# Ciclo de Vida de Desenvolvimento de Software

## Definição

O ciclo de vida de desenvolvimento de software é um conjunto estruturado de etapas, organizadas de forma sequencial ou iterativa, que orienta desde a concepção até a entrega e manutenção de um sistema. Para o projeto em questão — uma plataforma de chamados com inteligência artificial aplicada ao suporte técnico — foi adotado o Scrum, um framework ágil amplamente reconhecido na indústria de software para gerenciamento de projetos complexos.

O Scrum é um modelo iterativo e incremental que permite a entrega contínua de funcionalidades em ciclos curtos e controláveis, denominados sprints. Esse método proporciona alta flexibilidade para adaptação às mudanças, além de promover a melhoria contínua e o foco constante no valor entregue ao usuário.

## Etapas do Scrum no Projeto

Product Backlog: Representa a lista priorizada de funcionalidades e requisitos do sistema, incluindo abertura de chamados, classificação automatizada por IA, base de conhecimento (FAQ), painel do usuário, sistema de feedback, entre outros.

Sprint Planning (Planejamento da Sprint): Reunião onde são definidos os itens do backlog que serão trabalhados em cada sprint, com base na prioridade e capacidade da equipe.

Sprint: Período de desenvolvimento contínuo, geralmente entre 1 e 4 semanas, durante o qual uma parte funcional do sistema é implementada e preparada para entrega.

Daily Scrum: Reuniões diárias rápidas, de aproximadamente 15 minutos, para alinhamento da equipe, identificação de obstáculos e acompanhamento do progresso das atividades.

Sprint Review: Momento de apresentação dos resultados alcançados na sprint para stakeholders e membros do projeto, possibilitando feedback e validação das entregas.

Sprint Retrospective: Sessão interna de avaliação do processo da sprint, focada em identificar oportunidades de melhoria para os ciclos seguintes.

## Justificativa da Escolha do Scrum

A escolha do Scrum para o desenvolvimento do sistema de chamados com IA baseia-se em diversas vantagens que esse framework oferece:

Iteratividade e Entregas Frequentes: O sistema demanda testes constantes com usuários reais, bem como ajustes frequentes para aprimorar a inteligência artificial, características que se alinham perfeitamente com o desenvolvimento iterativo do Scrum.

Adaptação Rápida a Mudanças: Como os requisitos podem evoluir conforme o feedback do grupo de suporte técnico e da instituição, o Scrum permite incorporar essas mudanças de forma ágil, sem comprometer o planejamento geral.

Colaboração Contínua: O framework promove comunicação constante entre os membros da equipe, estimulando o engajamento e a divisão equilibrada das tarefas.

Melhoria Contínua: A evolução incremental do sistema, especialmente no que tange à IA, é facilitada pelo foco do Scrum na melhoria contínua ao longo dos sprints.

Visibilidade e Controle: As reuniões e entregas periódicas proporcionam transparência no andamento do projeto, permitindo que gestores e equipe façam ajustes tempestivos sempre que necessário.

### Ferramentas Utilizadas no Ciclo de Vida

Durante o desenvolvimento do projeto, foram utilizadas duas ferramentas principais para apoiar o ciclo de vida:

* **Kanban**: Implementado como quadro visual para organizar as tarefas, dividindo-as em colunas como "A Fazer", "Em Andamento" e "Concluído". Isso permitiu uma visão clara do progresso e facilitou o controle das atividades de cada membro da equipe.
* **GitHub**: Utilizado como repositório de código-fonte e controle de versões. Por meio do GitHub, o time pôde manter um histórico das alterações, trabalhar de forma colaborativa, revisar códigos e documentar funcionalidades em issues e commits.

Essas ferramentas foram integradas à metodologia Scrum para garantir maior organização, rastreabilidade, colaboração e transparência durante todo o ciclo de desenvolvimento.

## Referências Bibliográficas

SCHWABER, Ken; SUTHERLAND, Jeff. Scrum Guide – O Guia do Scrum: A regra do jogo. Scrum.org, 2020.

SOMMERVILLE, Ian. Engenharia de Software. 10. ed. São Paulo: Pearson Education do Brasil, 2019.

PRESSMAN, Roger S. Engenharia de Software: uma abordagem profissional. 8. ed. São Paulo: AMGH Editora, 2016.