

Projeto: Sistema de Suporte ao Método Kanban	Júlia Yuri Garcia Baba - 190057921
Especificação de Requisitos não expressos por meio de casos de uso	Data: 29/11/2023

Sistema de Suporte ao Método Kanban

Especificação de Requisitos não expressos por meio de casos de uso

1. Introdução

Este documento tem como objetivo fornecer uma visão abrangente dos requisitos de interface, funcionais e não funcionais para o desenvolvimento e implementação do Sistema de Suporte ao Método Kanban no Gerenciamento de Projetos. Ademais, o artefato busca delinear os requisitos essenciais, incluindo funcionalidades específicas, restrições, e considerações críticas para garantir a conformidade com padrões. Sua leitura é fundamental para todos os membros da equipe envolvidos no desenvolvimento, implementação e utilização contínua do sistema.

2. Requisitos não expressos por meio de casos de uso

2.1 Requisitos não funcionais

I. Prover serviços por meio de World Wide Web

- O sistema deve aderir aos padrões web relevantes, garantindo a compatibilidade com tecnologias e práticas modernas.
- O sistema deve ser compatível com os principais navegadores da web, garantindo uma experiência consistente para os usuários.
- O sistema deve suportar múltiplos idiomas, permitindo uma experiência global para usuários de diferentes regiões.

II. Segurança

- Os dados do usuário devem ser armazenados de forma segura, utilizando criptografia para proteger informações sensíveis, como senhas e dados pessoais.

III. Desempenho

- O sistema deve fornecer uma resposta rápida, com tempos de carregamento de páginas e execução de operações dentro de limites aceitáveis.

IV. Usabilidade

- A interface do usuário deve ser intuitiva e fácil de usar, permitindo que usuários com diferentes níveis de experiência interajam eficientemente com o sistema.

V. Documentação

- Deve ser fornecida uma documentação abrangente para usuários e desenvolvedores, detalhando o funcionamento do sistema e os procedimentos de uso.

2.2 Requisitos funcionais

I. Cadastro de Usuário

- O sistema deve permitir que os usuários se cadastrem fornecendo nome, endereço de correio eletrônico e senha.

II. Autenticação de Usuário

- Os usuários devem ser capazes de se autenticar no sistema através do fornecimento do endereço de e-mail e senha.
- Usuários devem ser autenticados para ter acesso aos serviços além do cadastro.

III. Gerenciamento de Conta

- Após autenticação, os usuários podem visualizar, editar e deletar sua conta.
- A visualização da conta apresenta os dados associados a ela.

- A edição da conta permite a modificação do nome e/ou senha.

IV. Gerenciamento de Quadros

- Usuários autenticados podem criar, visualizar e eliminar quadros, seguindo as regras de negócio.
- Ao criar quadro, dados necessários devem ser informados.

V. Regras do Sistema

- Usuários só podem eliminar quadros por eles criados.
- Eliminar um quadro resulta na eliminação dos cartões associados.
- O número máximo de cartões na coluna “Trabalho em Progresso” é igual ao limite informado no quadro.

VI. Gerenciamento de Cartões

- Usuários autenticados podem criar, visualizar, mover e eliminar cartões.
- Ao criar cartão, dados necessários devem ser informados.
- O sistema deve permitir a alteração do nome da coluna onde um cartão está localizado.

VII. Visualização de Cartões e Quadros

- Usuários devem informar código de quadro ou cartão para visualizar os mesmos.
- A visualização de cartão mostra dados do cartão.
- A visualização de quadro mostra dados do mesmo e códigos dos cartões associados a ele.

3. Qualidades do Sistema

3.1 Usabilidade (Usability)

- Interface Intuitiva: O sistema deve apresentar uma interface de usuário intuitiva e amigável para facilitar a navegação e interação, tornando-o acessível para usuários de diferentes níveis de experiência.
- Personalização de Contas: A capacidade de personalizar contas, incluindo a edição de informações como nome e senha, contribui para uma experiência mais individualizada e amigável.

3.2 Confiabilidade (Reliability)

- Autenticação Segura: Garantir autenticação segura para proteger as informações dos usuários e permitir o acesso apenas a usuários autorizados é fundamental para a confiabilidade do sistema.
- O sistema deve manter alta disponibilidade, com tempo mínimo de inatividade durante operações rotineiras.
- A frequência e gravidade de falhas devem ser mínimas, garantindo uma experiência de usuário confiável.
- Mecanismos de recuperação devem estar em vigor para restaurar o sistema a um estado estável após falhas inesperadas.

