

**CENTRO UNIVERSITÁRIO ESTÁCIO
JOÃO GOULART**

SISTEMA PARA lavanderia

Jéssica Almeida de Oliveira

**2024
Juiz de Fora/MG**

Sumário

1. DIAGNÓSTICO E TEORIZAÇÃO	5
1.1. Identificação das partes interessadas e parceiros	5
1.2. Problemática e/ou problemas identificados	6
1.3. Justificativa	7
1.4. Objetivos/resultados/efeitos a serem alcançados (em relação ao problema identificado e sob a perspectiva dos públicos envolvidos)	8
1.5. Referencial teórico (subsídio teórico para propositura de ações da extensão)	10
2. PLANEJAMENTO E DESENVOLVIMENTO DO PROJETO	12
2.1. Plano de trabalho (usando ferramenta acordada com o docente)	12
2.2. Descrição da forma de envolvimento do público participante na formulação do projeto, seu desenvolvimento e avaliação, bem como as estratégias pelo grupo para mobilizá-los.	14
2.3. Grupo de trabalho (descrição da responsabilidade de cada membro)	15
2.4. Metas, critérios ou indicadores de avaliação do projeto	15
2.5. Recursos previstos	17
2.6. Detalhamento técnico do projeto	19
3. ENCERRAMENTO DO PROJETO	33
3.1. Relatório Coletivo (podendo ser oral e escrita ou apenas escrita)	38
3.2. Avaliação de reação da parte interessada	38
3.3. Relato de Experiência Individual	40
3.1. CONTEXTUALIZAÇÃO	40
3.2. METODOLOGIA	41
3.3. RESULTADOS E DISCUSSÃO:	41
3.4. REFLEXÃO APROFUNDADA	41
3.5. CONSIDERAÇÕES FINAIS	41

1. DIAGNÓSTICO E TEORIZAÇÃO

1.1. Identificação das partes interessadas e parceiros

Jéssica Almeida de Oliveira só estudando do curso analista desenvolvimento de sistemas futura profissional da área de TI e cursando na Universidade Estácio de Sá (2024), A criação de um documento HTML para um lavadeira que lava roupas domésticas para poder transformar toda a roupa suja ou contaminada utilizada na instituição em roupa limpa. Este processo é extremamente importante para o bom funcionamento das unidades em relação à assistência direta ou indireta prestada aos adolescentes, sujeiras pesadas e profundas de áreas maiores com rapidez e agilidade! No caso de uso intensivo, a lavadora de alta pressão profissional é a melhor opção para limpar sujeiras pesadas e profundas de grandes áreas. É uma excelente máquina para ser usada em empreendimentos como lava car ou comércios. para ajudá pessoa que não tem tempo para lava roupas em casa cria esta página na intenção de atingir mais públicos

Lavar, secar peças de usuário, roupas de cama e mesa e outras similares, utilizando processos manuais e/ou mecânicos para eliminar sujeiras. Preço por diária: R\$ 160,00 (média) Produtos de limpeza: R\$ 55,00 partir de R\$ 1.000 e o preço médio fica em torno de R\$ 2.000 A atividade da lavadeira era uma das atribuições que estavam reservadas às escravas da casa dos senhores; junto às mucamas e às costureiras, representaram uma transformação da mulher com uma profissão, no século XIX.

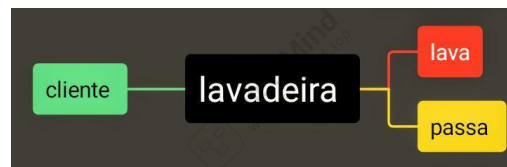
Também possui ampla experiência como docente e pesquisador. Lecionou em MBAs nas áreas de logística, gestão estratégica de processos e tecnologias aplicadas ao BPM, além de ministrar cursos de extensão em ferramentas como Visual Studio, HTML e administração de sistemas windows. Seus artigos publicados em revistas científicas e apresentações em congressos acadêmicos evidenciam seu compromisso com a disseminação do conhecimento e a busca contínua por soluções inovadoras. Entre seus trabalhos, destacam-se estudos sobre previsão de demanda, HTML e transformação digital para uma lavanderia

Com uma abordagem proativa e dinâmica, Jessica tem se destacado por sua capacidade de solucionar problemas complexos e promover melhorias contínuas. Suas competências em análise de dados, machine learning e prestação de serviços à experiência em coordenação de projetos, o tornam um profissional que presta

serviço de lavar e passar roupas com muita qualificação para enfrentar os desafios da transformação digital e da gestão estratégica em organizações de diferentes setores.

A seleção da empresa para o estudo foi realizada mediante uma colaboração entre os discentes, levando em conta fatores como a experiência passível de ser obtida, a visibilidade da companhia e o apoio oferecido. A organização em questão é a lavanderia no Distrito Industrial de Juiz de Fora, estado de Minas Gerais.

Imagem 01: Instalações da operadora Ambev



Fonte: O autor (2024).

1.2. Problemática e/ou problemas identificados

A fim de assegurar ao serviço de lava e passa roupa tanto dos transeuntes quanto dos colaboradores envolvidos, é imperativo que inspeções de serviço site sejam realizadas com regularidade, objetivando identificar problemas que possam comprometer a viagem e/ou prever futuras falhas. Atualmente, não existe um plano de manutenção capaz de mapear de forma preditiva, uma vez que não há uma base de registro dos resultados das inspeções. Consequentemente, há veículos que são avaliados mais de uma vez por unidade de tempo, enquanto outros jamais são inspecionados.

Em uma análise preliminar, constatou-se que um sistema de gestão das avaliações e inspeções de segurança demonstrou eficácia no mapeamento das falhas mais recorrentes, no levantamento das necessidades de materiais para substituição e na alocação adequada de mão de obra.

1.3. Justificativa

Que lavavam a roupa de outrem, utilizando, geralmente, os lavadeira públicos mas, também, as ribeiras ou as levadas.Organiza as roupas para serem lavadas e passadas, tira manchas, ting e dá acabamento em artigos do vestuário, cama, mesa e banho.Mas voltando séculos,

bem antes da “briga” pela patente da máquina elétrica, de acordo com o World Atlas, o alemão Jacob Christian Schaefer inventou em 1767 o modelo considerado a primeira máquina de lavar.²¹ Antes da invenção do sabão, para limpar algum tecido, era preciso, por exemplo, batê-lo sobre pedras em um rio. A fricção com a pedra e a ação da água arrancavam a sujeira da roupa – sobretudo substâncias gordurosas, que não se dissolvem na

trabalhavam as escravas recém-libertas, mulheres teoricamente livres, que foram jogadas à própria sorte na vida após o fim da escravidão, sem transição ou políticas de adaptação. produto líquido, rico em cloro, muito usado nas residências sendo aí usado para branqueamento de roupas e limpeza de ambientes

recursos. Segundo Favero (2020, p. 112), “a ausência de um registro sistemático de inspeções limita a capacidade de identificar padrões e prever falhas, resultando em uma gestão reativa e ineficiente.” Dessa forma, o desenvolvimento de um sistema para registrar e gerir os resultados das inspeções surge como uma solução imprescindível.

A implementação de um página web para inspeções de serviços permite atrair mais clientes com falhas mais recorrentes, a previsão de necessidades futuras e a alocação estratégica de recursos. Conforme Giorgi (2019, p. 88), “sistemas preditivos são fundamentais para prevenir problemas e mitigar riscos antes que estes comprometam a operação.” Essa abordagem, baseada em registros estruturados e análise de dados, contribui para a criação de um plano de manutenção eficiente, reduzindo o retrabalho e eliminando a duplicidade nas avaliações de veículos, como mencionado no cenário analisado.

Do ponto de vista da engenharia de manutenção, a aplicação de técnicas preditivas apoia-se em fundamentos estatísticos e matemáticos, sendo potencializada pelo uso de sistemas informatizados. Grus (2019, p. 96) destaca que “os modelos preditivos baseados em dados históricos são mais eficazes na antecipação de falhas, promovendo economia de recursos e aumento da confiabilidade.” Assim, a adoção de um sistema informatizado não apenas organiza os registros de inspeção, mas também fornece insumos para análises mais complexas, como a identificação de correlações entre falhas mecânicas e condições operacionais.

A literatura aponta que a falta de previsibilidade nas manutenções pode gerar impactos significativos em termos de custos operacionais e produtividade. Giorgi (2019) afirma que um sistema de gestão bem estruturado “reduz o tempo de inatividade e otimiza o uso de mão de obra e materiais, fatores críticos em operações logísticas.” Esses benefícios não apenas elevam o nível de segurança,

mas também melhoram os indicadores de desempenho organizacional, permitindo uma gestão mais sustentável e eficaz.

Por fim, o desenvolvimento de um site de gestão de inspeções de segurança atende à crescente demanda por soluções baseadas em tecnologia da informação, alinhando-se às melhores práticas do mercado. Como observa Favero (2020, p. 128), “a digitalização de processos críticos, como a manutenção de veículos, representa um passo essencial na modernização das operações, aumentando a transparência e a rastreabilidade das informações.” Portanto, a criação de um sistema preditivo contribui diretamente para o serviço e eficiência, demonstrando ser uma solução prática e sustentável para os desafios identificados.

