UNIVERSIDADE
BEIRA INTERIOR

A process model for quality in use evaluation on clinical decision support systems

Leonice Maria Reis de Souza Pereira

Tese para obtenção do grau de Doutor em
Engenharia Informática
(3º ciclo de estudos)

Orientador: Prof. Doutor Nuno Gonçalo Coelho Costa Pombo

Júri
Prof. Doutor Joaquim Mateus Paulo Serra
Prof. Doutor Rui Filipe Lima Maranhão de Abreu
Prof. Doutor Nuno Filipe Alves Gaiola Castela
Prof. Doutor Nuno Manuel Garcia dos Santos
Prof. Doutor Carlos Augusto da Silva Cunha
Prof. Doutor Sérgio Luís Proença Duarte Guerreiro
Prof. Doutor Nuno Gonçalo Coelho Costa Pombo

Covilhã, julho de 2022

Dedication

Dedico este trabalho a todos que, sob lágrimas e sorrisos, vibraram por mim, torceram por mim, oraram por mim. Obrigada, família linda e amigos do coração! Agradeço a Deus por ter vocês em minha vida.

Um abraço especial aos colegas de laboratório (ALLAB), Virginie Felizardo, Henriques Zacarias, Igor Matias, Paulo Silva, Geraldo Cangondo, Pedro Moreira, Melaku Girma e outros tantos com quem tive o prazer de trabalhar.

Aos queridos Prof. Dr. Nuno Pombo e Prof. Dr. Sofia Ouhbi, meus sinceros agradecimentos! Não teria chegado tão longe se não pudesse contar com pessoas tão competentes, dedicadas e maravilhosas como vocês.

Ao Prof. Dr. Nuno Garcia, meu caro amigo, por sua presença ao meu lado nos momentos complicados e por seus conselhos.

Ao meu parceiro por muitos anos, José Roberto, que me apoiou e incentivou na busca da realização dos meus projetos e sonhos.

Aos meus filhos Matheus e Vinicius, minha conquista é dedicada a vocês! Amo vocês, "daqui ao infinito", incondicionalmente.

I dedicate this work to all who, under tears and smiles, vibrated for me, cheered for me, prayed for me. Thank you, beautiful family and friends of the heart! I thank God for having you in my life.

A special hug to laboratory colleagues (ALLAB), Virginie Felizardo, Henriques Zacarias, Igor Matias, Geraldo Cangondo, Pedro Moreira, Melaku Girma, and many others with whom I had the pleasure to work with.

Prof. Dr Nuno Pombo and Prof. Dra. Sofia Ouhbi, my sincere thanks! I wouldn't have come so far if I were not supported by competent, dedicated and wonderful people like you.

To Prof. Dr. Nuno Garcia, my dear friend, for your presence at my side in difficult times and for your advice.

To my partner of many years, José Roberto, who supported and encouraged me to seek my projects and dreams.

To my sons Matheus and Vinicius, my achievement is dedicated to you! I love you "from here to infinity" unconditionally.

Acknowledgements

Thesis prepared at the Ambient Assisted Living Computing and Telecommunications Laboratory (ALLab), Instituto de Telecomunicações, Universidade da Beira Interior, and submitted to the Universidade da Beira Interior for defense in a public examination session.

The research has been partially funded by the FCT/MCTES through national funds, and when applicable, co-funded EU funds under the project UIDB/EEA/50008/2020 and Operação Centro 01-0145-FEDER-000019 – C4 – Centro de Competências em Cloud Computing, co-financed by the Programa Operacional Regional do Centro (CENTRO 2020), through the Sistema de Apoio à Investigação Científica e Tecnológica – Programas Integrados de IC&DT. I would also like to acknowledge the contribution of the COST Action IC1303: AAPELE—Archi- tectures, Algorithms and Protocols for Enhanced Living Environments and COST Action CA16226; SHELD-ON—Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology).

"O que distingue as mentes verdadeiramente originais
não é que elas sejam as primeiras a ver algo novo,
mas sim que são capazes de ver como novo o que é
antigo, conhecido, visto e desprezado por todos."
Friedrich Wilhelm Nietzsche

"Os ideais se parecem com estrelas no sentido de que
nunca os alcançamos, mas como navegadores,
dirigimos o curso de nossas vidas com eles."
Albert Schweitzer

"O tamanho dos seus sonhos deve sempre exceder
a sua capacidade de alcançá-los.
Se os seus sonhos não te assustam,
eles não são grandes o suficiente."
Ellen Johnson-Sirleaf"

List of publications

Articles included in the thesis resulting from this doctoral research programme

[1] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, N. Garcia, Clinical decision support systems for chronic diseases: A Systematic literature review, Comput. Methods Programs Biomed. 195 (2020) 105565. https://doi.org/10.1016/j.cmpb.2020.105565.

[2] L. Souza-Pereira, S. Ouhbi, N. Pombo, Quality-in-use characteristics for clinical decision support system assessment, Comput. Methods Programs Biomed. 207 (2021) 106169. https://doi.org/https://doi.org/10.1016/j.cmpb.2021.106169

[3] L. Souza-Pereira, S. Ouhbi, N. Pombo, A process model for quality in use evaluation of clinical decision support systems, J. Biomed. Inform. 123 (2021) 103917. https://doi.org/10.1016/j.jbi.2021.103917.

[4] L. Souza-Pereira, N. Pombo, S. Ouhbi, Software quality: application of a process model for quality-in-use assessment, Journal of King Saud University Computer and Information Sciences. 34, (2022). https://doi.org/10.1016/j.jksuci.2022.03.031

Resumo

Desenvolver ou adquirir software é um investimento caro e precisa ser justificado. Além de útil, o sistema deve ser confiável, eficiente e, entre outras características, atender às expectativas dos usuários [1, 2].

Não seria diferente no caso de um sistema de apoio à decisão clínica (CDSS, acrônimo em inglês), sistemas desenvolvidos para apoiar médicos e outros profissionais de saúde na tomada de uma decisão médica [3].

CDSSs são elaborados dentro de um contexto clínico, seguindo *guidelines* com propósitos variados, sejam para diagnósticos [4, 5, 6], acompanhamento do paciente [7, 8, 9], na prevenção [10] e tratamento de doenças [11, 12].

No entanto, apesar de todo os benefícios oferecidos por um CDSS, sua aceitação na área médica ainda é motivo de debate [13, 14]. Essa aceitação está ligada à percepção do usuário final, como

1) a facilidade de uso e utilidade do sistema;

2) a qualidade dos resultados produzidos e sua confiabilidade [14];

3) a acessibilidade contextual do sistema, muitas vezes não incluída na rotina e no fluxo de trabalho do profissional de saúde, e

4) o fato de muitos CDSSs não estarem integrados aos sistemas existentes [15].

Uma forma de estender o uso de CDSSs e disseminar suas contribuições positivas entre os profissionais de saúde é garantir a confiabilidade de seus resultados e a satisfação do usuário-final. Para tal deve-se seguir as melhores práticas da engenharia de software (SE, acrônimo em inglês) em sua concepção [16]. Isso implica em preocupar-se com a qualidade do sistema tanto no processo do projeto e desenvolvimento quanto em sua efetiva utilização.

Uma forma de certificar se um software obedece a essa premissa é realizando avaliações de qualidade. Avaliar a qualidade do software é medir suas características e subcaracterísticas de qualidade.

Para uma melhor estruturação desta medição foram desenvolvidos séries de padrões internacionais como guidelines de avaliação de qualidade de produtos de software. A série mais recente trata-se da ISO/IEC 25000 System and Software Quality Requirements and Evaluation (SQUARE) [17]. Dois padrões desta série foram abordadas nesta tese, sendo **1)** o modelos de qualidade de software e sistemas (ISO/IEC 25010) [18], no qual trabalhamos especificamente com o modelo de qualidade em uso, e **2)** o padrão de medição da qualidade em uso (ISO/IEC 25022) [19].

Qualidade em uso é o foco desta tese, através de sua avaliação no contexto de utilização de um CDSS.

O Modelo de qualidade em uso trata da qualidade do software quando em execução, referindo-se ao resultado da interação dos usuários e o software em um cenário específico.

Este modelo é composto de cinco características de qualidade:

- Eficácia (ou efetividade) - esta característica representa o nível de precisão e completude com que os usuários alcançam os objetivos específicos, durante a utilização do sistema ou produto de software;
- Eficiência - sua medição representa o nível de eficácia alcançada em relação aos recursos consumidos para o alcance das metas;
- Satisfação - trata do quanto as necessidades do usuário são satisfeitas dentro de um determinado contexto de uso do sistema ou produto de software. Esta característica é composta pelas subcaracterísticas Utilidade, Confiança, Prazer e Conforto do usuário em relação ao sistema;
- Livre de risco - trata do grau em que a qualidade de um sistema ou produto permite mitigar ou evitar riscos potenciais à vida humana, à situação econômica, à saúde ou ao meio ambiente, sendo estas suas três subcaracterísticas;
- Cobertura de contexto - trata do uso do sistema em todos os contextos específicos e/ou em contextos além dos inicialmente identificados, sendo composta pelas subcaracterísticas completude de contexto e flexibilidade do sistema.

Assim, para se medir a qualidade de um CDSS deve-se considerar tanto o contexto de utilização quanto a escolha da característica e subcaracterística que melhor condizem ao propósito da avaliação [20].

De acordo com Harrison *et al.* [21], Eficácia, Eficiência e Satisfação são considerados os principais critérios a serem avaliados para refletir a qualidade de uso. Tais características de qualidades em uso refletem o atendimento das necessidades e expectativas dos usuários dos sistemas, em especial ao usuário primário ou final, uma vez que estão diretamente relacionadas com a experiência do usuário. O modelo de qualidade em uso fornece uma contribuição poderosa para a prática de avaliar um sistema e determinar sua qualidade.

Como contribuição, propusemos um modelo de processo para avaliação de qualidade em uso de um CDSS através da medição, a priori, de duas características de qualidade - satisfação e eficiência. Acreditamos que tais características são importantes na avaliação de um CDSS devido estreita relação destas com a experiência do usuário-final e a usabilidade do sistema. Assim, quando mensuradas, tais características podem corroborar com a qualidade do CDSS e mitigar a não utilização e não aceitação desse tipo de software.

Nosso modelo proposto é definido por cinco (5) fases, a saber: **1)** Identificação de cenário e contexto de uso do sistema, **2)** seleção das medidas, métricas e métodos para mensurar as características, **3)** a medição da qualidade, **4)** a análise dos valores encontrados na medição e **5)** a apresentação dos resultados obtidos.

O resultado da aplicação do modelo de processo traduz-se em um conjunto de informações que nortearão um melhoramento do software, caso a medição das características fique abaixo de um padrão pré-definido pelos atores envolvidos no processo de medição do sistema.

Por outro lado, se a medição for positiva, isso vem ratificar a qualidade do sistema e ações poderão ser tomadas para disseminar esse bom resultado, buscando a adesão de mais utilizadores.

Como forma de validação do modelo proposto, após sua utilização para identificação de cenários e contexto-de-uso possíveis de serem mensurados, foi apresentado um CDSS da área oncológica a profissionais de saúde, estudantes de medicina e profissionais da área de qualidade de software que, ao final de sua utilização, responderam a um inquérito com o objetivo de avaliar o sistema.

A aplicação se deu de forma online, dado a necessidade de mantermos o distanciamento social e o de cumprirmos as orientações sanitárias.

As respostas serviram como fonte de dados para a medição das características de qualidade-em-uso do sistema.

Os resultados da aplicação revelou que nosso modelo de processo de avaliação é válido, relevante e de fácil utilização para identificar as características importantes em um sistema, bem como suas medições por meio das funções matemáticas do modelo ISO/IEC 25022.

Outras contribuições do nosso trabalho, temos

1) no âmbito acadêmico, um estudo significativo na área de qualidade de software, com foco em suas características, especialmente na qualidade em uso. Uma *guideline* para a coleta e mensuração dessas características foi construída em nosso modelo de processo;

2) na área de desenvolvimento de software, os profissionais podem contar com um processo simples e adaptável, aplicável a outros tipos de sistema, para mensuração da qualidade em uso de seus produtos.

Palavras-chave

Avaliação de qualidade em uso; qualidade em uso; qualidade de software; modelo de avaliação; ISO/IEC 25010; ISO/IEC 25022; CDSS; sistema de apoio à decisão clínica.

Bibliography

[1] M. Khalifa, "Clinical decision support: Strategies for success," in Procedia Comput. Sci., vol. 37, 2014, pp. 422–427. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187705091401028X xi

[2] D. Zikos, "A Framework to Design Successful Clinical Decision Support Systems," in Proc. 10th Int. Conf. PErvasive Technol. Relat. to Assist. Environ., ser. PETRA '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 185–188. [Online]. Available: https://doi.org/10.1145/3056540.3064960 xi

[3] P. Shankar and N. Anderson, "Advances in Sharing Multi-sourced Health Data on Decision Support Science 2016-2017," pp. 16–24, 8 2018. [Online]. Available: http://www.thieme-connect.de/DOI/DOI?10.1055/s-0038-1641215 xi

[4] Y. F. Zhang, L. Gou, T. Shu Zhou, D. Nan Lin, J. Zheng, Y. Li, and et al., "An ontology-based approach to patient follow-up assessment for continuous and personalized chronic disease management," Journal of Biomedical Informatics, vol. 72, pp. 45–59, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1532046417301569 xi

[5] S. Piri, D. Delen, T. Liu, and H. M. Zolbanin, "A data analytics approach to building a clinical decision support system for diabetic retinopathy: Developing and deploying a model ensemble," Decision Support Systems, vol. 101, pp. 12–27, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167923617300908 xi

[6] G. B. Laleci Erturkmen, M. Yuksel, B. Sarigul, T. N. Arvanitis, P. Lindman, R. Chen, and et.al., "A Collaborative Platform for Management of Chronic Diseases via Guideline-Driven Individualized Care Plans," Computational and Structural Biotechnology Journal, vol. 17, pp. 869–885, 1 2019. [Online]. Available: https://doi.org/10.1016/j.csbj.2019.06.003 xi

[7] F. Ali, S. M. R. Islam, D. Kwak, P. Khan, N. Ullah, S.-j. jo Yoo, and K. S. Kwak, "Type-2 fuzzy ontology–aided recommendation systems for IoT–based healthcare," Computer Communications, vol. 119, pp. 138–155, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366417310587 xi

[8] M. Peleg, Y. Shahar, S. Quaglini, T. Broens, R. Budasu, N. Fung, and et al., "Assessment of a personalized and distributed patient guidance system," International Journal of Medical Informatics, vol. 101, pp. 108–130, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1386505617300497 xi

[9] N. Pombo, P. Rebelo, P. Araújo, and J. Viana, "Design and evaluation of a decision support system for pain management based on data imputation and statistical models," Measurement: Journal of the International Measurement Confederation, vol. 93, pp. 480–489, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0263224116303682 xi

[10] A. Jimenez-Molina, J. Gaete-Villegas, and J. Fuentes, "ProFUSO: Business process and ontology-based framework to develop ubiquitous computing support systems for chronic patients' management," Journal of Biomedical Informatics, vol. 82, pp. 106–127, 6 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1532046418300650 xi

[11] E. Shalom, Y. Shahar, and E. Lunenfeld, "An architecture for a continuous, user-driven, and data-driven application of clinical guidelines and its evaluation," Journal of Biomedical Informatics, vol. 59, pp. 130–148, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1532046415002439 xi

[12] M. Afzal, M. Hussain, W. A. Khan, T. Ali, S. Lee, E.-N. Huh, and et al., "Comprehensible knowledge model creation for cancer treatment decision making," Computers in Biology and Medicine, vol. 82, pp. 119–129, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010482517300094 xi

[13] S. Khairat, D. Marc, W. Crosby, and A. Al Sanousi, "Reasons for physicians not adopting clinical decision support systems: Critical analysis," J. Med. Internet Res., vol. 20, no. 4, 2018. xi

[14] K. A. Alshare, M. K. Alomari, P. L. Lane, and R. D. Freeze, "Development and determinants of end-user intention: usage of expert systems," J. Syst. Inf. Technol., vol. 21, no. 2, pp. 166–185, 2019. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/JSIT-08-2018-0108/full/html xi

[15] C. J. Schuh, J. S. De Bruin, and W. Seeling, "Acceptability and difficulties of (fuzzy) decision support systems in clinical practice," in Proc. 2013 Jt. IFSA World Congr. NAFIPS Annu. Meet. IFSA/NAFIPS 2013, 2013, pp. 257–262. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6608409 xi

[16] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, and N. Garcia, "Clinical decision support systems for chronic diseases: A Systematic literature review," Computer Methods and Programs in Biomedicine, vol. 195, 2020. [Online]. Available: https://doi.org/10.1016/j.cmpb.2020.105565 xi

[17] "ISO 25000 STANDARDS," 2011. [Online]. Available: https://iso25000.com/index.php/en/iso-25000-standards xi

[18] "Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQUARE) — System and software quality models," p. 34, 2011. [Online]. Available: https://www.iso.org/standard/35733.html xi

[19] "Systems and software engineering — Systems and software quality requirements and evaluation (SQUARE) — Measurement of quality in use," p. 41, 2016. [Online]. Available: https://www.iso.org/standard/35746.html xi

[20] L. Souza-Pereira, S. Ouhbi, and N. Pombo, "Quality-in-use characteristics for clinical decision support system assessment," Comput. Methods Programs Biomed., vol. 207, p.

106169, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260721002431 xii

[21] R. Harrison, D. Flood, and D. Duce, "Usability of mobile applications: literature review and rationale for a new usability model," J. Interact. Sci., vol. 1, no. 1, p. 1, 2013. xii

Resumo Alargado

Introdução

Esta seção vem resumir, em Língua Portuguesa e seguindo o Acordo Ortográfico da Língua Portuguesa de 1990, o trabalho de investigação descrito na tese de doutoramento intitulada "A process model for quality in use evaluation on clinical decision support systems".

Neste capítulo são apresentados:

1) enquadramento da tese, definindo o problema abordado;

2) os objetivos da investigação;

3) a definição do argumento;

4) as principais contribuições deste estudo, e por fim, últimas seções do capítulo trazem de forma resumida;

5) as principais conclusões e perspectivas para investigação futura.

Enquadramento da Tese

Sistemas informatizados na área de médica são utilizados na tentativa de prover um melhor gerenciamento nos cuidados clínicos de pacientes, sejam estes crônicos ou não, e na busca de facilitar e suportar o trabalho do profissional de saúde. Como exemplo de um destes sistemas, temos o Sistema de Apoio à Decisão Clínica, ou **CDSS** como acrônimo em inglês para Clinical Decision Support System.

CDSS é uma ferramenta que busca aprimorar o conhecimento essencial dos profissionais de saúde e também pacientes, através de apoio em uma tomada de decisão clínica [1, 2, 3]. CDSSs fornecem ao seu usuário conhecimentos e informações específicas, filtradas de forma inteligente ou apresentadas em momentos apropriados [4].

É esperado que esses sistemas possam tanto fornecer uma melhor orientação clínica quanto capacitar um paciente em termos de consciência e conhecimento de sua condição clínica.

No entanto, apesar de todos os benefícios oferecidos por um CDSS, sua aceitação na área médica ainda é uma questão de debate [5, 6].

A aceitação e utilização de um CDSS se deve a fatores como a percepção do usuário final sobre a facilidade de uso e utilidade do sistema, bem como a qualidade e confiabilidade dos resultados fornecidos [6].

Outros motivos seriam [7, 8]:

1) O contexto do sistema, que por vezes atende a um único propósito e busca resolver um único problema, i.e., os sistemas versam sobre o acompanhamento clínico de uma única especialidade médica (e.g. câncer), não considerando comorbidades associadas (e.g. dor crônica);

2) A inclusão de atividades obrigatórias para a execução do sistema fora da rotina e do fluxo de trabalho do profissional de saúde (e.g., a obrigatoriedade da entrada de uma série de informações clínicas antes da geração de um diagnóstico);

3) Os sistemas não tratarem a interoperabilidade, i.e., não estarem integrados a sistemas já em utilização, gerando duplicidade de entrada de dados.

Um bom CDSS deve produzir resultados confiáveis, atender às expectativas do usuário e ser útil [9, 10]. Uma forma de garantir estas características seria desenvolvê-lo seguindo as melhores práticas da engenharia de software (SE, acrônimo em inglês) [8]. Uma forma de se verificar que as tais características estão incorporadas no produto de software é avaliar suas características de qualidade.

Medir as características de qualidade de um sistema pode contribuir para mitigar os motivos de não aceitação ou utilização reduzida, identificando deficiências e fraquezas que afetam sua confiabilidade, sua falha em atender às expectativas e necessidades dos usuários, ou mesmo identificar problemas em seu desenvolvimento. Portanto, é muito importante investigar as características relevantes para avaliar a qualidade do produto de software que permitam validar o contexto da proposta e comprovar os benefícios de sua utilização.

Para indicar o grau em que o sistema atende às necessidades das partes interessadas, sejam estas necessidades declaradas ou implícitas, foram criados padrões internacionais de conceitos, procedimentos e métricas de qualidade para medir a qualidade de software e sistemas [11]. A série mais recente de padrões é a Requisitos e Avaliação de Qualidade de Sistema e Software (ISO/IEC 25000) - *System and Software Quality Requirements and Evaluation* (SQUARE) [11], sendo dividida em modelos para requisitos, gestão, medição, e avaliação da qualidade de sistemas e produtos de software.

ISO/IEC 25010 [12] e ISO/IEC 25022 [13] são duas destas divisões, sendo que a primeira [12] trata da especificação dos modelos **1)** da qualidade do produto de software e **2)** da qualidade em uso (QiU, acrônimo em inglês) do sistema, e a segunda [13] trata de orientações e regras para a medição da QiU.

Devido às limitações de tempo e orçamento, é difícil avaliar todas as características de qualidade. Sendo assim, deve-se identificar os pontos críticos de qualidade que precisam ser avaliados no sistema ou produto de software.

O âmbito desta tese é limitado a sistemas computorizados que permitam o apoio à tomada de decisão clínica (e.g., CDSS) e a avaliação da qualidade em uso destes sistemas.
O foco está em mensurar características de QiU para melhorar a aceitação e o uso destes sistemas por parte dos profissionais de saúde. Essa medição busca favorecer a identificação de possíveis falhas nestes softwares que justifiquem sua não-aceitação. Tal reconhecimento permite correções e/ou aprimoramentos no software, contribuindo para minimizar esse problema, assim como outros identificados através do processo de mensuração.

Descrição do Problema e Objetivos da Investigação

O problema abordado versa sobre a utilização do CDSS pelos profissionais de saúde e a busca da avaliação da qualidade em uso destes sistemas como forma de contribuição para expansão de seu uso.

O estudo inicial teve como objetivo caracterizar os CDSS e como eram desenvolvidos sob a ótica da SE, elaborando assim nosso Estudo da Arte.

Inicialmente, pensou-se em restringir a pesquisa em CDSS voltados para o administração de doenças crônicas, mas com o decorrer do estudo e a definição completa do objetivo da tese, essa restrição foi eliminada.
Nesta etapa foram observadas

1) as finalidades destes sistemas;

2) a metodologia utilizada no desenvolvimento;

3) o uso de técnicas de SE em sua construção;

4) como eram coletadas e trabalhadas as informações clínicas;

5) como eram testados e validados esses sistemas.

Como resultado da pesquisa foram observados ausência na demonstração da utilização dos recursos de SE, tais como a explicação das fases de desenvolvimento e desenhos de estruturas [8].

Também foi observado que a interoperabilidade dos CDSSs com outros sistemas já utilizados pelos usuários (e.g., registro eletrônico de saúde ou EHR, acrônimo em inglês) não foi uma questão aprofundada nos estudos ou foi sequer mencionada. A interoperabilidade entre sistemas é um quesito importante para a aceitação do CDSS dentre os profissionais de saúde [8].

Como origem dos dados usados para validação e testes nestes sistemas usou-se, na maioria dos estudos, repositórios de banco de dados.

O critério de experiência do usuário foi utilizado para avaliação destes sistemas, como também a performance da arquitetura. Neste sentido, a acurácia, especificidade e sensibilidade

foram as métricas mais comuns.

Para a fase intermediária do estudo, já desconsiderada a restrição inicial de doenças crônicas, foram pesquisados pontos relativos a qualidade dos CDSS, uma vez que entendemos que a não-utilização do sistema poderia estar ligada ao não atendimento das expectativas do usuário, reflexo, talvez, da pouca utilização da SE.

Os pontos abordados para o segundo estudo foram: **1)** como é avaliada a qualidade dos CDSSs, **2)** quais as características de qualidade são mensuradas, **3)** quais pontos críticos da qualidade do CDSS se relacionam com sua utilização pelos profissionais de saúde [14]. Além das respostas a estes tópicos, observou-se que a experiência do usuário (UX, acrônimo em inglês) era importante para a aceitação do CDSS.

Elaborou-se, então, o principal objetivo desta tese, que seria criar um modelo de processo para medir as características de qualidade em uso de um CDSS como forma identificar problemas nos sistemas, e que viessem a contribuir para diminuir sua não-aceitação entre os profissionais de saúde. Tal modelo seguiu a série de padrões ISO/IEC 25000, particularmente ISO/IEC 25010 e ISO/IEC 25022, descrito no Capítulo IV e em [15].
Outros objetivos da tese foram:

- Investigar tecnologias utilizadas na construção de um CDSS, bem como suas aplicações;
- Medir a qualidade em uso de um protótipo ou aplicação de CDSS;
- Elaborar relatório com sugestões de reconstrução e aprimoramento do CDSS com base nas fragilidades observadas durante a medição da qualidade;

Argumento da Tese

Esta tese propõe um modelo de processo para avaliação da qualidade em uso de um sistema de apoio à decisão clínica. A metodologia da pesquisa consiste em abordagem, design e métodos. A abordagem usada foi mista, com aspectos qualitativos e quantitativos sendo aplicados ao decorrer do desenvolvimento da tese [16].

O desenho da pesquisa segue o padrão exploratório sequencial, onde tem-se a combinação de coleta e análise de dados (qualitativos e quantitativos) ao longo de uma sequência de fases [17, 18].

Os métodos de pesquisa são divididos em fases:

1) **Fase qualitativa** - Consistindo em **(a)** uma revisão sistemática de literatura (capítulo 2), **(b)** da pesquisa acerca dos conceitos de padrões e modelos e características de qualidade de software (capítulo 3) e **(c)** do desenvolvimento do modelo de avaliação da qualidade em uso de um CDSS (capítulo 4);

2) **Fase quantitativa** - Consistindo na aplicação do modelo de processo de avaliação em um CDSS (capítulo 5);

3) **Fase mista** - Consistindo na análise e interpretação dos valores conseguidos na mensuração das características de qualidade em uso durante a aplicação do modelo de avaliação.

Principais Contribuições

As principais contribuições desta tese estão distribuídas ao longo do documento em quatro capítulos. São artigos publicados em revistas internacionais. Na produção de todos eles foram utilizados as plataformas Mendeley[1] e Ryyan[2] para a organização dos estudos candidatos durante o processo de seleção. Microsoft Excel[3] e o Microsoft Visio[3] também foram utilizados para gerar os gráficos e figuras apresentados nos artigos.

A primeira contribuição foi uma Revisão Sistemática de Literatura (SLR, acrônimo em inglês) com a descrição de abordagens existentes de metodologias de desenvolvimento de CDSSs, propósitos, estruturas de SE utilizadas em seus desenhos, forma de coleta das informações clínicas e métodos utilizados para a validação dos sistemas. Este estudo está detalhado no capítulo 2, que consiste em um artigo publicado em *Computer Methods and Programs in Biomedicine Journal.*

A segunda contribuição desta tese é a descrição dos modelos de qualidade de software, em especial àqueles abordados pela série internacional de padrões ISO/IEC 25010: *Quality Models Division.* No sentido de buscar expandir o uso de CDSS entre os profissionais de saúde, identificamos também características de QiU para serem avaliadas no contexto destes sistemas. Trata-se de um artigo publicado em *Computer Methods and Programs in Biomedicine Journal* e descrito no capítulo 3.

A terceira contribuição, considerada como o objetivo principal, é a proposta de um modelo de processo para a avaliação da QiU de um CDSS, baseada nos estudos dos capítulos anteriores e outros estudos bibliográficos. Tal contribuição é apresentada no capítulo 4, em artigo publicado pela revista *Jounal of Biomedical Informatics.*

A quarta contribuição trata-se da validação do modelo desenvolvido através de sua aplicação para a medição de qualidade em uso de um CDSS da área Oncológica. A priori a ideia era aplicar o modelo proposto na avaliação de um CDSS em um hospital, *in loco*, na cidade da Covilhã. No entanto, com o advento da pandemia isso se tornou inexequível. A solução encontrada para a validação do modelo foi usá-lo para avaliar uma aplicação CDSS através da contribuição de profissionais e estudantes da área médica e da área de qualidade de software

[1] www.mendeley.com/
[2] https://rayyan.qcri.org
[3] Microsoft Corporation ®

de Portugal. Os resultados desta avaliação foram descritos em artigo publicado na revista *Journal of King Saud University - Computer and Information Sciences*. Esta parte do estudo está detalhada no capítulo 5.

Principais Conclusões / Detalhamento das fases

Esta tese é focada na avaliação da qualidade em uso de sistemas de apoio à decisão clínica e descreve o trabalho de investigação desenvolvido ao longo do período de doutoramento.

O estudo resultou em contribuições importantes para o estado da arte, conforme apresentado em seção anterior, e também permitiu que os objetivos da tese fossem atingidos, em especial a validação do modelo de processo de avaliação.

Primeira fase

Como primeira contribuição, uma SLR foi realizada de acordo com as orientações do *Preferred Reporting Items for Systematic Reviews and Meta-Analyses - PRISMA - statement*, destacando as técnicas de SE utilizadas no desenho destes sistemas bem como os métodos usados para suas avaliações. Neste último quesito, foi observado o uso da UX como um dos critérios para avaliar os sistemas, além do uso reduzido de um método bastante útil para tais avaliações: o questionário.

As cinco questões de pesquisa (RQ) apresentadas na SLR foram:

1) RQ1 - Quais são as motivações para o desenvolvimento do CDSS? Esta questão pretendeu atingir os principais objetivos do CDSS e seu contexto de desenvolvimento;

2) RQ2 - Quais são os estágios da SE seguidos ao longo do ciclo de vida de desenvolvimento do CDSS? Esta questão visou trazer à tona o estado atual da aplicação dos princípios de SE;

3) RQ3 - Como os dados são coletados e usados? A resposta a esta questão buscou identificar métodos de coleta de dados, bem como técnicas de ML para inferir conhecimento a partir desses dados;

4) RQ4 - Quais são as metodologias utilizadas para avaliar o CDSS? Aqui a intenção foi identificar quais técnicas de avaliação foram usadas em cada CDSS;

5) RQ5 - De que forma os autores provaram a validade do CDSS desenvolvido? Esta questão buscou levantar o estado atual do uso de métricas na atividade de validação.

Dos trezentos e trinta e nove estudos candidatos, quatorze deles foram selecionados e examinados. Os principais resultados foram condensados como respostas das RQs e apresentados da seguinte forma:

1) (RQ1) Os CDSSs encontrados nos estudos selecionados estavam focados no propósito de Gerenciamento de Acompanhamento, Prevenção e Triagem, Diagnóstico, Tratamento e Diretrizes, e ainda Gerenciamento de Informações Clínicas. Destes, 85.7% tratou do diagnóstico, sendo o propósito mais frequente. Foi sugerida a inclusão de abordagens de prevenção e triagem para os CDSSs de diagnóstico, uma vez que se acredita que triagem e prevenção adicionais da doença poderiam ser alcançadas com os mesmos dados usados para o diagnóstico. Além disso, isso poderia implicar em redução de custos e também em aumento da qualidade de vida dos pacientes com a prevenção precoce;

2) (RQ2) A representatividade das etapas da SE foi escassa e feita por diagramas Estruturais e Comportamentais acoplados a outros diagramas, como por exemplo, projeto arquitetônico ou (*framework*). Como uma estratégia para aumentar a replicabilidade do estudo, comparabilidade e avaliação mais precisa, foram recomendados estudos futuros para incluir mais representação de design e descrição de requisitos;

3) (RQ3) Os autores usaram Bancos de Dados, Sensores e/ou Auto relatórios como fontes de dados para seus CDSS. Bancos de dados foram os mais utilizados pelos estudos. Foi mostrado uma relação entre a fonte de dados, as técnicas de aprendizado de máquina e a interoperabilidade entre o CDSS e os sistemas existentes: **(a)** a maioria dos estudos que utilizaram o EHR abordou a interoperabilidade como uma característica importante do CDSS, e **(b)** a maioria dos estudos que usaram um conjunto de dados de repositórios aplicou técnicas de aprendizado de máquina para testar o modelo ou *framework*. Recomendou-se que a busca pela solução do problema de interoperabilidade seja uma etapa do projeto dos CDSSs, pois é fundamental para sua aceitabilidade e alinha parcialmente o desenvolvimento do software para conformidade com a ISO25010 [12];

4) (RQ4) Os sistemas foram avaliados usando os critérios de experiência do usuário, o desempenho de implementação arquitetônica e testes de software. Foi sugerido um uso mais amplo de questionários para avaliar a satisfação dos usuários e a integração do *feedback* do usuário no ciclo de vida de desenvolvimento do CDSS;

5) (RQ5) As métricas de precisão, especificidade e sensibilidade foram as as mais apresentadas nos estudos. Não houve medição de tempo ou esforço usado no desenvolvimento do aplicativo CDSS. Ou seja não foi possível determinar quanto esforço foi feito para atingir níveis aceitáveis ou ideais de precisão, especificidade e parâmetros de sensibilidade. Além disso, esse esforço deve ser considerado por si mesmo uma métrica, ou alternativamente, o grau de complexidade do sistema planejado. Portanto, recomendou-se que, quando se tratando de CDSSs complexos, uma métrica fosse descrita de forma relacionada ao esforço de desenvolvimento e à confiabilidade.

Além de todas as questões e respostas apresentadas na SLR, também foi possível descobrir, durante a pesquisa bibliográfica, que CDSS não são amplamente usados, embora sejam de grande benefício para os profissionais de saúde[5].
Acredita-se que a SLR pôde contribuir com os pesquisadores na verificação das etapas da SE aplicadas na construção de um CDSS.

No contexto de que a pouca utilização do CDSS pudesse estar ligado à qualidade do sistema, por não atender às expectativas do usuário, focarmos em descobrir e avaliar a qualidade destes sistemas.

Segunda fase

A etapa seguinte, ainda na fase qualitativa do estudo, foi descobrir os modelos estabelecidos para garantir a qualidade de um software. Para tal, foram feitas pesquisas relacionadas a modelos de qualidade e a características de qualidade observadas e avaliadas em softwares, especificamente em CDSS.

Vários modelos foram encontrados durante a pesquisa e cada um deles trabalham características consideradas importantes para a qualidade de um software. Este capítulo se limitará a apresentá-los, não entrando no mérito das características de qualidade abordadas, exceto para o modelo adotado pela tese. Os modelos mais conhecidos são:

- **McCall's Model** - este modelo define as características de qualidade de um produto agrupado em 3 categorias: **1)** operação do produto; **2)** fatores de revisão do produto e **3)** fatores de transição do produto [19, 20, 21];
- **Modelo Boehm** - este modelo melhora o Modelo McCall e a característica de qualidade mais importante é definida como utilidade geral, que consiste em utilidade, portabilidade e manutenibilidade [22, 19];
- **Modelo FURPS** - Este modelo categoriza os requisitos em funcionais e não funcionais [23];
- **Modelo de Qualidade de Dromey** - Este modelo é focado em confiabilidade e manutenção, baseando-se nas relações existentes entre as propriedades do software e suas características de qualidade [19];
- **Modelo de Qualidade ISO 9126-1** - é a primeira parte da norma ISO/IEC 9126 que trata do estabelecimento de um sistema de características e subcaracterísticas para a definição da qualidade de software. Nele, a qualidade do produto de software foi dividida em duas categorias principais: **1)** características de qualidade interna, que podem ser observadas sem executar o sistema, e **2)** características de qualidade externas, que só podem ser observadas durante a execução do sistema [24, 25].;
- **System and software quality models - ISO/IEC 25010** [12] está inserido na *Quality Model Division ISO/IEC 2501n*, que se trata de uma das divisões do *SQuaRE* [11]. Derivado do Modelo ISO 9126-1991, substituindo-o, em 2011, com algumas alterações, e.g., compatibilidade e segurança tornando-se características [24]. Este é o padrão seguindo por este estudo e as características abordadas por ele serão explicitadas no decorrer do capítulo.

ISO/IEC 25010 - Modelos de qualidade de sistema e software- está dividido em duas grandes dimensões:

1. **Modelo de qualidade do produto de software**, que consiste em oito características com propriedades estáticas de software (i.e., atributos internos de qualidade) e propriedades dinâmicas do sistema (i.e., atributos externos de qualidade);

2. **Modelo de qualidade em uso**, que consiste em cinco características, relacionando-se com o resultado da interação do usuário e o produto/software em um determinado contexto.

Características de qualidade dos modelos ISO/IEC 25010.

O padrão internacional ISO/IEC 25010 [12], recomenda para a especificação ou avaliação da qualidade do sistema de computador ou produto de software, que se faça uso de todas as características dos modelos, tanto de qualidade em uso quanto de qualidade do produto.
A divisão da **Qualidade do Produto de Software** apresenta as características [12]:

- Adequação Funcional - este recurso mostra o quanto um produto ou sistema atende às necessidades declaradas e implícitas por meio de suas funções quando usado nas condições especificadas;

- Eficiência de Desempenho - refere-se à quantidade de recursos utilizados nas condições estabelecidas. Esses recursos podem ser outros softwares e sistemas, materiais como impressoras, materiais de armazenamento e outros;

- Compatibilidade - o grau em que um componente, produto ou sistema pode trocar informações com outros e/ou executar as funções necessárias ao compartilhar um software ou um ambiente de hardware;

- Usabilidade - mostra o quanto um produto ou sistema pode ser utilizado por usuários específicos em um determinado contexto de uso, para atingir objetivos específicos com eficiência, eficácia e satisfação. Ela pode ser especificada ou medida como uma característica de qualidade do produto (em termos de subcaracterísticas), ou especificada ou medida como um subconjunto da qualidade em uso;

- Confiabilidade - o grau em que um sistema, produto ou componente desempenha funções específicas sob condições e períodos específicos. Falhas nos requisitos, projeto e implementação ou mudanças contextuais contribuem para limitações na confiabilidade;

- Segurança - preocupa-se com a quantidade de informações protegidas, bem como o grau adequado de acesso aos dados e níveis de autorização;

- Manutenibilidade - grau de eficiência e eficácia com que um sistema ou produto pode ser modificado;

- Portabilidade - medida de eficiência e eficácia na transferência de um componente, produto ou sistema de um ambiente para outro, seja ele de hardware, software ou outro ambiente de uso operacional.

Estas oito características são compostas de trinta e uma subcaracterísticas.

A divisão de **Qualidade em Uso** apresenta as características [12]:

- Eficácia ou Efetividade - representa o nível de precisão e completude com que os usuários alcançaram objetivos específicos na utilização do sistema ou produto;
- Eficiência - preocupa-se com os recursos despendidos para o alcance das metas. Sua medida relaciona o nível de eficácia alcançado com o dispêndio de recursos;
- Satisfação - trata do quanto as necessidades do usuário são satisfeitas dentro de um determinado contexto de uso do sistema ou produto. Aqui estão incluídos os desejos e expectativas dos usuários face ao sistema ou produto utilizado;
- Livre de risco - trata do grau em que a qualidade de um sistema ou produto permite mitigar ou evitar riscos potenciais à vida humana, à situação econômica, à saúde ou ao meio ambiente;
- Cobertura de contexto - A cobertura de contexto é a soma das características anteriores, quando um sistema ou produto é utilizado em contextos específicos e/ou em contextos além dos inicialmente identificados.

As características Satisfação, Livre-de-riscos e Cobertura de contexto somam outras nove subcaracterísticas.

Foi observado entre os estudos candidatos desta segunda fase que a preferência dos autores foi por utilizar as características de qualidade do produto de software, por exemplo, usabilidade, para avaliar os sistemas; as características de qualidade em uso raramente foram usadas para determinar a qualidade de um CDSS.

Das treze características de qualidade referenciadas pela norma ISO/IEC 25010, apenas duas não foram observadas nos estudos: liberdade de risco e cobertura de contexto. Cinquenta e seis características e subcaracterísticas foram extraídas dos estudos selecionados. A Usabilidade foi a característica mais medida, respondendo por 26,8%, seguida por Segurança, Confiabilidade e Manutenibilidade, respondendo por 12,5%, 12,5% e 10,7% respectivamente. Todas as demais características tiveram menos de 9,0% de representatividade. A Qualidade em Uso teve uma representatividade de 10,7%, somando todas as suas características e subcaracterísticas abordadas.

Terceira fase

A partir da constatação da ínfima medição da qualidade em uso, surgiu a ideia principal da tese, que foi a proposição um modelo de processo de avaliação de importantes características de qualidade em uso que viessem a contribuir para uma ampla utilização dos CDSS.

Existe uma relação intrínseca entre os modelos de qualidade do produto e qualidade em uso, que depende da visão e do tipo de usuário. Portanto, foi abordada a perspectiva do usuário primário (ou final) para avaliação das características de qualidade em uso.

Foram escolhidas as características **Satisfação** e **Eficiência** como exemplos para serem avaliadas nos CDSSs, pois o usuário se preocupa com sua experiência e produtividade [26] e tais características fazem parte do contexto da usabilidade do sistema.

As demais características estão mais ligadas aos outros tipos de usuários do sistemas: **1)** usuários secundários (e.g., desenvolvedores do sistema, analistas e engenheiros, provedores de conteúdos) e **2)** usuários indiretos (e.g., financiador do sistema, representante da empresa) [12].

Embora sejam abordadas estas duas características no exemplo ilustrativo da aplicação do modelo de processo de avaliação de QiU, ele não se limita a avaliar somente estas. Caso o avaliador queira fazer uso de outras características, isso é totalmente possível e a aplicação do modelo é factível.

A eficiência diz respeito aos resultados alcançados e aos recursos usados para alcançá-los. Portanto, é importante utilizar o mínimo de recursos (por exemplo, o tempo gasto para realizar uma tarefa, que é um recurso), reduzindo custos (trabalho por hora), mas continuando a trazer os resultados desejados.

