interoperabilidade entre os diferentes equipamentos médicos ou sistemas. Este define a forma como é disponibilizada a informação, o protocolo de transferência de dados, o formato dos ficheiros, a informação clínica e o encapsulamento de metadados, isto é, dados que descrevem outros dados. Este é utilizado por muitos países e organizações como o principal *standard* para guardar e transferir imagens médicas e assim construir o EHR [30], [31].

Até à sua criação, as imagens resultantes dos diferentes equipamentos médicos, como Raio-X, mesmo inseridas dentro do mesmo departamento, viam-se confinadas às suas salas. Estas eram então impressas em filmes para serem analisados pelos radiologistas, como ilustrado na Figura 9.



Figura 9 - Filme de Raio-X [32].

Quando processadas digitalmente, era possível visualizá-las e processá-las na consola do equipamento médico e muito raramente eram exportadas para diferentes locais de trabalho e, mesmo os sistemas de imagem médica, não estavam conectados entre si. Quando era necessário a transferência de exames entre equipamentos médicos e locais de trabalho, apenas seria possível com dispositivos amovíveis como disquetes e cassetes. No entanto, este método enfrentava um desafio relacionado com a leitura dos ficheiros, visto que não existia um padrão para o tipo de ficheiro utilizado, o que causava por vezes a inutilização das imagens, ao estarem dependentes do correto *software* para a sua leitura [30].

2.4 IHE

Integrating the Healthcare Enterprise (IHE) é considerada uma “iniciativa de profissionais de saúde e da indústria para melhorar a forma como os sistemas informáticos nos cuidados de saúde partilham informações” [33].

Fundada pela ACR e a HIMSS em 1998, o IHE teve a colaboração de ambas as instituições como forma de promover a interoperabilidade dos sistemas médicos. Assim, ambas as instituições realizaram eventos com os principais fornecedores da área de modo a informar e aconselhar os seus clientes na integração dos vários sistemas. Inicialmente, o principal foco desta organização era apenas a radiologia, no entanto, neste momento, trabalha nas mais diversas áreas da saúde incluindo, por exemplo, patologia, cardiologia ou farmácia [33], [34].

A *framework* desenvolvida pelo IHE é uma simples forma de normalizar a forma como as informações são partilhadas nos cuidados de saúde e fora deles. São pensados nos principais problemas da interoperabilidade, como o acesso e partilha das informações dos pacientes, o *workflow* clínico, a segurança informática, a administração e infraestrutura [33].

O processo descrito na Figura 10 faz com que os utilizadores e programadores de sistemas informáticos estejam mais conscientes e alinhados com o futuro da saúde.



Figura 10 - Framework desenvolvida pelo IHE (Adaptado de [33]).

O primeiro passo da *framework* criada pelo IHE é descrito como sendo a definição por especialistas dos casos de uso na partilha de informação crítica. Seguidamente, técnicos experientes desenvolvem especificações pormenorizadas de como será feita a comunicação entre sistemas correspondendo aos casos suprarreferidos. Nesta etapa são selecionados *standards*, como HL7 e DICOM, para otimização do processo. Estas especificações são implementadas na indústria, sendo intituladas de perfis IHE. Por fim, o IHE testa e avalia os sistemas dos fornecedores em eventos planeados previamente conhecidos como os *Connectathons*. IHE é também responsável pela organização de demonstrações com cenários clínicos reais de sistemas compatíveis com esta *framework* [33].

2.5 HL7

Todos os sistemas que envolvam cuidado médico têm como principal objetivo melhorar o atendimento e tratamento do paciente, simultaneamente reduzindo custos operacionais, erros humanos e redundância das tarefas. Desta forma, a integração dos sistemas é fundamental para que este objetivo seja concretizado. Assim, o HL7 é introduzido com esse objetivo bem presente, sendo que tenciona resolver os maiores problemas de integração na indústria da saúde [35].

Health Level 7, comumente reconhecido como HL7, é uma organização, fundada em 1987, que desenvolve *standards* com a principal função de realizar troca de informação entre diferentes dispositivos médicos [35].

2.5.1 Tipos do Standard HL7

A organização HL7 desenvolveu vários *standards* ao longo dos anos de investigação e desenvolvimento, mantendo-se dinâmico até aos dias de hoje (Figura 11). O seu desenvolvimento culmina em vários projetos de investigadores, hospitais e outros provedores de saúde [36].

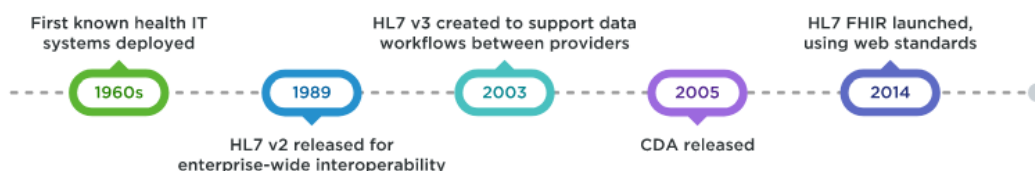


Figura 11 - Desenvolvimento dos *standards* HL7 ao longo do tempo [37].

O primeiro *standard* a ser desenvolvido por HL7 teve cerca de 20 anos de desenvolvimento contendo troca, gestão e integração das informações médicas, porém nunca foi implementado em serviços reais. Posteriormente à primeira versão, foi desenvolvida a versão 2.0 um ano mais

tarde, tornando-se a primeira versão com implementação no mundo real, sendo ainda nos dias de hoje a versão mais amplamente utilizada no mundo [35], [36].

Apesar de ser o *standard* mais utilizado mundialmente nos sistemas de saúde, a versão v2.0 não é perfeita, uma vez que é considerada difícil implementar as suas interfaces, muitos campos não estão claramente definidos e existem muitos outros que são opcionais. Todos estes desafios levaram ao desenvolvimento do *standard* HL7 v3.0 em 1997. Esta versão é baseada em informação orientada a objetos denominada de *Reference Information Model* (RIM), sendo o principal objetivo reduzir custos de implementação mantendo o *standard* entre os sistemas informáticos médicos [35].

Em 2005, o HL7 *Clinical Document Architecture* (CDA) foi criado, utilizando como base XML com o objetivo de estruturar e normalizar a troca de documentos clínicos, definindo de que forma estes são enviados e recebidos [35].

O mais recente *standard* desenvolvido é denominado de *Fast Healthcare Interoperability Resources* (FHIR). Este é uma combinação entre alguns dos melhores aspetos da versão v2.0 e da versão v3.0 com a tecnologia WEB, como a arquitetura *Representational State Transfer* (REST), de forma a melhorar a implementação do *standard* [38].

Apesar das suas diferenças de implementação e estrutura, todos existem em simultâneo e é normal em sistemas reais existirem várias versões do *standard* implementadas no mesmo sistema de informação de saúde [38].

2.5.2 Estrutura

Uma mensagem de HL7 consiste num ou vários segmentos, estando cada segmento dividido em campos separados pelo símbolo “|”. Cada um dos campos pode ser ainda dividido em subcampos separados pelo símbolo “^” [39].

Existem mais de 120 tipos diferentes de segmentos HL7 disponíveis. No seguinte exemplo são utilizados os segmentos MSH, PID, PV1 e EVN, sendo estes os segmentos obrigatórios numa mensagem do tipo ADT_A01, mensagem acionada para a admissão de pacientes:

```
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240617185  
912||ADT^A01|Obef0bdd-c553-401d-8d9f-14dc559ca9de|P|2.5.1  
PID||189316|||Fernandes^Rui||19770728|M|||Alameda dos  
Condes^^Queluz^^Portugal||917379814  
PV1|189316|||ICU|EMER|#####|20240617185912  
EVN||20240617185912
```

O segmento MSH é o cabeçalho obrigatório em qualquer tipo de mensagem, contendo informações acerca da mensagem como o sistema recetor, sistema que envia, a data e hora da mensagem, o evento e a versão. O PID contém informações demográficas dos pacientes como o nome, ID e morada. O PV1 é o segmento que define a estadia do paciente no provedor de

saúde como por exemplo, a localização. Finalmente o EVN é usado para comunicar informações sobre o evento para aplicações receptoras como código do evento e data [40].

2.5.3 Eventos HL7 v2.0

As mensagens na versão v2.0 são construídas com eventos que são acionados para especificar determinado tipo de ação. Existem vários tipos de mensagens como ADT, ORM, ORU [35].

A mensagem do tipo ADT (*Admission, Discharge, Transfer*) é usada como forma de transmitir nova informação demográfica e da visita do paciente no hospital. Como forma de diferenciar o tipo de evento, é acrescentado um código que o descreve. No caso da ADT existem entre os demais códigos o A01 para descrever a admissão de um paciente, A02 para transferência de um paciente para outra localização e o A03 para a alta do paciente [41].

A *General Order Message* (ORM) é um tipo de mensagem usado para descrever a transmissão de informação acerca do serviço, como requisição de novos serviços, cancelamento do existente, colocar algum serviço em espera entre outros. Da mesma forma que o tipo de mensagem ADT, existem códigos associados à ORM como O01, utilizada para iniciar a transmissão de informação acerca de um serviço [42].

A ORU (*Observation Result*) é o tipo de mensagem utilizado para transmitir informação acerca dos dados clínicos do paciente e usado como forma de responder a um serviço requerido pela ORM, sendo utilizado o código O01 [42].

2.6 Conclusão do Capítulo 2

Um simulador é considerado um equipamento desenvolvido para representar condições reais. Historicamente, a simulação tem sido essencial em áreas como a área militar, aviação, educação e saúde. Um dos primeiros simuladores foi o *Link Trainer*, desenvolvido por Edwin Albert Link nos anos 1920, utilizado para treinar pilotos durante a Segunda Guerra Mundial. Na educação, as simulações são usadas para desenvolver capacidades técnicas, pensamento crítico e trabalho em equipa, o que aumentou exponencialmente o uso de realidade virtual e jogos sérios, como o *Swift Playgrounds*, que ensina programação de forma fácil e interativa.

Na saúde, simulações realísticas são usadas para treinar cenários reais e de enorme responsabilidade sem riscos para pacientes ou sistemas de informação. Exemplos incluem a simulação discreta de eventos para prever o fluxo de pacientes em hospitais e o *Patient Manager*, sistema utilizado para o teste de integrações HL7, garantindo a sua conformidade com os *standards* aplicados a sistemas informáticos de saúde.

As máquinas de estados são modelos matemáticos utilizados para descrever comportamentos de sistemas através de estados e transições que geram um *output* dependendo do seu *input*, sendo um exemplo básico os torniquetes com inserção de moedas.

A Interoperabilidade é a capacidade de diferentes sistemas trocarem dados de forma segura e estruturada. Esta pode-se classificar em quatro níveis: fundamental, estrutural, semântico e organizacional. Exemplos de interoperabilidade incluem EHRs e o *standard* DICOM.

A organização IHE promove a interoperabilidade, tendo sido criados os perfis IHE. O HL7 é uma organização que desenvolve *standards* para compartilhamento de dados entre sistemas de informação na saúde, tendo vindo a evoluir em diferentes versões do *standard*, sendo a mais recente o FHIR, que combina aspetos das versões anteriores com a tecnologia WEB.

As mensagens HL7 consistem em segmentos divididos, estando estes mais uma vez divididos em campos. Os tipos de mensagens HL7 incluem, entre muitos outros ADT, (*Admission, Discharge, Transfer*), ORM (*General Order Message*) e ORU (*Observation Result*), cada uma com códigos específicos para diferentes eventos.

3 Métodos e Aplicação

Neste capítulo são mencionadas as tecnologias e bibliotecas utilizadas e a sua função no desenvolvimento do projeto. É igualmente explanado a razão pela qual elas foram escolhidas, assim como a metodologia definida para as aplicar. No final é apresentado o diagrama da máquina de estados criada.

3.1 Tecnologias utilizadas

Numa fase inicial do projeto foram ponderadas várias linguagens de programação para o desenvolvimento deste projeto, entre as quais Java, Python e C#. Após ponderação, o fator de decisão focava-se na linguagem mais bem preparada para a aplicação de bibliotecas HL7 necessárias para o projeto. Dentro das demais linguagens de programação foi feita uma pesquisa das bibliotecas HL7 mais utilizadas, sendo que para cada tipo de linguagem existe uma biblioteca correspondente. Assim, para a linguagem em Java a biblioteca utilizada é a HAPI, para Python é usada a `python-hl7` e para C# a `NHapi` [43], [44], [45].

A escolha final foi utilizar Java devido a conter a biblioteca mais sólida e com melhor documentação, a HAPI, sendo que as restantes bibliotecas são uma adaptação desta para as mais diversas linguagens de programação. Assim, Java oferece uma boa base para o desenvolvimento do *software* e HAPI estende esta funcionalidade para a área da saúde tornando possível a criação do simulador de integrações de HL7.

De forma a desenvolver o GUI (*Graphical User Interface*) deste projeto, foi escolhido o Java FX. Este *software* é uma ferramenta eficaz para criar aplicações gráficas, uma vez que inclui uma vasta gama de controlos aplicativos, uma API para gráficos 3D e capacidades de animação e efeitos [46].

De forma a manter o tipo de formação e estilo da interface gráfica da aplicação, recorreu-se à utilização e criação de um ficheiro CSS. *Cascading Style Sheets* é uma linguagem utilizada para descrever a apresentação de um sistema. Controla a disposição, as cores, os tipos de letra e o aspeto visual geral das aplicações. Os ficheiros CSS permitem separar o conteúdo do design, permitindo uma maior flexibilidade e controlo na especificação do estilo dos elementos, facilitando a manutenção e a atualização do seu aspeto [47].

HAPI, *HL7 Application Programming Interface*, é uma biblioteca desenvolvida pela rede de saúde Universitária no Canadá como sendo uma ferramenta *open-source* de análise de mensagens de HL7 orientada a objetos para Java. Apesar de não afiliados à entidade HL7, o seu desenvolvimento foi desenhado de forma a seguir o *standard* [45], [48].

A HAPI foi a biblioteca HL7 escolhida para auxiliar no desenvolvido deste projeto, sendo o motivo da sua escolha mencionado anteriormente.

Finalmente, de forma a desenvolver e compilar o código produzido, utilizou-se o *Integrated Development Environment (IDE) IntelliJ IDEA Community Version* [49].

3.2 Estrutura do Projeto Desenvolvido

O simulador desenvolvido encontra-se organizado da forma descrita graficamente na Figura 12:

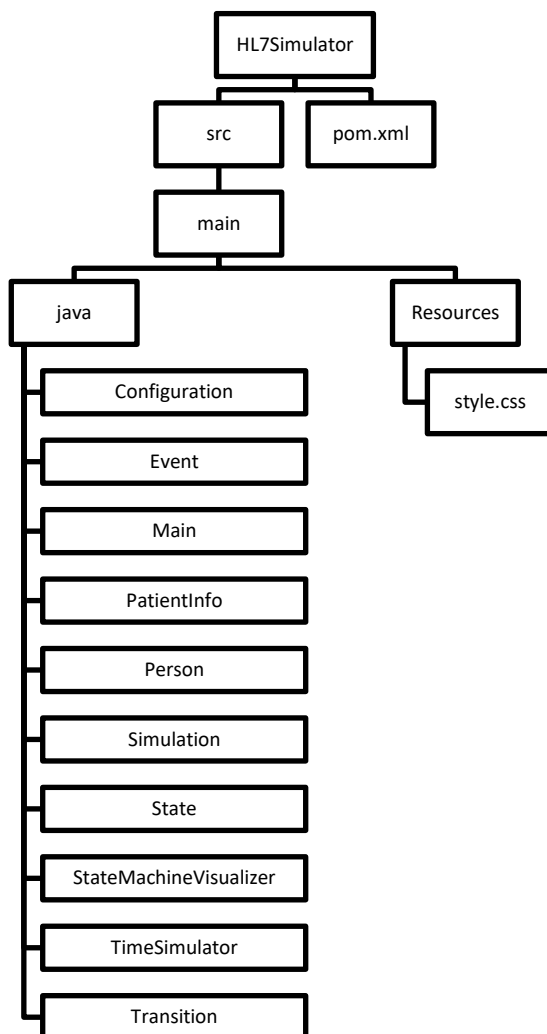


Figura 12 - Estrutura do Projeto Desenvolvido.