3.3 Desempenho (Performance)

- O sistema deve apresentar tempos de resposta ótimos para ações como criação de quadros, movimentação de cartões e recuperação de dados.
- O throughput deve ser otimizado para lidar com um número significativo de usuários simultâneos sem degradação de desempenho.
- O planejamento de capacidade deve ser considerado para acomodar o crescimento futuro em termos de dados e base de usuários.
- Os tempos de inicialização e desligamento devem estar dentro de limites aceitáveis para minimizar o tempo de espera do usuário.

3.4 Suporte (Supportability)

- O sistema deve ser projetado para facilitar adaptabilidade e atualizações para acomodar futuras melhorias.
- A compatibilidade com vários navegadores e dispositivos deve ser garantida para uma ampla base de usuários.
- Recursos de configurabilidade devem permitir que administradores personalizem o sistema conforme as necessidades específicas da organização.
- A escalabilidade é essencial para suportar um número crescente de usuários, quadros e cartões.
- A instalação do sistema deve ser direta, e documentação deve ser fornecida para fins de suporte e manutenção.

4. Interfaces do Sistema

4.1 Interfaces de Usuário

I. Painel de Autenticação

- Os campos de entrada para endereço de e-mail e senha devem ser apresentados de forma clara, com rótulos explicativos, facilitando a compreensão e prevenindo erros de entrada.
- Estabelecer uma política de senha que incentive a criação de senhas fortes, incluindo requisitos mínimos de comprimento, caracteres especiais e combinação de maiúsculas e minúsculas.

II. Painel Inicial

- Após o login, cada usuário deve ter um painel personalizado que exiba de forma clara e concisa as opções de busca de quadros e cartões, assim como as opções de personalização de conta.
- Após a busca, deve exibir opção de edição, visualização, movimentação e eliminação
- Deve haver opção de criar quadro ou cartão.

III. Painel de Cartão

- O painel deve mostrar todas informações relacionadas ao cartão (código, nome, descrição e coluna).

IV. Painel de Quadro

- O painel deve mostrar todas informações relacionadas ao quadro (código, nome, descrição e limite) e o código de todos os cards associados a ele.

V. Painel de Criação de Quadro

- O painel deve solicitar informações de código, nome, descrição e coluna.

VI. Interface de Erro

- Caso alguma ação infrinja as regras de negócio deve ser exibido uma mensagem de erro na tela.

VII. Menu Navegacional

- O menu de navegação deve ser organizado de forma lógica e categorizado de maneira clara para que os usuários possam acessar diferentes funcionalidades de forma direta e rápida.

4.1.1 Aparência e Sensação

A interface do usuário será desenvolvida usando tecnologias web modernas, como HTML5, CSS3 e JavaScript. O design seguirá as diretrizes de Material Design, adotando uma paleta de cores sóbrias, ícones intuitivos e transições suaves para criar uma experiência visualmente agradável.

4.1.2 Requisitos de Layout e Navegação

- O Painel de Autenticação apresentará campos de entrada responsivos com rótulos flutuantes para garantir clareza e reduzir erros de entrada.
- O Painel Inicial após o login será dinâmico, exibindo informações em cartões reutilizáveis. As opções de busca, edição, movimentação, eliminação e criação serão implementadas como botões interativos para atualizações em tempo real.
- Os painéis de Cartão e Quadro serão componentes reutilizáveis, garantindo uma estrutura consistente e interações suaves. A edição de cartões ou quadros será realizada por meio de modais interativos.
- Os Painéis de Criação de Quadro usarão formulários controlados em React, validando os dados do usuário antes da submissão. O backend será responsável por validar a consistência dos dados.

