



Instituto Politécnico do Cávado e do Ave
Escola Superior de Tecnologia

Engenharia em Desenvolvimento de Jogos Digitais

Ademar Baptista da Silva Valente nº23155

Prescrição de Terapêutica na Plataforma *Mozy*

Junho de 2024



Instituto Politécnico do Cávado e do Ave
Escola Superior de Tecnologia

Engenharia em Desenvolvimento de Jogos Digitais

Relatório de Estágio/Projeto

Trabalho efetuado sob a orientação de

Duarte Duque

Junho de 2024

Agradecimentos

Este trabalho não seria possível sem o auxílio de várias pessoas. Quero agradecer em primeiro lugar aos intervenientes diretos: os elementos da *Mike Team* que participaram ativamente para o meu crescimento, e que se preocuparam de forma constante com a minha integração tendo feito tudo para que me sentisse parte da equipa nestes últimos meses.

Em particular tenho de agradecer ao Diogo Silva por ser um *Team Leader* presente e sempre pronto a ajudar a resolver problemas; ao Nuno Vieira por todo o apoio e conselhos em todos os momentos; e ao Pedro Barbosa pela presença contínua e por ser incansável nas explicações como *Team Buddy*. Sem a ação cuidada destes intervenientes estou certo de que não teria tido tanto sucesso na concretização dos meus objetivos. Por toda a compreensão no momento de enquadrar a minha vida pessoal com a realização do estágio, o meu mais profundo obrigado a eles.

Do lado do Politécnico do Cávado e do Ave, quero aqui prestar os meus mais sentidos agradecimentos ao Professor Duarte Duque; pela máxima prontidão na resposta às dúvidas que me surgiram desde o primeiro momento (não só neste estágio, mas desde o momento de decisão em ingressar nesta Licenciatura), e pelo cuidado com que supervisionou todos os componentes desta disciplina.

Extrapolando para fora da realização deste estágio, tenho de prestar um agradecimento profundo aos companheiros mais presentes na realização desta Licenciatura. Pelo apoio no processo de aprendizagem, e pela amizade em todos os momentos; tenho de agradecer ao Frederico Silva, ao Luís Pereira, à Vânia Pereira, ao Henrique Azevedo e ao José Lourenço. A mão estendida com que me presentearam nos vários momentos de aflição fica para a minha história fora destes três anos. Obrigado.

Finalmente, para o meu motor de força e coragem todos os agradecimentos são poucos. Para a minha mulher Joana, que esteve presente em todos os momentos; que nunca deixou de acreditar que seria possível chegar ao lugar onde estou, e que uma a uma afastou todas as adversidades que apareciam no caminho. Tudo o que consegui realizar tem o seu cunho, e vou agradecer-lhe eternamente todos os frutos que possam advir disso.

Em último lugar, quero dedicar a realização deste trabalho aos meus filhos, João e Alice. Que um dia, no caminho que ainda têm para percorrer, possam contar aos seus com orgulho que o pai um dia decidiu que queria mudar a sua vida para ser mais feliz.

Resumo

Este documento explicita os processos e procedimentos efetuados no decorrer do estágio curricular realizado na empresa *Glintt Global Solutions, S.A.*; mais concretamente no desenvolvimento da Plataforma *Mozy*. Para tal efeito foi feita integração numa equipa de desenvolvimento da empresa, que se tem dedicado à prescrição de terapêutica.

Ao longo do documento são descritas todas as fases do projeto. É efetuada uma contextualização, de forma a compreender melhor a área de negócio e perceber o que o mercado oferece; seguindo-se uma descrição técnica do produto, bem como da sua arquitetura e dos seus constituintes. Finalmente, procede-se à descrição do trabalho efetuado, com a apresentação das metodologias de trabalho da equipa e demonstração das tarefas concretizadas ao longo do tempo de estágio.

No final concluiu-se que a solução atingiu as expetativas, cumprindo os requisitos definidos.

Palavras - chave: *Computerized Provided Order Entry, Mozy, Widget, Middleware, Composite.*

Índice

Agradecimentos	v
Resumo.....	vi
Índice de Figuras	ix
Índice de Tabelas.....	x
Lista de acrónimos	xi
1. Introdução	1
1.1. Enquadramento.....	1
1.2. Apresentação do Estágio.....	1
1.2.1. Planeamento	1
1.2.2. Reuniões de Acompanhamento	3
1.3. Apresentação da Organização.....	4
1.4. Contributos deste trabalho	6
1.5. Organização do Relatório	6
2. Matriz Teórica	8
2.1. Contextualização	8
2.1.1. <i>Computerized Physician Order Entry</i>	8
2.1.2. <i>Electronic Health Record</i>	9
2.2. Sistemas de gestão médica e hospitalar	10
2.2.1. <i>Globalcare</i>	11
2.2.2. Sistema Integrado de Gestão Hospitalar.....	12
2.2.3. <i>DrChrono</i>	12
2.2.4. <i>Carecloud Medical Software</i>	13
2.2.5. Comparativo entre as soluções apresentadas	13
2.3. Soluções CPOE existentes	14
3. Descrição do Produto	16
3.1. Tecnologias usadas na solução	16
3.1.1. <i>Frontend</i>	16
<i>React.js</i>	16
3.1.2. Sistemas de Comunicação entre peças	17
<i>Google Remote Procedure Call (gRPC)</i>	17
<i>GraphQL</i>	17
3.1.3. Sistemas de Comunicação Padronizados na Saúde	17

<i>Fast Healthcare Interoperability Resources (FHIR)</i>	17
3.2. Arquiteturas – atual vs. nova	18
4. Descrição Técnica	21
4.1. Metodologia – <i>Agile</i> e <i>Scrum</i>	21
4.2. Ferramentas de trabalho – <i>Confluence</i> e <i>Jira</i>	23
4.3. Mecanismos de Segurança.....	24
4.4. “Diário de Bordo”	25
5. Conclusões.....	33
5.1. Resumo do Relatório.....	33
5.2. Análise qualitativa dos objetivos iniciais.....	33
5.3. Limitações e trabalho futuro.....	35
5.4. Apreciações finais.....	36
Bibliografia	37
Anexos.....	39
Anexo I – <i>Workflow</i> das equipas de Desenvolvimento pela metodologia <i>Agile</i> (representação de todas as tecnologias utilizadas no processo contínuo).....	40
Anexo II – Organização do <i>SWS Cockpit</i> (<i>Confluence</i>)	41
Anexo III – <i>SWS Cockpit/Mike Team</i> – exemplo de <i>Kanban</i> (<i>Timeline</i>).....	42
Anexo IV – <i>SWS Cockpit/Mike Team</i> – exemplo de <i>Sprint</i> (<i>Backlog</i>)	43
Anexo V – <i>SWS Cockpit/Mike Team</i> – <i>Story</i> e <i>Subtasks</i>	44
Anexo VI – <i>SWS Cockpit/Mike Team</i> – Descrição de <i>Story</i>	45

Índice de Figuras

Figura 1 - Milestones de estágio.	2
Figura 2 - Organograma da empresa e localização do estágio.....	5
Figura 3 - Sistema CPOE integrado num sistema de software intra-hospitalar (EHR).	10
Figura 4 - Mercado CPOE global, projeção 2023 - 2030 por regiões mundiais.....	15
Figura 5 - Arquitetura Mozy.	19
Figura 6 - Atualização da Arquitetura Mozy.....	20
Figura 7 - Agile workflow.....	21
Figura 8 - testes unitários com 100% de aprovação.	24
Figura 9 - prescription-protocol-management-widget com as alterações requisitadas.....	27
Figura 10 - prescription-doses-recalculation-widget.	29
Figura 11 - clinical-data-widget.	31
Figura 12 - Solução do Design System existente.....	32
Figura 13 - Solução do Design System pretendida (esboço).	32
Figura 14 - Evolução da complexidade das tarefas.	34

Índice de Tabelas

Tabela 1 - Reuniões de Estágio.....	4
Tabela 2 - Vantagens e Desvantagens do uso de CPOE.	9
Tabela 3 - Comparativo entre as soluções apresentadas.	14
Tabela 4 -Especificidades Scrum - Mike Team.	22

Lista de acrónimos

API	Application Programming Interface
COVID-19	Coronavirus Disease 2019
CPOE	Computerized Physician Order Entry
DB	Database
DOM	Domain
ECOG	Eastern Cooperative Oncology Group
EHR	Electronic Health Record
EMR	Electronical Medical Record
FHIR	Fast HealthCare Interoperability Resources
gRPC	Google Remote Procedure Call
HL7	Health Level Seven
MCDT	Métodos Complementares de Diagnóstico e Terapêutica
RCM	Recipe Cycle Management
S.A.	Sociedade Anónima
SIGED	Subsistema de Gestão de Doentes
SIGEHP	Sistema Integrado de Gestão Hospitalar
SIGERH	Subsistema de Recursos Humanos Hospitalares
SISCLI	Processo Clínico Eletrónico
SISCONT	Subsistema de Serviços Financeiros Hospitalares
SISFARM	Subsistema de Farmácia Hospitalar
SISPRO	Subsistema de Aprovisionamento Hospitalar

1. Introdução

1.1. Enquadramento

O presente documento pretende apresentar o conteúdo funcional do meu estágio curricular, enquadrado no programa do 3º ano da Licenciatura em Engenharia em Desenvolvimento de Jogos Digitais, através da disciplina denominada como Projeto/Estágio. A disciplina tem uma forte componente prática, com a duração total estimada em 400 horas de trabalho; e foi realizada entre os dias 15 de fevereiro e 8 de junho do presente ano.

Desta forma, e por proposta pessoal minha, o estágio foi realizado na empresa **Glintt Global Solutions, S.A.**, tendo sido integrado numa equipa de desenvolvimento, diretamente ligada à criação e evolução do produto **Mozy**. No âmbito deste estágio tive a oportunidade de desempenhar funções de observação, colaboração e criação supervisionada. Com a permissão e supervisão da equipa, foi-me dada a oportunidade de realizar o estágio de forma híbrida, tendo sido disponibilizado computador da empresa para o efeito. Dentro da equipa de desenvolvimento tive o auxílio de todos os seus elementos, e pude explorar os vários problemas e soluções que compõem o trabalho diário da equipa, tendo, no entanto, um apoio mais continuado por parte do *Team Leader*, do orientador do estágio, e de um elemento específico da equipa, com o qual passei a fazer trabalho em colaboração e sob supervisão (denominado de *buddy*).

A *Mozy* é um produto destinado a serviços de saúde, que iniciou o seu processo de desenvolvimento em meados de julho de 2023. Procura ser um aplicativo que agrega as informações importantes de cada doente, e as disponibiliza através de interfaces direcionados aos vários tipos de profissionais de saúde, com o objetivo de fornecer soluções simplificadoras, claras e diretas para a otimização da prática de cuidados de saúde. Apesar de ter tido o seu lançamento em outubro de 2023 e ter já sido aplicado em algumas unidades de saúde, o seu desenvolvimento mantém-se em paralelo até à presente data.

1.2. Apresentação do Estágio

1.2.1. Planeamento

Tal como referido anteriormente, propus-me a realizar o estágio na empresa *Glintt Global* por várias razões. A empresa enquadra grande parte dos seus desenvolvimentos na área da saúde, e esse foi o primeiro motivo. Sou enfermeiro há 19 anos, e entendi que este estágio poderia ser um elo de ligação entre a minha profissão e as várias componentes teórico-práticas da Licenciatura. Para além disso, achei que ter conhecimentos teóricos na principal área de

desenvolvimento do produto *Mozy* me poderiam ajudar a ter uma evolução maior e mais sólida das minhas habilidades de programação.

Outra das razões, e uma das maiores motivações para efetuar o estágio no seio desta empresa, foi pelo fato de ter uma longevidade assinalável no setor que desenvolve, e consequentemente, uma base consolidada que senti que poderia trazer-me um *know-how* no que diz respeito a integrar-me no setor empresarial de desenvolvimento tecnológico.

Resumindo, escolhi realizar a disciplina Projeto/Estágio através deste estágio por reunir condições ótimas para o desenvolvimento de três elementos que considerei como fundamentais:

- Aproveitamento do legado teórico não relacionado com os conteúdos da licenciatura, que trago dos meus anos de experiência como enfermeiro;
- Aplicabilidade e desenvolvimento dos conteúdos programáticos adquiridos ao longo da execução da licenciatura, nomeadamente linguagens de programação, ambientes de desenvolvimento, e tecnologias aprendidas;
- Integração funcional no enquadramento do dia-a-dia de uma empresa de Desenvolvimento, e consequente aplicabilidade de metodologias de trabalho em equipas multidisciplinares.

Posto isto, e tendo tido a aceitação de todas as partes para a realização do estágio, procurei projetar os objetivos de forma cuidada, seguindo uma evolução favorável para a aplicação de conhecimentos e aquisição de novos, para além da melhor integração no seio da equipa em que me colocaram. O esquema abaixo apresentado procura representar cronologicamente o plano inicial para o crescimento no local de trabalho, e a tentativa de adquirir um aumento de competências de forma sólida e construtiva.

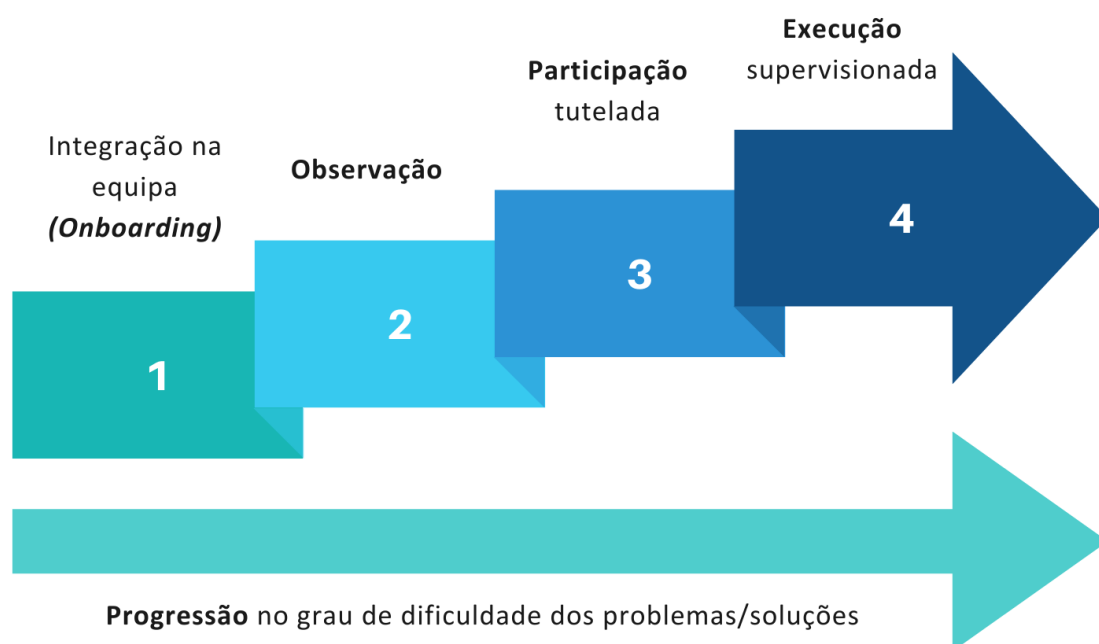


Figura 1 - Milestones de estágio.