A raiz do projeto foi denominada de “HL7Simulator”, contendo dois diretórios, um para o código propriamente dito (“SRC”) e outro para o ficheiro pom.xml, utilizado para definir as dependências, configurações, bibliotecas entre outros. Neste caso, utilizou-se para definir a biblioteca HAPI e a sua versão, o processamento de ficheiros JSON gerados no *software* e a versão utilizada no JavaFX. De igual forma, também é especificado neste ficheiro o repositório Maven, utilizado para extrair todas estas dependências assim como a versão do Java utilizada.

Dentro do diretório “SRC” é disponibilizado todo o material necessário para compilar o código do projeto, incluindo os subdiretórios “main” que por sua vez contém os subdiretórios java e “resources”. O diretório java contém todas as classes Java necessárias no projeto. De forma oposta, o diretório “resources” inclui todos os conteúdos não java, mas necessários para o

desenvolvimento e compilação do projeto. Neste caso, utilizou-se este diretório para guardar o ficheiro CSS denominado de style.css.

A Figura 13 descreve todas as classes criadas no contexto deste projeto, constituídas por métodos, variáveis, atributos e objetos, assim como a forma como se relacionam. Para um desenvolvimento funcional foram criadas as seguintes classes: “Configuration.java”, “Event.java”, “Main.java”, “PatientInfo.java”, “Person.java”, “Simulation.java”, “State.java”, “StateMachine.java”, “StateMachineVisualizer.java”, “TimeSimulator.java”, “Transition.java”.

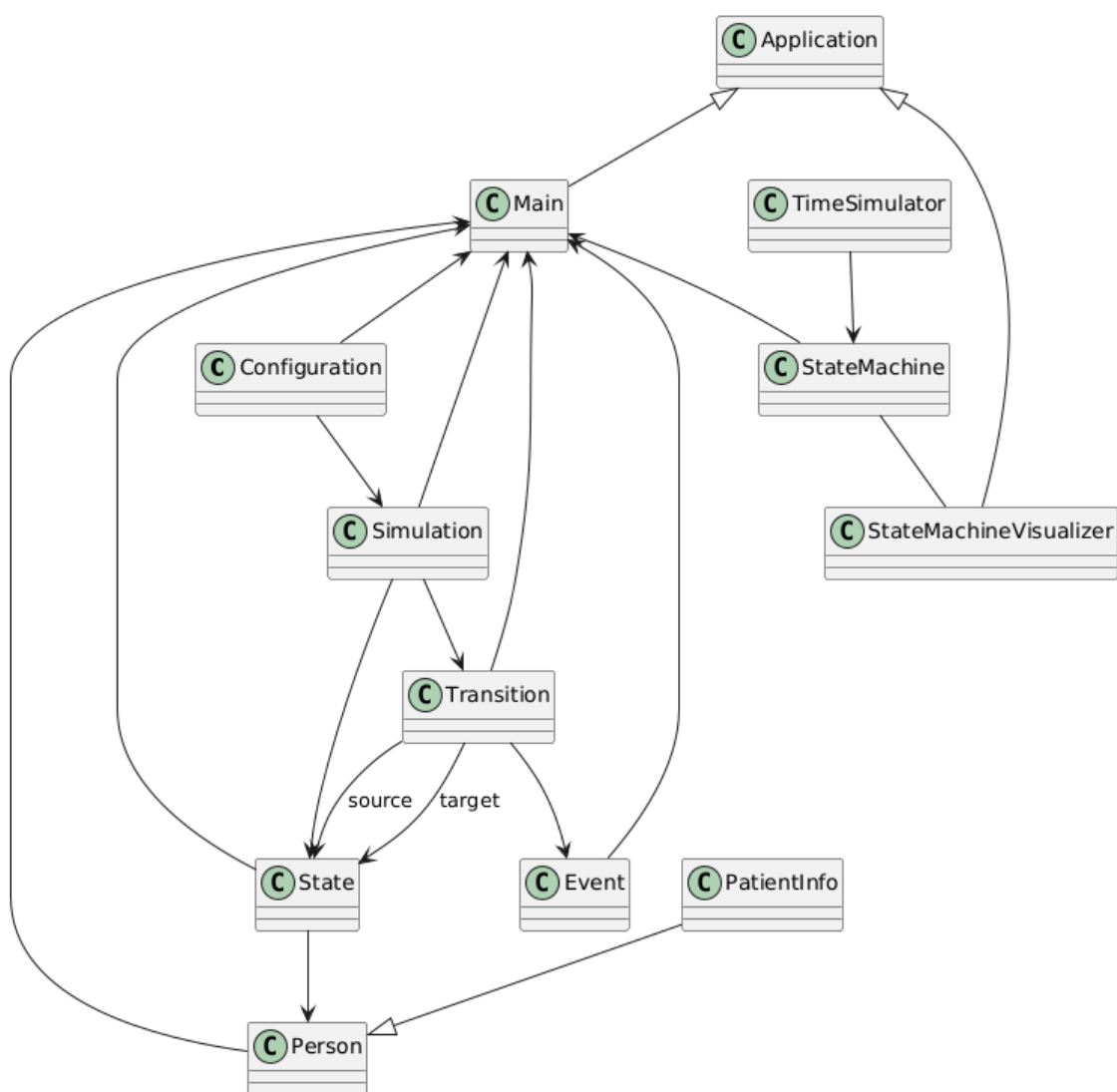


Figura 13 - Diagrama de Classes do Projeto Desenvolvido.

A classe "Main.java" é o principal interveniente na compilação e apresentação do código presente em todas as outras classes visto que todas estão presentes direta ou indiretamente (Figura 14). Estão presentes nesta classe as condições em que são criadas, guardadas e modificadas as configurações da máquina de estados, presente na classe Configuration.java, assim como criação da interface gráfica.

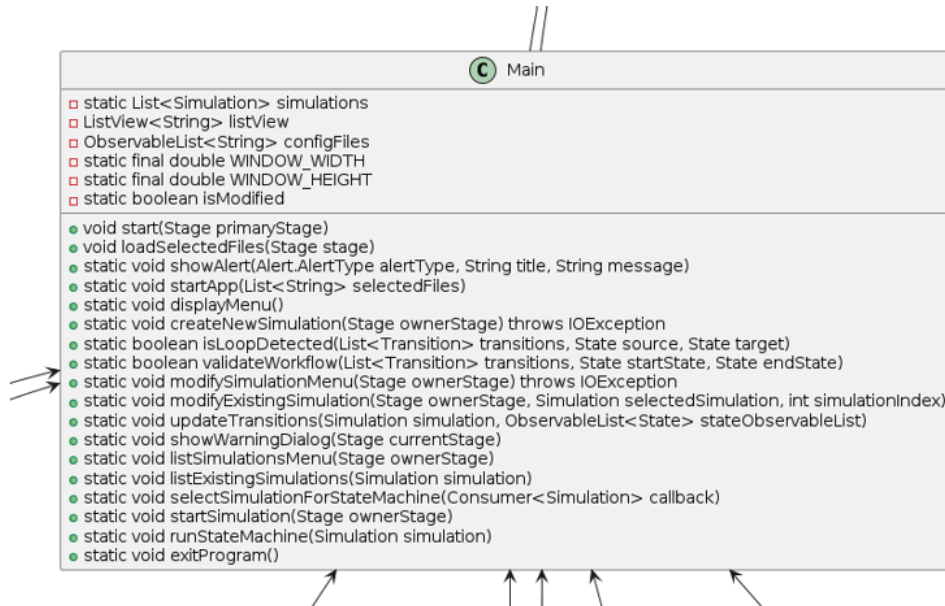


Figura 14 - Classe "Main.java".

A "Configuration.java" é a classe do programa que lida com a escrita, alteração e leitura dos ficheiros JSON utilizados para manter as informações necessárias para os elementos da máquina de estados (Figura 15).

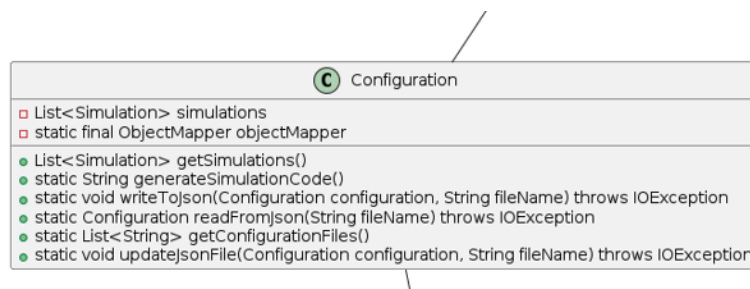


Figura 15 - Classe "Configuration.java".

Desta forma, é utilizada a classe "Simulation.java" (Figura 16) para gravar as informações acerca dos estados da máquina de estados, presente na classe "State.java", as transições entre estados, presente na classe "Transition.java", assim como o estado de início e fim da máquina de estados, bem como o nome da simulação correspondente.

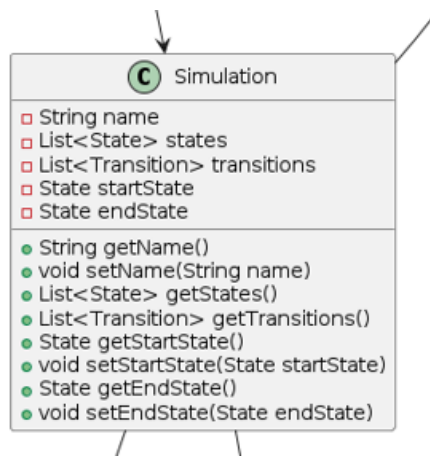


Figura 16 - Classe "Simulation.java".

Na classe "State.java" são definidos os nomes dos estados, a sua máxima capacidade, a lista das pessoas presentes no estado e o ID do estado correspondente (Figura 17). A definição de capacidade máxima é definida de forma que a entrada num estado não seja ilimitada, considerando que em cenários reais existe um limite de ocupação de espaço.

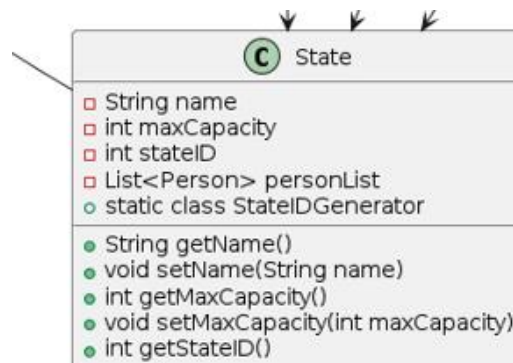


Figura 17 - Classe "State.java".

De igual forma, na classe "Transition.java" é definido o estado do início e fim da transição, a probabilidade desta transição acontecer e a frequência em que acontece uma determinada transição (Figura 18). A probabilidade é estipulada pelo utilizador, escolhendo um número entre 0.01 e 1, sendo 1 equivalente a 100% de probabilidade de a transição acontecer. Associada a esta probabilidade encontra-se a frequência que o utilizador define como sendo um número inteiro da quantidade de eventos que ocorrem por minuto. Estas duas variáveis são definidas em conjunto de forma que a máquina de estados se torna mais realista, visto que não existe um determinado tempo para as transições ocorrerem entre estados.

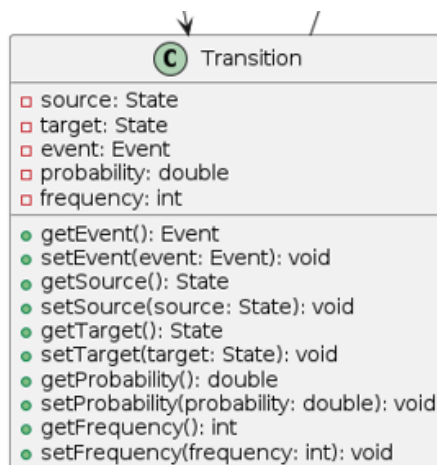


Figura 18 - Classe "Transition.java".

A cada transição está associado a classe "Event.java" que define os eventos que determinada transição aciona (Figura 19). Estes eventos podem gerar uma mensagem HL7 ou não gerar evento. Assim, nesta classe são definidos os nomes dos eventos, o seu ID e a lista de todos os eventos disponíveis na simulação.

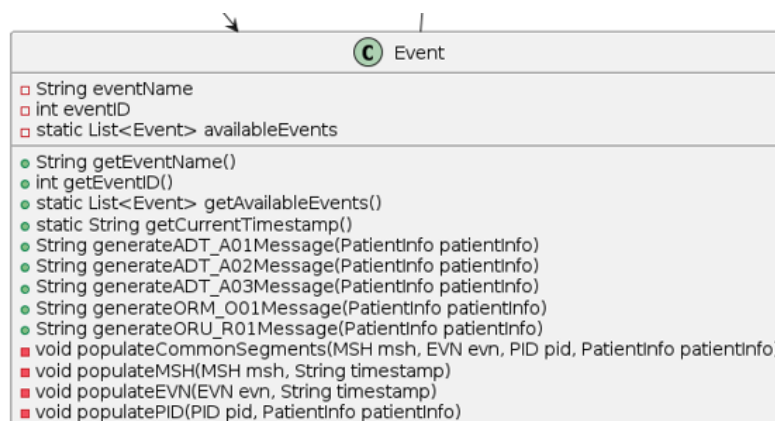


Figura 19 - Classe "Event.java".

A classe "Person.java" define todas as informações relativas às pessoas geradas na simulação, incluindo o nome, género, data de nascimento, morada, telemóvel e país. Esta classe também funciona como gerador aleatório destas informações que posteriormente serão utilizadas para preencher as mensagens de HL7 no decorrer da simulação, assim como em que posição da máquina de estado é que as pessoas se encontram. Associada a esta encontra-se a classe PatientInfo.java que contém informações relativas ao paciente, visto que nem todas as pessoas envolvidas na simulação necessitam de ser pacientes (Figura 20).



Figura 20 - Classes "Person.java" e "PatientInfo.java".

Por fim, a classe "StateMachine.java" define as condições em que a máquina de estados opera, considerando todas as informações guardadas na classe "Simulation.java" (Figura 21).



Figura 21 - Classe "StateMachine.java".

Associada a esta encontram-se as classes "TimeSimulator.java" para simular o tempo da simulação, de forma que o utilizador consiga acelerar o tempo relativo da simulação, assim como a classe "StateMachineVisualizer.java" para disponibilizar o diagrama da máquina de estados da simulação em questão (Figura 22).

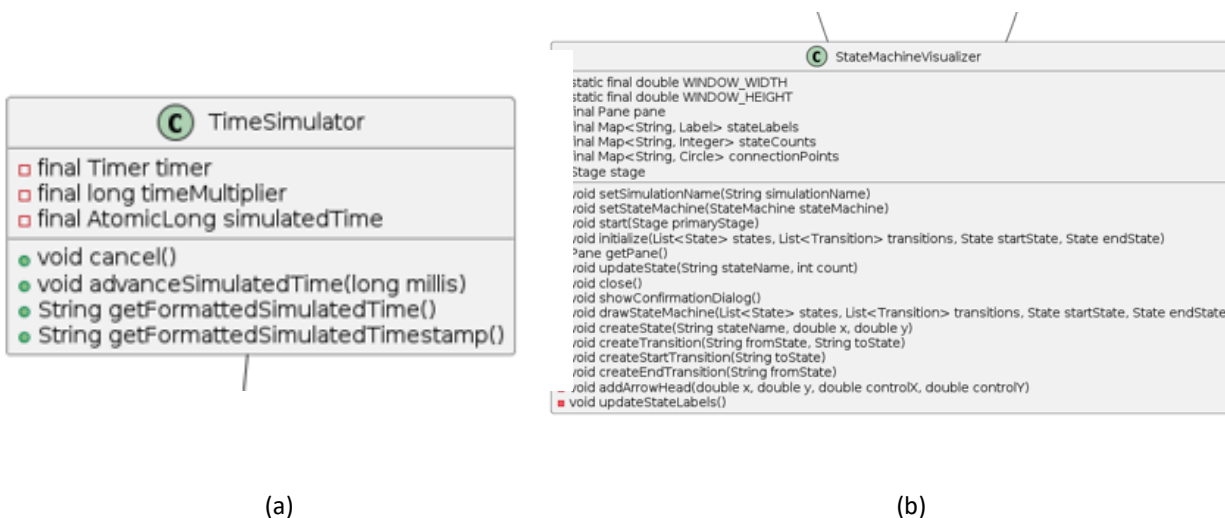


Figura 22 - Classes (a) "TimeSimulator.java" e (b) "StateMachineVisualizer.java".