1.4. Objetivos/resultados/efeitos a serem alcançados (em relação ao problema identificado e sob a perspectiva dos públicos envolvidos)

A segurança operacional é um dos pilares fundamentais para a sustentabilidade e eficiência das empresas que dependem de frotas de veículos. A ausência de processos sistematizados para inspeções regulares e manutenção preditiva pode resultar em falhas operacionais, aumentando os riscos para colaboradores e transeuntes, além de impactar negativamente nos custos e na confiabilidade das operações. Nesse contexto, a gestão eficiente das inspeções de segurança torna-se uma necessidade estratégica para mitigar riscos e garantir o bom funcionamento das atividades logísticas.

A implementação de um sistema informatizado para o registro e análise de inspeções de segurança permite não apenas organizar as informações, mas também integrar tecnologias de análise de dados e machine learning para prever falhas e otimizar recursos. De acordo com estudos recentes, as organizações que adotam soluções preditivas conseguem reduzir significativamente o tempo de inatividade dos veículos e os custos operacionais, além de aumentar a confiabilidade de suas operações. Essa abordagem se alinha às demandas do mercado por maior eficiência e segurança nas atividades empresariais.

Este projeto propõe o desenvolvimento e a implementação de um sistema de gestão de inspeções de segurança, com foco na modernização das práticas de manutenção e na garantia de segurança operacional. Por meio de uma abordagem

baseada em dados e tecnologias avançadas, busca-se transformar desafios em oportunidades, promovendo uma operação mais sustentável e competitiva.

Objetivos a serem alcançados:

1) Implementação de um Sistema de Gestão de Inspeções de Segurança

Desenvolver e implantar um sistema informatizado para o registro, acompanhamento e análise dos resultados das inspeções de segurança em veículos, assegurando a organização e centralização das informações.

2) Previsão de Falhas por Meio de Análises Preditivas

Utilizar técnicas de análise de dados e machine learning para identificar padrões e prever falhas futuras, permitindo a implementação de um plano de manutenção preditiva eficiente.

3) Otimização de Recursos Operacionais

Alocar de forma estratégica a mão de obra e os materiais necessários, com base em um planejamento fundamentado nos dados registrados, evitando duplicidade de inspeções e subutilização de recursos.

4) Aumento da Segurança e Redução de Riscos

Promover a segurança dos colaboradores e transeuntes, minimizando os riscos associados a falhas mecânicas e operacionais nos veículos.

5) Melhoria da Eficiência Operacional

Reduzir o tempo de inatividade dos veículos e garantir maior confiabilidade nas operações, impactando positivamente nos indicadores de desempenho e na satisfação dos stakeholders.

Resultados Esperados:

1) Base de Dados Centralizada e Estruturada

A criação de uma base de dados robusta com o histórico detalhado das inspeções de segurança, proporcionando acesso rápido e confiável às informações.

2) Plano de Manutenção Preditiva

Estabelecimento de um plano de manutenção baseado em previsões fundamentadas, reduzindo ocorrências inesperadas e aumentando a eficiência dos processos de manutenção.

3) Redução de Custos Operacionais

Otimização do uso de materiais, redução do retrabalho e eliminação de avaliações redundantes, resultando em uma operação mais econômica.

4) Maior Segurança nas Operações

Identificação e mitigação de riscos antes que comprometam a operação, garantindo um ambiente mais seguro para colaboradores e público em geral.

5) Eficiência e Confiabilidade na Logística

Aumento da confiabilidade dos veículos e da pontualidade nas entregas, contribuindo para a reputação da organização e para o alcance de seus objetivos estratégicos.

1.5. Referencial teórico

O desenvolvimento página web baseado no HTML distingue-se pela flexibilidade, interação contínua com os stakeholders e ciclos de entregas curtos, contrastando significativamente com os modelos tradicionais, como o modelo em cascata. A agilidade foca em responder rapidamente às mudanças e priorizar entregas de valor, enquanto os modelos tradicionais tendem a seguir um planejamento rigoroso e linear. De acordo com Beck et al. (2001), “a capacidade de atrair mais clientes às mudanças é uma das maiores forças do desenvolvimento ágil, pois permite alinhar o software às necessidades reais do cliente durante o processo.”

O primeiro passo essencial na metodologia ágil é a formação de equipes multidisciplinares e auto gerenciadas, que compartilham responsabilidades pelo sucesso do projeto. A colaboração estreita entre desenvolvedores, analistas, designers e clientes é fundamental. Segundo Schwaber e Sutherland (2020), “equipes ágeis são pequenas, autônomas e multifuncionais, o que promove uma entrega consistente de valor.” Em contraste, modelos tradicionais dividem as equipes por funções, o que pode gerar silos organizacionais e atrasos na comunicação.

Na sequência, ocorre a definição inicial dos requisitos de forma incremental e adaptativa, diferentemente dos modelos tradicionais, que exigem uma documentação extensa e fechada antes do desenvolvimento. No ágil, os requisitos são priorizados em backlog e refinados a cada iteração. Pressman (2020, p. 43) explica que “o backlog de produto é um elemento vivo, permitindo mudanças

contínuas conforme as necessidades do cliente evoluem.” Essa abordagem garante maior alinhamento entre o software entregue e as expectativas do cliente.

O planejamento iterativo é outro aspecto fundamental. As metodologias ágeis, como Scrum e Kanban, dividem o trabalho em sprints ou ciclos curtos, geralmente de 2 a 4 semanas. Sommerville (2019) observa que “o planejamento em ciclos curtos melhora a previsibilidade do projeto, ao mesmo tempo que permite ajustes rápidos para lidar com mudanças.” Por outro lado, no modelo em cascata, o planejamento completo ocorre antes da execução, reduzindo a capacidade de adaptação ao longo do desenvolvimento.

Durante cada sprint, o desenvolvimento é guiado por incrementos de software funcionais, que são apresentados ao cliente para feedback contínuo. Este passo assegura que o produto final esteja em conformidade com as expectativas reais do cliente, diferentemente dos modelos tradicionais, onde o cliente só visualiza o produto no final do projeto. Beck et al. (2001) destacam que “a entrega contínua de incrementos funcionais proporciona maior transparência no progresso e maior confiança no projeto.”

Outro elemento importante é a integração contínua e testes frequentes, que garantem a qualidade do software em cada etapa. O foco no feedback imediato distingue as metodologias ágeis, permitindo a detecção precoce de falhas. Conforme Fowler (2006), “a integração contínua reduz os custos de defeitos ao corrigi-los no momento em que são introduzidos, e não após a conclusão do ciclo de desenvolvimento.” Esse processo reduz drasticamente os custos de retrabalho, comuns nos modelos em cascata.

Além disso, a revisão e retrospectiva após cada sprint são etapas essenciais para o aprimoramento contínuo do processo. Nessas reuniões, a equipe identifica os pontos fortes e áreas de melhoria, promovendo ajustes no processo. Schwaber e Sutherland (2020) afirmam que “as retrospectivas ajudam as equipes a se tornarem mais eficientes ao refletirem sobre suas práticas e ao implementarem mudanças iterativas no processo.”

Finalmente, o ágil prioriza a entrega de valor ao cliente, um dos maiores diferenciais em relação aos modelos tradicionais. A abordagem incremental permite que o

cliente obtenha funcionalidades utilizáveis desde as primeiras etapas, aumentando o retorno sobre o investimento. Segundo Pressman (2020, p. 67), “a entrega de valor contínuo é o princípio norteador do desenvolvimento ágil, colocando as necessidades do cliente no centro de cada decisão.”

Os benefícios do desenvolvimento ágil incluem maior flexibilidade, menor risco de falhas devido ao feedback contínuo, melhor alinhamento com as expectativas do cliente e maior eficiência operacional. Embora os modelos tradicionais tenham seu lugar em projetos altamente regulamentados, o ágil provou ser a escolha preferida em ambientes dinâmicos e inovadores.

2. PLANEJAMENTO E DESENVOLVIMENTO DO PROJETO

2.1. Plano de trabalho

A seguir, apresenta-se o cronograma das principais atividades a serem executadas na empresa parceira. As etapas do desenvolvimento do software englobam desde as reuniões com os gestores da organização até a conclusão e entrega do projeto.

Considerando que a empresa em questão é um operador logístico de grande volume, a presença dos discentes estará condicionada à supervisão de um responsável técnico. Neste caso, o estudante de análise e desenvolvimento de sistema, Sra. Jessica Almeida de Oliveira, portador do documento, assumirá tal responsabilidade. O início das atividades no local somente poderá ocorrer após a assinatura dos documentos pertinentes pelos gestores e o registro da Anotação de Responsabilidade Técnica (ART).

