

Exercício 1: Explique o que é a elicitação de requisitos e por que ela é uma etapa crucial no desenvolvimento de software

R:

Elicitação de requisitos é uma fase do projeto onde são extraídas informações do cliente sobre o que ele deseja que seja construído. É a fase em que o profissional de TI entende a necessidade do cliente e o orienta. É o momento de conversa com o usuário, de sentimento sobre o que este espera que seja entregue a ele. Na elicitação de requisitos são percebidas as necessidades do sistema e as características que esse sistema deve ter. Apesar de parecer uma etapa relativamente simples do projeto, não é. É nesse momento que o analista recebe a função de solucionar um problema que o cliente tem. Em muitas vezes, o cliente deseja um sistema mas não sabe exatamente o que quer, ou ele sabe bem o que quer mas não consegue passar as informações de uma forma compreensível.

Exercício 2: Liste e explique três métodos de elicitação de requisitos. Dê exemplos de situações em que cada método seria mais eficaz.

R:

1. Entrevistas

Descrição: Entrevistas são conversas estruturadas ou semi-estruturadas entre o analista de requisitos e os stakeholders. Durante as entrevistas, o analista faz perguntas para obter informações detalhadas sobre os requisitos e as necessidades do projeto.

Exemplo de Situação Eficaz:

Projeto: Desenvolvimento de um novo software para gerenciamento de projetos.

Situação: Se a organização possui vários departamentos com necessidades distintas (como financeiro, recursos humanos, e TI), entrevistas com representantes de cada departamento permitem coletar requisitos específicos e detalhados. Isso é útil para entender as necessidades e expectativas de diferentes áreas envolvidas no projeto.

2. Workshops

Descrição: Workshops são sessões colaborativas onde um grupo de stakeholders se reúne para discutir e definir requisitos. Os workshops são interativos e podem incluir técnicas como brainstorming, priorização e modelagem de processos.

Exemplo de Situação Eficaz:

Projeto: Redesign de um sistema de gestão de clientes (CRM).

Situação: Em um workshop com equipe de vendas, atendimento ao cliente e marketing, todos os stakeholders podem discutir conjuntamente o que funcionou e o que não funcionou no sistema atual. Isso promove uma abordagem colaborativa para identificar requisitos e prioridades, garantindo que as necessidades de todos os departamentos sejam abordadas.

3. Prototipagem

Descrição: A prototipagem envolve a criação de modelos preliminares (protótipos) do sistema ou produto para que os stakeholders possam interagir e fornecer feedback. Esses protótipos ajudam a esclarecer e refinar os requisitos ao demonstrar como o produto final funcionará.

Exemplo de Situação Eficaz:

Projeto: Desenvolvimento de um novo aplicativo móvel para um serviço de entrega.

Situação: Criar um protótipo interativo do aplicativo permite que os usuários finais experimentem as funcionalidades propostas, como rastreamento de pedidos e interface de usuário. Feedback direto dos usuários pode ser usado para ajustar e melhorar o design e os requisitos antes do desenvolvimento completo.

Exercício 3: Você foi designado para coletar requisitos para um novo sistema de gerenciamento de inventário. Elabore cinco perguntas que você faria ao cliente durante uma entrevista para entender melhor os requisitos do sistema.

R:

- 1 - O que você espera do projeto?
- 2 - Em quanto tempo você deseja o serviço?
- 3 - Quais são as principais necessidades e características?
- 4 - Aceitaria uma nova adaptação da sua ideia?
- 5 - Confia em nossa equipe inteiramente para solucionar todos os seus problemas?

Exercício 4: Descreva as vantagens e desvantagens de utilizar workshops de requisitos como método de elicitação. Quais são os principais fatores para garantir o sucesso de um workshop?

R:

Vantagens de Utilizar Workshops de Requisitos

Colaboração e Consenso:

Vantagem: Workshops permitem a colaboração direta entre diferentes stakeholders, o que facilita a construção de consenso e a resolução de conflitos. Ao discutir os requisitos em grupo, os participantes podem entender melhor as necessidades uns dos outros e encontrar soluções que atendam a todos.

Exemplo: Em um workshop para o desenvolvimento de um sistema de gerenciamento de projetos, representantes de diferentes departamentos podem negociar e alinhar suas necessidades, resultando em um sistema que melhor atende a todos os envolvidos.

