

LEHSA - Laboratório de Estudos em Hidráulica e Saneamento Ambiental

Lucas Cabral Soares¹, Lucas Hemétrio Teixeira², Lucca Oliveira Vasconcelos de Faria³,
Maria Eduarda Amaral Muniz⁴, Vitor Stahlberg⁵

¹Instituto de Ciências Exatas e Informática
Pontifícia Universidade de Minas Gerais (PUC Minas)
Belo Horizonte – MG – Brasil

{lcsoares¹, lucas.hemetrio², lovfaria³}@sga.pucminas.br

{maria.amaral⁴, vitor.lagares⁵}@sga.pucminas.br

Resumo. *O gerenciamento eficiente dos recursos em um laboratório acadêmico é essencial para garantir que alunos de todos os níveis de ensino possam utilizá-los de forma eficaz. Atualmente, métodos manuais como papel e caneta, apresentam limitações significativas nesse processo. Este trabalho relata a criação de um sistema digital de gestão para o Laboratório de Estudos em Hidráulica e Saneamento Ambiental (LEHSA) no Instituto Federal de Sergipe (IFS), visando aprimorar a administração do laboratório diante dos desafios de crescimento e complexidade. O objetivo geral é desenvolver uma plataforma online para administrar os equipamentos e materiais do LEHSA, com o intuito de modernizar os métodos de gestão, aumentando a organização e eficiência do laboratório. Utilizando princípios da metodologia ágil e Engenharia de Software, este projeto não apenas supre as necessidades práticas do laboratório, mas também oferece valiosas oportunidades de aprendizado e aplicação de conhecimentos para os alunos envolvidos.*

1. Introdução

No mundo contemporâneo, a presença constante da tecnologia tem gerado uma demanda por adaptações nos métodos de gestão, sobretudo em ambientes especializados como os laboratórios. Este cenário motivou a reflexão sobre a necessidade de desenvolver sistemas digitais para o gerenciamento eficaz desses espaços, como é o caso do Laboratório de Estudos em Hidráulica e Saneamento Ambiental (LEHSA) no Instituto Federal de Sergipe (IFS). A dependência dos métodos manuais entre os envolvidos, revelou-se inadequada diante da crescente complexidade e demandas do laboratório. Esta transição para sistemas digitais não é apenas desejável, mas também uma necessidade essencial para garantir a sustentabilidade e eficiência das operações laboratoriais.

Atualmente, a sociedade humana como um todo vive em um momento no qual a tecnologia se vê sempre presente. A melhoria dos computadores trouxe uma realidade onde processos burocráticos, dentro de organizações de pequeno a grande porte e até em nível pessoal, fossem digitalizados. Em laboratórios, o controle de material, equipamentos, insumos e agendamentos é a base para a utilização eficaz do espaço. A tecnologia

possibilitou que a troca de informações necessárias nesses processos fosse feita com grandes grupos de pessoas, de maneira mais rápida e eficiente do que métodos mais tradicionais como papel e "boca a boca". Dito isso, o presente trabalho trata da construção de um sistema digital para o gerenciamento de um laboratório.

Nesse contexto, o problema abordado no projeto é o gerenciamento ineficiente do Laboratório de Estudos em Hidráulica e Saneamento Ambiental (LEHSA) pertencente ao Instituto Federal de Sergipe (IFS). O LEHSA possui sistemas pouco robustos para sua administração, dependendo de planilhas de Excel, papéis e comunicação direta entre pessoas. Isso se mostrou inviável com o crescimento das dependências e responsabilidades do laboratório.

O presente artigo está organizado da seguinte forma: na subseção 1.1 é demonstrado os objetivos partindo do geral para os específicos na subseção 1.2. Na subseção 1.3 justifica-se o projeto do sistema LEHSA. Na seção 2 é abordado o referencial teórico, desde os conceitos até a contemplação dos principais trabalhos relacionados ao tema do projeto. Por fim, na seção 3 é abordado a forma como foi conduzido o projeto detalhando todas as sprints.

1.1. Objetivo Geral

O objetivo geral do trabalho é desenvolver um sistema web capaz de gerenciar equipamentos e materiais do LEHSA, visando realizar migração de métodos de gerência analógicos para digitais para aprimorar a organização e eficiência do laboratório.