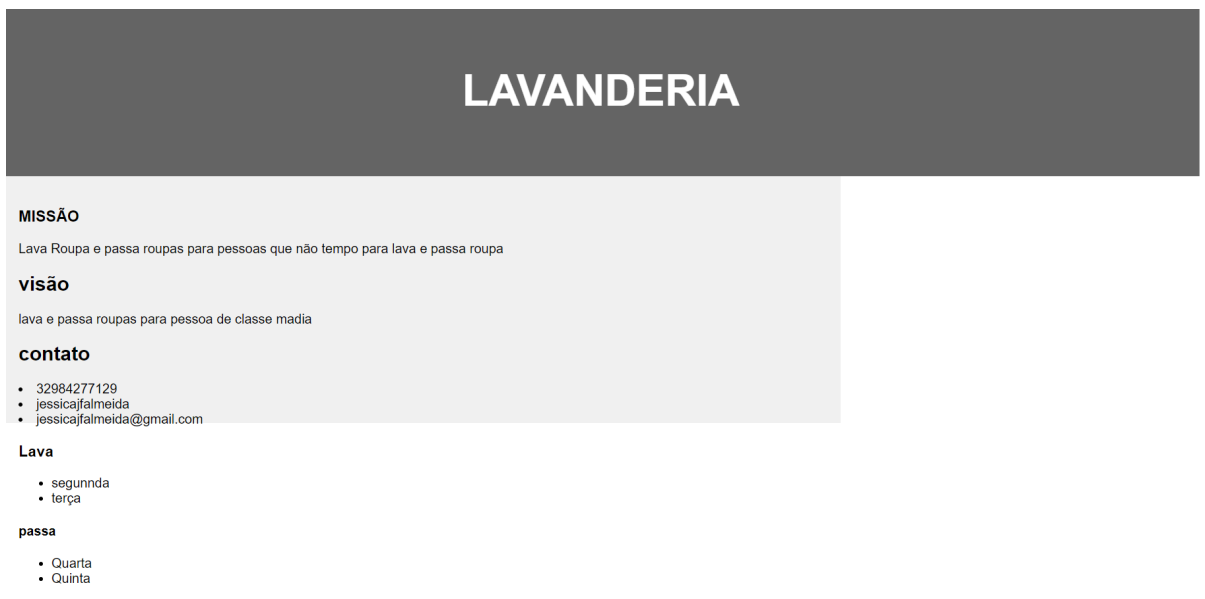


imagem 01: Cronograma de atividade Fonte: O autor.

2.2. Descrição da forma de envolvimento do público participante na formulação do projeto, seu desenvolvimento e avaliação, bem como as estratégias utilizadas pelo grupo para mobilizá-los.

A identificação e análise das necessidades de obter uma página HTML são etapas fundamentais no processo de desenvolvimento de software. O objetivo dessas etapas é entender as expectativas e requisitos dos stakeholders, a fim de desenvolver uma página web que atenda às suas necessidades e funcione de maneira eficiente e eficaz. Neste projeto, foram realizadas entrevistas com diversos grupos de stakeholders, incluindo o time de segurança do trabalho, os responsáveis pela logística, as gerências comercial e financeira, e o time de tecnologia da informação.

2.2.1) Entrevistas com Stakeholders

As entrevistas com os stakeholders foram conduzidas seguindo uma abordagem estruturada, com o objetivo de coletar informações detalhadas sobre as necessidades, expectativas e restrições associadas ao sistema de inspeção de segurança. A seguir, apresentamos um resumo das entrevistas realizadas com cada grupo de stakeholders:

- **Time de Segurança do Trabalho:** As entrevistas com o time de segurança do trabalho focaram em identificar os requisitos funcionais e não funcionais relacionados à inspeção de segurança, bem como as normas e regulamentações aplicáveis ao setor. Os membros do time de segurança do trabalho forneceram informações valiosas sobre os processos atuais de inspeção, as áreas de melhoria potenciais e as expectativas em relação ao novo sistema.
- **Responsáveis pela Logística:** As entrevistas com os responsáveis pela logística abordaram os aspectos operacionais e de gerenciamento relacionados ao sistema de inspeção de segurança. Os participantes compartilharam informações sobre os fluxos de trabalho atuais, os desafios enfrentados e as expectativas quanto à integração do novo sistema às operações logísticas existentes.
- **Gerências Comercial e Financeira:** As entrevistas com as gerências comercial e financeira tiveram como objetivo entender os impactos do sistema de inspeção de segurança nos processos comerciais e financeiros da organização. Os participantes forneceram informações sobre os requisitos de relatórios e análises, bem como as expectativas em relação ao retorno sobre o investimento e aos benefícios da página para a organização os dia e horário.
- **Time de Tecnologia da Informação:** As entrevistas com o time de tecnologia da informação focaram na avaliação dos recursos disponíveis e das restrições tecnológicas para o desenvolvimento e implantação do site de inspeção de serviço. Os membros do time de tecnologia da informação compartilharam informações sobre as plataformas, infraestrutura e ferramentas existentes, bem como as políticas e diretrizes o serviço da informação aplicáveis ao projeto.

Com base nas informações coletadas durante as entrevistas com os stakeholders, a equipe de projeto realizou uma análise detalhada das necessidades do sistema, identificando os requisitos funcionais e não funcionais, as restrições e os objetivos do projeto. Essa análise permitiu que a equipe desenvolvesse uma visão

abrangente e integrada das necessidades do sistema, garantindo que todas as expectativas e requisitos dos stakeholders sejam considerados no processo de desenvolvimento.

2.3. Grupo de trabalho

2.4. Metas, critérios ou indicadores de avaliação do projeto

O desenvolvimento de uma página HTML informatizada a lavanderia de segurança exige o estabelecimento de metas claras e critérios de avaliação que garantam a eficiência, a qualidade e a sustentabilidade do projeto. A aplicação dos princípios de uma página web permite estruturar essas metas e critérios de maneira mensurável e alinhada às melhores práticas do setor.

2.4.1) Metas do Projeto

Entrega do site no Prazo Estipulado: Garantir que o sistema informatizado seja desenvolvido e entregue dentro do cronograma previamente definido, minimizando atrasos e maximizando a eficiência do processo de desenvolvimento. Segundo Sommerville (2019, p. 134), “cumprir os prazos é essencial para alinhar o desenvolvimento do software aos objetivos estratégicos da organização.”

Funcionalidade e Usabilidade do site: Desenvolver uma página web que atenda aos requisitos funcionais e não funcionais, priorizando a facilidade de uso pelos operadores e gestores. De acordo com Pressman (2024), páginas que falham em atender aos critérios de usabilidade comprometem significativamente sua adoção e eficiência operacional.

Confiabilidade e Robustez: Assegurar que o sistema seja confiável e resistente a falhas, permitindo o uso contínuo mesmo em cenários de alta demanda. Como enfatiza Sommerville (2019), “a confiabilidade é um dos atributos mais críticos em sistemas de missão crítica, sendo essencial para a confiança dos usuários e para a continuidade das operações.”

Integração com Processos Operacionais Existentes: Garantir que a página seja capaz de se integrar com os processos operacionais e ferramentas já utilizadas pela organização, reduzindo resistências e promovendo uma transição suave. O trabalho de integração é vital, uma vez que, conforme Brooks (1995, p. 42), “a complexidade do página web é inerente e não pode ser eliminada, mas pode ser gerenciada por meio de boas práticas de desenvolvimento e integração.”

2.4.2) Critérios de Avaliação do Desempenho

Adesão ao Cronograma: Monitorar o progresso do projeto em relação ao cronograma planejado, avaliando entregas parciais em marcos estabelecidos. Sommerville (2019) destaca que “dividir o desenvolvimento em marcos menores permite maior controle sobre o progresso do projeto.”

Cobertura dos Requisitos: Avaliar se o sistema cumpre todos os requisitos levantados durante a etapa de análise, tanto os funcionais quanto os não funcionais, com validação por meio de testes de aceitação. Pressman (2020, p. 102) salienta que “a rastreabilidade dos requisitos é uma prática essencial para garantir que o produto final atenda às expectativas do cliente.”

Satisfação do Usuário: Medir a satisfação dos usuários com base em critérios de usabilidade, desempenho e confiabilidade, utilizando ferramentas como questionários e entrevistas após as etapas de teste. Brooks (1995) argumenta que a experiência do usuário deve ser o foco central no desenvolvimento de sistemas, pois é um dos principais indicadores de sucesso.

Eficiência Operacional: Analisar o impacto do página na operação, avaliando métricas como redução no tempo de inatividade dos veículos e na duplicidade de inspeções. Isso reflete a efetividade do sistema em cumprir os objetivos estratégicos estabelecidos.

Ao combinar metas bem definidas com critérios de avaliação robustos, este projeto será monitorado e ajustado continuamente para garantir sua entrega com qualidade, no prazo e de acordo com as necessidades organizacionais.

2.5. Recursos previstos

Neste projeto, serão utilizadas diversas ferramentas e metodologias para garantir um gerenciamento eficiente de mudanças e uma colaboração eficaz entre os membros da equipe. As ferramentas escolhidas incluem o GitHub para controle de versão, o Trello para gerenciamento de projetos e o Slack para comunicação e colaboração. A seguir, detalharemos como cada uma dessas ferramentas será aplicada no projeto.

GitHub como Ferramenta de Controle de Versão: O GitHub será utilizado como a principal ferramenta de controle de versão neste projeto, permitindo que a equipe gerencie e rastreie mudanças no código-fonte e na documentação de maneira eficiente e colaborativa. O GitHub oferece uma plataforma baseada em nuvem para hospedar repositórios Git, que são utilizados para armazenar e controlar todas as versões dos artefatos do projeto.

Neste projeto, adotaremos uma estrutura de ramificação que consiste em duas linhas principais: a linha de base (produção) e a linha de desenvolvimento.

