**UNIVERSIDADE PAULISTA**

Projeto Integrado Multidisciplinar

4º e 3º Período – 2025/2

Daniel Ricardo Mayer- R105FD0

Diogo Trindade Torres Carvalho - G911598

Gabriel José Cézar - G2674I0

Vitor Henrique Machado Torres- G99BFG1

**Gerenciamento De Projetos De Software.**

Este estudo visa examinar e utilizar ferramentas digitais na gestão de projetos, enfatizando suas funcionalidades, princípios e aplicações.

**Miro:**

Miro é uma ferramenta de colaboração visual amplamente utilizada por equipes que buscam otimizar seus processos de brainstorming, planejamento e execução de projetos. Com uma interface intuitiva e recursos que permitem a criação de quadros, mapas mentais e diagramas, Miro se destaca como uma solução eficaz para facilitar a comunicação e a criatividade em ambientes de trabalho remoto ou híbrido. A plataforma é especialmente popular entre profissionais de design, marketing e desenvolvimento de produtos, que necessitam de um espaço dinâmico para compartilhar ideias e feedbacks.

**Princípios do Miro:**

O Miro é uma ferramenta de colaboração visual que facilita o trabalho em equipe de forma interativa. Seus principais princípios são a colaboração em tempo real, permitindo que várias pessoas trabalhem juntas no mesmo quadro; a visualização de informações, com uso de mapas mentais, fluxogramas e post-its digitais; e a flexibilidade, já que pode ser aplicado em diferentes contextos como projetos, design e metodologias ágeis. Além disso, destaca-se pela integração com outras ferramentas, pela inclusividade, permitindo a participação de todos, e pelo engajamento, com recursos como votação e comentários. Por ser escalável, atende desde pequenos grupos até grandes organizações.

**Uso do Miro em projetos:**

O uso do Miro nos projetos está ligado a facilitar a organização e a colaboração da equipe em todas as fases do trabalho. Ele serve como um quadro branco digital onde é possível reunir ideias, planejar etapas, mapear processos e acompanhar o andamento das atividades.

Na prática, o Miro é usado para:

* Brainstorming e ideação: coletar ideias com post-its virtuais e organizar contribuições de todos.
* Planejamento de projetos: criar cronogramas, roadmaps e fluxos de trabalho.
* Metodologias ágeis: rodar cerimônias como retrospectivas, daily meetings e planejamento de sprints.
* Mapeamento de processos e jornadas: visualizar etapas de um projeto, jornada do cliente ou fluxos internos.
* Colaboração remota: unir equipes em tempo real, mesmo que cada membro esteja em um local diferente.
* Documentação visual: centralizar informações do projeto de forma clara e acessível.

**Trello:**

O Trello é uma ferramenta de organização baseada em “quadros virtuais”. Pense nele como uma grande parede onde você pode colar notas, fazer listas e arrumar tudo do jeito que preferir. Ele é usado por pessoas no mundo todo, tanto para coisas pessoais, como organizar tarefas de casa, quanto para grandes projetos, como os realizados em empresas.

**Princípios do Trello:**

O **Trello** é uma ferramenta de gerenciamento de projetos baseada no método **Kanban**, que organiza tarefas de forma visual. Seus principais princípios são a **simplicidade** e a **organização**, utilizando quadros, listas e cartões para estruturar o fluxo de trabalho. Ele valoriza a **colaboração em tempo real**, permitindo que várias pessoas acompanhem, comentem e atualizem atividades. Outro princípio importante é a **transparência**, já que todos os membros da equipe conseguem visualizar o andamento das tarefas. Além disso, o Trello permite **priorizar e acompanhar** as atividades com etiquetas, prazos e checklists, sendo flexível para diferentes tipos de projetos e integrando-se a diversas ferramentas digitais.

**Uso do Trello em projetos:**

O Trello é usado em projetos para organizar, acompanhar e gerenciar tarefas de forma visual e colaborativa. Ele permite criar quadros (projetos), listas (etapas do trabalho) e cartões (tarefas), facilitando a divisão de atividades, a atribuição de responsáveis e o controle de prazos.

Nos projetos, o Trello ajuda a:

* Planejar o trabalho de forma clara e estruturada.
* Acompanhar o andamento das tarefas em tempo real.
* Definir prioridades e prazos com etiquetas e checklists.
* Favorecer a colaboração entre os membros da equipe.
* Dar transparência ao progresso do projeto, deixando visível quem faz o quê

**Grafana:**

 Grafana é uma plataforma para visualizar e analisar métricas por meio de gráficos. Ele tem suporte para diversos tipos de bancos de dados — tanto gratuitos quanto pagos —, e pode ser instalado em qualquer sistema operacional.

Para facilitar a visualização dos gráficos, é possível criar dashboards dinâmicos que podem ser compartilhados com toda a equipe. Além disso, a ferramenta permite configurar alertas com base nas métricas, que são analisadas de forma contínua para notificar o usuário sempre que preciso, de acordo com as regras definidas por ele. É bastante utilizado por sistemas de monitoramento para gerar gráficos real-time.

**Princípios do Grafana:**

Os princípios do Grafana envolvem a visualização clara e interativa dos dados, a integração de múltiplas fontes de informação, a configuração de alertas em tempo real para monitoramento, a personalização dos dashboards com gráficos e filtros dinâmicos, e a colaboração, permitindo que equipes compartilhem análises e informações de forma eficiente.

**Uso de Grafana em Projetos:**

O Grafana é usado em projetos principalmente para monitoramento, análise e visualização de dados. Ele permite acompanhar o desempenho de sistemas, aplicações e infraestrutura em tempo real, identificar problemas rapidamente por meio de alertas, e apresentar informações de forma clara em dashboards interativos. Além disso, ajuda equipes a tomar decisões baseadas em dados, acompanhar métricas de projetos e compartilhar relatórios de desempenho de forma colaborativa.