Identificação Rápida de Requisitos:

Vantagem: Workshops podem ser uma maneira rápida e eficiente de identificar e documentar requisitos, pois permitem que várias informações sejam coletadas simultaneamente de diversos stakeholders.

Exemplo: Em um workshop, os participantes podem usar técnicas como brainstorming para gerar uma lista ampla de requisitos em um curto período, acelerando o processo de elicitación.

Feedback Imediato:

Vantagem: A interação direta durante o workshop proporciona feedback imediato sobre as ideias e propostas discutidas. Isso permite ajustes e refinamentos instantâneos.

Exemplo: Se um protótipo ou conceito é apresentado durante o workshop, os participantes podem fornecer feedback instantâneo sobre o design ou funcionalidade, permitindo ajustes antes que o projeto avance muito.

Desvantagens de Utilizar Workshops de Requisitos

Gerenciamento de Tempo:

Desvantagem: Workshops podem se alongar mais do que o previsto, especialmente se os participantes não forem bem gerenciados ou se houver muitos tópicos a serem discutidos.

Exemplo: Se o workshop envolver muitos stakeholders ou tópicos complexos, pode se tornar difícil manter o foco e o ritmo, levando a reuniões longas e pouco produtivas.

Conflito de Interesses:

Desvantagem: Com vários stakeholders participando, é possível que surjam conflitos de interesse e prioridades divergentes, tornando desafiador chegar a um consenso.

Exemplo: Se representantes de diferentes departamentos têm objetivos conflitantes, o workshop pode se tornar um campo de batalha em vez de uma colaboração produtiva.

Dependência da Facilitação:

Desvantagem: O sucesso do workshop depende fortemente da habilidade do facilitador em gerenciar discussões, manter o foco e garantir a participação efetiva de todos os envolvidos.

Exemplo: Um facilitador inexperiente pode falhar em moderar a discussão de maneira eficaz, resultando em desorganização e falta de resultados claros.

Principais Fatores para Garantir o Sucesso de um Workshop

Planejamento Adequado:

Fator: Definir claramente os objetivos do workshop, criar uma agenda detalhada e preparar materiais e atividades com antecedência são essenciais para o sucesso.

Ação: Estabelecer o que se espera alcançar com o workshop e comunicar isso aos participantes antes do evento.

Escolha de Participantes Relevantes:

Fator: Convidar as partes interessadas certas, que têm conhecimento e influência sobre os requisitos, é crucial para garantir que todas as perspectivas sejam consideradas.

Ação: Identificar e incluir representantes de todos os departamentos e níveis que serão impactados pelo projeto.

Facilitação Eficaz:

Fator: Ter um facilitador experiente e neutro que possa manter a reunião focada, gerenciar conflitos e garantir que todos tenham a oportunidade de contribuir.

Ação: O facilitador deve usar técnicas de gestão de grupo para manter a discussão produtiva e no caminho certo.

Documentação e Seguimento:

Fator: Registrar todas as discussões e decisões do workshop e garantir que haja um plano para revisar e validar as informações com os participantes.

Ação: Preparar um relatório com os requisitos discutidos e um plano de ação para acompanhamento, garantindo que as conclusões sejam bem documentadas e comunicadas.

Engajamento dos Participantes:

Fator: Garantir que todos os participantes estejam engajados e contribuam ativamente para o workshop.

Ação: Utilizar técnicas de participação ativa, como exercícios em grupo e discussões estruturadas, para manter os participantes envolvidos.

Exercício 5: Você está desenvolvendo um sistema de e-commerce e precisa entender as necessidades dos clientes. Crie um questionário com cinco perguntas que você enviaria aos usuários finais para coletar informações sobre suas expectativas em relação ao sistema.

R:

1 - Quais funcionalidades você considera mais importantes em um site de e-commerce?

2 - Como você avalia a importância de uma experiência de compra personalizada no e-commerce?

3 - Qual é o seu método de pagamento preferido para compras online?

4 - Qual é o nível de importância da velocidade de carregamento das páginas e do processo de checkout para você?

5 - Quais aspectos da política de devolução e troca são mais importantes para você ao fazer compras online?

Exercício 6: Explique como a análise de documentos pode ser usada na elicitação de requisitos. Que tipo de documentos você consideraria ao coletar requisitos para um sistema bancário?