3.3 Conclusão do Capítulo 3

No início do projeto, várias linguagens de programação foram consideradas, incluindo Java, Python e C#. A decisão final foi de utilizar Java devido à biblioteca HAPI, que oferece uma base sólida e documentação robusta para integração de HL7 no *software* desenvolvido. Para o desenvolvimento da interface gráfica, foi escolhido o Java FX utilizado juntamente com CSS, permitindo maior flexibilidade e conformidade na aparência.

A estrutura do projeto é definida pelo diretório principal, "HL7Simulator", que contém os diretórios "SRC" para o código em Java e "resources" para conteúdos não-Java.

A classe principal é a "Main.java", que compila e apresenta o código das outras classes. Configuration.java gere a escrita e leitura dos ficheiros JSON necessários para a máquina de estados. "State.java" define os nomes e capacidades máximas dos estados. A classe "Transition.java" estabelece as transições entre estados, assim como a probabilidades e frequência. Também a classe "Event.java" define os eventos associados às transições, incluindo a forma como são geradas as mensagens HL7.

O "Person.java" gera informações das pessoas na simulação, enquanto "PatientInfo.java" armazena dados específicos de pacientes. A classe "StateMachine.java" determina as condições operacionais da máquina de estados, utilizando as informações da classe "Simulation.java".

Esta organização estrutural do projeto facilita o desenvolvimento, manutenção e expansão do simulador criado.

4 Resultados e Discussão

Este capítulo inicia-se com a apresentação dos resultados do desenvolvimento do projeto, descrevendo as funcionalidades e conceitos desenvolvidos no contexto da aplicação. Numa segunda parte, é descrita a discussão dos resultados obtidos.

4.1 Apresentação de Resultados

4.1.1 *Configurações da Simulação*

De forma a que o utilizador tenha a possibilidade de definir o cenário a simular, este tem de configurar os elementos necessários para a simulação, isto é, os estados, as transições entre estados e os eventos associados às transições.

O programa é iniciado com a lista de ficheiros de configuração, do tipo JSON, que o utilizador possa já ter definido previamente ou que defina no momento. Caso o utilizador já tenha um ficheiro de configurações prévio, terá de fazer o *upload* do mesmo (Figura 23).

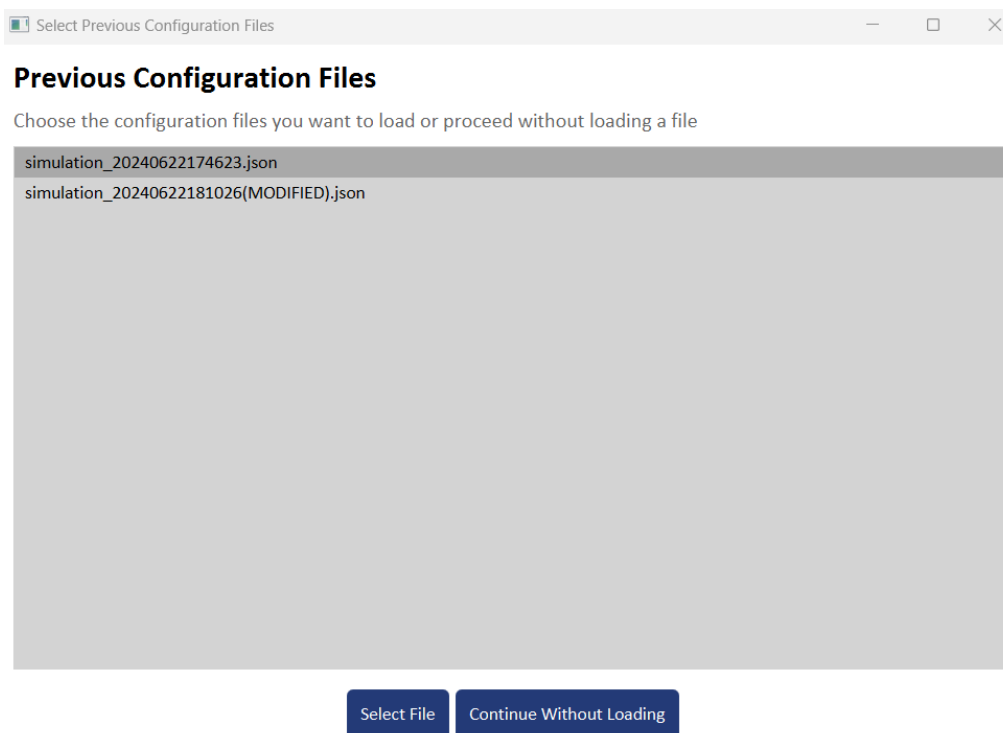


Figura 23 - Listagem de configurações já existentes.

No caso de não existirem quaisquer ficheiros de configuração, uma mensagem é apresentada ao utilizador (Figura 24).

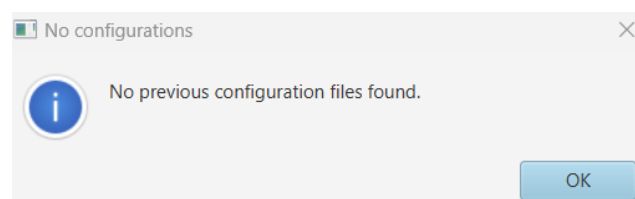


Figura 24 - Aviso caso não existam ficheiros de configuração.

Seguidamente é apresentado o menu principal da aplicação, constituído pelas opções possíveis do programa, nomeadamente a criação de uma nova configuração, modificação de uma configuração existente, listagem de configurações, início da simulação, carregamento de configurações existentes e a opção de saída do programa (Figura 25).

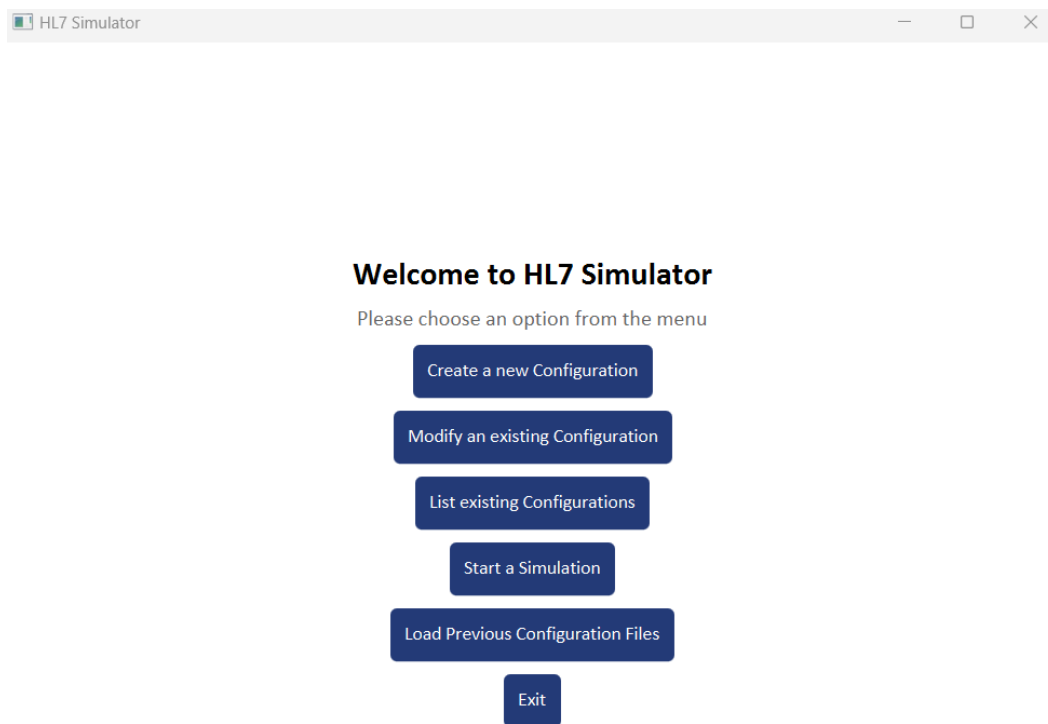


Figura 25 - Menu principal da aplicação.

Para que seja possível simular um cenário, uma configuração tem de ser definida previamente. Desta forma o utilizador deve escolher a primeira opção deste menu. Numa primeira fase é necessário definir o nome da simulação (Figura 26).

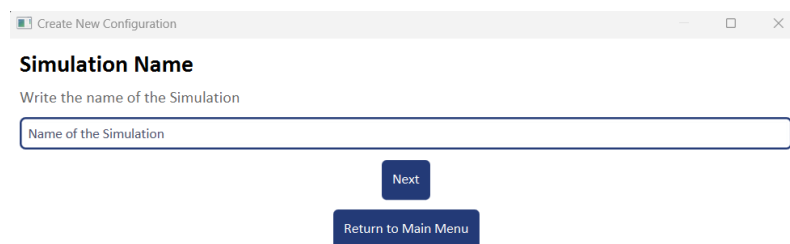


Figura 26 - Escolha do nome da Simulação.

Seguidamente, são escolhidos o nome dos estados e a sua capacidade máxima. O utilizador deve escolher no mínimo dois estados. Assim que o utilizador adiciona os estados estes são listados no espaço inferior, mencionando os detalhes definidos assim como o ID correspondente (Figura 27).

Create States for the Simulation

Write the name and the max capacity for the states of the Simulation

Enter the name of the state

State Name

Enter the max capacity of the state

Max Capacity

Entrada (ID: 314, Max Capacity: 50)
 Consulta (ID: 315, Max Capacity: 1)
 Sala de Espera (ID: 316, Max Capacity: 13)
 Saída (ID: 317, Max Capacity: 33)

Back Add State Next

Return to Main Menu

Figura 27 - Criação dos estados da Simulação.

Para que a máquina de estados funcione de forma eficaz é necessário definir o estado de início e o de fim. São então apresentadas as opções de estados disponíveis para o estado de início e o estado de fim (Figura 28).

Start and End States

Choose the Start and End state for your Simulation

Select Start State

Select End State

Back Next

Return to Main Menu

Start and End States

Choose the Start and End state for your Simulation

Select Start State

Entrada (ID: 314, Max Capacity: 50)
 Consulta (ID: 315, Max Capacity: 1)
 Sala de Espera (ID: 316, Max Capacity: 13)
 Saída (ID: 317, Max Capacity: 33)

Back Next

Return to Main Menu

(a) (b)

Figura 28 - Definição dos estados de início e de fim da Simulação (a), sendo apresentadas as opções possíveis de estados (b).

Depois de definir o estado de início e o de fim, é necessário definir as condições das transições e respectivos eventos de entrada da simulação e de saída da simulação. Assim, o utilizador deve definir a frequência de entrada na simulação (número inteiro) assim como a probabilidade dessa transição acontecer (valor entre 0.01 e 1) (Figura 29).

Create New Configuration

Start and End state Transitions

Configure the special transitions for the Start State (People added to the Simulation) and End State (People removed from the Simulation)

Set the frequency (per minute) of the Start State

Set the probability for the Start State (0.01 - 1)

Select Event for this Transition

Set the frequency (per minute) of the End State

Set the probability for the End state (0.01 - 1)

Select Event for this Transition

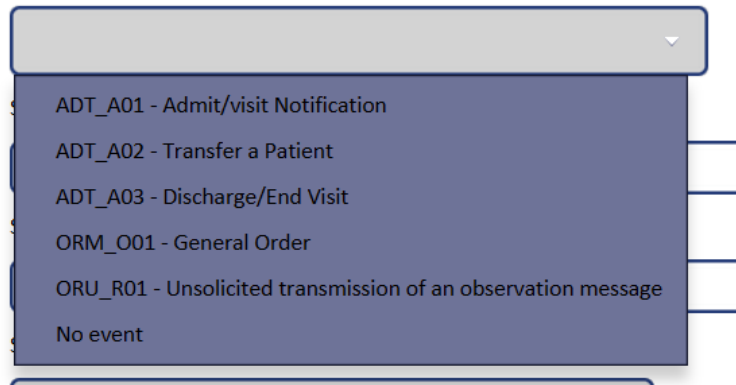
[Back](#) [Next](#)

[Return to Main Menu](#)

Figura 29 - Definição das condições da Transição de início e de fim da simulação.

Em cada transição podem ocorrer dois casos: A transição aciona a criação de uma mensagem HL7 de acordo com a lógica que deseja testar na simulação, ou a transição não aciona uma mensagem HL7, visto que, em situações reais, nem todas as transições exigem a criação de mensagens HL7 nos sistemas de informação. Assim, as opções disponíveis no programa criado são apresentadas na Figura 30.

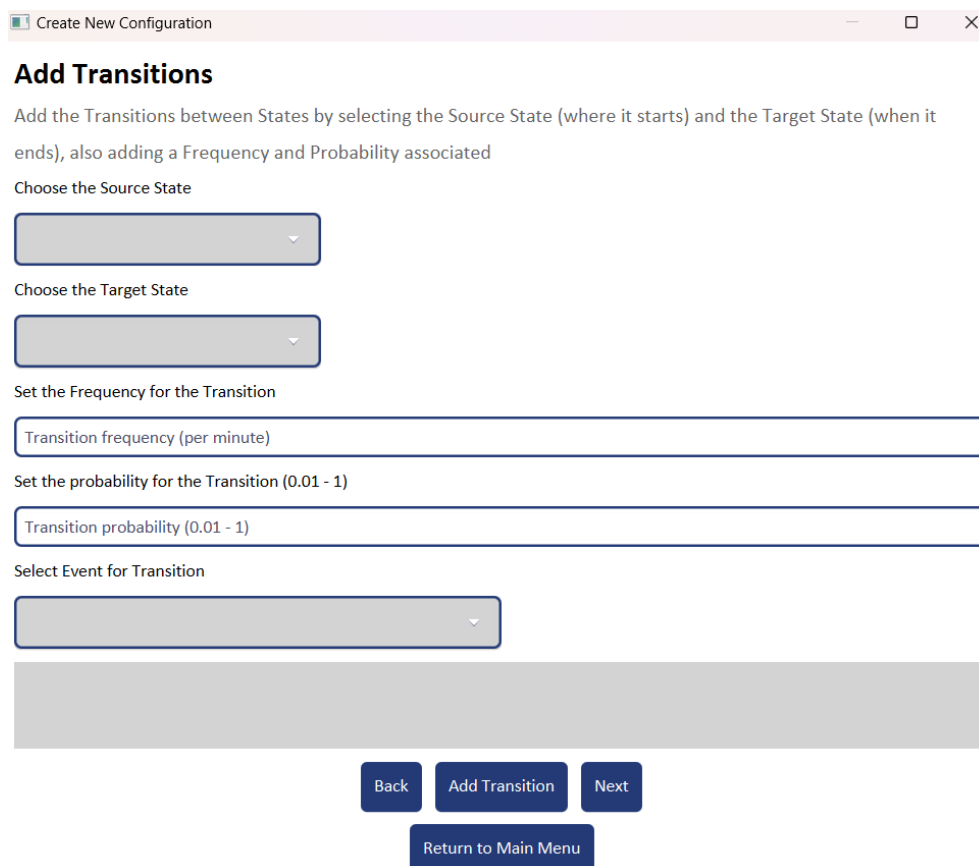
Select Event for this Transition



A dropdown menu with a light gray header and a dark blue body. The header contains a downward-pointing arrow. The body lists six options: ADT_A01 - Admit/visit Notification, ADT_A02 - Transfer a Patient, ADT_A03 - Discharge/End Visit, ORM_O01 - General Order, ORU_R01 - Unsolicited transmission of an observation message, and No event. To the right of the menu, there are three horizontal lines of varying lengths, suggesting a list of transitions.