1.2.2. Reuniões de Acompanhamento

Com vista a manter uma evolução favorável, todas as partes envolvidas (Aluno, Supervisor da Escola e Supervisor da Instituição) foram mantendo um contato contínuo. Ao longo da realização do estágio procurei sempre manter as partes responsáveis atualizadas de como estava a sentir a minha evolução no decorrer do estágio. Esta constante atualização de informação revelou grande importância, por forma a poder observar todos os ângulos da minha evolução; que por vezes não é perfeita quando se faz apenas uma autoavaliação dos procedimentos em que estamos imbuídos.

O acompanhamento com o Supervisor da Instituição foi diário, fruto de ter integrado todas as tarefas da *Mike Team*, inclusive as reuniões diárias online (*Daily Scrum*). Para além disso, tivemos reuniões intercalares onde foram abordadas questões relacionadas com a evolução do estágio, e direcionamento do foco para que pudesse exponenciar a minha experiência na empresa e na equipa.

Com o Supervisor da Escola procurei manter o contato relatando com frequência a evolução do estágio, as tomadas de decisão feitas no seio da empresa, e aconselhamento para elaboração deste relatório. O *feedback* foi, como é sempre, imediato.

Na tabela seguinte apresentam-se a listagem dos principais momentos de acompanhamento do trabalho realizado no âmbito deste estágio.

Data da Reunião	Membros reunidos	Tema(s) da Reunião
15 de fevereiro	Responsável por acolhimento nos escritórios da <i>Glintt Global</i> (presencial).	<ul style="list-style-type: none"> - Avaliação de conhecimentos; - Definição dos objetivos de estágio; - Entrega de permissões de acesso informático; - Iniciação de <i>Onboarding</i>.
7 de março	Supervisor da Escola (via <i>e-mail</i>).	<ul style="list-style-type: none"> - Apresentação do produto de estágio; - Preenchimento do Anexo IV (Termo de Estágio).
11 de março	<i>Scrum Master</i> e Supervisor da Instituição (via videochamada).	<ul style="list-style-type: none"> - Integração funcional na <i>Mike Team</i>; - Instalação de software para desenvolvimento; - Entrega de permissões de acesso informático para o projeto da equipa.
12 de março	<i>Buddy</i> (via videochamada)	<ul style="list-style-type: none"> - Apresentação;
18 de abril	<i>Team Leader</i> e Supervisor da Instituição (via videochamada).	<ul style="list-style-type: none"> - Ponto de situação de estágio;

		- Apresentação e discussão dos pontos principais do relatório de estágio.
19 de abril	Supervisor da Escola (via <i>e-mail</i>).	- Ponto de situação de estágio; - Apresentação e discussão dos pontos principais do relatório de estágio.
8 de maio	<i>Team Leader</i> e Supervisor da Instituição (via videochamada).	- Ponto de situação de estágio.
3 de junho	<i>Team Leader</i> , Supervisor da Instituição e Buddy (via videochamada).	- Apreciação do estágio; - Apresentação e discussão do relatório de estágio.
6 de junho	Supervisor da Escola (via <i>e-mail</i>).	- Apreciação do estágio; - Apresentação e discussão do relatório de estágio.

Tabela 1 - Reuniões de Estágio.

1.3. Apresentação da Organização

A *Glintt Global Solutions*, integrante do consagrado grupo *Global Intelligent Technologies* (*Glintt*), destaca-se como uma inovadora multinacional de tecnologia e consultoria, com uma trajetória de mais de duas décadas de sucesso. Esta organização orgulha-se de contar com uma equipa de mais de 1200 colaboradores especializados, distribuídos por vários escritórios estratégicos em 6 países: Portugal, Espanha, Angola, Brasil, Reino Unido e Irlanda. Atualmente, a *Glintt* é líder no fornecimento de consultoria e serviços tecnológicos na área da Saúde, com soluções implementadas em aproximadamente 430 hospitais e 500 clínicas. No setor da Farmácia, mais de 14.000 farmácias na Península Ibérica confiam no *software* de gestão da *Glintt*, que também oferece um abrangente portfólio que inclui desde a conceção e projeção de espaços comerciais até à automação, infraestruturas e consumíveis. ^[1]

Em meados de 2020 a organização lançou um ambicioso projeto chamado **Viewer**. Este projeto foi concebido para implementar uma solução orientada a micro aplicações, com o objetivo de atender às complexas necessidades informáticas clínicas de infraestruturas hospitalares. Embora a organização já possuísse soluções análogas, o projeto *Viewer* adotou uma abordagem diferenciadora: aprimorar o processo de desenvolvimento, incorporar tecnologias mais recentes e avançadas, assimilar os ensinamentos dos erros de soluções anteriores e, por fim, entregar um produto de valor acrescido ao cliente.

A visão do produto é clara: a solução deve capacitar os profissionais de saúde a otimizar o tempo gasto no manuseio do *software*. Almeja-se não apenas tempos de resposta ágil, mas também uma intuitiva organização da informação e dos processos, de modo que os usuários possam realizar suas tarefas com maior eficiência. Quanto menos tempo médicos e enfermeiros

dedicarem ao uso da aplicação, mais atenção poderão dispensar aos pacientes, resultando em benefícios para todos os envolvidos.

Nas consultas médicas, por exemplo, os médicos podem maximizar o tempo dedicado aos pacientes, graças à otimização do *software*. A solução possibilita que o médico se concentre em escutar e examinar o paciente, apoiado por um software que requer menos interação e tenta ser mais eficaz. O *Viewer* evoluiu, e foi lançado para utilização em 2022. Fruto da sua evolução, baseado na sua arquitetura (que será analisada mais à frente neste documento) e na cultura portuguesa, mudou o seu nome para **Mozy**.

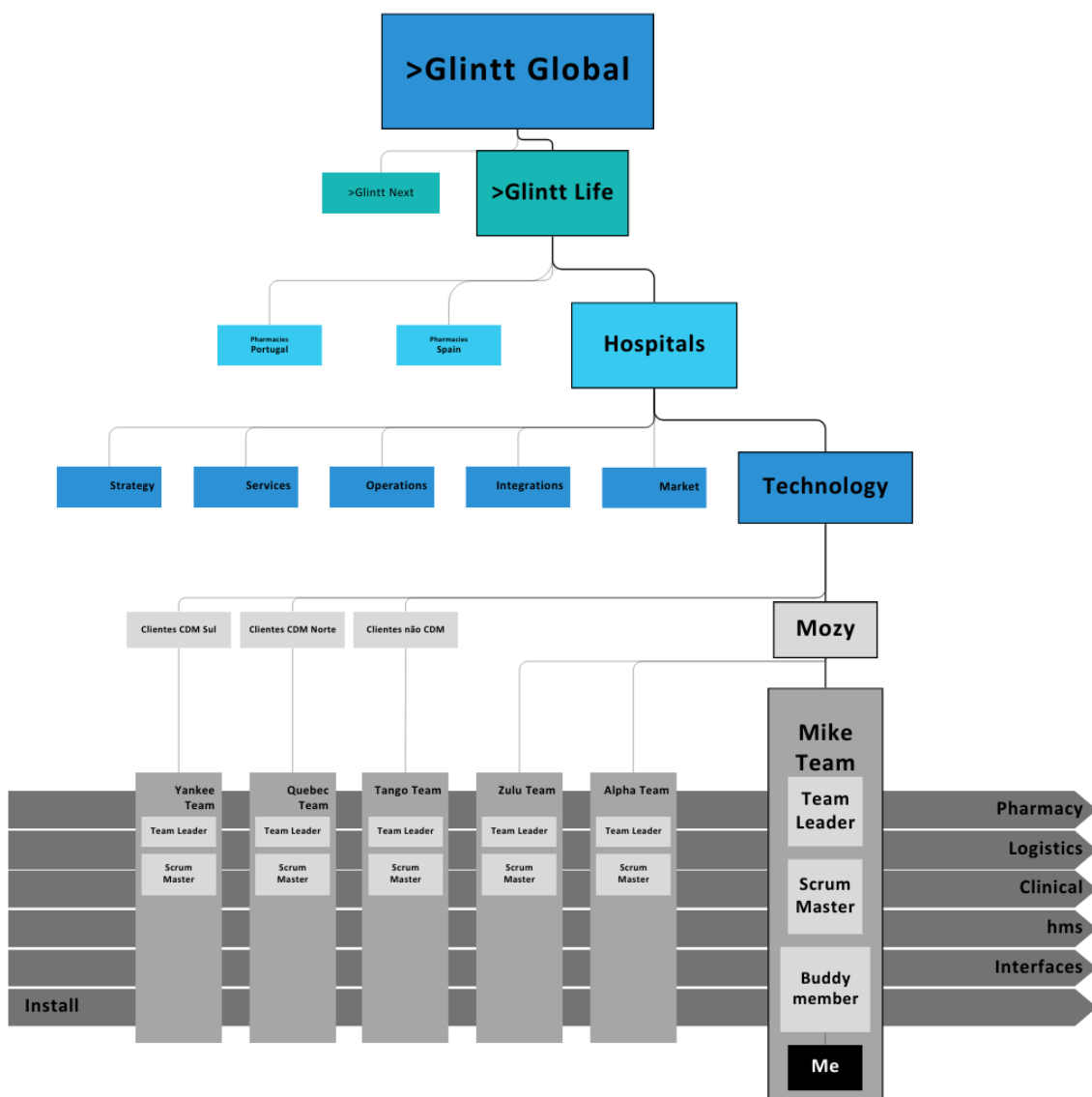


Figura 2 - Organograma da empresa e localização do estágio.

1.4. Contributos deste trabalho

Após a averiguação das competências adquiridas na Licenciatura, e para potencializar o conhecimento e desenvolver competências no âmbito empresarial, foi definido um único objetivo: integrar o normal funcionamento de uma equipa de desenvolvimento, e realizar as tarefas que um profissional recém-chegado à empresa realiza. Desta forma, é expectável adquirir a experiência total de uma integração à vida profissional, bem como realizar os passos funcionais dessa mesma integração. Torna-se assim possível realizar uma integração progressiva e fragmentada, com todo o cuidado necessário para uma aprendizagem inicial sólida.

Assim, o contributo do meu trabalho passa por tentar ser um elemento adicional da equipa, funcional, com um crescendo de competências e independências nas tarefas do dia-a-dia. Com o evoluir do tempo de realização do estágio, procura-se adquirir funcionalidades que permitam com segurança realizar todo o tipo de tarefas; cada vez com maior independência, e com grau de complexidade crescente. No final, o plano é acrescentar volume de trabalho realizado ao projeto desenvolvido pela equipa.

1.5. Organização do Relatório

O presente relatório é composto por cinco pontos fundamentais:

- **Introdução:** Esta seção fornece o enquadramento do estágio, a apresentação do mesmo, a apresentação da organização, e a projeção dos contributos para a realização do trabalho. A organização do relatório (aqui efetuada) também é descrita neste capítulo. A introdução serve para contextualizar o relatório e fornecer uma visão geral do que será discutido;
- **Matriz Teórica:** Este capítulo procura apresentar e explorar soluções de gestão hospitalar, comparando-as de acordo com aspetos tidos como relevantes na construção da *Mozy*, produto no qual foi centrado o estágio. O objetivo deste capítulo é fornecer um entendimento aprofundado do contexto em que o estágio foi realizado e das tecnologias utilizadas;
- **Descrição do Produto:** Neste capítulo procura-se relatar afincadamente o produto central do estágio: a *Mozy*. Aborda-se, sumariamente, as tecnologias usadas para a criação da solução; a arquitetura em vigor e transição para a nova arquitetura que progressivamente está a acontecer neste momento, tal como os constituintes da *Mozy*;
- **Descrição Técnica:** Aqui serão abordadas as metodologias e funcionalidades de trabalho, e os pontos importantes a retirar do estágio face ao que foi apreendido na licenciatura. Inclui secções relacionadas com a metodologia de trabalho utilizada (*Agile*), a estrutura de realização de tarefas e a segurança na realização do trabalho de desenvolvimento;

- **Conclusões:** Este capítulo contém um resumo do relatório, aspetos positivos e objetivos realizados, limitações e dificuldades para um trabalho futuro, e uma apreciação final. Tem como propósito resumir o relatório e refletir sobre o trabalho realizado.

Além disso, o relatório inclui uma **Bibliografia**, que lista todas as fontes que auxiliaram na realização do relatório, e **Anexos**, que contêm informações complementares que não foram incluídas no corpo principal do relatório por serem ilegíveis como Figura.

2. Matriz Teórica

Este capítulo procura aprofundar os conhecimentos teóricos na área de atuação do produto, de modo que seja possível uma contextualização de toda a sua estrutura e propósito. Inicialmente é feito um estudo do que é um **Computerized Physician Order Entry** (CPOE), propósito no qual se enquadram as funcionalidades da *Mozy*, seguido de uma avaliação dos seus benefícios e desvantagens.

Posto isso, procede-se à apresentação sucinta de algumas **soluções de gestão médica e hospitalar** e **soluções CPOE existentes** atualmente no nosso país e no mundo. Para além da apresentação, o capítulo encerra um estudo comparativo das soluções apresentadas face a aspetos relevantes no mercado atual.

Adicionalmente, será feita uma exposição simples e concisa das **ferramentas e tecnologias usadas** pelo produto, que foram decididas como sendo adequadas para a solução do problema de construção.