Satisfação está relacionada à percepção do usuário quanto às respostas oferecidas pelo sistema e a forma de interação com o mesmo. Refere-se a como as necessidades do usuário são atendidas em um contexto particular de uso do sistema;

Para **Satisfação**, as seguintes subcaracterísticas foram consideradas para nosso exemplo ilustrativo:

- Utilidade - esta subcaracterística permitirá verificar a satisfação do usuário quanto à sua percepção do alcance dos objetivos práticos, incluindo os resultados e consequências da utilização do CDSS;
- Confiança - permitirá avaliar o grau de confiança do usuário em relação ao comportamento do CDSS, se for como pretendido e esperado;
- Prazer - está relacionado às experiências do usuário e permitirá avaliar o grau de satisfação de suas necessidades de prazer.

Para **Eficiência**, os seguintes atributos foram considerados em nosso exemplo ilustrativo:

- Tempo da tarefa - permite determinar quanto tempo o usuário leva para completar sua tarefa com sucesso;
- Eficiência de tempo - a eficiência com que os usuários alcançam seus objetivos ao longo do tempo ao utilizar o CDSS;

- Custo-efetividade - é o custo para realizar uma tarefa com efetividade. Por exemplo, os custos podem incluir o custo dos recursos materiais e o custo do tempo dos usuários;
- Razão de tempo produtivo - refere-se à proporção de tempo que o usuário gasta realizando ações produtivas durante a utilização do CDSS.

Ao avaliar essas subcaracterísticas e atributos, a intenção foi medir a qualidade do CDSS e associar essa medida à aceitação do sistema pelo usuário.

Tais características foram escolhidas por estarem ligadas à experiência do usuário ao utilizar CDSS, sendo a UX fundamental para a boa aceitação e utilização do sistema.

Esta fase, ainda na parte qualitativa do estudo, buscou a concepção de um modelo de processo de avaliação destas características de qualidade.

Este modelo de processo foi desenvolvido considerando uma adaptação do método GQM, acrônimo para *Goal-Question-Metric*, para a definição das métricas e medidas a serem usadas na medição da qualidade.

O método GQM é mais comumente usado para avaliar a qualidade do desenvolvimento do sistema e compreende 3 etapas: **1)** definição de uma meta, **2)** para cada meta, definição de questões cujas respostas ajudam a atingir essa meta e, **3)** para cada questão, definição de um conjunto de métricas para se chegar à resposta. [27].

O modelo proposto para avaliação das características de QiU de um CDSS é composto por cinco fases ou etapas principais, sendo elas:

1) Identificação de cenários e contexto de uso;

2) Seleção de medidas, métodos e métricas a serem utilizadas;

3) Medição da qualidade;

4) Análise de valores encontrados na medição;

5) Apresentação de resultados.

Como a qualidade em uso está diretamente ligada à um contexto de uso, a primeira etapa do modelo se preocupa em identificar estes contextos e criar os cenários para serem avaliados.

Para uma execução bem feita desta fase, faz-se necessário um estudo detalhado do sistema, recorrendo à documentação de utilização do CDSS bem como, se necessário e possível, às entrevistas com projetistas e analistas do sistema para se entender o grau de complexidade na efetivação destas tarefas.

Nesta etapa têm-se:

1) a identificação das principais funcionalidades do CDSS e suas características;

2) qual o resultado produzido pela execução de cada uma destas funcionalidades;

3) quem são os usuários executores destas funções;

4) quais são os atributos de habilidade e experiências destes usuários.

É criada, assim, uma base dados com estas informações que, para critério ilustrativo da aplicação do modelo, foram distribuídas em tabelas de Funcionalidades, Resultados e Usuários.

Cada relacionamento entre as entidades dessas tabelas dá origem a um cenário diferente, que são por sua vez armazenados em uma tabela chamada de Cenários.

A partir do momento em que todos os cenários possíveis estão identificados, passa-se para a segunda etapa do modelo, onde os avaliadores irão definir as medidas, as métricas e os métodos a serem utilizados na avaliação da qualidade.

As medidas a serem utilizadas devem priorizar àquelas apresentadas pela ISO/IEC 25022 [13] para a medição das QiUs (eficiência e satisfação, no nosso estudo de caso). Se alguma nova medida for desenvolvida, ela deve ser justificada, segundo o padrão internacional ISO/IEC de medição de qualidade.

Para a identificação da melhor medida a ser utilizada, recorremos ao método GQM adaptado para criar a tríade, onde a questão elaborada refere-se um cenário específico, com foco na análise da relação funcionalidade-resultado. A lista GQM formulada também é guardada na base de dados.

De porte das informações armazenadas até aqui, ainda na segunda etapa do modelo, os avaliadores criam um plano de avaliação. Este plano contém:

- A definição da metodologia da avaliação: quais recursos serão utilizados e/ou disponibilizados, de que forma será feita a avaliação, etc;
- O desenvolvimentos dos questionários, se aplicável;
- O cronograma de avaliação e de apresentação de resultados;
- Um documento explicativo para orientar os usuários durante a aplicação da avaliação, com informações relativas às funcionalidades a serem avaliadas e qual o resultado esperado de todo o processo;
- A explicação de como se dará a apresentação dos resultados.

A medição em si acontece na terceira etapa do modelo, onde é executado o plano de avaliação, e está inserida na fase quantitativa da metodologia da pesquisa.

Nesta etapa o avaliador é, em parte, espectador do processo, não podendo ajudar o usuário a desenvolver a atividade. O usuário, por sua vez, só tem acesso aos recursos de ajuda disponibilizados pelo próprio CDSS.

Diz-se expectador "em parte" porque ele deve registrar os eventos ocorridos durante o processo, como por exemplo, interrupções, acesso ao manual de usuário ou ajuda *online*; ou ainda fazer a medição do tempo da execução da atividade ou outra métrica necessária.

A interação do avaliador e o usuário só acontece após a finalização da medição. Neste contato o avaliador registra as impressões, queixas e observações dos usuários no contexto avaliado.

Todo esse registro é detalhado e analisado na quarta etapa do modelo, onde acontece a valoração e interpretação dos resultados da avaliação.

Através da ISO/IEC 25022 [13], utilizada na concepção do nosso modelo de processo, temos um conjunto de exemplos de normalização de medidas de QiU, o que nos permite uma interpretação mais fácil do valor e significado da métrica.

Essas opções de normalização são: **1)** conformidade; **2)** *benchmarks*; **3)** séries temporais; **4)** proficiência; **5)** normas populacionais para satisfação.
Portanto, é nesta quarta etapa do modelo que as questões **(Q)** da tríade GQM são realmente respondidas, e os resultados da medição da função passam a ser interpretados.
Após essa interpretação, tem-se a quinta e última etapa do modelo, que trata da apresentação do resultado da avaliação da QiU.

Um relatório deve ser elaborado para apresentar tanto os pontos fortes do CDSS quanto suas deficiências. Neste relatório devem estar descritas as as características avaliadas associadas às medidas, uma breve explicação da função e das métricas utilizadas, os valores encontrados e a explicação do que eles significam no contexto da avaliação.

Com este conhecimento, os usuários poderão ratificar ou mudar suas impressões sobre o CDSS, e os demais *stakeholders* (equipe técnica, por exemplo) poderão melhorar os pontos de desvantagem e disponibilizar uma nova versão do CDSS.

O relatório da medição também pode referir-se aos usuários participantes e possíveis divergências em suas avaliações, sempre contextualizando a avaliação, imprevistos, nível de especialização e habilidades que possam ter impactado a avaliação.

Como recurso impactante na apresentação dos resultados, pode-se fazer uso de imagens, gráficos e tabelas, com cores ou tamanhos diversos. É importante salientar que esta fase faz parte da fase mista da metodologia de pesquisa, ou seja, com aspectos qualitativos (interpretação) e quantitativos (valoração).

Quarta fase

A quarta fase da elaboração da tese se concentra na aplicação e validação do modelo de processo proposto na fase anterior.

A ideia inicial para a aplicação do modelo proposto era usá-lo para avaliar um CDSS utilizado em um hospital. No entanto, a pandemia que assola o mundo até o momento, foi impeditivo a presença de um avaliador no hospital para realizar tal atividade. Além disso, o tempo dos profissionais de saúde passou a ser mais escasso e precioso, o que dificultaria suas participações na avaliação.

Assim, como alternativa, buscamos APPs de CDSSs que pudessem ser analisados e ter suas características de qualidades mensuradas, como um estudo de caso.
Buscamos na *Google Store* aplicações classificadas como CDSS, no intervalo de 2 a 5 de junho de 2021. Encontramos 248 apps que passaram por verificação e análise, até à eleição de um, para aplicarmos o nosso modelo.
Primeiramente foi verificado se a app realmente tratava-se de um CDSS. Esse processo resultou em **1)** 24 CDSS; **2)** 104 apps com funções educacionais (tutoriais, guidelines, e-books) e fora do contexto médico; **3)** 27 apps unicamente com funções de calculadoras clínicas (IMCs, dosagem de medicação, etc;) **4)** 28 MHRs, PHR etc; **5)** 65 agendas (nutricional, *fitness*, etc).

Após identificados os 24 CDSS, uma primeira filtragem foi feita, considerando aqueles que sofreram "updates" entre os anos de 2018 e junho de 2021. Esta filtragem nos retornou 18 APPs, que passaram por uma segunda filtragem relacionada ao ano de publicação e quantidade de downloads, da seguinte forma:

1) ano de publicação 2018, deveria ter quantidade de *downloads* igual ou superior a 5,000 (0 CDSS);

2) ano de publicação 2019, deveria ter quantidade de *downloads* igual ou superior a 1000 (0 CDSS);

3) ano de publicação 2021, deveria ter quantidade de *downloads* igual ou superior a 500 (1 CDSS);

4) ano de publicação 2021, deveria ter quantidade de *downloads* igual ou superior a 100 (5 CDSSs).

A segunda filtragem nos retornou, então, 6 CDSS.

Para a próxima etapa analisamos estes seis CDSSs de uma forma mais prática, através da execução de cada um. Durante o teste foram coletadas informações comuns, descritas na

Tabela 1, sem contudo avaliar as funcionalidades dos mesmos. O número de respostas positivas para cada pergunta da Tabela 1 foi o critério de classificação dos CDSS, apresentado na Tabela 2

Table 1: Questões da filtragem

Id.	Questão	Resp. positivas
1	A app possui informações acerca dos seus *updates*	3
2	Podemos sugerir mudanças, melhorias ou solictar novas funcinalidades na própria app?	2
3	Podemos enviar email para o suporte ou time de desenvolvedores?	5
4	A app oferece suporte de médicos?	6
5	A app apresenta termos/política de uso e termos legais ?	5
6	A app é direcionada a uma doença ou área específica??	4
7	Apresenta a possibilidade de salvar ou enviar resultados?	1
8	Apresenta relatório final com a sugestão / decisão?	5
9	Apresenta uma base teórica para apoiar os resultados?	6

Questões aplicadas aos 06 CDSS e o número de respostas positivas.

Table 2: Classificação dos CDSSs após a terceira filtragem

Id.	CDSS	Número de respostas positivas
1	ONCOassist	8
2	Neuro Mind	7
3	TherapySelector	7
4	CFS	5
5	MDCalc	5
6	Calculate	5

Classificação dos CDSSs por número de respostas positivas

Com a seleção de um único CDSS, o Oncoassist, partiu-se para a aplicação do nosso modelo, fase a fase:

1) Fase 1 - Identificamos todas as funcionalidades do sistema e elegemos os contextos de uso;

2) Fase 2 - Selecionamos as medidas, métricas e métodos a serem aplicados, através da criação de uma lista GQM de perguntas relativas às *features* do CDSS, usando o padrão ISO/IEC 25022 [12];

3) Fase 3 - Criamos um questionário *online* para a dar início à medição das características de qualidade em uso, particularmente as características Satisfação e Eficiência;

4) Fase 4 - Analisamos e mensuramos as respostas dos participantes e

5) Fase 5 -Apresentamos os resultados atráves da publicação de um artigo científico.

Detalhando um pouco mais as fases 3 e 4, foi aplicado um questionário aos utilizadores do CDSS e sumarizados os resultados através da utilização das funções matemáticas.
As funções matemáticas, tal como as características e subcaracterísticas às quais são associadas, estão descritas na Tabela 3 (em inglês).

O questionário foi dividido em duas partes: a parte específica, onde foi avaliadas duas funcionalidades do CDSS (criação de conta para acesso e *adjuvant tools » breast cancer*), e a parte geral, onde foi avaliado o CDSS como um todo.
Definimos um intervalo de valores em uma escala de 0-100 para considerarmos os resultados da medição: **1)** excelente se acima de 90,0, **2)** entre 80,0 e 90,0 é muito bom, **3)** entre 65,0 e 80,0 é bom, **4)** entre 60 e 65 é razoável, **5)** entre 51 e 60 que consideramos ruim e **6)** abaixo de 51, péssimo.

Table 3: Medidas e funções matemáticas

Carac.	Subcarac.	Medida	ID	Função	Descrição
Efficiency	N/A	Task Time	Ey 1-G	$X = T$	T = Time taken to successfully complete a task
Satisfaction	Trust	User Trust	STr 1-G	$X = A$	A = Psychometric scale value from a trust questionnaire
	(UX) Pleasure	User pleasure	SPl 1-G	$X = A$	A = Psychometric scale value from a pleasure questionnaire
	Usefulness	Satisfaction with features	SUs 2-G	$X = \sum(A_i)$	A_i = Response to a question related to a specific feature
			SUs 4-G	$X = \frac{A}{B}$	A = nº of user using a particular feature ; B = nº of user in a identified set of users of the system
			SUs 6-G	$X = \frac{A}{B}$	A = nº of user complaints for a particular feature; B = Total nº of user complaints about features
			SUs 3-G	$X = \frac{A}{B}$	A = nº of users using a particular feature; B = nº of potential users who could use the particular feature
			SUs 5-G	$X = \frac{A}{B}$	A = nº of user complaining; B = nº of user using the system
	General	Overall Satisfaction	SUs 1-G	$X = \sum(A_i)$	A_i = Response to a question

Funções usadas para a medição das características em uso do CDSS (em inglês).

A medição da funcionalidade específica e do CDSS, em termos de suas características QiU, revelou que o sistema atende às necessidades dos usuários, sendo classificado como **BOM**, atingindo 71,35 pontos em nossa escala de valores de classificação.

Os resultados da aplicação indicou que nosso modelo de processo de avaliação é válido, relevante e de fácil utilização para identificar as características importantes em um sistema, bem como facilitar a medição de suas características de qualidade em uso.
Outras contribuições do nosso trabalho, temos

1) no âmbito acadêmico, um estudo significativo na área de qualidade de software, com

foco em suas características, especialmente na qualidade em uso. Uma *guideline* para a coleta e mensuração dessas características foi construída em nosso modelo de processo;

2) na área de desenvolvimento de software, os profissionais podem contar com um processo simples e adaptável, aplicável a outros tipos de sistema, para mensuração da qualidade em uso de seus produtos.

Direções Para Trabalho Futuro

Uma das linhas de investigação que poderá ser desenvolvida no futuro, seria o aperfeiçoamento do modelo para atender a todos os *stakeholders* participantes no processo de utilização do sistema, avaliando mais, se não todas, características de qualidade em uso.

Ao se fazer isso, a possibilidade de novos métodos para o levantamento das medidas poderá surgir, uma vez que o objetivo da avaliação deixa de ser intrinsecamente ligado ao usuário final, incluindo também os financiadores e desenvolvedores do CDSS.

O mesmo poderá vir acontecer com as métricas, sendo necessário desenvolver novas formas de se chegar aos resultados que venham responder melhor ao que for solicitado pelos novos atores do processo.

Neste sentido estaríamos verificando, por completo, a qualidade em uso do CDSS.

Além disso, ainda há oportunidades para avaliação de sistema de apoio à decisão clínica instalados em *smartphones* de pacientes. Outro tipo de abordagem deverá ser feito e considerado neste caso, mas ainda como objeto de estudo a satisfação do paciente em utilizar a app.

Por fim, a medição da qualidade em uso do sistema voltado especificamente a responder os anseios dos desenvolvedores, no sentido da garantia de interoperabilidade, acordo com *guidelines* clínicas e a possibilidade de integração de apoio a multi-morbidades em um mesmo sistema.

Bibliography

[1] R. T. Sutton, D. Pincock, D. C. Baumgart, D. C. Sadowski, R. N. Fedorak, and K. I. Kroeker, "An overview of clinical decision support systems: benefits, risks, and strategies for success," npj Digit. Med., vol. 3, no. 1, pp. 1–10, 2020. [Online]. Available: http://dx.doi.org/10.1038/s41746-020-0221-y xix

[2] M. A. Balakhontceva, A. A. Funkner, A. A. Semakova, O. G. Metsker, N. E. Zvartau, A. N. Yakovlev, A. E. Lutsenko, and S. V. Kovalchuk, "Holistic Modeling of Chronic Diseases for Recommendation Elaboration and Decision Making," Procedia Computer Science, vol. 138, pp. 228–237, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050918316661 xix

[3] E. S. Berner, Clinical Decision Support Systems, 2nd ed., ser. Health Informatics. New York, NY: Springer New York, 2007. [Online]. Available: http://link.springer.com/10.1007/978-0-387-38319-4 xix

[4] P. Shankar and N. Anderson, "Advances in Sharing Multi-sourced Health Data on Decision Support Science 2016-2017," pp. 16–24, 8 2018. [Online]. Available: http://www.thieme-connect.de/DOI/DOI?10.1055/s-0038-1641215 xix

[5] S. Khairat, D. Marc, W. Crosby, and A. Al Sanousi, "Reasons for physicians not adopting clinical decision support systems: Critical analysis," J. Med. Internet Res., vol. 20, no. 4, 2018. xix, xxv

[6] K. A. Alshare, M. K. Alomari, P. L. Lane, and R. D. Freeze, "Development and determinants of end-user intention: usage of expert systems," J. Syst. Inf. Technol., vol. 21, no. 2, pp. 166–185, 2019. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/JSIT-08-2018-0108/full/html xix

[7] C. J. Schuh, J. S. De Bruin, and W. Seeling, "Acceptability and difficulties of (fuzzy) decision support systems in clinical practice," in Proc. 2013 Jt. IFSA World Congr. NAFIPS Annu. Meet. IFSA/NAFIPS 2013, 2013, pp. 257–262. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6608409 xx

[8] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, and N. Garcia, "Clinical decision support systems for chronic diseases: A Systematic literature review," Computer Methods and Programs in Biomedicine, vol. 195, 2020. [Online]. Available: https://doi.org/10.1016/j.cmpb.2020.105565 xx, xxi

[9] M. Khalifa, "Clinical decision support: Strategies for success," in Procedia Comput. Sci., vol. 37, 2014, pp. 422–427. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187705091401028X xx

[10] D. Zikos, "A Framework to Design Successful Clinical Decision Support Systems," in Proc. 10th Int. Conf. PErvasive Technol. Relat. to Assist. Environ., ser. PETRA '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 185–188. [Online]. Available: https://doi.org/10.1145/3056540.3064960 xx

[11] "ISO 25000 STANDARDS," 2011. [Online]. Available: https://iso25000.com/index.php/en/iso-25000-standards xx, xxvi

[12] "Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQUARE) — System and software quality models," p. 34, 2011. [Online]. Available: https://www.iso.org/standard/35733.html xx, xxv, xxvi, xxvii, xxviii, xxix, xxxiv

[13] "Systems and software engineering — Systems and software quality requirements and evaluation (SQUARE) — Measurement of quality in use," p. 41, 2016. [Online]. Available: https://www.iso.org/standard/35746.html xx, xxxi, xxxii

[14] L. Souza-Pereira, S. Ouhbi, and N. Pombo, "Quality-in-use characteristics for clinical decision support system assessment," Comput. Methods Programs Biomed., vol. 207, p. 106169, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260721002431 xxii

[15] ——, "A process model for quality in use evaluation of clinical decision support systems," J. Biomed. Inform., vol. 123, no. September, p. 103917, 2021. [Online]. Available: https://doi.org/10.1016/j.jbi.2021.103917 xxii

[16] P. Mihas, Learn to Use an Exploratory Sequential Mixed Method Design for Instrument Development. SAGE Publications, 2019. [Online]. Available: https://us.sagepub.com/sites/default/files/upm-assets/87148_book_item_87148.pdf xxii

[17] C. Meixner and J. D. Hathcoat, "The nature of mixed methods research," in Handbook of Research Methods in Health Social Sciences, P. Liamputtong, Ed. Springer Singapore, 2019, pp. 51–70. [Online]. Available: https://us.sagepub.com/sites/default/files/upm-assets/87147_book_item_87147.pdf xxii

[18] S. Ssemugabi, "Development and Validation of an Integrated Model for Evaluating E-Service Quality, Usability and User Experience (E-SQUUX) of Web-Based Applications in the Context of a University Web Portal," Ph.D. dissertation, University of South Africa, 2019. xxii

[19] "McCall Software Quality Model |Professionalqa.com," 2016. [Online]. Available: https://www.professionalqa.com/mc-call-software-quality-model xxvi

[20] "Software Quality Factors - Tutorialspoint," 2020. [Online]. Available: https://www.tutorialspoint.com/software_quality_management/software_quality_management_factors.htm xxvi

[21] K. Musa and J. Alkhateeb, "Quality Model Based on Cots Quality Attributes," Int. J. Softw. Eng. Appl., vol. 4, no. 1, pp. 1–8, jan 2013. xxvi

[22] M.-A. Côté, M. Ing, and E. Georgiadou, "Software quality model requirements for software quality engineering," 09 2014. [Online]. Available: https://www.researchgate.net/publication/228947488_Software_Quality_Model_Requirements_for_Software_Quality_Engineering xxvi

[23] F. S. Al-Obthani and A. A. Ameen, "Towards Customized Smart Government Quality Model," Int. J. Softw. Eng. Appl., vol. 9, no. 2, pp. 41–50, 2018. xxvi

[24] A. Abran, ISO 9126: Analysis of Quality Models and Measures. John Wiley & Sons, Inc, 2010, no. September 2010. xxvi

[25] A. Horbst, K. Fink, and G. Goebel, "The ISO/IEC 9126-1 as a supporting means for the system development process of a patient information web service," in Stud. Health Technol. Inform., vol. 116, 2005, pp. 967–970. [Online]. Available: https://www.researchgate.net/publication/7603331_The_ISOIEC_9126-1_as_a_Supporting_Means_for_the_System_Development_Process_of_a_Patient_Information_Web_Service xxvi

[26] N. Bevan, "Classifying and selecting UX and usability measures," in Work. Meaningful Meas. Valid Useful User Exp. Meas., no. 1, 2008, pp. 13–18. xxix

[27] Z. Bukhari, J. Yahaya, and A. Deraman, "A conceptual framework for metrics selection: SMeS," Int. J. Adv. Sci. Eng. Inf. Technol., vol. 8, no. 6, pp. 2294–2300, 2018. xxx

Abstract

Developing or purchasing software is an expensive investment and needs to be justified. Furthermore, the software must be useful in its purpose, reliable, efficient and, among other characteristics, meet the expectations of users [1, 2]. It would be no different in the case of a clinical Decision Support System - CDSS.

CDSS are systems developed to support clinicians and other health professionals in a medical decision making [3]. They are developed within a clinical context, following medical guidelines, with varied purposes such as diagnoses [4, 5, 6] patient monitoring [7, 8, 9], prevention [10] and disease treatment [11, 12]. Conversely, even with all the benefits offered by a CDSS, its acceptance in the medical field is still a matter of debate [13, 14].

The CDSS acceptance is linked to the perception of the end user, such as **1)** the system's ease of use and utility, **2)** the quality of its results and its reliability [14], **3)** the contextual accessibility of the system, sometimes not included in the health professional's routine and workflow, and **4)** the fact that numerous CDSSs are not integrated with existing systems [15].

One manner to extend the use and disseminate positive contributions of CDSSs to the medical world is to develop them in a reliable and useful way. For this, one must follow the best practices of software engineering (SE, acronym in English) [16] and be concerned with its quality, both in the design and development process and in its effective use.

Evaluating the quality of the software is to measure its characteristics and sub-characteristics of quality. In order to better structure the assessment, a series of international standards, with models and frameworks, were developed for assisting software developers in assessing the quality of software products. The latest series is the ISO/IEC 25000 - System and Software Quality Requirements and Evaluation (SQuaRE) [17].

Two of the SQuaRE divisions are addressed in this thesis: 1) Division of quality models standard (ISO/IEC 25010) [18], and 2) Quality measurement division standard (ISO/IEC 25022) [19]. The ISO/IEC 25010 are divided in product quality model and the quality model in use.

Quality in use (QiU), a model of ISO/IEC 25010, is the focus of this study, through its evaluation in the context of a CDSS. The quality in use model refers to the quality of the software when executed, mentioning the result of the interaction between users and the software system/product in a specific context. This model consists of five quality characteristics:

- Effectiveness - means the level of precision and completeness with which users achieve their specific goals when using the system;

- Efficiency - refers to the resources spent to achieve the goals and its measure is related to the level of effectiveness achieved with the consumed resources;
- Satisfaction - refers to whether user requirements are satisfied in a particular context of system use;
- Freedom from risk - refers to the degree to which the quality of a system reduces or avoids potential risks to human life, the economic situation, and health of the environment;
- Context coverage - deals with the use of the system in all specific contexts and/or in contexts that extend beyond the initially identified contexts. Context completeness and flexibility are the sub-characteristics that represent context coverage.

Thus, when measuring the quality of a CDSS, we must consider both the context of use and the choice of the characteristic and sub-characteristic that best suits the purpose of the measurement [20]. The QiU model provides a powerful contribution to the practice of evaluating a system and determining its quality.

According to Harrison *et al.* [21], Effectiveness, Efficiency and Satisfaction are considered the key criteria to reflect the quality of use. Therefore, these QiU characteristics meet the needs and expectations of the users of the systems, in our case of CDSSs, as they consider the user experience.

As a contribution, we proposed a process model to evaluate two QiU characteristics in a CDSS: satisfaction and efficiency. We believe these characteristics are important in the evaluation of a CDSS because, due to its links with the user experience and the usability of the system, when measured, can corroborate the quality of the CDSS and mitigate the non-use and non-acceptance of this type of software. Other contributions from our work are

1) in the academic context, a significant study in the area of software quality, focusing on its characteristics, especially on the quality in use. A guideline for collecting and measuring these characteristics was built into our process model;

2) in the area of software development, professionals can make use of a simple and adaptable process, applicable to other types of systems, to measure the quality-in-use characteristics of their products.

Keywords

Quality-in-use evaluation; quality in use; software quality, evaluation model, ISO/IEC 25010; ISO/IEC 25022; CDSS, clinical decision support system;

Bibliography

[1] M. Khalifa, "Clinical decision support: Strategies for success," in Procedia Comput. Sci., vol. 37, 2014, pp. 422–427. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187705091401028X xli

[2] D. Zikos, "A Framework to Design Successful Clinical Decision Support Systems," in Proc. 10th Int. Conf. PErvasive Technol. Relat. to Assist. Environ., ser. PETRA '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 185–188. [Online]. Available: https://doi.org/10.1145/3056540.3064960 xli

[3] P. Shankar and N. Anderson, "Advances in Sharing Multi-sourced Health Data on Decision Support Science 2016-2017," pp. 16–24, 8 2018. [Online]. Available: http://www.thieme-connect.de/DOI/DOI?10.1055/s-0038-1641215 xli

[4] Y. F. Zhang, L. Gou, T. Shu Zhou, D. Nan Lin, J. Zheng, Y. Li, and et al., "An ontology-based approach to patient follow-up assessment for continuous and personalized chronic disease management," Journal of Biomedical Informatics, vol. 72, pp. 45–59, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1532046417301569 xli

[5] S. Piri, D. Delen, T. Liu, and H. M. Zolbanin, "A data analytics approach to building a clinical decision support system for diabetic retinopathy: Developing and deploying a model ensemble," Decision Support Systems, vol. 101, pp. 12–27, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167923617300908 xli

[6] G. B. Laleci Erturkmen, M. Yuksel, B. Sarigul, T. N. Arvanitis, P. Lindman, R. Chen, and et.al., "A Collaborative Platform for Management of Chronic Diseases via Guideline-Driven Individualized Care Plans," Computational and Structural Biotechnology Journal, vol. 17, pp. 869–885, 1 2019. [Online]. Available: https://doi.org/10.1016/j.csbj.2019.06.003 xli

[7] F. Ali, S. M. R. Islam, D. Kwak, P. Khan, N. Ullah, S.-j. jo Yoo, and K. S. Kwak, "Type-2 fuzzy ontology–aided recommendation systems for IoT–based healthcare," Computer Communications, vol. 119, pp. 138–155, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366417310587 xli

[8] M. Peleg, Y. Shahar, S. Quaglini, T. Broens, R. Budasu, N. Fung, and et al., "Assessment of a personalized and distributed patient guidance system," International Journal of Medical Informatics, vol. 101, pp. 108–130, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1386505617300497 xli

[9] N. Pombo, P. Rebelo, P. Araújo, and J. Viana, "Design and evaluation of a decision support system for pain management based on data imputation and statistical models," Measurement: Journal of the International Measurement Confederation, vol. 93, pp. 480–489, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0263224116303682 xli

[10] A. Jimenez-Molina, J. Gaete-Villegas, and J. Fuentes, "ProFUSO: Business process and ontology-based framework to develop ubiquitous computing support systems for chronic patients' management," Journal of Biomedical Informatics, vol. 82, pp. 106–127, 6 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1532046418300650 xli

[11] E. Shalom, Y. Shahar, and E. Lunenfeld, "An architecture for a continuous, user-driven, and data-driven application of clinical guidelines and its evaluation," Journal of Biomedical Informatics, vol. 59, pp. 130–148, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1532046415002439 xli

[12] M. Afzal, M. Hussain, W. A. Khan, T. Ali, S. Lee, E.-N. Huh, and et al., "Comprehensible knowledge model creation for cancer treatment decision making," Computers in Biology and Medicine, vol. 82, pp. 119–129, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010482517300094 xli

[13] S. Khairat, D. Marc, W. Crosby, and A. Al Sanousi, "Reasons for physicians not adopting clinical decision support systems: Critical analysis," J. Med. Internet Res., vol. 20, no. 4, 2018. xli

[14] K. A. Alshare, M. K. Alomari, P. L. Lane, and R. D. Freeze, "Development and determinants of end-user intention: usage of expert systems," J. Syst. Inf. Technol., vol. 21, no. 2, pp. 166–185, 2019. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/JSIT-08-2018-0108/full/html xli

[15] C. J. Schuh, J. S. De Bruin, and W. Seeling, "Acceptability and difficulties of (fuzzy) decision support systems in clinical practice," in Proc. 2013 Jt. IFSA World Congr. NAFIPS Annu. Meet. IFSA/NAFIPS 2013, 2013, pp. 257–262. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6608409 xli

[16] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, and N. Garcia, "Clinical decision support systems for chronic diseases: A Systematic literature review," Computer Methods and Programs in Biomedicine, vol. 195, 2020. [Online]. Available: https://doi.org/10.1016/j.cmpb.2020.105565 xli

[17] "ISO 25000 STANDARDS," 2011. [Online]. Available: https://iso25000.com/index.php/en/iso-25000-standards xli

[18] "Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQUARE) — System and software quality models," p. 34, 2011. [Online]. Available: https://www.iso.org/standard/35733.html xli

[19] "Systems and software engineering — Systems and software quality requirements and evaluation (SQUARE) — Measurement of quality in use," p. 41, 2016. [Online]. Available: https://www.iso.org/standard/35746.html xli

[20] L. Souza-Pereira, S. Ouhbi, and N. Pombo, "Quality-in-use characteristics for clinical decision support system assessment," Comput. Methods Programs Biomed., vol. 207, p.

106169, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260721002431 xlii

[21] R. Harrison, D. Flood, and D. Duce, "Usability of mobile applications: literature review and rationale for a new usability model," J. Interact. Sci., vol. 1, no. 1, p. 1, 2013. xlii

Contents

List of Figures

List of Tables

Acronyms

ACM	Association for Computing Machinery
AI	Artificial Intelligence
ANN	Artificial Neural Network
ALLab	Assisted Living Computing and Telecommunications Laboratory
APP	Application - Software application
CDSS	Clinical Decision Support System
CDSSs	Clinical Decision Support Systems
EHR	Electronic Health Record
EMR	Electronic Medical Record
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
ISO	Organisation of Standardisation
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
GQM	Goal, Questions and Metrics
GQMdb	Goal-question-metric database
ML	Machine Learning
PHR	Personal Health Record
QiU	Quality-in-Use
RF	Random Forest
RQ	Research Question
SE	Software Engineering
SLR	Systematic Literature Review
SQUARE	System and Software Quality Requirements and Evaluation
SVM	Support Vector Machine
UBI	Universidade da Beira Interior
UX	User Experience
UXdb	User-Experience database

Chapter 1

Introduction

This chapter presents a introduction of the thesis entitled "A process model for quality in use evaluation on clinical decision support system", with the topics **1)** the thesis focus and scope, defining the problem addressed; **2)** the objectives of the research; **3)** the definition of the argument; **4)** the main contributions of this study, and, **5)** the thesis organisation.

1.1 Thesis Focus and Scope

Computerised systems in the medical area are used to provide better management in the clinical care of patients, whether or not they are chronic, and in the search to facilitate and support the work of the health professional. As an example of one of these systems, we have the Clinical Decision Support System (CDSS).

CDSS is a tool that seeks to improve the essential knowledge of health professionals and also patients, through support in clinical decision making [1, 2, 3]. CDSSs provide your user with specific knowledge and information, intelligently filtered or presented at appropriate times [4]. It is expected that these systems can both provide better clinical guidance and empower a patient in terms of awareness and knowledge of their clinical condition.
Conversely, despite all the benefits offered by CDSS, its acceptance in the medical field is still a matter of debate [5, 6]. The acceptance and use of a CDSS is because of factors such as the perception of the end user about the ease of use and usefulness of the system, as well as the quality and reliability of the results provided [6]. Other reasons would be [7, 8]:

1) The context of the system, which sometimes meets a single purpose and seeks to solve a single problem, i.e., the systems deal with the clinical follow-up of a single medical species (e.g. cancer), not considering associated comorbidities (e.g. chronic pain.);

2) Including mandatory activities for the execution of the system outside the routine and workflow of the health professional (e.g., the mandatory entry of a series of clinical information before the generation of a diagnosis);

3) Systems do not treat interoperability, i.e., are not integrated into systems already in use, generating duplicate data entry.

An advantageous CDSS should produce reliable results, meet user expectations, and be useful[9, 10]. One way to ensure these characteristics would be to develop it following the best practices of software engineering (SE) [8]. One way to verify that these features are incorporated into the software product is to evaluate its quality characteristics.
Measuring the quality characteristics of a system can contribute to mitigating the reasons

for non-acceptance or reduced use, identifying deficiencies and weaknesses that affect its reliability, its failure to meet the expectations and needs of users, or even identifying problems in its development.

Therefore, it is very important to investigate the relevant characteristics to evaluate the quality of the software product that allow to validate the context of the proposal and justify the benefits of its use.

To indicate the degree to which the system meets the needs of interested parties, whether these needs are stated or implied, international standards of concepts, procedures, and quality metrics were created to measure the quality of software and systems [11]. The latest series of standards is System and Software Quality Requirements and Assessment (ISO/IEC 25000) - *System and Software Quality Requirements and Evaluation* (SQUARE) [11], being divided into models for requirements, management, measurement, and quality evaluation of software systems and products.

The ISO/IEC 25010 [12] and the ISO/IEC 25022 [13] are two of these divisions, and the first [12] deals with the specification of the models **1)** of software product quality and **2)** of quality in use (QiU) of the system, and the second model [13] deals with guidelines and rules for measuring QiU.

Due to time and budget constraints, it is difficult to evaluate all quality characteristics. Therefore, one must identify the critical quality points that need to be evaluated in the system or software product.
The scope of this thesis is limited to computer systems that support clinical decision-making (e.g., CDSS) and evaluation of the quality in use of these systems.
The focus is on measuring QiU characteristics to improve the acceptance and use of these systems by health professionals. This measurement seeks to favour the identification of potential flaws in this software that justifies its non-acceptance. Such recognition allows fixes and/or improvements to the software, contributing to minimise the problem of its low use.

1.2 Problem Description and Research Objectives

The problem addressed is about the use of CDSS by health professionals and the search for quality assessment in the use of these systems as a way of contributing to the expansion of their use.
At this stage, **1)** the purposes of these systems; **2)** the methodology used in development; **3)** the use of Software Engineering (SE) techniques in its construction; **4)** as clinical information was collected and worked; **5)** as these systems were tested and validated.
As a result of the initial research, an absence was observed in the use's demonstration of SE resources, such as the explanation of development phases and design of structures [8]. It was also observed that the interoperability of CDSSs with other systems already used by

users (e.g., electronic health record (EHR)) was not an in-depth issue in the studies or was even mentioned. Interoperability between systems is an important issue for the acceptance of CDSS among health professionals [8].

For the intermediate phase of the study, points related to **1.** which qualities of CDSS are evaluated; **2.** which quality characteristics are measured; **3.** which critical points of CDSS quality are related to its use by health professionals [14]. Besides the answers to these topics, it was observed that the user experience (UX) was important for the acceptance of CDSS.

The primary objective of this thesis was to create a process model to measure the quality-in-use characteristics of a CDSS to identify problems in the systems that would contribute to its non-acceptance among health professionals. This creation followed the ISO/IEC 25000 series of standards, according to ISO/IEC 25010 and ISO/IEC 25022, described in Chapter IV and [15].
Other objectives were:

- Investigate technologies used in the construction of a CDSS, as well as its applications;
- Measure the quality in use of a prototype or CDSS application;
- Prepare report with suggestions for reconstruction and improvement of CDSS based on weaknesses observed during quality measurement;

1.3 Thesis argument

This thesis proposes a quality assessment model using a clinical decision support system. The research methodology comprises approach, design, and methods. The approach used was mixed, with qualitative and quantitative aspects being applied during the development of the thesis [16].
The design of the research follows the sequential exploratory pattern, where there is the combination of data collection and analysis (quantitative and qualitative) along a sequence of phases [17, 18].
The search methods are divided into phases:

1) **Qualitative phase** - comprising a systematic literature review (chapter 2), research on the concepts of standards, models, and software quality characteristics (chapter 3), and the development of the process model for evaluation of quality-in-use characteristics of a CDSS (chapter 4);

2) **Quantitative phase** - comprising applying the evaluation process template to a CDSS (chapter 5);

3) **Mixed phase** - comprising the analysis and interpretation of the values achieved in the measurement of quality characteristics in use during the application of the evaluation model.

1.4 Major Contributions

The main contributions of this thesis are distributed throughout the document in four chapters. They are articles published in international journals. In all of them were used the platforms Mendeley[1] and Ryyan[2] for the organisation of the candidate studies during the selection process. Microsoft Excel[3] and Microsoft Visio[3] were also used to generate the charts and figures presented in the articles.

The first contribution is a Systematic Literature Review (SLR) [8] with the description of existing approaches to CDSS development methodologies, purposes, SE structures used in their designs, way of collecting clinical information and methods used for the validation of systems. This study is detailed in chapter 2, which consists of an article published in *Computer Methods and Programs in Biomedicine Journal.*

The second contribution of this thesis is the description of software quality models, especially those addressed by the international series of ISO/IEC 25010 standards: *Quality Models Division* [14]. In order to expand the use of CDSS among health professionals, we also identified QiU characteristics to be evaluated in these systems. This is an article published in *Computer Methods and Programs in Biomedicine Journal* and described in chapter 3.

The third contribution, considered as the main aim, is the proposal of a process model for the evaluation of The QiU of a CDSS, based on studies of previous chapters and other bibliographic studies [15]. This contribution is presented in the chapter 4, in an article published by the magazine *Journal of Biomedical Informatics.*

The fourth contribution is the validation of the model developed through its application for quality measurement in use of a CDSS of the Oncology area [19]. The idea was to apply the proposed process model on the evaluation of a CDSS at a hospital, *in loco*, in the city of Covilhã. However, with the advent of the pandemic, this has become unenforceable. The solution found for the validation of the model was to use it to evaluate a CDSS application through the contribution of professionals and medical students from Portugal. The results of this evaluation were described in an article published in *Journal of King Saud University - Computer and Information Sciences.* This part of the study is detailed in chapter 5.

Other contributions concern

1) the focus of software quality characteristics, especially the quality in use, collaborating in the academic context with study material and information, in addition to having presented a guideline for collecting and measuring the quality characteristics through the developed process model;

[1] www.mendeley.com/
[2] https://rayyan.qcri.org
[3] Microsoft Corporation ®

2) a simple and adaptable QiU characteristics assessment process, applicable to other types of systems, as a way to collaborate with software development professionals during the quality control phase of their products.

1.5 Thesis Organisation

This thesis is organised in six main chapters. Except for the first, and sixth chapters, which presenting the introduction, and conclusions and future work, each of the main chapters is formed by an article published in international journals indexed in ISI.
The subjects and organisation of the main chapters of this thesis can be summarised as follows.

Chapter 1 describes the context of this thesis, explaining the scope and focus of the research work and presenting the problem addressed by the thesis and the objectives to be accomplished. In addition, the thesis statement and the adopted approach for solving the problem. The major contributions of this thesis are also included in a summarised form, followed by the description of the organisation and structure of the document.

Chapter 2 provides a systematic review of a published work on software engineering techniques and structures applied in the design of CDSSs for chronic diseases, as well as the description of existing approaches to CDSS development methodologies, purposes, form of collection of clinical information and methods used to validate those systems.

Chapter 3 brings another published work on context of software quality models, especially those addressed by the international series of ISO/IEC 25010 standards: *Quality Models Division* [12]. In order to expand the use of CDSS among health professionals, we also identified QIU characteristics to be evaluated on these systems.

Chapter 4 presents a published work with a proposed process model to evaluate the quality in use characteristics of a CDSS, with the aim of identifying problems with the system that can contribute to a non-acceptance of CDSS by the end-users, and offers alternatives to fix them. It was used the Goal-Question-Metrics (GQM) method was in an adapted form to gather the characteristics am metrics to be used to evaluate the usage scenarios of the CDSS. All the measurements followed the international ISO/IEC 25022 standard: *Measurement of quality in use* [13].