- 1) Linha de base (produção): A linha de base representará a versão atual do página em produção. Essa linha conterá o código-fonte e a documentação estáveis e aprovados, que são considerados prontos para implantação e uso pelos clientes finais. A linha de base será atualizada periodicamente, conforme as mudanças aprovadas sejam implementadas e testadas com sucesso.
- 2) Linha de desenvolvimento: A linha de desenvolvimento será usada para a implementação e teste de novos recursos, correções de bugs e melhorias no projeto. Os membros da equipe trabalharão nessa linha, criando branches específicas para cada tarefa ou mudança proposta. Após a conclusão e a aprovação da mudança, a branch será mesclada de volta à linha de desenvolvimento. Periodicamente, a linha de desenvolvimento será mesclada à linha de base após uma revisão cuidadosa e testes rigorosos, garantindo a estabilidade e a qualidade do software em produção.

Trello para Gerenciamento de Projetos: O Trello será utilizado como a ferramenta de gerenciamento de projetos neste projeto, permitindo que a equipe planeje, monitore e colabore em tarefas e atividades relacionadas ao projeto. O Trello utiliza um formato de quadro Kanban, que permite a criação de listas e cartões para representar tarefas, estados e informações relevantes.

A equipe utilizará o Trello para organizar e priorizar tarefas, atribuir responsabilidades e acompanhar o progresso das atividades do projeto. O quadro Trello será atualizado continuamente para refletir o status atual do projeto e as mudanças propostas, garantindo que todas as partes interessadas estejam cientes do progresso e das atividades em andamento.

Slack para Comunicação e Colaboração: O Slack será usado como a principal ferramenta de comunicação e colaboração neste projeto. O Slack oferece um ambiente centralizado para compartilhar informações, discutir problemas e tomar decisões relacionadas ao projeto. A equipe utilizará canais específicos para diferentes tópicos e atividades do projeto, garantindo que a comunicação seja organizada e relevante.

2.6. Detalhamento técnico do projeto

Os requisitos funcionais a seguir abordam as principais funcionalidades do software, fornecendo uma visão geral das ações e comportamentos esperados do sistema. Eles servirão como ponto de partida para o projeto, codificação, teste e validação do software, garantindo que ele atenda às necessidades dos usuários e contribua para o sucesso geral do projeto.

2.6.1) Identificação dos stakeholders:

➤ lava

➤ passa

2.6.2) Requisitos funcionais:

- Autenticação de usuários: Os usuários devem ser capazes de fazer login na página html com suas credenciais;
- Cadastro de inspetores: Permitir o cadastro e gerenciamento dos inspetores de segurança;
- Agendamento de inspeções: Os inspetores devem ser capazes de agendar inspeções futuras;
- Registro de inspeções: Os inspetores devem ser capazes de registrar informações detalhadas das inspeções realizadas, como data, hora, local, resultados e fotos.
- Geração de relatórios: O sistema deve ser capaz de gerar relatórios de inspeção personalizados por período, local e inspetor.
- Acompanhamento de ações corretivas: Permitir o registro e acompanhamento das ações corretivas necessárias após as inspeções.
- Notificações: Enviar notificações para os responsáveis pelas ações corretivas e para os gerentes de segurança.
- Integração com outros sistemas: O software deve ser capaz de se integrar a outros sistemas utilizados na empresa, como sistemas de gerenciamento de estoque e logística.

2.6.3) Requisitos não funcionais

Além dos requisitos funcionais, que descrevem as funcionalidades e comportamentos do sistema, os requisitos não funcionais são igualmente importantes no processo de desenvolvimento de software. Eles abordam aspectos relacionados à qualidade do sistema, como desempenho, segurança, usabilidade, confiabilidade e escalabilidade. Esses requisitos estabelecem como o software deve funcionar e impactam diretamente a experiência do usuário e a eficiência operacional do sistema.

Nesta seção, apresentamos uma lista de requisitos não funcionais identificados para a página HTML em desenvolvimento. Esses requisitos foram coletados e refinados em colaboração com os stakeholders e a equipe de desenvolvimento, assim como os requisitos funcionais. Eles são apresentados de forma clara, mensurável e verificável, para facilitar a avaliação da qualidade e o desempenho do software durante o processo de teste e validação.

Os requisitos não funcionais a seguir abordam características cruciais do sistema que garantem sua adequação ao propósito, eficiência e satisfação do usuário. Eles desempenham um papel vital no desenvolvimento de software, pois influenciam a arquitetura do sistema, as decisões de design e a seleção de tecnologias e componentes.

Requisitos não funcionais:

- Segurança: O site deve ser seguro e proteger as informações de acesso não autorizado.
- Usabilidade: A interface do usuário deve ser intuitiva e fácil de usar.
- Desempenho: O site deve responder rapidamente às ações do usuário e ser capaz de lidar com muitas inspeções e usuários.
- Escalabilidade: O site deve ser capaz de acomodar o crescimento da empresa e um aumento no número de inspeções e usuários.
- Portabilidade: O site deve ser compatível com diferentes dispositivos e plataformas, como smartphones, tablets e computadores.

2.6.7) Categorização

Categorizar os itens da especificação ajuda a organizar os requisitos e facilita a comunicação, priorização e implementação durante o desenvolvimento do site. Os requisitos foram divididos nas categorias a saber:

- Requisitos de Interface do Usuário (UI): Essa categoria inclui todos os requisitos relacionados à interface gráfica do usuário e à interação com o sistema. Os requisitos podem abordar elementos como layout, navegação, cores e tipografia.
- Requisitos de Processo: Essa categoria abrange os requisitos que descrevem as funcionalidades e processos do sistema, como criação de registros, atualizações, cálculos e geração de relatórios.
- Requisitos de Integração: Essa categoria inclui requisitos relacionados à integração do software com outros sistemas e serviços, como APIs, bancos de dados e sistemas legados.
- Requisitos de Desempenho: Essa categoria aborda os requisitos que especificam o desempenho do sistema em termos de velocidade, capacidade e eficiência.
- Requisitos de site Essa categoria inclui requisitos relacionados à proteção do sistema e dos dados do usuário contra acessos não autorizados, vazamentos de informações e outras ameaças à segurança.
- Requisitos de Usabilidade: Essa categoria aborda os requisitos que garantem a facilidade de uso do sistema, como a organização de informações, a consistência da interface do usuário e a disponibilidade de ajuda e suporte.
- Requisitos de Confiabilidade: Essa categoria inclui requisitos que especificam a confiabilidade do sistema, como a disponibilidade, a tolerância a falhas e a recuperação de erros.
- Requisitos de Escalabilidade: Essa categoria aborda os requisitos que garantem a capacidade do sistema de lidar com o crescimento do usuário e da carga de trabalho, como a capacidade de adicionar recursos e acomodar um maior volume de transações.

Devido à natureza deste estudo acadêmico, que possui restrições temporais e espaciais, inicialmente será atribuída prioridade aos requisitos relacionados à interface, segurança e usabilidade.

2.6.8) Requisitos de interface

Os requisitos de interface do usuário (UI) descrevem os elementos visuais e interativos do site, garantindo uma experiência agradável e eficiente para os usuários. Seguem alguns requisitos de interface do usuário para o site:

Layout responsivo: O site deve ter um layout responsivo que se adapte automaticamente a diferentes tamanhos e resoluções de tela, proporcionando uma experiência de usuário consistente em dispositivos móveis e desktops.

Navegação clara e intuitiva: O site deve incluir um menu de navegação fácil de usar e compreensível, permitindo que os usuários encontrem rapidamente as informações e funcionalidades desejadas.

Hierarquia visual: O design do site deve apresentar uma hierarquia visual clara, com títulos e subtítulos distintos e organizados de forma lógica, facilitando a leitura e compreensão das informações.

Cores e tipografia consistentes: O site deve utilizar uma paleta de cores harmoniosa e tipografia consistente em todas as páginas, garantindo uma aparência profissional e agradável.

Imagens e ícones de alta qualidade: O site deve incluir imagens e ícones de alta qualidade e relevantes para o conteúdo, melhorando a estética e a compreensão das informações apresentadas.

Formulários de entrada de dados user-friendly: Os formulários de entrada de dados no site, como campos de pesquisa e formulários de contato, devem ser fáceis de usar e fornecer feedback adequado aos usuários.

Acessibilidade: O site deve seguir as diretrizes de acessibilidade para garantir que os usuários com deficiências possam navegar e interagir com o conteúdo de forma eficiente.

Feedback do usuário: O site deve fornecer feedback visual e sonoro aos usuários quando eles interagem com elementos da interface, como botões e links, para garantir uma experiência de usuário agradável e eficiente.

Páginas de erro personalizadas: O site deve incluir páginas de erro personalizadas (como 404 - Página não encontrada) que informem os usuários sobre problemas e ofereçam opções para retornar à navegação.

Idiomas suportados: Se necessário, o site deve oferecer suporte a múltiplos idiomas e permitir que os usuários selecionem seu idioma preferido.

Esses requisitos de interface do usuário servem como base para o design e desenvolvimento do site, garantindo que ele seja atraente, funcional e fácil de usar para todos os visitantes.

2.6.9) Requisitos de segurança

Os requisitos de segurança são essenciais para proteger o site e os dados dos usuários contra ameaças e atividades maliciosas. Seguem abaixo os requisitos de segurança que deverão ser adotados ou levados em consideração durante o desenvolvimento da aplicação:

Autenticação e autorização:

- Implementar um site de autenticação de usuários robusto com senhas seguras, exigindo complexidade mínima (comprimento, caracteres especiais, números e letras maiúsculas e minúsculas).
- Permitir autenticação de dois fatores (2FA) como uma opção de segurança adicional.
- Controlar o acesso a recursos com base nos privilégios e funções dos usuários.