R:

A análise de documentos é um método valioso na elicitação de requisitos, pois permite que os analistas revisem e compreendam informações relevantes que já estão registradas e documentadas. Esse método é particularmente útil para obter uma visão detalhada e precisa das necessidades e requisitos de um sistema, aproveitando documentos existentes que podem fornecer insights e contexto valiosos.

Regulamentos da Comissão de Valores Mobiliários (CVM), normas de proteção de dados como a Lei Geral de Proteção de Dados (LGPD) no Brasil, e regulamentações da Autoridade Monetária (como o Banco Central).

Exercício 7: Como o uso de protótipos pode ajudar na elicitação de requisitos? Descreva uma situação em que o uso de um protótipo seria particularmente útil.

R:

O uso de protótipos é uma abordagem eficaz na elicitação de requisitos, pois permite que stakeholders e desenvolvedores visualizem e interajam com uma representação preliminar do sistema. Essa interação direta ajuda a esclarecer e refinar requisitos, além de identificar problemas e oportunidades de melhoria de forma mais concreta. Aqui estão algumas maneiras pelas quais protótipos ajudam na elicitação de requisitos.

Visualização das Funcionalidades:

Descrição: Protótipos permitem que os stakeholders vejam e experimentem o layout, a navegação e as funcionalidades do sistema antes de ser desenvolvido. Isso ajuda a garantir que o sistema atenda às suas expectativas e requisitos.

Benefício: Ajuda a alinhar as expectativas dos stakeholders com o que será entregue, reduzindo o risco de mal-entendidos e alterações de requisitos tardias.

Exercício 8: Você está liderando uma sessão de brainstorming para um novo aplicativo de saúde. Quais regras e técnicas você usaria para garantir que a sessão de brainstorming seja produtiva?

R:

Regras para uma Sessão de Brainstorming Produtiva.

Encorage a Participação de Todos:

Regra: Todos os participantes devem se sentir à vontade para contribuir com suas ideias, independentemente de quão grandes ou pequenas elas possam parecer.

Exemplo: Comece a sessão lembrando a todos que não há ideias ruins e que todas as contribuições são valiosas.

Suspenda o Julgamento:

Regra: Evite avaliar ou criticar ideias enquanto a sessão de brainstorming está em andamento. O objetivo é gerar o máximo de ideias possível sem filtrar ou julgar.

Exemplo: Informe aos participantes que todas as ideias serão discutidas mais tarde e que o foco atual é a quantidade, não a qualidade.

Construa sobre as Ideias dos Outros:

Regra: Incentive os participantes a pegar ideias apresentadas por outros e expandi-las ou combiná-las para criar novas soluções.

Exemplo: Use frases como “Eu gosto da ideia de X. E se combinássemos isso com Y para criar Z?”

Seja Específico nas Ideias:

Regra: Evite generalidades e incentive os participantes a detalhar suas ideias com exemplos concretos e aplicações práticas.

Exemplo: Em vez de “melhorar a experiência do usuário”, peça para detalhar como uma nova funcionalidade pode ser implementada e qual seria o impacto para o usuário.

Limite o Tempo da Sessão:

Regra: Estabeleça um tempo limite para a sessão de brainstorming para manter a energia e o foco dos participantes.

Exemplo: Realize a sessão em blocos de 30 a 60 minutos, com intervalos curtos se necessário, para manter a produtividade.

Exercício 9: O que é uma análise de stakeholders e como ela influencia o processo de elicitação de requisitos? Dê exemplos de como diferentes stakeholders podem ter requisitos conflitantes.

R:

É um processo essencial no gerenciamento de projetos e na elicitação de requisitos que envolve identificar, compreender e gerenciar as expectativas e necessidades dos diferentes indivíduos ou grupos que têm interesse ou são afetados pelo projeto.

O objetivo da análise de stakeholders é garantir que todos os aspectos e perspectivas relevantes sejam considerados, contribuindo para o sucesso do projeto e minimizando riscos associados à falta de alinhamento com as partes interessadas.

Exemplos de Requisitos Conflitantes entre Stakeholders

Clientes vs. Reguladores:

Conflito: Clientes podem desejar uma interface do usuário simples e intuitiva, enquanto reguladores podem exigir funcionalidades complexas para garantir conformidade e rastreamento de transações.