Chapter 5 provides a published work with results of the application of the proposed process method, with the measures and analysis approached by online questionnaire to health professionals and medical students when using a CDSS of oncology area. The results reached, as well as the values found, elucidated the viability of the proposed process model for the measurement of quality in use characteristics of a CDSS and confirm its significance in the identification of quality characteristics that would affect the use of the systems by end-users.

Chapter 6 presents the conclusion of this work, with the most important conclusions and contributions of this thesis and directions for future research work.

Bibliography

[1] R. T. Sutton, D. Pincock, D. C. Baumgart, D. C. Sadowski, R. N. Fedorak, and K. I. Kroeker, "An overview of clinical decision support systems: benefits, risks, and strategies for success," npj Digit. Med., vol. 3, no. 1, pp. 1–10, 2020. [Online]. Available: http://dx.doi.org/10.1038/s41746-020-0221-y 1

[2] M. A. Balakhontceva, A. A. Funkner, A. A. Semakova, O. G. Metsker, N. E. Zvartau, A. N. Yakovlev, A. E. Lutsenko, and S. V. Kovalchuk, "Holistic Modeling of Chronic Diseases for Recommendation Elaboration and Decision Making," Procedia Computer Science, vol. 138, pp. 228–237, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050918316661 1

[3] E. S. Berner, Clinical Decision Support Systems, 2nd ed., ser. Health Informatics. New York, NY: Springer New York, 2007. [Online]. Available: http://link.springer.com/10.1007/978-0-387-38319-4 1

[4] P. Shankar and N. Anderson, "Advances in Sharing Multi-sourced Health Data on Decision Support Science 2016-2017," pp. 16–24, 8 2018. [Online]. Available: http://www.thieme-connect.de/DOI/DOI?10.1055/s-0038-1641215 1

[5] S. Khairat, D. Marc, W. Crosby, and A. Al Sanousi, "Reasons for physicians not adopting clinical decision support systems: Critical analysis," J. Med. Internet Res., vol. 20, no. 4, 2018. 1

[6] K. A. Alshare, M. K. Alomari, P. L. Lane, and R. D. Freeze, "Development and determinants of end-user intention: usage of expert systems," J. Syst. Inf. Technol., vol. 21, no. 2, pp. 166–185, 2019. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/JSIT-08-2018-0108/full/html 1

[7] C. J. Schuh, J. S. De Bruin, and W. Seeling, "Acceptability and difficulties of (fuzzy) decision support systems in clinical practice," in Proc. 2013 Jt. IFSA World Congr. NAFIPS Annu. Meet. IFSA/NAFIPS 2013, 2013, pp. 257–262. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6608409 1

[8] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, and N. Garcia, "Clinical decision support systems for chronic diseases: A Systematic literature review," Computer Methods and Programs in Biomedicine, vol. 195, 2020. [Online]. Available: https://doi.org/10.1016/j.cmpb.2020.105565 1, 2, 3, 4

[9] M. Khalifa, "Clinical decision support: Strategies for success," in Procedia Comput. Sci., vol. 37, 2014, pp. 422–427. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187705091401028X 1

[10] D. Zikos, "A Framework to Design Successful Clinical Decision Support Systems," in Proc. 10th Int. Conf. PErvasive Technol. Relat. to Assist. Environ., ser. PETRA '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 185–188. [Online]. Available: https://doi.org/10.1145/3056540.3064960 1

[11] "ISO 25000 STANDARDS," 2011. [Online]. Available: https://iso25000.com/index.php/en/iso-25000-standards 2

[12] "Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQUARE) — System and software quality models," p. 34, 2011. [Online]. Available: https://www.iso.org/standard/35733.html 2, 5

[13] "Systems and software engineering — Systems and software quality requirements and evaluation (SQUARE) — Measurement of quality in use," p. 41, 2016. [Online]. Available: https://www.iso.org/standard/35746.html 2, 5

[14] L. Souza-Pereira, S. Ouhbi, and N. Pombo, "Quality-in-use characteristics for clinical decision support system assessment," Comput. Methods Programs Biomed., vol. 207, p. 106169, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260721002431 3, 4

[15] ——, "A process model for quality in use evaluation of clinical decision support systems," J. Biomed. Inform., vol. 123, no. September, p. 103917, 2021. [Online]. Available: https://doi.org/10.1016/j.jbi.2021.103917 3, 4

[16] P. Mihas, Learn to Use an Exploratory Sequential Mixed Method Design for Instrument Development. SAGE Publications, 2019. [Online]. Available: https://us.sagepub.com/sites/default/files/upm-assets/87148_book_item_87148.pdf 3

[17] C. Meixner and J. D. Hathcoat, "The nature of mixed methods research," in Handbook of Research Methods in Health Social Sciences, P. Liamputtong, Ed. Springer Singapore, 2019, pp. 51–70. [Online]. Available: https://us.sagepub.com/sites/default/files/upm-assets/87147_book_item_87147.pdf 3

[18] S. Ssemugabi, "Development and Validation of an Integrated Model for Evaluating E-Service Quality, Usability and User Experience (E-SQUUX) of Web-Based Applications in the Context of a University Web Portal," Ph.D. dissertation, University of South Africa, 2019. 3

[19] 4

Chapter 2

Clinical Decision Support Systems for Chronic Diseases: A Systematic Literature Review

The chapter presents and discusses concepts of state of the art of clinical decision support systems focusing in software engineering techniques and evaluation methods.

The chapter is presented as an article named “Clinical decision support systems for chronic diseases: a systematic literature review.”

Leonice SouzaPereira, Nuno Pombo, Sofia Ouhbi, Virginie Felizardo and Nuno Garcia.

Published in Computer Methods and Programs in Biomedicine, 2020, in the area of Computer Science.

H−Index 95, Q1, by Scimago Journal & Country Rank.

Computer Methods and Programs in Biomedicine 195 (2020) 105565

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Clinical decision support systems for chronic diseases: A Systematic literature review

Leonice Souza-Pereira[a,b,c,1,2,*], Nuno Pombo[b,c], Sofia Ouhbi[d], Virginie Felizardo[b,c], Nuno Garcia[b,c]

[a] Instituto Federal do Triângulo Mineiro - Campus Uberlândia Centro, Brasil
[b] Instituto de Telecomunicações, Lisboa, Portugal
[c] Universidade da Beira Interior, Covilhã, Portugal
[d] Computer Science and Software Engineering Department, CIT, UAE University, UAE

ARTICLE INFO

Article history:
Received 17 February 2020
Revised 24 April 2020
Accepted 21 May 2020

Keywords:
Chronic disease
Clinical decision support systems
Software engineering,

ABSTRACT

A Clinical Decision Support System (CDSS) aims to assist physicians, nurses and other professionals in decision-making related to the patient's clinical condition. CDSSs deal with pertinent and critical data, and special care should be taken in their design to ensure the development of usable, secure and reliable tools. **Objective:** This paper aims to investigate existing literature dealing with the development process of CDSSs for monitoring chronic diseases, analysing their functionalities and characteristics, and the software engineering representation in their design. **Methods:** A systematic literature review (SLR) is conducted to analyse the literature on CDSSs for monitoring chronic diseases and the application of software engineering techniques in their design. **Results:** Fourteen included studies revealed that the most addressed disease was **diabetes** (42.8%) and the most commonly proposed approach was **diagnostic** (85.7%). Regarding data sources, the studies show a predominance on the use of **databases** (85.7%), with other data sources such as **sensors** (42.8%) and **self-report** (28.6%) also being considered. Analysing the representation for engineering techniques, we found **Behaviour diagrams** (42.8%) to be the most frequent, closely followed by **Structural diagrams** (35.7%) and **others** (78.6%) being largely mentioned. Some studies also approached the **requirement** specification (21.4%). The most common target evaluation was the **performance** of the system (64.2%) and the most common metric was **accuracy** (57.1%). **Conclusion**: We conclude that software engineering, in its completeness, has scarce representation in studies focused on the development of CDSSs for chronic diseases.

1. Introduction

Chronic disease is a condition that persists for a long period, usually three months or more [1,2]. It cannot be prevented by vaccines or cured by medication, and it is often characterised by a complex combination of comorbidities in the ageing patient population [3,4]. Chronic diseases include conditions such as heart disease, stroke, respiratory diseases, cancer and diabetes. In high-income countries, its occurrence accounts not only for billions of dollars in annual medical expenditures but also for indirect costs such as a negative impact on the quality of life, and decreased worker productivity [5]. Furthermore, chronic diseases are the leading causes of death in the USA, as heart disease accounts 1/3 of deaths, and cancer causes around 600.000 deaths per year [6]. This is also dramatic in Europe, as heart disease remains a major cause of death, causing around 800,000 deaths annually [7]. In the last years, emerging and disruptive technologies paved the way for the adoption of computerised systems in healthcare. This innovation boom also included the development of clinical decision support systems (CDSS) as a tool to enhance practitioners (and patients) with the essential knowledge to support clinical decision-making [8]. According to Shankar *et al.* [9] a CDSS *"provides physicians, employees, patients or other individuals with knowledge and information specific to each person, intelligently filtered or presented at appropriate times, to improve health and health care"*. In fact, it is expected that these systems may not only provide a better clinical orientation but also empower patients in terms of awareness and deeper knowledge of their condition. In other words, the goal of a CDSS is to make better use of the existing data and to extend the

* Corresponding author at: Universidade da Beira Interior
E-mail address: leonice.pereira@ubi.pt (L. Souza-Pereira).
[1] Computer Science Department - University of Beira Interior
[2] Add: Marquês de Ávila e Bolama Street, 6201-001, Covilhã - Portugal

https://doi.org/10.1016/j.cmpb.2020.105565
0169-2607/

information on which decisions are based. Since chronic diseases require a medium-long term capability to collect, to process, and to analyse data in continuous or at least in a recurrent manner, it becomes clear that developing CDSS with the before-mentioned abilities is a challenging and critical activity because any failure may cause significant or critical problems to the users and the health systems. For that reason, software engineering (SE) principles are desirable throughout the CDSS development life-cycle to enable software engineers to cope with the increasing complexity of current health systems requirements. As defined in [10], the SE is *"the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software"*. Therefore, as the CDSS can represent the cornerstone of clinical decision-making, it is opportune to analyse whether its development procedure followed the best practices in SE, since the guarantee of a well-developed and effective software depends on good Software Engineering. It is also relevant to determine its ability to produce accurate outputs in response to urgent and complex clinical problems. With these principles in mind, this systematic literature review (SLR) aims to study the existing literature on CDSSs for chronic diseases, focusing on SE principles and techniques during the CDSS development life-cycle, as well as machine learning techniques for acquisition of knowledge. The reason for the correlation between CDSS and Software Engineering lies in the opportunity for studies and research, as it is a large and under studied area, which could bring improvements and responses to existing gaps and future needs.

1.1. Machine learning

Artificial Intelligence (AI) is globally accepted and used for designing medical applications to support medical practitioners in diagnosing and treating patients effectively and efficiently. Machine Learning (ML) is one of the most relevant emerging AI techniques, giving systems the ability to extract new medical and clinical knowledge from existing information.

ML has revolutionised the use of large and complex datasets within computational models. However, this technological development that allows us to handle big data and multidimensional data structures to deliver reliable and accurate performance is still challenging. The scientific advances in ML approaches is encouraging and studies show promising results and useful applications [11] also in the CDSS area, making these systems more accurate in the support and management of complex treatments, and also reducing the instability between medical diagnosis by achieving better clinical results [12].

This SLR found ML Techniques in the select studies, especially regarding the use of metrics that proved the efficiency of their respective models and systems, *e.g.* Support Vector Machine (SVM) [13], Artificial Neural Network (ANN) [14] and Random Forest (RF) [15]. Subsection 3.3 presents the Machine Learning Techniques and the studies that used them.

1.2. Related work

To the best of our knowledge, this is the first systematic review that brings together chronic diseases, CDSS and software engineering (SE) techniques. In [16], the authors presented a systematic review focused on the evaluation of different models observed in the life-cycle of ubiquitous software development. Contrariwise, [17] and [18] focused their systematic review on CDSS for specific chronic diseases - cancer and cardiovascular diseases. Finally, in [19], the authors performed a systematic review showing the benefits of CDSSs when applied outside academic centres. The researchers sought to evaluate the effect of CDSSs on clinical outcomes, health care processes, and other characteristics for both commercially developed and locally developed CDSS (in clinical centres). As a result, the authors affirmed that evidence of clinical, economic, workload and efficiency results remained scarce.

1.3. Organisation

The remainder of this paper is organised as follows. Section 2 summarises the review methodology used in this paper. Section 3 presents the results of the review, including the SE principles as well as evaluation techniques and the metrics found in the selected studies. Section 4 discusses the main findings and presents the implications and limitations of this work. Finally, we conclude in Section 5.

2. Methods

This section explains the methodology used for conducting this review.

2.1. Research questions

This SLR focuses on assessing SE principles used in the development of CDSSs for chronic diseases, and aims to answer five research questions (RQ) as presented in Table 1.

2.2. Search strategy

The search for articles was made with a query in ScienceDirect, ACM and IEEE Xplore databases, described as follows:

("care decision support system" **OR** "clinical decision support system" **OR** "CDSS"**) AND (**"pain" **OR** "chronic disease"**).**

The use of the keyword "pain" alongside with "chronic disease" is related to the fact that very often pain is not regarded as a chronic disease in itself, when compared to other types of chronic diseases such as *e.g.* diabetes. Yet, there are a large number of chronic conditions that are mostly manifested through pain. Moreover, pain is considered the fifth vital sign for indicating basic bodily functions, health and quality of life [5,20,21]. Pain is defined chronic when it persists over a long period of time, usually more than 3 months. Furthermore, most of the times, chronic pain is associated with other chronic diseases [5] such as cancer. So, the term *pain* was included in the search string because it allows the identification of research papers that otherwise could be missed, being the case of those referring to pain when this is the main symptom but its cause is not always known [22].

Two reviewers evaluated the studies, and another was considered as an adjudicator in case of divergence of opinion, which was not necessary.

The rationale to choose the ScienceDirect, ACM and IEEE Xplore databases lied in the fact that they are focused on computer science, including software and information systems and its applications (*e.g.* medical field as the CDSS). Additional databases were searched such as *e.g.* Scopus, however the findings were overlapped with the results obtained from the previous research.

2.3. Eligibility criteria

The studies included in this review attended the following inclusion criteria. (1) Presented/proposed a CDSS, (2) related to chronic disease management, (3) were published between January 2010 and August 2019 (4) in journals and conferences indexed in the previously mentioned scientific databases.

Moreover, studies that met any exclusion criteria (EC) below were not considered in this SLR:

- EC1 - studies not focused on chronic diseases or chronic pain;

Table 1
Research Questions.

RQ-ID	Question	Reason
RQ1	What are the motivations are behind the development of the CDSS?	This question intends to reach the principal goals of the CDSS and its development context.
RQ2	What are the SE stages that are followed along the complete CDSS development life-cycle?	This question aims to bring the current state of the application of principles in SE to light.
RQ3	How data are used?	The answer to this question aims to identify methods to collect data, as well as ML techniques to infer knowledge based on such data.
RQ4	What are the methodologies used to evaluate the CDSS?	Here the intention is to identity which evaluation techniques were used on each CDSS.
RQ5	In what ways have the authors proved the validity of the developed CDSS?	This question aims to raise the current state on the use of metrics in the validation activity. The goal is to determine the effectiveness of the CDSS to improve decision-making.

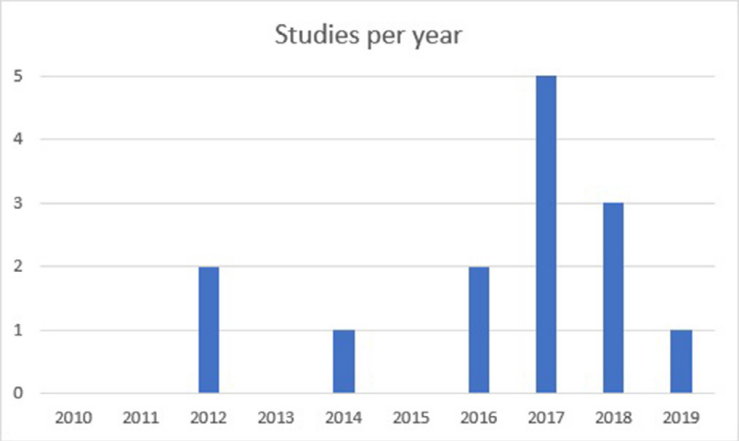

Fig. 1. Studies from 2010 to 2019.

- EC2 - studies that do not describe at least one stage of the software engineering process;
- EC3 - studies in conceptual stage or results.

3. Results

The selection procedure followed the PRISMA statement [23]. As depicted in Fig. 2, our query identified 343 studies in which 329 papers were found in ScienceDirect, 3 (three) studies in ACM and 11 (eleven) in IEEE Xplore. After removing duplicates (n=4), the majority of the studies (n=249) were discarded in an analysis of its title and abstract. Thus, 90 (ninety) full-text papers were eligible for inclusion, and of these, an additional 76 (seventy-six) studies were excluded. Thirty-four studies were not focused on chronic diseases or pain (EC1); an additional thirty-four papers did not describe any software engineering process (EC2); at last, eight studies were discarded because they were presented as in a conceptual stage, or did not present results (EC3). Finally, 14 (fourteen) studies were selected for this SLR as presented in Table 2. In addition, the publication trend of included papers is depicted in Fig. 1, showing an increased relevance of the topic in the last four years.

3.1. Motivation for the development of a CDSS

There are a multitude of complementary purposes that may be aimed by a CDSS, such as follow-up management, prevention or screening, diagnosis, treatment (or guidelines for the treatment procedure), and information management [38]. They are presented in Table 3.

The term "follow-up management" means that the system can monitor the patient's situation and send informative alerts and reminders either to him/her or to the caregiver and/or physician. "Prevention/screening" is used to express the possibility of the patient developing a disease, while "diagnosis" implies the identification of a disease. Treatment and guidelines are associated with drug ordering, prescription testing, or counselling for behaviour change. Finally, "information management" refers to documentation and data submission forms.

Table 3 shows that 35.7% of the studies aimed to follow-up and provide treatment/guidelines; 14.3% sought to prevent or to screen a disease; 21.4% of the studies were focused on information management and the majority (85.7%) deal with diagnosis.

3.2. Software engineering stages

SE is fundamental to guide the development of reliable systems, regardless of its context. In fact, there is a myriad of models that may guide the development of a software project such as Waterfall, Rapid Prototyping, Spiral, Agile, and Incremental, among others [39]. However, every different model has in common at least four stages, namely the software requirements, design, development and testing tasks, described in the next subsections.

3.2.1. Requirements specification

The requirements engineering stage comprises elicitation and analysis, specification and validation [10]. Elicitation and analysis activities are done in interaction with the stakeholders; specification aims to convert such requirements into a standard form; the validation activity verifies if the requirements define the system according to the stakeholders' expectations [39].

The outcome of this activity is usually a document containing a brief description of the system to be developed as functional requirements (what the system should do) and non-functional requirements (characteristics that the system must have). This

Table 2
Studies included in the SLR, with information about the published year, diseases, number of references in the article and a brief summary of their proposal.

Study	Year	Disease(s)	References	Study Proposal
Laleci *et al.* [24]	2019	Diabetes, Renal, Cardiovascular, Depression	44	The authors presented a method and implementation of a semi-automatic care plan management tool to suggest personalised recommendations for individualised care plans.
Jimenez-Molina *et al.* [25]	2018	Respiratory disease	96	The authors proposed a framework for the development of chronic disease support systems and applications as an answer to shortcomings as the integration issues between applications and existing healthcare systems.
El-Sappagh *et al.* [26]	2018	Diabetes	52	The authors proposed and implemented a new semantically interpretable FRBS framework for diabetes diagnosis using multiple aspects of knowledge-fuzzy inference, ontology reasoning, and a FAHP.
Ali *et al.* [27]	2018	Diabetes	66	Drugs and food are recommended using a based type-2 fuzzy ontologyâaided system, applied on IoT-based healthcare, to efficiently monitor the patientâs body.
Aborokabah et al.[28]	2017	Cardiovascular	36	The authors used support vector machine (SVM) to propose a context-aware clinical decision support model for heart failure risk prediction.
Peleg *et al.* [29]	2017	Gestational Diabetes, Atrium Fibrillation (AF)	43	The authors described and implemented the MobiGuideâs architecture, a project aimed to establish a ubiquitous, user-friendly, patient-centred mobile decision-support system.
Zhang *et al.* [30]	2017	Diabetes	63	The author modified their preview clinical decision support system (CDSS) to improve the efficiency of data access in the context of chronic disease patient follow-up assessment.
Afzal *et al.* [31]	2017	Cancer	54	It was proposed an automated knowledge acquisition methodology directly from documents for cancer treatment.
Piri *et al.* [32]	2017	Diabetes	61	The authors built a CDSS for prediction of diabetic retinopathy.
Shalom *et al.* [33]	2016	Pre-Eclampsia and Toxaemia	44	The authors were concerned with designing, implementing and evaluating a new architecture to support decisions based on continuous realistic guidelines (GL).
Pombo *et al.* [34]	2016	Pain	124	The paper presented a CDSS based on data imputation principles for pain evaluated.
Seixas *et al.* [35]	2014	Alzheimer and Mild Cognitive Impairment	90	The authors proposed a Bayesian network decision model for supporting diagnosis of dementia, AD and Mild Cognitive Impairment (MCI).
Kong *et al.* [36]	2012	Pain	47	This paper described a clinical decision support system (CDSS) for risk stratification of patients with cardiac chest pain.
Anooj P.K[37].	2012	Cardiovascular	39	The author presented a weighted fuzzy rule-based clinical decision support system (CDSS) for the diagnosis of heart disease.

Table 3
Characteristics of the studies according to their purposes.

Study	Follow-up management	Prevention/screening	Diagnosis	Treatment / guidelines	Information management
Laleci *et al.* [24]			√		
Jimenez-Molina *et al.* [25]	√	√	√	√	√
El-Sapparh *et al.* [26]			√		
Ali *et al.* [27]	√		√	√	
Aborokabah et al.[28]			√		
Peleg et al. [29]	√		√	√	
Zhang *et al.* [30]			√		
Afzal *et al.* [31]				√	√
Piri et al. [32]			√		
Shalom *et al.* [33]	√		√	√	√
Pombo *et al.* [34]	√				
Seixas *et al.* [35]			√		
Kong *et al.* [36]		√	√		
Anooj P.K[37].			√		

document will guide the specification of the tasks required to be performed by the software [40].

Although the higher relevance of this stage, our results evidenced that only three studies (21.4%) addressed requirements engineering principles, in particular, Peleg *et al.*[29], Shalom *et al.* [33] and Jimenez-Molina *et al.* [25] presented requirements elicitation and process specification.

Regarding non-functional requirements, these studies presented 1) organisational requirements such as organisation process and clinical pathways, stakeholderâs coordination, data and knowledge management, 2) product requirements, such as distributed computing, efficiency requirements, performance, process orchestration, and 3) external requirements, such as portability and interoperability.

3.2.2. Design

Software design can be seen as the activity in which software requirements are analysed to produce a description of the soft-

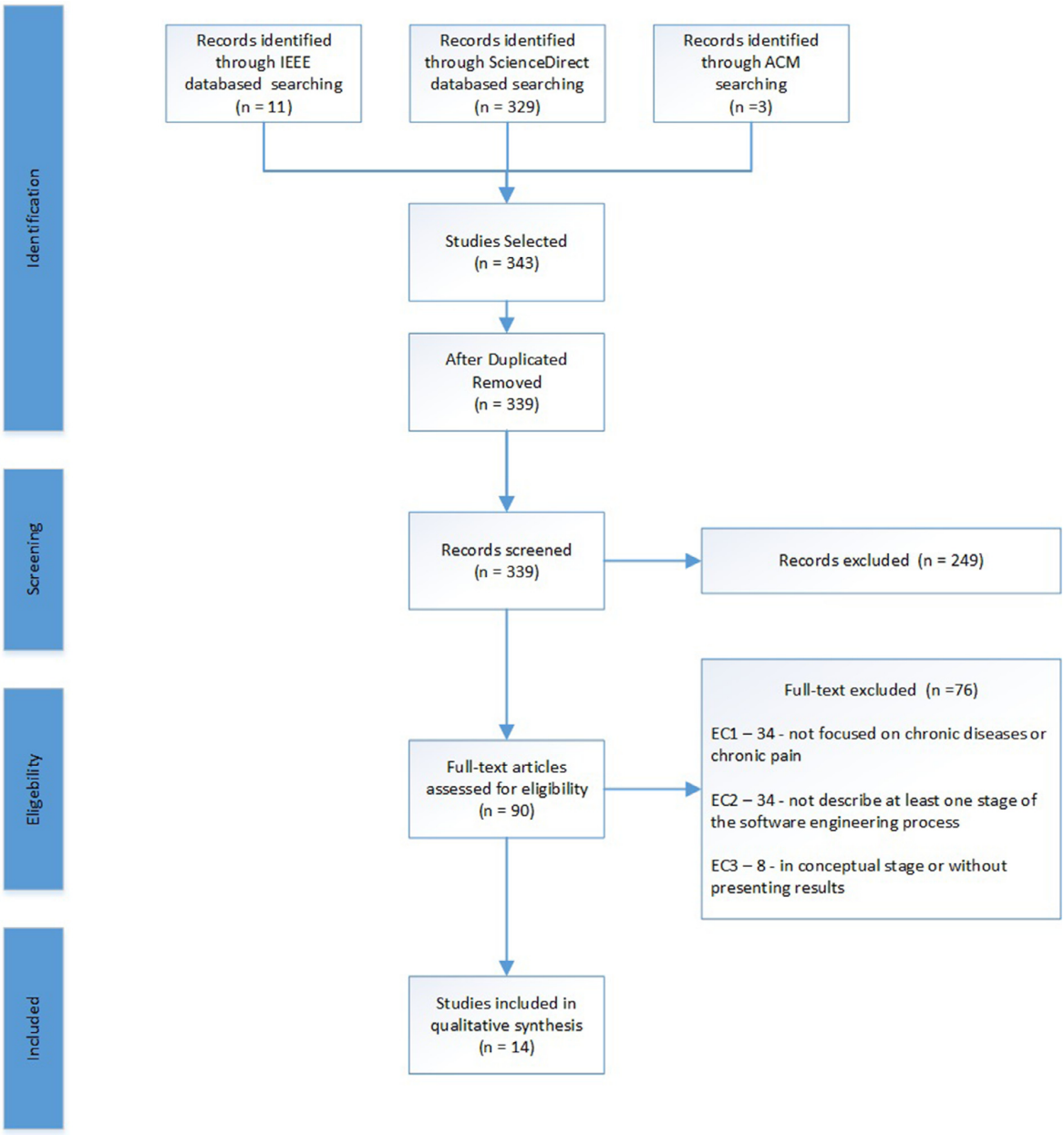

Fig. 2. Study Workflow.

ware's internal structure. The result of this activity are graphical artefacts that describe the software architecture, *i.e.*, it describes how the software is decomposed and organised into components, and also how these components interface with each other and with the surrounding environment. Many notations exist to represent software design artefacts, presenting a different view or perspective of that system. Unified Modelling Language (UML) is a *de facto* standard that allows the visualisation of system artefacts through standardised diagrams such as *e.g.* Structural Diagrams (from static models), Behaviour Diagrams (from dynamic models), and Interaction Diagrams.

Structural Diagrams

Structural diagrams are used to document the static aspects of the system, that is, they represent its stable structures.

The following Structural Diagrams were found in the selected studies:

- Class Diagram - it describes the system structure, showing classes and their attributes, methods and relationships between classes [39];
- Component Diagram - it shows the relationship between different components or modules (or independent systems) in a system [41].

These diagrams were presented in 35.7% of the studies. On the one hand, Jimenez-Molina *et al.* [25], Zhang *et al.* [30], Piri *et al.* [32], and Shalom *et al.* [33] presented Class Diagrams. On the other hand [24] adopted the Component Diagram.

Behaviour Diagrams

Behavioural Diagrams describe the dynamic behaviour of a running system. They can be modelled from the perspective of data processed by the system or events that stimulate system responses [39]. The main Behaviour Diagrams found were:

- Activity Diagram - describes dynamic aspects of the system, modelling the flow of information from one activity to another. It can be used to model data processing, where each activity represents a process step [39];
- State Diagram - show system states and events that cause transitions from one state to another, modelling behaviours of a system in response to internal or external events. Although it does not show the data flow within the system, these diagrams allow the inclusion of additional information about the processing performed in each state [39];
- Use Case Diagram - Use Case modelling is widely used to support requirements and describes what the user expects from the system [39];
- Sequence Diagram - This diagram is used to represent scenarios of interaction for a particular functionality of a system [42], mainly to model interactions between actors and objects from a system [39].

Studies that presented the above-mentioned diagrams account for 42.8% of the total papers, of which [25,30,31,34,35] used Activity Diagrams and [33] presented Sequence Diagrams.

Other diagrams

The system's architecture is the representation of the features of a system, as well as their interactions and restrictions, either internally (software) or external (hardware or users) [43]. This diagram gives the reader a general idea of how your system or model will work.

Other representations (*e.g.* system's architecture), and diagrams (*e.g.* decision tables and communication diagrams), flowcharts, and frameworks were presented in the selected studies. They account for 78.6% of the studies [24–27,30,33–37].

3.2.3. Development

The term software development or construction refers to software creation activities through the combination of coding, verification, unit testing, integration testing and debugging [10]. This software engineering stage is closely linked to the design stage (which provides inputs to the processes) and the testing stage (which uses its outputs). At the same time, it is observed that there is no borderline definition of these stages, since tests of the units occur as they are being built, mainly to certify the integration between the structures and the correct use of the requirements. Only Laleci *et al.* [24] wrote about using unitary tests to check the integration between the components.

Only four studies (28.6%) presented a part of the coding or algorithms, used as an illustration of functions or artefacts of the system. These were Ali *et al.* [27], El-Sappagh *et al.* [26], Jimenez-Molina *et al.* [25] and Zhang *et al.* [30].

3.2.4. Testing

The software test consists of verifying whether a program provides the expected behaviours and outputs with a finite set of test cases, selected in an execution domain [10]. It should be present on the development and maintenance life-cycle, starting with the early stages of the software requirements process. Its planning must be developed continuously and systematically, as software development procedures.

The testing stage within the studies, for the most part, was for performance evaluation and they were addressed in Section 3.5 - Evaluation and Metrics.

3.2.5. Maintenance

The purpose of software maintenance is to modify existing software and preserve its integrity after modification requests.

Such adjustments are recorded and tracked, as well as the impact generated by them. The coding and other software artefacts are modified, and tests are performed, providing a new version of the software. Only Laleci *et al.* [24] have reported to use the testers' feedback to improve the system.

3.3. Machine learning techniques

Machine learning (ML) techniques are divided between supervised learning and unsupervised learning. In supervised learning the training set comprises the desired input and output pairs, and the goal is to learn a mapping between the input and output spaces [44]. When the desired output is not part of the training set, and the output may return uncertain answers, we have unsupervised learning [44].

Machine learning techniques were found in [26,28,31,32,35–37], accounting 50.0% of total. Authors used it for the diagnosis of diseases.

All studies utilised pre-processing techniques, as follows:

- El-Sappagh *et al.* [26] - inputting missing values, identifying outliers, encoding categorical or nominal features;
- Aborokbah *et al.* [28] - minimise all forms of noise, without specifying how;
- Afzal *et al.* [31] - dimensional reduction, missing values imputation;
- Piri *et al.* [32] - information extraction and cleaning, data integration, aggregation, and representation;
- Seixas *et al.* [35] - filter attributes with high missing values ratio, or those that are not relevant;
- Kong *et al.* [36] - filter of specific information;
- Annoj [37] - removing missing values and noisy information.

El-Sappagh *et al.* [26] used The Eclipse IDE[3] with JDK8[4] and MSAccess[5] to develop and test their ML algorithm. The MATLAB Toolkit[6] environment was used on the studies [35–37]. Other studies did not specify which development environment was used.

Characteristics and studies that used ML techniques can be seen in Table 5. Metrics such as accuracy, sensitivity, among others proved the effectiveness of the predictive model that used these techniques and are presented in Table 6.

3.4. Data sources

The data sources found in the selected studies can be classified in three groups:

- Database - This group includes external datasets, used mostly as simulated data to apply and test CDSS models, or Electronic Health Records (EHR) from hospitals when the CDSS is integrated into it;
- Sensors - Data is gathered by sensors somehow connected to patient;
- Self-report - The patient provides directly the necessary input to the system, often using questionnaires or transmitting data via an application or equipment.

Table 4 shows data from databases was used in 85.7% of the studies, while data from sensors was used in 42.8% and self-report was used in 28.6% of the studies.

[3] https://www.eclipse.org/ide/
[4] https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html
[5] https://www.microsoft.com
[6] https://www.mathworks.com/products/matlab.html

Table 4
Characteristics of the studies according to their data source and the software engineering stages presented.

Study	Data Source			Software Engineering stages			
	Database	Sensors	Self- report	Requi- rements	Structural Diagrams	Behaviour Diagrams	Other Diagram
Laleci *et al.* [24]	√	√	√		√		√
Jimenez-Molina *et al.* [25]	√	√		√	√	√	√
El-Sappagh *et al.* [26]	√						√
Ali *et al.* [27]		√					√
Aborokabah et al. [28]	√	√					√
Peleg *et al.* [29]	√	√	√	√			
Zhang *et al.* [30]	√				√	√	√
Afzal *et al.* [31]	√					√	
Piri *et al.* [32]	√				√		
Shalom *et al.* [33]	√	√	√	√	√	√	√
Pombo *et al.* [34]			√			√	√
Seixas *et al.* [35]	√					√	√
Kong *et al.* [36]	√						√
Anooj P. K[37].	√						√

Table 5
Studies and Machine Learning Techniques presented.

Machine Learning Classification	Techniques	Studies
Supervised	Bayesian Network	Seixas*et al.* [35]
	Decision Tree	Piri *et al.* [32] Afzal *et al.* [31]
	Support Vector Machine	Aborokbah *et al.* [28]
	Random Forest	Piri *et al.* [32]
	Logistic Regression	Piri *et al.* [32]
	Neural Network	Piri *et al.* [32]
No-	Ruled Based learning	Kong *et al.* [36]
Supervised	Fuzzy based learning	Anooj P.K[37]. El-Sappagh *et al.* [26]

Table 6
Performance assessment of the studies and their metrics.

Study	Accuracy	Specificity.	Sensitivity.	Correctness.	Completeness.	Precision	F1	P-Value
Laleci *et al.* [24]								p=0.0076
Jimenez-Molina *et al.* [25]	79.30	78.00						
El-Sappagh *et al.* [26]	98.33		98.33		98.75	98.29		
Ali *et al.* [27]	83.00		81.0			97.00		
Aborokabah et al.[28]	82.0	87.5	76.9					
Peleget al. [29]		64.00	64.00					p=0.0312
Zhang *et al.* [30]	99.93	99.98	99.99					
Afzal *et al.* [31]	80.7							
Piriet al. [32]	92.76	95.30	90.22					
Shalom *et al.* [33]				94.50	93.98			
Pombo *et al.* [34]								p <0.05
Seixas *et al.* [35]							80.0	
Kong *et al.* [36]								p=0.0076
Anooj P.K. [37]	57.85	68.75	45.22					

The CDSS usually is interconnected with external or third-party computerised systems (*e.g.* EHR) already used by physicians and health professionals. This may be justified not only by the increasing adoption of CDSS, but also, as a tentative to reduce redundancy, duplicate inputs, and inconsistent information [45]. Additionally, it may have the advantage to provide standardised users' interfaces and interoperability services.

Laleci *et al.* [24], Jimenez-Molina *et al.* [25], and Peleg *et al.* [29] used an EHR as a data source; Zhang *et al.* [30], utilised a dataset.

Aborokbah *et al.* [28], Piri *et al.* [32], Kong *et al.* [36] and Anooj P.K[37]. used datasets as the source of information; El-Sappagh *et al.* [26], Afzal *et al.* [31] and Seixas *et al.* [35], adopted EHR jointly a dataset in their systems and also worked with machine learning techniques.

On the studies [24–26,29], the interoperability approach between the developed system and other existing systems, such as the EHR, was also verified.

3.5. Evaluation and metrics

All parts that concern the development of a system such as requirement, design, development and test, must be implemented following the best practices for quality and efficiency. The System and Software Quality Requirements and Evaluation (SQuaRE) is a series of standards that deal with the software quality (ISO2500) [46] and involve the evaluation part. One of these standards, ISO/IEC 25,010 [47], along with Sommerville [39], presents essential quality characteristics for good software evaluation. Some of them are highlighted as follows:

- Maintainability (Quality characteristic **Maintainability**) - the software must be written according to customer needs and, at the same time, be proper to change. It refers to how easy, safe and cheap is the maintenance of the software;

Table 7
Metrics found in the studies and their mathematical formulas/concepts.

Metric	Formulae	What it means/measure
Accuracy	$\frac{TP+TN}{TP+FP+FN+TN}$	In the Numerator, we have the correct predictions (True positives (TP) and True Negatives(TN)); in the denominator all prediction, the right and the wrong ones. It means the number of correct predictions made by the model when compared with all predictions, *i.e.* how often the classifier hit its predictions.
Sensitivity or Recall	$\frac{TP}{TP+FN}$	It measures the frequency of true positives were correctly predicted.
Specificity	$\frac{TN}{TN+FP}$	It measures the proportion of true negatives that were correctly predicted. It is the opposite of Sensitivity.
Completeness		The system produces all the answers required by its specification[52]; all specified tasks and user objectives were addressed in software production [47].
Correctness		All answers of the system are compatible with its specification [52]; the system provides the correct results with the required degree of accuracy [46].
Precision (Pr)	$\frac{TP}{TP+FP}$	It measures the ratio of model hit to both positive and negative prediction, which matches the reality, i.e. those who were classified as correct, how many were effectively correct.
F1 Score (F1)	$\frac{2*Pr*Sensitivity}{Pr+Sensitivity}$	Combines precision and recall translating overall model quality into a singlenumber.
P-Value		Measure of how much evidence you have against the null hypothesis

- Trust and protection (Quality characteristics **Security and Reliability**) - the software must be reliable and safe against damages in system failure events, invasions and non-allowed access;
- Efficiency (Quality characteristic **Performance**) - system resources such as processing and memory must be used effectively and intelligently;
- Usability (Quality characteristic **Usability**) - The system can be used by users efficiently, effectively and satisfactorily within a specified context of use;
- Interoperability (Quality characteristic **Compatibility**) - in addition to bringing all features they have to offer, the software must also be interoperable with existing systems, to ensure that information is exchanged between systems, products or components.

Laleci *et al.* [24] divided their evaluation into layers, where in the first layer, they applied tests to ensure integration of the components, and in the second layer they used questionnaires to assess the usability, through of the Questionnaire for User Interaction Satisfaction (QUIS)[7] and another questionnaire developed by them.

At the same way, Jimenez-Molina *et al.* [25] used questionnaires to assess the usability and the usefulness of the system. To measure the functionality and feasibility of the developed framework, the authors used the data transmission overload technique to achieve a good throughput performance.

The study from Peleg *et al.* [29] used only an original questionnaire to assess the user's satisfaction. Peleg *et al.* [29] also sought to assess the patient's quality of life by applying questionnaires EuroQoL[8] and AF Effect in QualiTy of life survey (AFEQT) [48].

Most of the studies sought to evaluate the performance of the system. El-Sappagh *et al.* [26] and Afzal *et al.* [31] used external validation (external data) to validate the system performance.

The Pellet reasoner [49] was used by Ali *et al.* [27] and Zhang *et al.* [30], which also used expert's evaluation [50,51] alongside Shalom *et al.* [33].

To verify their system performance, an internal validation (K-fold Cross Validation) was used by Aborokbah *et al.* [28](k=7), Seixas *et al.* [35] (k=4), Kong *et al.* [36] (k=10) and Anooj P.K[37]. (k=10), while Piri *et al.* [32] used Hold-out (60/40). Pombo *et al.* [34] used statistical methods to demonstrate this. At last, tools for program testing and validation, and defect testing were found in Shalom *et al.* [33]. Fig. 3 shows studies and what they sought to validate. The performance assessment of the selected studies is presented in Table 6 and definition of its metrics can be seen in Table 7.

[7] https://isr.umd.edu/news/story/quis-questionnaire-for-user-interaction-satisfaction-70-isr-ip

[8] http://www.euroqol.org/)

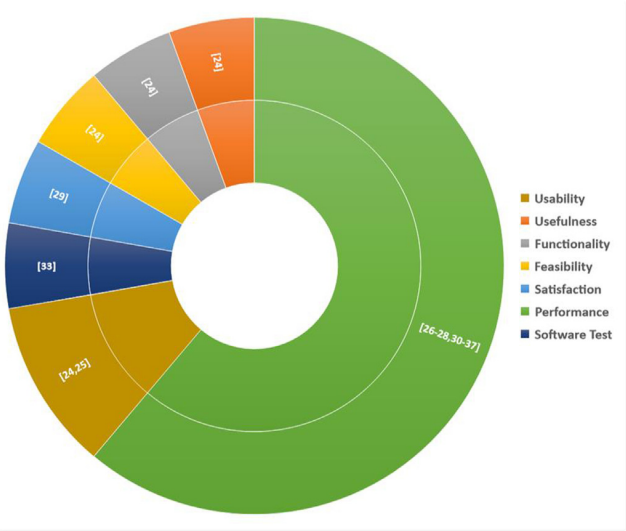

Fig. 3. Studies and their target evaluation.

The best metric (or general average, when exhibited) is presented if a set of tests with several results was found. Accuracy was the measure most utilised to validate results, accounting for 57.14%. All authors reported that they had achieved the objectives proposed by their studies, improving the decision-making process, as well as the accuracy and reliability of the proposed system.

3.6. Reported limitations

Due to the wide variety of methodologies and different goals of the selected studies, it was not possible to categorise them in terms of limitations. Thus, Table 8 presents the reported limitations and difficulties reported in each study. The studies [28–30,35,37] did not mention any limitations.