1.2. Objetivos Específicos

1. Coletar e analisar os requisitos do sistema
2. Prototipar e desenvolver os requisitos definidos utilizando princípios da metodologia ágil
3. Validar o software finalizado com o Product Owner
4. Implantar o sistema validado

1.3. Justificativa

O projeto aqui descrito proporciona uma ferramenta que melhora o âmbito administrativo de uma instituição federal, assim ajudando a educação pública. Ademais, o desenvolvimento do trabalho traz uma oportunidade para os membros envolvidos aplicarem e melhorarem seus conhecimentos, criando também novas competências.

2. Referencial Teórico

Nesta seção, são apresentados os conceitos e características dos seguintes assuntos: o que é extensão universitária; breve descrição do parceiro; projeto do software; metodologia ágil; modelagem de banco de dados; trabalhos relacionados.

2.1. Extensão Universitária

A Extensão Universitária, ao longo do tempo, tem experimentado significativas transformações conceituais, sendo atualmente reconhecida como uma atividade intrínseca à missão da universidade, caracterizando-se, portanto, como obrigatória. Além dessa obrigação institucional, Colamarco (2023) diz que a extensão apresenta características

essenciais, destacando-se como um processo educativo, cultural, político e tecnológico. Ainda de acordo com Colamarco (2023), a extensão trata-se de uma interação dinâmica entre a instituição acadêmica e outros setores externos à comunidade acadêmica, fundamentada na troca de conhecimentos e práticas.

Neste contexto, a Extensão Universitária pode ser compreendida como um mecanismo de comunicação recíproca, onde a universidade contribui com sua expertise para a comunidade, enquanto esta última enriquece o processo de aprendizagem institucional. A participação ativa dos alunos desempenha um papel fundamental nesse processo, envolvendo-os na concepção, implementação, avaliação e discussão das ações extensionistas.

2.2. Parceiro

O Laboratório de Estudos em Hidráulica e Saneamento Ambiental (LEHSA) é um componente essencial do curso de Engenharia Civil e do curso técnico em Edificações do Instituto Federal de Sergipe (IFS), Campus Estância. Além disso, estabelece colaborações com programas de pós-graduação dedicados ao estudo do meio ambiente e engenharia.

O laboratório visa atender as demandas acadêmicas dos cursos mencionados, e também oferecer serviços especializados em saneamento ambiental para a comunidade regional, impulsionando atividades de pesquisa e extensão. Em suas operações, o LEHSA concentra-se principalmente em hidráulica, saneamento ambiental e energias limpas e renováveis, adotando tecnologias emergentes como facilitadoras da inovação.

A contribuição do LEHSA para a formação dos estudantes é notável, proporcionando experiências práticas, conduzindo projetos de pesquisa e extensão e promovendo uma abordagem multidisciplinar do conhecimento. De acordo com Rafael Oliva (2022), este laboratório é reconhecido por sua infraestrutura completa, sua natureza multidisciplinar e sua produção científica e inovadora, atraindo estudantes, professores e pesquisadores em busca de recursos de qualidade.

Em termos práticos, o LEHSA oferece uma ampla gama de benefícios para estudantes, professores, pesquisadores e a comunidade em geral. Seu espaço físico abriga bancadas de análises e áreas de experimentação, enquanto sua capacidade técnica permite a realização de uma variedade de testes e medições relacionados à qualidade da água e tratamento de efluentes.

O LEHSA se destaca como um recurso valioso para a pesquisa e prática no campo da engenharia civil e ambiental, refletindo seu compromisso em promover o avanço científico e tecnológico em sua área de atuação.

2.3. Projeto do Software

No projeto do sistema não se entra em detalhes de implementação de cada unidade de código [Valente 2022], ao invés disso, leva-se em conta de que forma o sistema vai funcionar para atender aos requisitos colhidos e consolidados na fase de levantamento de requisitos e análise de requisitos. Diferente da fase de análise de requisitos, o projeto considera os recursos tecnológicos disponíveis, adicionando-se as chamadas “restrições tecnológicas” [Bezerra 2007].