Proteção de dados:

- Criptografar dados confidenciais, como informações de pagamento e senhas, usando algoritmos de criptografia utilizados e comprovados.
- Utilizar conexões HTTPS seguras com certificados SSL válidos para garantir a privacidade e integridade dos dados transmitidos.
- Estabelecer políticas de retenção de dados e remoção segura de informações pessoais quando não forem mais necessárias.

Prevenção de ataques comuns:

- Proteger o site contra possíveis ataques de injeção de SQL, garantindo a validação e a higienização adequadas das entradas do usuário e utilizando consultas parametrizadas.
- Implementar medidas para evitar ataques de Cross-Site Scripting (XSS), como a higienização de entradas do usuário e o uso de cabeçalhos de segurança.
- Prevenir ataques de falsificação de solicitações entre sites (CSRF) usando tokens e verificação de referência.

Monitoramento e auditoria:

- Estabelecer um sistema de registro e monitoramento para identificar atividades suspeitas e potenciais violações de segurança.
- Realizar auditorias de segurança periódicas para identificar e corrigir vulnerabilidades no site e na infraestrutura.
- Estabelecer planos de resposta a incidentes de segurança para lidar com violações e minimizar seus impactos.

Segurança do servidor e da infraestrutura:

- Manter o software do servidor, como sistemas operacionais e aplicativos, atualizado com as últimas correções de segurança.
- Implementar firewalls e site de detecção e prevenção de intrusão (IDS/IPS) para proteger o site contra ameaças externas.
- Garantir que os backups sejam realizados regularmente e armazenados em locais seguros, possibilitando a recuperação rápida em caso de falha do site ou perda de dados.

2.6.10) Priorização

A priorização de requisitos é um aspecto fundamental no gerenciamento de projetos de site pois ajuda a garantir que os recursos e esforços sejam alocados de forma eficiente, maximizando o valor entregue ao usuário e às partes interessadas. Neste contexto, é crucial estabelecer um processo de priorização sistemático e baseado em critérios objetivos, que permita a tomada de decisões informadas e a alocação eficaz de recursos (Karlsson, 1995; Leffingwell, 2011).

Para priorizar os requisitos do projeto, os seguintes critérios podem ser considerados:

Impacto no usuário e nos negócios:

- Avaliar o grau em que o requisito afeta positivamente a experiência do usuário e atende às necessidades das partes interessadas.
- Priorizar requisitos que tenham um impacto significativo na satisfação do usuário, na eficiência operacional e nos objetivos de negócio.

Custo e esforço de desenvolvimento:

- Estimar o custo e o esforço necessários para implementar cada requisito, incluindo desenvolvimento, teste e integração.
- Priorizar requisitos que possam ser implementados com um esforço e custo relativamente baixos, gerando um retorno rápido sobre o investimento (ROI).

Riscos e incertezas:

- Identificar riscos e incertezas associados a cada requisito, como a complexidade técnica, a disponibilidade de recursos e a dependência de outros requisitos ou componentes.
- Priorizar requisitos com riscos e incertezas mais baixos, a fim de minimizar os possíveis impactos negativos no projeto e garantir a entrega dentro dos prazos e do orçamento estabelecidos.

Interdependências e restrições:

- Analisar as interdependências entre os requisitos e identificar quaisquer restrições, como a necessidade de cumprir regulamentações ou a dependência de recursos externos.
- Priorizar requisitos que possam ser implementados independentemente ou que sejam fundamentais para a realização de outros requisitos de alta prioridade.

Para facilitar o processo de priorização, pode-se utilizar técnicas e ferramentas como a **matriz de priorização**, o método **MoSCoW** (Must-have, Should-have, Could-have, Won't-have) ou o método **Analytic Hierarchy Process** (AHP) (Karlsson, 1995; Leffingwell, 2011). O envolvimento das partes interessadas e dos usuários é essencial para garantir que suas necessidades e expectativas sejam consideradas ao longo do processo e que a priorização reflita os objetivos e valores do projeto.

Em conclusão, a priorização de requisitos é uma etapa crítica no gerenciamento de projetos de software, que permite a alocação eficiente de recursos e a maximização do valor entregue. Ao seguir um processo sistemático e baseado em critérios objetivos, é possível tomar decisões informadas e garantir o sucesso do projeto e a satisfação dos usuários e das partes interessadas.

A matriz MoSCoW é uma técnica de priorização de requisitos que classifica os requisitos em quatro categorias: Must-have (M), Should-have (S), Could-have (C) e Won't-have (W). Abaixo está um exemplo de matriz MoSCoW para um projeto de sistema de gerenciamento de pedidos de uma loja online:

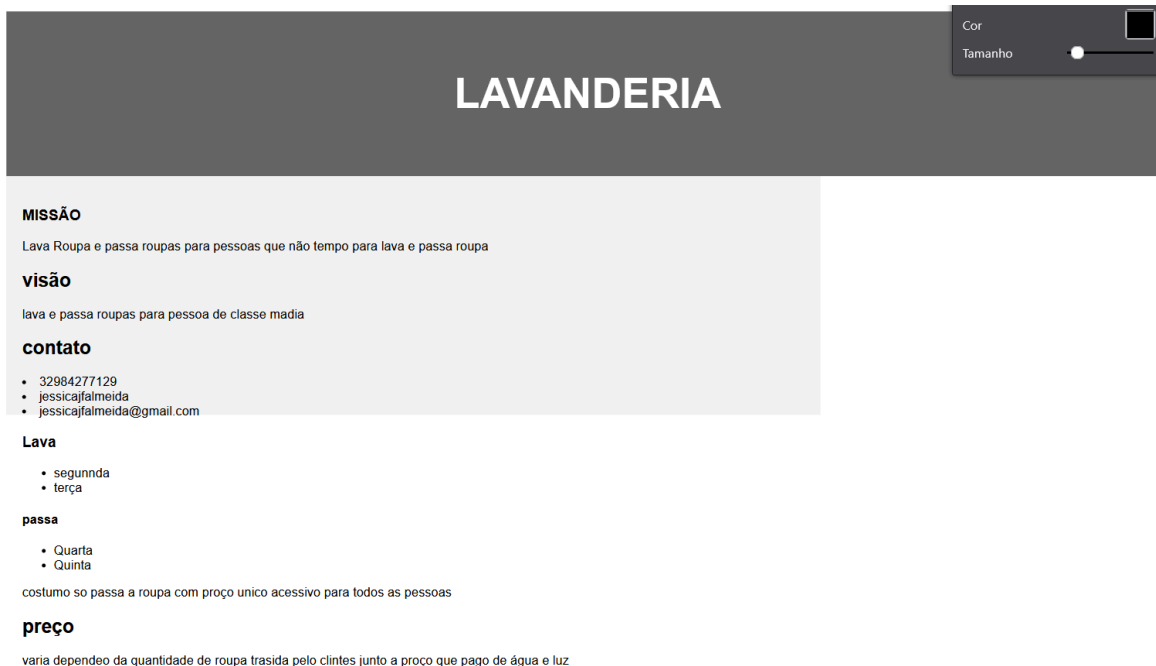


imagem 2 fonte autor

Neste exemplo, a matriz MoSCoU é organizada em uma tabela com as quatro categorias de priorização, e os requisitos são listados em cada categoria de acordo com sua importância para o projeto:

- **Must-have (M):** Requisitos críticos que são essenciais para o funcionamento básico do sistema e sem os quais o projeto não pode ser considerado bem-sucedido.
- **Should-have (S):** Requisitos importantes que agregam valor ao sistema e melhoram significativamente a experiência do usuário, mas cuja ausência não impede o funcionamento básico do sistema.
- **Could-have (C):** Requisitos desejáveis que podem melhorar a experiência do usuário ou a eficiência operacional, mas não são essenciais para o funcionamento do sistema e podem ser adiados ou implementados em uma versão futura.

- Won't-have (W): Requisitos de baixa prioridade que não serão implementados no escopo atual do projeto, mas que podem ser reconsiderados em atualizações futuras ou em projetos separados.

O código ajuda a equipe do projeto a entender as prioridades e a focar nos requisitos críticos, garantindo que os recursos e o tempo sejam alocados de forma eficiente e que o projeto atenda às necessidades e expectativas dos usuários e das partes interessadas.

2.6.11) Validação

O objetivo da validação do site de Inspeção de Segurança de serviço de Frota (SISV) é garantir que o sistema atenda às expectativas e necessidades das partes interessadas, incluindo requisitos funcionais e não funcionais. A validação envolve a verificação da conformidade do sistema com os requisitos especificados e a avaliação da adequação do sistema ao seu propósito pretendido.

Abordagem: A validação do SISV será realizada por meio de uma combinação de técnicas de validação, incluindo revisões de requisitos, inspeções formais, testes e demonstrações do sistema. Essas atividades de validação serão conduzidas em várias fases do projeto, envolvendo as partes interessadas para garantir a qualidade e a conformidade do sistema.

Revisões de requisitos: As revisões de requisitos serão realizadas com as partes interessadas para verificar se os requisitos documentados são completos, corretos, consistentes e compreensíveis. Durante as revisões, as partes interessadas serão encorajadas a fornecer feedback e identificar possíveis lacunas, ambiguidades ou inconsistências nos requisitos.