Figura 30 - Opções disponíveis para os Eventos.

Após esta definição inicial e final da simulação, é necessário definir os caminhos disponíveis entre os restantes estados. Desta forma, a lógica anterior mantém-se, acrescentando apenas a opção de escolha do estado onde a transição inicia e onde a transição acaba (Figura 31).



The screenshot shows a window titled 'Create New Configuration' with standard window controls. The main heading is 'Add Transitions'. Below it, there is a descriptive text: 'Add the Transitions between States by selecting the Source State (where it starts) and the Target State (when it ends), also adding a Frequency and Probability associated'. The form includes several input fields: 'Choose the Source State' (a dropdown menu), 'Choose the Target State' (a dropdown menu), 'Set the Frequency for the Transition' (a text input field with the placeholder 'Transition frequency (per minute)'), 'Set the probability for the Transition (0.01 - 1)' (a text input field with the placeholder 'Transition probability (0.01 - 1)'), and 'Select Event for Transition' (a dropdown menu). At the bottom of the window, there are four buttons: 'Back', 'Add Transition', 'Next', and 'Return to Main Menu'.

Figura 31 - Definição das transições entre estados, assim como os eventos.

Na zona inferior da janela são igualmente apresentadas as transições criadas, assim como as suas características (Figura 32).

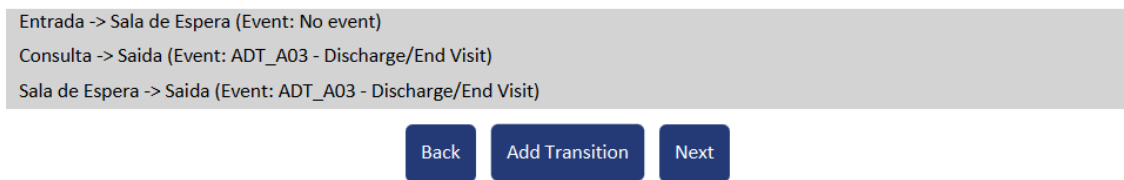


Figura 32 - Listagem das transições criadas.

Caso as definições não sejam corretamente definidas, por exemplo, no caso do caminho escolhido não ser definido deste o estado de início até ao estado de fim, um aviso de erro é apresentado ao utilizador para que este possa corrigir as suas configurações (Figura 33).

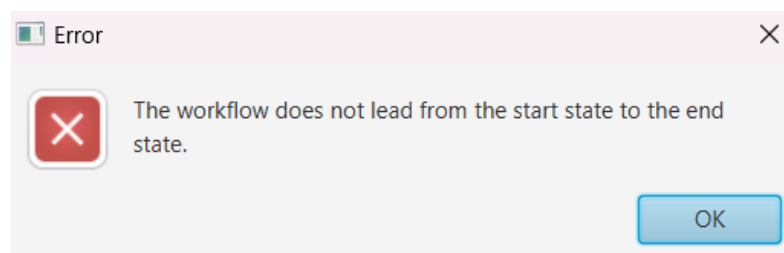


Figura 33 - Aviso apresentado caso as configurações das transições não sejam coerentes.

Finalmente, são apresentados os detalhes da simulação criada assim como o diagrama da máquina de estados (Figura 34).

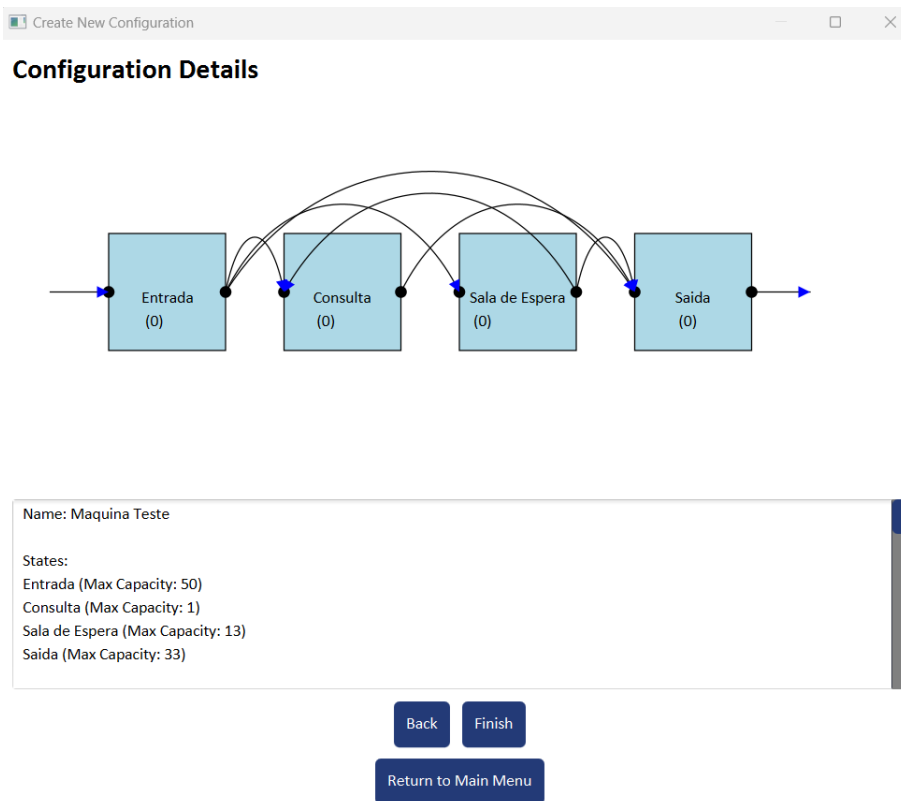


Figura 34 - Detalhes da Configuração da Simulação.

Assim que a opção “Finish” é seleccionada, um aviso é dado ao utilizador de que o ficheiro de configuração foi guardado com o nome apresentado. Os ficheiros são guardados numa pasta denominada de “Existing Configurations”. O nome do ficheiro foi definido com o prefixo “simulation_” sendo o sufixo a data da criação da configuração de forma que o ficheiro tenha um nome único (Figura 35).

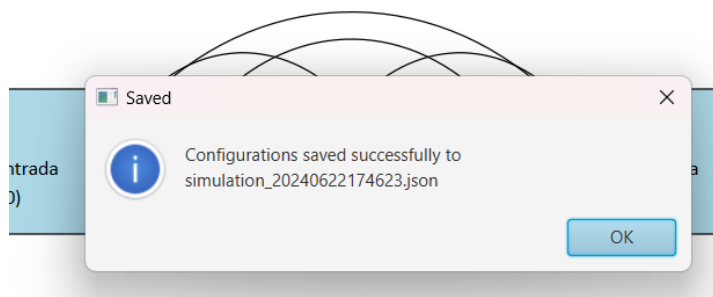


Figura 35 - Aviso apresentado aquando da criação da configuração.

Nestes ficheiros JSON, são guardadas as informações necessárias para o comportamento desejado da máquina de estados. Um dos exemplos do ficheiro de configuração JSON é apresentado no Código 1.

```

{
  "simulations": [
    {
      "name": "Maquina Teste",
      "states": [
        {
          "name": "Entrada",
          "maxCapacity": 50,
          "stateID": 314
        },
        {
          "name": "Saida",
          "maxCapacity": 33,
          "stateID": 317
        }
      ],
      "transitions": [
        {
          "source": null,
          "target": {
            "name": "Entrada",
            "maxCapacity": 50,
            "stateID": 314
          },
          "event": {
            "eventName": "ADT_A01 - Admit/visit Notification",
            "eventID": 1
          },
          "probability": 0.6,
          "frequency": 3
        },
        {
          "source": {
            "name": "Saida",
            "maxCapacity": 33,
            "stateID": 317
          },
          "target": null,
          "event": {
            "eventName": "No event",
            "eventID": 0
          },
          "probability": 0.1,
          "frequency": 4
        },
        {
          "source": {
            "name": "Entrada",
            "maxCapacity": 50,
            "stateID": 314
          },
          "target": {
            "name": "Saida",
            "maxCapacity": 33,
            "stateID": 317
          },
          "event": {
            "eventName": "ADT_A03 - Discharge/End Visit",
            "eventID": 3
          },
          "probability": 0.3,
          "frequency": 2
        }
      ],
      "startState": {
        "name": "Entrada",
        "maxCapacity": 50,
        "stateID": 314
      },
      "endState": {
        "name": "Saida",
        "maxCapacity": 33,
        "stateID": 317
      }
    }
  ]
}

```

Código 1 - Exemplo de uma configuração no ficheiro JSON.

Se o utilizador pretender modificar algum aspeto de uma anterior configuração, a segunda opção do menu principal deve ser selecionada. Antes de qualquer alteração o utilizador deve escolher qual a configuração a alterar (Figura 36).

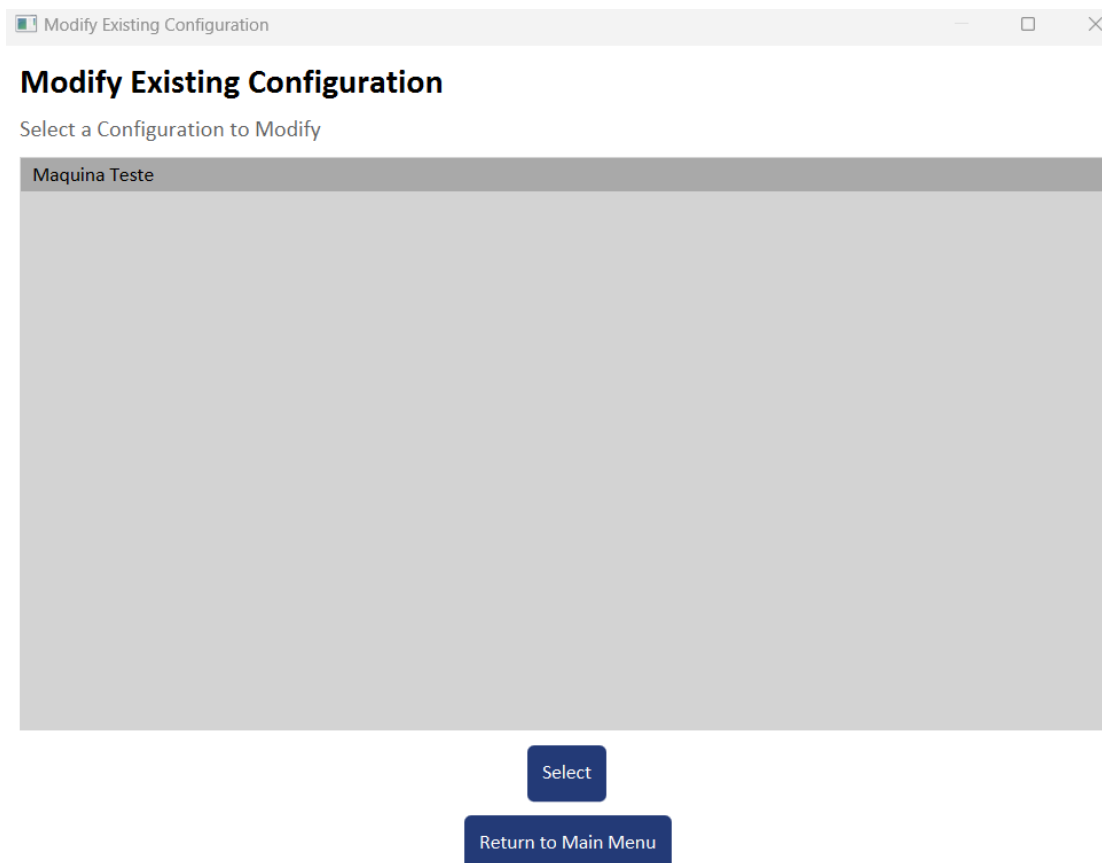


Figura 36 - Seleção da configuração a modificar.

A lógica da modificação é a mesma utilizada na criação da configuração, sendo que os valores anteriores são apresentados nos campos correspondentes. Na modificação dos estados e das transições também é possível adicionar novos estados e transições, respetivamente.

No final, são igualmente apresentados os detalhes da configuração após as alterações efetuadas, assim como o diagrama atualizado (Figura 37).

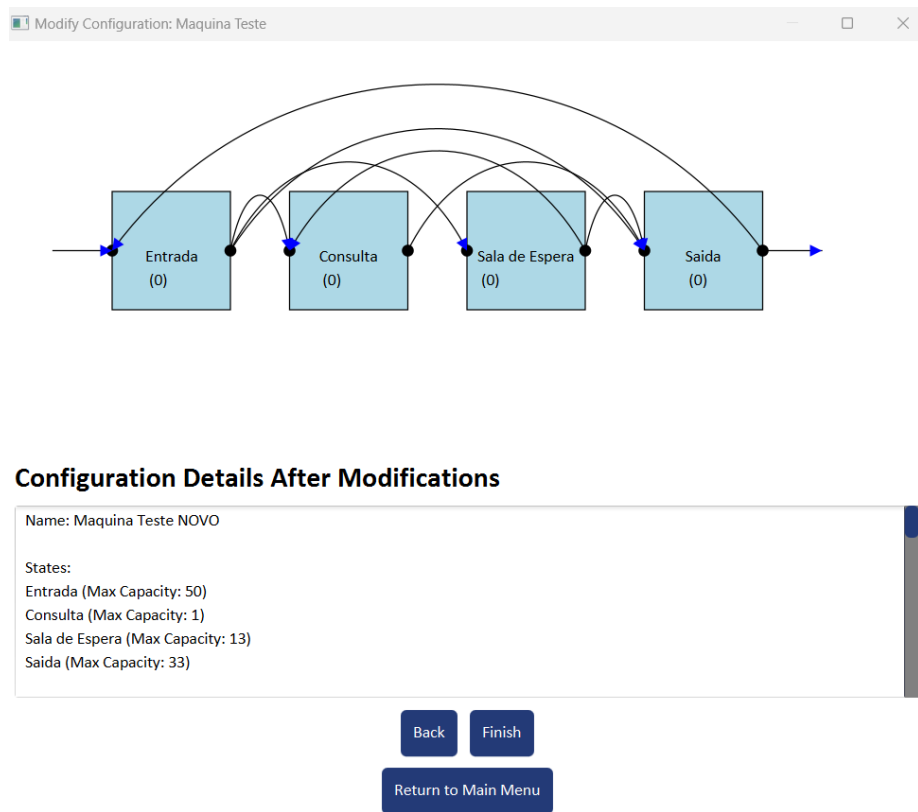


Figura 37 - Detalhes da configuração após modificação.

Tal como acontecia na opção de uma nova configuração, assim que o utilizador clicar em “Finish”, também é apresentado um aviso com o nome do ficheiro de configuração JSON. Caso este seja modificado, é acrescentado o sufixo “(MODIFIED)”, sendo igualmente guardado na pasta “Existing Configurations “(Figura 38).

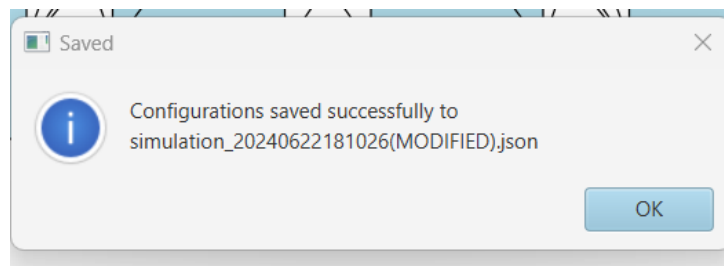


Figura 38 - Aviso apresentado aquando da modificação da configuração.