Finalmente, será apresentado a **Mozy**, produto que foi a peça central de atuação e aprendizagem. Dada a sua complexidade, procurar-se-á apresentar a arquitetura conceptual, e a transição que a mesma está a ter neste momento; bem como os constituintes básicos (as “peças”) e as suas funcionalidades.

2.1. Contextualização

2.1.1 *Computerized Physician Order Entry*

O *Computerized Physician Order Entry* (CPOE) é um sistema digital que permite aos profissionais de saúde inserir instruções médicas para o tratamento de pacientes de forma eletrónica. As ordens inseridas são comunicadas através de uma rede de computadores para a equipe médica ou para os departamentos (farmácia, laboratório ou radiologia) responsáveis por cumprir a ordem. Isto proporciona uma redução do tempo necessário para distribuir e completar ordens; e aumenta a eficiência ao reduzir erros de transcrição, prevenindo a entrada de ordens duplicadas e simplificando a gestão de inventário e faturação.

As funcionalidades do CPOE são frequentemente adotadas para reduzir erros médicos no trabalho.^[2] Estudos mostram que a implementação do CPOE pode reduzir significativamente a incidência de erros de medicação. O CPOE pode ajudar a evitar erros de medicação, pois uma ordem de medicação deve cumprir certos critérios, como boa legibilidade e abrangência. Além disso, as funcionalidades do CPOE, como suporte de dosagem, alertas de interação adversa e suporte à decisão clínica, também auxiliam um tratamento ao doente correto e coerente.^[3]

No entanto, a utilização de sistemas digitais deste género e de outros pode ter as suas desvantagens; nomeadamente uma grande dependência da tecnologia para as práticas na

saúde, o aumento da carga de trabalho, e o surgimento de novos erros de prescrição fruto de erros de *software*.

Dentro das várias análises efetuadas, cito um estudo feito (Rouayroux et al., 2019) nos departamentos de diabetes e cardiologia de um hospital, imediatamente após a implementação de um sistema CPOE. Os resultados demonstraram que, ao início, os tipos de erro mais comuns estavam relacionados com a informatização e com a não conformidade com diretrizes ou contraindicações. Antes da implementação, as duas categorias de erro mais comuns eram indicações não tratadas e dosagens supratrapêuticas. Estes resultados indicaram que o perfil de erro de prescrição se alterou, com os erros de informatização a tornarem-se os mais frequentes. No entanto, um ano após a implementação, verificou-se um decréscimo deste tipo de erros, possivelmente associado a uma melhor formação por parte dos profissionais de saúde para o uso destes sistemas e à melhoria do software. Os principais benefícios e desvantagens, extraídos da literatura supracitada, são sumariados na tabela abaixo apresentada.^[4]

Vantagens	Desvantagens
Redução de erros na prescrição de medicamentos e na prestação de cuidados	Dependência nos aparelhos tecnológicos inerentes a esta prática
Aumento global na rapidez de comunicação com outros sistemas encadeados	Aumento da carga de trabalho para os profissionais de saúde
Aumento do retorno no investimento	Erros iniciais provocados associados ao uso de software

Tabela 2 - Vantagens e Desvantagens do uso de CPOE.

2.1.2 *Electronic Health Record*

O CPOE é geralmente integrado num **Electronic Health Record** (EHR), ou Registro Eletrónico de Saúde.

EHR, ou Electronic Health Record (Registro Eletrónico de Saúde), é um sistema digital que armazena e gere informações de saúde do paciente. As funcionalidades de um EHR podem incluir questões importantes como o acesso a informações completas e atualizadas do paciente, gerenciamento de registos, comunicação entre os profissionais de saúde, rastreamento e análise de dados de saúde da população, e melhoria da qualidade de atendimento.

Além disso, o EHR permite gerir a informação de saúde dos pacientes num formato digital e os seus dados podem ser partilhados com outros fornecedores através de múltiplas organizações. Ele concede acesso a ferramentas baseadas em provas que os fornecedores podem utilizar para tomar decisões e recomendações sólidas sobre os cuidados de um paciente.

Uma integração bem executada do CPOE com o EHR simplifica os fluxos de trabalho clínicos, eliminando redundâncias na entrada de dados. Quando um médico insere uma ordem através do CPOE, os dados relevantes são instantaneamente atualizados no EHR. Isso permite que os profissionais de saúde tomem decisões baseadas em informações atualizadas do paciente.

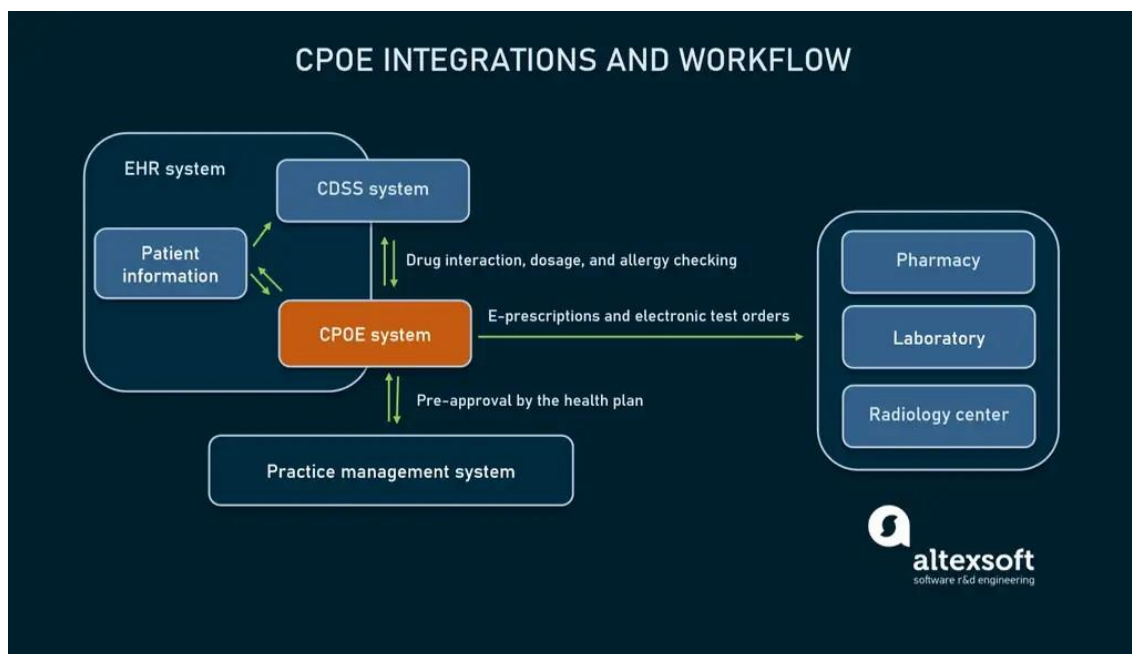


Figura 3 - Sistema CPOE integrado num sistema de software intra-hospitalar (EHR).

A figura apresenta, de um modo genérico, como o um sistema CPOE interage com o sistema eletrónico global da instituição da saúde que incorpora. De uma forma muito simples, o utilizador dá entrada no sistema EHR através do CPOE, e as alterações que por ele sejam feitas (prescrição, alteração ou anulação de terapêutica ou exames, por exemplo) vão ser guardadas nos registos do doente e recebidas por outros serviços/áreas de atuação hospitalar (radiologia, laboratório, farmácia, enfermagem).^[5]

Este fluxo contínuo gera automatismos positivos, diminui o tempo expedido na execução dos mesmos, diminui os erros; e para efeitos de qualidade de trabalho traz transparência aos procedimentos, pelo seu registo claro e direto.

2.2. Sistemas de gestão médica e hospitalar

Uma vez que a solução *Mozy* foi desenvolvida para integrar um sistema EHR, segue-se com um estudo de soluções presentes no mercado. O estado da arte deste tipo de softwares aqui apresentado procura fazer um estudo sumário das principais características de cada um deles, e uma vez mais os prós e contras que cada um dos softwares traz ao utilizador.

Por forma a poder estabelecer também uma simetria entre diferentes realidades do mercado, apresentaremos dois produtos nacionais e dois internacionais.

Assim, no mercado nacional é pertinente apresentar um produto da *Glintt* que atua neste âmbito, e que lidera o mercado na área da saúde: o **Globalcare**. Por outro lado, será apresentado um concorrente direto do mercado, utilizado em várias instituições âmbito hospitalar; o Sistema Integrado de Gestão Hospitalar (**SIGHP**). Pretende-se desta forma trazer conteúdo ao nosso estudo, fruto da exposição da concorrência no mercado; fator muito importante para a constante melhoria de qualidade nos produtos apresentados. É pertinente ainda referir que outras soluções presentes no mercado não foram alvo de estudo por falta de documentação existente, ou indisponibilidade dos seus representantes em divulgar especificidades dos seus produtos.

No campo internacional serão analisadas também duas soluções de implementação: a **Dr.Chrono**; e a **CareCloud** Medical Software. Estes *softwares* integram as melhores avaliações do *Business News Daily*.^[6]

2.2.1. *Globalcare*

O *Globalcare* é um *software* de implantação hospitalar desenvolvido pela *GlinttGlobal*. Identifica-se como sendo uma solução modular, que atua de forma interdependente nas diversas áreas hospitalares em que é implementado.^[7] De forma resumida, os módulos de atuação desta solução são:

- Sistema de Gestão Hospitalar - responsável pelo registo e monitorização do processo do paciente desde a admissão até à alta; e procedimentos relacionados com faturação;
- Clínico - EHR que gere todo o registo de informações clínicas do doente;
- Meios Complementares de Diagnóstico e Terapêutica (MCDT) - módulo que encerra a execução e registos de todos os exames efetuados, segmentados por tipo e especialidade;
- Farmácia e Logística - onde se descreve o circuito dos medicamentos;
- Interoperabilidade - plataforma que liga sistemas, permitindo o consumo de informações entre eles, que possibilita uma monitorização contínua e faculta acesso da solução a terceiros;
- Envolvimento do Paciente - área constituída por funcionalidades que permitem ao paciente marcação e remarcação de consultas, acesso a exames efetuados e histórico de faturação.

2.2.2. Sistema Integrado de Gestão Hospitalar

O Sistema Integrado de Gestão Hospitalar (SIGEHP) é uma solução desenvolvida pela *sisbit*, uma organização orientada para a elaboração de produtos na área da saúde. O SIGEHP tem por objetivo dar resposta às necessidades de informatização global das instituições hospitalares, e permite configuração e adaptação às especificidades dos diversos tipos de organização.^[8] O *software* é composto pelos seguintes subsistemas:

- Subsistema de Gestão de Doentes (SIGED) - responsável por gerir os momentos de admissão e alta do paciente, processando todos os dados necessários à faturação, e guardar documentos externos necessários para o preenchimento do processo do doente. Gere também os MCDT;
- Processo Clínico Eletrónico (SISCLI) - propõe uma dinâmica de trabalho “paper-free” e, tendo assim uma informatização nas áreas de internamento, Urgência, Consulta Externa, Hospital de dia, Bloco Operatório, o que inclui o resultado de meios complementares de diagnóstico. Possui configuração de perfis de atuação médica e de enfermagem;
- Subsistema de Farmácia Hospitalar (SISFARM) - responsável pela gestão e aquisição de *stock* de medicamentos, bem como questões de faturação inerentes ao consumo;
- Subsistema de Aprovisionamento Hospitalar (SISPRO) - que controla questões de logística de materiais de consumo hospitalar;
- Subsistema de Recursos Humanos Hospitalares (SIGERH) - módulo que faz toda a gestão de profissionais da instituição hospitalar;
- Subsistema de Serviços Financeiros Hospitalares (SISCONT) - que integra vários componentes modulares importantes para o funcionamento organizacional, como o Módulo de gestão de tesouraria, o módulo de gestão de terceiros, o Módulo de Orçamentação e o Módulo de Contabilidade Geral e Analítica.

2.2.3. DrChrono

A DrChrono é uma empresa sediada em Sunnyvale, Califórnia em 2009, e apresenta um produto com o mesmo nome, utilizado largamente no seu país de origem (estimados 17 milhões de utilizadores^[9]). A aquisição do *software* pode ser feita em vários formatos e pacotes, podendo incluir ou não a totalidade das suas funcionalidades, que são:

- Registo Médico Eletrónico (EMR) - plataforma totalmente integrada para atender às necessidades do médico, de acordo com a sua especialidade. Modular e personalizável, foi concebida para ser de fácil e rápida utilização;
- Telesaúde - que permite agendar e realizar consultas (sujeitas a normas e padrões de qualidade específicos e pré-determinados) diretamente na plataforma;
- Gestão de Ciclo de Receitas (RCM) - auxiliar nos processos de faturação pelos serviços e pelo qual geram receitas; sendo uma solução pelas funcionalidades que tem para seguir planeamentos de gestão financeira;

- Gestão de Prática Médica (Smart Practice Management) - com diversas funcionalidades de auxílio para a prática do utilizador, como lembretes, *tooltips*, etc.;
- Aplicativo *mobile* - com todas as funcionalidades do EHR em sistemas *Apple*;
- Faturação (*Billing*) - com serviços de pagamentos e grafismos detalhados dos serviços prestados.

2.2.4. Carecloud Medical Software

A *CareCloud Inc.* é uma empresa sediada em Nova Jérсия e fundada em 1999. Inicialmente dedicada a serviços de faturação, evoluiu os seus produtos na área da saúde e tem como *software EHR* o *Carecloud Medical Software*, que conta com as seguintes funcionalidades:

- Registo Médico Eletrónico (*EMR*) - um sistema modular que possibilita ao médico gerir e escolher visualizar apenas o que acha importante para a sua prática. Tem registos guardados em *Cloud* para acesso remoto acessível;
- Telesaúde (*talkEMR*) - uma plataforma que permite teleconsulta dentro dos trâmites legais patenteados, nas diversas especialidades e áreas de trabalho da saúde;
- Faturação - com disponibilização de serviços de pagamento remoto;
- Acessibilidade *mobile* e telemonitorização - para possibilitar monitorização em tratamentos domiciliários, e manutenção de vigilância em situações clínicas que necessitem;
- Interoperabilidade - com apresentação de vários parceiros de MCDT com os quais efetua partilha de dados promotora da transparência do serviço que presta ao paciente;
- Inteligência Artificial (*cirrusAI powered by Google*) - que fornece um conjunto de ferramentas auxiliares para a execução de boas práticas de trabalho.^[10]

2.2.5. Comparativo entre as soluções apresentadas

Para se executar um estudo comparativo simples procedeu-se à escolha de critérios que podem ser determinantes na implementação da CPOE no sistema EHR que a instituição hospitalar tenha adotado. Assim, foi decidida a utilização os seguintes critérios:

- Interoperabilidade - visto que a flexibilidade do sistema EHR em execução pode ser determinante para a correta implementação e adaptabilidade de um novo produto CPOE;
- Preço (valor por profissional, por mês) - uma vez que pode ditar em termos orçamentais a decisão de compra do produto tecnológico pela instituição;
- Sistema de faturação integrado - ferramenta importante para a gestão de uma empresa;
- Telesaúde - ponto indicador de vanguarda no atendimento na saúde, e visto como mais-valia fundamental nos dias de hoje;

- Acesso remoto (do profissional de saúde e do doente) - por motivos semelhantes ao ponto anterior.