Exemplo: Um cliente pode preferir um processo de abertura de conta rápido e simplificado, enquanto os reguladores podem exigir etapas adicionais para verificar a identidade e prevenir fraudes.

Equipe de Desenvolvimento vs. Usuários Finais:

Conflito: A equipe de desenvolvimento pode preferir usar tecnologias e abordagens que otimizem a eficiência do código e o desenvolvimento, enquanto os usuários finais podem exigir uma interface mais amigável e personalizável.

Exemplo: A equipe de desenvolvimento pode optar por uma estrutura de dados complexa para melhorar o desempenho, mas os usuários finais podem achar essa complexidade difícil de entender e usar.

Gerentes de Projetos vs. Equipe de Suporte ao Cliente:

Conflito: Gerentes de projetos podem se concentrar em cumprir prazos e orçamentos, enquanto a equipe de suporte ao cliente pode precisar de mais tempo para treinar e adaptar-se ao novo sistema.

Exemplo: O gerenciamento pode querer lançar uma nova funcionalidade rapidamente para atender a uma demanda de mercado, enquanto a equipe de suporte pode precisar de mais tempo para garantir que o suporte adequado e o treinamento estejam disponíveis para os clientes.

Exercício 10: Após coletar uma lista extensa de requisitos, você precisa priorizá-los. Descreva três técnicas que você poderia usar para priorizar os requisitos e explique como cada técnica funciona.

R:

1. Técnica MoSCoW

Descrição: A técnica MoSCoW é uma metodologia de priorização que classifica os requisitos em quatro categorias baseadas na importância e necessidade.

Categorias:

M (Must Have): Requisitos obrigatórios que são essenciais para que o sistema funcione. Sem estes requisitos, o sistema não atende ao seu propósito básico.

S (Should Have): Requisitos importantes, mas não vitais. Eles adicionam valor significativo, mas a ausência deles não impede o funcionamento do sistema.

C (Could Have): Requisitos desejáveis que são bons de ter, mas não críticos. Eles podem ser incluídos se houver tempo e recursos disponíveis.

W (Won't Have): Requisitos que foram considerados, mas que não serão incluídos na versão atual do sistema. Podem ser revisados para futuras versões.

Como Funciona:

1. **Listagem:** Compile uma lista completa de requisitos.
2. **Classificação:** Classifique cada requisito nas categorias MoSCoW com base na sua importância e necessidade.
3. **Revisão:** Revise a lista com as partes interessadas para garantir que todos concordem com as classificações.

2. Método de Pontuação e Classificação (Scoring Model)

Descrição: O método de pontuação e classificação é uma técnica quantitativa onde cada requisito é avaliado e atribuído uma pontuação com base em múltiplos critérios. A soma das pontuações ajuda a determinar a prioridade dos requisitos.

Como Funciona:

Definição de Critérios: Estabeleça critérios de avaliação para os requisitos, como valor de negócio, custo de implementação, impacto no usuário, etc.

Pontuação: Atribua uma pontuação para cada requisito em relação a cada critério. Pontuações podem ser em uma escala (por exemplo, 1 a 5).

Cálculo da Pontuação Total: Some as pontuações para cada requisito para obter uma pontuação total.

Prioritização: Classifique os requisitos com base nas pontuações totais.

3. Análise de Valor vs. Esforço

Descrição: A análise de valor versus esforço é uma técnica que avalia a relação entre o valor que um requisito traz para o projeto e o esforço necessário para sua implementação. Requisitos são então classificados com base nessa relação.

Como Funciona:

Avaliação do Valor: Determine o valor que cada requisito traz para o projeto, seja em termos de benefícios para os usuários, impacto no negócio, ou outros fatores relevantes.

Avaliação do Esforço: Estime o esforço necessário para implementar cada requisito, considerando o tempo, custo e recursos necessários.

Classificação: Classifique os requisitos em uma matriz de valor vs. esforço:

Alta Valor/baixo Esforço: Prioridade alta.

Alta Valor/alto Esforço: Prioridade média.

Baixo Valor/baixo Esforço: Prioridade média.

Baixo Valor/alto Esforço: Prioridade baixa.

Exercício 11: Um cliente deseja desenvolver um novo sistema de reservas para um hotel. Descreva como você conduziria o processo de elicitação de requisitos para este projeto. Inclua os métodos que você utilizaria e como você envolveria os diferentes stakeholders.