A terceira opção é utilizada como forma de listar as configurações disponíveis no programa. Assim, o utilizador deve escolher uma configuração da lista apresentada e posteriormente ver ou rever os detalhes da mesma (Figura 39).

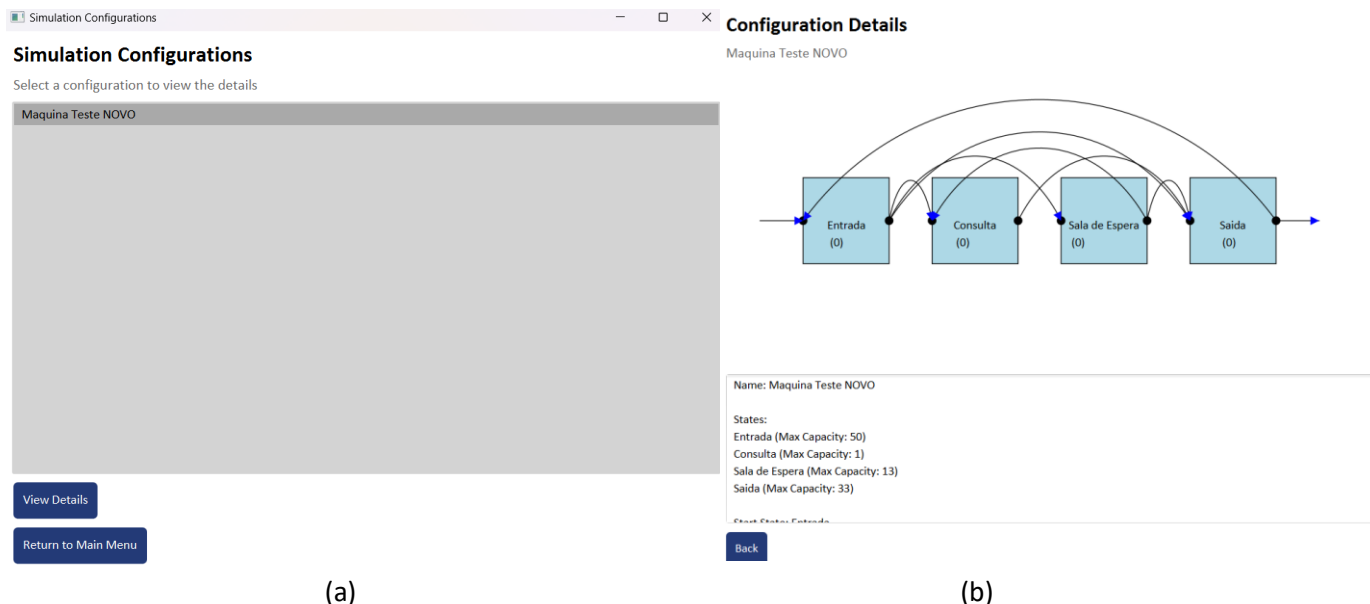


Figura 39 - Listagem das configurações existentes (a) e detalhes da configuração escolhida pelo utilizador (b).

4.1.2 Simulação

A quarta opção do menu principal define o início da simulação. No entanto, é necessário existir uma configuração para que seja possível iniciar a simulação, caso contrário será apresentado um aviso de erro.

Após selecionar esta opção, o utilizador deve escolher qual a simulação a testar sendo de seguida apresentada a página inicial da simulação (Figura 40). Aqui é também possível definir a velocidade da simulação podendo ser em tempo-real (1.0x) ou duas vezes mais rápido (2.0x).

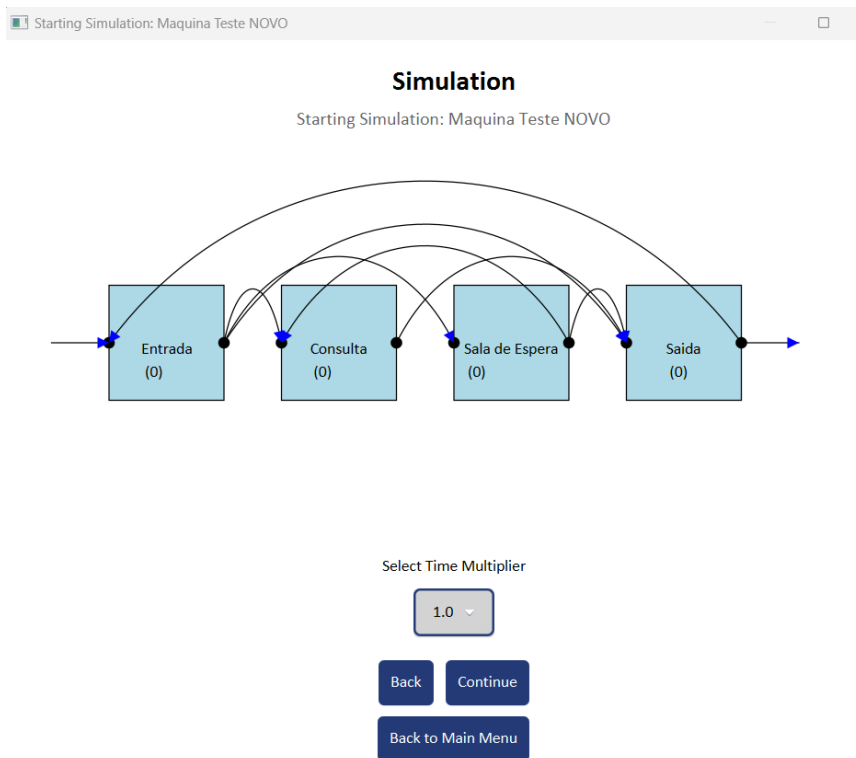


Figura 40 - Inicio da Simulação escolhida.

Assim que o utilizador clicar em "Continue", a simulação é iniciada, sendo apresentado o diagrama da máquina de estados com o nome dos estados e as transições entre si, sendo que o número localizado inferiormente ao nome do estado varia consoante a população de pessoas nos estados da simulação (Figura 41).

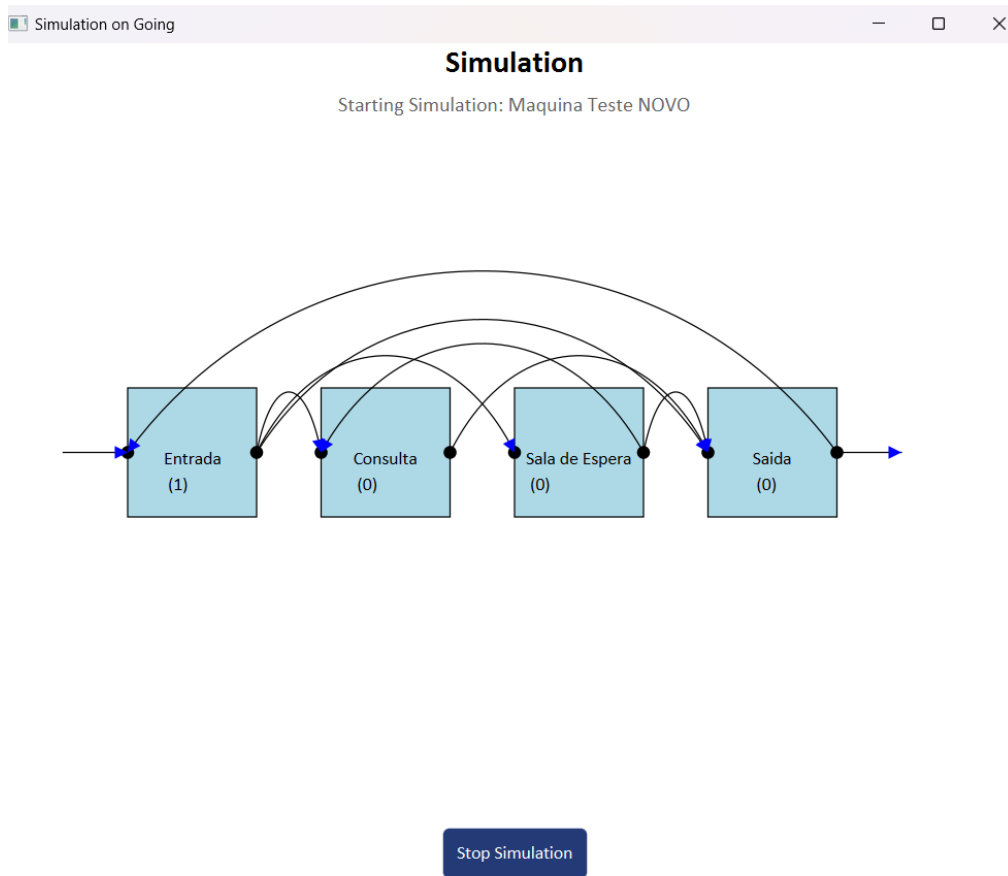


Figura 41 - Simulação a decorrer.

Assim que todas as pessoas geradas na simulação saírem da mesma, esta é considerada completa e um aviso é apresentado com as informações de qual o diretório dos *outputs* da simulação (Figura 42). Sempre que é feita uma simulação pela primeira vez, é criada uma subpasta com o nome da simulação na pasta principal "Simulation Logs". Esta subpasta contém os ficheiros de todas as simulações efetuadas até ao momento, igualmente divididas por pastas com o nome da simulação.

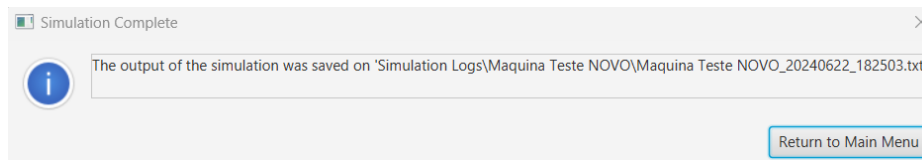


Figura 42 - Aviso aquando do término da simulação.

Para melhor compreensão dos resultados da simulação, o utilizador deve aceder a este diretório e aceder ao ficheiro de texto (.txt) que contém todas as informações, passo a passo, tomadas na simulação (Figura 43). Este ficheiro contém informações sobre quais os estados inicializados na simulação, quais as pessoas que se movimentaram na simulação, gerando ou não mensagens de HL7, assim como o estado da simulação nos vários momentos.

```

1 Thread#25 - 2024-06-22 18:25:03 - Starting state machine for simulation: Maquina Teste NOVO
2 Thread#25 - 2024-06-22 18:25:03 - State lists initialized for states: Sala de Espera, Saida, Consulta, Entrada
3 Thread#25 - 2024-06-22 18:25:03 - 6 - Pedro Oliveira (ID:778084) entered the simulation
4 Thread#25 - 2024-06-22 18:25:03 - 6 - Event: ADT_A01 - Admit/visit Notification
5 Thread#25 - 2024-06-22 18:25:03 - 6 -
6 MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622182503||ADT^A01|20da3c32-1290-49d3-9c00-a6d57c4d45ae|P|2.5.1
7 PID|778084|||Oliveira^Pedro||19660202|M|||Praceta da feira^^Lisboa^^^Portugal||910699015
8 EVN|778084|I|ICU|EMER|||||20240622182503
9 Thread#46 - 2024-06-22 18:25:03 - 7 - João Martins (ID:836011) entered the simulation
10 Thread#46 - 2024-06-22 18:25:03 - 7 - HL7 Event: ADT_A01 - Admit/visit Notification
11 Thread#46 - 2024-06-22 18:25:03 - 7 -
12 MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622182503||ADT^A01|1b6332fb-01ac-49d0-bc92-e730c6818a31|P|2.5.1
13 PID|836011|||Martins^João||19710902|M|||Rua da luz^^Coimbra^^^Portugal||964604824
14 EVN|836011|I|ICU|EMER|||||20240622182503
15 Thread#46 - 2024-06-22 18:25:03 - Current state of the simulation:
16 Entrada: Pedro Oliveira (ID:778084), João Martins (ID:836011)
17 Consulta:
18 Sala de Espera:
19 Saida:
20 Thread#46 - 2024-06-22 18:25:03 - Updated person count: 2
21 Thread#46 - 2024-06-22 18:25:38 - 8 - Moved from Entrada to Saida
22 Thread#46 - 2024-06-22 18:25:38 - 8 - HL7 Event: ADT_A03 - Discharge/End Visit
23 Thread#46 - 2024-06-22 18:25:38 - 8 -
24 MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622182538||ADT^A03|8022e569-c53c-4283-b8c0-44252f4f6ce3|P|2.5.1
25 PID|778084|||Oliveira^Pedro||19660202|M|||Praceta da feira^^Lisboa^^^Portugal||910699015
26 EVN|778084|I|Ward|E|||||20240622182538
27 Thread#46 - 2024-06-22 18:25:38 - 8 - Moved from Entrada to Saida
28 Thread#46 - 2024-06-22 18:25:38 - 8 - HL7 Event: ADT_A03 - Discharge/End Visit
29 Thread#46 - 2024-06-22 18:25:38 - 8 -
30 MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622182538||ADT^A03|b81027e5-f900-494f-99b4-725b7256900b|P|2.5.1
31 PID|836011|||Martins^João||19710902|M|||Rua da luz^^Coimbra^^^Portugal||964604824
32 EVN|836011|I|Ward|E|||||20240622182538

```

Figura 43 - Exemplo das informações apresentadas no ficheiro de texto após a simulação.

Finalmente, a última opção do menu define o fim do programa, apresentando uma mensagem informativa com o nome da autor e nome do programa (Figura 44).

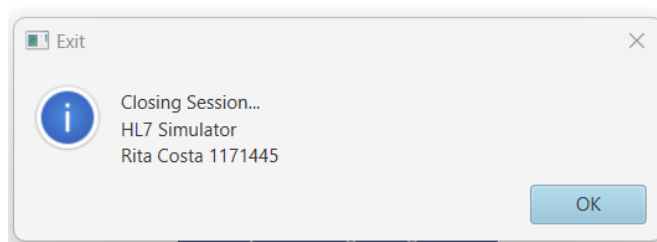


Figura 44 - Término do programa.

4.2 Discussão de Resultados

Seguidamente são apresentados os resultados de três testes no simulador desenvolvido, tendo diferentes dimensões e complexidade, de forma a testar diferentes cenários realistas. O primeiro caso apenas contém dois estados, o segundo contém três estados e o terceiro caso contém cinco estados.

4.2.1 Caso 1

Analizando o caso de uma simulação apenas com dois estados nomeados de *Entrada* e *Saida* (Figura 45). A simulação foi efetuada em tempo-real.

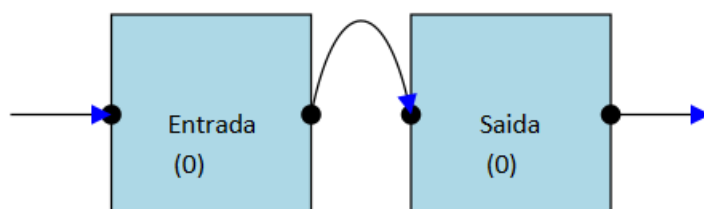


Figura 45 - Simulação exemplo com 2 estados.

As transições definidas para este caso são as seguintes:

Capacidade Máxima do estado *Entrada*: 50

Capacidade Máxima do estado *Saida*: 3

***null* -> *Entrada* (Tipo de mensagem HL7: ADT_A01 - *Admit/visit Notification*)**

Frequência (*null* para *Entrada*): 60

Probabilidade (*null* para *Entrada*): 1.0

***Saida* -> *null* (Tipo de mensagem HL7: Sem mensagem HL7 associada)**

Frequência (*Saida* para *null*): 2

Probabilidade (*Saida* para *null*): 0.5

***Entrada* -> *Saida* (Tipo de mensagem HL7: ADT_A03 - *Discharge/End Visit*)**

Frequência (*Entrada* para *Saida*): 22

Probabilidade (*Entrada* para *Saida*): 1.0

Neste exemplo, simularemos que a afluência de entrada é maior que a afluência de saída, algo que em situações reais poderia acontecer, por exemplo, no caso de acidentes ou até mesmo uma pandemia, o que envolve muitas pessoas. Estes acontecimentos consequentemente aumentam a afluência aos hospitais. Assim, o estado de entrada está estipulado para uma capacidade máxima de 50 pessoas e o estado de saída de 3 pessoas. Caso o limite seja atingindo, as pessoas não conseguem transitar para o estado seguinte (Figura 46).