A divisão do projeto de software, proposta por Royce (1970), compreende duas

atividades principais: o projeto da arquitetura e o projeto detalhado. No projeto da arquitetura, ocorre a distribuição das classes de objetos em subsistemas e componentes, levando em conta também a distribuição física desses componentes em hardware. Nessa etapa, são comumente utilizados diagramas que modelam a arquitetura física do sistema, destacando os relacionamentos entre os componentes de software e hardware, conhecidos como diagramas de implementação [IBM 2024].

Por outro lado, no projeto detalhado, são definidas as funcionalidades de cada módulo, a interface com o usuário e o banco de dados [Bezerra 2007]. Para essa etapa, são utilizados diversos diagramas UML, como de classes, de casos de uso, de interação, de estado e de atividade, que ajudam a representar de forma clara e detalhada o funcionamento do sistema.

2.4. Metodologia Ágil

A metodologia ágil tem como foco a entrega de aplicações funcionais criadas mediante iterações rápidas. A ideia é entregar partes de um software de forma a aumentar a satisfação do cliente. Isso é feito seguindo abordagens adaptáveis, bem como o trabalho em equipe [Red Hat 2022].

State of Agile (2020) realizou um levantamento de empresas de diferentes setores que utilizam as práticas ágeis. a maior parte das empresas são do setor de desenvolvimento de software (37%) e em segundo lugar, o setor de TI (26%). Além disso os cinco principais motivos que levam as empresas a adotar as metodologias ágeis são: acelera a entrega do software em 71%, eleva a capacidade de gerenciamento de prioridade em 63%, aumenta a produtividade em 51%, melhora o alinhamento entre times de tecnologia e negócios em 47% e eleva a qualidade de software em 47%. Dessa forma, a metodologia ágil oferece uma abordagem eficaz e adaptável para o desenvolvimento de software, permitindo que as empresas alcancem melhores resultados em termos de qualidade, velocidade e satisfação do cliente.

O presente trabalho interdisciplinar LEHSA utiliza a metodologia Scrum que segundo State of Agile (2020) , possui 58% de aplicabilidade no mercado. Tal metodologia tem como essência uma equipe auto-organizada que entrega valor ao cliente em um período denominado Sprint. Além disso o Scrum utiliza de artefatos cuja função é fornecer informações de planejamento e tarefas para as equipes. Existem dois tipos de artefatos: *product backlog* e *sprint backlog*. O primeiro é a lista de tarefas da equipe, que pode ser revista de forma constante, o segundo, é a lista de itens a serem concluídas pela equipe no ciclo atual do Sprint. Essa lista é definida por decisão de quais itens do product backlog trabalhar antes da Sprint [Amazon Web Services 2024].

2.5. Modelagem de Banco de Dados

A modelagem de dados é um processo fundamental no desenvolvimento de software, onde se busca compreender e representar os dados de um determinado contexto de forma estruturada. Segundo Machado (2020), esse processo envolve a mineração das informações pertinentes ao problema em questão, organizando-as em um modelo lógico de dados. Uma das características essenciais da modelagem de dados é sua capacidade de fornecer diferentes níveis de abstração, ocultando detalhes sobre o armazenamento físico dos dados e concentrando-se na representação lógica.

Dentro desse contexto, o Modelo Entidade-Relacionamento (MER) se destaca como uma das principais abordagens para modelagem de dados em sistemas de banco de dados relacionais. Proposto inicialmente por Chen (1976), o MER tem como base a teoria relacional de Codd (1985), representando o mundo real por meio de entidades e seus relacionamentos. O MER é conceitual, fornecendo uma visão abstrata dos objetos de dados, sendo seu principal instrumento o diagrama de Entidade-Relacionamento.

Ao longo dos anos, diversos estudiosos contribuíram para expandir e aprimorar a modelagem de dados, incorporando mecanismos de abstração como classificação, generalização e agregação. Essas abstrações auxiliam os analistas na compreensão e modelagem de problemas complexos, permitindo a representação de entidades, relacionamentos e atributos de forma clara e concisa.

Além de seu papel fundamental no projeto de banco de dados, o MER também pode ser aplicado em outras áreas, como a modelagem de processos de negócios e o desenvolvimento de sistemas orientados a objetos. No entanto, para produzir modelos eficazes, é essencial um profundo entendimento dos conceitos relacionados à realidade em estudo, e se torna fundamental realizar verificações e validações rigorosas, envolvendo tanto os usuários finais quanto profissionais técnicos, a fim de garantir a precisão e a qualidade do modelo desenvolvido.

