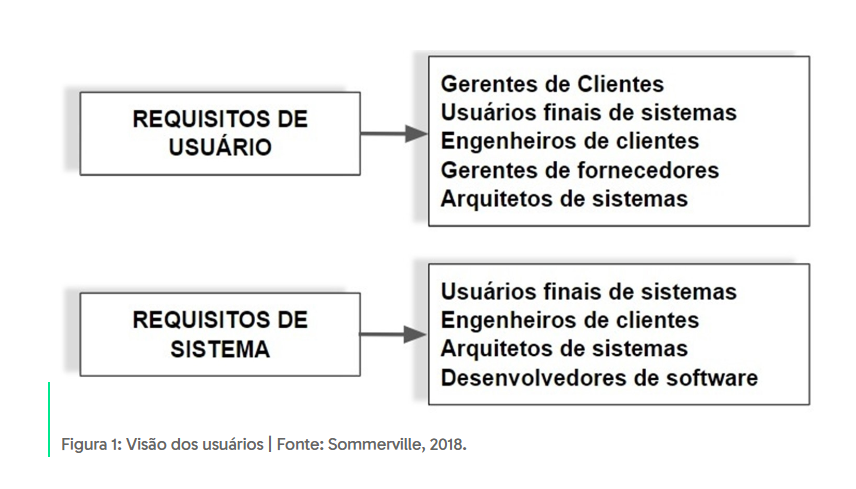
1. **Análise de requisitos:**

Nesta etapa, os requisitos para os produtos a serem projetados são obtidos a partir da interação com o cliente e as partes interessadas. Perguntas como “Quem pode usar o produto?”, “Qual é o objetivo do produto?”, “Que tipo de dados entra no produto?” e “Que tipo de dados é retornado pelo produto como saída?” são todas respondidas nesta etapa. Um documento de especificação de requisitos é criado.

* 1. **Requisitos Funcionais:**

Representam as necessidades que o sistema deve prover, de que forma as entradas são executadas para geração das saídas, assim como o que o sistema não deve fazer. Dizemos também que os requisitos funcionais representam os procedimentos que realizam o negócio, independente da tecnologia. Por exemplo: em um sistema de vendas, temos os seguintes requisitos funcionais:

RF01 - Vender produto

RF02 - Cadastrar cliente

RF03 - Cadastrar produto

RF04 - Cadastrar Vendedor

RF05 - Emitir Nota Fiscal

RF06 - Geral ranking de melhores clientes

* 1. **Requisitos não Funcionais:**

**Representam restrições adotadas para definição do como será implementado o sistema.**

**São aqueles que definem características que devem ser incluídas no sistema sob o foco do Produto, Organizacional e Externos**

**Linha do tempo

Descrição gerada automaticamente**

**Linha do tempo

Descrição gerada automaticamenteDiagrama, Linha do tempo

Descrição gerada automaticamente**

**Diagrama

Descrição gerada automaticamente**

* Mapeamento de stakeholders;
* Mapeamento de necessidades pontuadas pelos usuários;
* Mapeamento de necessidades percebidas sobre os usuários;
* Mapeamento de objetivos e escopo do software;
* Mapeamento de possíveis soluções ferramentais de UX;
* Mapeamento de dados de entrada, processamento dos dados e dados de saída;
* Criação de fluxo de usabilidade via wireframe ou esboço;
* Estudo de viabilidade de propostas sugeridas;
* Criação de proposta final de UI Design

**Estudo de viabilidade:**

Nesta etapa, a análise das condições para realizar o projeto é feita. A viabilidade técnica, econômica, operacional e legal do projeto é avaliada. Os riscos, benefícios e custos do projeto são estimados.

* Avaliar a viabilidade técnica do projeto, considerando as tecnologias, as ferramentas, os recursos humanos e os prazos necessários para o desenvolvimento.
* Avaliar a viabilidade econômica do projeto, estimando os custos, os benefícios, o retorno sobre o investimento e o ponto de equilíbrio do projeto.
* Avaliar a viabilidade operacional do projeto, analisando as necessidades e as expectativas dos usuários, os processos envolvidos, os riscos potenciais e as medidas de mitigação.
* Avaliar a viabilidade legal do projeto, verificando as normas, as leis, os regulamentos e as questões éticas que podem afetar o projeto.
* Elaborar um relatório de viabilidade com os resultados da análise, as conclusões e as recomendações para a tomada de decisão.