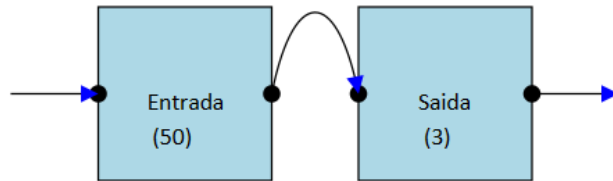


Figura 46 - Estado da simulação de 2 estados.

Com a interface gráfica apenas é possível visualizar o número de pessoas em cada estado a determinado momento. Para uma melhor compreensão do comportamento do simulador, é gerado um ficheiro de texto que complementa a análise da simulação.

Aquando do início da simulação é gerada a primeira pessoa que inicia a simulação, sendo que o evento associado à Entrada é do tipo ADT_A01:

```
Thread#25 - 2024-06-20 11:09:10 - Pedro Gomes (ID:962027) entered the simulation
Thread#25 - 2024-06-20 11:09:10 - Event: ADT_A01 - Admit/visit Notification
Thread#25 - 2024-06-20 11:09:10 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240620110
910||ADT^A01|0a28bb0b-55a9-4179-984c-f163e70a4b58|P|2.5.1
PID|962027|||Gomes^Pedro||19880906|M|||Estrada nacional
4^^Amadora^^^Portugal||916269945
PV1|962027|||ICU|EMER|20240620110910
EVN||20240620110910
```

Desde a entrada da primeira pessoa até ao final do primeiro minuto, entraram 15 pessoas, sendo possível visualizar o estado do sistema nesse momento:

```
Thread#49 - 2024-06-20 11:09:10 - Current state of the simulation:
Entrada: Pedro Gomes (ID:962027), Rita Costa (ID:558428)
Saída:
(...)
```

Thread#50 - 2024-06-20 11:10:01 - Current state of the simulation:

Entrada: João Nunes (ID:222051), Patrícia Rodrigues (ID:334566), Bárbara Rodrigues (ID:457086), Rui Silva (ID:248380), José Rodrigues (ID:792156), Rui Costa (ID:218658), João Rodrigues (ID:563351), Patrícia Santos (ID:144841), Guilherme Gomes (ID:603682), Sandra Fernandes (ID:488841), Sofia Silva (ID:491720), Bárbara Oliveira (ID:968790), José Morais (ID:329850)

Saida: Miguel Oliveira (ID:705724), José Maia (ID:258597), Miguel Maia (ID:209605)

Thread#49 - 2024-06-20 11:10:00 - Updated person count: 15

Considerando que o segundo estado tem uma capacidade baixa, ocorrem situações em que, apesar da transição poder acontecer devido à frequência e probabilidade definidas, como a capacidade máxima é atingida, a pessoa não pode transitar para o estado seguinte:

Thread#50 - 2024-06-20 11:09:54 - Max capacity reached for state: Saida

No entanto, à medida que pessoas transitavam para o estado *Saida* e eventualmente eram retiradas da simulação, eventualmente os pacientes presentes no estado *Entrada* poderiam transitar para o estado *Saida*. A ordem de transição foi estipulada por ordem de chegada, usando o *First-In, First- Out* (FIFO).

Analisando o caso de uma das pessoas da simulação, Rui Fernandes com ID 477253. O Rui entrou na simulação no dia 20-06-2024 às 11:10:03, gerando o seguinte *output*:

Thread#49 - 2024-06-20 11:10:03 - Rui Fernandes (ID:477253) entered the simulation

Thread#49 - 2024-06-20 11:10:03 - HL7 Event: ADT_A01 - Admit/visit Notification

Thread#49 - 2024-06-20 11:10:03 -

MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240620111003||ADT^A01|102ed377-6f10-4a4c-9005-6495fb53a155|P|2.5.1

PID||477253||Fernandes^Rui||19850317|M|||Travessa da alegria^^Funchal^^^Portugal||966123707

PV1|477253||ICU|EMER|||||||||||||||||||||||||||||||||||||20240620111003

EVN||20240620111003

Às 11:14:40 transitou para o estado *Saida*, sendo posteriormente atingido a capacidade máxima do estado:

```
Thread#50 - 2024-06-20 11:14:40 - Rui Fernandes (ID:477253) moved from Entrada to Saida
Thread#50 - 2024-06-20 11:14:40 - HL7 Event: ADT_A03 - Discharge/End Visit
Thread#50 - 2024-06-20 11:14:40 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240620111
440||ADT^A03|c9e88b67-86cf-435f-a14f-f1fa1be99502|P|2.5.1
PID||477253||Fernandes^Rui||19850317|M|||Travessa da
alegria^^Funchal^^^Portugal||966123707
PV1|||Ward1|E|#####|20240620111440
EVN|A03|20240620111440
Thread#50 - 2024-06-20 11:14:40 - Max capacity reached for state: Saida
```

Posteriormente, às 11:25:58, foi removido da simulação. Como esta transição foi configurada para não gerar uma mensagem de HL7, apenas é apresentada a mensagem de que não existe uma mensagem associada à transição:

```
Thread#50 - 2024-06-20 11:25:58 -- Rui Fernandes (ID:477253) left the simulation
Thread#50 - 2024-06-20 11:25:58 - HL7 Event: No event
Thread#50 - 2024-06-20 11:25:58 - No HL7 message associated with this event
```

A simulação termina assim que todos as pessoas são removidas da simulação. Neste caso, o utilizador decidiu parar a simulação mesmo não tendo sido removidas todas as pessoas da simulação, desta forma, é também apresentado o aviso no ficheiro da ação efetuada pelo utilizador:

```
Thread#25 - 2024-06-20 11:40:36 - Simulation was stopped by the user.
```

É considerado de seguida o caso da mesma simulação ser efetuada ao dobro da velocidade. Seguindo a lógica anterior, veremos o percurso de uma das pessoas envolvidas na simulação, o caso da Beatriz Maia com ID 526959.

A Beatriz entrou na simulação para o estado *Entrada* no dia 20-06-2024 às 11:50:19, tendo sido gerada uma mensagem HL7 ADT_A01 relacionada com a admissão de pacientes:

```
Thread#53 - 2024-06-20 11:50:19 - Beatriz Maia (ID:526959) entered the simulation
Thread#53 - 2024-06-20 11:50:19 - HL7 Event: ADT_A01 - Admit/visit Notification
Thread#53 - 2024-06-20 11:50:19 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240620115
019||ADT^A01|22f23f60-9c19-4433-be4a-a6032923ee7e|P|2.5.1
PID||526959|||Maia^Beatriz||19520728|F|||Alameda dos
condes^^Porto^^^Portugal||967888732
PV1||526959|||ICU|EMER|||||||||||||||||||||||||||||||||||||20240620115019
EVN||20240620115019
```

Posteriormente, foi movida para o estado seguinte, *Saida*, às 11:51:09, tendo sido gerada uma mensagem HL7 do tipo ADT_A03, relacionada com a alta de um paciente:

```
Thread#53 - 2024-06-20 11:51:09 - Beatriz Maia (ID:526959) moved from Entrada to Saida
Thread#53 - 2024-06-20 11:51:09 - HL7 Event: ADT_A03 - Discharge/End Visit
Thread#53 - 2024-06-20 11:51:09 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240620115
109||ADT^A03|0fbb6a5e-6448-4276-b707-f966592be84b|P|2.5.1
PID||526959|||Maia^Beatriz||19520728|F|||Alameda dos
condes^^Porto^^^Portugal||967888732
PV1|||Ward1|E|||||||||||||||||||||||||||||||||||||20240620115109
EVN|A03|20240620115109
```

Finalmente, foi removida da simulação às 11:51:43, não gerando qualquer tipo de mensagem HL7:

```
Thread#53 - 2024-06-20 11:51:43 - Beatriz Maia (ID:526959) left the simulation
Thread#53 - 2024-06-20 11:51:43 - HL7 Event: No event
Thread#53 - 2024-06-20 11:51:43 -- No HL7 message associated with this event
```

Pela análise do primeiro caso com dois estados, *Entrada* e *Saida*, é notável a complexidade da gestão do fluxo de pessoas no meio hospitalar. Na simulação em tempo-real, foi observado que a entrada de pessoas é bastante mais elevada do que a saída, situação que se assemelha a eventos de elevada afluência, como acidentes, pandemias, entre outros.

A capacidade máxima do primeiro estado foi fixada para 50 pessoas enquanto a do segundo estado foi apenas 3 pessoas, o que causou um impedimento constante das pessoas transitarem para o estado final quando essa capacidade máxima era atingida.

Foram feitas duas simulações uma sendo em tempo-real e a segunda com o dobro da velocidade, garantindo sempre a transição das pessoas pela lógica FIFO, de forma a ser mais rápido para o utilizador gerar múltiplas simulações.

Estes resultados sublinham a importância da gestão eficaz da capacidade dos estados em sistemas hospitalares, especialmente em situações de grande afluência, para evitar sobrecargas e garantir um atendimento contínuo e eficiente.

4.2.2 Caso 2

Analisando um caso em que a simulação contém três estados, *Entrada*, *Consulta* e *Saida*, efetuada em tempo-real (Figura 47).

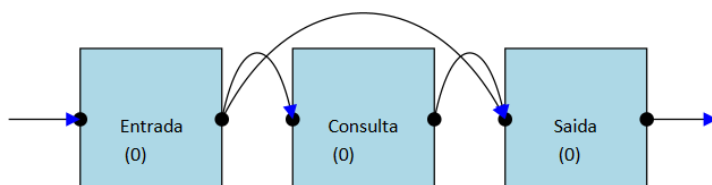


Figura 47 - Simulação exemplo com 3 estados.

As transições definidas para este caso são os seguintes:

Capacidade Máxima do estado *Entrada*: 10

Capacidade Máxima do estado *Consulta*: 1

Capacidade Máxima do estado *Saida*: 30

***null* -> *Entrada* (Tipo de mensagem HL7: ADT_A01 - *Admit/visit Notification*)**

Frequência (*null* para *Entrada*): 2

Probabilidade (*null* para *Entrada*): 0.5

Saida -> null (Tipo de mensagem HL7: Sem mensagem HL7 associada)

Frequência (Saida para null): 1

Probabilidade (Saida para null): 0.3

Entrada -> Consulta (Tipo de mensagem HL7: ORM_O01 - General Order)

Frequência (Entrada para Consulta): 1

Probabilidade (Entrada para Consulta): 0.2

Entrada -> Saida (Tipo de mensagem HL7: ADT_A03 - Discharge/End Visit)

Frequência (Entrada para Saida): 1

Probabilidade (Entrada para Saida): 0.5

Consulta -> Saida (Tipo de mensagem HL7: ADT_A03 - Discharge/End Visit)

Frequência (Consulta para Saida): 2

Probabilidade (Consulta para Saida): 0.6

Neste caso pretende-se simular uma situação de afluência normal de um provedor de saúde, sendo que o movimento entre estados das pessoas decorre a um ritmo natural do dia a dia (Figura 48).

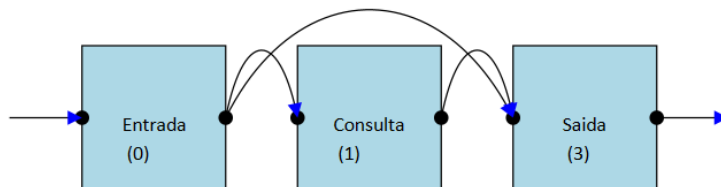


Figura 48 - Estado da Simulação de 3 estados.

Considerando o caso de uma das pessoas presentes nesta simulação, Ana Santos (ID:309284). Esta entrou na simulação a 22-06-2023 às 12:10:24:

Thread#47 - 2024-06-22 12:10:24 - Ana Santos (ID:309284) entered the simulation

Thread#47 - 2024-06-22 12:10:24 - HL7 Event: ADT_A01 - Admit/visit Notification

Thread#47 - 2024-06-22 12:10:24 -

MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622121024||ADT^A01|57029a9c-1d67-42f1-971c-50157acad721|P|2.5.1

PID|309284||Santos^Ana||19740506|F||Praça da

répulica^^Setúbal^^Portugal||933644653

PV1|309284||ICU|EMER|||||||||||||||||||||||||||||||||||||20240622121024

EVN||20240622121024

Thread#47 - 2024-06-22 12:25:00 - Ana Costa (ID:405649) moved from Consulta to Saida
Thread#47 - 2024-06-22 12:25:00 - HL7 Event: ADT_A03 - Discharge/End Visit
Thread#47 - 2024-06-22 12:25:00 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622125
000||ADT^A03|704f066e-dba8-4b72-8865-1e0bdc08c6ee|P|2.5.1
PID||405649||Costa^Ana||19840705|F||Avenida da
liberdade^^Funchal^^^Portugal||932267232
PV1|||Ward1|E|||||||||||||||||||||||||||||||||||||20240622125000
EVN|A03|20240622125000

Finalmente, é removida da simulação às 12:26:01, tendo a simulação terminado às 12:31:11 por todas as pessoas terem sido removidas da simulação:

Thread#47 - 2024-06-22 12:26:01 - All people have reached the end state.
Thread#47 - 2024-06-22 12:26:01 - Ana Costa (ID:405649) left the simulation
Thread#47 - 2024-06-22 12:26:01 - HL7 Event: No event
Thread#47 - 2024-06-22 12:26:01 - No HL7 message associated with this event

(...)

Thread#47 - 2024-06-22 12:31:11 - All people have been removed from the simulation.

De igual forma, quando a simulação é feita ao dobro da velocidade, a situação mantém-se, mas a um ritmo mais acelerado.

Procedendo à análise do caso de José Fernandes (ID:794505). Entrou na simulação às 12:33:07 no mesmo dia:

Thread#25 - 2024-06-22 12:33:07 -- José Fernandes (ID:794505) entered the simulation
Thread#25 - 2024-06-22 12:33:07 - Event: ADT_A01 - Admit/visit Notification
Thread#25 - 2024-06-22 12:33:07 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622123
307||ADT^A01|b26b69a7-c249-4b96-a104-d0acc930fcd|P|2.5.1
PID||794505||Fernandes^José||19680110|M||Rua da
luz^^Queluz^^^Portugal||918314803
PV1|794505||ICU|EMER|||||||||||||||||||||||||||||||||||||20240622123307
EVN||20240622123307

De seguida foi movido para o estado *Consulta* às 12:33:52:

Thread#52 - 2024-06-22 12:33:52 - José Fernandes (ID:794505) moved from Entrada to Consulta

Thread#52 - 2024-06-22 12:33:52 - HL7 Event: ORM_001 - General Order

Thread#52 - 2024-06-22 12:33:52 -

MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622123352||ORM^O01|32f3b745-e0f5-4afa-bf4f-825c57e4bbdb|P|2.5.1

PID||794505|||Fernandes^José||19680110|M|||Rua da

luz^^Queluz^^^Portugal||918314803

PV1|||

ORC|SC|12345|54321

OBR|||Test^Test Description

Finalmente, terminou o seu percurso no estado final, *Saida*:

Thread#52 - 2024-06-22 12:34:50 - José Fernandes (ID:794505) moved from Consulta to Saida

Thread#52 - 2024-06-22 12:34:50 - HL7 Event: ADT_A03 - Discharge/End Visit

Thread#52 - 2024-06-22 12:34:50 -

MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622123450||ADT^A03|8c02e640-2233-4579-8075-4e98736dfcd8|P|2.5.1

PID||794505|||Fernandes^José||19680110|M|||Rua da

luz^^Queluz^^^Portugal||918314803

PV1|||Ward1|E|||||||||||||||||||||||||||||||||||||20240622123450

EVN|A03|20240622123450

A análise da simulação com três estados, *Entrada*, *Consulta* e *Saida*, simula o funcionamento do sistema em um cenário de afluência normal num prestador de saúde. Assemelhando-se ao caso anterior, a simulação foi efetuada em tempo-real e no dobro da velocidade, sendo as pessoas geridas de acordo com a lógica FIFO.