2.6. Trabalhos relacionados

Alguns trabalhos de construção de sistema de agendamento já foram feitos. A monografia de Kieras (2019) apresenta o desenvolvimento de uma solução para gestão de agendamentos de serviços prestados por profissionais diversos. A solução também possui aplicação da arquitetura REST de forma a permitir a comunicação entre aplicações, no entanto, esse trabalho é mais generalista não sendo direcionada a nenhuma entidade específica, além disso é um sistema multiplataforma não tendo foco na web.

Uma monografia que contém direcionamento para um setor específico é a de Nakagawa (2014), no qual é feito um sistema gerenciador de agendamento aplicado a um salão de beleza, ou seja, qualquer empresa de salão de beleza pode se cadastrar no sistema. Este trabalho também utiliza de diagramas UML como o diagrama de caso de uso e de classe para facilitar o entendimento dos requisitos e da implementação. No entanto, foi usado o Extreme Programming (XP) como metodologia ágil, que não é planejado de uma maneira tão flexível, com sprints e backlog priorizado como o Scrum. Ao invés disso, é dado o foco em práticas de engenharia específicas, tornando a equipe menos flexível.

Assim como em Kieras (2019), o site SimplyBook apresenta uma solução Web mais abrangente se apresentando como um “Sistema de Reservas Online para Todas as indústrias baseadas em serviços”. Outra semelhança é ser multiplataforma. No sistema, o cliente pode “criar” um site ou, se já tiver um, incorporar um componente de interface de usuário na aplicação. O site é uma solução meramente comercial contrastando com os demais que são soluções que nascem no campo acadêmico.

O presente trabalho, diferente dos trabalhos supracitados tem o objetivo de projetar o sistema com base nas demandas específicas do setor em que será implementado, garantindo integração com as operações existentes.

3. Metodologia

Com base na metodologia Scrum que foi a escolhida para a realização do projeto, foram realizadas várias iterações conhecidas como "Sprints", cada uma com duração média de três semanas [Pereira et al. 2007]. Cada sprint foi planejada e executada com base em um conjunto de objetivos definidos pelo Product Owner (coordenador do laboratório) em colaboração com a equipe de desenvolvimento. O foco principal durante as Sprints foi na entrega de incrementos de funcionalidade, priorizando as necessidades do cliente e respondendo rapidamente aos feedbacks recebidos por Whatsapp e através de uma reunião por Sprint, em média. O número de Sprints são cinco e nas subseções a seguir serão detalhadas as atividades realizadas em cada uma, destacando o que foi feito e os desafios enfrentados ao longo do processo.

3.1. Sprint 1

A sprint 1, como abertura do projeto, focou na definição de um parceiro, bem como a criação do grupo participante do projeto em si. Após esse processo, foram feitas as primeiras reuniões com o cliente e discussões via Whatsapp e Google Meet para estabelecer os objetivos e requisitos iniciais do sistema a ser desenvolvido. Além disso, uma apresentação de slides foi preparada e exibida durante uma das aulas de Trabalho Interdisciplinar.

O projeto definido foi a criação de um sistema web visando o gerenciamento do laboratório LESH, o que inclui gerenciamento de itens, usuarios, agendamentos para uso do laboratório e aluguel de itens do laboratório. As tecnologias planejadas para a implantação do sistema seriam: Java Spring Boot e PostgreSQL SGBD para o back-end e React para o front-end.

Feita essa primeira fase de levantamento de requisitos primários, foi feito o planejamento para próxima Sprint, dividindo as responsabilidades e escolhendo os primeiros requisitos a serem implementados.

3.2. Sprint 2

A sprint 2 teve início com uma reunião com o cliente para se definir concretamente as regras de negócio com objetivo de fornecer base para a construção do serviço no *back-end*, além de passar uma ideia de como seria o *front-end*. Ademais, foram desenvolvidos diagramas de Entidade Relacionamento e de Casos de Uso que serviriam como base para a implementação do software.

Na Sprint 2 foram implementados os casos de uso referentes a Item e Usuários: os itens são todos os aparelhos, insumos e produtos pertencentes ao LESH e os usuários compreendem os alunos, técnicos responsáveis por orientar os alunos e professores responsáveis, que seriam administradores do sistema. O foco inicial foi na implementação dos casos de uso de item e usuário, juntamente com o desenvolvimento de funcionalidades como gerenciamento dessas entidades, login e segurança do sistema.

