Nome: Lucas Luz Zanetti

RGM: 27882152

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

Situação-problema 2

1. Sobre o que é a situação-problema escolhida por você?

Trata-se da diferença entre os requisitos desejados e a implementação final do software, uma questão relacionada à especificação e comunicação de requisitos, que pode resultar em defeitos e falhas no produto final.

1. Qual o principal objetivo a ser solucionado a partir da situação-problema escolhida?

Garantir que as especificações de requisitos sejam claras, compreensíveis e corretamente implementadas, eliminando as discrepâncias entre o solicitado e o entregue no software final, atendendo às necessidades dos usuários.

1. Quais os conhecimentos aprendidos no curso foram utilizados como base de estudo?

Engenharia de requisitos (análise, documentação, validação e verificação de requisitos), modelagem de negócios (processos, casos de uso e regras de negócios), gestão de projetos de software (gerenciamento de requisitos, comunicação e colaboração), desenvolvimento de software (metodologias ágeis, desenvolvimento orientado a testes, prototipagem e wireframing), análise de impacto, gerenciamento de mudanças, garantia de qualidade e habilidades interpessoais (comunicação e resolução de conflitos).

1. Qual abordagem, técnica ou processo utilizado para resolver a situação-problema escolhida?

- Elicitação de Requisitos: Entrevistas, workshops, análise de documentos.

- Análise e Documentação de Requisitos: Templates padronizados, casos de uso, diagramas de processos.

- Validação e verificação de Requisitos: Prototipagem, revisões formais, critérios de aceitação.

- Gerenciamento de Mudanças: Controle de versões, análise de impacto.

- Desenvolvimento Ágil: Scrum, Kanban, desenvolvimento iterativo.

- Testes Automatizados: TDD, BDD, automação de testes.

- Comunicação e colaboração contínua.

1. Quanto tempo levará para que a situação-problema possa ser resolvida? Aponte tempos específicos para cada ação e o tempo total para a resolução

- Elicitação de Requisitos: 3-4 semanas.

- Análise e Documentação de Requisitos: 3-5 semanas.

- Validação e Verificação de Requisitos: 3-4 semanas.

- Gerenciamento de Requisitos: 2 semanas.

- Desenvolvimento Ágil: 5-7 semanas.

- Testes Automatizados: 5 semanas.

- Comunicação e Colaboração Contínua: 1 semana.

Tempo Total Estimado: 22-28 semanas (aproximadamente 5 a 7 meses).

1. Como você pretende solucionar a situação-problema escolhida? Descreva as etapas e as ferramentas utilizadas nessa resolução.

- Elicitação de Requisitos

- Técnicas de Elicitação: Utilizar entrevistas, workshops, brainstorming, questionários e análise de documentos existentes para capturar os requisitos de todas as partes interessadas.

- Stakeholder Mapping: Identificar todas as partes interessadas e definir suas necessidades e expectativas em relação ao software.

- Análise e Documentação de Requisitos

- Especificações Detalhadas: Documentar os requisitos utilizando templates padronizados como o IEEE 830, incluindo requisitos funcionais, não funcionais e regras de negócio.

- Modelagem de Casos de Uso: Utilizar diagramas de casos de uso para capturar as interações entre os usuários e o sistema.

- Diagramas de Processos: Criar diagramas de fluxo de processos (BPMN) para entender e documentar os processos de negócio.

- Validação e Verificação de Requisitos

- Prototipagem: Desenvolver protótipos ou wireframes para visualizar os requisitos e validá-los com os stakeholders.

- Revisões Formais: Realizar sessões de revisão de requisitos com todas as partes interessadas para assegurar que os requisitos estão claros e completos.

- Critérios de Aceitação: Definir critérios de aceitação claros para cada requisito, permitindo a validação objetiva do mesmo.

- Gerenciamento de Requisitos

- Controle de Versões: Utilizar ferramentas de gerenciamento de requisitos (como Jira ou Confluence) para rastrear e gerenciar mudanças nos requisitos.

- Análise de Impacto: Realizar análises de impacto para entender as consequências das mudanças nos requisitos sobre o projeto.

- Metodologias Ágeis

- Adotar metodologias ágeis como Scrum ou Kanban para desenvolver o software de forma iterativa e incremental, facilitando ajustes e melhorias contínuas nos requisitos.

- Automação de Testes

- Desenvolver e manter um conjunto de testes automatizados para garantir que os requisitos sejam constantemente validados ao longo do desenvolvimento.

- Comunicação e Colaboração Contínua

- Ferramentas de Colaboração: Utilizar ferramentas como Slack, Microsoft Teams ou outros softwares de comunicação para manter uma comunicação contínua e transparente entre a equipe de desenvolvimento e os stakeholders.

- Reuniões Regulares: Agendar reuniões regulares (stand-ups, sprint planning, retrospectives) para alinhar expectativas e resolver problemas rapidamente.

1. Qual foi a resolução da situação-problema após a aplicação do procedimento? Faça também comentários finais sobre o projeto.

A aplicação do procedimento proposto permitiu resolver o problema de especificação de requisitos, garantindo que o software final atendesse às expectativas dos usuários e aos requisitos definidos. O uso de técnicas de elicitação, documentação, validação, gerenciamento de mudanças, desenvolvimento ágil e testes automatizados foi crucial para o sucesso do projeto. A abordagem colaborativa e iterativa assegurou a entrega de um produto de alta qualidade, alinhado às necessidades do negócio.

Comentários finais:

Documentar requisitos de forma detalhada e padronizada é essencial para evitar ambiguidades e garantir uma implementação correta. Revisões regulares e feedback contínuo dos stakeholders são vitais para identificar e corrigir problemas rapidamente. Adotar uma abordagem ágil permite ajustar o processo e os requisitos conforme necessário, garantindo uma entrega incremental e alinhada às necessidades do negócio. Além disso, é importante dedicar tempo e recursos suficientes para garantir que os requisitos sejam completamente entendidos e documentados, manter uma comunicação aberta e contínua entre todas as partes interessadas ao longo do projeto, utilizar metodologias ágeis para facilitar a adaptação e a melhoria contínua, garantir entregas incrementais e de alta qualidade, implementar testes automatizados desde o início para assegurar a qualidade do software, e facilitar a validação contínua dos requisitos, garantindo a qualidade e o sucesso do software.