A simulação demonstrou a importância de gerir efetivamente a capacidade e o fluxo entre os estados para garantir que o sistema funcione sem sobrecargas. As diferentes trajetórias dos pacientes na simulação demonstram como o sistema pode gerar fluxos variáveis, desde movimentos diretos entre o primeiro estado e o último ou até transições mais complexas envolvendo estados intermédios.

4.2.3 Caso 3

Analisando um caso mais complexo com cinco estados: *Entrada*, *Sala de espera*, *Consulta*, *Exame* e *Saida* (Figura 49).

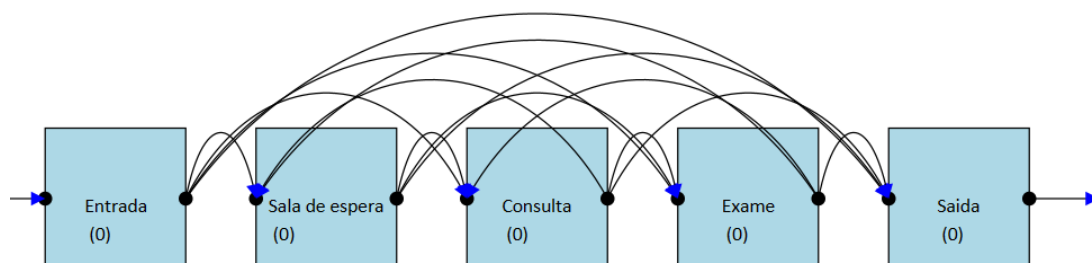


Figura 49 - Simulação exemplo com 5 estados.

As transições definidas para este caso são as seguintes:

Capacidade Máxima do estado *Entrada*: 10

Capacidade Máxima do estado *Sala de espera*: 10

Capacidade Máxima do estado *Consulta*: 1

Capacidade Máxima do estado *Exame*: 1

Capacidade Máxima do estado *Saida*: 33

***null* -> *Entrada* (Tipo de mensagem HL7: ADT_A01 - *Admit/visit Notification*)**

Frequência (*null* para *Entrada*): 1

Probabilidade (*null* para *Entrada*): 0.3

***Saida* -> *null* (Tipo de mensagem HL7: Sem mensagem HL7 associada)**

Frequência (*Saida* para *null*): 4

Probabilidade (*Saida* para *null*): 0.7

***Entrada* -> *Sala de espera* (Tipo de mensagem HL7: Sem mensagem HL7 associada)**

Frequência (*Entrada* para *Sala de espera*): 1

Probabilidade (*Entrada* para *Sala de espera*): 0.3

***Entrada* -> *Saida* (Tipo de mensagem HL7: ADT_A03 - *Discharge/End Visit*)**

Frequência (*Entrada* para *Saida*): 3

Probabilidade (*Entrada* para *Saida*): 0.5

***Entrada* -> *Consulta* (Tipo de mensagem HL7: ORM_O01 - *General Order*)**

Frequência (*Entrada* para *Consulta*): 8

Probabilidade (*Entrada* para *Consulta*): 0.7

Entrada -> Exame (Tipo de mensagem HL7: ORU_R01 - Unsolicited transmission of an observation message)

Frequência (*Entrada para Exame*): 1

Probabilidade (*Entrada para Exame*): 0.1

Sala de espera -> Consulta (Tipo de mensagem HL7: ORM_O01 - General Order)

Frequência (*Sala de espera para Consulta*): 2

Probabilidade (*Sala de espera para Consulta*): 0.3

Sala de espera -> Exame (Tipo de mensagem HL7: ORU_R01 - Unsolicited transmission of an observation message)

Frequência (*Sala de espera para Exame*): 1

Probabilidade (*Sala de espera para Exame*): 0.8

Sala de espera -> Saida (Tipo de mensagem HL7: ADT_A03 - Discharge/End Visit)

Frequência (*Sala de espera para Saida*): 2

Probabilidade (*Sala de espera para Saida*): 0.3

Consulta -> Sala de espera (Tipo de mensagem HL7: Sem mensagem HL7 associada)

Frequência (*Consulta para Sala de espera*): 1

Probabilidade (*Consulta para Sala de espera*): 1.0

Consulta -> Exame (Tipo de mensagem HL7: ORU_R01 - Unsolicited transmission of an observation message)

Frequência (*Consulta para Exame*): 2

Probabilidade (*Consulta para Exame*): 0.5

Consulta -> Saida (Tipo de mensagem HL7: ADT_A03 - Discharge/End Visit)

Frequência (*Consulta para Saida*): 2

Probabilidade (*Consulta para Saida*): 0.2

Exame -> Saida (Tipo de mensagem HL7: ADT_A03 - Discharge/End Visit)

Frequência (*Exame para Saida*): 2

Probabilidade (*Exame para Saida*): 1.0

Exame -> Consulta (Tipo de mensagem HL7: ORM_O01 - General Order)

Frequência (*Exame para Consulta*): 1

Probabilidade (*Exame para Consulta*): 0.4

Exame -> Sala de espera (Tipo de mensagem HL7: Sem mensagem HL7 associada)

Frequência (*Exame para Sala de espera*): 5

Probabilidade (*Exame para Sala de espera*): 0.7

Neste caso, a simulação acrescenta à anterior os estados *Sala de espera* e *Exame* para simular um *workflow* hospitalar mais completo, mantendo as condições a simular (Figura 50).

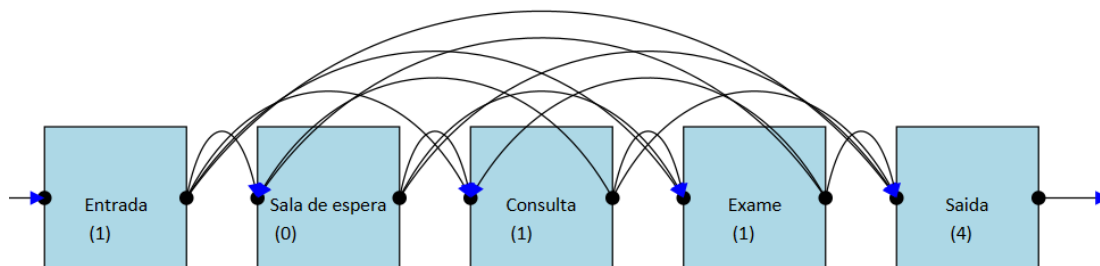


Figura 50 - Estado da simulação de 5 estados.

Considerando o caso de Pedro Fernandes com ID 249215. O paciente entrou na simulação para o estado *Entrada* no dia 21-06-2024 às 12:55:11, sendo movido para a *Consulta* poucos minutos depois:

```
Thread#25 - 2024-06-21 12:55:11 -- Pedro Fernandes (ID:249215) entered the simulation
Thread#25 - 2024-06-21 12:55:11 - Event: ADT_A01 - Admit/visit Notification
Thread#25 - 2024-06-21 12:55:11 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240621125
511||ADT^A01|1ae809a3-3da0-42c7-b0f5-5294d04a9bee|P|2.5.1
PID||249215|||Fernandes^Pedro||19650113|M|||Alameda dos
condes^^Almada^^^Portugal||962150360
PV1|249215|||ICU|EMER|||||||||||||||||||||||||||||||||||||20240621125511
EVN||20240621125511
```

(...)

```
Thread#61 - 2024-06-21 12:57:46 - Pedro Fernandes (ID:249215) moved from Entrada to
Consulta
Thread#61 - 2024-06-21 12:57:46 - HL7 Event: ORM_O01 - General Order
Thread#61 - 2024-06-21 12:57:46 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240621125
746||ORM^O01|8e1e13d1-a032-442d-9e92-eede6c26d614|P|2.5.1
PID||249215|||Fernandes^Pedro||19650113|M|||Alameda dos
condes^^Almada^^^Portugal||962150360
PV1|||
ORC|SC|12345|54321
OBR|||Test^Test Description
```

De seguida, entrou no estado *Sala de espera* às 13:00:54 no mesmo dia:

Thread#61 - 2024-06-21 13:00:54 - Pedro Fernandes (ID:249215) moved from Consulta to Sala de espera

Thread#61 - 2024-06-21 13:00:54 - HL7 Event: No event

Thread#61 - 2024-06-21 13:00:54 - No HL7 message associated with this event

Após sensivelmente 4 minutos, o paciente iniciou o *Exame*:

Thread#62 - 2024-06-21 13:04:36 - Pedro Fernandes (ID:249215) moved from Sala de espera to Exame

Thread#62 - 2024-06-21 13:04:36 - HL7 Event: ORU_R01 - Unsolicited transmission of an observation message

Thread#62 - 2024-06-21 13:04:36 -

*MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622130436||ORU^R01|0c2aa6af-3c9c-4844-84a0-f6d09397045c|P|2.5.1
PID||249215|||Fernandes^Pedro||19650113|M|||Alameda dos
condes^^Almada^^Portugal||962150360*

ORC|RE

OBR|||Test^Test Description

OBX|1|TX|12345^Observation||Observation Value for testing

Finalmente, foi dirigido para o estado *Saida* às 13:11:20 tendo sido removido da simulação momentos depois:

Thread#62 - 2024-06-21 13:11:20 - Pedro Fernandes (ID:249215) moved from Exame to Saida

Thread#62 - 2024-06-21 13:11:20 - HL7 Event: ADT_A03 - Discharge/End Visit

Thread#62 - 2024-06-21 13:11:20 -

*MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240622131120||ADT^A03|4e9d59c7-4f3a-4826-b736-4bb5241731f0|P|2.5.1
PID||249215|||Fernandes^Pedro||19650113|M|||Alameda dos
condes^^Almada^^Portugal||962150360*

PV1|||Ward1|E|||||||||||||||||||||||||||||||||||||20240622131120

EVN|A03|20240622131120

De igual forma, o utilizador tem a possibilidade de correr a simulação com o dobro da velocidade. Considerando o caso de Miguel Rodrigues (ID: 021821), este entrou na simulação para o estado inicial *Entrada* no dia 21-06-2024 às 13:16:04:

Thread#73 - 2024-06-21 13:16:04 - Miguel Pereira (ID:777921) entered the simulation
Thread#73 - 2024-06-21 13:16:04 - HL7 Event: ADT_A01 - Admit/visit Notification
Thread#73 - 2024-06-21 13:16:04 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240621131604|ADT^A01|890b6c38-2f94-41a6-83fc-bbe429b63941|P|2.5.1
PID||777921||Pereira^Miguel||19980313|M||Largo da
perdição^^Funchal^^Portugal||934221566
PV1|777921||ICU|EMER|||||||||||||||||||||||||||||||||||||20240621131604
EVN||20240621131604

Às 13:17:21, foi movido para o estado *Consulta*, tendo sido posteriormente movido para a *Sala de espera* momentos depois:

Thread#73 - 2024-06-21 13:17:21 - Miguel Pereira (ID:777921) moved from Entrada to Consulta
Thread#73 - 2024-06-21 13:17:21 - HL7 Event: ORM_O01 - General Order
Thread#73 - 2024-06-21 13:17:21 -
MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240621131721|ORM^O01|ce6e9f62-06f7-4c20-929e-7df8dfab200b|P|2.5.1
PID||021821||Rodrigues^Miguel||19600901|M||Largo da perdição^^Vila Nova de Gaia^^Portugal||933221647
PV1|||
ORC|SC|12345|54321
OBR|||Test^Test Description

(...)

Thread#73 - 2024-06-21 13:18:55 - Moved from Consulta to Sala de espera
Thread#73 - 2024-06-21 13:18:55 - HL7 Event: No event
Thread#73 - 2024-06-21 13:18:55 - No HL7 message associated with this event

Seguidamente, deu-se o início do *Exame* às 13:20:46 com término na *Saida* às 13:21:46:

Thread#74 - 2024-06-21 13:20:46 - Moved from Sala de espera to Exame

Thread#74 - 2024-06-21 13:20:46 - HL7 Event: ORU_R01 - Unsolicited transmission of an observation message

Thread#74 - 2024-06-21 13:20:46 -

MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240621132046||ORU^R01|0cd6e3ad-43f4-48f9-aa6c-71864c46f8f0|P|2.5.1

PID||021821||Rodrigues^Miguel||19600901|M||Largo da perdição^^Vila Nova de Gaia^^^Portugal||933221647

ORC|RE

OBR|||Test^Test Description

OBX|1|TX|12345^Observation||Observation Value for testing

(...)

Thread#74 - 2024-06-21 13:21:46 - Moved from Exame to Saida

Thread#74 - 2024-06-21 13:21:46 - HL7 Event: ADT_A03 - Discharge/End Visit

Thread#74 - 2024-06-21 13:21:46 -

MSH|^~\&|SendingSystem|SendingFacility|ReceivingSystem|ReceivingFacility|20240621132146||ADT^A03|98476116-8ee1-45c4-873f-b20bfe29f3cf|P|2.5.1

PID||021821||Rodrigues^Miguel||19600901|M||Largo da perdição^^Vila Nova de Gaia^^^Portugal||933221647

PV1|||Ward1|E|20240621132146

EVN|A03|20240621132146

Finalmente, é feita a remoção do paciente da simulação:

Thread#74 - 2024-06-21 13:21:50 - Miguel Rodrigues (ID:021821) left the simulation

Thread#74 - 2024-06-21 13:21:50 - HL7 Event: No event

Thread#74 - 2024-06-21 13:21:50 - No HL7 message associated with this event

A simulação com cinco estados, *Entrada*, *Sala de espera*, *Consulta*, *Exame* e *Saida*, oferece uma visão detalhada do *workflow* hospitalar, proporcionando uma análise mais complexa do fluxo de pacientes. As transições entre estados foram definidas com frequências e probabilidades específicas para refletir o movimento realístico dos pacientes num prestador de saúde.

A análise revela que, mesmo num cenário complexo, o simulador conseguiu lidar eficazmente com o fluxo de pacientes, garantindo que cada etapa do processo fosse documentada com precisão.

4.3 Conclusão do Capítulo 4

O programa desenvolvido permite ao utilizador definir cenários de simulação configurando estados, transições e eventos, assim como a frequência com que as transições ocorrem e probabilidade de as mesmas ocorrerem de forma a tornar o simulador mais realista.

Ao iniciar o programa, são listados os ficheiros de configuração JSON existentes ou uma mensagem, se não se encontrar alguma configuração prévia. O menu principal apresenta diversas opções como criar, modificar, listar configurações bem como iniciar a simulação e sair da mesma.

Para simular, é necessário definir um nome para a simulação, estados e capacidades máximas correspondentes. Devem ser definidos o estado inicial e final, transições com frequência e probabilidade, e eventos associados. O utilizador pode visualizar os detalhes das configurações, que são salvas num ficheiro de configuração JSON com um nome único.

Para modificar uma configuração, o utilizador seleciona a configuração a alterar, adicionando novos estados e transições ou alterando os já existentes. No final são apresentados os novos detalhes juntamente com o digrama da máquina de estados. De igual forma, estes podem ser apresentados na terceira opção do menu – Listagem de configurações existentes.

Para iniciar a simulação, é necessário escolher uma configuração e definir a sua velocidade. A simulação apresenta o diagrama da máquina de estados e, no seu decorrer, o número de pacientes por estado.

Após a conclusão, as transições da simulação são guardadas num ficheiro de texto que detalha os eventos HL7 e os movimentos na simulação, ajudando na análise dos resultados.