Expondo então os critérios acima referenciados, foi possível esquematizar as opções que cada um dos softwares providencia através da tabela abaixo apresentada. Não foi possível obter informações relativas a certos critérios e soluções, estando devidamente assinaladas como N/A.

Software	Interoperabilidade	Preço	Sistema de faturação	Telesaúde	Acesso Remoto
<i>Globalcare</i>	✓	N/A	✓	✓	✓
SIGEHP	N/A	N/A	✓	N/A	×
<i>DrChrono</i>	✓	249\$ até 599\$	✓	✓	✓
<i>Carecloud</i>	✓	628\$	✓	✓	✓

Tabela 3 - Comparativo entre as soluções apresentadas.

2.3. Soluções CPOE existentes

Em linha com o estudo efetuado nos *softwares* EHR, procurou-se fazer uma análise similar para as soluções CPOE presentes quer no mercado nacional, quer a nível mundial. No entanto, encontrar de forma unitária uma solução CPOE é cada vez mais complicado. Os produtores de *software* nesta área procuram apresentar a sua gama de produtos de uma forma cada vez mais global, aglomerando nos seus sistemas todas as funcionalidades anteriormente estudadas e não apresenta soluções CPOE individualizadas/independentes.

Foi encontrado um estudo que projeta a evolução do uso de soluções CPOE integradas nos sistemas de saúde até 2030. O estudo é projetado com base num crescimento anual de 6,39%; e justifica os dados de crescimento que apresenta pelo aumento de interesses governamentais, o desenvolvimento e melhoria das instituições de saúde, a preocupação pela transparência de processos como modelo a seguir, e a velocidade favorável do crescimento tecnológico no momento em que o estudo foi projetado.^[11]

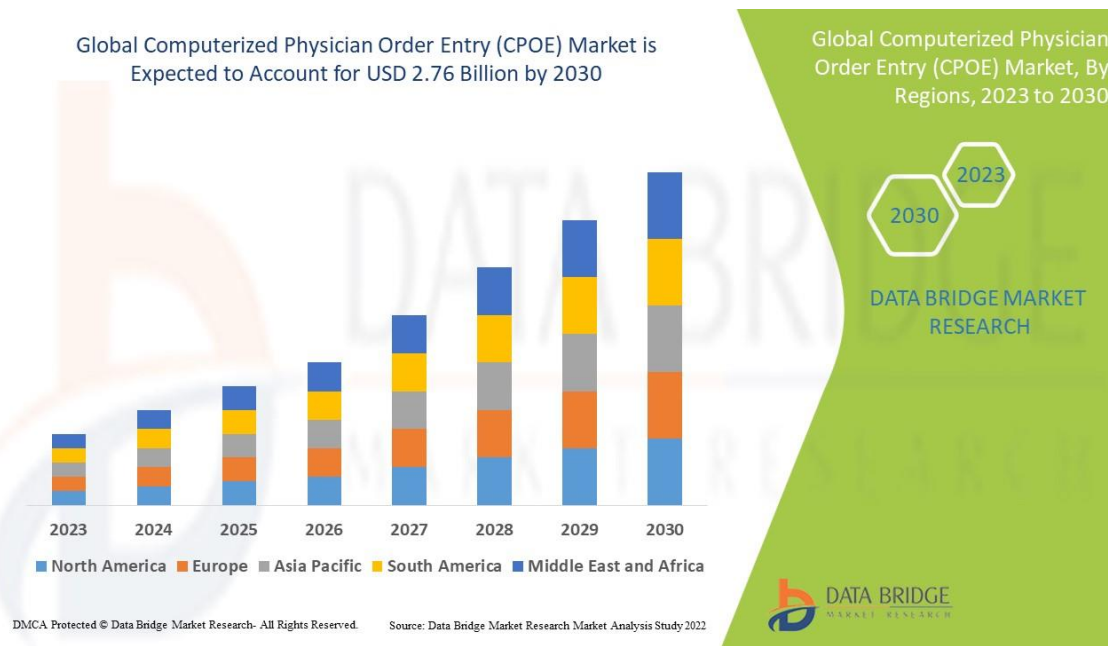


Figura 4 - Mercado CPOE global, projeção 2023 - 2030 por regiões mundiais.

Analisando de forma sumária a segmentação do documento, podemos retirar que o crescimento dos CPOE, e a sua imposição no mercado, vai ser mais eficaz se forem otimizadas as seguintes competências:

- Sistema com capacidade de ser integrado num ou em vários EHR's, por forma a ser apelativo às instituições de saúde em fases iniciais de crescimento;
- Modo de entrega dos serviços baseados na *cloud* em detrimento de produtos com entrega via *web* ou de armazenamento local;
- Componentes leves em modo serviços, em detrimento de instalações de *hardware/software*;
- Globalidade de abrangência nas diferentes áreas de saúde, e implementação em Hospitais, Clínicas, Enfermarias ou Casas de Saúde.

Este estudo apresenta ainda, tal como foi visto anteriormente, as vantagens da utilização de um sistema CPOE, bem como o impacto que o COVID-19 teve no crescimento de utilização deste tipo de meios de registo, fruto da sua rapidez na recolha e armazenamento de informação dos doentes.

3. Descrição do Produto

Neste capítulo será abordado o objeto de trabalho do estágio do autor: a aplicação *Mozy*. Tal como foi dito na nota introdutória deste documento, a solução procurou suprir a necessidade de implementar uma CPOE limpa, coerente, intuitiva e transparente; que desta forma procurasse reduzir de forma crescente a taxa de erros na prática dos diferentes profissionais de saúde. Neste sentido, o objetivo passou e passa por evoluir de forma constante para algo que seja usado de forma simples, rápida, precisa e intuitiva.

A *Mozy* é uma aplicação com uma estrutura micro aplicacional, permitindo a sua comercialização por funcionalidade e por peças de *software* (como um mosaico, alegoria de onde advém o nome “internacionalizado” da nossa aplicação). O paradigma arquitetural associado possibilita assim montar uma instância da aplicação focada unicamente nas funcionalidades do CPOE.

Dada a sua complexidade, o capítulo apresenta uma breve contextualização de algumas das tecnologias usadas na solução; progredindo para a apresentação de duas arquiteturas – a atual e a próxima (neste momento a ser implementada, e a substituir gradualmente a anterior), e da estrutura dos componentes de ambas.

3.1. Tecnologias usadas na solução

3.1.1. *Frontend*

React.js

React é uma ferramenta *JavaScript* usada para criar interfaces de utilizador. Baseia-se em componentes que podem ser reutilizados e que gerem seu próprio estado, formando a interface.

Uma característica importante do *React* é o uso do *Virtual DOM (domain)*, que pode ser renderizado tanto no cliente quanto no servidor, permitindo uma comunicação bidirecional. O *Virtual DOM* renderiza partes da árvore de nós com base nas mudanças de estado, minimizando a manipulação do *DOM*. Isso permite que o *React* interaja com o *DOM* em memória, em vez do *DOM* gerado pelo navegador, resultando em um desempenho mais robusto. Um algoritmo compara o *Virtual DOM* e o *DOM*, atualizando apenas os nós que mudam. ^[12]

Isso permite o desenvolvimento de aplicações *web* complexas, onde os dados podem mudar sem a necessidade de atualizar a página no navegador. Por isso foi a ferramenta escolhida no desenvolvimento do *frontend* da *Mozy*.

3.1.2. Sistemas de Comunicação entre peças

Google Remote Procedure Call (gRPC)

O gRPC é uma estrutura de Chamada de Procedimento Remoto (RPC) que pode ser executada em qualquer ambiente. É um projeto de código aberto que possui um sistema baseado em contratos de dados e um protocolo de comunicação que pode ser usado em várias plataformas, simplificando e gerenciando a comunicação entre serviços. Os contratos de dados são definidos em arquivos de *protocol buffers* usando uma linguagem específica chamada *proto3*, que permite a definição de vários tipos numéricos, datas, listas, entre outros, para definir as mensagens de entrada e saída.¹³

Pela rapidez de performance e desenvolvimento, estabilidade alta e prioridade na comunicação entre serviços; foi a ferramenta utilizada para a comunicação interna no *backend*, visto ser valorizada a rapidez de execução.

GraphQL

O *GraphQL* é uma linguagem de consulta e um ambiente de execução para *API*. Permite que os clientes definam exatamente os dados que precisam, tornando as respostas mais precisas. É uma tecnologia independente de base de dados e pode ser implementada em qualquer contexto onde uma *API* é necessária. Desenvolvido originalmente pelo *Facebook* e disponível publicamente desde 2015, o *GraphQL* permite que os clientes façam solicitações personalizadas aos serviços de dados, garantindo que apenas os dados necessários sejam retornados pelo servidor. Isso torna a interação mais flexível e eficiente, evitando o envio de dados desnecessários que poderiam sobrecarregar a largura de banda.^[14]

Para o *frontend* é necessária flexibilidade na especificação dos dados a receber em todos os pedidos, de forma a filtrar apenas pelo estritamente necessário. Desta forma, o *GraphQL* foi a ferramenta escolhida.

3.1.3. Sistemas de Comunicação Padronizados na Saúde

Fast Healthcare Interoperability Resources (FHIR)

O FHIR é um padrão criado pela *Health Level Seven (HL7)* especificamente para o setor de saúde. Ele é composto por componentes modulares chamados recursos, que podem ser combinados para criar soluções que resolvem problemas clínicos e administrativos. Este padrão começou a ser desenvolvido em 2012, em resposta à necessidade do mercado de melhores métodos para

a troca de dados de saúde, num momento em que a quantidade desses dados estava a aumentar rapidamente.^[15]

Além de facilitar a troca de dados, o FHIR oferece várias vantagens para os desenvolvedores, como a fácil implementação, o uso gratuito sem restrições, o acesso a ferramentas online, e uma base em vários padrões web.

Por essas razões os microsserviços FHIR foram utilizados na solução, sendo uma das bases do *Backoffice*.

3.2. Arquiteturas – atual vs. nova

A *Mozy* é uma aplicação web hospitalar já implementada em vários estabelecimentos que atuam na área da Saúde. No entanto, e de forma contínua e paralela, encontra-se numa fase de desenvolvimento. Apesar de se propor a responder a várias necessidades hospitalares, atualmente apresenta apenas funcionalidades a serem usadas pelos profissionais de saúde. De forma simplificada e direcionada para o âmbito e fluxo do desenvolvimento do projeto CPOE, identificam-se as seguintes peças de *software* da *Mozy*:

- **Cockpit** – Representa a peça de *frontend* que orquestra e encapsula os *widgets*. Permite diferentes configurações de ecrã com diferentes *widgets*, disposições, tamanhos e fluxos de negócio. Inclui um módulo de autenticação para o profissional de saúde se ligar.
- **Widget** – Cada peça deste tipo representa um *microfrontend*. É responsável por representar visualmente um cenário através de componentes *UserInterface* (UI). Todas estas peças de software são desenvolvidas com recurso ao *React*, utilizam o *GraphQL* para comunicar com a peça *middleware* e utilizam componentes e *templates* de *frontend* disponibilizados por um *design system* do qual a *Glintt* é responsável.
- **Middleware** – Serve como intermediário entre a camada de negócio e os *widgets*. É responsável por responder a pedidos de leitura e escrita de uma ou mais *widgets*. Para isso, comunica com a camada de negócio – o *composite* – através de pedidos gRPC. Recebe e trata linguagem FHIR dos *composites*, mapeando-o em cada caso para uma estrutura mais simples e direta, e envia-a para o widget, numa dinâmica *Single Responsibility Principle* (SRP).
- **Composite** – Onde é trabalhado o negócio. Comunicam com os micro serviços, através de gRPC, enviando-lhes instruções de pedidos a realizar à base de dados. A resposta é devolvida ao *middleware*.
- **Microservices** – É um tipo de peça que realiza CRUD (*Create, Read, Update, Delete*) pedidos à base de dados.¹⁶ Recebe as instruções necessárias do *composite*, como o tipo de operação, os recursos envolvidos e os filtros a aplicar. A base de dados está dividida em coleções e entidades, que representam sempre um recurso FHIR. A resposta é devolvida ao *composite*. O armazenamento é feito numa base de dados não relacional, o *MongoDB*.