Inspeções formais: As inspeções formais envolvem a revisão detalhada dos artefatos do projeto, como código-fonte, casos de teste e documentação do usuário, para verificar se estão em conformidade com os requisitos especificados e as

normas de qualidade. As inspeções formais serão realizadas por membros da equipe do projeto e especialistas independentes, conforme necessário.

Testes do sistema: Os testes do site serão realizados para verificar se o SISV atende aos requisitos funcionais e não funcionais. Os testes incluirão testes de unidade, integração, sistema e aceitação do usuário. Os casos de teste serão desenvolvidos com base nos requisitos documentados e nas especificações de teste, e os resultados dos testes serão registrados e analisados para identificar defeitos e áreas de melhoria.

Demonstrações do site: As demonstrações do sistema serão realizadas com as partes interessadas para obter feedback sobre a funcionalidade, usabilidade e desempenho do SIS.

Durante as demonstrações, os usuários poderão interagir com o sistema e avaliar sua adequação às suas necessidades e expectativas.

Critérios de aceitação: Os critérios de aceitação do SIS incluem a conformidade com os requisitos funcionais e não funcionais documentados, a aprovação das partes interessadas com base em revisões de requisitos e demonstrações do sistema, e a conclusão bem-sucedida dos testes do sistema. Além disso, o sistema deve atender aos critérios de desempenho, usabilidade, segurança e escalabilidade estabelecidos nos requisitos não funcionais.

Responsabilidades: A equipe do projeto será responsável por realizar as atividades de validação e garantir a qualidade e a conformidade do SIS. As partes interessadas, incluindo usuários, gerentes, desenvolvedores e outros envolvidos no projeto, serão responsáveis por fornecer feedback, participar das revisões de requisitos e demonstrações

2.6.11) Gerenciamento de mudanças

O gerenciamento de mudanças é um componente essencial do desenvolvimento de site, pois permite que o projeto se adapte a novos requisitos, condições e restrições (Lehman, 1980). Um modelo eficaz de gerenciamento de

mudanças garante que as mudanças nos requisitos, no código e na documentação sejam devidamente controladas, comunicadas e implementadas, minimizando o impacto negativo no projeto e garantindo a qualidade e a conformidade do sistema (Boehm, 1981).

2.6.12) Processo de Gerenciamento de Mudanças

O processo de gerenciamento de mudanças consiste nas seguintes etapas (Rajlich & Bennett, 2000):

Identificação da mudança: As mudanças podem ser identificadas por membros da equipe do projeto, partes interessadas ou outros envolvidos no projeto. As mudanças podem resultar de novos requisitos, feedback do usuário, mudanças nas condições do mercado ou avanços tecnológicos.

Avaliação da mudança: A mudança identificada deve ser avaliada em termos de impacto no projeto, incluindo o escopo, o cronograma, os recursos e os riscos associados. A avaliação deve considerar a necessidade e a prioridade da mudança, bem como os custos e benefícios envolvidos.

Aprovação da mudança: A mudança proposta deve ser aprovada por um comitê de controle de mudanças ou por partes interessadas relevantes antes de ser implementada. O comitê pode aprovar, rejeitar ou solicitar mais informações sobre a mudança.

Implementação da mudança: Se a mudança for aprovada, a equipe do projeto deve implementá-la, modificando os requisitos, o código, a documentação e outros artefatos do projeto conforme necessário. A implementação da mudança deve ser planejada e gerenciada para minimizar o impacto negativo no projeto e garantir a qualidade e a conformidade do sistema.

Comunicação da mudança: A mudança aprovada e implementada deve ser comunicada a todas as partes interessadas e envolvidos no projeto, incluindo desenvolvedores, testadores, usuários e gerentes. A comunicação deve incluir informações sobre a natureza da mudança, o impacto no projeto e as ações necessárias para se adaptar à mudança.

Monitoramento e controle da mudança: A equipe do projeto deve monitorar e controlar o processo de gerenciamento de mudanças, garantindo que as mudanças sejam implementadas de forma eficaz e que os impactos negativos no projeto sejam minimizados. O monitoramento e controle incluem a análise de métricas, a identificação de problemas e a implementação de ações corretivas conforme necessário.

2.6.13) Ferramentas e Técnicas de Gerenciamento de Mudanças

O gerenciamento de mudanças pode ser facilitado por meio do uso de ferramentas e técnicas apropriadas. Essas ferramentas e técnicas auxiliam na organização, rastreamento e comunicação das mudanças no projeto, garantindo que todas as partes interessadas estejam atualizadas e envolvidas no processo.

Sistemas de Controle de Versão: Os sistemas de controle de versão, como Git, SVN ou ClearCase, são fundamentais para rastrear e gerenciar mudanças no código-fonte, documentação e outros artefatos do projeto. Essas ferramentas permitem que os desenvolvedores colaborem efetivamente, revertam alterações indesejadas e rastreiem o histórico de mudanças ao longo do tempo.

Sistemas de Rastreamento de Problemas e Solicitações de Mudança: Ferramentas como Jira, Bugzilla ou Mantis ajudam a gerenciar e rastrear problemas, solicitações de mudança e tarefas relacionadas ao projeto. Esses sistemas permitem que os membros da equipe colaborem na resolução de problemas e na implementação de mudanças aprovadas, garantindo que todas as mudanças sejam atribuídas, rastreadas e documentadas.

Software de Gerenciamento de Projetos: Ferramentas de gerenciamento de projetos, como Microsoft Project, Smartsheet ou Asana, podem ser usadas para planejar e monitorar o andamento do projeto, incluindo a implementação de mudanças. Essas ferramentas fornecem recursos de planejamento, alocação de recursos e acompanhamento de progresso, o que permite gerenciar efetivamente o impacto das mudanças no cronograma e nos recursos do projeto.

Revisões e Inspeções: As revisões de código e as inspeções formais são técnicas que podem ser usadas para validar a qualidade das mudanças implementadas e garantir a conformidade com os requisitos e padrões do projeto. Essas práticas envolvem a revisão detalhada dos artefatos do projeto por membros da equipe e especialistas independentes, conforme necessário.

Integração Contínua e Entrega Contínua (CI/CD): A automação do processo de compilação, teste e implantação por meio da integração contínua e da entrega contínua ajuda a garantir que as mudanças sejam validadas e implantadas de maneira eficiente e consistente. Ferramentas como Jenkins, GitLab CI ou Travis CI podem ser usadas para configurar pipelines de CI/CD, o que facilita a detecção e correção de problemas antes que eles cheguem à produção.

Comunicação e Colaboração: A comunicação eficaz é fundamental para gerenciar as mudanças no projeto. Ferramentas de comunicação e colaboração, como Slack, Microsoft Teams ou Trello, ajudam a manter as partes interessadas informadas e envolvidas no processo de gerenciamento de mudanças. Essas ferramentas fornecem um espaço centralizado para compartilhar informações, discutir problemas e tomar decisões relacionadas às mudanças no projeto.

3. ENCERRAMENTO DO PROJETO

Os site web representam um complexo conjunto de processos interativos e transacionais implementados na infraestrutura da internet, possibilitando a configuração e a implementação de soluções digitais multifacetadas, tais como portais online, aplicativos web e outros serviços similares. Esta infraestrutura é construída sobre uma arquitetura que, em sua essência, é bifurcada em duas categorias fundamentais: o **back-end** e o **front-end**.

O **front-end**, frequentemente referido como a camada de apresentação, constitui a interface do usuário, responsável por captar as interações do usuário e apresentar os resultados das operações requisitadas. Essa camada emprega predominantemente tecnologias como HTML (**HyperText Markup Language**), CSS (**Cascading Style Sheets**) e **JavaScript**, para criar interfaces gráficas interativas. Por outro lado, o **back-end**, frequentemente denominado como a camada de

processamento ou servidor, constitui a infraestrutura que opera nos bastidores, encarregada da gestão de dados e da implementação das regras de negócios. Este recebe as solicitações vindas do **front-end**, processa-as conforme as regras de negócios estabelecidas e devolve os resultados pertinentes. O **back-end** utiliza uma variedade de bancos de dados e linguagens de programação, como HTML , css , javascript,

A bifurcação entre **front-end** e **back-end** é estrategicamente delineada por razões pragmáticas e estruturais. Em primeiro lugar, esta divisão possibilita a separação de responsabilidades, permitindo que os desenvolvedores possam se especializar e focar em áreas particulares de competência. Em segundo lugar, facilita a manutenção e a escalabilidade do sistema, visto que as alterações podem ser implementadas em um segmento sem interferir no outro. Adicionalmente, esta separação estratégica proporciona maior segurança, pois as informações sensíveis, tais como detalhes de transações e dados dos usuários, são preservadas no lado do servidor, onde podem ser adequadamente resguardadas. Desta forma, a divisão entre **back-end** e **front-end** é um pilar crucial para a configuração de sistemas web eficientes, robustos e seguros.

Para o desenvolvimento lançou-se mão de ferramentas de renome e grande aceitação no mercado visando obter suporte e disponibilidade de servidores durante a implantação evitando assim a utilização de frameworks. Como linguagens de **front-end** foram utilizadas HTML, CSS e JavaScript. Para atuar no **back-end** utilizou-se o PHP. Como banco de dados o MySQL foi o escolhido, devido a sua praticidade e aceitação no mercado.

Por questões que extrapolam o conteúdo discutido em sala de aula até o momento, não utilizamos nenhum padrão de projeto.