4.1.3 Consistência

- Os botões de navegação, como "Voltar" e "Página Inicial", devem manter uma posição consistente em todas as telas do sistema.
- A ordem e o layout dos menus de navegação devem permanecer constantes, proporcionando uma experiência previsível.
- O tamanho e a forma das áreas de tela principais, como janelas modais e painéis laterais, devem seguir um padrão consistente para garantir familiaridade ao usuário.
- Elementos gráficos, como ícones e botões, devem ter dimensões uniformes em toda a aplicação.
- Campos de entrada e áreas de apresentação de dados devem seguir uma estrutura consistente em termos de alinhamento e margens.
- Rótulos de campos devem manter a mesma posição em relação aos campos correspondentes em diferentes telas.
- O uso de terminologia específica deve ser consistente em todo o sistema. Por exemplo, se um termo específico é utilizado para descrever uma ação, esse mesmo termo deve ser empregado de maneira uniforme em todas as partes da aplicação.
- Mensagens de erro, alertas e instruções devem seguir uma linguagem padronizada para facilitar a compreensão do usuário.

4.1.4 Personalização do Usuário e Requisitos de customização

- O sistema deve identificar automaticamente atributos do usuário, como preferências de idioma, configurações de exibição e histórico de atividades. Com base nesses atributos, o sistema deve apresentar automaticamente conteúdo relevante, como quadros recentes, cartões destacados ou atualizações pertinentes.
- Os usuários terão a capacidade de personalizar a disposição dos elementos na tela inicial, permitindo que escolham a ordem e a visibilidade de componentes específicos. Configurações de personalização devem ser persistentes, mantendo-se mesmo após o fechamento e reabertura da aplicação.
- Os usuários podem escolher quais informações desejam exibir ou ocultar em diferentes áreas da interface. A personalização pode incluir a escolha de widgets, painéis e tipos de notificações que são relevantes para as necessidades individuais do usuário.

4.2 Interfaces para sistemas externos ou dispositivos

4.2.1 Interfaces de Software

O sistema integrará um serviço de autenticação baseado em OAuth 2.0 para garantir a segurança e a conformidade com padrões modernos. Isso implica na implementação de fluxos de autorização adequados e no manuseio seguro de tokens de acesso.

Além disso, o sistema poderá incorporar bibliotecas de software reutilizáveis, como bibliotecas de gráficos ou módulos de gerenciamento de estado. Nesse caso, é crucial garantir a compatibilidade entre versões dessas bibliotecas e adaptar seu funcionamento aos requisitos específicos do sistema.

4.2.2 Interfaces de Hardware

O software será projetado para ser compatível com telas sensíveis ao toque, possibilitando interações em dispositivos que suportam essa funcionalidade. Isso requer uma adaptação adequada da interface do usuário para interação por toque e um reconhecimento preciso de gestos.

O sistema então deverá suportar diferentes dispositivos de entrada, como teclados e mouses, garantindo uma experiência consistente. Isso implica em mapear corretamente comandos de entrada para ações no sistema e realizar testes abrangentes de compatibilidade com uma variedade de dispositivos.

4.2.3 Interfaces de Comunicação

A comunicação entre o cliente e o servidor ocorrerá por meio dos protocolos HTTP/HTTPS, garantindo segurança nas transmissões por meio de certificados SSL/TLS. A implementação de APIs RESTful facilitará a interoperabilidade e a comunicação eficiente.

Em situações em que a integração com outros sistemas é necessária, como sistemas legados ou serviços externos, será crucial estabelecer uma definição clara dos formatos de dados e protocolos de comunicação. Além disso, a implementação de tratamento de erros e fallbacks garantirá a resiliência em cenários de integração.

5. Regras de Negócio

5.1 Regras de Gerenciamento de Conta

5.1.1 Regra de Criação de Conta – ID: 1

Se um usuário pretende criar uma conta, ele deve fornecer um nome, endereço de e-mail e senha válidos. O sistema deve armazenar essas informações para fins de autenticação.

5.1.2 Regra de Autenticação – ID: 2

Para acessar serviços relacionados à conta, um usuário deve ser autenticado. A autenticação requer o fornecimento de um endereço de e-mail e senha válidos.

5.1.3 Regra de Acessos da Conta – ID: 3

Uma vez autenticado, o usuário tem acesso a serviços como visualização, edição e exclusão de sua conta. A visualização da conta apresenta os dados do usuário, enquanto a edição permite alterações no nome e/ou senha.

5.2 Regras de Gestão do Sistema Kanban

5.2.1 Regra de Gerenciamento de Quadros – ID: 4

Usuários autenticados podem criar, visualizar e excluir quadros. A criação de um quadro

requer o fornecimento dos dados necessários, e a exclusão de um quadro só é permitida se o usuário o criou. A exclusão de um quadro resulta na eliminação de todos os cartões associados a ele.