As peças apresentadas são usadas na arquitetura atual, e tal como descrito anteriormente, encadeiam-se da seguinte forma:

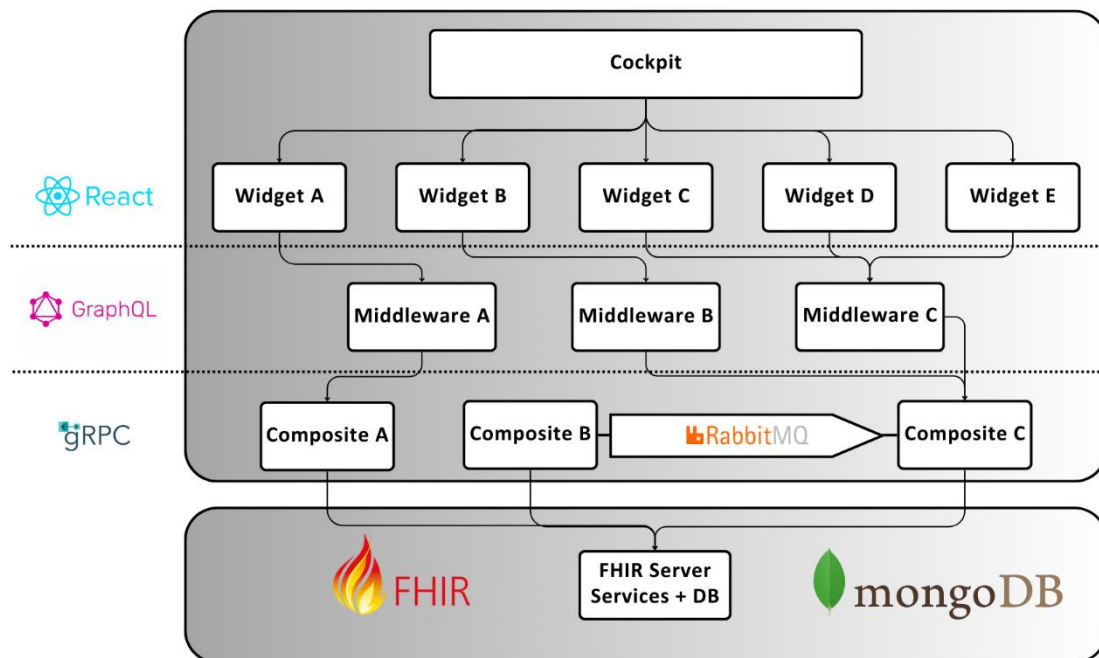


Figura 5 - Arquitetura Mozy.

Em paralelo com o momento em que decorreu o estágio, a arquitetura passou por um período de transformação e transição para uma nova arquitetura funcional. O objetivo dessa mudança passa por reduzir o número dos componentes base da solução (peças), e, consecutivamente; reduzir os custos técnicos e melhorar a consistência no armazenamento de dados.

Dessa forma, existe a remoção das peças de *middleware* e são geradas peças maiores que integram os *composites* – as **API**. Por outro lado, o *FHIR server* migra para uma seção central – a **Cockpit.Framework**, e os micro serviços utilizados são guardados em base de dados, através do *mongoDB*.

Por fim, no que diz respeito a tecnologias utilizadas, a arquitetura deixa cair a utilização de *gRPC* e passa a utilizar *GraphQL* com **Hot Chocolate**; o que vai possibilitar reduzir a complexidade das comunicações; e potencializar o uso da tecnologia e das suas *features*.^[17]

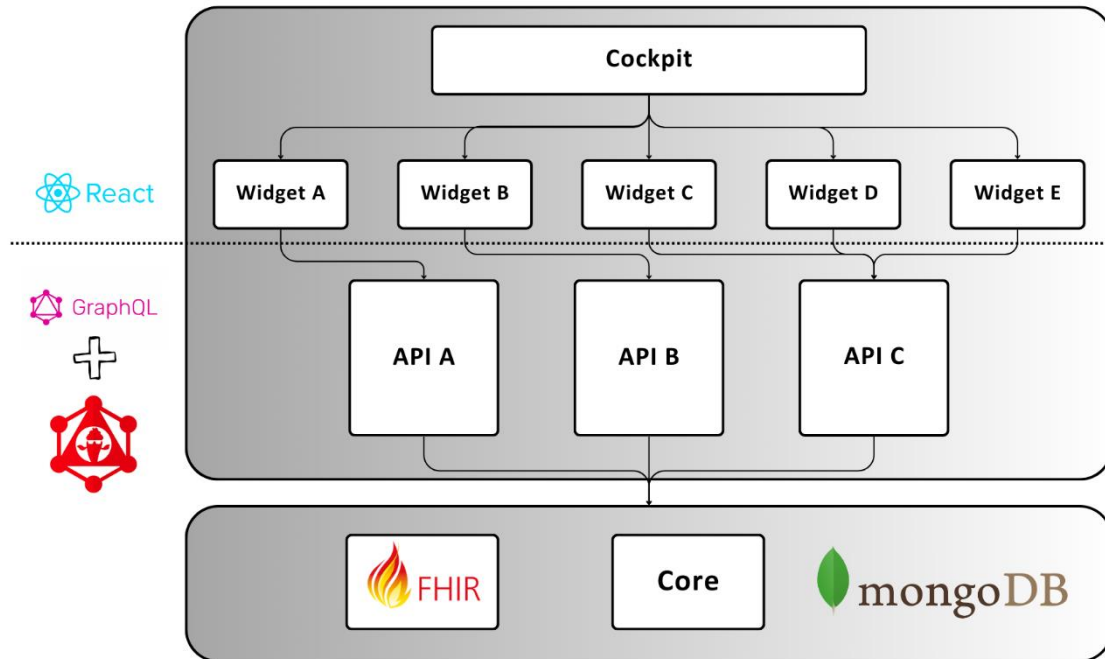


Figura 6 - Atualização da Arquitetura Mozy.

É possível observar que o produto mantém-se em constante mutação e reciclagem. A alteração estrutural procurou diminuir as dimensões da solução em termos de espaço e complexidade, diminuir a produção de material desnecessário e acelerar a velocidade de execução.

4. Descrição Técnica

Findada a parte descritiva dos conteúdos teóricos que foram trabalhos com o produto *Mozy*, serão apresentados neste capítulo a execução dos conteúdos práticos, ou seja, o fluxo de trabalho realizado propriamente dito. Neste capítulo procurar-se-á, de forma fiel, dissecar os diversos momentos de estágio, e apresentar os pontos fundamentais que pautam os mecanismos de trabalho da *Mike Team*.

Para o fazer da forma mais estruturada possível será abordada a metodologia de trabalho, as ferramentas utilizadas, e todos os processos organizacionais da abordagem *Scrum*. Para além disso, será descrito todo o procedimento relacionado com a estrutura de implementação e processo de segurança que são realizados antes da fusão com a solução *main*; passos essenciais para manter solidez num projeto desta magnitude.

Finalmente, será de forma resumida apresentado um “diário de bordo” com todas as atividades no tempo de estágio, e processos realizados ao longo dos meses de permanência nos escritórios da *Glintt*.

4.1. Metodologia – *Agile* e *Scrum*

Uma metodologia de desenvolvimento engloba um conjunto de convenções que é aplicada ao fluxo de trabalho, e que toda a equipa se compromete a seguir e a cumprir como linhas orientadoras. De forma transversal a todas as equipas da Tecnologia os princípios *Agile* são as linhas orientadoras para realização desse trabalho.^[18]

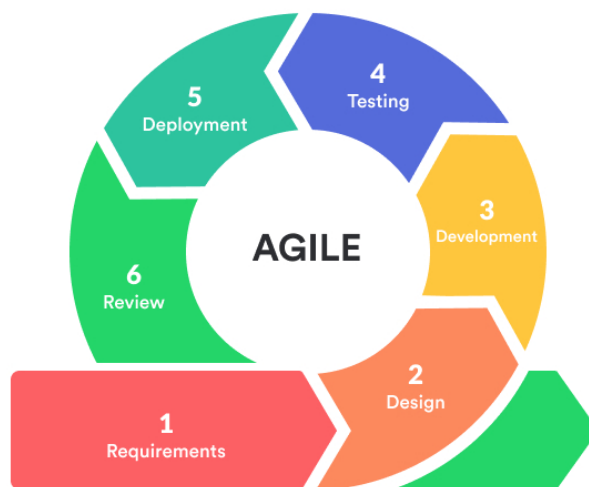


Figura 7 - Agile workflow.

Visto ter integrado a equipa em todas as suas funcionalidades, os valores inerentes à metodologia são aplicados no planeamento funcional da *Mozy*, e pude vivenciar na rotina de trabalho determinados pontos-chave, que são:

- Processo iterativo realizado em **sprints curtas com duas semanas de duração**, com entrega no final de cada uma delas;
- Colaboração contínua e intensa no processo de realização das tarefas, sendo o processo de **entreaajuda e comunicação** constante;
- *Feedback* contínuo do cliente com o *Product Owner* da equipa, o que torna a realização e priorização de tarefas um **processo dinâmico**;
- Equipas com **processos de trabalho independentes**, com distribuição de tarefas discutida e decidida em equipa.

Este conjunto de valores está patenteado pela *framework* utilizada pela equipa, a abordagem **Scrum**.^[19] No que diz respeito a todas equipas de desenvolvimento, a estrutura e planeamento da abordagem *Scrum* reúne as seguintes especificidades:

Papéis	<ul style="list-style-type: none"> - Product Owner, responsável pela apresentação de tarefas; - Scrum Master, responsável pelo planeamento e organização das reuniões da equipa.
Reuniões	<ul style="list-style-type: none"> - Daily Scrum, reunião de aproximadamente 15 minutos, em que se discute a ordem de trabalho de cada um dos elementos da equipa, e o que foi realizado no dia anterior; - Sprint Planning, realizada no início de cada sprint, em que se discutem as tarefas que vão ser realizadas no tempo de duração da sprint; - Sprint Grooming, reunião em que se faz avalia cada uma das tarefas quanto à sua complexidade, e se distribuem as tarefas pela equipa; - Sprint Review, realizada no final da sprint, onde se apresenta o trabalho executado por cada um dos elementos.
Classificação das tarefas	<ul style="list-style-type: none"> - A classificação de tarefas quanto à sua complexidade é discutida em equipa com o <i>Product Owner</i> e feita com o acordo de todos. Para pontuar a mesma usam-se os valores da Sequência de Fibonacci, sendo que quanto maior for o valor maior é a complexidade.

Tabela 4 -Especificidades Scrum - Mike Team.

Para além disto, é importante referir que a resolução de uma tarefa pode ter um ou vários componentes que têm de ser criados ou desenvolvidos, e que passam por um processo com pontos transversais a cada componente, que são:

- **Git** é utilizado como ferramenta de versionamento;^[20]
- Cada componente do sistema tem o seu repositório no **Bitbucket**;^[21]
- Cada repositório segue a estratégia de *Feature Branch*, sendo criado um ramo (*branch*) novo para cada desenvolvimento;
- Cada desenvolvimento, em cada peça, origina um *Pull Request*, que tem de ser aceite pelos membros da equipa, após *Code Review*;
- Um *Pull Request*, quando aceite pelos membros da equipa, passa por um processo de validação de código. A ferramenta **Jenkins** executa este processo, através de uma pipeline, assegurando a qualidade e correta testagem do desenvolvimento.^[22]

4.2. Ferramentas de trabalho – *Confluence* e *Jira*

Para organizar, compartimentar e documentar todas as metodologias de trabalho, foram adotados pela *Glintt* os produtos de software **Atlassian**, que possui na sua gama várias ferramentas que auxiliam em toda a gestão de processos de trabalho; razão pela qual é globalmente utilizada em todo o tipo de empresas e vertentes negociais.^[23]

Uma vez mais, por integrar completamente todas as metodologias da *Mike Team*, tive as credenciais necessárias para aceder a estas ferramentas; que me auxiliaram na resolução das tarefas que me foram indicadas para desenvolver, e que foram um auxílio fundamental para a compreensão do trabalho que estava a ser executado.

Assim, a empresa usa o *Confluence* como ferramenta de armazenamento de documentação para tudo o que está relacionado com os processos trabalho da empresa. Na área do *cockpit* operacional podemos ter acesso material de apoio para a instalação de software necessário para iniciar funções na equipa (*Onboarding*), bem como todo o tipo de esclarecimentos sobre o produto e os seus componentes. Esta área tornou-se um contributo bastante útil na fase inicial do estágio, por ter bastante informação sobre o que ia ser o tema principal do trabalho, e por abranger vários tutoriais e estudos sobre as diversas tecnologias que compõem a solução, sobre as quais nunca tinha estudado.

No *Jira* podemos encontrar aquilo que vulgarmente se chamam de artefactos da *framework Scrum*. Trata-se do quadro-guia de trabalho de cada equipa, onde se apresentam todas as tarefas requisitadas para a construção do produto. Engloba tarefas quanto à sua etiologia (desenvolvimento ou correção de erros encontrados pelo cliente), divide-as quanto à sua integração (na *sprint* atual ou em *backlog*), e estado de realização (*in progress*, *implemented*, *done*).

Dá ao utilizador a possibilidade de visualizar as tarefas a executar, bem como o que está a ser pedido na tarefa; estando a explicação da mesma **sempre** descrita na linguagem **Gherkin**, por forma a uniformizar e automatizar procedimentos.^[24]

Para além da vista **Backlog**, é possível ainda visualizar o fluxo da *sprint* na vista **Active sprints**, onde podemos ver o estado das tarefas que estão a decorrer (*to do*, *in progress*, *in review*, *done*) no modo **Kanban**.

4.3. Mecanismos de Segurança

Um dos pontos fundamentais na execução de uma solução deste grau de complexidade e importância é garantir que as suas funcionalidades e operações intrínsecas não se deixam corromper ou imiscuir por erros de construção de código. Por isso mesmo, o produto *Mozy* tem uma estrutura de testes incorporada nas diversas fases do processo de implementação, que faz com que o processo seja seguro e o código redigido limpo.

Os primeiros testes efetuados à nova solução são **os testes unitários**. Este tipo de testes foca-se, como o nome indica, em testar unidades do código do sistema. Valida o correto funcionamento de uma pequena porção de código, de acordo com o que se espera que produza.

Cada peça da arquitetura tem um conjunto de *scripts* que possuem valores fictícios. Após alterações efetuadas, correndo no terminal o comando **yarn test:unit**, poderemos observar se as alterações efetuadas não tem erros na execução, e se os testes iniciais do componente conseguem dar cobertura ao que foi acrescentado.

```
PASS tests/unit/pages/home/index.test.js
Home Page Tests
  ✓ should match snapshot (9 ms)
  ✓ should handle api fail (276 ms)
  ✓ should handle response with data (285 ms)

-----
File                                     | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
-----
All files                               |    100   |    100   |    100   |    100   |
containers/clinical-data-management     |    100   |    100   |    100   |    100   |
  index.jsx                             |    100   |    100   |    100   |    100   |
containers/clinical-records             |    100   |    100   |    100   |    100   |
  index.jsx                             |    100   |    100   |    100   |    100   |
containers/serum-creatinine             |    100   |    100   |    100   |    100   |
  index.jsx                             |    100   |    100   |    100   |    100   |
containers/utils                         |    100   |    100   |    100   |    100   |
  constants.jsx                         |    100   |    100   |    100   |    100   |
pages/home                              |    100   |    100   |    100   |    100   |
  index.jsx                             |    100   |    100   |    100   |    100   |
state/mutations                         |    100   |    100   |    100   |    100   |
  index.js                              |    100   |    100   |    100   |    100   |
state/queries                           |    100   |    100   |    100   |    100   |
  index.js                              |    100   |    100   |    100   |    100   |
-----
Test Suites: 2 passed, 2 total
Tests:       13 passed, 13 total
Snapshots:   1 passed, 1 total
Time:        31.857 s
Ran all test suites.
Done in 42.29s.
```

Figura 8 - testes unitários com 100% de aprovação.

