

Projeto Integrador Transdisciplinar em Engenharia de Software I

Nome: Leonardo Vinicius Berilo Brito

RGM: 27844641

Qual situação-problema você escolheu para criar o seu projeto de intervenção?

☐ Situação-problema 1

A Situação-Problema escolhida:

Refere-se à disparidade entre os requisitos desejados e a implementação final do software. Tal questão está ligada à precisão e clareza na definição e comunicação dos requisitos, os quais podem ocasionar uma série de imperfeições e falhas no produto concluído.

Objetivo:

Assegurar que as especificações dos requisitos sejam precisas, claras e seguidas à risca, prevenindo desacordos entre o que foi solicitado e o que é entregue no software final, cumprindo, dessa forma, as demandas dos usuários.

Revisão de Conceito:

- Engenharia de requisitos
- Modelagem de negócios
- Gestão de projetos de software
- Desenvolvimento de software
- Análise de impacto
- Gerenciamento de mudanças
- Garantia de qualidade e habilidades interpessoais.

Metodologia:

- Levantamento de Requisitos: Realização de entrevistas, workshops e análise de documentos.
- Documentação e Análise de Requisitos: Utilização de modelos padronizados, casos de uso e diagramas de processos.
- Confirmação e validação de Requisitos: Elaboração de protótipos, revisões formais e definição de critérios de aceitação.
- Administração de Mudanças: Monitoramento de versões e análise de impacto.
- Abordagem Ágil no Desenvolvimento: Implementação de Scrum, Kanban e métodos iterativos.
- Testes Automatizados: Emprego de TDD, BDD e automação de testes.
- Foco na Comunicação e Colaboração Constante.

Tempo:

- Levantamento de Requisitos: 3-4 semanas.
 - Análise e Registro de Requisitos: 3-5 semanas.
 - Confirmação e Avaliação de Requisitos: 3-4 semanas.
 - Gestão de Requisitos: 2 semanas.
 - Desenvolvimento Iterativo: 5-7 semanas.
 - Testes Automatizados: 5 semanas.
 - Diálogo e Parceria Constantes: 1 semana.
- Total Previsto: 22-28 semanas (aproximadamente de 5 a 7 meses).

Procedimento e material utilizado:

- Levantamento de Requisitos (Métodos de Levantamento: Empregando entrevistas, reuniões, sessões de brainstorming, questionários e análise de documentação atual para identificar os requisitos de todos os envolvidos; Mapeamento de Partes Interessadas: Identificando todas as partes envolvidas e delineando suas necessidades e expectativas em relação ao software).
- Análise e Registro de Requisitos (Detalhando as Especificações: Registrando os requisitos por meio de modelos padronizados como o IEEE 830. Incluindo requisitos funcionais, não funcionais e regras de negócio; Elaboração de Casos de Uso: Utilizando diagramas de casos de uso para representar as interações entre os usuários e o sistema; Representação de Processos: Desenvolvimento de diagramas de fluxo de processos (BPMN) para compreender e documentar os processos de negócio.).
- Confirmação e Comprovação de Requisitos (Prototipagem: Elaborar protótipos ou wireframes para visualizar os requisitos e validar junto aos interessados; Avaliações Formais: Realizar sessões de revisão de requisitos com todas as partes envolvidas para garantir a clareza e abrangência dos requisitos; Critérios de Aprovação: Estabelecer critérios de aceitação bem definidos para cada requisito, facilitando a validação objetiva dos mesmos).
- Gerência de Requisitos (Controle de Versões: Empregando ferramentas de gerenciamento de requisitos (como Jira ou Confluence) para monitorar e controlar alterações nos requisitos; Análise de Impacto: Realizando análises de impacto para compreender as repercussões das alterações nos requisitos no projeto).
- Utilização de Metodologias Ágeis: A aplicação de metodologias ágeis como Scrum ou Kanban é adotada para o desenvolvimento do software de maneira iterativa e incremental, o que possibilita a realização de ajustes e melhorias contínuas nos requisitos.
- Test Automation: Elabore e preserve um conjunto de testes automatizados para assegurar a constante validação dos requisitos durante todo o processo de desenvolvimento.
- Permanente Comunicação e Colaboração (Ferramentas de Colaboração: Opte por utilizar plataformas como Slack, Microsoft Teams ou outros sistemas de comunicação para garantir uma comunicação constante e transparente entre a equipe de desenvolvimento e os stakeholders; Reuniões Periódicas: Programe en-

Resultado e discussão:

A implementação do método proposto resolveu a questão da especificação de requisitos, certificando-se de que o software final entregue satisfazia as expectativas dos utilizadores e os requisitos estabelecidos. A utilização de técnicas de elicitação, documentação, validação, gestão de alterações, desenvolvimento ágil e testes automatizados desempenhou um papel crucial no êxito do projeto. A abordagem colaborativa e iterativa garantiu a entrega de um produto de excelência, perfeitamente alinhado com as exigências empresariais.

Os requisitos estão claros e completos graças a uma cuidadosa elicitação e documentação. Todas as partes interessadas compreenderam e definiram os requisitos de forma transparente. A comunicação contínua e colaborativa entre os stakeholders e a equipe de desenvolvimento evitou mal-entendidos e garantiu alinhamento constante. O controle rigoroso de versões e a análise de impacto ajudaram a gerenciar mudanças nos requisitos sem afetar o cronograma ou a qualidade do projeto. A metodologia ágil e os testes automatizados asseguraram que o software desenvolvido atendesse aos requisitos especificados e às expectativas dos usuários.

Comentários:

Registrar com detalhes e seguindo um padrão é essencial para prevenir ambiguidades e garantir que a implementação seja feita corretamente. Revisões periódicas e constante feedback dos interessados são fundamentais para identificar e solucionar eventuais problemas de forma ágil. Optar por uma abordagem ágil possibilita a adaptação do processo e dos requisitos quando necessário, assegurando uma entrega gradual que atende às demandas do negócio.

No futuro, é importante reservar tempo e recursos suficientes para assegurar que os requisitos sejam compreendidos e documentados de forma abrangente. Também é essencial manter uma comunicação aberta e constante entre todos os envolvidos no projeto, além de adotar metodologias ágeis para favorecer a adaptação e o aprimoramento contínuo. Dessa maneira, é possível garantir entregas incrementais de alta qualidade. A introdução de testes automatizados desde o início é fundamental para a qualidade do software e para agilizar a validação contínua dos requisitos, culminando no sucesso e qualidade do software.