4. Discussion

4.1. Main findings

4.1.1. Motivation

We observed that the overwhelming majority of the studies [24–30,32,33,35–37] are associated with the diagnosis of a disease, followed by treatment and follow-up proposal. Proposals with the lowest representativeness were disease prevention and screening,

Table 8
Limitations reported by the studies.

Study	Reported Limitations
Laleci *et al.* [24]	The mapping of the clinical terminology code set required for semantic interoperability was limited. The reconciliation of clinical guidelines was manually made by a group of experts in the area.
Jimenez-Molina *et al.* [25]	The efficiency of the ontology arrange would be affected by the type of ontology and by the relations among resources, as well as the general inefficiency of the semantic Web engines that implement the application of queries in SPARQL. The gap between the guideline of treatment and its application, which varies from hospital to hospital, requires an analysis of the organisational process to highlight commonalities in the treatment of chronic patients.
El-Sapparh *et al.* [26]	The system does not support full interoperability with the EHR and does not consider data regarding patients from social media.
Ali *et al.* [27]	In IoT-based health prescribing systems, there are still limitations to intelligent semantic knowledge for the use of monitoring testing using body measurements.
Afzal *et al.* [31]	Authors analysed clinical documents from just one hospital. Authors could not compare results with existing systems due to the extensive amount of customisation required to implement the existing systems for a custom domain; Authors tested the prediction model for just one site, regarding "oral cavity"
Piri *et al.* [32]	The non-analytic nature of the Electronic Medical Record (EMR) data presented itself as a difficulty for the study because of its shortcomings; Authors did not have information about the time a patient's disease was first diagnosed. So authors could not incorporate the duration of time the patients lived with diabetes into their models although the duration of the disease can be considered a strong predictor of retinopathy; machine learning technique has several parameters that can be adjusted, and those values cannot be considered optimal.
Shalom *et al.* [33]	It was only tested on several dozens of patients.
Pombo *et al.* [34]	Data imputation using linear regression is sensitive to outliers; Generalisability should be addressed with caution because the sample included a relatively homogeneous group of patients recruited from one treatment centre. It is necessary to evaluate the proposed system to follow-up participants for longer periods of time.
Kong *et al.* [36]	There is a need for a dynamic clinical risk model because the patient's clinical status may change during the diagnostic process and the same risk factor may play different roles in predicting clinical risk. In BRB training must be considered both the parameters of knowledge representation and the structure of BRB.

addressed by only two studies - Jimenez-Molina et al. [25] and Kong *et al.* [36].

Román-Villarán *et al.* present in their study the result of others conducted by the National Institute of Health and Care Excellence (NICE) in the U.K. and one of their suggestions is that there is "(.) the need to advance the interaction between electronic clinical guidelines and basic medical knowledge in order to allow the guides to be adapted to a specific patient (...)" [53].

Peleg said in [54] that clinical guidelines and care flow need to be integrated to provide patient-specific advice when and where needed.

Therefore, we agree with these authors and the underline the importance of designing an adaptive CDSS, capable of treating the disease and providing care to the patient in a particular and personalised manner.

4.1.2. Software engineering

Based on the selected studies, the techniques of Software Engineering for CDSS design are still incipient. This is evidenced by the low percentage of their representativeness in the researched papers. The absence of its stages in these papers does not mean that they have not been used in the development of the application, but suggests the existence of a gap that can impair the replication of the study experience. The system's architecture is seen in the studies, highlighting the importance of its representation for a good understanding of the system design. All studies explained the functioning of the proposed architecture and/or framework, but it was observed that many did not provide additional discussion on the development stage and none addressed the maintenance stage. Another problem observed was the minor concern with the representation of the requirements of the systems, functional and non-functional.

4.1.3. Data sources

We observed that self-reporting was the less used data source (28.6%), because of the nature of the system and the patient's interaction with the CDSS. The database showed to be the main source of clinical data (85.70%), where 41.66% were from dataset repositories or clinical centres, and 33.33% were from EHR. Both data sources were used by 25.00% of the studies.

We found a relation between the data source and the use of machine learning techniques in studies using non-EHR datasets. The vast majority of the studies (5 out of 6) used the EHR were also concerned with their interoperability, as these are required to exchange information between other systems and the CDSS.

Studies that worked with machine learning techniques also used dataset as a data source and did not address interoperability, except [26], that also used the EHR as a source. Fig. 4 shows this relation.

4.1.4. Evaluation and metrics

Questionnaires, tests, and statistics were used to evaluate the application of the CDSS, and most of the studies addressed more than one type of assessment, *i.e.* the authors used a hybrid assessment. Questionnaires are considered an efficient validation tool [55,56], especially when associated with another quantification method. They are used to evaluate user acceptance of the application regarding usability, effectiveness, reliability, and usefulness. Nevertheless, only 21.4% of studies have used it.

Software tests were found in the selected papers in general to validate the framework or model of the CDSS. It is the process in which system requirements and components are evaluated and executed, either manually or using automation tools, to increase the quality of the software while making sure that the software is producing expected reliable results [57]. A significant part of studies (50.0%) tested their systems or frameworks using real or simulated data, in real or simulated scenarios, alongside the technical testing of whole system or its parts.

Mathematical metrics allow, through data analysis, to establish a level of reliability, accuracy, specificity, and sensibility expected from the developed application. The metrics can be seen in Table 6. Although all studies showed at least one metric confirming somehow the viability and good performance of the system, we observed that none addressed the assessment of the complexity (or size) of the system (or application). Another observation concerns the quality of the systems and the software, as this has not been fully addressed according to standards set by [46].

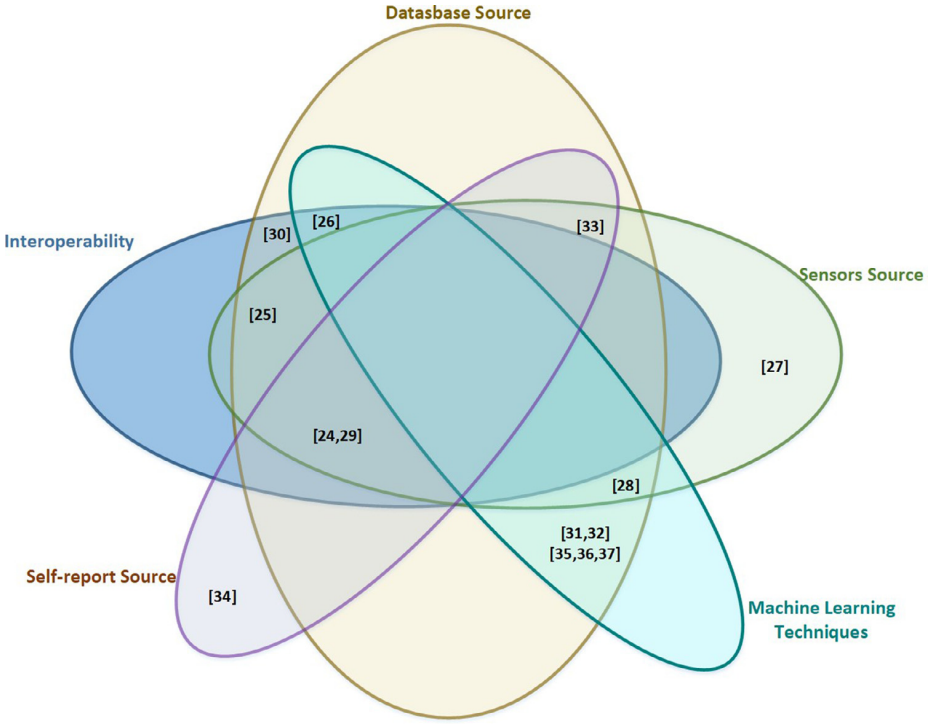

Fig. 4. Studies and the relationship with Machine Learning Techniques, Interoperability and Data source.

4.2. Study considerations

As mentioned in subsection 1.2 this study found no systematic review addressing the proposed theme. Therefore, this study allows for researchers and practitioners to further consider some implications on relevant topics.

4.2.1. Motivation/proposal

We suggest that studies, especially those that have the diagnosis as a proposal, should also include characteristics of prevention and screening. This is very important for diagnostic-focused CDSS because while the system is diagnosing a particular disease, it can also be expect that this same system may identify the onset of another disease (prevention/screening). This would cause reduced treatment costs or yet an alternative treatment proposal, and a significant increase in the patient's quality of life. Another suggestion is working on the developing adaptive CDSS, with personalised treatments.

4.2.2. Software engineering principles

We consider it important to highlight the development stage. This necessarily implies describing the stages of testing and maintenance, since they are closely interconnected. The tests were mostly done to evaluate the performance of the system (or framework) after it was finished, without even considering unit tests and planning for the maintenance of the system. We also consider useful to present Structural diagrams (*e.g.* class diagram) and Behaviour diagrams (*e.g.* Activity or Sequence diagrams), as done in [25,30,33], as it is the best way to represent the systemâs design. These structures allied to the presentation of the requirements can contribute to the replicability of a studyâs results.

4.2.3. Data sources

Semantic and syntactic interoperability are required to ensure that previously used systems promote the user acceptance of the new system, while also enforcing data integration between these two. It is emphasised here that semantic interoperability is the ability of systems to exchange information and the syntactic interoperability concerns the mapping between different data structures. It is not possible to conceive the development of a CDSS without worrying about communication between systems and the reuse of information from existing databases. We strongly recommend that studies make efforts to ensure interoperability between CDSS and other systems, regardless of their data source.

4.2.4. Evaluation

We recommend the extensive use of questionnaires as a means of evaluation, mainly to ensure user satisfaction and system usability, which also suggests a greater concern with the product quality model and compliance with ISO 25,000 [46].

4.2.5. Measure and metrics

We could not measure the time or effort used in developing a CDSS application nor how the architecture interfered with the complexity of the work. So, we strongly recommend that these features are present in future articles in order to facilitate its comprehension, replication and comparability.

4.3. Study limitations

This systematic literature review has some limitations: 1) we referred to the research only in works written in English, which implied that works with a significant presence of software engineering techniques in their composition maybe have not been considered; 2) there was no verification of cross-reference between the selected studies, which could also contribute to the loss of important information and lack of suitable comparability; yet this is partly justified by the limitation that follows; 3) the research returned only limited number of studies, and these were highly heterogeneous, with significant differences in its models and assessment techniques. Thus, it was not possible to fully compare the

selected studies, for example, in terms of performance; 4) this research focused only on CDSSs for chronic diseases, as we intend to continue the work in this area. It summarised software engineering applied in this type of systems development, and this will be useful for our future work. Yet, other CDSSs focused on non-chronic diseases would add to the heterogeneity of the systems, therefore adding an undesirable research variable and driving this research away from its focus; 5) the choice of research libraries was motivated by our focus on finding papers dealing with software engineering in CDSSs. Other databases were considered but only contributed to return a high number of duplicates.

5. Conclusion

This systematic literature review has synthesised and summarised characteristics from existing literature dealing with the design of CDSSs for monitoring chronic diseases. Fourteen studies were examined, and the main findings are condensed as follows:

- (RQ1) The CDSSs found in the selected studies are focused on the purpose of Follow-up management, Prevention and Screening, Diagnosis, Treatment and Guidelines, and Information management (see Table 3). Of them, Diagnosis was the most frequent. We suggested the inclusion of Prevention and Screening approaches to the Diagnostic CDSSs because we believe that additional disease screening and prevention could be achieved with the same data used for diagnosis. Moreover, this could imply cost reductions and also an increase of quality of life for the patients.
- (RQ2) The representativeness of Software Engineering stages was scarce and done by Structural and Behaviour diagrams coupled with others (*e.g.* architectural or framework design). As a strategy to increase study replicability, comparability and more precise assessment, we recommend future studies to include more design representation, and requirements description (see Table 4).
- (RQ3) Authors used Databases, Sensors and/or Self-reports as data sources to their CDSS (see Table 4). Databases were the most utilised by the studies, where we found a relation between the data source, machine learning techniques and interoperability between the CDSS and existing systems: 1) most studies that used EHR approached the interoperability as an important characteristic of the CDSS; 2) most studies that used a dataset from repositories applied machine learning techniques to test the model or framework. We recommend that the search for the solution of the interoperability problem needs to be a stage of the CDSSs design, since it is fundamental for its acceptability and partly aligns the software development for compliance with [47].
- (RQ4) Systems were evaluated using the criteria of user's experiences, the performance of architectural implementation and software tests. We suggest a wider use of questionnaires to assess the satisfaction of users, and the integration of user feedback into the CDSS development life cycle.
- (RQ5) The accuracy, specificity and sensitivity metrics were the most presented metrics. Although tests have been done by some studies, we could not measure the time or effort used in developing the CDSS application *i.e.* it was not possible to determine how much effort was made to achieve acceptable or optimal levels for accuracy, specificity and sensitivity parameters. Moreover, this effort should in itself be considered as a metric, or alternatively, the degree of complexity of the planned system. Therefore, we recommend that at least, complex CDSSs should be described in terms of a metric that can be then related to development effort and reliability.

With these findings, we believe that this SLR will help researchers to know how the software engineering stages are being applied to the construction of a CDSS, especially for chronic diseases. As future work, we propose a study that covers the development of a CDSS to find approaches to solve gaps encountered in this SLR, such as software requirements, the little-explored stage of development and the interoperability issue. Another proposal is to focus on Software Engineering techniques applied on the construction of a CDSS with personalised recommendations for individualised care plans.

Funding

This work was funded by FCT/MEC through national funds and when applicable co-funded by FEDER—PT2020 partnership agreement under the project UIDB/EEA/50008/ 2020. This article is based upon work from COST Action IC1303-AAPELE-Architectures, Algorithms and Protocols for Enhanced Living Environments and COST Action CA162 26-SHELD-ON-Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology). COST (European Cooperation in Science and Technology) is a funding agency for research and innovation networks. Our Actions help connect research initiatives across Europe and enable scientists to grow their ideas by sharing them with their peers. This boosts their research, career and innovation. More information in www.cost.eu. The first author was supported by incentive grant "Bolsa BID / UBI-Santander Universidades / 2018".

Declaration of Competing Interest

The authors declare that they do not have any financial or non-financial conflict of interests

Acknowledgements

The first author likes to acknowledge to *IFTM - Instituto Federal do Triângulo Mineiro*.

References

[1] World Health Organization (WHO), WHO - Reports, 2018.
[2] Center for Disease Control and Prevention, National Center for Health Statistcs (CDC-NCHS), 2019.
[3] M.A. Balakhontceva, A.A. Funkner, A.A. Semakova, O.G. Metsker, N.E. Zvartau, A.N. Yakovlev, A.E. Lutsenko, S.V. Kovalchuk, Holistic modeling of chronic diseases for recommendation elaboration and decision making, Procedia Comput. Sci. 138 (2018) 228–237, doi:10.1016/J.PROCS.2018.10.033.
[4] S. Bernell, S.W. Howard, Use your words carefully: what is a chronic disease? Front. Public Health 4 (8) (2016) 2–4, doi:10.3389/fpubh.2016.00159.
[5] N. Pombo, P. Araújo, J. Viana, Knowledge discovery in clinical decision support systems for pain management: a systematic review, Artif. Intell. Med. 60 (1) (2014) 1–11, doi:10.1016/j.artmed.2013.11.005.
[6] Centers for Disease Control and Prevention, National Center for Chronic Disease Prevention and Health Promotion (NCCD-PHP), 2019.
[7] W. Kaplan, Priority Medicines for Europe and the World 2013 Update 7.4 5. Demography, Global Burden of Disease and the Preliminary List of Priorities., Technical Report, WHO Headquarters, 2013.
[8] N. Bouchemal (Ed.), Intelligent systems for healthcare management and delivery: Advances in Healthcare Information Systems and Administration, IGI Global, 2019, doi:10.4018/978-1-5225-7071-4.
[9] P. Shankar, N. Anderson, Advances in Sharing Multi-sourced Health Data on Decision Support Science 2016–2017, 2018. 10.1055/s-0038-1641215
[10] , Guide to the software engineering body of knowledge: Trial version, A. Abran, J.W. Moore, P. Boursque, P. Depuis (Eds.), 2001.
[11] N. Pombo, N. Garcia, K. Bousson, Classification techniques on computerized systems to predict and/or to detect apnea: a systematic review, Comput. Methods Program. Biomed. 140 (2017) 265–274, doi:10.1016/j.cmpb.2017.01.001.
[12] M.M. Baig, H.G. Hosseini, M. Lindén, Machine learning-based clinical decision support system for early diagnosis from real-time physiological data, IEEE Region 10 Annual International Conference, Proceedings/TENCON (2017) 2943–2946, doi:10.1109/TENCON.2016.7848584.
[13] C. Cortes, V. Vapnik, Support-Vector networks, Mach. Learn. 20 (1995) 273–297, doi:10.1007/BF00994018.

[14] D.C. Park, M.A. El-Sharkawi, R.J. Marks, L.E. Atlas, M.J. Damborg, Electric load forecasting using an artificial neural network, IEEE Trans. Power Syst. 6 (2) (1991) 442–449, doi:10.1109/59.76685.
[15] Tin Kam Ho, Random decision forests, in: Proceedings of 3rd International Conference on Document Analysis and Recognition, 1, 1995, pp. 278–282 vol.1, doi:10.1109/ICDAR.1995.598994.
[16] A. Sanchez Guinea, G. Nain, Y. Le Traon, A systematic review on the engineering of software for ubiquitous systems, J. Syst. Softw. 118 (2016) 251–276, doi:10.1016/j.jss.2016.05.024.
[17] M. Beauchemin, M.T. Murray, L. Sung, D.L. Hershman, C. Weng, R. Schnall, Clinical decision support for therapeutic decision-making in cancer: a systematic review, Int. J. Med. Inf. 130 (7) (2019) 9, doi:10.1016/j.ijmedinf.2019.07.019.
[18] G.J. Njie, K.K. Proia, A.B. Thota, R.K. Finnie, D.P. Hopkins, S.M. Banks, D.B. Callahan, N.P. Pronk, K.J. Rask, D.T. Lackland, T.E. Kottke, Clinical decision support systems and prevention: acommunity guide cardiovascular disease systematic review, Am. J. Prev. Med. 49 (5) (2015) 784–795, doi:10.1016/j.amepre.2015.04.006.
[19] T.J. Bright, A. Wong, R. Dhurjati, E. Bristow, L. Bastian, R.R. Coeytaux, et al., Effect of clinical decision-Support systems, Ann. Intern. Med. 157 (1) (2012) 29, doi:10.7326/0003-4819-157-1-201207030-00450.
[20] M. McCaffery, C.L. Pasero, Pain ratings: the fifth vital sign, Am. J. Nurs. 97 (2) (1997) 15–16, doi:10.1097/00000446-199702000-00010.
[21] M. Merboth, S. Barnason, Managing pain: the fifth vital sign, Nurs. Clin. North Am. 35 (2) (2000) 375383.
[22] A.B.P. Abrams, R. Berger, T. Buffington, B. Collett, A. Emmanuel, M. Fall, P. Hanno, F. Howard, J. Hughes, C. Nickel, J. Nordling, D. Tripp, K. Vincent, U. Wesselmann, A.C.d.C. Williams, Classification of Chronic Pain, Second Edition - Revised, Technical Report, IASP, 2011. S3-S12
[23] D. Moher, L. Shamseer, M. Clarke, D. Ghersi, A. Liberati, M. Petticrew, P. Shekelle, L.A. Stewart, Preferred reporting items for systematic review and meta-analysis protocols (PRISMA-P) 2015 statement, Syst. Rev. 4 (1) (2015) 1, doi:10.1186/2046-4053-4-1.
[24] G.B. Laleci Erturkmen, M. Yuksel, B. Sarigul, T.N. Arvanitis, P. Lindman, R. Chen, et.al., A collaborative platform for management of chronic diseases via guideline-Driven individualized care plans, Comput. Struct. Biotechnol. J. 17 (2019) 869–885, doi:10.1016/j.csbj.2019.06.003.
[25] A. Jimenez-Molina, J. Gaete-Villegas, J. Fuentes, ProFUSO: business process and ontology-based framework to develop ubiquitous computing support systems for chronic patients' management, J. Biomed. Inform. 82 (2018) 106–127, doi:10.1016/j.jbi.2018.04.001.
[26] S. El-Sappagh, J.M. Alonso, F. Ali, A. Ali, J.H. Jang, K.S. Kwak, An ontology-based interpretable fuzzy decision support system for diabetes diagnosis, IEEE Access 6 (c) (2018) 37371–37394, doi:10.1109/ACCESS.2018.2852004.
[27] F. Ali, S.M.R. Islam, D. Kwak, P. Khan, N. Ullah, S.-j. jo Yoo, K.S. Kwak, Type-2 fuzzy ontologyaided recommendation systems for IoTbased healthcare, Comput. Commun. 119 (2018) 138–155, doi:10.1016/j.comcom.2017.10.005.
[28] M.M. Aborokbah, S. Al-Mutairi, A.K. Sangaiah, O.W. Samuel, Adaptive context aware decision computing paradigm for intensive health care delivery in smart citiesa case analysis, Sustain. Citie. Soc. 41 (2017) —-, doi:10.1016/j.scs.2017.09.004.
[29] M. Peleg, Y. Shahar, S. Quaglini, T. Broens, R. Budasu, N. Fung, et al., Assessment of a personalized and distributed patient guidance system, Int. J. Med. Inform. 101 (2017) 108–130, doi:10.1016/j.ijmedinf.2017.02.010.
[30] Y.F. Zhang, L. Gou, T. Shu Zhou, D. Nan Lin, J. Zheng, Y. Li, et al., An ontology-based approach to patient follow-up assessment for continuous and personalized chronic disease management, J. Biomed. Inform. 72 (2017) 45–59, doi:10.1016/j.jbi.2017.06.021.
[31] M. Afzal, M. Hussain, W.A. Khan, T. Ali, S. Lee, E.-N. Huh, et al., Comprehensible knowledge model creation for cancer treatment decision making, Comput. Biol. Med. 82 (2017) 119–129, doi:10.1016/j.compbiomed.2017.01.010.
[32] S. Piri, D. Delen, T. Liu, H.M. Zolbanin, A data analytics approach to building a clinical decision support system for diabetic retinopathy: developing and deploying a model ensemble, Decis. Support Syst. 101 (2017) 12–27, doi:10.1016/j.dss.2017.05.012.
[33] E. Shalom, Y. Shahar, E. Lunenfeld, An architecture for a continuous, user-driven, and data-driven application of clinical guidelines and its evaluation, J. Biomed. Inf. 59 (2016) 130–148, doi:10.1016/j.jbi.2015.11.006.
[34] N. Pombo, P. Rebelo, P. Araújo, J. Viana, Design and evaluation of a decision support system for pain management based on data imputation and statistical models, Measurement: Journal of the International Measurement Confederation 93 (2016) 480–489, doi:10.1016/j.measurement.2016.07.009.
[35] F.L. Seixas, B. Zadrozny, J. Laks, A. Conci, D.C. Muchaluat Saade, A bayesian network decision model for supporting the diagnosis of dementia, alzheimers disease and mild cognitive impairment, Comput. Biol. Med. 51 (2014) 140–158, doi:10.1016/j.compbiomed.2014.04.010.
[36] G. Kong, D.-L.L. Xu, R. Body, J.-B.B. Yang, K. Mackway-Jones, S. Carley, A belief rule-based decision support system for clinical risk assessment of cardiac chest pain, Eur. J. Oper. Res. 219 (3) (2012) 564–573, doi:10.1016/j.ejor.2011.10.044.
[37] P. Anooj, Clinical decision support system: risk level prediction of heart disease using weighted fuzzy rules, J. King Saud Univ. - Comput. Inf. Sci. 24 (1) (2012) 27–40, doi:10.1016/j.jksuci.2011.09.002.
[38] L. Marco-Ruiz, C. Pedrinaci, J.A. Maldonado, L. Panziera, R. Chen, J.G. Bellika, Publication, discovery and interoperability of clinical decision support systems: A Linked data approach, J. Biomed. Inf. 62 (2016) 243–264, doi:10.1016/j.jbi.2016.07.011.
[39] I. Sommerville, Software engineering (10th edition), 10th, Pearson, 2016.
[40] R. Maitino Neto, Engenharia de software, Editora e Distribuidora Educacional S.A, Londrina - PR - Brasil, 2016.
[41] L. Bass, P. Clements, R. Kazman, Software architecture in practice, 3, 2nd, Pearson Education (US), 2003.
[42] F. Campean, U. Yildirim, Enhanced sequence diagram for function modelling of complex systems, Procedia CIRP 60 (2017) 273–278, doi:10.1016/j.procir.2017.01.053.
[43] A. Jansen, J. Bosch, Software Architecture as a Set of Architectural Design Decisions, in: 5th Working IEEE/IFIP Conference on Software Architecture (WICSA'05), 2005, IEEE, 2005, pp. 109–120, doi:10.1109/WICSA.2005.61.
[44] O. Simeone, A very brief introduction to machine learning with applications to communication systems, IEEE Trans. Cognit. Commun. Netw. 4 (4) (2018) 648–664, doi:10.1109/TCCN.2018.2881442.
[45] H. Shah, G. Krishnan, P. Williams, A. Vogler, R.D. Allard, P.M. Nadkarni, Interoperability and integration considerations for a process-oriented clinical decision support system, in: 2011 IEEE World Congress on Services, 2011, pp. 437–442, doi:10.1109/SERVICES.2011.14.
[46] The ISO/IEC 25000 series of standards, System and Software Quality Requirements and Evaluation, 2019.
[47] The ISO/IEC 25000 series of standards - B, ISO 25000 Software and Data Quality, 2017.
[48] J. Spertus, P. Dorian, R. Bubien, S. Lewis, D. Godejohn, M.R. Reynolds, D.R. Lakkireddy, A.P. Wimmer, A. Bhandari, C. Burk, Development and validation of the atrial fibrillation effect on quality-of-life (AFEQT) questionnaire in patients with atrial fibrillation, Circulation: Arrhythmia and Electrophysiology 4 (1) (2011) 15–25, doi:10.1161/CIRCEP.110.958033.
[49] E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur, Y. Katz, Pellet: a practical OWL-DL reasoner, J. Web Semant. 5 (2) (2007) 51–53, doi:10.1016/j.websem.2007.03.004.
[50] S. Rajeshkumar, R. Omar, M. Mahmud, Taxonomies of user experience (UX) evaluation methods, Int. Conf. Res. Innovat. Inf. Syst. ICRIIS 2013 (2013) 533–538, doi:10.1109/ICRIIS.2013.6716765.
[51] N. Bevan, Classifying and selecting UX and usability measures, in: Workshop: Meaningful Measures: Valid Useful User Experience Measurement., 2008, pp. 13–18.
[52] W. Drabent, Correctness and completeness of logic programs, ACM Trans. Comput. Logic 17 (3) (2016) 1–32, doi:10.1145/2898434.
[53] E. Román-Villarán, F.P. Pérez-Leon, G.A. Escobar-Rodriguez, A. Martínez-García, C. Álvarez-Romero, C.L. Parra-Calderóna, An ontology-based personalized decision support system for use in the complex chronically ill patient, Stud. Health Technol. Inf. 264 (2019) 758–762, doi:10.3233/SHTI190325.
[54] M. Peleg, Computer-interpretable clinical guidelines: amethodological review, J. Biomed. Inf. 46 (4) (2013) 744–763, doi:10.1016/j.jbi.2013.06.009.
[55] N. Nieveen, E. Folmer, Formative evaluation in educational design research, Des. Res. (2013) 152–169.
[56] B. Laugwitz, T. Held, M. Schrepp, Construction and evaluation of a user experience questionnaire, in: Lecture Notes in Computer Science, 5298, 2008, pp. 63–76, doi:10.1007/978-3-540-89350-9_6.
[57] I. Hooda, R. Singh Chhillar, Software test process, testing types and techniques, Int. J. Comput. Appl. 111 (13) (2015) 10–14, doi:10.5120/19597-1433.

Leonice Maria Reis de Souza Pereira is a PhD student in Computer Science and Researcher in Assisted Living Computing and Telecommunications Laboratory — ALLab, a research laboratory within the Networks Group of the Instituto de Telecomunicações at the University of Beira Interior (UBI), Covilhã, Portugal. She graduated in Computer Science (Federal University of Uberlândia -UFU, Uberlândia, Brasil), Specialist in Communication Networks - Artificial Intelligence (Gama Filho Universtity — UGF — Rio de Janeiro, Brasil) and M. Sc. in Data Visualisation and Processing (UFU, Uberlândia, Brasil). She is involved in project of Software Engineering and Ambient Assisted Living. Her research interests include Health monitoring, Medical signal acquisition, analysis, and processing, Ambient Assisted Living, Internet of Things (IoT), Artificial Intelligence (AI), predictive algorithms and machine learning.

Dr. Nuno Pombo is an Assistant Professor at University of Beira Interior (UBI), Covilhã, Portugal. His current research interests include: information systems (with special focus on clinical decision support systems), data fusion, artificial intelligence, and software. He is the coordinator of the Assisted Living Computing and Telecommunication Laboratory (ALLab) at UBI. He is also member of BSAFE Lab, and Instituto de Telecomunicações - IT at UBI. Member of COST Actions: CA15109 "European Cooperation for Statistics of Network Data Science (COSTNET)"

Dr. Sofia Ouhbi is an Assistant Professor at the UAEU. She holds two doctoral degrees in Software Engineering from ENSIAS, Morocco (2015) and from University of Murcia, Spain (2017). Prior to joining UAEU, Dr. Ouhbi was an Assistant Professor at the International University of Rabat and was a Visiting Researcher at the IT University of Copenhagen in Denmark. She has an active participation in organising and chairing conferences and she has experience as a peer-reviewer for journals listed in Thomson Reuters JCR. She has co-authored more than 20 publications in peer-reviewed conferences and JCR journals. Dr. Ouhbi has intersectoral skills through work experience both in academia and industry (in Hewlett-Packard 2009–2011). She has contributed to many national and international research projects and technology transfer contracts whose topics were related to Software Engineering and Health Informatics.

Virginie Felizardo is currently a PhD student in Computer Science and Researcher in Assisted Living Computing and Telecommunications Laboratory — ALLab, a research laboratory (since 2010) within the Networks Group of the Instituto de Telecomunicações at the University of Beira Interior (UBI), Covilhã, Portugal. She graduated in Biomedical Sciences (UBI, Covilhã, Portugal) and, in 2010, received M.Sc. in Electrotechnical Engineering — Bionic Systems (UBI, Covilhã, Portugal). She is involved in project management and implementation activities in BSAFE LAB — Law enforcement, Justice and Public Safety Research and Technology Transfer Laboratory, a multidisciplinar research laboratory in UBI. Also giving assistance in the teaching of "Programming" and "Mobile Computation in Health" for Biomedical Science and Bio-engineering courses and support in research projects about biosignal acquisition, analysis, and processing. Her research interests include sensors for biomedical applications, biomedical instrumentation, Health monitoring, Medical signal acquisition, analysis, and processing, Ambient Assisted Living, predictive algorithms and machine learning.

Dr. Nuno M. Garcia holds a PhD in Computer Science Engineering from the Universidade da Beira Interior (UBI, Covilhã, Portugal) (2008) and is a 5-year BSc in Mathematics/Informatics (Hons.) also from UBI (1999–2004). He was an entrepreneur (1988–2004), member of the Research Team at Siemens SA (2004–2007) and Nokia Siemens SA (2007–2008), and Head of Research at PLUX SA (2008–2010). Currently serving as Vice Dean of the Faculty of Engineering at UBI (2018-), he is an Assistant Professor at the Computer Science Department at UBI (2010-) and Invited Associate Professor at the Universidade Lusófona de Humanidades e Tecnologias (Lisbon, Portugal, 2010-). He was the founder and is a researcher of the Assisted Living Computing and Telecommunications Laboratory (ALLab, 2010), a research group within the Instituto de Telecomunicações at UBI. He was also co-founder and is Chair of the Executive Council of the BSAFE LAB — Law enforcement, Justice and Public Safety Research and Technology Transfer Laboratory, a multidisciplinar research laboratory in UBI (2015). He is the coordinator of the Cisco Academy at UBI (2010-) and was Chair of the COST Action IC1303 AAPELE — Architectures, Algorithms and Platforms for Enhanced Living Environments (Brussels, Belgium, 2013–2017). He is member of ISOC and of the Non-Commercial Users Constituency, a group within GNSO in ICANN. His main interests include Next-Generation Networks, predictive algorithms for healthcare and well-being, distributed and cooperative algorithms, and the battle for a Free and Open Internet.

Chapter 3

Quality-in-use Characteristics for Clinical Decision Support System Assessment

The chapter presents and discusses concepts of software quality focusing on the standard ISO/IEC 25010, and its Software Product Quality and Quality in Use divisions.

It was identified quality characteristics approached in the evaluation of clinical decision support systems and presented key quality in use characteristics to be assessed during this measurement.

The chapter is presented as an article named "Evaluation Model of Quality in Use Characteristics for Clinical Decision Support Systems."

Leonice Souza−Pereira, Sofia Ouhbi and Nuno Pombo.

Published in Computer Methods and Programs in Biomedicine, 2021, in the area of Computer Science.

H−Index 95, Q1, by Scimago Journal & Country Rank.

Computer Methods and Programs in Biomedicine 207 (2021) 106169

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Quality-in-use characteristics for clinical decision support system assessment

Leonice Souza-Pereira[a,b,c,*], Sofia Ouhbi[d], Nuno Pombo[b,c]

[a] Instituto Federal do Triângulo Mineiro - Campus Uberlândia Centro, Brasil
[b] Instituto de Telecomunicações, Lisboa, Portugal
[c] Universidade da Beira Interior, Covilhã, Portugal
[d] Computer Science and Software Engineering Department, CIT, UAE University, UAE

ARTICLE INFO

Article history:
Received 24 September 2020
Accepted 4 May 2021

Keywords:
Clinical decision support systems
CDSS
Software quality
Quality-in-use
Quality assessment.

ABSTRACT

Background: Clinical decision support systems (CDSSs) are developed to support healthcare practitioners with decision-making about therapy and diagnosis' confirmation, among others. Although there are many advantages of using CDSSs, there are still many challenges in their adoption. Therefore, it is essential to ensure the quality of the system, so that it can be used confidently and securely.

Objective: This study aims to propose a set of (sub)characteristics which should be considered in evaluating the quality-in-use of CDSSs, based on the ISO/IEC 25010 standard and on existing literature.

Methods: We reviewed the existing literature on CDSS assessment and presented a list of quality characteristics evaluated.

Results: Ten quality characteristics and 56 sub-characteristics were identified and selected from the literature, in which usability was evaluated the most. An example of a scenario has been presented to illustrate our assessment approach of **satisfaction** and **efficiency** as important quality-in-use characteristics to be applied in the evaluation of a CDSS.

Conclusion: The proposed approach will contribute in bridging the gap between the quality of CDSSs and their adoption.

1. Introduction

Clinical decision support systems (CDSSs) are tools with essential clinical knowledge, designed to assist medical professionals in decision making [1,2]. The purpose of these systems is to provide the users (healthcare professionals, patients, and caregivers) with an intelligent way to monitor, manage, and improve patients' health with crucial and important information [3]. Moreover, CDSSs have the potential to empower patients with regard to awareness and more in-depth knowledge of their condition [4]. CDSSs have been found, when well-designed, to be efficient, improving clinical outcomes and health processes [5–7]. A good CDSS represents a system that produces reliable results, meets user expectations, and is useful [8,9]. One way to produce good quality CDSSs is to follow software engineering (SE) best practices [4]. If these systems are poorly developed, they can undermine health care delivery and expose patients to risk. Therefore, it is very important to investigate relevant characteristics to evaluate CDSS quality and avoid the unpleasant situation of having to deploy an unreliable system.

Due to time and budget limitations, it is difficult to evaluate all quality characteristics, and a trade-off should be made to pinpoint the critical quality features that should be evaluated in a software product. These characteristics should enable the validation of the context of the system's proposal to verify its usefulness and confirm the benefits of its use. International standards of concepts, procedures and quality metrics were developed to measure the quality of a system, i.e., measure the degree to which the system meets the needs, declared or implicit, of stakeholders, adding value to it [10]. The most recent series of standards is the ISO/IEC 25000 [11], which include models and frameworks that can assist software engineers evaluate the quality of software products.

This paper aims to identify the quality characteristics to be used for CDSS quality-in-use evaluation. To do so, we reviewed the existing literature on CDSS and we built on the ISO/IEC 25010 standard. A list of important quality characteristics has been generated

* Corresponding author at: Computer Science Department, University of Beira Interior, Marquês de Ávila e Bolama Street, Covilhã 6201-001, Portugal.
E-mail address: leonice.pereira@ubi.pt (L. Souza-Pereira).

https://doi.org/10.1016/j.cmpb.2021.106169
0169-2607/

and presented in Section 2.2. The proposed characteristics will contribute to a better and more accurate assessment of CDSSs.

This paper is organised as follows: In Section 2, we present a set of concepts and definitions necessary to guide the understanding of our study proposal. Section 3 brings studies with similar approach to our study, fitting as related works. The methodology comes in Section 4, explaining how we selected relevant literature on quality-in-use of CDSS. In Section 5, we highlight the selected (sub)characteristics identified in literature. In Section 6, we discuss the main findings and expose our considerations on how to address CDSS quality-in-use. Finally, we conclude this study, presenting its limitations and proposal for future work research in Section 7.

2. Background

2.1. Software quality models

Software Quality models aim at creating a framework for the assessment of software product quality. The series of standards ISO/IEC 25000, also known as SQuaRE (System and Software Quality Requirements and Evaluation), is the most recent version of these indicators and brings a set of standards to identify, manage and evaluate system and software quality [11]. It comprises the following divisions:

- ISO/IEC 2500n: Quality Management Division
- ISO/IEC 2501n: Quality Model Division
- ISO/IEC 2502n: Quality Measurement Division
- ISO/IEC 2503n: Quality Requirements Division
- ISO/IEC 2504n: Quality Evaluation Division
- ISO/IEC 25050–ISO/IEC 25099: SQuaRE Extension Division

Several models have been proposed to assess software quality. The most well-known models are presented below.

McCall's model - The McCall Factor Model defines the quality characteristics of a product grouped in 3 categories: (1) product operation, with correctness, reliability, efficiency, integrity, and usability; (2) product revision factors, with maintainability, flexibility and testability, and (3) product transition factors, with deal portability, reusability and interoperability [12–14].

Boehm model - This model improves the McCall Model. According to this model, a software system must be useful to be considered as a quality system. In this model, the most important quality characteristic is defined as general utility, which consists of: as utility, portability, and maintainability [12,15].

FURPS model - This model categorises requirements into functional and non-functional ones and the quality characteristics approached are: functional, usability, reliability, performance, and supportability [16].

Dromey's quality model - This model is focused on reliability and maintainability. It is based on relationships that exist between the software properties and their quality characteristics [12]. Dromey [17] has built a framework to analyse the "quality of software components through the measurement of tangible quality properties" [15].

ISO 9126-1 quality model - It is the first part of the norm ISO/IEC 9126, which primarily deals with the establishment of a system of characteristics and sub-characteristics for the definition of software quality. Software product quality was divided into two primary categories: (1) internal quality characteristics, that can be observed without executing the system, and (2) external quality characteristics, that can only be observed when executing the system [18,19].

ISO/IEC 25010 model - [10] It is one of SQuaRE's divisions (or ISO/IEC 25000 series of International Standards [11].) It is derived from the ISO 9126-1991 model, replacing it in 2011 with some amendments as, e.g., compatibility and security becoming characteristics [18]. It is divided into two wide dimensions:

1. **Software product quality model**, which consists of eight characteristics, and relates to static properties (internal quality attributes) of software and dynamic properties of the computer system (external quality attributes).
2. **Quality-in-use model**, which consists of five characteristics, and relates to the outcome of product/software interaction in a particular context.

All characteristics of part (1) and most of those of part (2) are subdivided into sub-characteristics [11,20].

2.2. Quality characteristics of the ISO/IEC 25010 models

According to the international standard ISO/IEC 25010 [10], it is recommended that specification or evaluation of the quality of the computer system or software product should use all the characteristics of both quality-in-use and product quality models. This subsection presents the models' characteristics.

Software Product Quality is divided into [10]:

- Functional Suitability - this feature shows how much a product or system meets the stated and implied needs through its functions when used under specified conditions;
- Performance Efficiency - concerns the amount of resources used under established conditions. These resources can be other software and systems, materials such as printers, storage material and others;
- Compatibility - the degree to which a component, a product or a system can exchange information with others and/or perform the necessary functions when sharing a software or a hardware environment;
- Usability - shows how much a product or system can be used by specific users in a specified context of use, to achieve specific goals with efficiency, effectiveness, and satisfaction. It can be specified or measured as a product quality characteristic (in terms of sub-characteristics), or specified or measured as a subset of quality-in-use;
- Reliability - the degree to which a system, product or component performs specific functions under specific conditions and periods. Failures in requirements, design and implementation or contextual changes contribute to limitations in reliability;
- Security - it is concerned with how much information is protected, as well as the appropriate degree of access to data and levels of authorisation;
- Maintainability - the degree of efficiency and effectiveness with which a system or a product can be modified;
- Portability - measure of efficiency and effectiveness in transferring a component, a product or a system from one environment to another, whether it relates to hardware, software, or another operating use environment.