Seguem-se **os testes de aceitação**, que são feitos exclusivamente por um elemento da equipa. O objetivo dos testes de aceitação é o de verificar se o sistema cumpre os requisitos do cliente. Assim, descrevem o comportamento que o *software* deve ter em determinados cenários. Estes testes são feitos para todos os casos de uso, validando o comportamento dos mesmos.

Para além desta parte de segurança, existem ainda os procedimentos relacionados com a fusão de uma *feature branch* com o *main* do componente, como descrito anteriormente, que são:

- Aprovação de 3 *reviewers* (outros elementos das equipas *Mike* e *Alpha*);
- Aprovação de 2 *default reviewers* (*Team leaders*);
- Execução (*build*) sem falhas na plataforma Jenkins.

No final de todos estes parâmetros da metodologia de trabalho avaliados, a redação do código segue um conjunto de regras apertadas (como o número máximo de caracteres numa linha, ou a colocação de objetos por ordem alfabética) que solidificam a solução, e tornam a sua manutenção mais simples.

4.4. “Diário de Bordo”

Finalizadas todas descrições, torna-se pertinente apresentar cronologicamente as tarefas realizadas no tempo de estágio, a descrição do que foi realizado e a forma como foram completadas as tarefas que me foram sendo entregues. O objetivo será o de verificar o fluxo de trabalho executado, e posteriormente analisar o mesmo face à sua complexidade. Todas as tarefas enunciadas foram executadas com a observação/apoio/tutela do *Buddy* da equipa.

SWCK 33331 – *Therapeutic protocol line information*

Sprint: PI01_04 | Mike Team 1.6

Grau de Dificuldade (segundo Fibonacci): 5

Tempo de Realização: 11 – 13 de março

Tipo de tarefa: Desenvolvimento.

Tipo de colaboração: Observação integral da execução.

Descrição: Como Prescritor

Eu quero visualizar os protocolos na lista de pedidos

Para que possa rapidamente ver os protocolos na prescrição atual e consultar/editar a informação associada aos mesmos.

Atividades realizadas:

Tal como apresenta a descrição, o pretendido era que o utilizador tivesse a oportunidade de prescrever um protocolo terapêutico listado, bem como observar os anteriores, editá-los e registar informação que achasse importante e adicionar à prescrição.

Para realizar esta tarefa foram necessárias intervenções em várias peças da *Mozy* no âmbito da prescrição terapêutica. De uma forma vertical, partindo da camada mais interna para a mais externa, realizaram-se alterações no **core-js** (adicionada nomenclatura em português e inglês, bem como adicionados *mappers* de *GraphQL* para efetuar comunicação de novos métodos entre o *composite* e o *middleware*); no **medication-composite** (criação de métodos de adição ou *update* de prescrição); no **therap-prescription-middleware** (semelhante ao implementado no *composite*, mas com o intuito de receber e enviar a nova informação vinda do *front-end*); e no **therap-prescription-widget** (adicionados novos campos de preenchimento na prescrição de protocolos, tal como pedido na descrição).

No final, foram atualizados os testes unitários por forma a dar cobertura total a todas as alterações, e feitos os testes de aceitação por outro elemento de Qualidade da equipa.

SWCK 32900 – Implement Audit&Logs (simple medicines and therapeutic protocols)

Sprint: PI01_04 | Mike Team 1.6

Grau de Dificuldade (segundo Fibonacci): 5

Tempo de Realização: 18 – 20 de março

Tipo de tarefa: Desenvolvimento.

Tipo de colaboração: Observação integral da execução.

Descrição: Como Utilizador

Eu quero aceder aos *Application Logs*

Para saber quem fez o quê e quando.

Atividades realizadas:

A tarefa efetuada é uma atualização cíclica de processos relacionados com auditoria. Resumidamente o que foi feito foi implementação de novos pontos de “*Audit*” e “*Log*” em vários componentes da *Mozy*, bem como a atualização e remoção de outros implementados anteriormente. Estes métodos, mantidos atualizados, permitem em contexto de desenvolvimento ter a perceção do desempenho do sistema e de rastrear *bugs*, bem como entender o histórico de desenvolvimento de um projeto para efeitos de melhoria e de auditoria estrutural.

No final, foram atualizados os testes unitários para dar cobertura a todas as operações efetuadas.

SWCK 34496 – *prescription-protocol-management-widget* / *add layout buttons to widget and call mutation*

Sprint: PI01_04 | Mike Team 1.7

Grau de Dificuldade (segundo *Fibonacci*): sem pontuação atribuída (subtarefa).

Tempo de Realização: 2 – 3 de Abril

Tipo de tarefa: Desenvolvimento.

Tipo de colaboração: Participação na execução.

Descrição: sem descrição associada (subtarefa).

Atividades realizadas:

Subtarefa inserida numa tarefa que consistia em adicionar botões de seleção rápida de protocolos terapêuticos prescritos a um doente. A tarefa foi construída na ***prescription-protocol-management-widget*** e consistia em adicionar botões de “Ver”, “Editar” e “Cancelar(X)” para cada um dos protocolos contidos na lista do paciente.

Fizeram-se no final, uma vez mais, os testes unitários com a cobertura necessária a este repositório.

Detalhe

PROTOCOLO CARBOPLATINA E VINORELBINA

Parâmetros gerais protocolo

Ciclos Ativos 1-2 Início protocolo 14/04/2023 Ciclo Atual* 3 Data Ciclo Atual* 18/05/2022

Ativação por ☒ Data 03/04/2023 a 03/04/2023 ☐ Ciclo(s)

Tipo de Tratamento seleccionar Linha de Tratamento 1ª Linha

Observações

Grupos (5)

DESCRIÇÃO	EV	CALENDÁRIO
<input checked="" type="checkbox"/> Grupo (A) Ondansetrom 6Mg + Cloreto de sódio 9 mg/mL	EV	C1 D1 Ver Editar X
<input checked="" type="checkbox"/> Grupo B (2) - Doxorubicina 2 mg/ml 10 ml + Cloreto De Sódio 9 Mg/ML	EV	C1 D1 Ver Editar X
<input type="checkbox"/> Grupo C (1) - CAPTOPRIL 25 MG	EV	Ver Editar X
<input type="checkbox"/> Grupo D (2)- DEXAMETASONA 4MG/ ML + CLORETO SOD. 0,9% 100ML	EV	Ver Editar X

Cancelar Outras ações Adicionar Guardar

Figura 9 - *prescription-protocol-management-widget* com as alterações requisitadas.

SWCK 34615 – HPA / Route on widget title not working

Sprint: PI02_04 | Mike Team 2.1

Grau de Dificuldade (segundo *Fibonacci*): 1

Tempo de Realização: 9 – 10 de Abril

Tipo de tarefa: Correção de Defeito reportado pelo Cliente

Tipo de colaboração: Participação na execução.

Descrição: Foi reportado pelo cliente a seguinte situação: “Quando tento selecionar as notas clínicas, devia ser direcionado para o histórico de notas clínicas, mas nada acontece.”

Atividades realizadas:

Atividade realizada no ***backoffice-api*** (componente da nova arquitetura), que consistiu em corrigir a rota que direcionava para a *widget* pretendida pelo utilizador. Esta tarefa não necessitou de alteração aos testes unitários.

SWCK 34605 – Recalculation of doses based on data clinical changes

Sprint: PI02_04 | Mike Team 2.1/2.2

Grau de Dificuldade (segundo *Fibonacci*): 21

Tempo de Realização: 15 – 30 de Abril

Tipo de tarefa: Desenvolvimento.

Tipo de colaboração: Participação na execução.

Descrição: Como Prescritor

Eu quero ter acesso a uma *widget* com o Recálculo de Doses

Para que possa de forma rápida ver o recálculo com base na alteração de dados clínicos.

Atividades realizadas:

Nesta tarefa foi construída uma nova *widget* chamada de ***prescription-doses-recalculation-widget***, com o intuito de dar ao utilizador a possibilidade de visualizar um novo quadro com dosagens de medicamentos prescritos ao doente após uma alteração dos dados do mesmo (peso, altura, creatinina sérica), permitindo ao prescrito ver as duas dosagens (a atual e alterada) e dando a opção de cancelar ou guardar essas mesmas alterações.

Uma vez mais foram efetuadas implementações em várias peças do *software*, nomeadamente; na criação de novas *labels* e respetivas traduções no *core-js*; criação de novos métodos de recálculo de doses, bem como as funcionalidades de editar e salvar esses cálculos, no *medication-composite* e no *pharmacy-api*; e a criação da *widget* anteriormente descrita.

No final, foram atualizados os testes unitários por forma a dar cobertura total a todas as alterações, e feitos os testes de aceitação por outro elemento da equipa.

DESCRIÇÃO	DOSE ATUAL	DOSE ALTERADA
(A) - Ondansetrom	6Mg	12Mg
Fórmula Aplicada: 5 Auc (Fator Alterado)		
(A) - Cloreto De Sódio	9Mg	15Mg
(B) - Doxorrubicina	2Mg	8Mg
(B) - Cloreto De Sódio	9Mg	15Mg
(C) - Captopril	25Mg	31Mg
(D) - Dexametasona	4Mg	10Mg
(D) - Cloreto De Sódio	9Mg	15Mg
Paracetamol	500Mg	1G

Figura 10 - *prescription-doses-recalculation-widget*.

SWCK 35203 – Iteration of the workflow to save the information about the doses

Sprint: PI02_04 | Mike Team 2.1

Grau de Dificuldade (segundo *Fibonacci*): 3

Tempo de Realização: 17 – 24 de abril

Tipo de tarefa: Desenvolvimento.

Tipo de colaboração: Realização tutelada.

Descrição: Como Prescritor

Eu quero assegurar que as informações relativas à dose e respectivas fórmulas estão corretamente salvas no circuito

Para que possa visualizar e editar de forma rápida a informação da fórmula e que ela fica salva no resto do fluxo do trabalho.

Atividades realizadas:

A intervenção presente nesta história tinha o objetivo de assegurar que informações relacionadas com a dose e as fórmulas aplicadas à mesma ficassem salvas no sistema de forma correta e eficaz no *workflow*, informações essas que incluem a validação da terapêutica e a administração por parte da enfermagem. As alterações necessárias para dar resolução e resposta a esta tarefa passaram por trabalhar no *medication-composite*, e alterar os métodos de instrução de dose, dose principal e fórmula; por forma a poder conter dentro (*array*) várias instruções de dose e várias concentrações de dose.

No final foi dada cobertura aos testes unitários necessária.

SWCK 33831 – Consult and insert the diagnose and disease state, associated with the protocol

Sprint: PI02_04 | Mike Team 2.3

Grau de Dificuldade (segundo *Fibonacci*): 13

Tempo de Realização: 7 – 15 de maio

Tipo de tarefa: Desenvolvimento.

Tipo de colaboração: Participação na execução.

Descrição: Como Prescritor

Eu quero ter acesso à informação sobre o diagnóstico associado ao protocolo do paciente

Para que possa associar a escala *ECOG Performance Status*, associar um diagnóstico com protocolo, e complementar esse diagnóstico com um estadio.

Atividades realizadas:

Para realizar as atividades pedidas nesta tarefa foram trabalhadas duas *widgets*, tendo-se adicionado campos de escrita na *clinical-data-widget* e botões de “Dados Clínicos” na *therap-prescription-widget*. Por forma a integrar as novas funcionalidades, criaram-se métodos para obter e atualizar os diagnósticos associados aos protocolos terapêuticos no *medical-records-api*; e finalmente criadas rotas de acesso e novas *labels* no *core-js*. Para além disso, foi necessária

intervenção do **Design System** para a criação de um novo componente que visualmente proporcionasse todas estas *features*.

No final, foram atualizados os testes unitários por forma a dar cobertura total a todas as alterações, e feitos os testes de aceitação por outro elemento da equipa.

Figura 11 - clinical-data-widget.

SWCK 35300 – Insert Treatment Plans using the ‘Add’ option

Sprint: PI02_04 | Mike Team 2.4

Grau de Dificuldade (segundo *Fibonacci*): 5

Tempo de Realização: 21 – 22 de maio

Tipo de tarefa: Desenvolvimento.

Tipo de colaboração: Realização tutelada.

Descrição: Como Prescritor

Eu quero inserir um plano de Tratamento usando a opção ‘Adicionar’ presente na lista de pedidos

Para que possa ver, editar e/ou inserir novos planos de Tratamento.

Atividades realizadas:

O plano inicial desta atividade passava por adicionar funcionalidades a um botão da *widget therap-prescription-widget*. O processo passava por criar no botão “Adicionar” um sistema de funcionalidades “*dropdown*” que direcionava o utilizador para o “Plano de Tratamento”, “Terapêutica” e “Atos”. No entanto, após análise da *widget* e consulta do *Design System*, percebeu-se que não existia a possibilidade de numa *widget* de 3 botões (valor máximo do sistema) existirem 2 com funcionalidade “*dropdown*”. Por esse motivo, a tarefa foi adiada até nova indicação superior.

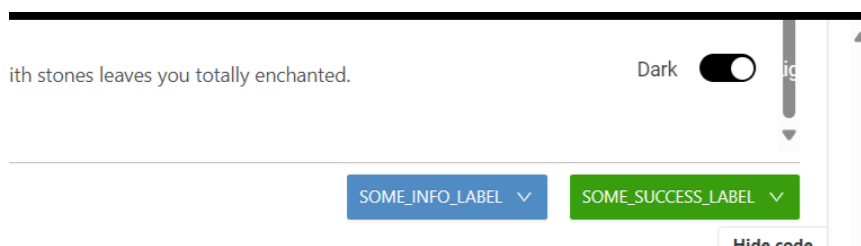


Figura 12 - Solução do Design System existente.