Foram estudados três casos de aplicação com diversas dimensões, demonstrando a flexibilidade e detalhe do *software* na simulação de integrações de HL7 em prestadores de saúde.

5 Conclusão

Neste capítulo são abordadas as conclusões finais do projeto, pensando também nas limitações sentidas no seu desenvolvimento, assim como perspectivas futuras de melhoria do *software* criado.

5.1 Conclusão Finais

Com o desenvolvimento deste projeto, centrado na criação de um simulador de integrações HL7, é mais uma vez demonstrada a relevância da simulação em sistemas de saúde, especialmente nos sistemas de informação. Assim, este projeto teve como objetivo o desenvolvimento de uma ferramenta que permite testar as integrações de HL7 em prestadores de saúde. Considerando que estes sistemas são críticos, esta ferramenta auxilia o teste das integrações sem comprometer os ambientes de produção reais.

A implementação do simulador, suportada por tecnologias como Java, JavaFX, CSS, e a biblioteca HAPI, revelou-se eficaz na simulação de um ambiente realista de vários cenários hospitalares. A escolha destas tecnologias, fundamentada na robustez e documentação das mesmas, permitiu desenvolver um sistema intuitivo, flexível e escalável, capaz de atender às necessidades dos prestadores de cuidados de saúde. Ao promover um ambiente seguro para a realização de simulações, reduz-se o impacto negativo nos sistemas de produção, permitindo aos prestadores de cuidados de saúde o teste e validação das suas integrações de forma mais eficiente e segura.

As configurações do simulador, que incluem a definição de estados, transições e eventos, bem como a sua capacidade de definir as frequências com que as transições são realizadas e probabilidade de estas ocorrerem, destacam-se pela sua capacidade de modelar cenários reais

sem qualquer risco. A interface gráfica desenvolvida facilita a interação do utilizador com o sistema, permitindo a criação e modificação de configurações de forma intuitiva.

Os casos de estudo apresentados demonstram o funcionamento do simulador em diferentes cenários, desde os mais simples aos mais complexos, validando a sua capacidade de gerar resultados coerentes e realistas nos diversos cenários. A análise das transições geradas pelas simulações, contendo mensagens HL7 e movimentos de pacientes, proporciona uma visão detalhada dos eventos simulados, auxiliando na avaliação e melhoria dos sistemas de saúde.

Considera-se que os principais objetivos do projeto foram atingidos sendo que foi desenvolvido um simulador de integrações HL7 que se apresenta como uma ferramenta útil para a área da saúde, promovendo a melhoria contínua dos sistemas de informação em saúde, que de igual forma, contribui para a melhoria da qualidade dos cuidados prestados e da segurança dos pacientes.

5.2 Limitações e Investigação Futura

Apesar de atingidos os objetivos mencionados, foram identificadas algumas limitações ao longo do seu desenvolvimento. O âmbito do mestrado em engenharia biomédica não é unicamente centrado em informática, o que resultou em algumas dificuldades iniciais na definição de critérios importantes para a programação em Java. Esta barreira inicial exigiu um esforço adicional para adquirir conhecimentos na linguagem e aprendizagem das técnicas necessárias para o desenvolvimento do simulador. De igual forma, a literatura existente sobre as simulações de integrações HL7 é escassa, o que limitou a possibilidade de comparação e validação com outros projetos semelhantes.

Numa perspetiva futura, existem vários aspetos a melhorar e a complementar neste projeto. Em primeiro lugar, a melhoria das informações disponibilizadas nas mensagens HL7, incluindo a introdução de mais tipos de mensagens, pode aumentar a precisão e a abrangência das simulações. De igual forma, o envolvimento de médicos, técnicos e outros profissionais de saúde no simulador, não se restringindo apenas a pacientes, poderia tornar a simulação mais realista e representativa da realidade hospitalar.

Outra área a melhorar seria a forma como o utilizador interage com o tempo da simulação. Poder-se-ia aplicar uma maior flexibilidade para o utilizador controlar a velocidade de acordo com o estudo a ser feito, o que resultaria em melhores resultados. A interface gráfica também poderia ser aprimorada, proporcionando mais informações ao utilizador em tempo-real sem este necessitar de analisar o ficheiro de texto todas as vezes que é feita uma simulação, o que tornaria este tipo de transações apenas um complemento e não uma necessidade.

Feitas estas melhorias, o código do simulador poderia ser disponibilizado para a comunidade científica, através da disponibilização do simulador em formato *open-source*, o que incentivaria contribuições de desenvolvedores externos, promovendo um ciclo contínuo de melhorias e inovação nos sistemas de informação.

Referências Bibliográficas

- [1] H. Singh, M. Kalani, S. Acosta-Torres, T. Y. El Ahmadieh, J. Loya, and A. Ganju, "History of simulation in medicine: from Resusci Annie to the Ann Myers Medical Center," *Neurosurgery*, vol. 73 Suppl 1, no. SUPPL. 4, Oct. 2013, doi: 10.1227/NEU.0000000000000093.
- [2] B. Blazona and M. Koncar, "HL7 and DICOM based integration of radiology departments with healthcare enterprise information systems," *Int J Med Inform*, vol. 76, no. SUPPL. 3, pp. S425–S432, Dec. 2007, doi: 10.1016/J.IJMEDINF.2007.05.001.
- [3] "Introduction to HL7 Standards | HL7 International." Accessed: May 18, 2024. [Online]. Available: <http://www.hl7.org/implement/standards/index.cfm?ref=nav>
- [4] "Sectra's history: 40 years of knowledge and passion | Investor Relations at Sectra." Accessed: Nov. 12, 2023. [Online]. Available: <https://investor.sectra.com/this-is-sectra/sectras-history/>
- [5] "Sectra får beställning som är värd 100 miljoner kronor." Accessed: Feb. 10, 2024. [Online]. Available: https://corren.se/bli-prenumerant/artikel/jp3648ql/oc-1m1kr_s_23
- [6] "Sectra in brief | Investor Relations at Sectra." Accessed: Nov. 12, 2023. [Online]. Available: <https://investor.sectra.com/this-is-sectra/sectra-in-brief/>
- [7] F. Lateef, "Simulation-based learning: Just like the real thing," *J Emerg Trauma Shock*, vol. 3, no. 4, p. 348, Oct. 2010, doi: 10.4103/0974-2700.70743.
- [8] "simulador | Dicionário Infopédia da Língua Portuguesa." Accessed: Jul. 08, 2024. [Online]. Available: <https://www.infopedia.pt/dicionarios/lingua-portuguesa/simulador>
- [9] M. Aebersold, "The History of Simulation and Its Impact on the Future," *AACN Adv Crit Care*, vol. 27, no. 1, pp. 56–61, 2016, doi: 10.4037/AACNACC2016436.
- [10] "Link Trainer Flight Simulator - Naval Air Station Fort Lauderdale Museum." Accessed: Feb. 10, 2024. [Online]. Available: <https://www.nasflmuseum.com/link-trainer.html>
- [11] P. Tong and C. Bil, "Effects of errors on transfer of aircraft flight training," 2007, Accessed: Feb. 10, 2024. [Online]. Available:

https://www.researchgate.net/publication/288205920_Effects_of_errors_on_transfer_of_aircraft_flight_training

- [12] “Kriegsspiel – Kriegsspiel, the 19th Century military training system and other map related, umpire moderated wargames.” Accessed: Feb. 10, 2024. [Online]. Available: <https://kriegsspielorg.wordpress.com/>
- [13] F. Laamarti, M. Eid, and A. El Saddik, “An overview of serious games,” *International Journal of Computer Games Technology*, vol. 2014, 2014, doi: 10.1155/2014/358152.
- [14] C. V. de Carvalho, C. V. de Carvalho, and A. Coelho, “Game-Based Learning, Gamification in Education and Serious Games,” *Game-Based Learning, Gamification in Education and Serious Games*, p. 182, Apr. 2022, doi: 10.3390/BOOKS978-3-0365-3809-9.
- [15] “Manual de Uso do Swift Playgrounds para iPadOS - Suporte da Apple (BR).” Accessed: Jul. 01, 2024. [Online]. Available: <https://support.apple.com/pt-br/guide/playgrounds-ipad/welcome/ipados>
- [16] H. Singh, M. Kalani, S. Acosta-Torres, T. Y. El Ahmadieh, J. Loya, and A. Ganju, “History of simulation in medicine: from Resusci Annie to the Ann Myers Medical Center,” *Neurosurgery*, vol. 73 Suppl 1, no. SUPPL. 4, Oct. 2013, doi: 10.1227/NEU.0000000000000093.
- [17] F. Lateef, “Simulation-based learning: Just like the real thing,” *J Emerg Trauma Shock*, vol. 3, no. 4, p. 348, Oct. 2010, doi: 10.4103/0974-2700.70743.
- [18] “How Discrete-Event Simulation Helps Hospitals - BigBear.ai.” Accessed: Jun. 19, 2024. [Online]. Available: <https://bigbear.ai/blog/what-is-discrete-event-simulation-and-how-does-it-help-in-hospital-process-optimization-and-capacity-planning/>
- [19] “User Manual.” Accessed: Jul. 01, 2024. [Online]. Available: <https://gazelle.ihe.net/gazelle-documentation/Order-Manager/user.html>
- [20] “Understanding State Machines.” Accessed: Feb. 15, 2024. [Online]. Available: <https://www.freecodecamp.org/news/state-machines-basics-of-computer-science-d42855debc66/>
- [21] J. Yao, H. Yan, S. Das, J. F. Klemic, J. C. Ellenbogen, and C. M. Lieber, “Nanowire nanocomputer as a finite-state machine,” *Proc Natl Acad Sci U S A*, vol. 111, no. 7, pp. 2431–2435, Feb. 2014, doi:

10.1073/PNAS.1323818111/-
/DCSUPPLEMENTAL/PNAS.201323818SI.PDF.

- [22] "Finite-state machine." Accessed: Jun. 18, 2024. [Online]. Available: <https://www.cs.mtsu.edu/~xyang/3080/fsm.html>
- [23] M. Zhong, X. Tong, and X. Liu, "A new fast PWM modulation strategy for power electric converter based on finite set state machine theory," *Energy Reports*, vol. 9, pp. 90–96, Oct. 2023, doi: 10.1016/J.EGYR.2023.08.070.
- [24] "Interoperability in Healthcare | HIMSS." Accessed: Feb. 11, 2024. [Online]. Available: <https://www.himss.org/resources/interoperability-healthcare>
- [25] "HIMSS Proposes a New Interoperability Definition in Healthcare | Imprivata." Accessed: Feb. 18, 2024. [Online]. Available: <https://www.imprivata.com/uk/node/99796>
- [26] E. Li, J. Clarke, H. Ashrafian, A. Darzi, and A. L. Neves, "The Impact of Electronic Health Record Interoperability on Safety and Quality of Care in High-Income Countries: Systematic Review," *J Med Internet Res*, vol. 24, no. 9, Sep. 2022, doi: 10.2196/38144.
- [27] O. Pournik, T. Mukherjee, L. Ghalichi, and T. N. Arvanitis, "How Interoperability Challenges Are Addressed in Healthcare IoT Projects," *Stud Health Technol Inform*, vol. 309, pp. 121–125, Oct. 2023, doi: 10.3233/SHTI230754.
- [28] J. Doll, J. Malloy, and J. Bland, "The Promise of Interoperability," *Am J Occup Ther*, vol. 75, no. 4, Jun. 2021, doi: 10.5014/AJOT.2021.049002.
- [29] E. P. Ambinder, "Electronic Health Records," *J Oncol Pract*, vol. 1, no. 2, p. 57, Jul. 2005, doi: 10.1200/JOP.2005.1.2.57.
- [30] M. Larobina, "Thirty Years of the DICOM Standard," *Tomography*, vol. 9, no. 5, pp. 1829–1838, Oct. 2023, doi: 10.3390/TOMOGRAPHY9050145.
- [31] M. Mantri, S. Taran, and G. Sunder, "DICOM Integration Libraries for Medical Image Interoperability: A Technical Review," *IEEE Rev Biomed Eng*, vol. 15, pp. 247–259, 2022, doi: 10.1109/RBME.2020.3042642.
- [32] "X-ray film viewer wall series standards | CABLAS - Schermature Anti-x." Accessed: Feb. 13, 2024. [Online]. Available: <https://www.cablas.com/en/x-ray-film-viewer-wall-series-standards/>

- [33] “Integrating the Healthcare Enterprise (IHE) - IHE International.” Accessed: Feb. 18, 2024. [Online]. Available: <https://www.ihe.net/>
- [34] M. Eichelberg, E. Poiseau, B. B. Wein, and J. Riesmeier, “Extending the IHE initiative to Europe: experiences,” *Int Congr Ser*, vol. 1256, no. C, pp. 903–909, Jun. 2003, doi: 10.1016/S0531-5131(03)00234-6.
- [35] T. Al-Enazi and S. El-Masri, “HL7 engine module for healthcare information systems,” *J Med Syst*, vol. 37, no. 6, pp. 1–8, Dec. 2013, doi: 10.1007/S10916-013-9986-8/METRICS.
- [36] X. Wang, X. Tang, L. Yin, and W. Liu, “HL7 and the transmission of dynamic signal in HL7 standard,” *2013 ICME International Conference on Complex Medical Engineering, CME 2013*, pp. 124–127, 2013, doi: 10.1109/ICCME.2013.6548224.
- [37] “HL7 Resource: Understanding HL7 Standard Versions.” Accessed: Feb. 18, 2024. [Online]. Available: <https://www.interfaceware.com/hl7-standard-versions>
- [38] R. Noumeir, “Active Learning of the HL7 Medical Standard,” *J Digit Imaging*, vol. 32, no. 3, pp. 354–361, Jun. 2019, doi: 10.1007/S10278-018-0134-3/TABLES/1.
- [39] “HL7 Resource: Understanding the HL7 Message Structure.” Accessed: Feb. 18, 2024. [Online]. Available: <https://www.interfaceware.com/hl7-message-structure>
- [40] “HL7 v2.5.1 Chapter 2.” Accessed: Jul. 07, 2024. [Online]. Available: <https://www.hl7.eu/HL7v2x/v251/std251/ch02.html>
- [41] “HL7 V2.5.1 Chapter 3.” Accessed: Jun. 18, 2024. [Online]. Available: <https://www.hl7.eu/HL7v2x/v251/std251/ch03.html>
- [42] “HL7 V2.5.1 Chapter 4.” Accessed: Jun. 18, 2024. [Online]. Available: <https://www.hl7.eu/HL7v2x/v251/std251/ch04.html>
- [43] “NHapi.” Accessed: May 18, 2024. [Online]. Available: <https://nhapi.sourceforge.net/home.php>
- [44] “python-hl7 - Easy HL7 v2.x Parsing — python-hl7 0.4.3.dev documentation.” Accessed: May 18, 2024. [Online]. Available: <https://python-hl7.readthedocs.io/en/latest/>
- [45] “HAPI – The Open Source HL7 API for Java.” Accessed: Feb. 13, 2024. [Online]. Available: <https://hapifhir.github.io/hapi-hl7v2/index.html>
- [46] “JavaFX.” Accessed: Jun. 19, 2024. [Online]. Available: <https://openjfx.io/>

- [47] "CSS | MDN." Accessed: Jun. 19, 2024. [Online]. Available: <https://developer.mozilla.org/pt-BR/docs/Web/CSS>
- [48] X. Lu, Y. Gu, J. Zhao, N. Yu, and W. Jia, "Research and implementation of medical information format conversion based on HL7 Version 2.x," *2011 International Conference on Computer Science and Service System, CSSS 2011 - Proceedings*, pp. 2440–2443, 2011, doi: 10.1109/CSSS.2011.5974909.
- [49] "IntelliJ IDEA – the Leading Java and Kotlin IDE." Accessed: Jun. 20, 2024. [Online]. Available: <https://www.jetbrains.com/idea/>