Fig. 1 presents the product quality characteristics and sub-characteristics according to ISO/IEC 25010. **Quality-in-use** characteristics are:

- Effectiveness - it represents the level of accuracy and completeness with witch users have achieved specific objectives in using the system or product;
- Efficiency - it is concerned with the resources expended to achieve the goals; Its measure relates the level of effectiveness achieved to the expenditure of resources;
- Satisfaction - it is about how much the user's needs are satisfied within a specified context of use of the system or product. Herein are included users' desires and expectations *vis-a-vis* the system or product used;

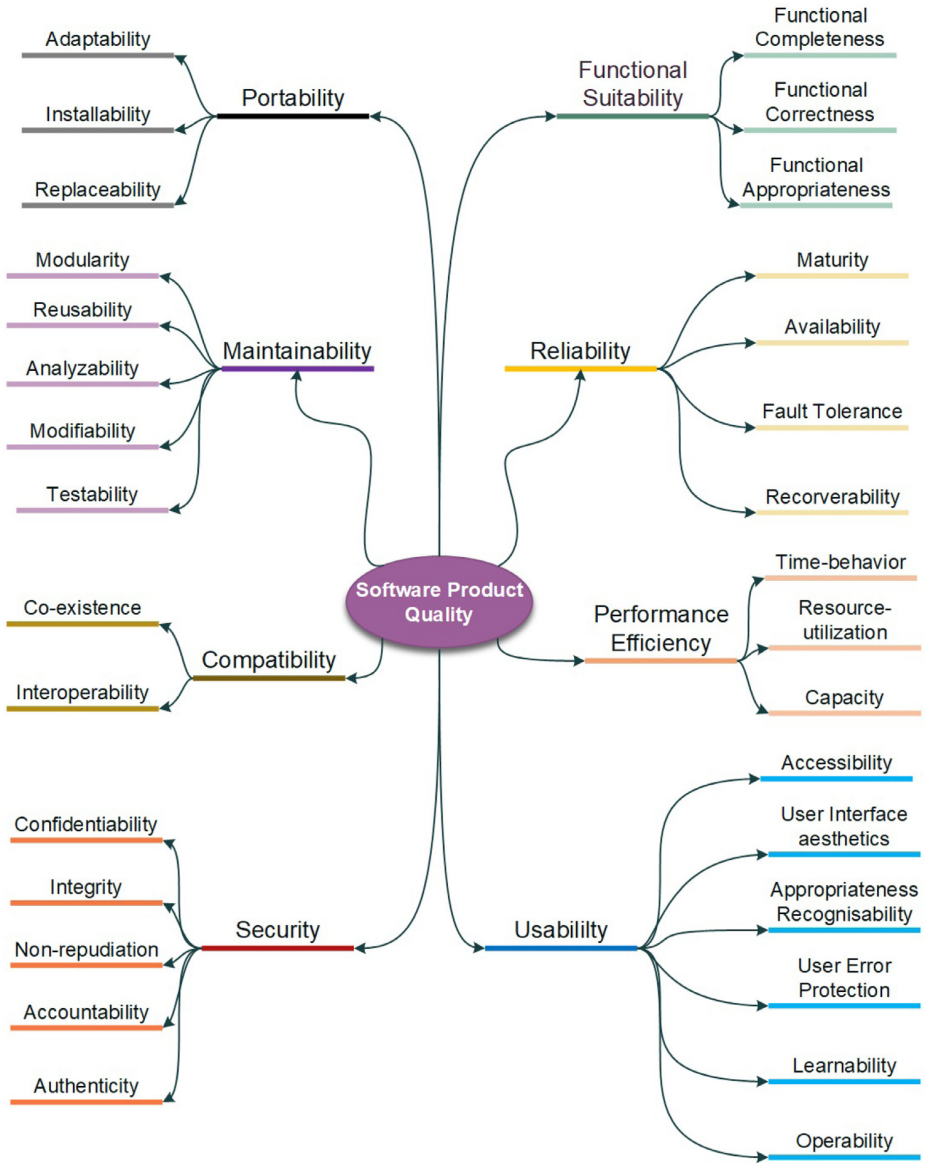

Fig. 1. Quality characteristics and sub-characteristics of the ISO/IEC 25010 Software Product Quality Model.

- Freedom from risk - it is about the degree to which the quality of a system or a product allows for mitigation or avoidance of potential risks to human life, economic status, health, or the environment;
- Context coverage - The context coverage is the sum of those previous characteristics, when a system or a product is used in specific contexts and/or in contexts beyond those initially identified.

Fig. 2 presents the ISO/IEC 25010 quality-in-use model.

2.3. Relationship between the models

According to the software quality model ISO/IEC 25010, the quality-in-use is influenced by the software product quality (external and internal) as shown in Fig. 3, which is based on ISO/IEC 25010 [11] and ISO/IEC 25022 [21]. All quality characteristics are linked through the influences that they exert on each other throughout the life cycle of a system. The quality process affects the internal properties of software product quality, which in turn affects its external properties; consequently, the quality-in-use of the software product is also affected in its various contexts of use [10]. Therefore, the quality-in-use is the effect of the quality of the software product.

The reverse-path can be understood by the link of dependence: the quality-in-use depends on the quality of the software product, which in turn depends on the quality of the process. The quality-in-use of a system is influenced by the impact of the software over stakeholders [22]. This means that the importance of quality characteristics will depend on the stakeholder's vision of the system or product, besides the objectives of the project. Assessing or measuring the quality-in-use of a system can say much about the product quality.

Some transverse product quality characteristics influence different user perspectives. Table 1 represents the perspectives of users regarding product quality characteristics that influence the quality-in-use characteristics, where Performance Efficiency is a transverse product quality characteristic that influences both the perspective of the primary user and that of the indirect user, as well as

Fig. 2. Quality characteristics and sub-characteristics of the ISO/IEC 25010 - Quality in Use.

Table 1
Quality-in-use by users perspective.

User	Product quality Characteristics
Primary	Functional suitability, Performance Efficiency, Reliability, Security, Usability
Secondary	Compatibility, Maintainability, Portability
Indirect	Performance Efficiency, Reliability, Security

Quality in use according to the classification of users and association of software product quality characteristics; primary users (people who interact with the system to achieve the primary goals.), secondary users (content providers, system managers/administrators, maintainers, analysts, etc.) and indirect users (people who receive output, but do not interact with the system.) [10].

the characteristics Reliability and Security. The transverse product quality characteristics are shown in Fig. 4.

3. Related works

Despite all the benefits of CDSS, their acceptance in the medical field is still a subject of debate [23,24]. Related works are presented below to support our goal of considering software quality-in-use to be of great importance for assessing a CDSS.

Yusof et al. [25] created a structure to evaluate Health Information Systems whereas incorporating the concept of fit between Human, Organisation, and Technology called HOT-fit. In their work, they affirmed the level of use of the system that can affect the degree of user satisfaction and vice versa, either positively or negatively. When the users are able to explore and use all the features and functions of the system, they will be more motivated and happy to use the system more and more.

Kilsdonk et al. [26] sought to determine the factors that were associated with the acceptance of CDSS by physicians by means of a systematic review that evaluated CDSS implementations. The same was done by Shibl et al. [27], where the authors developed a model to assess the acceptance of CDSS by general practitioners, based on the reasons for or against its use. The authors stated that the model indicated the utility, facilitating conditions, ease of use and confidence in the knowledge base as the main factors.

Khairat et al. [23] sought to develop a CDSS design framework to achieve user acceptance through a proposal of two models: (i) a model that aimed to optimise CDSS design based on the users' needs and expectations, and (ii) a model that relates to users' processes that manage CDSS outputs.

Alshere et al. [24] discussed in their work the use of specialised systems in the area of health, suggesting the question of the amount of effort spent in using them as a positive (or negative) influence on the professional intention to use the system. According to the authors, end-users are more concerned with using a more user-friendly, easy-to-use, and learning system rather than just focusing on its usefulness.

The aforementioned studies focused only on a limited set of characteristics related to quality-in-use, while our study focuses on identifying relevant quality characteristics reported in literature, which should be considered in the evaluation of CDSSs.

4. Methodology

To achieve the aim of this study, we studied articles on system and software quality and quality-in-use related to CDSS. To select relevant publications for our study, we followed the PRISMA[1] protocol. The research string used to identify relevant studies was as follows: (CDSS **OR** "Decision Support System") **AND** (Quality **OR** "quality-in-use" **OR** Satisfaction **OR** Effectiveness **OR** Efficiency **OR** Safety **OR** Accessibility **OR** User **OR** ISO 25010), ranging from 2010 to 2020, in the Scopus database and Google Scholar.

The candidate articles were extensively screened and relevant data were extracted and presented in Section 5 and in Tables 2 and 3. We verified which quality characteristics were addressed by the selected studies and which methods were used by the authors to evaluate them. The selected characteristics were grouped according to the ISO/IEC 25010 standard and presented in percentile form. The methods presented are those exposed in the studies to assess the characteristics addressed.

No comparisons of quality measure between studies were carried out, since the satisfactory value in its measurements depends on the requirements and purposes of the system or software; in quality-in-use, comparisons are only valid if measured within the

[1] http://www.prisma-statement.org/

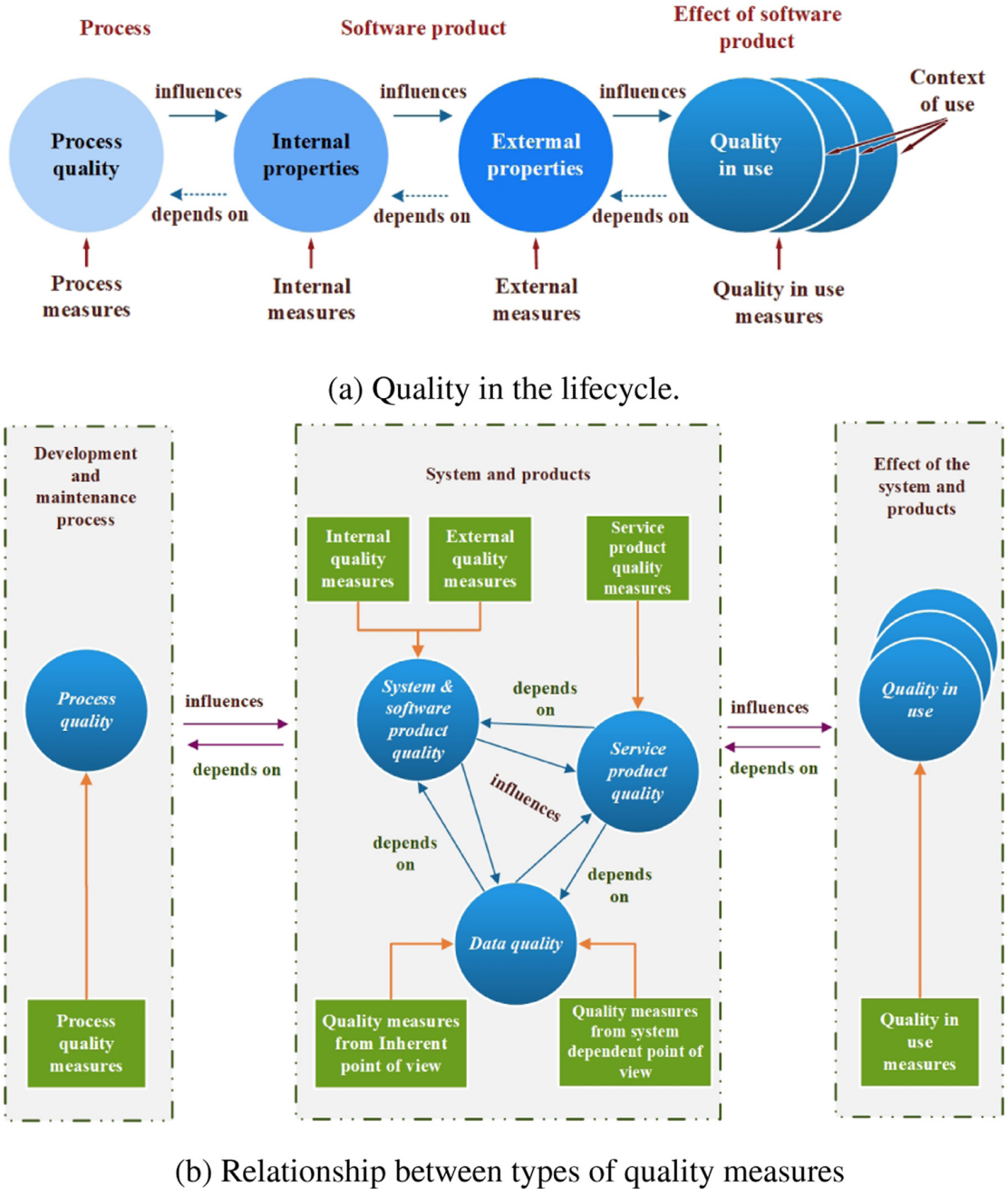

Fig. 3. Influences and dependencies between software quality models, where Figure 3a is based on Annex C-2 from [11] and Figure 3b is based on Annex E-1 from [21].

same context of use. After investigating the most frequently reported quality characteristics in the selected studies, we recommended quality-in-use characteristics we consider essential to assess in a CDSS. We emphasise that this work is not concerned with the metrics used to evaluate the characteristics and sub-characteristics obtained from the various studies, even if commented on in the results. Therefore, this information did not influence our contributions.

The Mendeley[2] and Ryyan[3] platforms were used to organise the candidate papers during the selection process. Microsoft Excel and Microsoft Visio were also used to generate the charts and figures presented in this study.

5. Results

5.1. Filtering candidate studies

The search provided 93 studies from Scopus and 271 others from Google Scholar. After excluding duplicates, citations, patents, reviews, books and book chapters and studies that did not address the evaluation of software quality characteristics, five articles were eligible for our study. Four other articles that were used in our previous work, a systematic literature review about CDSS for chronic disease [4], were included because they were considered relevant

[2] www.mendeley.com/
[3] https://rayyan.qcri.org

Table 2
Selected studies and the assessment methods.

ID study	Year	Quality model	Quality characteristic	Assessment method
Blank et al. [28]	2013	Software Product	Usability	Questionnaires, observations, groups discussion and interviews
Mahadevaiah et al. [29]	2020	Software Product	Usability, Reliability, Performance efficiency, Compatibility, Maintainability, Security, Functional Suitability	not determined
		Quality-in-use	Efficiency	
Grout et al. [30]	2018	Software Product	Usability, Reliability	Surveys
		Quality-in-use	Satisfaction	Questionnaires
Kilsdonk et al. [31]	2013	Software Product	Usability	Think-aloud analysis,
		Quality-in-use	Efficiency, Effectiveness	Statistical methods
Kadi et al. [32]	2016	Software Product	Compatibility, Performance Efficiency, Maintainability, Portability, Security, Usability, Functional Suitability and Reliability	Statistical methods
Laleci et al. [33]	2019	Software Product	Compatibility, Security and Usability	Heuristic evaluation, Spontaneous feedback, Cards product reaction and QUIS
Shalom et al. [34]	2016	Software Product	Functional Suitability	Statistical methods
Zhang et al. [35]	2017	Software Product	Functional Suitability	Statistical methods
Peleg et al. [36]	2017	Quality-in-use	Satisfaction	Questionnaires

The selected studies, their publication years and the methods used to assess the quality characteristics.

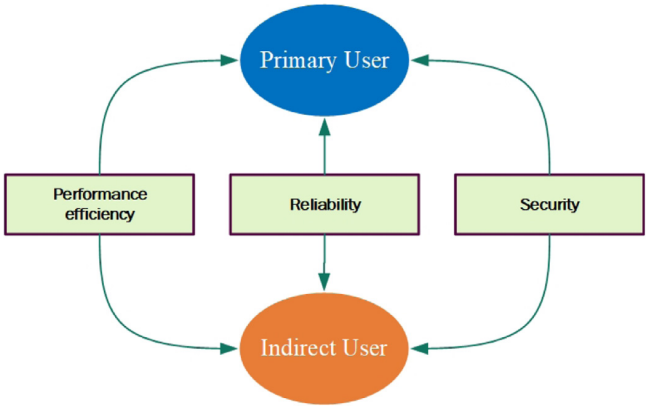

Fig. 4. Transverse quality characteristics by users' perspectives.

for this work. Table 2 summarises methods used in the selected studies for CDSS quality, including the model, the characteristic and the assessment method. An overview of each selected study and the characteristics evaluated is presented in the subsections below.

5.2. The selected studies

The main goal of the study carried out by Blank et al. [28] was to describe the quality of prenatal and maternal care - QUALMAT - by assessing the usability and the acceptance of the proposed system. The project QUALMAT, implemented in three African countries (Burkina Faso, Ghana, and Tanzania), aimed to improve the performance and motivation of healthcare personnel in rural areas, and consequently to improve the quality of primary maternal health care services. From the CDSS implementation point of view, authors combined quantitative and qualitative assessments to determine the system usability through questionnaires, ethnographic studies, group meetings, and individual interviews.

Mahadevaiah et al. [29] brought a guidance to the adoption of a commercial CDSS with selection, acceptance testing, commissioning, implementation and quality assurance stages. All the phases presented features of quality that should be measured and evaluated in a CDSS. The authors stated that one quality indicator of a CDSS is its performance. However, they also affirmed that it is not always possible to measure this indicator, especially if there is no gold standard for performance. They recommended that acceptability should be considered and weighed against performance when choosing a CDSS, considering its acceptance by end-users. The quality characteristics assessed in the study were usability, reliability, maintainability, security, functional suitability, efficiency, and effectiveness. The mechanisms to achieve this were not determined, but the authors focused on statistical analysis as a way of measuring certain features.

Grout et al. [30] examined the long-term acceptability of a paediatric CDSS called CHICA - Child Health Improvement through Computer Automation, an evidence-based system that has been in use since 2004 at several urban community clinics in Indianapolis (USA). The CDSS users responded annually to surveys to assess acceptance, usability, and perceived effectiveness of the system aiming for a continuous quality improvement. To test the correlation between the research items, the authors calculated the Phi coefficient and performed logistic regression. The same was used to verify user satisfaction regarding familiarity with the system and CDSS maturity. Comparisons were made between the responses of new and old users to the most recent version of the system (in 2016) and the responses obtained in previous years (until 2011), where a trend to increase satisfaction over time was observed.

Kilsdonk et al.[31] followed a UCD (user-centred design) approach in the development of a CDSS prototype, implementing an expert-guided paper-based guideline to monitor childhood cancer survivors. To verify whether their user-centred CDSS prototype could offer better support to healthcare professionals, the authors compared the usability problems in retrieving information from their CDSS with those of the guideline. For this, the think-aloud analysis method was used, based on verbal protocols and presentation of video recordings made while the healthcare practitioners used both the CDSS and the paper-based guideline into two different scenarios. The authors registered usability problems as they appeared or increased in frequency (if they already existed in the database). The data were classified according to degree of severity. Statistical methods were used to determine the usability of both CDSS and guideline, and thus establish the best CDSS. The study also looked into the measurement of quality-in-use through effectiveness and efficiency, which were measured in terms of completeness and time-task, respectively.

Kadi et al.[32] presented a checklist of requirements for the development of a decision support system (DSS) for heart disease, and verified whether or not these requirements influenced the software product quality. The requirements were divided into groups and tested using metrics of the ISO/IEC 9126-2 model to define this influence. The ISO/IEC 25010 models weres considered as a source of characteristics and sub-characteristics of quality. The authors highlighted the degree of influence on the compatibility, performance efficiency, maintainability, portability, usability, functional suitability, reliability, and security characteristics.

In order to suggest personalised recommendations to individualised care plans, Laleci et al. [33] developed a semi-automatic care plan management tool, called C3-Cloud, integrated with clinical decision support services. Jakob Nielsen's walk-through method was used to measure the system's usability, using an iterative and holistic approach, through heuristic evaluation, spontaneous feedback, reaction to the product on cards and questionnaire for user interaction satisfaction (QUIS). In addition, the authors were concerned with seeking techniques and semantic interoperability of the CDSS, besides the security of access through authentication, authorisation and audit.

Shalom et al. [34] presented in their study requirements to implement an architecture, called Picard DSS, for realistic automated application of continuous guidelines based decision support. Within the pre-eclampsia and toxaemia domains, the authors conducted a technical and functional evaluation of their architecture, besides of a clinical-oriented assessment. Statistical methods were used to determine the functional suitability characteristic of the product, using scenarios with different complexities of the diseases to assess the correctness and completeness of their product.

Using an ontology-based framework to propose a continuous and personalised chronic disease management, Zhang et al. [35] developed a CDSS aimed at supporting the follow-up assessments of ill patients at home. The authors evaluated their system technically and functionally, through a case study of type 2 diabetic patient follow-up assessment. Functional evaluation was focused on effectiveness characteristic, in terms of completeness and accuracy. The completeness of the system was determined by the percentage of system-generated assessment criteria and the criteria evaluated by the nursing staff. The accuracy was obtained by means of nurses' annotations during follow-up visits, that determined if the CDSS outcome was following the gold standard classification of diabetic disease, and mathematical metrics to count it.

Inserted in the Mobiguide project context, Peleg et al. [36] aimed to assess the feasibility and potential of their mobile decision support system (MDSS) developed for patients and healthcare providers in the domains of Atrial Fibrillation and Gestational Diabetes Mellitus. Although the authors did not declare to use any system quality model, due to the characteristics evaluated and presented in the article, this study probably used the ISO/IEC 25010 and ISO/IEC 9126 models. The authors measured the quality-in-use in assessing the level of satisfaction, using questionnaires administered on patients.

Table 3
Characteristics and sub-characteristics.

Characteristics	Sub-characteristics	Quantity	Studies
Compatibility	Co-existence	1	[32]
	Interoperability	2	[29,32]
	Unspecified	1	[33]
Functional	Functional Completeness	3	[29,34,35]
Suitability	Functional Correctness	1	[34]
	Functional Appropriateness	1	[32]
Maintainability	Modularity	1	[32]
	Reusability	1	[32]
	Analyzability	1	[32]
	Modifiability	2	[29,32]
	Testability	1	[32]
Performance Efficiency	Time behaviour	2	[29,32]
	Resource utilisation	1	[32]
Portability	Replaceability	1	[32]
	Adaptability	1	[32]
	Installability	1	[32]
Reliability	Availability	2	[29,32]
	Maturity	2	[29,30]
	Fault tolerance	2	[29,32]
	Recoverability	1	[32]
Security	Confidentiality	1	[32]
	Integrity	1	[32]
	Non-repudiation	1	[32]
	Accountability	1	[32]
	Authenticity	2	[32,33]
	Unspecified	1	[29]
Usability	Accessibility	1	[29]
	Appropriateness Recognisability	4	[28,29,31,32]
	Learnability	3	[28,29,32]
	Operability	3	[28,29,32]
	Helpfulness	1	[32]
	Attractiveness	1	[32]
	Unspecified	2	[30,33]
Effectiveness	–	1	[31]
Efficiency	–	2	[29,31]
Satisfaction	Usefulness	1	[30]
	Trust	1	[30]
	Unspecified	1	[36]

Characteristics and sub-characteristics extracted from the selected studies with the quantity of each sub-characteristic found. "Unspecified" means the sub-characteristic was not specified and could not be identified from the results.

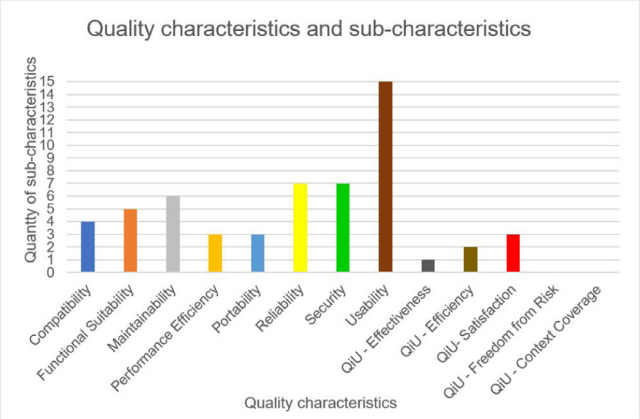

Fig. 5. Quality characteristics and quantity of sub-characteristics assessed in the selected studies. These sub-characteristics are described in Table 3.

5.3. Approached quality characteristics

Table 3 presents the studies and correlated them with characteristics and sub-characteristics assessed. Of the twelve quality characteristics referenced by the ISO/IEC 25010 standard, only two were not observed in the studies: freedom from risk and context coverage. Fifty-six sub-characteristics were identified from the selected studies as shown in Table 3. Some sub-characteristics were neither expressed nor identified in their respective studies. When possible, they were deducted by the metric used or by the result found; otherwise, we classified them as Unspecified in Table 3. It should be noted that the quality-in-use characteristics Effectiveness and Efficiency do not have sub-characteristics, according to the ISO/IEC 25010 models.

We observed that the software product quality characteristic Usability was the most measured, accounting for 26.8% of the features assessed, followed by Security, Reliability and Maintainability accounting for 12.5%, 12.5%, and 10.7% respectively. All other characteristics had less than 9.0% of representativeness. Quality-in-Use had a representativeness of 10.7%, adding up all their characteristics addressed.

Fig. 5 shows the quality characteristics found in the selected studies with the quantity of sub-characteristics identified. In addition, the quality characteristics distributed by the studies is presented in Fig. 6. Various methods were used to assess those quality characteristics and sub-characteristics. They can be seen in Table 2. However, the methods most used by the authors were questionnaires and statistical methods. By statistical methods, we mean mathematical functions associated with methods and sub-characteristics, whether defined by the ISO/IEC 25022 standard or proposed by the evaluators.

6. Discussion

A CDSS is not an easy system to develop. It is generally morbidity-specific and targeted at expert users. Besides this, it requires agreement with medical guidelines, defined specific taxonomies and precision in the results, since it impacts the management of a person's clinical health. Therefore, it is evident that all quality characteristics are of great importance to ensure quality and usefulness of these systems. However, it is almost impossible to specify, assess or measure all sub-characteristics of a computer system or software product [10,22]. Likewise, it is very difficult to measure quality-in-use attributes for all scenarios of user tasks. Considering the characteristics noticed in selected studies and the acquired knowledge through literature reviews, we investigated the quality-in-use attributes that we could identify as relevant to assess the CDSS usage and application. We highlight quality charac-

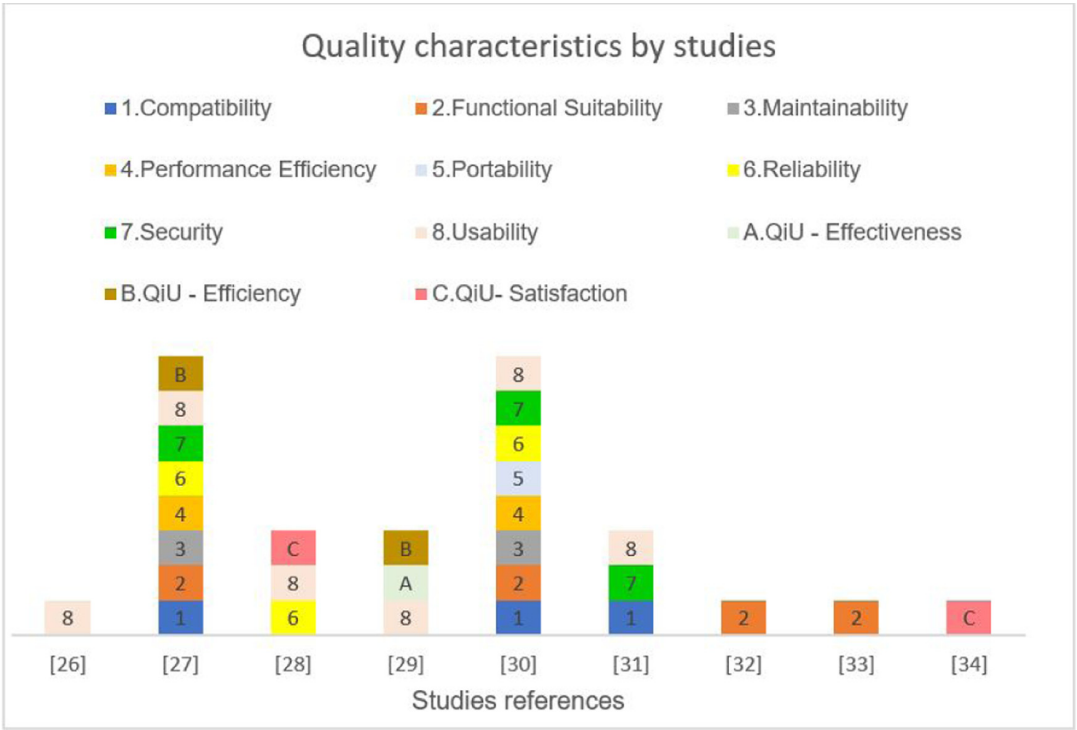

Fig. 6. Selected studies and approached quality characteristics.

teristics that should help researchers and professionals to develop a CDSS with more chance to be utilised, associating them to gaps found during the literature search.

6.1. Important quality characteristics for CDSS evaluation

As aforementioned in Section 2.3, there is an intrinsic relation between the models of product quality and quality-in-use that depends on the view and type of users. So, we addressed the primary user perspective to choose the quality-in-use characteristics to be assessed.

Effectiveness is the relationship between the expected results and the results obtained. The CDSS is expected to meet this requirement, given the quality project applied in its construction and the mandatory reliability of its results.

Efficiency is about the results achieved and the resources used to achieve them. Therefore, it is important to use the least amount of resources (e.g., the time spent to perform a task is a resource), reducing costs (hourly work) but continuing to bring the desired results.

We chose the characteristics **Satisfaction** and **Efficiency** to be evaluated in CDSSs, because the user is concerned with the his/her experience and his/her productivity [37]. In addition, some efficiency metrics use effectiveness measures, as shown by ISO/IEC 25022 in annex D 3.3.3 [21]

For **Satisfaction**, the following sub-characteristics should be considered:

- Usefulness - this sub-characteristic will allow verifying the user's satisfaction regarding his/her perception of the achievement of practical objectives, including the results and consequences of the CDSS usage;
- Trust - this will allow for assessing the degree of user confidence regarding the behaviour of the CDSS, if it is as intended and expected;
- Pleasure - it is related to user experiences and will enable to assess the degree to which his/her needs for pleasure are satisfied.

For **Efficiency**, the following attributes should be considered:

- Task time - this will enable to determine how long the user would take to complete their task successfully;
- Time efficiency - the efficiency with which users achieve their goals over time when using the CDSS;
- Cost-effectiveness - it is the cost to perform a task with effectiveness. For example, costs could include the cost of material resources and the cost of users' time;
- Productive time ratio - it concerns the proportion of time the user spends doing productive actions while using the CDSS.

When assessing these sub-characteristics and attributes, the intention is to measure the quality of the CDSS and associate this measure with the acceptance of the system by the user.

6.2. Illustrative example

To illustrate the importance of these two quality-in-use characteristics within a given context, we propose the following scenario for a CDSS developed for the diagnosis and treatment of chronic pain associated with repetitive activities (for example, work-related musculoskeletal disorders - WMSDS). The scenario is presented in Fig. 7.

1. The system requires the entry of clinical data, either (i) manually or (ii) through an interoperable interface between the CDSS and an Electronic Health Record system - EHR;
2. Two possibilities for an interoperable interface option: (a) one that demands browsing several screens to establish the connectivity between the CDSS and EHR or (b) one that demands fewer screens to complete the connectivity with the EHR;
3. A primary user as an end-user;
4. A task to enter a patient's personal and clinical data into CDSS to obtain as outcome a treatment suggestion and/or request for clinical examinations;
5. Satisfaction (pleasure and trust) and efficiency (cost-effectiveness) as quality-in-use characteristics and sub-characteristics addressed.
6. Usability is the product quality characteristic that will impact the quality-in-use model.

Here, we illustrated two result possibilities of the use of the system in the presented context. The first is a negative result, with the option scenario (1-ii; 2-a), where the user is unable to use the system in all its completeness of resources and functions to perform its task. This result can be explained if the system's learning curve is very high; if to ensure the use of interoperability functionality between systems, the user must memorise several difficult and not very intuitive procedures, while navigating the screens.

The user might have difficulties and a bad experience, especially in the use of all available resources and functions. Therefore, they could opt for the manual input (1-i), which is more time consuming and stressful. This has a major impact on the task's execution time. When the system's efficiency is compromised at the user level, they might stop using the system. Their satisfaction decreases, mainly for not using the integration of CDSS with EHR, functionality that would be there to improve the process and the result.

At the second possibility and positive result, now with the scenario (1-ii,2-b), and where the task is successfully and easily accomplished, the analysis happens in the same way. Since the system is efficient enough to allow the task to be completed in a short (or reasonable) time, the user feels motivated to know and master the system. Extracting all the CDSS features that allow for improving the process is a reason for the user to continue its use. The user feels pleasure and satisfaction, and this makes the CDSS's usage continuous and faster, fulfilling the maxim "*practice makes perfect*."

Here, there is a feedback loop, with either positive or negative results. This could even give us the scope for wrong reasoning that Satisfaction and Efficiency are directly proportional characteristics. However, it not so simple. There are software product characteristics that directly affect the quality-in-use characteristics.

For example, if the user executes a task with efficiency but the system does not offer a reliable result, the satisfaction would produce an opposite behave; the user's satisfaction would not increase. Nigel Bevan [38] affirmed that "measures of satisfaction can provide a useful indication of the user's perception of usability, even if it is not possible to obtain measures of effectiveness and efficiency."

Therefore, assessing the quality-in-use characteristics Satisfaction and Efficiency allow for contributing to the system's classification in terms of its overall quality, since the quality-in-use characteristics are directly associated with the quality of the software product characteristics.

Once inconsistent or unacceptable values in quality-in-use characteristics are detected, adjustments and corrections in the system can be carried out to mitigate such problems. In the case of CDSS, this could come to contribute to the gap's resolution regarding the acceptance and use of the system mainly by health professionals.

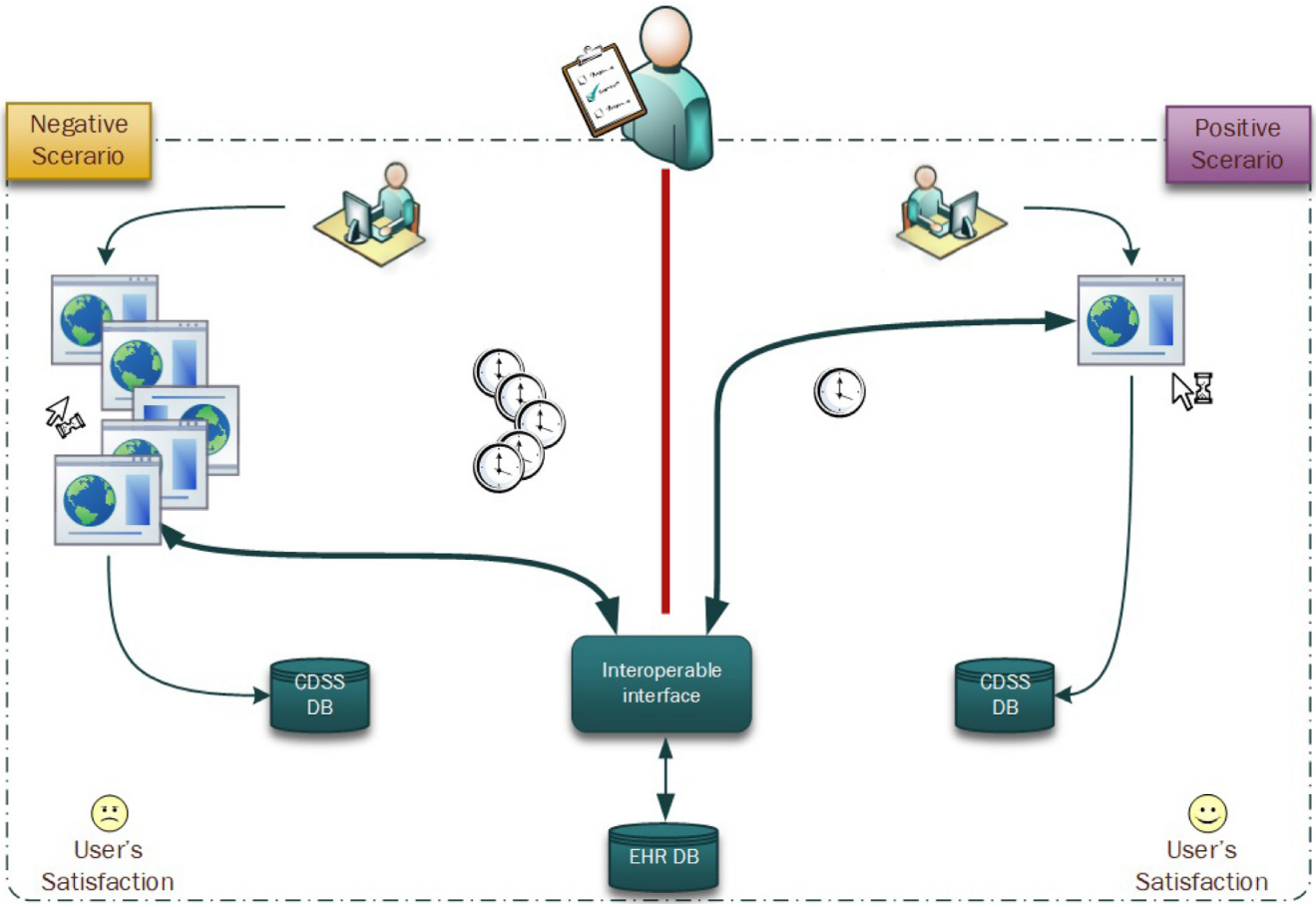

Fig. 7. Negative and positive scenarios of a task execution.

7. Conclusion and future works

Developing or purchasing software is an expensive investment and it needs to be justified. Besides being useful in its purpose, the system must also be reliable, efficient and, among other features, must meet the expectations of users. A way to ensure these characteristics, even in minimum standard values, is to be concerned with its quality, both in the design and development process and in its actual use.

Measuring the software quality is measuring the quality characteristics. So, the sub-characteristics found in the selected papers are a summary of the characteristics measured by the authors to identify the quality of CDSS. In the case of measuring quality-in-use, which is the focus of our work, the context of use must also be considered besides the choice of the characteristic and sub-characteristics that best suit the purpose of the measurement.

When evaluating the quality of a software system, it is necessary to consider the importance of the quality characteristics and sub-characteristics to evaluate. Software quality characteristics can affect each others; therefore, it is important to determine which ones should be used in the quality evaluation. The study of Hovorushchenko [39] presented this cross-correlation between characteristics and sub-characteristics, stating that during the evaluation of software quality,

According to Harrison et al. [40], Effectiveness, Efficiency and Satisfaction are considered the main criteria to be assessed to reflect the quality of use. Therefore, these characteristics of qualities-in-use meet the needs and expectations of users of the systems, in our case of CDSSs, as they reflect the user experience. The quality-in-use model provide a powerful contribution to the practice of evaluating a system and determining its quality. Therefore, as a contribution, we proposed two quality-in-use characteristics, Satisfaction and Efficiency, we believe to be important in the evaluation of a CDSS. These characteristics, when evaluated, can corroborate the quality of the CDSS and mitigate the non-use and non-acceptance of this type of software, due to their links with the user experience and the usability of the system.

Quality in use is portrayed as the usability of the system [37] and measuring the usability of the systems is the way to obtain a more complete understanding of the users' needs and improve the product, providing a better user experience [41]. However, we observed that, among the candidate studies, the authors' preference is to use of the software product quality characteristics, e.g., usability, to assess the systems; quality-in-use characteristics are rarely used to determine the quality of a CDSS.

As a limitation, our study might have missed relevant publications, even though we conducted the search in the Scopus and Google Scholar databases, which are among the largest digital libraries available for publications.

As future work, we intend to create a guideline to measure these characteristics of quality-in-use in CDSS, according to ISO/IEC 25022 standard "SQuaRE - Measurement of quality-in-use," and present its application in a CDSS prototype.

Another direction for future work would be the analysis of the cross-reference of quality-in-use characteristics and sub-

characteristics. This analysis will allow us to verify the dependencies of these characteristics and their joint measures.

Funding

This work is fund by FCT/MCTES through national funds and when applicable co-funded by FEDER-PT2020 partnership agreement under the project UIDB/EEA/50008/2020. (Este trabalho é financiado pela FCT/MCTES através de fundos nacionais e quando aplicável co-financiado por fundos comunitários no âmbito do projeto UIDB/EEA/50008/2020). This article is based upon work from COST Action IC1303-AAPELE-Architectures, Algorithms and Protocols for Enhanced Living Environments and COST Action CA162 26-SHELD-ON-Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology). More information in http://www.cost.eu. This study also is partially funded by UAEU.

Declaration of Competing Interest

Authors declare that they have no conflict of interest.

Acknowledgements

Leonice Souza-Pereira would like to thank the *IFTM - Instituto Federal do Triângulo Mineiro* for their support.