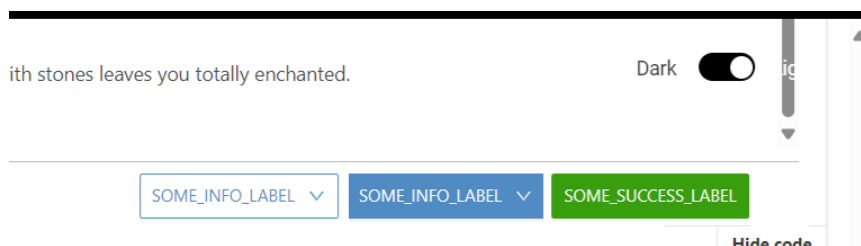


Figura 13 - Solução do Design System pretendida (esboço).

5. Conclusões

5.1. Resumo do Relatório

De um modo geral, o relatório apresenta todos os momentos importantes na realização e execução deste estágio. Após ter sido feita a receção no seio da empresa, houve tempo numa fase inicial para perceber o âmbito de atuação da *Glantt*, bem como explorar o produto no qual ia trabalhar durante o tempo do estágio: a *Mozy*.

Para além disso, o acesso a todos os tutoriais de integração às tecnologias adotadas, e a explicação teórica do *software*, permitiu cimentar os conceitos em volta deste tipo de produtos, por forma a compreender de uma forma mais abrangente o real propósito dos mesmos. Para apoiar ainda mais a construção deste documento, foi feita uma pesquisa sobre produtos equivalentes no mercado.

De seguida, procedeu-se à compreensão da arquitetura *Mozy*, e dos seus componentes, para poder facilitar a integração no fluxo de trabalho da *Mike Team*, equipa que me recebeu na execução do estágio, e com a qual pude partilhar a experiência de todas as tarefas que normalmente se encontram presentes no fluxo de trabalho de uma equipa de desenvolvimento.

Nessa fase, e tal como foi aqui relatado, pude ajudar na realização de várias tarefas presentes nas sprints quinzenais da equipa, que se rege pela metodologia de trabalho *Agile*. Fui integrado em todas as reuniões da equipa, e a partir da minha integração na mesma, fui realizando tarefas sob a forma de observação, participação ou execução tutelada.

5.2. Análise qualitativa dos objetivos iniciais

Nesta fase do relatório torna-se importante tirar ilações do caminho percorrido por forma a poder potencializar toda a aprendizagem que o tempo de estágio nos trouxe. Para que seja feita essa análise, a melhor maneira será a de analisar os objetivos a alcançar, elencados na nota introdutória deste relatório; e verificar a sua correspondência com o que foi realmente realizado, avaliando as dificuldades na realização.

Em primeiro lugar, voltando-me para as principais razões que me levaram a escolher a *Glantt* como local de estágio, posso afirmar que a escolha não poderia ter sido a melhor na minha opinião. Olhando para as razões de escolha do local de estágio pude constatar que a minha experiência na área da saúde possibilitou uma melhor compreensão do que era pedido na realização das tarefas (por estar dentro do contexto do setor), o que permitiu que houvessem menos preocupações de integração nessa matéria.

Para além disso, face ao que foi desenvolvido, houve uma grande aplicabilidade de conceitos dados em várias disciplinas da estrutura curricular da Licenciatura, desde o *frontend* até ao

backend; o que enriquece a experiência e lhe confere elegibilidade, apesar de não abordar o conceito de Jogos Digitais *per si*.

Finalmente, a visão macro de como é trabalhar num meio de desenvolvimento empresarial superou a expectativa inicial. Apesar de almejar integrar todas as funcionalidades de um *Developer* no processo de execução deste estágio, não o esperava fazer desde o início. De fato, foi isso que aconteceu desde o momento em que integrei a *Mike Team*, passando (de forma acompanhada) a executar toda e qualquer tarefa de desenvolvimento que fosse proposta. Isso torna-se possível pela forma acompanhada com que a equipa desenvolve o produto, havendo uma grande predisposição de entreadajuda no seio da equipa.

Voltando-me agora para a análise do fluxo de trabalho propriamente dito, e olhando para o que inicialmente tinha projetado, posso concluir que existiram diferenças, não por falta de nenhuma das partes, apenas porque o fluxo de trabalho em causa tem características diferentes do que inicialmente tinha sido pensado. Todos os momentos de evolução no estágio foram realizados com uma elevada taxa de aproveitamento (Integração – Observação – Participação – Execução); não seguindo, no entanto, esta evolução estandardizada que inicialmente dei como a ideal. Uma vez mais referindo o processo de trabalho acompanhado no seio da *Mike Team*, as duas últimas fases das *milestones* apresentadas misturam-se muitas vezes; quer pela prestabilidade dos elementos da equipa, quer porque algumas tarefas têm um grau de complexidade que as faz logo à partida terem de ser realizadas em participação conjunta. Posso então concluir que todas as fases foram realizadas com sucesso, mas não de forma sequencial.

Finalmente, e em linha de pensamento com o exposto acima, em termos palpáveis, não foi possível seguir uma linha de progressão na complexidade das tarefas, apenas e só porque não é assim que a *Mike Team* trabalha.

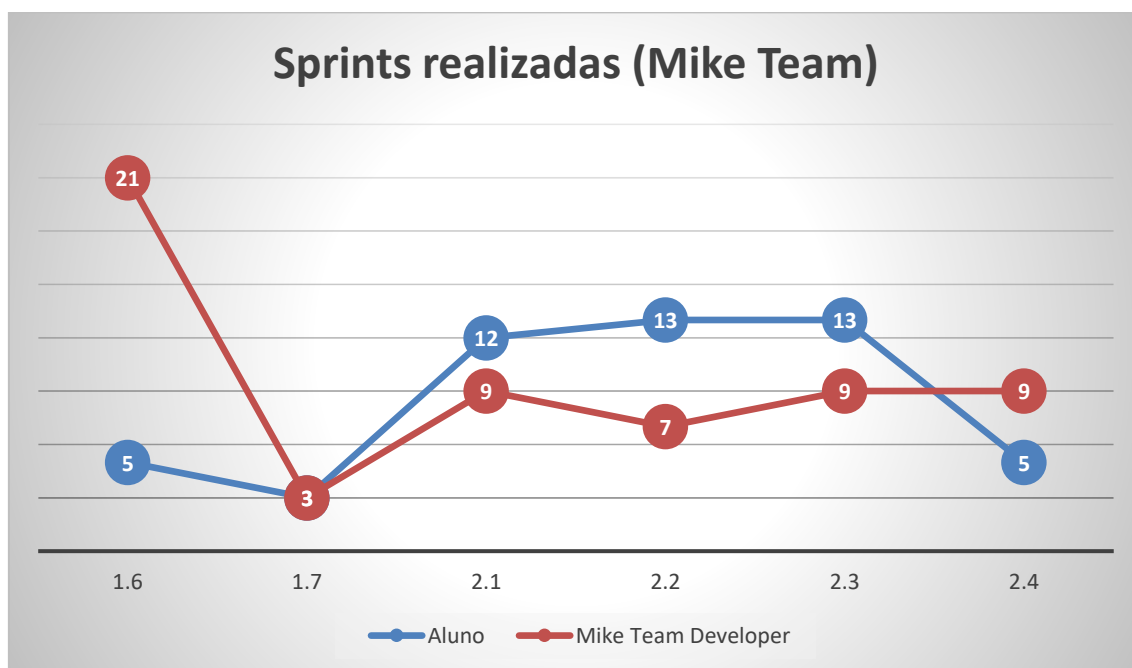


Figura 14 - Evolução da complexidade das tarefas.

O gráfico apresentado na Figura 14 demonstra isso mesmo. Nele é possível observar o somatório de dificuldade nas tarefas realizadas ao longo da *sprint* por mim (nas vertentes de observação, participação ou execução tutelada) e de um elemento de *Mike Team* escolhido aleatoriamente para esta análise.

Deste modo é possível concluir que nos processos de trabalho da equipa não existe uma regra evolutiva na execução de tarefas. As mesmas são distribuídas nas reuniões de *grooming*, e de uma forma relativamente aleatória. Isto permite enriquecer o *Developer* em várias áreas do processo de desenvolvimento do produto, porque dá uma maior versatilidade na execução dos vários processos de desenvolvimento (*frontend* e *backend*), bem como permite um maior e constante reconhecimento dos diversos componentes da *Mozy*, que cria automatismos na resolução de problemas e aumenta o conhecimento global de toda a arquitetura.

Em último lugar, eno seguimento de tudo o que foi dito anteriormente, é possível afirmar que o acompanhamento foi máximo, e muito bem estruturado. A designação de um elemento fixo para acompanhar o meu estágio (*Buddy*) foi benéfica para automatizar processos de trabalho de forma evolutiva, e reduziu os momentos em que não existiu apoio ou ausência de tarefas. Todo o processo de crescimento no âmago da empresa foi cuidado, planeado, e com o ritmo apropriado para que no fim se conclua que tive um crescimento sustentado e sem grandes problemas por resolver.

5.3. Limitações e trabalho futuro

Nesta secção pretende-se abordar os pontos menos positivos ao longo do estágio. De fato, recuando e olhando para o trabalho exercido, há muito pouco para analisar neste tópico em específico, a não ser apresentar pequenas questões que possam ter desacelerado o processo de crescimento na integração funcional na equipa, ambas de carácter estrutural que são incontornáveis.

A primeira questão prende-se com o fato de ter três dias de estágio por semana, o que em termos de fluxo de trabalho corta o processo de acompanhamento, e aliado ao fato de quinzenalmente ser imperativo apresentar resultados por forma a manter a boa rentabilidade da equipa, fez com que muitas das tarefas executadas não pudessem ser feitas de forma independente. Para além disso, em termos de curva evolutiva, o tempo de estágio infelizmente revela-se curto para que pudesse atingir os objetivos pessoais que pretendia para a minha evolução na área.

Outro ponto prende-se precisamente com a questão da entrega de trabalho. Apesar de entender que consegui de forma positiva implementar soluções para a equipa, senti que a pressão da entrega em determinados momentos não permitiu uma maior variedade de execuções. Tal como entendo que um dos maiores ganhos que obtive na realização do estágio foi o de criar ferramentas de automatismo que me permitiram diminuir as dificuldades na execução, estou também consciente que a equipa tem os próprios automatismos e momentos

de aceleração na entrega de tarefas, e que por estar numa fase de aprendizagem prematura, nem sempre consegui acompanhar.

Finalmente, a execução do estágio vem alargar horizontes face ao que podem ser as saídas profissionais futuras. Posso concluir que pela aprendizagem recebida pela estrutura curricular desta Licenciatura a saída não tem necessariamente de ser para o campo de desenvolvimento de Jogos Digitais. O *core* dado pelos conteúdos programáticos ao longo destes três anos permite com solidez dizer que procurar empregabilidade nas diversas áreas de desenvolvimento é perfeitamente exequível e alcançável, e os objetivos atingidos neste estágio são sem dúvida prova disso mesmo.

5.4. Apreciações finais

A realização do estágio nos escritórios da *Glintt Global* não só alcançou as metas por mim traçadas inicialmente, como as superou em vários capítulos. Não deixando passar a minha experiência profissional em vão (apesar de ser numa área diferente) só tenho a deixar elogios a todo o processo de estruturação deste estágio, que desde o primeiro contato foi cuidado e respeitador. Existiu sempre o cuidado em fazer uma anamnese do que tinha em termos de conhecimentos adquiridos, para que fosse possível enquadrar-me numa equipa e numa solução que fizesse com que o meu processo de aprendizagem fosse positivo; e depois o acompanhamento no seio da equipa foi sempre o mais cuidado possível.

Desta forma, posso dizer: que sinto que contribui positivamente para o incremento de trabalho na *Mike Team*; que consegui criar um conjunto de automatismos que me serão úteis no futuro para aplicar nos meus processos de trabalho; e que saio com uma visão acrescida do que é o trabalho de equipa nesta área.

A experiência foi enriquecedora e bem-sucedida, tendo, na minha opinião, concretizado os pressupostos a que este tipo de ensinamento obriga, bem como os meus pessoais. As expectativas trazidas para o campo de estágio na área de aplicação de conhecimentos anteriores foram concretizadas e excedidas, e a passagem que precisava de ter para poder fundir o mundo profissional da Enfermagem com o mundo profissional da Programação foi melhor do que pensei ser possível, visto ter mais paralelismo do que equacionei.

Bibliografia

- [1] <https://www.glinttglobal.com>, <consultado a 29-04-2024>.
- [2] <https://www.healthcareitnews.com/news/cpoe-cuts-medication-errors-study-shows>, <consultado a 02-05-2024>
- [3] <https://jelvix.com/blog/computerized-provider-order-entry>, <consultado a 02-05-2024>.
- [4] Rouayroux, N. et al. (2018) *Medication prescribing errors: a pre- and post-computerized physician order entry retrospective study*, International Journal of Clinical Pharmacy, 41(1), pp. 228–236. Disponível para *download*: <https://link.springer.com/article/10.1007/s11096-018-0747-0>.
- [5] <https://www.altexsoft.com/blog/cpoe-systems-and-electronic-prescribing-software>, <consultado a 02-05-2024>.
- [6] <https://www.businessnewsdaily.com/10914-best-electronic-health-records-systems.html>, <04-05-2024
- [7] <https://globalcare.glintt.com>, <consultado a 08-05-2024>.
- [8] <https://www.sisbit.pt/portfolio/sigehp>, <consultado a 08-05-2024>.
- [9] <https://www.drchrono.com/about>, <consultado a 10-05-2024>.
- [10] <https://www.carecloud.com/solutions>, <consultado a 10-05-2024>.
- [11] <https://www.databridgemarketresearch.com/reports/global-computerized-physician-order-entry-cpoe-market>, <consultado a 11-05-2024>.
- [12] <https://react.dev>, <consultado a 12-05-2024>.
- [13] <https://grpc.io/about>, <consultado a 12-05-2024>.
- [14] <https://graphql.org>, <consultado a 12-05-2024>.
- [15] <https://hl7.org/fhir/R4/summary.html>, <consultado a 14-05-2024>.
- [16] <https://www.educative.io/blog/crud-operations>, <consultado a 14-05-2024>.
- [17] <https://chillicream.com/docs/hotchocolate/v13>, <consultado a 14-05-2024>.
- [18] <https://www.agilealliance.org/agile101>, <consultado a 18-05-2024>.
- [19] <https://www.scrum.org/learning-series/what-is-scrum/what-is-scrum>, <consultado a 18-05-2024>.