R:

Para desenvolver um novo sistema de reservas para um hotel, a elicitação de requisitos deve ser conduzida de forma estruturada e abrangente para garantir que o sistema atenda às necessidades dos usuários finais e aos objetivos do negócio. O processo deve envolver diferentes stakeholders e utilizar vários métodos para obter uma compreensão completa dos requisitos. Aqui está um plano detalhado para conduzir a elicitação de requisitos para este projeto:

1. Planejamento da Elicitação de Requisitos

Passos:

Definir o Escopo: Determine o escopo do sistema de reservas, incluindo as funcionalidades principais e as expectativas gerais do cliente.

Identificar Stakeholders: Liste todos os stakeholders envolvidos, como clientes do hotel, funcionários da recepção, gerentes de hotel, equipe de TI, e até mesmo fornecedores de sistemas de pagamento.

2. Coleta de Informações

Métodos e Envolvimento:

Entrevistas com Stakeholders

Descrição: Conduza entrevistas individuais com stakeholders chave para entender suas necessidades e expectativas.

Envolvimento:

Gerente de Hotel: Discuta as funcionalidades desejadas, como integração com sistemas de gerenciamento de propriedade e relatórios analíticos.

Receptionistas e Funcionários: Entenda o fluxo de trabalho diário e os desafios enfrentados com o sistema atual.

Clientes do Hotel: Obtenha feedback sobre as expectativas para uma interface amigável e opções de personalização.

Workshops de Requisitos

Descrição: Realize workshops colaborativos com múltiplos stakeholders para discutir e priorizar requisitos.

Envolvimento:

Equipe de Desenvolvimento e TI: Inclua-os para garantir que as expectativas tecnológicas e limitações sejam discutidas.

Equipe de Marketing e Vendas: Discuta como o sistema pode suportar promoções, pacotes e ofertas especiais.

Gerente de Operações: Envolver os para garantir que o sistema atenda às necessidades operacionais e de gerenciamento.

Observação

Descrição: Observe como o sistema de reservas atual (se houver) é usado no dia a dia.

Envolvimento:

Recepcionistas e Funcionários: Observe a interação deles com o sistema atual para identificar pontos de dor e áreas de melhoria.

Análise de Documentos

Descrição: Revise documentos existentes, como manuais do sistema atual, relatórios de desempenho e feedback de usuários.

Envolvimento:

Gerente de TI e Administradores de Sistema: Revise a documentação técnica existente para entender as integrações e requisitos técnicos.

Pesquisas e Questionários

Descrição: Envie questionários para clientes e funcionários para obter feedback amplo e quantitativo.

Envolvimento:

Clientes do Hotel: Coleta de feedback sobre preferências e expectativas para o sistema de reservas.

Funcionários: Obtenha insights sobre necessidades operacionais e sugestões de melhorias.

3. Análise e Documentação de Requisitos

Passos:

Compilação e Revisão de Dados: Compile as informações obtidas através dos diferentes métodos e revise com os stakeholders para garantir precisão.

Criação de Documentos de Requisitos: Desenvolva um Documento de Requisitos de Software (DRS) detalhado que inclui requisitos funcionais e não funcionais.

Prototipagem: Crie protótipos de baixa fidelidade para validar e refinar os requisitos com os stakeholders.

4. Validação e Refinamento

Métodos e Envolvimento:

Revisão de Requisitos com Stakeholders

Descrição: Realize reuniões para revisar os requisitos documentados e garantir que todas as necessidades foram corretamente capturadas.

Envolvimento:

Todos os Stakeholders: Confirme que os requisitos atendem às expectativas e faça ajustes conforme necessário.

Testes de Aceitação

Descrição: Defina critérios de aceitação com base nos requisitos e execute testes para garantir que o sistema atenda aos critérios definidos.

Envolvimento:

Usuários Finais e Testadores: Realize testes de usabilidade e funcionalidade com usuários reais para validar o sistema.

5. Implementação e Feedback

Passos:

Desenvolvimento e Implementação: Conduza o desenvolvimento do sistema de acordo com os requisitos e protótipos validados.

Treinamento e Suporte: Ofereça treinamento para os funcionários e suporte durante a transição para o novo sistema.

Coleta de Feedback Pós-Implementação: Recolha feedback contínuo dos usuários para identificar áreas de melhoria e ajustar o sistema conforme necessário.