3.1) MVP: Produto Mínimo Viável

O Produto Mínimo Viável (MVP, do inglês "**Minimum Viable Product**") é um conceito fundamental no desenvolvimento ágil de software, definido como a versão mais simples de um produto que ainda consegue entregar valor aos usuários finais e clientes (Ries, 2011). Segundo Blank (2013), MVP é o meio pelo qual as empresas

podem testar suas hipóteses de negócios, medindo como os usuários interagem com o produto, para depois iterar e refinar o produto com base em feedbacks reais.

O MVP é de grande importância no desenvolvimento de software por uma série de razões. Segundo Maurya (2012), permite que as equipes de desenvolvimento validem as hipóteses mais críticas sobre o produto e o mercado, minimizando assim o risco de construir um produto que ninguém deseja. Além disso, o MVP auxilia na otimização dos recursos de desenvolvimento, pois as equipes podem evitar a construção de funcionalidades desnecessárias, focando apenas naquelas que oferecem o maior valor para os usuários.

Portanto, o MVP é uma ferramenta poderosa que equilibra as necessidades de entrega rápida, aprendizado validado, e gestão de risco efetiva. Ele permite que as empresas avancem rapidamente, aprendam com os usuários reais, e ajustem suas estratégias de produto com base em evidências, em vez de suposições.

3.2) Apresentação do MVP à empresa parceira

A equipe do Laboratório de Transformação Digital (LTD) promoveu no dia 24 de maio de 2023, às 14:30 horas, uma reunião multidisciplinar envolvendo gerentes das áreas de Logística, Financeiro e Recursos Humanos. O objetivo central do encontro era apresentar o software recém-desenvolvido, expondo suas funcionalidades distintas e destacando as peculiaridades operacionais que permitem a integração harmoniosa entre os diferentes departamentos.

Participaram dessa reunião Adilson Duarte Paradinha, Gerente de Operações de Distribuição (GOD), Luciene Maria Brandão de Giacomo e Barros, Gerente Administrativo/Financeiro, Júlia Ferraz, Analista sênior do setor de Gente, Anderson Barboza da Cruz, professor designado, Clara Rocha Ferreira, aluna membro do LTD, Franciele Nascimento Araújo, aluna membro do LTD, Jéssica Torres Bezerra, aluna membro do LTD, e Samuel Moisés Guimarães, aluno membro do LTD.

Durante a apresentação, foram detalhados os componentes específicos do site web, incluindo a metodologia de uso, entradas e saídas de dados, e o funcionamento dos formulários. Esta fase da demonstração possibilitou um entendimento aprofundado sobre o mecanismo do software, proporcionando aos

presentes a oportunidade de avaliar suas funcionalidades à luz de suas respectivas necessidades departamentais.

3.2.1) Detalhamento do software a empresa parceira

Na apresentação do software, a primeira tela evidenciada foi a tela de login. Esta é responsável por garantir a autenticação do usuário, um componente essencial para a segurança do sistema. A funcionalidade foi detalhadamente explicada, incluindo os procedimentos de autenticação, restrições de acesso e o redirecionamento subsequente. Esse elemento permite que os usuários acessem as áreas específicas do sistema pertinentes às suas respectivas funções e responsabilidades.

A segunda tela apresentada foi a tela principal. Esta serve como uma espécie de painel de controle, onde os usuários podem navegar para as diferentes funções e recursos do software. A tela principal desempenha um papel significativo na determinação da usabilidade e eficiência do software, proporcionando uma visão geral e facilitando o acesso às diferentes partes do sistema.

Finalmente, foram demonstradas as telas de cadastro de veículos, usuários e checklists. Estas telas são dedicadas à entrada de novos dados no sistema, desempenhando um papel crucial na manutenção e atualização de informações. Além disso, também foram apresentadas as funcionalidades de saída de dados, que incluem a visualização das avaliações realizadas e a disponibilização de informações gerais. Estas funções garantem que os usuários possam acessar e interpretar os dados armazenados de maneira eficaz, promovendo uma gestão de dados mais eficiente e orientada a resultados.

A reunião concluiu-se com a aprovação unânime da solução proposta pelo Laboratório de Transformação Digital. Todos os participantes expressaram sua satisfação com o software apresentado, destacando positivamente não apenas as funcionalidades, mas também a estética da solução, incluindo layout e esquema de

cores. Tal consenso evidencia o alinhamento entre o produto desenvolvido e as expectativas e requisitos dos diversos stakeholders envolvidos.

3.3) Particularidades do projeto

O projeto em questão se configura em duas frentes distintas: uma de caráter público e outra de natureza privada. A seção pública do projeto foi disponibilizada na plataforma GitHub(<https://github.com/Samuelmgm/taruma>), que serviu como um ambiente de colaboração eficaz para os alunos, permitindo-lhes compartilhar e gerenciar as versões mais recentes do projeto.

Por outro lado, a parte privada do projeto está confinada às instalações da empresa e contém informações de natureza sensível. Esses dados sensíveis são rigorosamente protegidos e não foram disponibilizados aos alunos. Essa medida de segurança foi implementada para garantir a conformidade com a Lei Geral de Proteção de Dados (LGPD, Lei nº 13.709), promulgada pelo governo brasileiro em 2018.

A LGPD estabelece diretrizes claras sobre o tratamento de dados pessoais, garantindo a privacidade e a segurança desses dados em todo o território brasileiro (Brasil, Lei nº 13.709, de 14 de agosto de 2018). Assim, o resguardo de dados sensíveis dentro do âmbito corporativo, tal como implementado neste projeto, constitui uma prática de conformidade crucial para salvaguardar a privacidade e os direitos dos titulares de dados.

3.1. Relato Coletivo:

Considerações do grupo sobre o atingimento dos objetivos sociocomunitários estabelecidos para o projeto.

3.1.1. Avaliação de reação da parte interessada

O **feedback**, no contexto do desenvolvimento de software, pode ser conceituado como um conjunto de observações, comentários ou avaliações, destinados a apontar áreas de sucesso ou melhorias necessárias em um produto ou processo. Este feedback pode ser proveniente de várias fontes, incluindo usuários

finais, membros da equipe de desenvolvimento, partes interessadas ou ferramentas de software automatizadas. As formas de coletar feedback podem variar desde entrevistas individuais, pesquisas, grupos focais, sessões de testes de usabilidade, até análises automatizadas de dados de uso do software.

No âmbito do desenvolvimento de site , especialmente após a apresentação de um Produto Mínimo Viável (MVP), o **feedback** é de suma importância. O MVP representa a versão mais simplificada de um produto que pode ser lançada para o mercado com o intuito de validar as hipóteses fundamentais do negócio. A partir desse ponto, o **feedback** torna-se um instrumento essencial para a iteração contínua e o refinamento do produto. Ele permite identificar rapidamente quais características são valorizadas pelos usuários, que precisam ser modificadas e quais podem ser descartadas.

Além disso, pode fomentar a melhoria contínua, aumentar a satisfação do usuário e fortalecer a posição competitiva do produto no mercado. Portanto, no contexto da engenharia de software, a cultura de feedback deve ser estimulada e mantida, uma vez que sua falta pode levar a uma série de problemas, como desenvolvimento ineficiente, baixa adoção do usuário e, eventualmente, falha do produto.

Os membros da Empresa parceira se mostraram entusiasmados não somente pelo software, mas com a integração empresa-escola, estreitando os laços entre a academia e o mercado de trabalho transformando a empresa em uma extensão do laboratório de transformação digital.

Ao término da apresentação, todos presentes fizeram elogios aos alunos e à Estácio se mostrando gratos pela iniciativa. Foram solicitadas algumas alterações no software:

- 1) Mudar cor de fundo para azul;
- 2) Mudar título da aplicação para gerenciamento de 5s, por questões legais;
- 3) Entrega do projeto em arquivo zipado a ser enviado para o time de ti para validação e alteração das regras conforme necessário.
- 4) Que dêmos continuidade em um segundo momento do LTD;

Todos os alunos se mostraram gratos pela vivência obtida do decorrer do processo, mesmo com as limitações impostas pelo tamanho da empresa, normas, regras de segurança internas, leis e regulamentos. Tais eventos mostraram os verdadeiros desafios do desenvolvimento e reafirmaram que o analista e desenvolvedor de site tem compromisso com o cliente final não com o site. Estamos falando de vidas e não de site . Estamos falando da segurança do condutor e sua equipe e de todos os transeuntes envolvidos no ecossistema do trânsito brasileiro.

Como forma reafirmar o feedback, lhes foi enviado uma cópia deste documento solicitando a aprovação ou não do sistema. Todas as vertentes envolvidas leram o documento em sua plenitude e assinaram a carta de aprovação:

- Análise e desenvolvimento de site , do aluna curso de Análise e Desenvolvimento de SiTE, representando o LTD.

3.2. Relato de Experiência Individual

3.2.1. CONTEXTUALIZAÇÃO

3.2.2. METODOLOGIA

.

3.2.3. RESULTADOS E DISCUSSÃO:

.

3.2.4. REFLEXÃO APROFUNDADA

.

3.2.5. CONSIDERAÇÕES FINAIS

A análise detalhada dos requisitos funcionais e não funcionais do site de inspeção de segurança, juntamente com a coleta de informações e expectativas dos stakeholders, forneceu uma base sólida para o desenvolvimento do projeto. Através das entrevistas e discussões com os membros da equipe de segurança do trabalho, responsáveis pela logística, gerências comercial e financeira, e o time de tecnologia

da informação, foi possível identificar as necessidades e os objetivos específicos do sistema, bem como as restrições e os recursos disponíveis.