[20] <https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F>, <consultado a 19-05-2024>.

[21] <https://bitbucket.org/product/guides/getting-started/overview#a-brief-overview-of-bitbucket>, <consultado a 19-05-2024>.

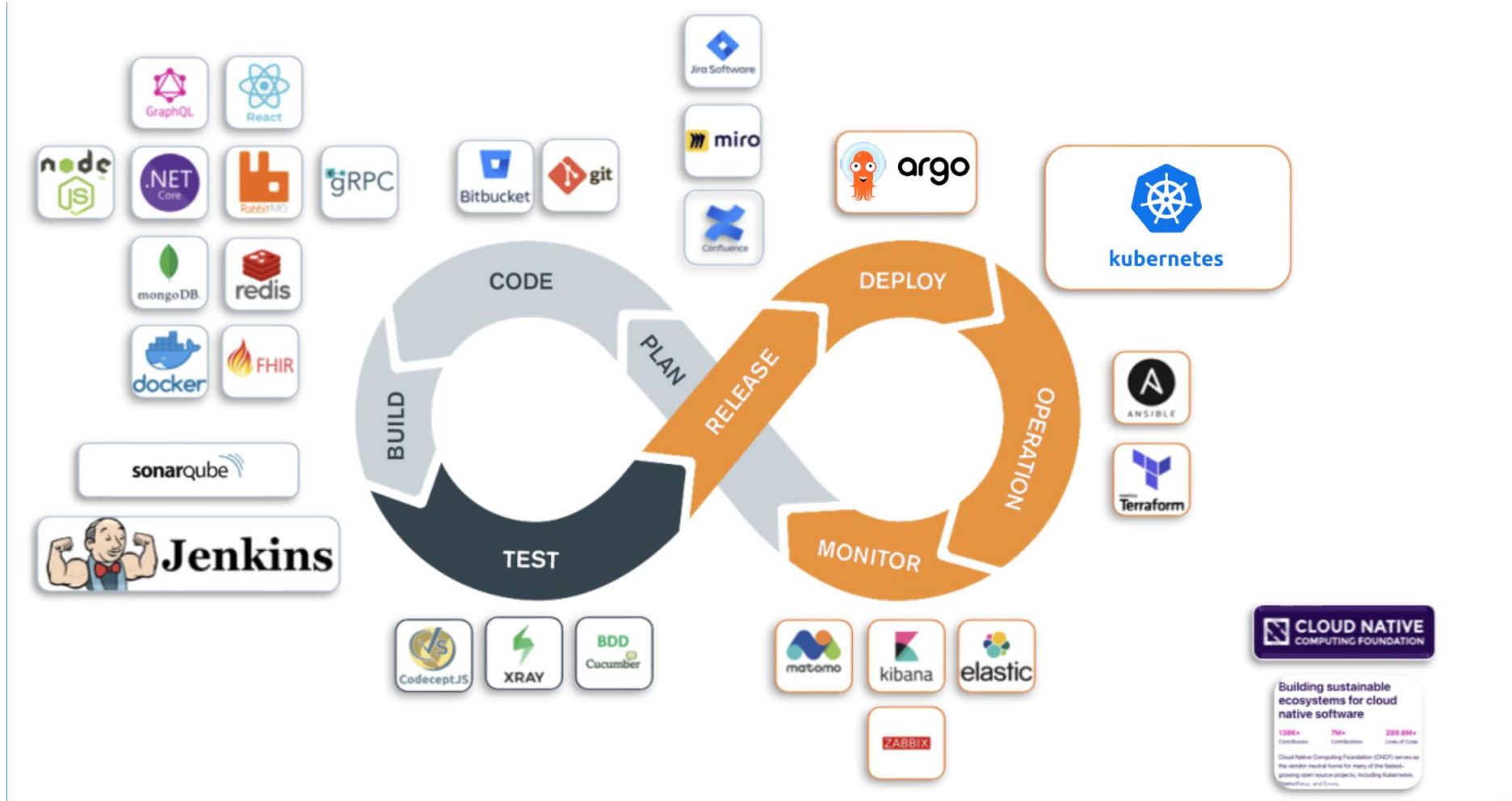
[22] <https://www.jenkins.io>, <consultado a 19-05-2024>.

[23] <https://www.atlassian.com>, <consultado a 26-05-2024>.

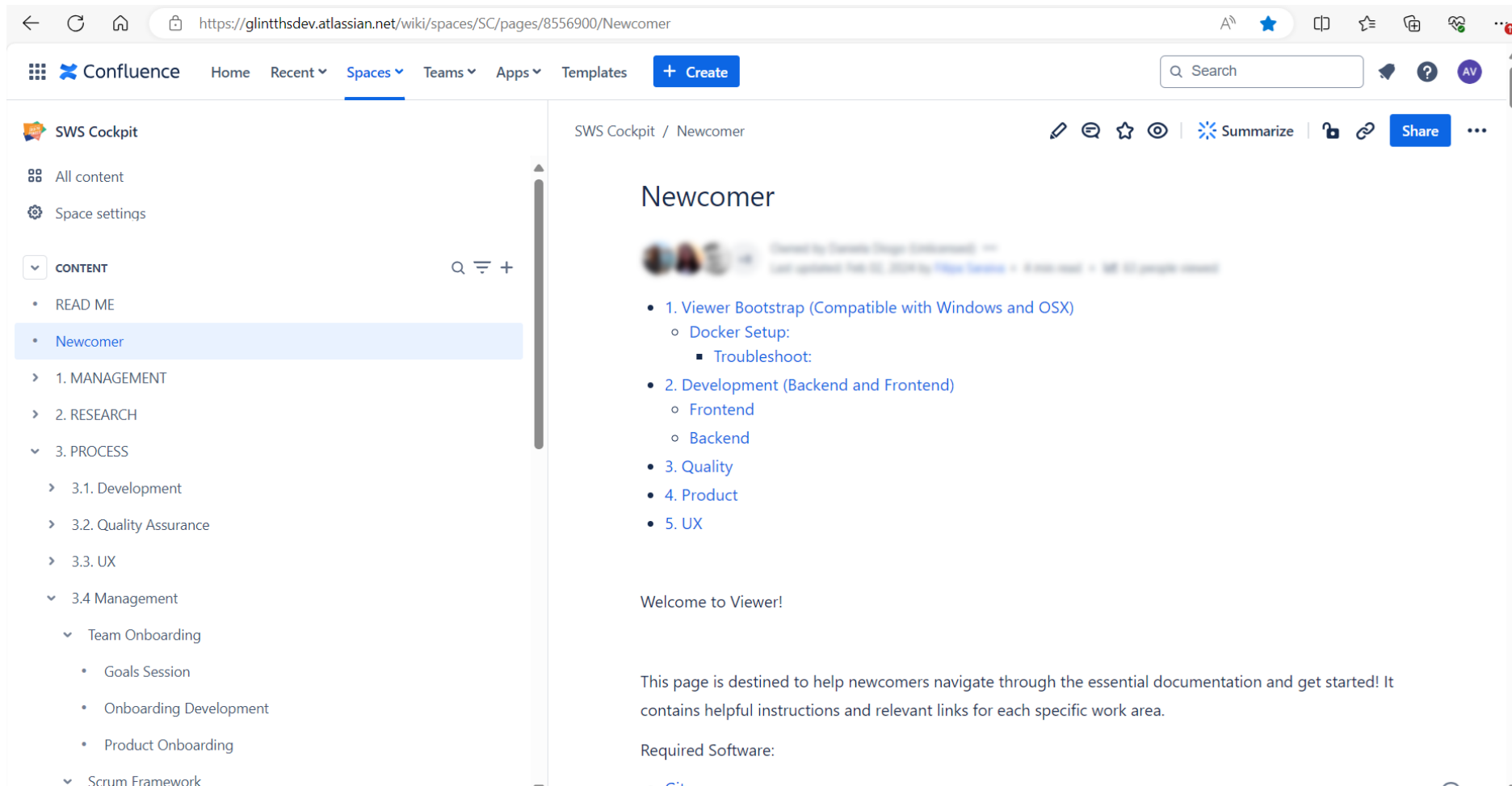
[24] <https://cucumber.io/docs/gherkin/reference>, <consultado a 25-05-2024>.

Anexos

Anexo I – Workflow das equipas de Desenvolvimento pela metodologia *Agile* (representação de todas as tecnologias utilizadas no processo contínuo)



Anexo II – Organização do *SWS Cockpit* (Confluence)



The screenshot displays the Confluence interface for the 'SWS Cockpit' space. The left sidebar shows the 'CONTENT' section with a tree structure: 'READ ME', 'Newcomer' (highlighted), '1. MANAGEMENT', '2. RESEARCH', '3. PROCESS' (expanded), '3.1. Development', '3.2. Quality Assurance', '3.3. UX', '3.4 Management' (expanded), 'Team Onboarding' (expanded), 'Goals Session', 'Onboarding Development', 'Product Onboarding', and 'Scrum Framework'. The main content area shows the 'Newcomer' page, which includes a list of topics: '1. Viewer Bootstrap (Compatible with Windows and OSX)' (with sub-items 'Docker Setup' and 'Troubleshoot'), '2. Development (Backend and Frontend)' (with sub-items 'Frontend' and 'Backend'), '3. Quality', '4. Product', and '5. UX'. Below the list, there is a 'Welcome to Viewer!' section and a paragraph stating: 'This page is destined to help newcomers navigate through the essential documentation and get started! It contains helpful instructions and relevant links for each specific work area.' The 'Required Software:' section is partially visible at the bottom.

Anexo III – SWS Cockpit/Mike Team – exemplo de Kanban (Timeline)

PI02_04 | Mike Team 2.3

3 days

Search

Epic Label Type Quick filters

GROUP BY Stories Insights View settings

TO DO 1

IN PROGRESS 8

IN REVIEW 1 MAX 2

DONE

SWCK-33943 Dev | Presentation of the formula zone in the widget (9 subtasks) CPOE TREATMENT PLANS | THERA... IN PROGRESS

Functional tests
SWCK-35065

Pharmacy-API | getPrescriptionMedicationData return the dosage calculo and protocolated factor
SWCK-34973

Prescription-Management-Wwidget | Acceptance tests
SWCK-35064

Prescription-Management-Widget | create new component to calculate dose and create new setting
SWCK-34879

Proto-bufs | include activity definitions in GetPrescriptionMedicationDataResponseMs
SWCK-34974

Core-js | new mappers in activityDefinition to return dosage type and quantity
SWCK-34975

Medication-composite | include reference to activitydefinition in medicationRequests (ConvertPlanDefinitionToRequestGroup)
SWCK-34976

Anexo IV – SWS Cockpit/Mike Team – exemplo de Sprint (Backlog)

Projects / SWS Cockpit / Mike Team

Backlog

Q















Version ▾
Epic ▾
Label ▾
Type ▾
More ▾

Insights

▾
PI01_04 | Mike Team 1.6
11 Mar – 22 Mar (6 issues)

<input type="checkbox"/>	>	SWCK-33331 Dev 3. Therapeutic protocol line information		PROTOCOL PRESCRIPTI...	IMPLEMENTED ▾	5
	>	SWCK-33812 Dev Protocol header - access to the Therapeutic Protocol Detail		PROTOCOL PRESCRIPTI...	IMPLEMENTED ▾	2
	>	SWCK-33813 Dev Protocol header - information about the cycles and activation		PROTOCOL PRESCRIPTI...	IN PROGRESS ▾	21
	>	SWCK-33814 Dev Protocol header - additional information about the treatment		PROTOCOL PRESCRIPTI...	IN PROGRESS ▾	8
	>	SWCK-33816 Dev Table - description of the group, that expands and show the medicines		PROTOCOL PRESCRIPTI...	IN PROGRESS ▾	13
	>	SWCK-33817 Dev Table - options presented in the group line		PROTOCOL PRESCRIPTI...	SELECTED FOR DEVELOP...	21

Anexo V – SWS Cockpit/Mike Team – Story e Subtasks

▼  SWCK-33331 Dev 3. Therapeutic protocol line information	 PROTOCOL PRESCRIPTI...	IMPLEMENTED ▼
 SWCK-34143 functional tests		IMPLEMENTED ▼
 SWCK-34179 therap-prescription-widget acceptance tests		IMPLEMENTED ▼
 SWCK-34138 therap-prescription-middleware update createOrUpdatePrescriptionDraft to receive de list of protoc...		IMPLEMENTED ▼
 SWCK-34139 protoBufs update CreateOrUpdatePrescriptionDraft contract to receive list os protocols to remove		IMPLEMENTED ▼
 SWCK-34140 medication composite update CreateOrUpdatePrescriptionDraft method to remove protocols from p...		IMPLEMENTED ▼
 SWCK-34141 therap-prescription-middleware unit tests		DONE ▼
 SWCK-34142 medication-composite unit tests		IMPLEMENTED ▼
 SWCK-34144 therap-prescription-widget unit tests		IMPLEMENTED ▼
 SWCK-34146 therap-prescription-widget update ProtocolItem component with new field/s		IMPLEMENTED ▼
 SWCK-34309 core-js create getReplaces and getStatus mappers		IMPLEMENTED ▼
 SWCK-34330 core-js unit tests		IMPLEMENTED ▼
 SWCK-34344 core-js create labels Active and Inactive		IMPLEMENTED ▼

Anexo VI – SWS Cockpit/Mike Team – Descrição de Story

Projects

Filters

Dashboards

Teams

Plans

Apps

Create

Projects /

MOZY /

SWCK-33941 /

SWCK-34605

Dev | Recalculation of doses based on data clinical changes

Attach

Create subtask

Link issue

Create

Test Coverage

SmartDraw Connector

Description • Unsaved changes

As a Prescriber

I want to access the Recalculation of doses widget

So that I can quickly view the recalculation of doses based on data on clinical changes

Scenario 1:

Given I want to prescribe therapeutic protocols

When the user clicks on 'Edit'/'Editar' to access the **Therapeutic Protocol Detail** and there is an automatic update in the weight/height of the patient

Then should exist the 'Dose Recalculation widget' to view the medicines with recalculated doses

Notes:

Name of the new widget: prescription-doses-recalculation-widget

The new widget should open on the right side of the screen, and on the left side, the Therapeutic Protocol Detail