References

[1] R.T. Sutton, D. Pincock, D.C. Baumgart, D.C. Sadowski, R.N. Fedorak, K.I. Kroeker, An overview of clinical decision support systems: benefits, risks, and strategies for success, npj Digit. Med. 3 (1) (2020) 1–10, doi:10.1038/s41746-020-0221-y.
[2] E.S. Berner, Clinical Decision Support Systems, Health Informatics, secdond ed., Springer New York, New York, NY, 2007, doi:10.1007/978-0-387-38319-4.
[3] P. Shankar, N. Anderson, Advances in sharing multi-sourced health data on decision support science 2016–2017, IMIA Schattauer GmbH Adv. 27 (1) (2018) 16–24, doi:10.1055/s-0038-1641215.
[4] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, N. Garcia, Clinical decision support systems for chronic diseases: a systematic literature review, Comput. Methods Prog. Biomed. 195 (2020), doi:10.1016/j.cmpb.2020.105565.
[5] T.J. Bright, A. Wong, R. Dhurjati, E. Bristow, L. Bastian, R.R. Coeytaux, G. Samsa, V. Hasselblad, J.W. Williams, M.D. Musty, L. Wing, A.S. Kendrick, G.D. Sanders, D. Lobach, Effect of clinical decision-support systems, Ann. Intern. Med. 157 (1) (2012) 29, doi:10.7326/0003-4819-157-1-201207030-00450.
[6] P. Fraccaro, M. Arguello Castelerio, J. Ainsworth, I. Buchan, Adoption of clinical decision support in multimorbidity: a systematic review, JMIR Med. Inform. 3 (1) (2015) e4, doi:10.2196/medinform.3503.
[7] S. Van de Velde, A. Heselmans, N. Delvaux, L. Brandt, L. Marco-Ruiz, D. Spitaels, H. Cloetens, T. Kortteisto, P. Roshanov, I. Kunnamo, B. Aertgeerts, P.O. Vandvik, S. Flottorp, A systematic review of trials evaluating success factors of interventions with computerised clinical decision support, Implement. Sci. 13 (1) (2018) 1–11, doi:10.1186/s13012-018-0790-1.
[8] M. Khalifa, Clinical decision support: strategies for success, in: Procedia Comput. Sci., volume 37, 2014, pp. 422–427, doi:10.1016/j.procs.2014.08.063.
[9] D. Zikos, A framework to design successful clinical decision support systems, in: Proc. 10th Int. Conf. PErvasive Technol. Relat. to Assist. Environ., in: PETRA '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 185–188, doi:10.1145/3056540.3064960.
[10] of CanadaS.C., CAN/CSA-ISO/IEC 25010:12, Systems and software engineering—Systems and software quality requirements and evaluation (SQuaRE)—System and software quality models, Can. Stand. Assoc. 12 (2012) 46. https://www.scc.ca/en/standardsdb/standards/26856
[11] ISO 25000 STANDARDS, 2019, https://iso25000.com/index.php/en/iso-25000-standards.
[12] McCall Software Quality Model |Professionalqa.com, 2016, https://www.professionalqa.com/mc-call-software-quality-model.
[13] Software Quality Factors - Tutorialspoint, 2020, https://www.tutorialspoint.com/software_quality_management/software_quality_management_factors.htm.
[14] K. Musa, J. Alkhateeb, Quality model based on cots quality attributes, Int. J. Softw. Eng. Appl. 4 (1) (2013) 1–8, doi:10.5121/ijsea.2013.4101.
[15] M.-A. Côté, M. Ing, E. Georgiadou, Software quality model requirements for software quality engineering, 2014, https://www.researchgate.net/publication/228947488_Software_Quality_Model_Requirements_for_Software_Quality_Engineering.
[16] F.S. Al-Obthani, A.A. Ameen, Towards customized smart government quality model, Int. J. Softw. Eng. Appl. 9 (2) (2018) 41–50, doi:10.5121/ijsea.2018.9204.
[17] R.G. Dromey, A model for software product quality, IEEE Trans. Softw. Eng. 21 (2) (1995) 146–162, doi:10.1109/32.345830.
[18] A. Abran, ISO 9126: Analysis of Quality Models and Measures, John Wiley & Sons, Inc, 2010, doi:10.1002/9780470606834.ch10.
[19] A. Horbst, K. Fink, G. Goebel, The ISO/IEC 9126-1 as a supporting means for the system development process of a patient information web service, in: Stud. Health Technol. Inform., volume 116, 2005, pp. 967–970. https://www.researchgate.net/publication/7603331_The_ISOIEC_9126-1_as_a_Supporting_Means_for_the_System_Development_Process_of_a_Patient_Information_Web_Service
[20] ISO/IEC 25010 Software Quality Model, 2017, https://blog.codacy.com/iso-25010-software-quality-model/.
[21] ISO/IEC 25022:2016 (en) - Systems and software engineering—Systems and software quality requirements and evaluation (SQuaRE)—Measurement of quality in use, 2020, https://iso25000.com/index.php/en/iso-25000-standards.
[22] N. Djordjevic, Software quality standards, Vojnoteh. Glas. 65 (1) (2017) 102–124, doi:10.5937/vojtehg65-10668.
[23] S. Khairat, D. Marc, W. Crosby, A. Al Sanousi, Reasons for physicians not adopting clinical decision support systems: critical analysis, J. Med. Internet Res. 20 (4) (2018), doi:10.2196/medinform.8912.
[24] K.A. Alshare, M.K. Alomari, P.L. Lane, R.D. Freeze, Development and determinants of end-user intention: usage of expert systems, J. Syst. Inf. Technol. 21 (2) (2019) 166–185, doi:10.1108/JSIT-08-2018-0108.
[25] M.M. Yusof, J. Kuljis, A. Papazafeiropoulou, L.K. Stergioulas, An evaluation framework for health information systems: human, organization and technology-fit factors (HOT-fit), Int. J. Med. Inform. 77 (6) (2008) 386–398, doi:10.1016/j.ijmedinf.2007.08.011.
[26] E. Kilsdonk, L.W.P. Peute, S.L. Knijnenburg, M.W.M. Jaspers, Factors known to influence acceptance of clinical decision support systems, Stud. Health Technol. Inform. 169 (2011) 150–154, doi:10.3233/978-1-60750-806-9-150.
[27] R. Shibl, M. Lawley, J. Debuse, Factors influencing decision support system acceptance, Decis. Support Syst. 54 (2) (2013) 953–961, doi:10.1016/j.dss.2012.09.018.
[28] A. Blank, H. Prytherch, J. Kaltschmidt, A. Krings, F. Sukums, N. Mensah, A. Zakane, S. Loukanova, L.L. Gustafsson, R. Sauerborn, W.E. Haefeli, "Quality of prenatal and maternal care: bridging the know-do gap" (QUALMAT study): an electronic clinical decision support system for rural Sub-Saharan Africa, BMC Med. Inform. Decis. Mak. 13 (1) (2013), doi:10.1186/1472-6947-13-44.
[29] G. Mahadevaiah, R.V. Prasad, I. Bermejo, D. Jaffray, A. Dekker, L. Wee, Artificial intelligence-based clinical decision support in modern medical physics: selection, acceptance, commissioning, and quality assurance, Med. Phys. 47 (5) (2020) e228–e235, doi:10.1002/mp.13562.
[30] R.W. Grout, E.R. Cheng, A.E. Carroll, N.S. Bauer, S.M. Downs, A six-year repeated evaluation of computerized clinical decision support system user acceptability, Int. J. Med. Inform. 112 (January) (2018) 74–81, doi:10.1016/j.ijmedinf.2018.01.011.
[31] E. Kilsdonk, L.W. Peute, R.J. Riezebos, L.C. Kremer, M.W.M. Jaspers, From an expert-driven paper guideline to a user-centred decision support system: a usability comparison study, Artif. Intell. Med. 59 (1) (2013) 5–13, doi:10.1016/j.artmed.2013.04.004.
[32] I. Kadi, A. Idri, S. Ouhbi, Quality evaluation of cardiac decision support systems using ISO 25010 standard, in: Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, 0, 2016, pp. 1–8, doi:10.1109/AICCSA.2016.7945657.
[33] G.B. Laleci Erturkmen, M. Yuksel, et al., A collaborative platform for management of chronic diseases via guideline-driven individualized care plans, Comput. Struct. Biotechnol. J. 17 (2019) 869–885, doi:10.1016/j.csbj.2019.06.003.
[34] E. Shalom, Y. Shahar, E. Lunenfeld, An architecture for a continuous, user-driven, and data-driven application of clinical guidelines and its evaluation, J. Biomed. Inform. 59 (2016) 130–148, doi:10.1016/j.jbi.2015.11.006.
[35] Y. fan Zhang, L. Gou, T. shu Zhou, D. nan Lin, J. Zheng, Y. Li, J. song Li, An ontology-based approach to patient follow-up assessment for continuous and personalized chronic disease management, J. Biomed. Inform. 72 (2017) 45–59, doi:10.1016/j.jbi.2017.06.021.
[36] M. Peleg, Y. Shahar, S. Quaglini, T. Broens, R. Budasu, N. Fung, A. Fux, G. García-Saéz, A. Goldstein, A. González-Ferrer, H. Hermens, M.E. Hernando, V. Jones, G. Klebanov, D. Klimov, D. Knoppel, N. Larburu, C. Marcos, I. Martínez-Sarriegui, C. Napolitano, A. Pallas, A. Palomares, E. Parimbelli, B. Pons, M. Rigla, L. Sacchi, E. Shalom, P. Soffer, B. van Schooten, Assessment of a personalized and distributed patient guidance system, Int. J. Med. Inform. 101 (2017) 108–130, doi:10.1016/j.ijmedinf.2017.02.010.
[37] N. Bevan, Classifying and selecting UX and usability measures, in: Work. Meaningful Meas. Valid Useful User Exp. Meas., 2008, pp. 13–18.
[38] N. Bevan, Usability as quality of use, Softw. Qual. J. 130 (1995) 115–130.
[39] T. Hovorushchenko, O. Pomorova, Evaluation of mutual influences of software quality characteristics based ISO 25010:2011, in: Comput. Sci. Inf. Technol. - Proc. 11th Int. Sci. Tech. Conf. CSIT 2016, 2016, pp. 80–83, doi:10.1109/STC-CSIT.2016.7589874.
[40] R. Harrison, D. Flood, D. Duce, Usability of mobile applications: literature review and rationale for a new usability model, J. Interact. Sci. 1 (1) (2013) 1, doi:10.1186/2194-0827-1-1.
[41] Y. Anzai, K. Ogawa, H. Mori, N. Bevan, Symbiosis of Human and Artifact Usability is Quality of Use (1995) 349–354.

Leonice Maria Reis de Souza Pereira is a Ph.D. student in Computer Science and Researcher in Assisted Living Computing and Telecommunications Laboratory - AL-Lab, a research laboratory within the Networks Group of the Instituto de Telecomunicações at the University of Beira Interior (UBI), Covilhã, Portugal. She graduated in Computer Science (Federal University of Uberlândia -UFU, Uberlândia, Brasil), Specialist in Communication Networks - Artificial Intelligence (Gama Filho Universtity "UGF" Rio de Janeiro, Brasil) and M. Sc. in Data Visualisation and Processing (UFU, Uberlândia, Brasil). She is involved in project of Software Engineering and Ambient Assisted Living. Her research interests include Health monitoring, Ambient Assisted Living, Internet of Things (IoT), Artificial Intelligence (AI), predictive algorithms and machine learning.

Dr. Sofia Ouhbi is an Assistant Professor at the UAEU. She holds two doctoral degrees in Software Engineering from ENSIAS, Morocco (2015) and from University of Murcia, Spain (2017). Prior to joining UAEU, Dr. Ouhbi was an Assistant Professor at the International University of Rabat and was a Visiting Researcher at the IT University of Copenhagen in Denmark. She has an active participation in organising and chairing conferences and she has experience as a peer-reviewer for journals listed in Thomson Reuters JCR. She has co-authored more than 20 publications in peer-reviewed conferences and JCR journals. Dr. Ouhbi has intersectoral skills through her work experience both in academia and industry (in Hewlett-Packard 2009–2011). She has contributed to many national and international research projects and technology transfer contracts whose topics were related to Software Engineering and Health Informatics.

Dr. Nuno Pombo is an Assistant Professor at University of Beira Interior (UBI), Covilhã, Portugal. His current research interests include information systems (with special focus on clinical decision support systems), data fusion, artificial intelligence, and software. He is the coordinator of the Assisted Living Computing and Telecommunication Laboratory (ALLab) at UBI. He is also member of BSAFE Lab, and Instituto de Telecomunicações - IT at UBI. Member of COST Actions: CA15109 "European Cooperation for Statistics of Network Data Science (COSTNET)."

Chapter 4

A process model for quality in use evaluation of clinical decision support systems

The chapter presents a proposed process model for evaluation of quality in use characteristics of clinical decision support systems.

The international standard ISO/IEC 25022 - Measurement of quality in use, was used as a guideline for the measurement, as well as an adaptation of the GQM method for the choice of questions and metrics.

The chapter is presented as an article named “A process model for quality in use evaluation of clinical decision support system.”

Leonice Souza−Pereira, Sofia Ouhbi and Nuno Pombo.

Published in Journal of Biomedical Informatics, 2021, in the area of Computer Science.

H−Index 103, Q1, by Scimago Journal & Country Rank.

Journal of Biomedical Informatics 123 (2021) 103917

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Original Research

A process model for quality in use evaluation of clinical decision support systems☆

Leonice Souza-Pereira [a,b,c,*], Sofia Ouhbi [d], Nuno Pombo [b,c]

[a] Instituto Federal do Triângulo Mineiro, Campus Uberlândia Centro, Brazil
[b] Instituto de Telecomunicações, Lisboa, Portugal
[c] Universidade da Beira Interior, Covilhã, Portugal
[d] Computer Science and Software Engineering Department, CIT, UAE University, United Arab Emirates

ARTICLE INFO

Keywords:
Clinical decision support systems
CDSS
Evaluation model
Quality in use
Quality measures

ABSTRACT

Context: Clinical decision support systems (CDSSs) are used to help healthcare professionals in making decisions, offering them a tool for improved medical care practices based on monitoring and management procedures. Although CDSSs exhibit many advantages, challenges remain in terms of their adoption in the clinician community. One such issue is related to user satisfaction and the system reliability. Ensuring the quality of CDSSs is a way to improve their acceptance and adoption.
Objective: This study aims to propose a process model for evaluation of the quality in use characteristics of a CDSS to identify deficiencies that reduce its use by healthcare professionals.
Methods: We reviewed the existing literature on CDSS assessment and developed a process model based on the international standards ISO/IEC 25010 System and software quality models, and ISO/IEC 25022 Measurement of quality in use. To select measures for evaluating these characteristics, we adopted the Goal-Question-Metric (GQM) method. We evaluated the quality in use characteristics because they can represent system usability. Measurement of these characteristics helps us understand user needs, improve the user experience, and mitigate the low acceptance of CDSS, particularly by the primary users.
Results: We developed a process model for measuring the quality in use (QiU) characteristics of CDSSs, explaining its applicability through an illustrative example focused on the characteristics of satisfaction and efficiency.
Conclusion: We consider that the proposed process model will benefit the CDSS adoption and contribute to the improvement of the quality of such systems by measuring its QiU.

1. Introduction

The quality of software products refers to the degree to which a software system satisfies the specified requirements, as well as users needs, and expectations [1].

A way to verify this degree is to measure the quality attributes of the software, which allows us to identify characteristics that must be improved to increase the software quality [2].

International standards of concepts, procedures, and metrics have been developed to measure software quality, and the latest series is ISO/IEC 25000 [3]. It includes models and frameworks for assisting software developers in assessing the quality of software products.

Clinical decision support systems (CDSSs) are tools designed to assist health professionals in decision-making [4,5] and also patients in terms of awareness and better understanding of their conditions [6]. CDSSs are tools that can empower healthcare professionals with improved medical health practices based on monitoring and management procedures [7–9].

An adequate CDSS produces reliable results, satisfies user expectations, and is useful [10,11]; however, despite the benefits of CDSSs, their

☆ This work is funded by FCT/MCTES through national funds and when applicable co-funded by FEDER—PT2020 partnership agreement under the project UIDB/EEA/50008/2020. (Este trabalho é financiado pela FCT/MCTES através de fundos nacionais e quando aplicável co-financiado por fundos comunitários no âmbito do projeto UIDB/EEA/50008/2020). This article is based upon work from COST Action IC1303-AAPELE-Architectures, Algorithms and Protocols for Enhanced Living Environments and COST Action CA162 26-SHELD-ON-Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology). More information in www.cost.eu. This study also is partially funded by UAEU.

* Corresponding author at: Computer Science Department - Universidade da Beira Interior, Rua Marquês de Ávila e Bolama, 6201-001 Covilhã, Portugal.
E-mail address: leonice.pereira@ubi.pt (L. Souza-Pereira).

https://doi.org/10.1016/j.jbi.2021.103917
Received 18 May 2021; Received in revised form 15 September 2021; Accepted 15 September 2021
Available online 24 September 2021

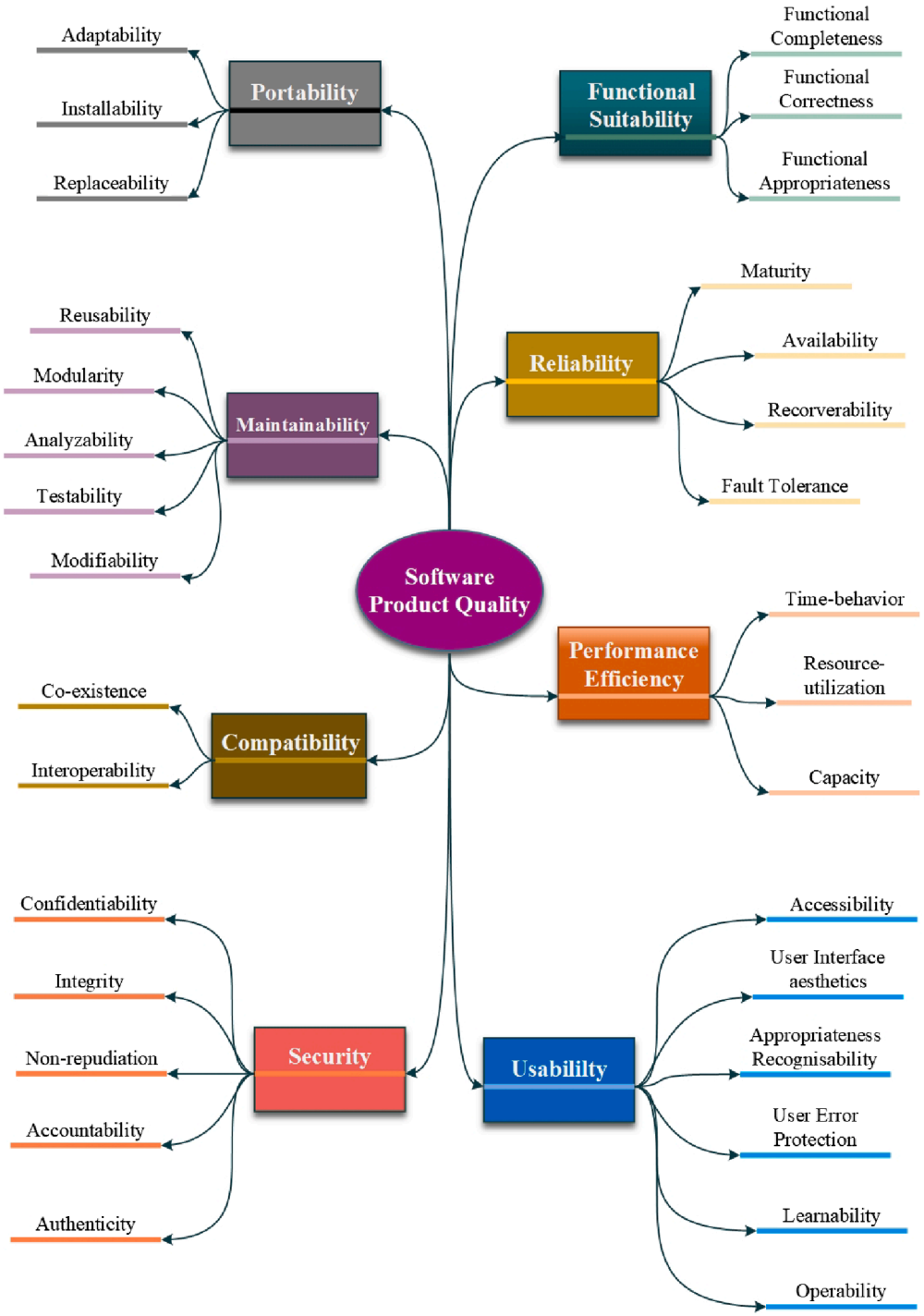

Fig. 1. Quality characteristics and sub-characteristics of the ISO/IEC 25010 - Software Product Quality.

acceptance in the medical field is still contentious [12,13]. CDSS adoption is related to several factors, e.g., the end user's perception of the system's ease of use, its usefulness, the quality of its outputs and its reliability [13].

Khairat et al. [12] presented a list of unfavourable reasons for the acceptance of CDSSs that included two quality sub-characteristics, i.e., trust and overall user satisfaction. The authors suggested that including physicians in the design process could help ensure that user needs and expectations are satisfied. In addition, Moja et al. [14], found that usability is a key facilitator or barrier to the acceptance and use of CDSSs, such acceptance depends on its workflow-oriented context-sensitive accessibility [15], as well as its availability, and interoperability. CDSSs are often developed for a very specific purpose (e.g., cancer diagnosis); however, the development process may not consider other routine activities (e.g. considering pathogenic comorbidity) from the user's workflow, or the system may not be integrated into existing in-use systems [6].

Measuring quality characteristics can contribute to mitigating the reasons for the non-acceptance of CDSSs, since such measurements can identify shortcomings and weaknesses of the system that affect its reliability, and failures to satisfy user requirements, or even they can recognise problems in CDSSs development.

However, due to time and budget constraints, all quality characteristics are difficult to assess; therefore, the critical quality resources to be

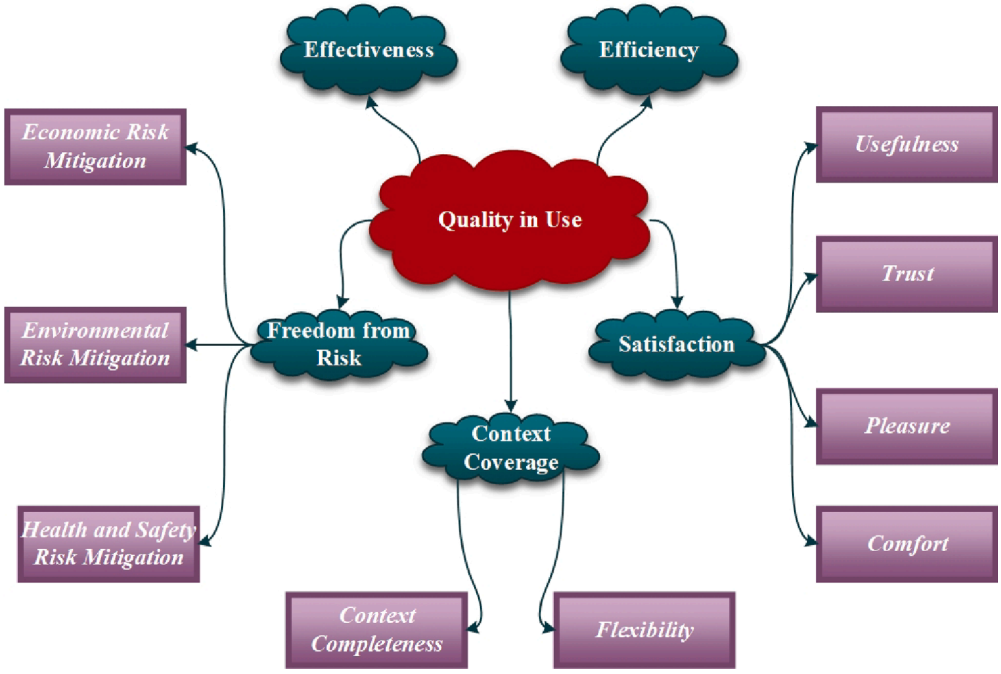

Fig. 2. Quality characteristics and sub-characteristics of the ISO/IEC 25010 - Quality in Use.

Table 1
Normalisation options.

Measure ID	Measure name	Measurement Function	Confor mance	Bench mark	Time series	Profi ciency	Population norm
SUs-1-G	Overall Satisfaction	$X = \sum(A_i)$; A_i= Response to a question	√	√	√	√	√
SUs-2-G	Satisfaction with features	$X = \sum(A_i)$; A_i =Response to a question related to a specific feature.	√	√	√	√	
SUs-3-G	Discretionary usage	$X = \frac{A}{B}$; A = Number of users using a specific function, application or system; B = Number of potential users who could have used the specif function, application or system.	√	√	√	√	
Ey-5-S	Fatigue	$X = 1 - \frac{A}{B}$; A = Current performance; B = Initial performance.	√	√	√	√	

Examples of normalisation of measures, where **SUs** means (S) Satisfaction characteristic, (Us) Usefulness sub-characteristic; **Ey** means Efficiency characteristic [18].

measured in a software product, in this case a CDSS, must be identified in a CDSS.

In our previous study [16], we collected the characteristics assessed in nine studies that evaluated the quality of CDSSs. We observed that software product quality characteristics are evaluated more than quality in use (QiU) characteristics. Therefore, we infer that the assessment of quality in use was very little addressed, especially if we consider how important it can be in measuring the quality perceived by end-users.

In this study, we present a process model for quality in use (QiU) assessment of a CDSS. We reviewed the existing literature on CDSS and software quality and built it on the ISO/IEC 25010 and ISO/IEC 25022 standards. The proposed process model is expected to contribute to better and more accurate CDSS assessment and subsequently to mitigate their non-acceptance by health professionals.

The remainder of this paper is organised as follows. In Section 2 we present background information and definitions required to understand the proposed model. The study methodology is described in Section 3, including the selection process of the relevant literature on QiU for CDSSs. The proposed process model and its phases are presented in Section 4. Section 5 discusses a case-study simulation of the proposed model in application. The paper is concluded in Section 7, including a discussion of identified limitations and corresponding future work.

2. Background

2.1. Software quality models

ISO/IEC 25000, which is also refereed to the System and Requirements for Software Quality and Evaluation (SQuaRE) [3], is the latest version of a series of standards formed to create a framework to evaluate the quality of software products. It consists in five divisions, which are 1) ISO/IEC 2500n - Quality Management Division, 2) ISO/IEC 2501n - Quality Model Division, 3) ISO/IEC 2502n - Quality Measurement Division, 4) ISO/IEC 2503n - Quality Requirements Division and 5) ISO/IEC 2504n - Quality Evaluation Division. Two of these divisions, ISO/IEC 2501n and ISO/IEC 2502n, are addressed in this study.

ISO/IEC 2501n - Quality Model Division The standards in this division presents quality models for computer systems and software products, quality in use, and data. It comprises ISO/IEC 25010 - System and software quality models, and ISO/IEC 25012 - Data Quality model.

ISO/IEC 25010 - System and software quality model division is part of the standard ISO/IEC 2501n [17] and comprises two main subdivisions:

1. **The software product quality model**, that includes eight characteristics and relates to the static properties (i.e., internal quality

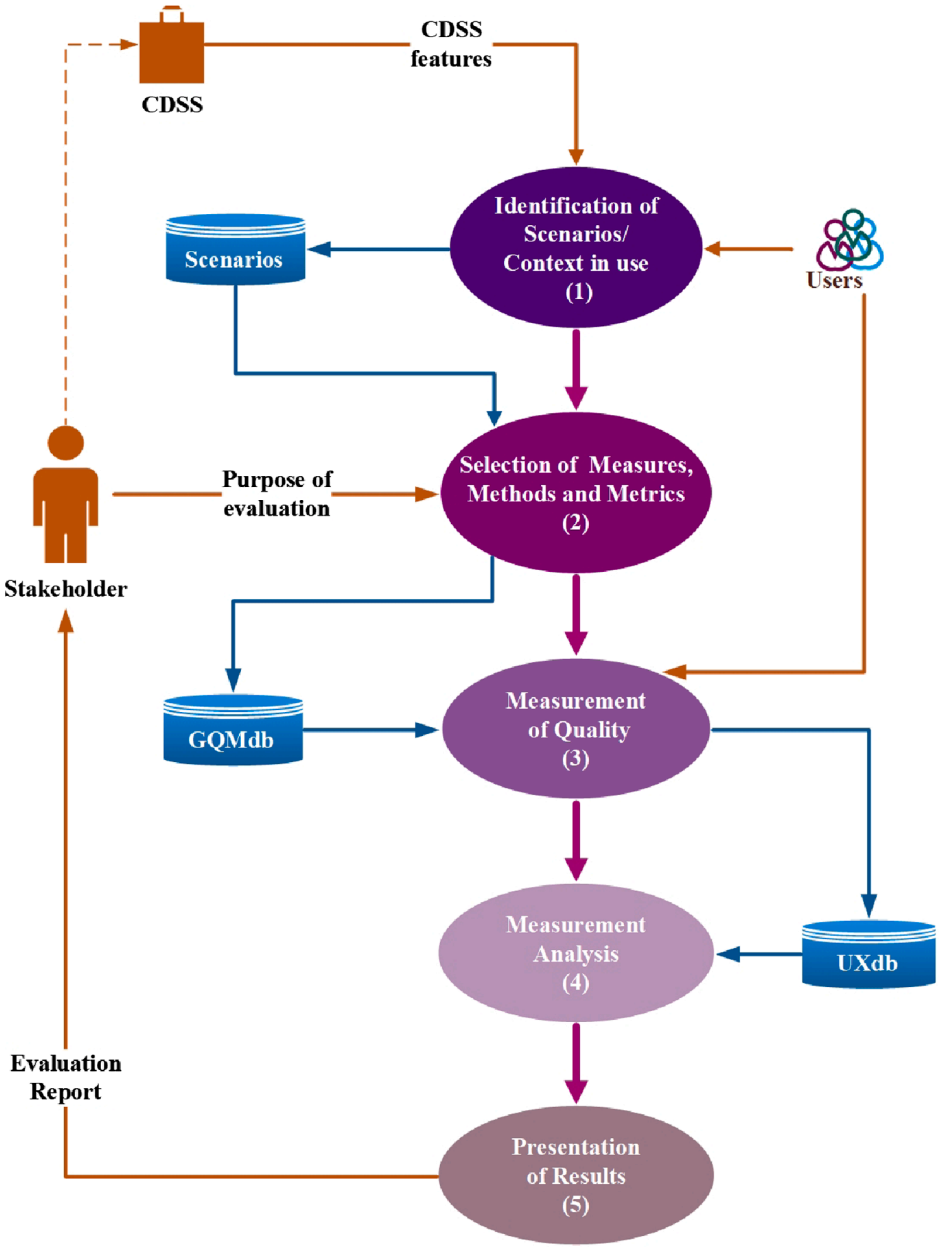

Fig. 3. Proposed process model for QiU evaluation.

attributes) of software and dynamic properties of the computer system (i.e., external quality attributes).

2. **The QiU model**, which includes five characteristics and considers the user's interaction with the software in a context. According to the international standard ISO/IEC 25010, the QiU characteristics are described as follows [17].
   - Effectiveness - means the level of precision and completeness with which users achieve their specific goals when using the system.
   - Efficiency - refers to the resources spent to achieve the goals and its measure is related to the level of effectiveness achieved with the consumed resources.
   - Satisfaction - refers to whether user requirements are satisfied in a particular context of system use.
   - Freedom from risk - refers to the degree to which the quality of a system reduces or avoids potential risks to human life, the economic situation, and health of the environment.
   - Context coverage - deals with the use of the system in all specific contexts and/or in contexts that extend beyond the initially identified contexts. Context completeness and flexibility are the sub-characteristics that represent context coverage.

Fig. 1 shows the ISO/IEC 25010 software product quality model, while Fig. 2 shows the ISO/IEC 25010 QiU model.

ISO/IEC 25022 - Measurement of Quality in Use is a SQuaRE's standard [18], part of the ISO/IEC 2502n division, that deals with the measurement of software QiU characteristics of a software product. It comprises a series of measures associated with each QiU characteristic, including the methods and functions that are applied for measurement. Moreover, it provides instructions for normalisation and interpretation of these measures (see Table 1) as well as a guideline to develop the QiU evaluation process. The proposed model (Section 4) follows the guidelines of this standard.

2.2. GQM method

The GQM method, which was developed by Basili and Weiss [19],

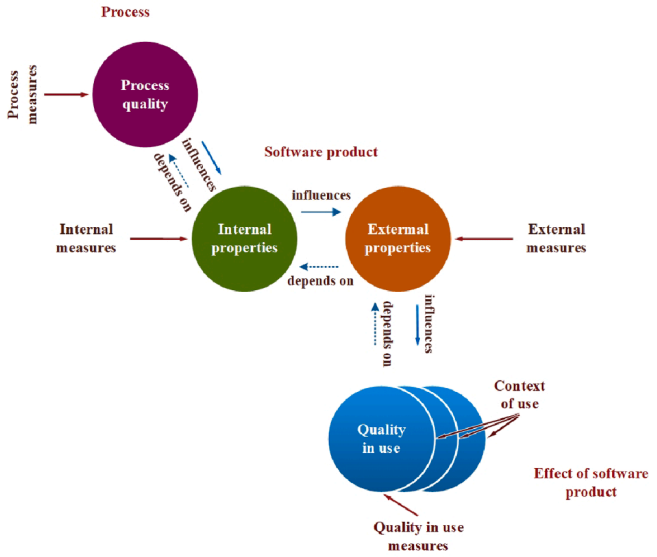

Fig. 4. Quality in the life-cycle of software development.

includes four phases.

1. The planning phase, where the project to be measured, is selected and defined.
2. The definition phase, where the measurement goal, corresponding questions, employed metrics, and hypotheses are documented.
3. The data collection phase.
4. The interpretation phase, where data is collected and transformed to measurement results by applying predefined metrics.

The results of the last step provide answers to the defined questions, denoting the achievement of the goal and its evaluation. Several studies have employed the GQM approach and adapted this method to match their proposals, simplified it, or used it to develop their own methods [20–24].

In this study, we adapted this method to select the measures to evaluate QiU characteristics.

2.3. User experience, usability, and QiU

Usability can be defined by the extent or degree "to which a system, product or service can be used by specified users to achieve specific objectives with effectiveness, efficiency and satisfaction in a specified context of use" [25,17].

In addition to being a product quality characteristic, usability also can be considered a subset of the QiU, once it can be directly specified or calculated via QiU measures [17,18].

Although usability is an important factor in the adoption of a system, limiting the QiU of the system to only usability is a mistake. Other characteristics, e.g., flexibility, stability, performance, and portability are also important [26].

According to [25], user experience (UX) represents a user's perceptions and responses that result from using a of a system, product, or service. Therefore, although it is somewhat redundant, UX is the experience observed or acquired by a user when using the system relative to system's presentation, functionality, performance, usability, interactive behaviour and the assistance resources.

In terms of measuring usability, it can be considered that UX partly reflects the QiU of the system, because usability is considered the heart of UX and also part of QiU [26–28]. A system's QiU denotes how well a system's UX is performing, and an effective and efficient UX is an important element in any software product.

2.4. Previews considerations

In this section, we present some useful definitions to help understand the QiU evaluation process model that will be presented in this study, particularly in Fig. 3.

- CDSS: represents a computer-based program or a prototype to assist healthcare professionals in decision making. It can also be the executable module(s) of a CDSS, a mobile or web application (app). Key features are extracted from it such as functionalities (1) that produce the expected outcomes from CDSS, (2) that are required for another functionality to produce a result, or (3) that trigger or impact other important functionality. Such features will be evaluated during the process.
- Users: These are the end-users that execute and/or operate the CDSS's tasks. From users are extracted information as skills, experiences, and expectancy about the system. Users are the focus of this study as the most important actors in the CDSS acceptance process.
- Stakeholder: A stakeholder is a person or group of people who (1) requests the evaluation and defines its proposal, (2) receives the result of the QiU measurement, (3) finances the evaluation, (4) operates the system, or (5) develops the system. The end-user is among the stakeholders.
- Evaluation Purpose: This represent the reason why the QiU will be measured. The evaluation purpose is determined by a set of ideas and needs that justify the measurement, and the measures to be evaluated are based on this purpose.
- GQMdb: Goal-question-metric database (GQMdb) is a dataset with the questions elaborated with the adapted method goal-question-metric (GQM), which is employed to assess the system quality. GQM comprises three stages, i.e., (1) defining a goal, (2) defining questions for each goal that help to realise the goal, and (3) defining a set of metrics to provide an answer to each question [20].
- UXdb: User experience database (UXdb) is a dataset containing the results of the measurement process and the UX interpretations. This information is used to produce the final report.
- Evaluation report: The evaluation report describes the interpretations of measurement results, identifies critical points, and provides suggestion for improving the system.

3. Methodology

For this study we screened articles on system and software quality and QiU related to CDSSs.

In our previous studies [6,16], we identified software engineering techniques employed in the development of CDSS and the quality metrics adopted for evaluation purposes. In [6], we selected 339 studies in a systematic literature review, using the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA)[1] statement, and 14 candidate studies remained after the eligibility and exclusion criteria were applied. Such studies were detailed in terms of (1) main proposal, (2) data-sources, (3) software engineering techniques, (4) performance assessment, and (5) adopted evaluation metrics.

In [16] we collected the characteristics assessed in nine studies that evaluated the quality of CDSS where we observed that software product quality characteristics were the focus of the evaluation process.

Both our previous studies are relevant because software quality depends on the quality policies implemented during the software development process. Fig. 4 shows the dependence of the quality characteristics and the process of life-cycle of software development.

The rationale for our research methodology is based on the observed fact that, to the best of our knowledge, QiU assessment is understudied or even neglected in literature.

[1] http://www.prisma-statement.org/.

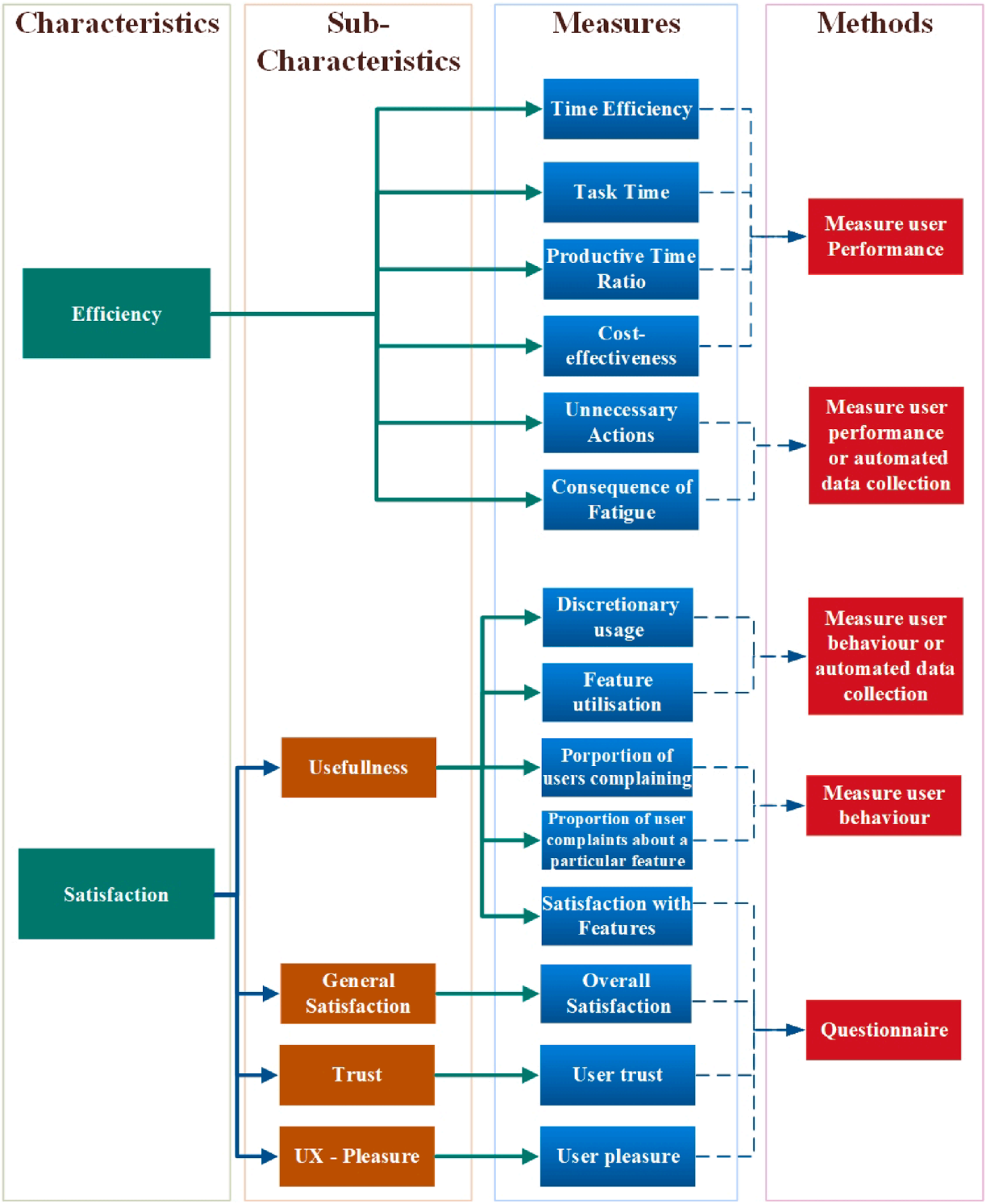

Fig. 5. Characteristics, sub-characteristics, measures and methods considered by the proposed evaluation model according to ISO/IEC 25022. All measures of satisfaction QiU are classified as general, and are therefore recommended to be measured. For efficiency QiU, only *task time* is considered a general measure.

Therefore, measuring these characteristics is a way to understand user requirements and needs. Thus, we can determine what prevents widespread use of CDSSs, provide reasons to improve CDSS, and provide a better UX [29].

Thus, we created a process model for evaluation of QiU characteristics for CDSSs (Fig. 3).

We adapted the GQM method as way to select the measures to evaluate QiU characteristics, and inserted it on one of the process model phases - Phase 2 (SubSection 4.2). We used this GQM's adaptation to select the measures to assess the satisfaction and efficiency QiU, characteristics approached in our previous study [16] as important to reduce the non-acceptance of CDSS, as a sample for an illustrative explanation of the proposed process model.

A possible formulation of a adapted GQM list is presented in Appendix B. The goal (G) of each question is a specific scenario (*Sn*), with the focus on the (*FO*) analysis. The formulated adapted GQM list is recorded in GQMdb.

Table 2
Functionalities (F).

Id	Functionality description
f_1	Calculate survival rate
f_2	Calculate prognostic score
f_3	<other functionality>
f_4	<other functionality>
f_n	<other functionality>

$F = \{f_1, f_2, \ldots, f_n\}$ where $n \geqslant 1$.

Table 3
Outcomes (O).

Id	Outcome
o_1	Survival rate for Colon Cancer
o_2	Survival rate for Breast Cancer
o_3	Score for renal carcinoma
o_4	<other outcome>
o_m	<other outcome>

$O = \{o_1, o_2, \ldots, o_m\}$ where $m \geqslant 1$.

Table 4
Functionality-Outcome (FO).

Id	Functionality-Id	Outcome-Id
fo_1	f_1	o_1
fo_2	f_1	o_2
fo_3	f_2	o_3
fo_4	f_3	o_4r
fo_k	f_n	o_m

$FO = F \rightarrow O$ or $FO' = F \rightarrow F$ and $k \geqslant 1$.

Table 5
Users (U).

Id	User identification	Experience level	Ability level
u_1	John	5	8
u_2	Moanna	8	7
u_3	Marie	10	10
u_4	Peter	7	9
u_p	Lucca	5	6

Users selected to evaluate the CDSS, where $p \geqslant 5$.

Table 6
Scenarios - $Sn(U, FO)$.

Scenario	User	Functionality-Outcome
sn_1	u_1	fo_1
sn_2	u_2	fo_1
- -	- -	- -
sn_5	u_5	fo_1
sn_6	u_1	fo_2
sn_7	u_2	fo_2
- -	- -	- -
sn_h	u_5	fo_k

$Sn = sn_1, sn_2, \ldots, sn_h$ where $h = p^*k$

4. Proposed process model for evaluation of QiU

Fig. 3 presents the proposed model for evaluating the QiU characteristics of a CDSS. The proposed model comprises five main phases: (1) identification of scenarios and context of use, (2) selection of measures, methods, and metrics, (3) quality in use measurement, (4) measurement analysis, and (5) presentation of results. These phases are detailed below.