Com a conclusão bem-sucedida do projeto de desenvolvimento do sistema, podemos avaliar que o produto foi bem recebido pela empresa parceira. Isto não só valida a eficácia das técnicas e estratégias implementadas durante o ciclo de vida do desenvolvimento, mas também reforça a relevância do alinhamento contínuo entre as expectativas do cliente e a entrega. A aceitação positiva do sistema por parte da empresa parceira evidencia que o projeto conseguiu cumprir o seu objetivo, contribuindo de maneira significativa para a melhoria de seus processos.

Com base na experiência adquirida neste projeto, recomenda-se para futuros projetos de sistemas web a implementação do padrão de projeto Model-View-Controller (MVC). Este é um padrão de arquitetura de software que divide a aplicação em três componentes interconectados: o Modelo, que representa a estrutura de dados e as regras de negócio; a Visão, que é a representação visual dos dados, ou seja, a interface do usuário; e o Controlador, que é o intermediário entre o Modelo e a Visão, processando todas as entradas do usuário e atualizando a Visão de acordo.

A adoção do padrão MVC pode proporcionar vários benefícios. Primeiramente, ao separar as preocupações de apresentação e lógica de negócios, o MVC melhora a organização do código, facilita a manutenção e aumenta a flexibilidade do site. Além disso, este padrão favorece o desenvolvimento paralelo, permitindo que diferentes partes do sistema sejam desenvolvidas simultaneamente por equipes distintas. Finalmente, a utilização do MVC pode melhorar a eficiência no processo de testes, já que cada componente pode ser testado independentemente. Portanto, o padrão de projeto MVC se apresenta como uma proposta valiosa para otimizar a qualidade e a eficácia de futuros projetos de sistemas web.

É imperativo destacar que o sucesso deste projeto se deve em grande parte ao papel determinante desempenhado pela equipe de desenvolvimento, composta majoritariamente por mulheres, representando 75% do time. Esta constituição vai na contramão do cenário típico na indústria de tecnologia, que é notoriamente dominada por homens. A participação substancial das mulheres neste projeto desafiou as estereótipos de gênero prevalecentes e demonstrou o poder e a

capacidade das mulheres em entregar um sistema complexo e eficiente. A sua competência técnica, capacidade de resolver problemas e compromisso com a excelência foram fundamentais para a execução bem-sucedida do projeto.

O valor que as mulheres agregaram a este projeto vai além das suas habilidades técnicas. Elas trouxeram uma diversidade de perspectivas que enriqueceram o processo de tomada de decisões e o design do sistema. Diversos estudos têm mostrado que equipes diversificadas são mais inovadoras e eficazes na resolução de problemas complexos. Esta diversidade de experiências e pontos de vista, aliada à excelência técnica das mulheres na equipe, culminou na entrega de um sistema que excedeu as expectativas do cliente. Este projeto é um testemunho do valor que as mulheres trazem para o campo da tecnologia e um exemplo a ser seguido em futuros projetos..

ANEXO I: Carta de apresentação

CARTA DE APRESENTAÇÃO	
<p>Vimos por desta apresentar o grupo de acadêmicos do CENTRO UNIVERSITÁRIO ESTÁCIO DE JUIZ DE FORA listados na tabela a final deste documento, a fim de convidá-lo a participar de uma atividade extensionista associada à disciplina Tópicos de Big Data em Python, sob responsabilidade do Prof. Anderson Barboza da Cruz.</p> <p>Em consonância ao Plano Nacional de Educação e demais normativas educacionais vigentes, o CENTRO UNIVERSITÁRIO ESTÁCIO DE JUIZ DE FORA desenvolve atividade extensionista que, norteados pela metodologia de aprendizagem baseada em projetos, tem por princípios fundantes o diagnóstico dos problemas/demandas/necessidades, a participação ativa dos interessados/públicos participantes, a construção dialógica, coletiva e experiencial de conhecimentos, o planejamento de ações, o desenvolvimento e avaliação das ações, a sistematização dos conhecimentos, a avaliação das ações desenvolvidas.</p> <p>Nesse contexto, a disciplina acima mencionada tem como principal escopo os temas relacionados à desenvolvimento de sistemas para análise de dados e Big Data.</p> <p>Sendo assim, pedimos o apoio dessa organização/entidade/coletivo/associação/outra, que aqui chamaremos de parte interessada, para a</p>	<p>Como se trata de atividade de ensino/aprendizagem de caráter extensionista, prevista no Projeto Pedagógico do Curso, salientamos que:</p> <ul style="list-style-type: none">• não há cobrança de remuneração de qualquer natureza por parte do CENTRO UNIVERSITÁRIO ESTÁCIO DE JUIZ DE FORA, seus alunos ou o docente da disciplina, à parte interessada;• as atividades desenvolvidas no âmbito do projeto extensionista não configuram relação de trabalho entre os alunos e o docente do CENTRO UNIVERSITÁRIO ESTÁCIO DE JUIZ DE FORA, disciplina Tópicos de Big Data em Python, e a parte interessada;• os resultados do projeto só poderão ser implantados para uso efetivo mediante Anotação de Responsabilidade Técnica de um profissional habilitado;• os resultados do projeto podem ser implantados pela parte interessada para fins lucrativos, sem a necessidade de pagamento de quaisquer benefícios aos alunos, ac

ANEXOII: Carta de autorização

10:03

informações que o(s) aluno(s) poderá(ão) ter acesso. Desde já nos colocamos à sua disposição para quaisquer esclarecimentos. <INFORMAÇÕES DE CONTADOS DOS MEMBROS DO GRUPO E DO DOCENTE>.

Grupo de Alunos

<Jéssica Almeida de Oliveira
Matrícula: <202202146421>

<NOME COMPLETO DO ALUNO 2>
Matrícula: <MATRÍCULA DO ALUNO 2>

<NOME COMPLETO DO ALUNO 3>
Matrícula: <MATRÍCULA DO ALUNO 3>

<NOME COMPLETO DO ALUNO 4>
Matrícula: <MATRÍCULA DO ALUNO 4>

<NOME COMPLETO DO ALUNO 5>
Matrícula: <MATRÍCULA DO ALUNO 5>

osamente,

Vera

Anderson Barboza da Cruz
Tópicos de Big Data em Python
Semestre 2024.02
Matrícula: 10642068

Juiz de Fora, Agosto de 2024

ANEXO IV: FOLHA DE APROVAÇÃO

Criar um site para um lavanderia para melhor atender os clientes sendo um site fácil de mexe

1.1 Objetivos desta especificação

Criar um site para um pessoa solicitar o serviço

Lava Roupa e passa roupas para pessoas que não tempo para lava e passa roupa

1.2 Escopo do sistema

1.2.1 Nome do sistema e de seus componentes principais

Sistema para um lavadeira e passadeira pode oferecer seus serviço

1.2.2 Missão do sistema

Trazer mais informações para dentro do serviço do meu parceiro

lava e passa roupas para pessoa de classe média

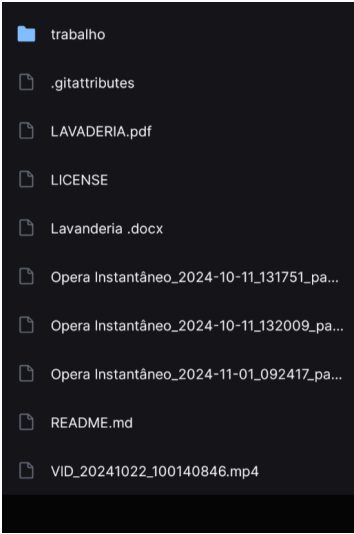
1.2.3 Limites do softwares

a iteia e cria um site para pode atender melhor os meus clientes sente site fácil

Visão sistema

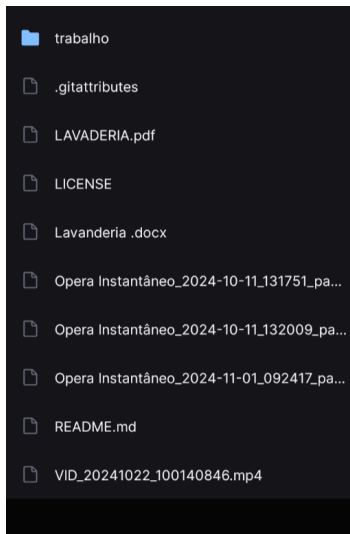
Oferecer meus serviços para maior número de cliente possíveis para melhor atender

ANEXO V: REPOSITÓRIO



ANEXO VI: MANUAL DE INSTRUÇÕES

Ui



REFERÊNCIAS

CHUNG, L.; NIXON, B. A.; YU, E.; MYLOPOULOS, J. **Non-Functional Requirements in Software Engineering**. Springer, 2000.

SOMMERVILLE, I. **Engenharia de software**. 10. ed. Pearson, 2018.

WIEGERS, K. E.; BEATTY, J. **Software Requirements**. 3. ed. Microsoft Press, 2013.

KARLSSON, J. A Systematic **Approach to Software Requirements Prioritization**. In: Proceedings of the Third International Workshop on Requirements Engineering: Foundations for Software Quality (REFSQ '95), 1995.