



Sistema de gerenciamento hospitalar

Tarefa: 1. Identificação dos requisitos: - analise o cenário descrito e liste os principais requisitos funcionais que o sistema deve atender. Cada requisito funcional deve ser descrito de maneira clara e objetiva, explicando o que o sistema deve ser capaz de fazer. - Liste também os requisitos não funcionais, como desempenho, usabilidade, segurança e compatibilidade, que devem ser levados em consideração no desenvolvimento do sistema.

Requisitos funcionais:

Agendamento de consultas: gerenciar marcação, alteração e cancelamento de consultas, exibir disponibilidade dos médicos, enviar notificações automáticas.

Controle de estoques: registrar movimentações de medicamentos, emitir alertas para baixos estoques, acessar histórico de estoque.

Prontuários médicos: criar, atualizar e visualizar prontuários, anexar exames, acesso controlado por permissões.

Comunicação entre setores: trocar mensagens e enviar alertas entre setores, manter registro do histórico de comunicações.

Gerenciamento de funcionários: registrar e gerenciar informações dos funcionários, controlar escalas.

Gestão de pacientes: cadastrar e atualizar dados dos pacientes, acessar histórico médico

Requisitos não funcionais:

Desempenho: O sistema deve ter um tempo de resposta rápido, suportar muitos usuários simultâneos.

Usabilidade: interface fácil de usar, adaptável e acessível em diversos dispositivos.

Segurança: criptografia nos dados; autenticação em duas etapas, registro de atividades.

Escalabilidade: suportar aumento de usuários e dados.

Internacionalização: suporte a múltiplos idiomas e configurações regionais

Backup e recuperação: backup automático e planos de recuperação

Acessibilidade: acessível para pessoas com deficiências

Compatibilidade: funciona com vários navegadores e sistemas; integra-se com sistemas hospitalares já existentes.

2. Priorização de requisitos: classifique os requisitos identificados em ordem de prioridade, justificando a escolha dos requisitos mais críticos para o sucesso do sistema. Explique como a priorização de requisitos pode impactar o cronograma e o custo do projeto.

Funcionais

Alta prioridade:

Prontuários médicos: fundamental para a gestão das informações dos pacientes.

Agendamento de consultas: vital para a organização das consultas e gerenciamento do fluxo de pacientes.

Gestão de pacientes: essencial para atualizar e acessar informações dos pacientes.

Média prioridade:

Controle de estoques: assegura a disponibilidade contínua de medicamentos.

Comunicação entre setores: facilita a coordenação entre diferentes áreas.

Baixa prioridade:

Gerenciamento de funcionários: importante para a administração, mas pode ser implementado em etapas posteriores.

Não funcionais**Alta prioridade:**

Segurança: crucial para proteger os dados e garantir a integridade das informações.

Desempenho: fundamental para garantir respostas rápidas e suportar muitos

Usabilidade: assegura que o sistema seja fácil de usar para todos os usuários.
Usuários

Média prioridade:

Escalabilidade: importante para suportar o crescimento do hospital, mas pode ser abordado posteriormente.

Backup e recuperação: essencial para a proteção dos dados, mas pode ser implementado em etapas.

Compatibilidade: importante para garantir integração com sistemas já existentes.

Baixa prioridade:

Acessibilidade: assegura inclusão de todos os usuários, mas pode ser tratada após as funcionalidades principais

Internacionalização: só necessária se houver a demanda para suportar múltiplos idiomas.

Impacto no cronograma e custo:

Cronograma: comece pelos requisitos mais importantes para garantir que o sistema funcione bem desde o início. Requisitos menos urgentes podem ser adicionados depois.

Custo: requisitos essenciais podem aumentar o custo inicial, mas evitam problemas futuros. Funcionalidades secundárias podem ser implementadas mais tarde para ajudar a controlar melhor os gastos

3. Análise de impacto: - identifique e descreva possíveis desafios e impactos que esses requisitos podem trazer ao processo de desenvolvimento e à operação do sistema. - Discuta como as interdependências entre os requisitos podem afetar a implementação do sistema.

Desafios e impactos:

Prontuários médicos: garantir a segurança dos dados e integrá-los com outros sistemas pode ser complicado e caro.

Agendamento de consultas: manter o sistema funcionando bem e alinhado com a disponibilidade dos médicos pode ser desafiador e afetar a satisfação dos usuários.

Gestão de pacientes: manter os dados atualizados é essencial para um bom atendimento; erros podem afetar as decisões médicas.

Controle de estoques: gerenciar o inventário com precisão é crucial para evitar falta de medicamentos, o que pode impactar o atendimento.

Comunicação entre setores: manter uma comunicação eficiente é importante para a coordenação interna e para evitar erros.

Gerenciamento de funcionários: gerenciar as escalas e informações da equipe pode ser complicado e afetar a operação do hospital.

Interdependências e impactos:

Segurança e prontuários médicos: proteger bem os dados dos prontuários é muito importante.

Desempenho e agendamento de consultas: O sistema precisa funcionar bem para gerenciar o agendamento de consultas sem problemas.

Gestão de pacientes e controle de estoques: mais pacientes podem aumentar a demanda por medicamentos.