5.2.2 Regra de Gerenciamento de Cartões – ID: 5

Os usuários autenticados podem criar, visualizar, mover e excluir cartões nos quadros. A criação de um cartão requer dados específicos. Os usuários podem mover um cartão, alterando o nome da coluna onde o cartão está localizado. O sistema garante que a quantidade de cartões na coluna “trabalho em andamento” não ultrapasse o limite especificado no quadro.

5.3 Regras de Validação de Dados

5.3.1 Regra de Formato de e-mail – ID: 6

Valida o formato de um endereço de e-mail. O endereço de e-mail deverá seguir o padrão “nome@domínio”, onde ‘nome’ é composto de 2 a 10 caracteres, e ‘domínio’ é composto de 2 a 20 caracteres. Cada caractere deve ser uma letra (A-Z ou a-z), um dígito (0-9) ou um ponto final (.). O símbolo '@' não pode ser imediatamente precedido ou sucedido por um ponto final. Não deve haver pontos consecutivos.

5.3.2 Regra de Valor da coluna – ID: 7

Garante que o valor da coluna no sistema seja uma das opções válidas: “Solicitado”, “Em execução” ou “Concluído”.

5.3.3 Regra de Validação de senha – ID: 8

Valida o formato de uma senha. A senha deve consistir em cinco caracteres, incluindo pelo menos uma letra maiúscula, uma letra minúscula, um dígito e um sinal de pontuação (. , ; ? !). Nenhum caractere pode ser duplicado.

5.3.4 Regra de Validação de código – ID: 9

Valida o formato de um código. O código deve seguir o padrão LLDD, onde 'L' é uma letra maiúscula (A-Z) e 'D' é um dígito (0-9).

5.3.5 Regra de Formato do limite – ID: 10

Valida o formato de um limite. O limite deve ser uma das opções válidas: 5, 10, 15 ou 20.

5.3.6 Regra de Validação do texto – ID: 11

Valida o formato de um texto. Cada caractere deve ser uma letra (A-Z ou a-z), dígito (0-9), sinal de pontuação (. , ; ? !) ou espaço. Não deve haver espaços consecutivos, nem sinais de pontuação consecutivos, nem acentos. O primeiro caractere deve ser uma letra maiúscula e o primeiro caractere após um sinal de pontuação (exceto vírgula ou ponto e vírgula) deve ser uma letra maiúscula.

6. Restrições do Sistema

6.1 Linguagem de Implementação

A equipe de desenvolvimento deverá aderir ao uso de ferramentas específicas para controle de versão, integração contínua e teste automatizado. As ferramentas serão definidas no início do projeto e devem ser seguidas durante todo o ciclo de vida do desenvolvimento.

6.2 Ferramentas de Desenvolvimento

A equipe de desenvolvimento deverá aderir ao uso de ferramentas específicas para controle de versão, integração contínua e teste automatizado. As ferramentas serão definidas no início do projeto e devem ser seguidas durante todo o ciclo de vida do desenvolvimento.

6.3 Componentes de Terceiros

A integração com serviços externos, como o sistema de autenticação baseado em OAuth 2.0, é uma restrição do sistema. A equipe de desenvolvimento deve garantir a implementação correta dos fluxos de autorização e a manipulação segura de tokens de acesso.

6.4 Suporte a Navegadores

O sistema deve ser compatível com os principais navegadores da web, como Chrome, Firefox, Safari e Edge. As funcionalidades devem garantir uma experiência consistente para os usuários, independentemente do navegador escolhido.

6.5 Limites de Recursos

O sistema deve operar dentro dos limites aceitáveis de recursos, incluindo tempo de resposta para ações críticas e uso de memória durante operações rotineiras. Testes de desempenho devem ser conduzidos para garantir a conformidade com esses limites.

6.6 Requisitos de Hardware

O software deve ser projetado para ser compatível com dispositivos de entrada comuns, como teclados, mouses e telas sensíveis ao toque. A interface do usuário deve ser adaptada para garantir uma interação eficiente em diferentes dispositivos.

6.7 Compatibilidade com Dispositivos

A interface do usuário deve ser responsiva e garantir uma experiência consistente em dispositivos de diferentes tamanhos e resoluções de tela. Testes de compatibilidade devem ser realizados para garantir que a interface seja funcional em dispositivos variados.