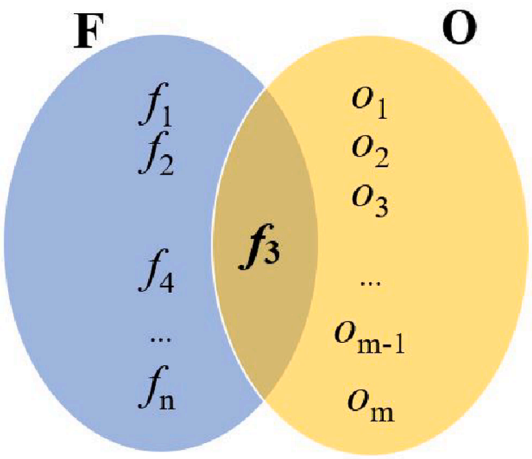

Fig. 6. Venn diagram of functionalities and outcomes (showing f_3 as functionality (producing an outcome) and outcome (triggered by f_4)).

Table 7
Simulated values for efficiency measures.

Goal: 1		
User ID	Task-Time (min)	Events
U1	6	user used online help
U2	5	–
U3	6	user was interrupted twice
U4	3	–
U5	7	user used online help

Assumed rounded values for a measurement of efficiency QiU, using GQM G_1 (Appendix B).

4.1. Phase 1: identification of scenarios and contexts-in-use

This first phase identifies the contexts for using the CDSS and the scenarios where the main functionalities are performed. Here, these features and the results they produce, as well as the skill level of users using these features, need to be known.

This phase involves questions designed to identify the main functionalities of the CDSS and their key characteristics. These questions help us discover the tasks that are required to achieve the desired outcome of a specific functionality and identify the difficulty level of performing each task (additional information is provided in Appendix A). Note that user skills are got using the same method (Appendix A).

This is necessary because this functionality will be tested by users with unique skills and expertise, therefore, with different difficulties and perceptions. Each user running one of these features implies a different scenario. This idea is best explained when we represent it through tables.

From the CDSS, we get information about the CDSS's features, main functionalities, and outcomes. A mapping of the functionalities is performed and recorded. Section 5 - Illustrative Example, brings tables with this data.

Note that we only consider the key features, that accomplish (or interfere with) the major objectives of the CDSS, and the data comprises the scenarios considered in the evaluation.

4.2. Phase 2: selection of measures, metrics and methods

Using the identified scenarios, the evaluators must determine which

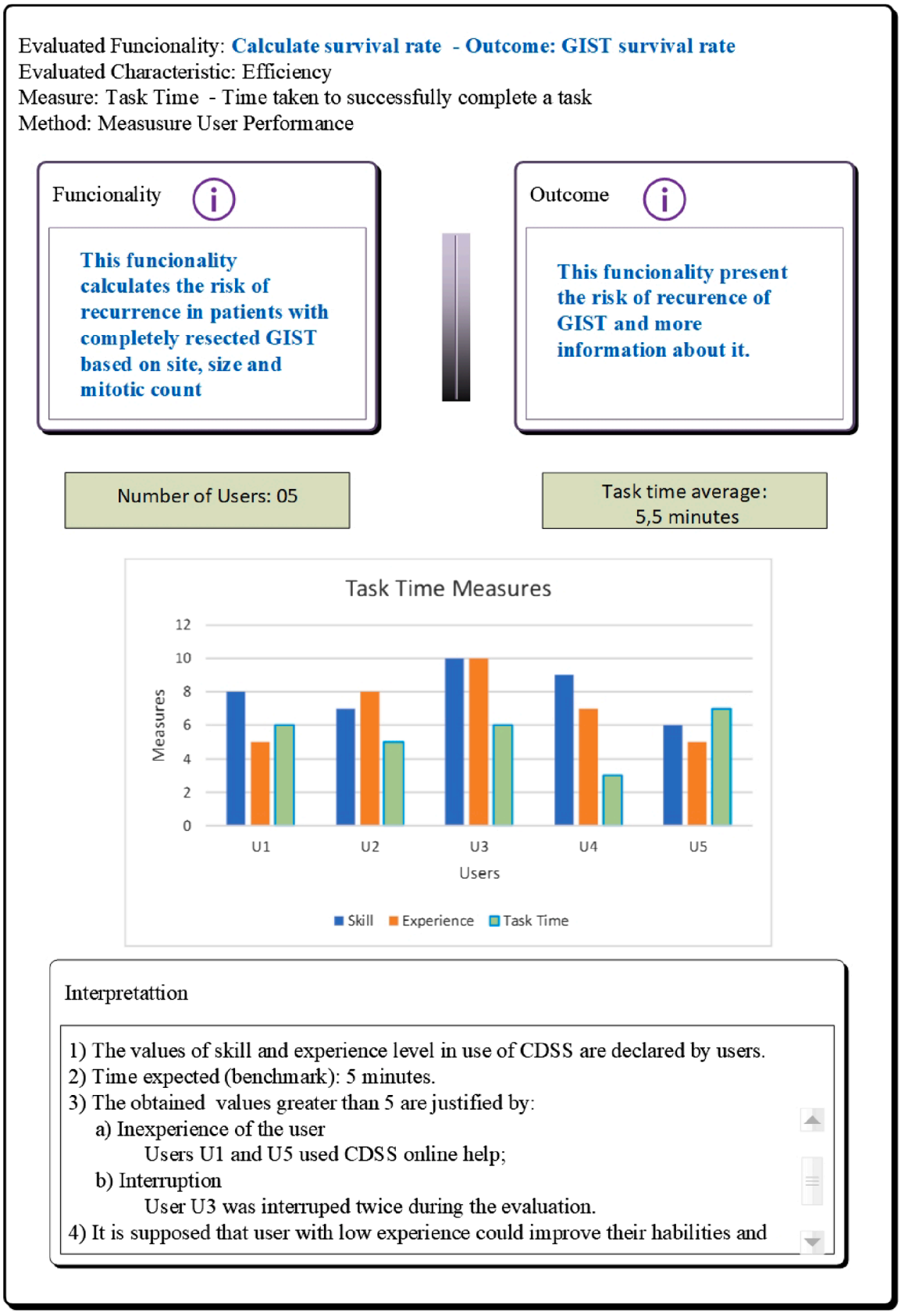

Fig. 7. Example measurement report: efficiency/task time.

Table 8
Questions to gathering information about the CDSS and Users.

ID	Target	Question	Information
01		What is the CDSS proposal?	This proposal is important because it will guide the quality analyst in the creation of questionnaires to measure some characteristics, e.g., satisfaction measure.
02	CDSS	What are the CDSS' outcomes?	This is necessary to know which outcomes will be used in the quality measurement.
03		What and how many are the CDSS's functionalities?	Here the intention is identifying the features used in the production of outcomes.
04		What are the benchmarks of the functionalities?	Based and similar system or functionality, a benchmark can be established for comparisons of results obtained.
05		What knowledge is expected from the user to perform each functionality?	Some activities require more experience to be performed. The comparison between the knowledge necessary and the user's knowledge is used to explain the metrics' results.
06		Which functionalities can be assessed?	It is necessary to select functionalities to be evaluated.
07	USER	What does the user expect from the system?	Here we collect the expectancy of user in relation to outcome produced. This is used to compare the goal of the CDSS and to measure the user's satisfaction.
08		What are the user's knowledge and expertise levels?	It is very important. If these levels are short, the time spent in the task's execution probably will be long. This must be considered when producing the end-report.

measures to apply to each scenario, according to the selected QiU characteristic, which can be any characteristic of the ISO/IEC 25010 QiU model.

Once the characteristics and sub-characteristics are selected, the corresponding measures must be selected. These measures are classified as **Generally (G)**, when they are generally applicable and used in a wide range of situations, and **Specialised (S)** measures when they are used for specific needs (e.g. measuring the time efficiency to know the efficiency with which users reach their goals over time when using the system) [18].

According ISO/IEC 25022, for each measure, the standard presents a function and a method to apply to obtain this measure. The measurement of all the quality measures classified as **(G)** by the SQuaRE document is strongly recommended, and if any exclusion is made, a valid justification for the exclusion is required [18]. Note that a justification is also necessary for any additional measure.

The scope of the satisfaction characteristic can be measure by an overall generic measure of satisfaction, a specific satisfaction sub-characteristic or a combining measures of individual sub-characteristics [18]. To evaluate the satisfaction characteristics in a CDSS, we consider measuring the following sub-characteristics.

- We consider general satisfaction to verify the overall user satisfaction.
- We consider usefulness to verify the user's satisfaction regarding their perception of the achievement of practical objectives.
- We consider trust to assess whether the CDSS is behaving as intended and expected, which helps us measure the degree of user confidence in the system.
- We consider the pleasure to assess the degree to which the user's needs for pleasure are satisfied.

Note that efficiency has no sub-characteristics, unlike the satisfaction characteristic (Fig. 2).

Fig. 5 shows the sub-characteristics, measures, and methods of ISO/IEC 25022 [18] that can be considered by the proposed model, according to our suggestion for the characteristics to be evaluated.

Design the evaluation plan: The evaluation plan is prepared using the chosen scenarios to be evaluated; the selected characteristics and sub-characteristics (if any); the GQM list with questions, measures, metrics, and the functions to be applied on the measurement.

This plan must contain the planning and implementation methodology, resource availability, questionnaire formulations (if applicable), user schedules, a schedule for the disclosure, and other actions required for the evaluation.

A guideline explaining the target scenario to be assessed should be given to the user. Therefore, the user will be informed regarding which functionality will be evaluated and the expected result.

4.3. Phase 3: QiU measurement

In this phase, the measurement function is calculated by performing user tests and collecting data from the scenario(s) selected by the evaluator according to the evaluation plan.

Here, in this assessment stage, when possible, give preference to performing the measurement in the real working environment of the user. Users perform differently when subjected to workload pressure, noise interference and other factors that are part of their normal routines.

This can allow the user's interactions with the CDSS to be more accurate. Another key factor is the lack of user support (except those provided by the CDSS).

Table 9
Adapted GQM.

ID	Goal	Scen*	Questions	Measures	Methods
G_1	Measurement of Efficiency on fo_1 execution	(Sn_1) to (Sn_5)	How many tasks are necessary to perform the functionality? How difficult is to achieve the outcomes? How long does it take to perform each task? How long does it take to produce an outcome? How long does it take to run the functionality?	task time, time-efficiency, cost-effectiveness.	measure user performance.
G_3	Measurement of Efficiency on fo_2 execution	(Sn_6) to (Sn_{10})	How long does it take for the user to execute successfully the feature?	task time	measure user performance.
G_{15}	Measurement of Satisfaction on fo_2 execution	(Sn_{h-4}) to (Sn_h)	How is the general satisfaction with CDSS? How is the general satisfaction with this functionality? Is the outcome according to what it is expected? How reliable do you think this result is?	general, usefulness, trust	questionnaires

Scen* = scenarios where the same functionality-outcome fo_k is measured for different users.

The evaluator must accompany the users and document their impressions and difficulties or problems encountered during task execution. In addition, the evaluator measures the time spent performing the target activity.

However, the evaluator cannot help the users during the task execution. Note that the process can be recorded if the evaluator's presence causes user discomfort.

The interaction between the evaluator and user occurs at the end of the evaluation session to discuss and clarify observations, and confirm the UX collected during the measurement. Here, the UX refers to how the user perceived and interacted with the CDSS.

All this information is recorded in UXdb and the results of the measurement of quality are actually the UX extracted from users, after their interaction with the CDSS.

According to ISO/IEC 25022 [18], in terms of evaluation context, reliable results could be obtained with a group of eight participants [2]. On the other hand, Nielsen *et al.* [30,31] demonstrated, using a mathematical model, that five is an optimal number of users to evaluate the systems, because they found 75% of the usability problem (achieving the best cost-benefit ratio).

Although the authors have reached this value in their research, it is important that the cost-benefit is in-line with the importance given to the evaluation and the benefits achieved with it. The context to be assessed and what is expected of this assessment should be considered when defining a number of assessment participants.

Therefore, the evaluator should consider the proposal of the CDSS and how accurate the results of its functionalities must be in order to choose the number of users for its validation.

4.4. Phase 4: measurement analysis

The results are analysed after measuring the features. ISO/IEC 25022 [18] provides an annex with examples of normalisation of QiU measures, allowing easier interpretation of its values and meaning. These options are conformance, benchmarks, time series, proficiency, and population norms for satisfaction.

Thus, in this stage, the questions (Q) from GQM are really answered, and the function measurement results are interpreted. In the interpretation, the skills of the users who performed the activities, the degree of difficulty of execution, the experience and knowledge of the system, any interruptions that occurred, the task duration, and the provided answers must be considered.

Besides formulae to normalise the results, the ISO/IEC 25022 Standard facilitates and guides the interpretation of these measurements[3], and provides several methods to achieve this.

4.5. Phase 5: presentation of results

The preparation of a measurement report is essential to present the strengths and the weaknesses of the CDSS. This report allows the users to ratify or change their impressions regarding the CDSS, and other stakeholders (e.g., a technical team) can improve weaknesses and provide a new version of the CDSS. The measurement report should describe the assessed characteristics associated with the measured characteristics, a brief explanation of the function and metrics used, the measured values, and explanations of what they mean in the assessment's context. Furthermore, the report can refer to taking part in users and possible divergences in their evaluations. It contextualises the assessment, unforeseen events, level of expertise, and skills that may have impacted the assessment. Note that different visualisations, e.g., images, graphs, and tables, can be used to improve the measurement report.

5. Illustrative explanation

This section illustrates a brief and example of the evaluation of a hypothetical scenario of a fictitious CDSS, from which some data has been extracted and saved in tables. For explanation purposes, we name these tables as Functionality (F) table (Table 2). Likewise, the results/outputs of this functionality are equally mapped and registered in Outcome (O) table (Table 3). Special attention must be paid to functionalities that are triggered by other functionalities. These must be recorded in both tables (F) and (O) since they are both Functionality and Outcome.

As an illustration criterion, we present a GQM list (Appendix B), with data from a simulated scenario and QiU characteristics, measures, and functions. In addition, Moja et al. [14] presented usability as an important factor for acceptance. Usability translates into efficiency, effectiveness, and satisfaction [17,18]. So, we presented the efficiency and satisfaction QiU characteristics in our GQM list to be measured, once the evaluation of such characteristics is important to solve the problem of acceptance of CDSS by health professionals [16], which is one aim of this measurement.

5.1. Explanation of tables

Observe that we use general names for fictitious features and results only to exemplify the CDSS characteristics. Therefore, assuming a CDSS aimed at oncology, the tables can be explained as follows.

Table 2 **(F)**- *Functionalities:*. In this table are recorded the main functionalities of the CDSS. In Table 2 (**F**)- Functionalities we found two of them: f_1 - Calculate survival rate, and f_2 - Calculate prognostic score.

Table 3 **(O)**- *Outcomes:*. In this table are recorded the outcomes produced by the functionalities of Table 2 (**F**). In our example of Table 3 (**O**) we found three of them: o_1 - Colon Cancer survival rate, o_2 - Breast Cancer Survival rate, and o_3 - IMDC for renal carcinoma.

Table 4 **(FO)**- *Functionality-Outcome:*. This table represents the normalisation of Table 3 (**O**) once the same outcome could be produced by different functionalities. So, it represents the relation between Tables 2 (**F**) and 3 (**O**). Our example brings the outcomes o_1 and o_2 as results of the same functionality f_1, and the outcome o_3 as result of f_2.

Table 5 **(U)**- *Users:*. This table brings information about users. This information is useful to define the scenarios and to interpret the measurement results.

Table 6 **(SN)**- *Scenarios:*. Considered the **target** of our measurement, this table is the real representation of the CDSS scenarios, where the functionality execution by the users is related. Each user executing one functionality that produces one outcome is considered a scenario.

The relationship between tables (F) and (O) affords another normalised Functionality-Outcome (FO) table, as shown in Table 4 and Fig. 6.

In other words, in this phase, the association between a single selected user (u) (from Table 5 and a single functionality-outcome (fo) (from Table 4) is formed, thereby creating a dataset with the possible contexts in use, i.e., the scenarios (Table 6). Here, the relation $p \nleftrightarrow k$ is established, where **one** user must perform k relations, and **one** relation must be performed by p users.

Therefore, the maximum number of scenarios (h) is expressed as follow:

$$h = p^{*}k \tag{1}$$

where p is the number of users and k is the number of relations to be executed.

[2] ISO/IEC 25022, annex D, D4 [18] *by controlling the context of evaluation, experience has demonstrated that reliable results can be obtained with a sample of only eight participants (ISO/IEC 25062).*

[3] Annex B and Section 6.3 in [18]

5.2. Phases identification

Therefore, summarising the steps we must take to have the QiU characteristics of our example measured, we have:

1. Phase 1: Identify the scenarios.
   (a) We identify the main CDSS functionalities and outcomes; we collect the user skills (e.g., Tables 2–5).
   (b) Scenarios are identified (e.g., Table 6).
2. Phase 2: Select measures, metrics and methods.
   (a) Stakeholders and evaluators select QiU to be measured, e.g., satisfaction and efficiency QiU characteristics.
   (b) A GQM list is developed for the scenarios, using the adapted GQM method to select the measures to assess the satisfaction and efficiency characteristics (e.g., Appendix B).
   (c) The scenario to be evaluated is selected (e.g., Sn_1, that is John (u_1) executing the "calculate survival rate" feature (f_1) to get the survival rate for colon cancer (o_1))
   (d) The evaluation design document is created.
   (e) One item from the GQM list is selected (e.g., Goal G_1 - Measurement of Efficiency on fo_1 execution - Appendix B)
3. Phase 3: QiU measurement.
   (a) The evaluator accompanies each user in the test process.
   (b) The user follows the guidelines to perform the GQM selected goal $G1_1$ - Measurement of Efficiency on fo_1 execution.
   (c) The user performs the feature.
   (d) The evaluator measures the time required to successfully complete the task;
   (e) The evaluator notes all the unforeseen events that occurred during the activity.
   (f) The evaluator collects the UX.
4. Phase 4: Measurement analysis.
   (a) The evaluator interprets the collected data.
   (b) The evaluator formalises the result of Goal G_1 - Measurement of Efficiency on fo_1 execution.
5. Phase 5: Presentation of results.
   (a) The evaluator generates the evaluation report with the interpretation of the results.
   (b) The evaluator presents the results to the stakeholders.

Based on the this scenario, we assume that five users participated in the assessment.The evaluator observed each user, collected the corresponding information, and noted unforeseen occurrences during the evaluation period (Table 7). After all measurement process declared on the evaluation plan are performed, the evaluator meets with each user to verify their perspectives and observations about the process. This process is required to complete the interpretation of the general and individual results.

Then, to ensure that the interpretations are consistent with the data provided by the users, the evaluator generates the report to present to the stakeholders.

A possible interpretation of the results is shown in Fig. 7 (only the GQM G_1 assessment is shown in this figure).

6. Considerations

This paper studies the QiU as the cornerstone of a process model evaluation of CDSS. The novelty of this model relies on the combination of principles from both ISO/IEC 25010 and ISO/IEC 25022 standards, and the use of the GQM method to select functionalities to be assessed. Thus, aiming to improve the CDSS acceptance, the proposed process model combines satisfaction and efficiency characteristics as key performance indicators to illustrate the evaluation of the end-user experience.

In line with this, we expect to pave the way for the detection of limitations or inconsistencies related to either specification, design, or deployment of CDSS. In addition, our model supports the incremental development of the CDSS based on its assessment, which may contribute not only to continuous improvement of the software quality but also to increase its adoption by the healthcare community. The rationale is that since the CDSS are assessed more accurately and focused on the quality, then it probably will challenge practitioners and researcher on the development of novel practices and/or to improve existing ones during the CDSS development life cycle.

Furthermore, the proposed process model may be extended to similar domains (e.g., Personal Health Record systems - PHR), as long as the focus is still centred on the end-user. When this scenario occurs, we may maintain the principles of the proposed process model, i.e., the GQM and QiU settings. Otherwise, if the evaluation requires a different focus, for instance, on safety or performance, or yet developers as goal users, we should adopt another method to gather information or adapt the GQM method in another way, besides choosing other QiU than the presented in this study, in order to adapt to the functionalities and characteristics that refer to the focus.

7. Conclusion and future work

In this paper, we proposed a process model for the evaluation of quality in use of clinical decision support systems following the ISO/IEC 25022 and ISO/IEC 25010 standards.

The proposed process model employs an adapted GQM method to determine the measures to evaluate.

An illustrative example was provided to highlight the measurement of one just instance of our GQM list (Appendix B), with satisfaction and efficiency characteristics as examples of QiU to be measured.

Measuring the QiU characteristics may contribute to solving the problem of health professionals' reluctance to use these systems, because, through the identification of the causes and the possibility of fixing them, we can enhance their acceptance.

In future work, we plan to apply the proposed process model to a module or prototype clinical decision support system, which will begin in September 2021 at a public hospital in Covilhã, Portugal. The process will involve a specialised team of health professionals as end-users.

CRediT authorship contribution statement

Leonice Souza-Pereira: Data curation, Formal analysis, Investigation, Methodology. **Sofia Ouhbi:** Conceptualization, Writing-review & editing, Supervision, Project administration, Validation. Software.. **Nuno Pombo:** Conceptualization, Writing-review & editing, Supervision, Project administration, Validation, Supervision, Project administration, Funding acquisition, Resources, Software.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Leonice Souza-Pereira would like to thank the *IFTM - Instituto Federal do Triângulo Mineiro, Uberlândia, Brasil,* for their support, as well the *Fundação para Ciência e a Tecnologia, Portugal* and the United Arab Emirates University (UAEU).

Appendix A. Collecting information

An example of gathering information to help in the scenario structuring is presented in Table 8.

Appendix B. Adapted GQM

A fictitious example of GQM for the chosen scenarios and measures is presented in Table 9.

References

[1] J.P. Miguel, D. Mauricio, G. Rodríguez, A Review of Software Quality Models for the Evaluation of Software Products, Int. J. Softw. Eng. Appl. 5 (6) (2014) 31–53, https://doi.org/10.5121/ijsea.2014.5603. arXiv:1412.2977.

[2] F. Pinciroli, Improving Software Applications Quality by Considering the Contribution Relationship among Quality Attributes, Procedia Comput. Sci. 83 (2) (2016) 970–975, https://doi.org/10.1016/j.procs.2016.04.194.

[3] ISO 25000 STANDARDS. 2019a. https://iso25000.com/index.php/en/iso-25000-standards.

[4] R.T. Sutton, D. Pincock, D.C. Baumgart, D.C. Sadowski, R.N. Fedorak, K.I. Kroeker, An overview of clinical decision support systems: benefits, risks, and strategies for success, npj Digit. Med. 3 (1) (2020) 1–10, https://doi.org/10.1038/s41746-020-0221-y.

[5] E.S. Berner, Clinical Decision Support Systems, Health Informatics, second ed., Springer, New York, New York, NY, 2007. doi: 10.1007/978-0-387-38319-4. http://link.springer.com/10.1007/978-0-387-38319-4. ISBN 10: 0-387-33914-0.

[6] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, N. Garcia, Clinical decision support systems for chronic diseases: A Systematic literature review, Comput. Methods Programs Biomed. 195 (2020), https://doi.org/10.1016/j.cmpb.2020.105565.

[7] P. Shankar, N. Anderson, Advances in Sharing Multi-sourced Health Data on Decision Support Science 2016–2017, IMIA and Schattauer GmbH Adv. 27 (1) (2018) 16–24, https://doi.org/10.1055/s-0038-1641215, http://www.thieme-connect.de/DOI/DOI?10.1055/s-0038-1641215.

[8] T.J. Bright, A. Wong, R. Dhurjati, E. Bristow, L. Bastian, R.R. Coeytaux, et al., Effect of Clinical Decision-Support Systems, Ann. Intern. Med. 157 (1) (2012) 29, https://doi.org/10.7326/0003-4819-157-1-201207030-00450, http://annals.org/article.aspx?doi=10.7326/0003-4819-157-1-201207030-00450.

[9] P. Fraccaro, M. Arguello Castelerio, J. Ainsworth, I. Buchan, Adoption of Clinical Decision Support in Multimorbidity: A Systematic Review, JMIR Med. Informat. 3 (1) (2015) e4, https://doi.org/10.2196/medinform.3503.

[10] M. Khalifa, Clinical decision support: Strategies for success, in: Procedia Comput. Sci., vol. 37, 2014, pp. 422–427. doi: 10.1016/j.procs.2014.08.063. http://www.sciencedirect.com/science/article/pii/S187705091401028X, ISBN 9781614995388.

[11] D. Zikos, A Framework to Design Successful Clinical Decision Support Systems, in: Proc. 10th Int. Conf. PErvasive Technol. Relat. to Assist. Environ. PETRA '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 185–188. doi: 10.1145/3056540.3064960. doi:10.1145/3056540.3064960, ISBN 9781450352277.

[12] S. Khairat, D. Marc, W. Crosby, A. Al Sanousi, Reasons for physicians not adopting clinical decision support systems: Critical analysis, J. Med. Internet Res. 20 (4) (2018), https://doi.org/10.2196/medinform.8912.

[13] K.A. Alshare, M.K. Alomari, P.L. Lane, R.D. Freeze, Development and determinants of end-user intention: usage of expert systems, J. Syst. Inf. Technol. 21 (2) (2019) 166–185, https://doi.org/10.1108/JSIT-08-2018-0108, https://www.emerald.com/insight/content/doi/10.1108/JSIT-08-2018-0108/full/html.

[14] L. Moja, E.G. Liberati, L. Galuppo, M. Gorli, M. Maraldi, O. Nanni, et al., Barriers and facilitators to the uptake of computerized clinical decision support systems in specialty hospitals: Protocol for a qualitative cross-sectional study, Implement Sci. 9 (1) (2014) 1–6, https://doi.org/10.1186/s13012-014-0105-0.

[15] C.J. Schuh, J.S. De Bruin, W. Seeling, Acceptability and difficulties of (fuzzy) decision support systems in clinical practice, in: Proc. 2013 Jt. IFSA World Congr. NAFIPS Annu. Meet. IFSA/NAFIPS 2013, 2013, pp. 257–262. https://ieeexplore.ieee.org/abstract/document/6608409. doi:10.1109/IFSA-NAFIPS.2013.6608409, ISBN 9781479903474.

[16] L. Souza-Pereira, S. Ouhbi, N. Pombo, Quality-in-use Characteristics for Clinical Decision Support System Assessment, Comput. Methods Programs Biomed. 207 (2021) 106169, https://doi.org/10.1016/j.cmpb.2021.106169.

[17] Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQUARE) — System and software quality models, 2011. https://www.iso.org/standard/35733.html.

[18] Systems and software engineering — Systems and software quality requirements and evaluation (SQUARE) — Measurement of quality in use, 2016. https://www.iso.org/standard/35746.html.

[19] V.R. Basili, D.M. Weiss, A Methodology for Collecting Valid Software Engineering Data, IEEE Trans. Softw. Eng. SE-10 (6) (1984) 728–738, https://doi.org/10.1109/TSE.1984.5010301.

[20] Z. Bukhari, J. Yahaya, A. Deraman, A conceptual framework for metrics selection: SMeS, Int. J. Adv. Sci. Eng. Inf. Technol. 8 (6) (2018) 2294–2300, https://doi.org/10.18517/ijaseit.8.6.6441.

[21] K. Sacha, Evaluation of expected software quality: A customer's viewpoint, Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics) 2006;3922 LNCS(May), 2006, pp. 170–183. doi:10.1007/11693017_14.

[22] N. Tsuda, H. Washizaki, K. Honda, H. Nakai, Y. Fukazawa, M. Azuma, et al., Wsqf: Comprehensive software quality evaluation framework and benchmark based on square, in: 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP), 2019, pp. 312–321, https://doi.org/10.1109/ICSE-SEIP.2019.00045.

[23] J.M. Alves, A. Savaris, C.G. Von Wangenheim, A. Von Wangenheim, Software quality evaluation of the laboratory information system used in the Santa Catarina state integrated telemedicine and telehealth system, in: Proc - IEEE Symp Comput Med Syst 2016, 2016, pp. 76–81. doi: 10.1109/CBMS.2016.51.

[24] van Solingen, R., Berghout, E. The Goal/Question/Metric Method: a practical guide for quality improvement of software development. McGraw Hill, ISBN 1999;7 (October 2015):7. https://www.academia.edu/26126148/The_Goal_Question_Metric_method_a_practical_guide_for_quality_improvement_of_software_development.

[25] ISO 9241-210:2019: Ergonomics of human-system interaction - Human-centred design for interactive systems, 2019b. URL https://www.iso.org/standard/77520.html.

[26] I. Evans, C. Porter, M. Micallef, J. Harty, Test tools: an illusion of usability?, in: 2020 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), 2020, pp. 392–397. doi: 10.1109/ICSTW50294.2020.00070.

[27] H.M. Hassan, G.H. Galal-Edeen, From usability to user experience, in: 2017 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), 2017, pp. 216–222, https://doi.org/10.1109/ICIIBMS.2017.8279761.

[28] I. Atoum, C.H. Bong, Measuring Software Quality in Use: State-of-the-Art and Research Challenges, ASQSoftw. Quality Profess. 17 (2015) 4–15. URL http://arxiv.org/abs/1503.06934. arXiv:1503.06934.

[29] N. Bevan, Usability is Quality of Use, in: Y. Anzai, K. Ogawa, H. Mori (Eds.), Symbiosis Hum. Artifact, Advances in Human Factors/Ergonomics, vol. 20, Elsevier, 1995, pp. 349–354. doi: 10.1016/S0921-2647(06)80241-8.

[30] J. Nielsen, T.K. Landauer, A mathematical model of the finding of usability problems, In: Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems. CHI '93, Association for Computing Machinery, New York, NY, USA, 1993, pp. 206–213. doi:10.1145/169059.169166, ISBN 0897915755.

[31] Heuristic Evaluation: How-To: Article by Jakob Nielsen, 1994. URL https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/.

Chapter 5

Software quality: application of a process model for quality-in-use assessment

The chapter presents the results of the application of a process model for the evaluation of quality-in-use characteristics in clinical decision support systems, carried out through a questionnaire.

An adaptation of the GQM method was used to elaborate the enquiries and select the metrics to apply to the assessment. The questionnaire was composed of 26 questions, elaborated to measure the satisfaction and the efficiency quality-in-use characteristics of an oncological CDSS.

The chapter is presented as an article named "Software quality: application of a process model for quality-in-use assessment."

Leonice Souza−Pereira, Nuno Pombo and Sofia Ouhbi.

Published by Journal of King Saud University Computer and Information Sciences,2022, in the area of Computer Science.

H−Index 33, Q1, by Scimago Journal & Country Rank.

Journal of King Saud University – Computer and Information Sciences 34 (2022) 4626–4634

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Software quality: Application of a process model for quality-in-use assessment

Leonice Souza-Pereira [a,b,*], Nuno Pombo [a,b], Sofia Ouhbi [c]

[a] Universidade da Beira Interior, Covilhã, Portugal
[b] Instituto de Telecomunicações, Lisboa, Portugal
[b] Department of Computer Science and Software Engineering, CIT, UAE University, UAE

ARTICLE INFO

Article history:
Received 8 February 2022
Revised 9 March 2022
Accepted 31 March 2022
Available online 21 April 2022

Keywords:
Evaluation Model
Quality in use
Quality measures
Software quality assessment
ISO-IEC 25022
Clinical Decision Support Systems (CDSS)

ABSTRACT

Quality-in-use characteristics can be measured to assess the quality of the software and even point out problems in its production since they reflect the quality of the system perceived by the end-users identified in the running systems. Measuring such characteristics is a way of verifying that the software meets the requirements and expectations of the interested parties.

This study presents the validation of a process model for the evaluation of the quality-in-use characteristics of a system. The model works with the identification and measurement of important characteristics of the system to determine the quality of the software, and thus identify potential problems that may inhibit its use.

We applied our process model in a clinical decision support system in the oncology field using a questionnaire for gathering data for the measurement. The questionnaire mainly sought to identify user satisfaction with the use of the software, and the answers permitted us to measure the quality-in-use characteristics of the system.

Our experiments revealed the adequacy of our evaluation process model and the evaluated CDSS reached a score of 71.35, which is advantageous according to our standard of measures and meets the expectations of users.

1. Introduction

External quality attributes of a software, such as maintainability and usability, are affected by subjective factors, such as user experience (UX) (Sommerville, 2016), knowledge, and digital literacy. In addition, usability can be defined by the extent or the level "to which a system, product or service can be used by specified users to achieve specific objectives with effectiveness, efficiency and satisfaction in a specified context of use" (ISO/IEC, 2019; ISO/IEC, 2011). Indeed, the degree in which a software system satisfies specifics requirements, as well as users' needs and expectations, refers to the quality of the software product (Miguel et al., 2014).

With these principles in mind, to have an effective evaluation and guarantee of the software quality, it is necessary to follow models that may describe and track the quality, during the software development cycle, and in the course of its subsequent maintenance activity (Arnicane et al., 2020).

Thus, international software quality standards emerged to guide the achievement and measurement of this quality throughout concepts, procedures, and metrics . The recent series ISO/IEC 25000 - System and Requirements for Software Quality and Evaluation (SQuaRE) standard (ISO/IEC, 2019) includes models and frameworks that could be used to assist software developers in evaluating the quality of their products.

One of these models is the Quality in Use (QiU) model, composed of multiple characteristics (ISO/IEC, 2011), and other one which offers methods and metrics to measure that QiU (ISO/IEC, 2016).

According to Djordjevic (Djordjevic, 2017), product quality attributes are the cause, and QiU attributes are the effect. Whence, measuring and evaluating the QiU characteristics is a way to confirm the external quality of the software (measurement of behaviour), which can verify its internal quality (static measurement of intermediate products) . Measuring the QiU characteristics allows identifying system functionalities that must be improved to increase its quality (Pinciroli, 2016).

Since the usability is considered the cornerstone of the UX (Evans et al., 2020; Hassan and Galal-Edeen, 2017; Atoum and Bong, 2015) then is quite similar to measure any of them. In fact,

* Corresponding author.
E-mail addresses: leonice.pereira@ubi.pt (L. Souza-Pereira), ngpombo@di.ubi.pt (N. Pombo), sofia.ouhbi@uaeu.ac.ae (S. Ouhbi).

https://doi.org/10.1016/j.jksuci.2022.03.031
1319-1578/

the UX partially reflects the QiU of the system because it is a major parameter of user acceptance and satisfaction when using software. In addition, the QiU assumes extreme relevance when the software is operated either by non-digital users or users whose background or experience differs from the computer science umbrella. Congruently, scenarios in which the decision-making relies on software, such as healthcare or industry, are highly dependant on the QiU of the system.

Into the healthcare context, practitioners use computer systems to support them in the clinical decision-making, in terms of awareness and better understanding of their conditions (Souza-Pereira et al., 2020).

However, these systems, a.k.a. Clinical Decision Support Systems (CDSSs) challenge for its adoption due to a myriad of causes, such as: (1) end-user's perception of the system, (2) system usefulness, and (3) reliability of system outputs (Alshare et al., 2019; Khalifa, 2014; Zikos, 2017).

Therefore, measuring the quality characteristics of a CDSS can allow us to identify reasons for non-acceptance of CDSSs, since the values found can point to deficiencies and weaknesses in the system, which affect its usability. The measurement also permits to identity the non-compliance with user requirements, which harms UX, and it can even point to problems in the system's development.

In this paper, we have the ambition to validate a process model for quality in use assessment of a CDSS, developed in our previous work (Souza-Pereira et al., 2021b), into a CDSS context and present results and lessons learned.

Our proposed process model, based on the Goal-Question-Metric (GQM) method and following the international standards ISO/ IEC 25010 and ISO/ IEC 25022, aims to identify deficiencies in a CDSS by measuring QiU characteristics and, thus, contribute to the increase of the acceptance of the software by the medical community. This paper is organised as follows. In Section 2 we present background information and definitions required to understand our proposed evaluation process model and its validation.

The preparation for validating the proposed model is presented in Section 3, through the steps for choosing the CDSS to be evaluated and applying the model. Section 4 presents the application of the proposed process model. Section 5 discusses results of the measurement. The conclusion, limitations of the study, and future work are presented in Section 6.

2. Background

2.1. Software quality models

ISO/IEC 25000, or SQuaRE, (ISO/IEC, 2019), is a series of standards formed to create a framework for evaluating the quality of software products. SQuaRE is split into five divisions. We addressed two of them in our previous studies (Souza-Pereira et al., 2021a, 2021b): (1) Quality Model Division - ISO/IEC 2501n, and (2) Quality Measurement Division - ISO/IEC 2502n.

ISO/IEC 25010 - System and software quality model (ISO/IEC, 2011) is part of the ISO/IEC 2501n, together with the data quality model - ISO/IEC 25012. It comprises two main subdivisions:

1. **The software product quality model**, which includes eight characteristics and relates to the static properties (i.e., internal quality attributes), and dynamic properties (i.e., external quality attributes).
2. **The Quality in Use model**, which includes five characteristics and considers the user's interaction with the software in a context.

ISO/IEC 25022 - Measurement of Quality in Use (ISO/IEC, 2016) is the SQuaRE's standard that normalises the measurement of QiU characteristics of a software product. It comprises a series of measurements associated with the QiU characteristic (Efficiency, Effectiveness, Satisfaction, Freedom from risk and Context Coverage), including methods and mathematical functions that are applied for measurement, in addition to instructions for normalisation and interpretation of these measurements. It also presents a guideline for evaluating the QiU of a software system.

2.2. GQM method

The GQM method, which was developed by Basili and Weiss (Basili and Weiss, 1984), includes four phases: (1) The planning phase, where the project to be measured, is selected and defined; (2) the definition phase, where the measurement goal, corresponding questions, employed metrics, and hypotheses are documented; (3) the data collection phase, and (4) the interpretation phase, where the collected data are transformed to measurement results by applying predefined metrics.

Several studies have employed the GQM approach (Bukhari et al., 2018; Sacha, 2006; Tsuda et al., 2019; Alves et al., 2016; van Solingen and Berghout, 1998), adapting it to match their proposals, simplifying it, or using it to develop their own methods. We adapted the GQM to select measures to assess QiU characteristics.

3. Methodology

This section presents the steps considered for the application of our proposed QiU evaluation process model, from the choice of the CDSS to the measurement of the characteristics.

3.1. The CDSS's choice

The initial idea was to apply our proposed process model in a hospital, working directly with health professionals, with a CDSS already in use by them. Unfortunately, this became impossible due the Covid-19 pandemic situation, which affected and changed all situations that could be considered normal, especially within a medical-clinical environment. Thus, the solution was to apply the assessment in a free CDSS, in a non-restrictive way. We used the Google Store to select a CDSS that best met our expectations in order to identify potential usage scenarios and select QiU characteristics for the measurement, which in our case study was limited to Satisfaction and Efficiency QiUs.

3.2. The gathering process

The gathering process took place from the 2nd to the 5th of June 2021, with no new apps submitted after this date being considered. We found 248 apps that were carefully studied and classified into groups in order to identify the software's purposes.

CDSS was classified within its main proposal as:

1. CDSS;
2. educational software (service protocols, videos, glossaries, procedure flowchart, technical-scientific content, use case), guidelines, interpreters, chats, e-books software, guidelines, interpreters of clinical definitions;
3. clinical calculators (the ones with only this type of functionality),
4. Medical Health Record (MHR), Personal Health Record (PHR), workout apps, self-care app;
5. agendas or outside the clinical context and others.

Fig. 1 presents the selection process of the CDSS used in this study.

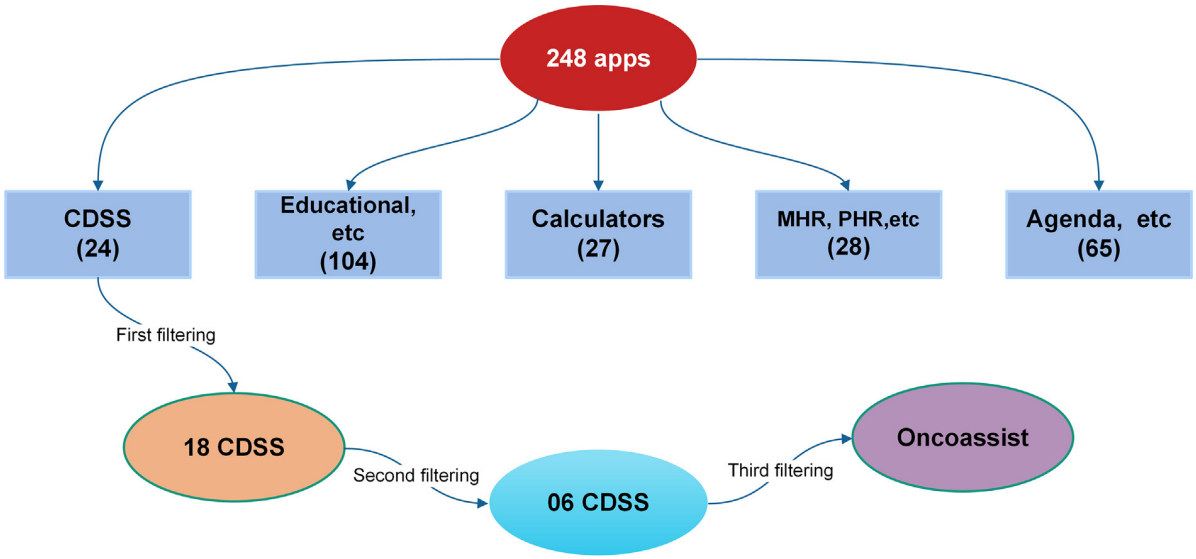

Fig. 1. Found Apps and the filtering process.

After this classification, 24 CDSS were identified, which were submitted to three filtering criteria.

First filtering: We considered the CDSS that underwent updates between the year 2018 and June 2021. This reduced our selection of CDSS to 18 systems.

Second filtering:The downloads of these CDSS should comply with a quantity criterion based on the year of publication:

1. year of publication 2018, should have a number of downloads equal to or greater than 5,000 (0 CDSS).
2. year of publication 2019, should have a number of downloads equal to or greater than 1000 (0 CDSS).
3. year of publication 2020, should have a number of downloads equal to or greater than 500 (1 CDSS).
4. year of publication 2021, should have a number of downloads equal to or greater than 100 (5 CDSSs).

After the second filtering, 06 CDSSs were selected from the 18 coming from the first filtering.