**Design do projeto**:

Nesta etapa, o design do produto é criado a partir da documentação de requisitos. O design define a arquitetura, os componentes, as interfaces, os módulos e as funcionalidades do software. O design também especifica os padrões de codificação, teste e documentação a serem seguidos.

* Revisão dos requisitos: Entender completamente a documentação de requisitos antes de iniciar o design.
* Definição dos objetivos do design: Ter clareza sobre o que o design precisa alcançar em termos de funcionalidades e arquitetura.
* Design das interfaces: Definir as interfaces entre os componentes, módulos e sistemas externos.
* Detalhamento dos módulos: Descrever cada módulo/componente em detalhes, incluindo suas responsabilidades e funcionalidades.
* Especificação das funcionalidades: Documentar claramente as funcionalidades que cada módulo deve oferecer.
* Definição das estruturas de dados: Projetar as estruturas de dados necessárias para suportar as funcionalidades do software.
* Padrões de codificação: Definir padrões de codificação a serem seguidos por toda a equipe para garantir consistência e legibilidade do código.
* Diretrizes de teste: Especificar as diretrizes para os testes que serão realizados durante o desenvolvimento e após a conclusão do projeto.
* Estratégia de documentação: Planejar como a documentação do código e do design será feita, garantindo que seja completa e atualizada.
* Revisão do design: Realizar revisões técnicas do design com a equipe para garantir a qualidade e identificar possíveis melhorias.
* Considerações de desempenho: Levar em conta aspectos de desempenho e otimização no design do projeto.
* Tratamento de erros e exceções: Definir como os erros e exceções serão tratados pelo software.
* Considerações de segurança: Incluir medidas de segurança relevantes no design, como autenticação, autorização e proteção contra vulnerabilidades conhecidas.
* Compatibilidade: Garantir que o design leve em conta a compatibilidade com diferentes sistemas operacionais, navegadores ou plataformas, se aplicável.
* Requisitos não funcionais: Assegurar que os requisitos não funcionais, como desempenho, usabilidade e escalabilidade, estejam considerados no design.
* Plano de evolução do design: Considerar como o design será adaptado para futuras mudanças e atualizações.
* Controle de versão: Definir a estratégia e ferramentas de controle de versão a serem usadas para o código e documentação do design.
* Revisão final do design: Realizar uma revisão final do design antes de prosseguir para a fase de implementação.

**Codificação:**

Nesta etapa, o código-fonte do software é escrito de acordo com o design e os padrões definidos. As ferramentas e linguagens de programação adequadas ao tipo, tamanho e objetivo do software são escolhidas. O código-fonte é organizado em unidades lógicas e documentado.

**Teste:**

Nesta etapa, o software é testado para verificar se ele atende aos requisitos e funciona corretamente. Os testes podem ser realizados em diferentes níveis, como unidade, integração, sistema e aceitação. Os testes também podem ser classificados em funcionais, não funcionais, estruturais e orientados a defeitos. Os testes visam identificar e corrigir erros, falhas e defeitos no software.

**Implantação:**

Nesta etapa, o software é instalado e configurado no ambiente do cliente. O software pode ser implantado em diferentes formas, como pacote, download ou serviço web. O software também pode ser implantado em diferentes estágios, como alfa, beta ou produção. A implantação envolve atividades como treinamento, suporte e feedback dos usuários.

**Manutenção:**

Nesta etapa, o software é monitorado e atualizado para garantir seu funcionamento adequado e sua satisfação dos usuários. A manutenção pode ser preventiva, corretiva, adaptativa ou evolutiva. A manutenção envolve atividades como correção de erros, melhoria de desempenho, adição de novas funcionalidades e adaptação a novas tecnologias ou requisitos.