6.8 Segurança de Dados

A segurança dos dados do usuário é uma prioridade. Todas as informações sensíveis, como senhas e dados pessoais, devem ser armazenadas de forma segura, utilizando técnicas adequadas de criptografia. O acesso não autorizado deve ser estritamente impedido.

6.9 Documentação

Deve ser fornecida uma documentação abrangente para usuários e desenvolvedores, detalhando o funcionamento do sistema e os procedimentos de uso. A documentação deve ser mantida atualizada e refletir com precisão o estado atual do sistema.

6.10 Escalabilidade

O sistema deve ser projetado levando em consideração a escalabilidade. Deve ser capaz de suportar um número crescente de usuários, quadros e cartões sem comprometer significativamente o desempenho. O planejamento de capacidade deve ser considerado para acomodar o crescimento futuro.

6.11 Adaptação e Atualizações

O sistema deve ser projetado para facilitar a adaptação e atualizações para acomodar futuras melhorias. A compatibilidade com diferentes versões de bibliotecas e módulos de software reutilizáveis deve ser considerada para garantir uma evolução contínua do sistema.

7. Conformidade do Sistema

7.1 Requisitos de Licenciamento

O sistema será desenvolvido e distribuído sob uma licença de código aberto, como a Licença MIT. Os requisitos de licenciamento incluem a obrigação de incluir a declaração de licença em todas as distribuições do software e garantir que as modificações no código fonte sejam devidamente documentadas.

7.2 Avisos legais, de direitos autorais e outros

O software deve incluir avisos legais apropriados, incluindo disclaimers de responsabilidade e garantias. Deve ser incluída uma notificação de direitos autorais indicando os proprietários dos direitos do software. Além disso, devem ser fornecidas informações sobre qualquer patente ou marca registrada associada.

7.3 Padrões Aplicáveis

- Segurança de Dados: O sistema deve aderir aos padrões de segurança de dados estabelecidos pela ISO 27001.
- Compatibilidade com Navegadores: A interface do usuário deve seguir as diretrizes estabelecidas pelo W3C para garantir a compatibilidade com os principais navegadores.
- Padrões de Desenvolvimento Web: O desenvolvimento web seguirá as práticas recomendadas do OWASP para garantir a segurança da aplicação.
- Arquitetura: Adoção de uma arquitetura modular, como arquitetura hexagonal ou arquitetura limpa, para separar as preocupações do negócio e da infraestrutura.

8. Documentação do Sistema

8.1 Documentação do Usuário

- Será fornecida uma documentação abrangente para os usuários, abordando tópicos como registro de conta, autenticação, criação e gestão de quadros e cartões, além de outros serviços oferecidos pelo sistema.
- A documentação do usuário será apresentada de forma clara e concisa, utilizando terminologia acessível a usuários com diferentes níveis de experiência.
- A responsabilidade pela criação e manutenção da documentação do usuário será atribuída à equipe de desenvolvimento, com revisões regulares para garantir a precisão e relevância das informações.

8.2 Sistema de Ajuda Online

- Um sistema de ajuda online será implementado para oferecer suporte contextual aos usuários enquanto interagem com o sistema.
- A ajuda online incluirá informações detalhadas sobre funcionalidades específicas, guias passo a passo e solução de problemas comuns.
- A equipe de desenvolvimento será responsável por manter a ajuda online atualizada de acordo com as atualizações do sistema.

8.3 Avisos de Ajuda

- A interface do usuário incluirá avisos de ajuda que fornecerão informações contextuais sobre elementos específicos, funcionalidades ou possíveis ações.
- Os avisos de ajuda serão redigidos de maneira clara e objetiva, auxiliando os usuários na compreensão rápida e eficiente do sistema.

8.4 Responsabilidade pela Documentação

- A equipe de desenvolvimento será responsável por criar, manter e atualizar toda a documentação do sistema.
- Revisões periódicas serão conduzidas para garantir que a documentação esteja alinhada com as alterações no sistema e que continue sendo uma fonte confiável de informações para os usuários.

8.5 Versionamento da Documentação

- A documentação será versionada de acordo com as versões do software, garantindo que os usuários possam acessar informações relevantes para a versão específica que estão utilizando.
- As versões anteriores da documentação serão arquivadas e permanecerão disponíveis para referência, se necessário.