Exercício 12: Após a elicitação, os requisitos precisam ser validados. Explique o que é a validação de requisitos e descreva três técnicas que podem ser usadas para validar os requisitos coletados.

R:

A validação de requisitos é o processo de garantir que os requisitos coletados e documentados atendem às necessidades e expectativas dos stakeholders e estão corretos, completos e viáveis.

Aqui estão três técnicas comuns usadas para validar os requisitos:

Revisão de Requisitos (ou Inspeção de Requisitos):

Descrição: Envolve a reunião de uma equipe de revisão, composta por stakeholders, analistas de negócios, desenvolvedores e outros especialistas relevantes, para examinar o documento de requisitos. Cada membro revisa o conteúdo com base em seu conhecimento e experiência, procurando erros, inconsistências e áreas de melhoria.

Objetivo: Garantir que todos os aspectos dos requisitos estão claros, completos e corretos, e que todas as partes interessadas têm uma compreensão compartilhada dos requisitos.

Protótipos:

Descrição: Criação de modelos ou versões simplificadas do sistema ou produto para representar visualmente os requisitos. Protótipos podem variar de papel e lápis a protótipos interativos de alta fidelidade.

Objetivo: Permitir que os stakeholders vejam e interajam com uma representação do sistema antes de sua construção final. Isso ajuda a identificar e corrigir mal-entendidos e mudanças nos requisitos de forma mais rápida e eficaz.

Testes de Aceitação de Usuário (UAT - User Acceptance Testing):

Descrição: Envolve a execução de testes onde os usuários finais ou representantes dos stakeholders validam se o sistema atende aos seus requisitos e expectativas. Esses testes são realizados em um ambiente que simula o uso real do sistema.

Objetivo: Confirmar que o sistema ou produto atende às necessidades e expectativas dos usuários finais e que os requisitos foram implementados corretamente.

Exercício 13: Uma startup deseja criar um aplicativo de entrega de alimentos que conecta restaurantes a consumidores. O aplicativo deve permitir que os usuários façam pedidos de forma rápida e eficiente, acompanhem o status do pedido em tempo real, e ofereça aos restaurantes uma plataforma para gerenciar seus menus e pedidos. Identifique os requisitos.

R:

Cadastro e Autenticação:

Usuários (Consumidores): Devem ser capazes de criar contas, fazer login e gerenciar suas informações pessoais (nome, endereço, telefone, etc.).

Restaurantes: Devem ser capazes de se cadastrar, fazer login e atualizar suas informações de restaurante (nome, endereço, tipo de cozinha, etc.).

Pesquisa e Navegação:

Consumidores: Devem ser capazes de pesquisar restaurantes por localização, tipo de comida, e outros filtros. A navegação deve ser intuitiva e permitir a visualização de menus e opções de pratos.

Restaurantes: Devem ter uma interface para adicionar e gerenciar itens do menu, incluindo descrições, preços e imagens.

Pedido e Checkout:

Consumidores: Devem ser capazes de selecionar itens do menu, adicionar ao carrinho, revisar o pedido e proceder ao checkout. O checkout deve incluir opções de pagamento (cartão de crédito, débito, etc.) e a escolha do método de entrega (retirar no local, entrega em domicílio).

Restaurantes: Devem receber notificações de novos pedidos, visualizar detalhes do pedido e atualizar o status (em preparo, pronto para entrega, etc.).

Rastreamento de Pedido:

Consumidores: Devem poder acompanhar o status do pedido em tempo real, desde a confirmação até a entrega.

Restaurantes: Devem atualizar o status do pedido e fornecer informações de rastreamento.

Gerenciamento de Pedidos e Menus:

Restaurantes: Devem ter a capacidade de atualizar o menu, adicionar ou remover itens, e ajustar preços. Também devem poder gerenciar e revisar pedidos, incluindo histórico de pedidos.

Avaliações e Feedback:

Consumidores: Devem ser capazes de avaliar restaurantes e fornecer feedback sobre a qualidade da comida e do serviço.

Restaurantes: Devem ter acesso às avaliações e feedbacks dos consumidores.

Notificações:

Consumidores: Devem receber notificações sobre o status do pedido, ofertas especiais e promoções.

Restaurantes: Devem receber notificações sobre novos pedidos e atualizações de status.