Houve dificuldades no *back-end* ao integrar com o *front-end* no módulo de segurança, contudo, superou-se isso com o auxílio do *ChatGPT*, que identificou o erro exato no código.

3.3. Sprint 3

A sprint 3 teve início com a modelagem de classe via diagrama de classe, definindo as estratégias de associações, heranças e implementações de interfaces, com o objetivo de melhor representar a estrutura do código no *back-end*, facilitando a implementação. No *front-end* foi feita as telas com base nos requisitos indicados no diagrama de caso de uso.. Além disso, Foi desenvolvido o Modelo Entidade Relacionamento (MER) com o intuito de melhor entender como se daria o mapeamento objeto-relacional.

Na Sprint 3 foi implementado o caso de uso referente ao agendamento de um ou mais usuários a quem é chamado de “solicitante”. O agendamento se refere a funcionalidade de usuários poderem marcar uma data e horário de início e fim para usar o laboratório, podendo usar itens pré-selecionados. Tal agendamento pode ou não ter um técnico - orientador dos solicitantes no uso do laboratório. A determinação do técnico é de responsabilidade do administrador do sistema.

Foi definido que apenas os solicitantes do agendamento, o técnico encarregado ou o administrador do sistema teriam permissão para visualizar informações específicas de um agendamento. Isso reflete a preocupação de manter o sistema seguro garantindo que somente usuários autorizados tenha acesso a certas informações, conforme foi orientado pelo cliente.

Uma das principais preocupações no desenvolvimento do CRUD para agendamento, foi estabelecer as diferentes regras de negócio que regem os agendamentos na camada *service*. Isso incluiu a verificação do limite de agendamentos realizados para cada solicitante, assegurando que nenhum usuário ultrapasse sua quota estipulada.

No *front-end*, foram desenvolvidas telas que permitem aos usuários visualizarem os itens disponíveis para agendamento bem como solicitarem agendamentos. Adicionalmente, foi criada uma tela para o administrador visualizar os agendamentos, e outra para exibir detalhes específicos de cada agendamento, incluindo os itens agendados e os usuários envolvidos. Nessa última tela, é possível aprovar ou recusar uma solicitação e atribuir um técnico ao agendamento.

No final da sprint 3 foi feita uma reunião com o cliente a fim de apresentar os avanços do sistema como o cadastro de itens do laboratório e cadastro do usuário via interface.

3.3.1. Tecnologias

No *back-end* foi utilizado o *Spring Boot Framework* do ecossistema *Spring*, utilizando a arquitetura modular [DATASUS 2024]. Tal arquitetura consiste em módulos independentes sendo que cada módulo tem uma responsabilidade, são eles: *security*, *model*, *service*, *repository*, *controller* e *exception*.

Em *model* são definidos os atributos das entidades. *Service* é a camada responsável pela implementação das regras de negócio. O módulo *repository* é o responsável por fazer acesso ao banco de dados *postgresql* mediante o *Spring Data JPA*. O módulo *controller* tem a função de por meio dos *endpoints* na URL do site, fazer acessos aos *service* afim de que o usuário faça uso de algum requisito do sistema.

No módulo de segurança foi usado o *Spring Security*. Por último, o módulo de *exception* contendo todas as possíveis exceções de erros no acesso da API, levando em conta o código de status HTTP.

No desenvolvimento do *front-end*, foi adotado o *Next.js* como o principal *framework*, complementado pelo *Tailwind CSS* para estilização e *shadcn/ui* para componentes pré-projetados.

3.4. Sprint 4

Na sprint 4 a implementação do sistema teve continuidade com o desenvolvimento dos casos de uso referente a empréstimo. Um empréstimo se assemelha a um agendamento envolvendo usuário do sistema, itens e os administradores. A diferença está no fato de que os itens no empréstimo serão utilizados em outro lugar que não é o laboratório lehsa, sendo necessário informar o endereço do lugar de uso para efetuar o empréstimo. Além disso, somente um usuário é responsável pelo empréstimo e não há técnicos ou professores envolvidos, o uso adequado passa a ser de total compromisso do usuário.