Third filtering: We present in Table 1 questions to be answered by the 6 CDSSs identified in the previous step, after being executed and tested for analysis and collection of characteristics. We classified them according to the number of positive answers. This step was not concerned with the specific functionalities of CDSS.

The CDSS with the most positive responses was **Oncoassist**[1], a CDSS of the oncological area, answering 8 "yes" to the 9 questions, which was chosen. The classification of CDSS is presented in Table 2. After choosing CDSS to be evaluated, we move on to applying our proposed measurement process model.

Table 1
Filtering Questions.

Id.	Question	Positive answers
1	Does the app provide information about updates in the app?	3
2	Can we suggest changes, improvements or additional features in the app itself?	2
3	Can we send email to support/developer team?	5
4	Does the app offer an official support by the Doctors?	6
5	Does it present terms of use, or privacy Policy, or disclaimer?	5
6	Does the app target at a specific disease or area?	4
7	Does it present the possibility to save or send results?	1
8	Does it present a final report with the suggestion/ decision?	5
9	Does it present a theoretical basis to support results?	6

Questions applied to the 06 CDSS and the number of positive answers.

Table 2
CDSSs classification after third filtering

Id.	CDSS	Number of positive answers
1	Oncoassist	8
2	Neuro Mind	7
3	TherapySelector	7
4	CFS	5
5	MDCalc	5
6	Calculate	5

CDSSs classification by number of positive responses.

3.3. Process model for evaluation of QiU

This study aims to present the validation of our process model for the evaluation of QiU characteristics for CDSSs. This model, exhibited in Fig. 2, consists of five main phases:

1. Identification of scenarios and context of use: this phase identifies the contexts of use of a system, in our case study, of a CDSS. Here, the scenarios where the main system functions are performed are pointed out;
2. Selection of measures, methods and metrics: in this phase, the selection of the characteristics and sub-characteristics of QiU to be evaluated, as well as the choice of the quality measure to be measured, takes place. Such measures are classified as **general**, when they are generally applicable and used in a wide range of situations, and **specialised**, when they are used for specific needs;
3. Quality in use measurement: in this phase, the mathematical function associated with the chosen quality measure is applied through tests with users and data collection of the scenario(s) selected by the evaluator;

[1] oncoassist.com

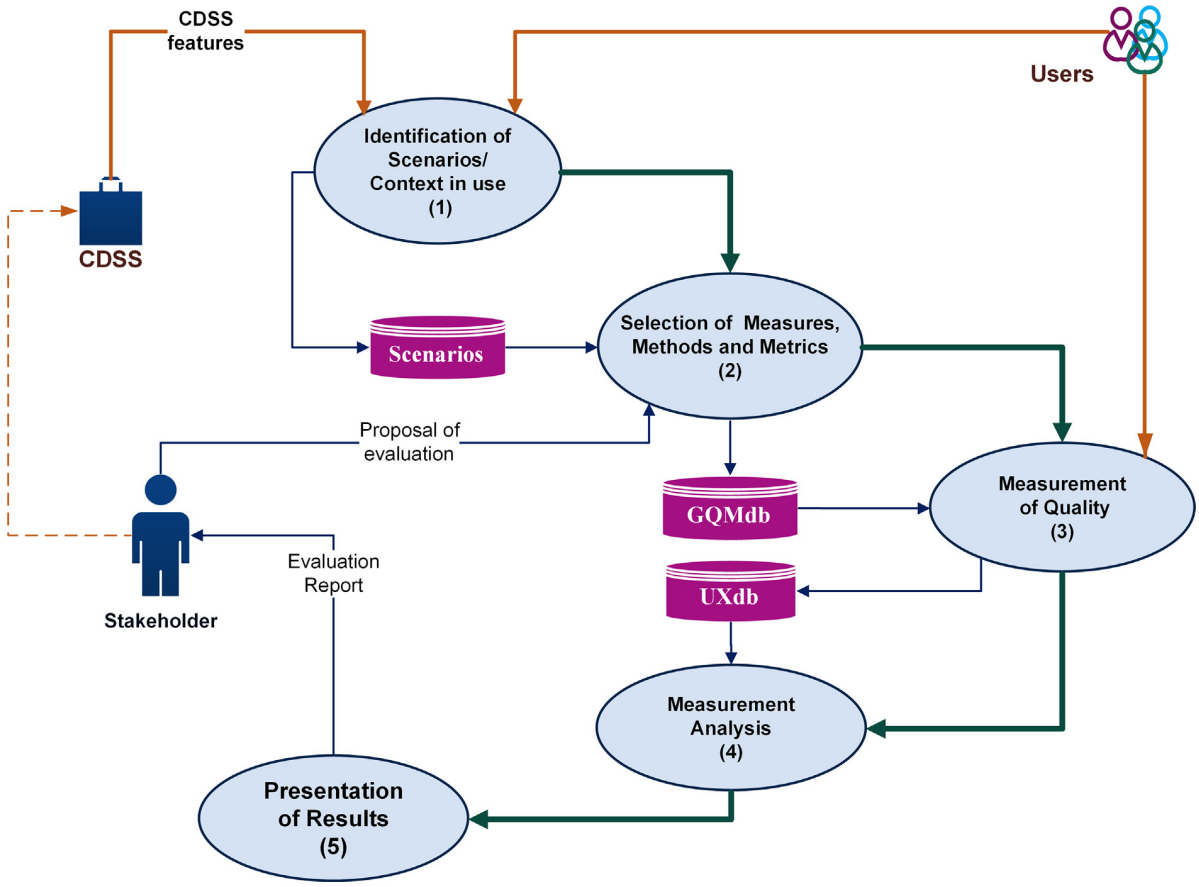

Fig. 2. Process model for QiU evaluation.

4. Measurement analysis: the results are analysed after measuring the characteristics. In this step, we have the answers to the questions (Q) of the GQM method as well as the interpretation of the measurement results of the mathematical functions;
5. Presentation of results: In this phase, the results obtained by the interpretation of the measured values are presented. It is suggested that the presentation is through a measurement report, where the strengths and weaknesses of the CDSS and graphic images are presented for better and easier visualisation.

These phases are described in our study (Souza-Pereira et al., 2021b) in detail, presenting the QiU characteristics and measures to assess them as well.

4. Proposed model application

The following subsections bring this application and comprise the four first phases of our model. The fifth phase is described in Section 5.

4.1. Identification of Scenarios and Contexts-in-Use on the CDSS

We have identified 21 functionalities in the CDSS Oncoassist and 18 outcomes. Table 3 presents the functionality-outcomes relation. The number of candidate scenarios is related to the functionalities of the CDSS and outcomes produced by them, as well as the number of users running the system during quality assessment (Souza-Pereira et al., 2021b).

We had no idea how many participants we would have in the validation process of the proposed model. So, there was no way to know in advance how many scenarios we would have since it was related to the number of users.

Furthermore, given the circumstances of measuring quality, it would be impractical to address all QiUs and all scenarios.

So we limited ourselves to the Satisfaction and Efficiency QiUs for our initial idea of applying the model. We also limited in addressing the general use of CDSS and in two specific functionalities, identified in Table 3, to compose our scenarios. These functionalities were, namely, the **creation of access credentials** (FO1) and the use of the functionality - **Adjuvant Tools ≫Breast Cancer** (FO14).

4.2. Selection of Measures, Metrics and Methods

Once we identified the scenarios and predefined the Satisfaction and Efficiency QiUs characteristics, the next step was to select the measures, metrics and methods to achieve the measurement of these characteristics.

Our proposed model adapted the GQM method to create a list of requests/responses related to CDSS functionalities to be evaluated, and to CDSS in general. This list is presented in Table 4. Based on this GQM list, we drew up an online questionnaire as a method to get information for the assessment of QiU of the CDSS.

Surveys or questionnaires are suggested by ISO/IEC 25022 (ISO/IEC, 2016) as a method to achieved the measurement and also applied in several studies (Kitchenham and Pfleeger, 2008; Lund, 2001; Lewis, 1995; Ives et al., 1983; Pavlič et al., 2018; Bauk et al., 2014; Nakai et al., 2016; Jung et al., 2004; Lew et al., 2012; Chen and Lee, 2009). For the creation of our questionnaire, we based on these studies and also in examples of enterprises focused

Table 3
Functionalities-Outcomes

ID	Functionality	Outcome
FO1	Account ≫ Sign up	User account created
FO2	Account ≫ Settings ≫Logout	Quit of the system
FO3	Account ≫ Settings ≫Change country	User labor country details
FO4	Account ≫ Settings ≫Edit Profile	User data changed
FO5	About ≫ Settings ≫About ≫Oncoassist	Information about the authors and about the app are shown
FO6	About ≫ Settings ≫Terms and conditions	End User License Agreement are shown
FO7	About ≫ Settings ≫Privacy Policy	Personal data collected and the privacy policy are shown
FO8	About ≫ Settings ≫Feedback	App improvement or report incidents are suggested
FO9	About ≫ Settings ≫Tell a friend	App link are sent
FO10	Favourite	Favourite a page/ functionality
FO11	Search	Search execution
FO12	Adjuvant Tools ≫ Colon Cancer	Estimated survival rates
FO13	Adjuvant Tools ≫ GIST	Estimated survival rates
FO14	Adjuvant Tools ≫ Breast cancer	Estimated survival rates
FO15	Adjuvant Tools ≫ Lung Cancer	Estimated survival rates
FO16	Formulas	Formulas results
FO17	Prognostic Scores	Calculated prognostic scores
FO18	Toxicity Grading (CTCAE)	Toxicity results
FO19	AJCC/TNM staging	AJCC
FO20	Drug Interaction Checker	Anti-cancer drug interaction
FO21	Advance Breast Cancer tools	Line of treatment in different clinical trials (BOLERO, PALOMA, Manarch, SOLAR, MONALEESA)

Functionalities and their outcomes of the Oncoassist CDSS.

on user experiences and research expertise (Nielsen, 1994; Nielsen, 2020).

A guideline explaining the target scenarios to be evaluated was distributed to the participants, instructing them to use the system in its entirety in order to get feedback on the system's overall usability. So we could subtract your satisfaction from your answers when using the app.

According to ISO/IEC 25022, international standard used in our study, each measure is related to a function and a method to apply to obtain this measure (ISO/IEC, 2016). Table 5 presents the measures and mathematical functions used in our measurement.

4.3. QiU Measurement and Analysis

The questionnaire was applied from October 16Th to November 2Nd. We distributed it to health professionals, students of medicine, bio-medicine, biomedical engineering, and computer engineering of software quality area. Only users who had not registered on the Oncoassist platform had participated in the quality measurement process, because one of the questions concerned the access credentials creation. We obtained 18 completed questionnaires, whose answers were grouped and measured using selected mathematical functions from the ISO/IEC 25022 standard, according to each characteristic and sub-characteristic evaluated.

Several studies were used as a support for the analysis and summary of the results, where it was possible to observe the use of weights for the characteristics (IBM-Corporation, 2014; Sulla-Torres et al., 2020), scores (IBM-Corporation, 2014; Chen and Lee, 2009), System Usability Scale (SUS) (Bangor et al., 2008; MeasuringU, 2021) and also statistical methods (Jung et al., 2004). We based on those studies to elaborate our range of values to comparisons, considering acceptable levels of quality (gold-standards) to decide if the CDSS is suitable for the intended purpose.

We convert the range of values to a scale of 0–100 and consider the results.

(1) excellent if above 90.0, (2) between 80.0 and 90.0 is very good, (3) between 65.0 and 80.0 is good, (4) between 60 and 65 is reasonable, (5) between 51 and 60 we consider bad and (6) below 51, awful.

We summarised the measures using the average (simple or weighted) or simple sum of the results of mathematical functions of the ISO/IEC 25022 standard, and compare the result of each measure with a score considered standard.

In the end, we were able to specify the value for each question and get the overall value for each QiU characteristic and sub-characteristic, and subsequently classified it as satisfactory or not in the context of the CDSS.

5. Presentation of results

The measurement and analysis of the found values were presented separately: (1) considering the specific functionality, and (2) the CDSS in general. In both were considered (a) each sub-characteristic measure, (b) each characteristic measure, and (c) overall measure.

Within this division, the Efficiency and Satisfaction QiUs were also separately considered. For Satisfaction characteristic assessment, 3 sub-characteristics were measured in the specific functionality part, and 4 sub-characteristics for the general part, as can be observed in Table 4. Finally, we presented the final classification of the CDSS. Two functionalities of the Oncoassist CDSS were chosen to be evaluated: Adjuvant tools ≫Breast Cancer and Creation of access credentials.

5.1. Adjuvant tools functionality results

The Adjuvant Tools feature ≫Breast Cancer was rated by participants as a good feature, with 77% positive responses about its usefulness in aiding decision making, a score of 78% about visual aids being important for a good understanding of the presented results, and 83% about the feature being easy to use. These are measures from the Trust (the first two) and (UX) Pleasure (the last) sub-characteristics. The majority of participants took less than 6 min to execute the functionality and to understand the results presented. Fig. 3 shows the time spent in this task.

In the context of this functionality, we present in Fig. 4 measurement of the efficiency and satisfaction characteristics through their sub-characteristics measures.

Two participants (11%) complained about the functionality related to the layout (badly arranged buttons, field size, not fixed menu bar), and not using the keyboard to fill in the data. Therefore, we can deduce the evaluated functionality was accepted and well-rated by the users, with an average measurement of 70,42%.

Table 4
The GQM list.

Section	Question	Charact.Subcharac
	How do you classify your knowledge on Oncology	N/A
Personal	What platform did you use to access the software?	N/A
	How long did it take you to run the functionality, from data entry to understanding the result?	Efficiency
	In your opinion, how do you classify the usefulness of this functionality to help decision making?	Satisfaction-Trust
	Visual information, graphics and charts presented on the results are important for my understanding.	Satisfaction-Trust
	The presentation of the icons is important for understanding the results.	Satisfaction-Trust
	The functionality is easy to use.	Satisfaction - (UX)-Pleasure
	Data entry is made easy.	Satisfaction - Usefulness
Adjuvant Tools Breast Cancer	The help option, next to each field to be filled out, is useful for using the functionality.	Satisfaction - Usefulness
	The results presentations are explanatory	Satisfaction - Usefulness
	You believe you would use this functionality frequently to help me in decision making.	Satisfaction - Usefulness
	Do you have any complaint or suggestion regarding the functionality Adjuvant Tools ≫Breast Cancer?	Satisfaction - Usefulness
	How long did it take you to create your account?	Efficiency
	How do you classify your experience in the account creation	Satisfaction - (UX) - Pleasure
	How do you rate your experience using the software?	Satisfaction - (UX) - Pleasure
	The software is a useful application in making support medical decisions (rate)	Satisfaction - Trust
	The software seems trustworthy to you	Satisfaction - Trust
	The layout of software is well done.	Satisfaction - Usefulness
	The results presentation are clear and objective.	Satisfaction - Usefulness
	References and the additional explanations to support the results are important and useful.	Satisfaction - Usefulness
	The software has important functionalities.	Satisfaction - Usefulness
	This software is easy to remember.	Satisfaction - (UX) - Pleasure
	You feel comfortable using the software.	Satisfaction - (UX) - Pleasure
General	This software is easy to use.	Satisfaction - (UX) - Pleasure
	Your overall satisfaction with using this software is high.	General
	You believe you would use this software frequently to help you in decision making.	Satisfaction - Usefulness
	Do you have any complaint or suggestion to be done about the software?	Satisfaction - Usefulness

The questions of GQM list and the QiU to be measured.

Table 5
Measures and mathematical functions.

Charac.	Subcharac.	Measure	ID	Function	Description
Efficiency	N/A	Task Time	Ey 1-G	$X = T$	T = Time taken to successfully complete a task
Satisfaction	Trust	User Trust	STr 1-G	$X = A$	A = Psychometric scale value from a trust questionnaire
	(UX) Pleasure	User pleasure	SPl 1-G	$X = A$	A = Psychometric scale value from a pleasure questionnaire
	Usefulness	Satisfaction with features	SUs 2-G	$X = \sum(A_i)$	A_i = Response to a question related to a specific feature
			SUs 4-G	$X = \frac{A}{B}$	A = number of users using a particular feature; B = number of users in an identified set of users of the system
			SUs 6-G	$X = \frac{A}{B}$	A = number of user complaints for a particular feature; B = Total number of user complaints about features
			SUs 3-G	$X = \frac{A}{B}$	A = number of users using a particular feature; B = number of potential users who could use the particular feature
			SUs 5-G	$X = \frac{A}{B}$	A = number of users complaining; B = number of users using the system
	General	Overall Satisfaction	SUs 1-G	$X = \sum(A_i)$	A_i = Response to a question

Functions used to measure the QiU characteristics of the CDSS (according to ISO/IEC, 2016).

5.2. CDSS general measurement results

Although it was considered a specific functionality to be measured, the results' presentation of the creation of access credentials was integrated into the group of general measures of the CDSS, to be according to our GQM list and our distribution of questions in the applied questionnaire.

Therefore, about the Creation of access credentials functionality, some users took over 3 min in this task, which reduced the efficiency rate in the task execution. The time spent to create the account to access the system is presented in Fig. 5 with a score of 51,2%.

Considering the whole system, the efficiency and satisfaction characteristics measures are shown in Fig. 6, with all satisfaction sub-characteristics scored above 65%.

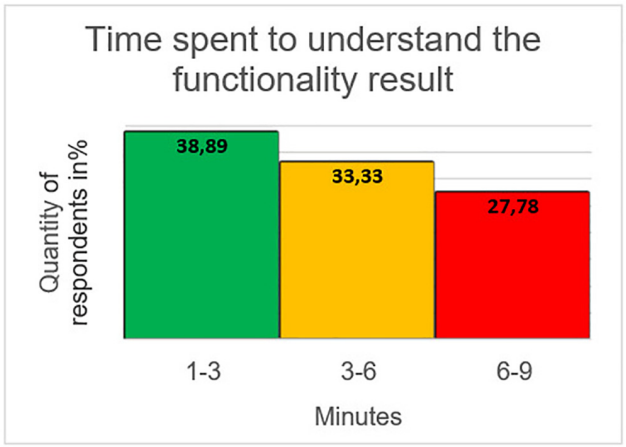

Fig. 3. Task time measurement.

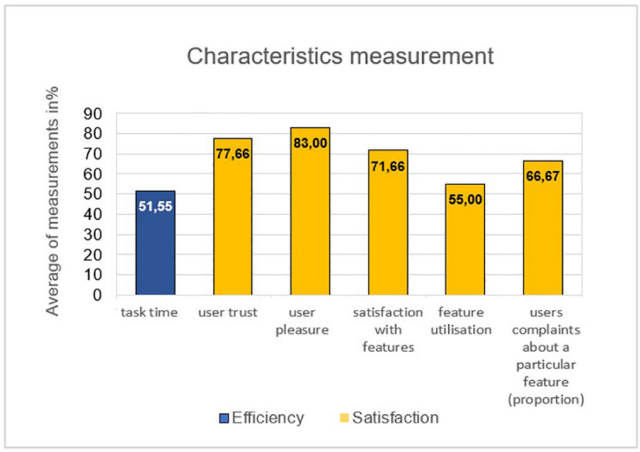

Fig. 4. Measures of efficiency and satisfaction characteristics of the functionality Adjuvant tools ≫Breast Cancer.

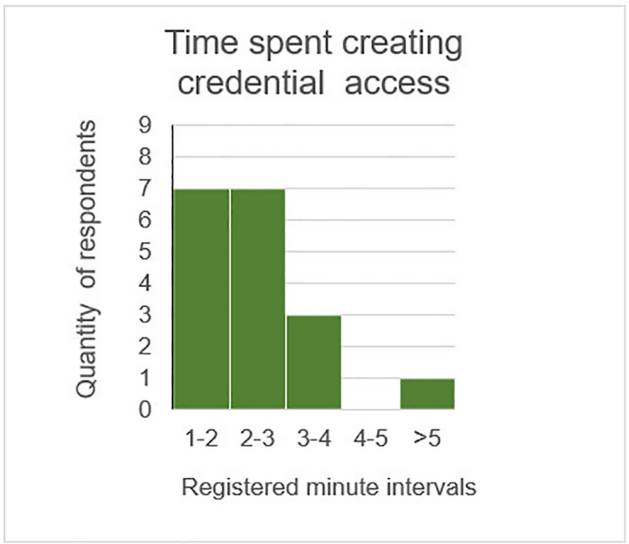

Fig. 5. Task time on credentials creation.

None of the participants ruled out the use of CDSS, with the majority (61,11%) claiming they could use it. This result is presented in Fig. 7, through the discretionary usage measure. The

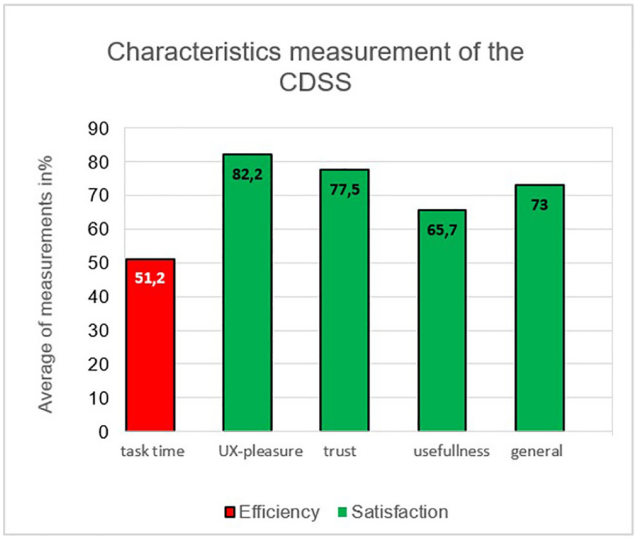

Fig. 6. Measures of efficiency and satisfaction characteristics of the CDSS.

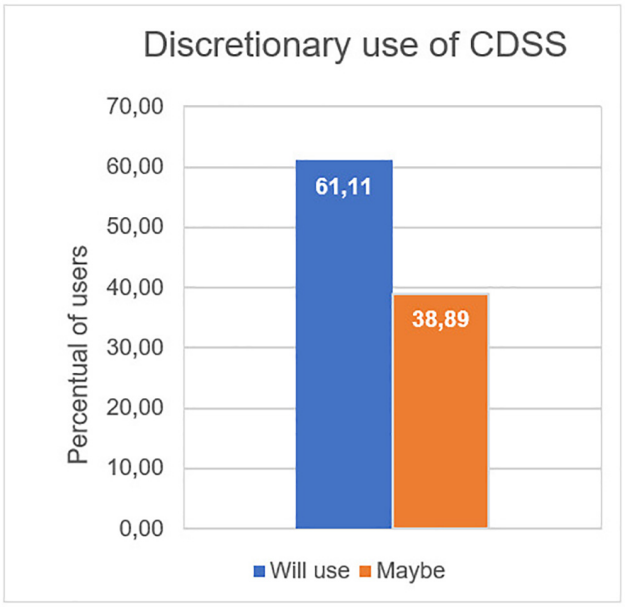

Fig. 7. CDDS usage.

average value of the CDSS general measurement, considering both functionalities Satisfaction and Efficiency, was 72,28%.

5.3. Final results

Measuring the QiU characteristics of the CDSS, considering the scenarios where two specific characteristics were assessed and also the system as a whole, we found a significant value of 71.35%, being the CDSS classified as GOOD according to our scale of values, and meeting the needs and expectations of users. The found values are presented in Fig. 8.

6. Conclusion, limitations and future work

In this paper, we validated a process model for the evaluation of quality in use of a CDSS, following the ISO/IEC 25022 and ISO/IEC 25010 standards. The proposed process model employed an

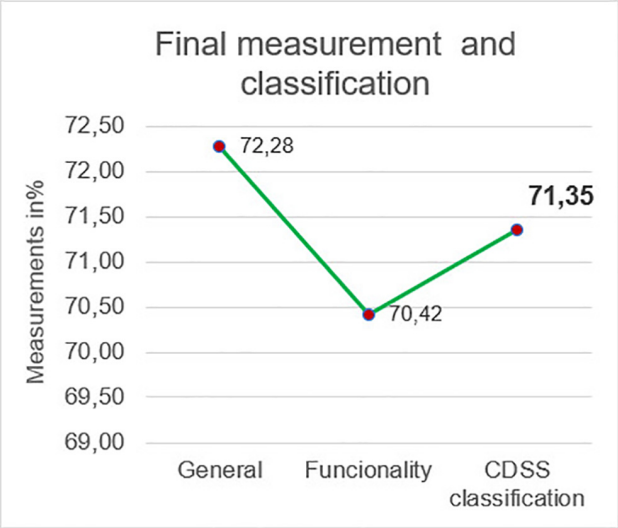

Fig. 8. Classification of the Oncoassist CDSS.

adapted GQM method to determine the measures and metrics to be used in evaluating the QiU characteristics, which in our case study were the Efficiency and Satisfaction characteristics.

We searched for a free-use CDSS to apply our process model and, firstly, we found 248 apps. These apps went through three judicious selection filters, and finally, we selected a CDSS: Oncoassist.

We applied this process model to Oncoassist CDSS through an online questionnaire created from our GQM list and distributed to healthcare professionals, software quality engineers, and students in the fields of medicine, biomedicine, and biomedical engineering.

The measurement of the QiU characteristics of the Oncoassist system reached 71,35% in our gold standard score range, clarifying that CDSS is a advantageous system and meets users' expectations.

We conclude that our proposed process model is useful to select and assess QiU characteristics, contributing to software quality professionals in the search for confirmation of the quality of a system in terms of meeting the needs and expectations of the user.

As a limitation of our work, we can mention the fact that the evaluation did not take place in a hospital environment, neither as a result of a daily CDSS usage by a medical team. These constraints, limited the adoption of additional efficiency measures than those employed in our online survey.

Besides this, we have not experienced the health professional's work routine with the CDSS, which could affect the measurement of QiU, based on the way the system had been used and perceived. Another detail that can be considered a deficiency is the number of responses obtained, which were 18 questionnaires answered. This can be considered a bias in the measurement and final classification of the system, even though studies have corroborated with the fact that, for measuring usability, few answers are enough to formalise reliable results. The standard ISO/IEC 25022 (ISO/IEC, 2016) brings that "...by controlling of the context of evaluation, experience has shown that reliable results can be obtained with a sample of only eight participants...", and Nielsen et al. (Nielsen and Landauer, 1993) demonstrated, using a mathematical model, that five is an ideal number of users to evaluate systems, as they found 75% of the usability problems. A lesson learned from this study was to realise that online surveys elucidate a low adherence by participants. Only eighteen people of the 76 invited professionals responded to our invitation. This corroborates the importance of the presence of the evaluator with the system user at the time of the evaluation.

As a challenge, we can mention the need to change this low engagement culture to this type of metric, perhaps through a more collaborative proposal, with the use of intelligent agents in the generation of questions from previous answers, in order to identify where one answer contradicts another, or even prevents it from happening.

For future work, we intend to apply the model to measure the QiU of a CDSS in use within a hospital environment, and use it also to measure all QiU characteristics, not just those that impact the end-user.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

Leonice Souza-Pereira would like to thank the IFTM - Instituto Federal do Triângulo Mineiro Campus Uberlândia Centro, Uberlândia, Brasil, for their support. This work is supported by FCT/MCTES through national funds and when applicable co-funded by FEDER—PT2020 partnership agreement under Grant UIDB/EEA/50008/2020. (Este trabalho é financiado pela FCT/MCTES através de fundos nacionais e quando aplicável co-financiado por fundos comunitários no âmbito do projeto UIDB/EEA/50008/2020); Co-financed by the Portugal 2020 Program (PT 2020), in the framework of the Regional Operational Program of the Center (CENTRO 2020) and the European Union through the Fundo Europeu de Desenvolvimento Regional (FEDER), under the Grant Centro-01-0247-FEDER-072632— "NomaVoy - Nomad Voyager".

References

Alshare, K.A., Alomari, M.K., Lane, P.L., Freeze, R.D., 2019. Development and determinants of end-user intention: usage of expert systems. J. Syst. Inf. Technol. 21, 166–185. https://doi.org/10.1108/JSIT-08-2018-0108. URL:https://www.emerald.com/insight/content/doi/ 10.1108/JSIT-08-2018-0108/full/html.

Alves, J.M., Savaris, A., Von Wangenheim, C.G., Von Wangenheim, A., 2016. Software quality evaluation of the laboratory information system used in the Santa Catarina state integrated telemedicine and telehealth system. Proc. - IEEE Symp. Comput. Med. Syst. 2016-Augus, 76–81. doi: 10.1109/CBMS.2016.51.

Arnicane, V., Borzovs, J., Nesaule-Erina, A., 2020. Do we really know how to measure software quality? CEUR Workshop Proc. 2620, 9–16.

Atoum, I., Bong, C.H., 2015. Measuring Software Quality in Use: State-of-the-Art and Research Challenges. ASQ.Software Quality Professional 17, 4–15. URL:http://arxiv.org/abs/1503.06934, arXiv:1503.06934.

Bangor, A., Kortum, P.T., Miller, J.T., 2008. An empirical evaluation of the system usability scale. Int. J. Hum. Comput. Interact. 24, 574–594. https://doi.org/10.1080/10447310802205776.

Basili, V.R., Weiss, D.M., 1984. A Methodology for Collecting Valid Software Engineering Data. IEEE Trans. Softw. Eng. SE-10, 728–738. https://doi.org/10.1109/TSE.1984.5010301.

Bauk, S., Šćepanović, S., Kopp, M., 2014. Estimating Students' Satisfaction with Web Based Learning System in Blended Learning Environment. Educ. Res. Int. 2014, 1–11. https://doi.org/10.1155/2014/731720.

Bukhari, Z., Yahaya, J., Deraman, A., 2018. A conceptual framework for metrics selection: SMeS. Int. J. Adv. Sci. Eng. Inf. Technol. 8, 2294–2300. https://doi.org/10.18517/ijaseit.8.6.6441.

Chen, J.K., Lee, Y.C., 2009. A new method to identify the category of the quality attribute. Total Qual. Manag. Bus. Excell. 20, 1139–1152. https://doi.org/10.1080/14783360902781832.

Djordjevic, N., 2017. Software quality standards. Vojnoteh. Glas. 65, 102–124. https://doi.org/10.5937/vojtehg65-10668.

Evans, I., Porter, C., Micallef, M., Harty, J., 2020. Test tools: an illusion of usability? In: 2020 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW). IEEE, Porto, PT, pp. 392–397. https://doi.org/10.1109/ICSTW50294.2020.00070.

Hassan, H.M., Galal-Edeen, G.H., 2017. From usability to user experience. In: 2017 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS). IEEE, Okinawa, JP, pp. 216–222. https://doi.org/10.1109/ICIIBMS.2017.8279761.

IBM-Corporation, 2014. Questionnaire template scoring formulas - ibm documentation. URL:https://www.ibm.com/docs/en/opw/7.3.0?topic=assessments-questionnaire-template-scoring-formulas.
ISO/IEC, ISO 25000 STANDARDS. URL:https://iso25000.com/index.php/en/iso-25000-standards.
ISO/IEC, 2011. Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQUARE) – System and software quality models. URL:https://www.iso.org/standard/35733.html.
ISO/IEC, 2016. Systems and software engineering – Systems and software quality requirements and evaluation (SQUARE) – Measurement of quality in use. URL: https://www.iso.org/standard/35746.html.
ISO/IEC, 2019. ISO 9241–210:2019: Ergonomics of human-system interaction - Human-centred design for interactive systems. URL:https://www.iso.org/standard/77520.html.
Ives, B., Olson, M.H., Baroudi, J.J., 1983. The measurement of user information satisfaction. Commun. ACM 26, 785–793. https://doi.org/10.1145/358413.358430.
Jung, H.W., Kim, S.G., Chung, C.S., 2004. Measuring software product quality: a survey of iso/iec 9126. IEEE Software 21, 88–92. https://doi.org/10.1109/MS.2004.1331309.
Khalifa, M., 2014. Clinical decision support: Strategies for success, in: Procedia Comput. Sci., Elsevier, Halifax, CA. pp. 422–427. URL:http://www.sciencedirect.com/science/article/ pii/S187705091401028X, doi: 10.1016/j.procs.2014.08.063.
Kitchenham, B.A., Pfleeger, S.L., 2008. Personal opinion surveys, in: Guid. to Adv. Empir. Softw. Eng.. Springer, London, London, UK, pp. 63–92. URL:https://link.springer.com/chapter/10.1007/978-1-84800-044-5_3, doi: 10.1007/978-1-84800-044-5_3.
Lew, P., Abbasi, M.Q., Rafique, I., Wang, X., Olsina, L., 2012. Using web quality models and questionnaires for web applications understanding and evaluation. In: 2012 Eighth International Conference on the Quality of Information and Communications Technology. IEEE, Lisbon, PT, pp. 20–29. https://doi.org/10.1109/QUATIC.2012.68.
Lewis, J.R., 1995. IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. Int. J. Human- Comput. Interact. 7, 57–78.
Lund, A.M., 2001. Measuring usability with the USE questionnaire. Usability interface 8, 3–6.
MeasuringU, 2021. Questionnaires - measuringu. URL:https://measuringu.com/category/questionnaires/.
Nakai, H., Tsuda, N., Honda, K., Washizaki, H., Fukazawa, Y., 2016. Initial framework for software quality evaluation based on iso/iec 25022 and iso/iec 25023. In: 2016 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, Vienna, Au, pp. 410–411. https://doi.org/10.1109/QRS-C.2016.66.
Nielsen, J., 1994. Heuristic Evaluation: How-To conduct. URL:https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/.
Nielsen, J., 2020. usability heuristics for user interface design. 10. URL:https://www.nngroup.com/articles/ten-usability-heuristics/.
Nielsen, J., Landauer, T.K., 1993. A mathematical model of the finding of usability problems, in: Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA. p. 206–213. doi: 10.1145/169059.169166.
P. Miguel, J., Mauricio, D., Rodríguez, G., 2014. A Review of Software Quality Models for the Evaluation of Software Products. Int. J. Softw. Eng. Appl. 5, 31–53. doi: 10.5121/ijsea.2014.5603, arXiv:1412.2977.
Pavlič, L., Okorn, M., Heričko, M., 2018. The use of the software metrics in practice. CEUR Workshop Proc. 2217, 27–30.
Pinciroli, F., 2016. Improving Software Applications Quality by Considering the Contribution Relationship among Quality Attributes. Procedia Comput. Sci. 83, 970–975. https://doi.org/10.1016/j.procs.2016.04.194. URL:https://doi.org/10.1016/j.procs.2016.04.194.
Sacha, K., 2006. Evaluation of expected software quality: A customer's viewpoint. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 3922 LNCS, 170–183. doi: 10.1007/11693017_14.
van Solingen, R., Berghout, E., 1998. The Goal/Question/Metric Method: a practical guide for quality improvement of software development, volume 7. McGraw-Hill Education. URL:https://www.academia.edu/26126148/The_Goal_Question_Metric_method_a_practical_guide_for_quality_improvement_of_software_development.
Sommerville, I., 2016. Software engineering. Addison Wesley, New York, USA.
Souza-Pereira, Leonice, Pombo, Nuno, Ouhbi, Sofia, 2021a. Quality-in-use Characteristics for Clinical Decision Support System Assessment. Comput. Methods Programs Biomed. 207. https://doi.org/10.1016/j.cmpb.2021.106169.
Souza-Pereira, Leonice, Pombo, Nuno, Ouhbi, Sofia, 2021b. A process model for quality in use evaluation of clinical decision support systems. J. Biomed. Inf. 123. https://doi.org/10.1016/j.jbi.2021.103917.
Souza-Pereira, L., Pombo, N., Ouhbi, S., Felizardo, V., Garcia, N., 2020. Clinical decision support systems for chronic diseases: A Systematic literature review. Comput. Methods Programs Biomed. 195,. https://doi.org/10.1016/j.cmpb.2020.105565 105565.
Sulla-Torres, J., Gutierrez-Quintanilla, A., Pinto-Rodriguez, H., Gómez-Campos, R., Cossio-Bolaños, M., 2020. Quality in use of an android-based mobile application for calculation of bone mineral density with the standard ISO/IEC 25022. Int. J. Adv. Comput. Sci. Appl. 11, 158–163. https://doi.org/10.14569/IJACSA.2020.0110821.
Tsuda, N., Washizaki, H., Honda, K., Nakai, H., Fukazawa, Y., Azuma, M., Komiyama, T., Nakano, T., Suzuki, H., Morita, S., Kojima, K., Hando, A., 2019. Wsqf: Comprehensive software quality evaluation framework and benchmark based on square. In: 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP). IEEE, Montreal, CA, pp. 312–321. https://doi.org/10.1109/ICSE-SEIP.2019.00045.
Zikos, D., 2017. A Framework to Design Successful Clinical Decision Support Systems, in: Proc. 10th Int. Conf. PErvasive Technol. Relat. to Assist. Environ., Association for Computing Machinery, New York, NY, USA. pp. 185–188. URL: https://doi.org/10.1145/3056540.3064960, doi: 10.1145/3056540.3064960.

Chapter 6

Conclusion and future work

This chapter presents the main conclusions drawn from the research work described in this thesis and also discusses some research topics that may be addressed, as a continuation or a complement of the work developed in this doctoral program.

6.1 Conclusion

This thesis is focused on software quality-in-use measurement and describes the research work developed with the purpose of presenting a new process model for quality in use evaluation, using a clinical decision support system as a case study.

The research method comprised qualitative and quantitative phases, with three articles being produced in the qualitative phase, including a SLR, and another in the quantitative phase, with the results of measuring the quality-in-use characteristics, obtained by applying the evaluation process model.
Plus, the research work was conducted following standards and guidelines, namely: ISO/IEC 25022, ISO/IEC 25010 and Prisma statement.

A CDSS, as all other software, must be produced under requirements and also follow the development rules and standards. Therefore, the practice of software engineering comes to meet these exigencies.

As a state-of-the art, our first concern and study, a systematic literature review was proposed, which presented a set of studies focusing on CDSS, their approaches to software engineering techniques, proposal, evaluation methods and metrics.
In that study, it was observed the CDSS focus on diagnosis was the most frequent proposal. We suggest a combination of prevention, screening, and diagnosis approaches in the same CDSS, in order to share the same data collected from the patient.

We also observed a scarce representation of software engineering structures in the studies selected for the SLR, as well as identifying that the most common forms of system validation were UX, architectural implementation performance, and software testing criteria, whereas accuracy, specificity, and sensitivity were the most presented metrics. In this context, we suggest the broader use of questionnaires to assess user satisfaction and integrate user feedback into the CDSS development life-cycle.

During the verification and validation of the proposed solution to mitigate the above mentioned gaps, we identified an additional one, namely, the non-acceptance or the reduce use of CDSS by the clinical staff. For that reason, we focused on verifying whether the quality of the CDSS would be a cause for this limitation. In line with this, our research was steered in order to accommodate software quality principles.

Thus, our second study was focused on identifying important quality characteristics for a CDSS, and how these qualities are measured. The ISO/IEC 25010 standard was used to guide us in this identification.

In the selected studies, we found the qualities related to the development of the software product, internal and external, were the most use and evaluated. The measurement of quality-in-use characteristics was despicable, by comparison.

Conversely, we believed that not using CDSS was more associated with the user experience, since, hypothetically, the quality of development was being evaluated and, even so, the system was little used.

Therefore,our objective was to choose QiU characteristics to be evaluated in order to improve the use of CDSSs. As result, we elected satisfaction and efficiency as the most important characteristics to be evaluated. Both features, added to effectiveness, reflect the system's usability, but the first two are more linked to the end-user, who uses CDSS or not.

The next step was to develop a model to collect the QiU characteristics of a CDSS, measure them, and identify whether the CDSS met user expectations.
The classification of the CDSS, within the scale of great to terrible, would elucidate the possibility of the CDSS being used or not, and, in the same way, the measures would provide parameters for identifying problems in the system's development.

Our third study brought a process model to assess QiU characteristics and applied to a CDSS. The model, developed according to ISO/IEC 25010 and ISO/IEC 25022 SQuaRE Standard, was based on GQM method to select the main functionalities of the system, as well the key characteristics, metrics, and measures to be evaluated in each functionality. An illustrative example was used to highlight the QiU Satisfaction and Efficiency measurement from a list of GQM, as examples of measurement characteristics.

The idea was to show that measurement of QiU characteristics could contribute to solving the problem of health professionals' reluctance to use CDSSs through the identification of causes and the possibility of correcting them.

For the conclusion to our study, it was necessary to validate our model.

Initially, the idea was to apply our measurement model to a CDSS that was being used in a hospital and realise its quality in use and utilisation rate.
However, this became impractical because of the COVID-22 pandemic. We set out to measure the QiU of a free system, searched on the Google Store, and selected among many others.

Three judicious selection filters were applied to 248 found apps and, in the end, we elected the ONCOASSIST as the CDSS to have the Satisfaction and Efficiency QiU characteristics evaluated.

We applied this process model to Oncoassist CDSS through an online questionnaire created from our GQM list and distributed to healthcare professionals, software quality engineers, and students in the fields of medicine, biomedicine, and biomedical engineering.

The measurement of the QiU characteristics of the Oncoassist system reached 71,35% in our gold standard score range, clarifying that CDSS is a advantageous system and meets users' expectations.

In this context, we concluded that our proposed process model was suitable to select and assess QiU characteristics of the CDSS. In addition, as far of our knowledge this was the first process model capable to select system functionalities, to identify the most adequate QiU characteristics for such functionalities, and to measure the system quality.

Measuring the QiU characteristics may help to solve the problem of health professionals' reluctance to use these systems, through the identification of the causes and the possibility of fixing them. These measurements contribute to software quality professionals in the search for confirmation of the quality of a system to meet the needs and expectations of the users.

This work makes a significant contribution for researchers and practitioners, in the area of software quality, through the study of quality characteristics, especially the quality in use. In addition, a guideline for the collection and measurement of these characteristics was built through the proposal of the process model. The proposed model can be applied to any system for collecting and measuring any quality characteristic giving practitioners the ability to tune and adapt the system if necessary.

6.2 Future work

For future work, we intend to expand the range of QiU characteristics to be measured, also measuring those that impact other stakeholders, such as managers, decision makers, and system developers.
In addition, a cross-reference analysis of the characteristics and sub-characteristics of quality in use and software product quality could also be worked on, especially if the measurement

is directed to the system development team. This analysis will allow us to verify the dependencies of these characteristics and their joint measures and their impact on both the development and usability of the CDSS.