Comunicação e gerenciamento de funcionários: uma boa comunicação ajuda a resolver problemas e a usar os recursos de forma mais eficiente.

Escalabilidade e usabilidade: precisamos planejar a escalabilidade para que o sistema continue fácil de usar conforme cresce.

Essas conexões entre os requisitos precisam de um bom planejamento para garantir que o sistema funcione de forma eficaz e coordenada.

4. Propostas de melhoria: - sugira possíveis melhorias ou adições aos requisitos que poderiam aumentar a eficiência, segurança ou usabilidade do sistema. - Discuta como essas melhorias poderiam ser integradas ao projeto sem comprometer o prazo e o orçamento.

Propostas de melhoria:

Segurança adicional: adicionar autenticação biométrica e criptografia avançada para proteger dados sensíveis.

Como deve ser feito: introduzindo essas medidas aos poucos, começando pelas partes mais críticas para não atrasar o projeto.

Desempenho e escalabilidade: usando cache e otimize o banco de dados para melhorar o desempenho e suportar mais usuários.

Como deve ser feito: fazendo essas melhorias em etapas, começando com as partes mais importantes e testando à medida que avança.

Usabilidade avançada: melhore a interface com suporte para temas e acessibilidade, como leitura de tela.

Como deve ser feito: implementando as melhorias gradualmente, começando com as mais essenciais e ajustando com base no feedback.

Comunicação e colaboração: adicionando chat em tempo real e integre com plataformas de comunicação existentes.

Como deve ser feito: comece com uma solução básica e adicione mais funcionalidades conforme necessário, usando apis prontas para facilitar a integração.

Gestão de recursos: use alertas e análise preditiva para antecipar as necessidades de recursos médicos.

Como deve ser feito: desenvolvendo e implementando um sistema de alertas em fases, começando com alertas simples e expandindo com o tempo.

Automação e relatórios: adicione ferramentas para gerar relatórios e análises automaticamente.

Como deve ser feito: comece com automação básica e adicione funções mais avançadas conforme o projeto avança.

Integração ao projeto

Fases e prioridades: faça as melhorias em etapas para controlar melhor o tempo e o orçamento, começando pelas mais importantes.

Testes e feedback: teste as melhorias sempre e ajuste conforme o feedback para evitar problemas.

Tecnologia pronta: use ferramentas e soluções já existentes para economizar tempo e recursos.

Orçamento e prazo: planeje o orçamento e o cronograma para cada etapa das melhorias e ajuste conforme necessário para manter o projeto no caminho certo.

Tarefa: faça a análise crítica do exemplo de ERS para o sistema de gerenciamento de biblioteca digital com objetivo de torná-lo mais completo, considere os pontos:

- **Clareza e detalhamento**
- **Melhorias específicas de requisitos**
- **Sugestão de melhoria**
- **Cobertura de funcionalidades**
- **Consistência e organização**
- **Viabilidade técnica**
- **Segurança**
- **Usabilidade e acessibilidade**
- **Priorização dos requisitos**

Clareza e detalhamento: está bem-organizado e cobre o básico, mas poderia ter mais detalhes. Os requisitos não funcionais são meio vagos e precisam de definições mais claras ou testes de usabilidade. A seção de definições também é curta e poderia incluir termos como "acervo" e "usuário administrador".

Melhorias específicas de requisitos: os requisitos de usabilidade poderiam ser mais detalhados, como seguir diretrizes para garantir acessibilidade. Já os requisitos de segurança deveriam incluir mais detalhes sobre proteção contra-ataques e criptografia de dados, além das senhas.

Sugestão de melhoria: os requisitos de performance poderiam ser mais detalhados, incluindo o número de consultas simultâneas e transações em picos. A gestão de usuários também poderia ter opções para diferentes níveis de permissão, tornando o gerenciamento mais flexível

Cobertura de funcionalidades: O ERS está bom, mas precisa incluir relatórios e estatísticas, integração com sistemas externos e opções de notificações como SMS.

Consistência e organização: está bom, mas poderia ser melhorado reorganizando as seções. Por exemplo, os requisitos funcionais podem ser divididos em categorias como "gestão de usuários" e "gestão de acervo". Além disso, as considerações finais poderiam incluir algo sobre "futuras expansões" ou "requisitos adicionais" para versões futuras.

Viabilidade técnica: os requisitos parecem viáveis, mas seria bom considerar a escalabilidade, já que o só fala em 500 usuários simultâneos e não menciona o crescimento futuro. Também seria útil detalhar a compatibilidade, especificando não só os navegadores, mas também diferentes sistemas operacionais e dispositivos.

Segurança: além das melhorias em segurança já mencionadas, seria importante adicionar autenticação vários fatores e incluir requisitos para gerar logs e auditar ações no sistema para monitorar atividades suspeitas e garantir conformidade.

Usabilidade e acessibilidade: O design responsivo precisa de mais detalhes sobre como a interface se adapta a diferentes dispositivos e resoluções, e deve incluir testes específicos para dispositivos móveis. Além disso, é importante garantir a acessibilidade seguindo diretrizes para atender a todos os usuários.