Com isso em mente, o desenvolvimento do empréstimo apresentou poucas dificuldades para o time, tendo um alto reuso de código já feito. Entretanto, a equipe teve que adicionar funcionalidades ao agendamento para acomodar pedidos feitos com o cliente durante uma reunião. Esse contratempo foi exacerbado por erros no versionamento do sistema no GitHub. Os erros foram corrigidos e as alterações feitas, o andamento da sprint continuou dentro do tempo restante sem mais problemas aparentes.

No processo de desenvolvimento, foi realizado uma avaliação heurística sobre o design das telas do sistema. Em conjunto com outras equipes, que também estavam desenvolvendo seus próprios projetos de software, os trabalhos desenvolvidos tiveram seus designs avaliados por outro grupo, sendo que um grupo não poderia ter seu trabalho avaliado pelo grupo que ele avaliou.

A avaliação heurística seguiu as heurísticas de Nielsen. Esses princípios são utilizados para avaliar a usabilidade de sistemas interativos, servindo como diretrizes para identificar e solucionar problemas de design. As heurísticas incluem visibilidade do estado do sistema, correspondência entre o sistema e o mundo real, controle e liberdade do usuário, consistência e padrões, prevenção de erros, reconhecimento em vez de recordação, flexibilidade e eficiência de uso, design estético e minimalista, ajuda aos usuários a reconhecer, diagnosticar e corrigir erros, e ajuda e documentação. Essas diretrizes são amplamente utilizadas em testes de usabilidade para melhorar a experiência do usuário.

4. Resultados

4.1. Avaliação Heurística

5. Conclusões e trabalhos futuros

Referências

Amazon Web Services (2024). Amazon AWS. <https://www.amazon.com.br/AWS-Scrum/s?k=AWS+Scrum>.

- Bezerra, E. (2007). *Princípios de Análise e Projeto de Sistemas com UML*. Elsevier, 2ª edition.
- Chen, P. P.-S. (1976). The entity-relationship model—toward a unified view of data. *ACM Trans. Database Syst.*, 1(1):9–36.
- Codd, E. F. (1985). Is your dbms really relational? *Computer World*.
- Colamarco, M. (2023). O que é Extensão Universitária. <https://www.youtube.com/watch?v=3ZdLq7KDK2kab>_{channel = Extens}
- DATASUS (2024). Backend Java - Wiki COATIC. <https://wiki-coatic.saude.gov.br/backend-java: :text=Modularidade,escalados>
- IBM (2024). Topologies Deployment Diagrams. <https://www.ibm.com/docs/pt-br/rsas/7.5.0?topic=topologies-deployment-diagrams>.
- Kieras, R. W. (2019). Sistema para agendamento de serviços. Master's thesis, Universidade Tecnológica Federal do Paraná, Ponta Grossa, Paraná, Brasil. https://riut.utfpr.edu.br/jspui/bitstream/1/16826/1/PG_COADS_2019_102.pdf.
- Machado, F. N. R. (2020). *Banco de dados – projeto e implementação*. Saraiva.
- Nakagawa, J. M. (2014). Sistema Gerenciador de Agendamento de Serviços: Um estudo de caso em salão de beleza. Master's thesis, Universidade Tecnológica Federal do Paraná, Cornélio Procópio, Paraná, Brasil. https://repositorio.utfpr.edu.br/jspui/bitstream/1/28362/1/CP_COADS_2014_113.pdf.
- Pereira, P., Torreão, P., and Maral, A. S. (2007). Entendendo scrum para gerenciar projetos de forma Ágil. *Mundo PM*, 1(14):64–71.
- Red Hat (2022). O que é a metodologia ágil. <https://www.redhat.com/pt-br/topics/devops/what-is-agile-methodology>.
- Royce, W. W. (1970). Managing the Development of Large Software Systems. *Proceedings of IEEE WESCON*, 26:1–9.
- SimplyBook (2024). SimplyBook.me. <https://simplybook.me/pt/>.
- State of Agile (2020). 14th Annual State of Agile Report. <https://www.qagile.pl/wp-content/uploads/2020/06/14th-annual-state-of-agile-report.pdf>.
- Valente, M. T. (2022). *Engenharia de Software Moderna*. Editora Independente, 1ª edition.