**Slack:**

Slack é uma plataforma de comunicação assíncrona e síncrona para equipes, que permite mensagens em canais públicos ou privados, mensagens diretas, integração com outras ferramentas, compartilhamento de arquivos, notificações etc. É usado tanto em ambientes corporativos quanto acadêmicos para melhorar colaboração, coordenação e reduzir uso de e-mails.

**Princípios do Slack:**

O princípio do slack nas organizações defende que a utilização máxima de recursos, ou seja, operar constantemente no limite de 100%, é contraproducente, pois sistemas sobrecarregados não possuem capacidade para lidar com imprevistos, aprendizado ou inovação. A folga é definida como o tempo ou recurso intencionalmente não alocado, que permite à organização adaptar-se, aprender e responder de forma ágil a novas oportunidades ou crises. Nesse contexto, torna-se mais importante realizar as tarefas corretas (efetividade) do que executar com perfeição tarefas que não agregam valor real para a organização. Além disso, a metodologia enfatiza que os colaboradores devem dispor de tempo não preenchido para refletir, colaborar e tomar decisões estratégicas, e não apenas para executar atividades operacionais.

**Uso de Slack em Projetos:**

No contexto de gestão de projetos, o slack (ou folga) representa o tempo extra disponível em uma tarefa sem que isso afete a duração total do projeto ou o início de outra atividade. Ele funciona como uma margem de segurança, permitindo que atrasos pontuais em determinadas etapas sejam absorvidos sem comprometer o prazo final. O uso adequado do slack possibilita maior flexibilidade no cronograma, redução de riscos e maior capacidade de adaptação a imprevistos. Além disso, auxilia os gestores a identificarem quais tarefas são críticas (sem folga) e quais possuem margem de ajuste, otimizando a priorização de esforços.

**Porque usar o Trello?**

Escolhemos o Trello para o PIM por ser visual, simples e seguir o Kanban, facilitando acompanhar o trabalho e colaborar em tempo real. Teremos um único quadro com etapas claras, cartões padronizados e rótulos por tipo, área e prioridade. Automações ajudam a gerenciar o fluxo e alertar sobre prazos.

**PostgreSQL** **e pgAdmin**

O **PostgreSQL** é um sistema de gerenciamento de banco de dados relacional (SGBD) de código aberto, amplamente utilizado em ambientes acadêmicos e corporativos por sua confiabilidade e conformidade com os padrões SQL. De acordo com Silberschatz, Korth e Sudarshan (2019, p. 104), “o PostgreSQL é considerado um dos SGBDs mais poderosos do mercado, destacando-se por sua extensibilidade e por garantir integridade e segurança aos dados armazenados”.

Para facilitar a administração do banco de dados, é utilizado o **pgAdmin**, ferramenta oficial de gerenciamento do PostgreSQL. Trata-se de uma interface gráfica que permite executar consultas SQL, monitorar o desempenho, gerenciar usuários e realizar backup e restauração de dados. Segundo Souza e Pereira (2021, p. 58), “o pgAdmin é utilizado para administrar bancos PostgreSQL, por oferecer um ambiente intuitivo que reduz a complexidade da gestão de dados e aumenta a produtividade do administrador”.

No contexto do sistema de agendamento, o PostgreSQL é responsável por armazenar informações essenciais, como dados de pacientes, horários e registros de confirmações ou cancelamentos. Já o pgAdmin possibilita ao desenvolvedor e ao administrador manter o banco organizado

**Visual Studio**

**Visual Studio** é uma das IDE mais completas e utilizadas no mercado. IDE é criado pela Microsoft, que é um software que reúne várias ferramentas em um único aplicativo, ela auxilia o desenvolvedor desde a escrita do código até os testes, publicação e manutenção do programa, atualmente é um dos ambientes mais completo, tanto para um programador que está iniciando quanto para um mais experiente. IDE suporta várias linguagens de programações, como **C#, C++, Python, Javascript,** entre outros, assim podemos trabalhar com vários segmentos, por exemplo, de software. Além disso, o Visual studio suporta integração com serviços em nuvem, como o Microsoft Azure, poderíamos chegar a um software web que foi desenvolvido com ele e já publicado e pronto para escalar.

Alguns de seus recursos são **IntelliSense** um comando completo, depurador integrado, criação de projetos com várias linguagens e possível de fazer commits no repositório git. Qualidade se dá pelo fato de no Software ser possível instalar outros Soft chamados de extensões e plugins, que tornam o software mais **flexível** conforme a equipe e o projeto.

**A linguagem C#**

C# (C Sharp) é uma linguagem de programação desenvolvida pela **Microsoft no início dos anos 2000** como parte da plataforma. É uma combinação de linguagem de **modernidade,** de força de tipagem, de **orientação a objetos** e de características muito mais produtivas e simples na comparação de Java e C++... Segundo Macoratti (2012, p. 5), “C# é uma linguagem elegante e orientada a objetos, projetada para criar aplicações robustas e seguras, possuindo suporte a múltiplos paradigmas de programação e integração nativa com o .NET Framework”.

Conforme Balena (2006, p. 3), “a linguagem C# foi concebida para ser simples e poderosa, fornecendo ao desenvolvedor tanto a facilidade de uso quanto os recursos necessários para aplicações corporativas complexas”. Dessa forma, a linguagem se consolidou como uma das principais escolhas no mercado para o desenvolvimento de sistemas **desktop, web e mobile**.

**Programação orientada a objetos**

O modelo de **programação orientada a objetos (POO)** representa uma maneira de construir software estruturando o código em entidades como classes e objetos, estabelecendo uma ponte entre a lógica da programação e cenários do cotidiano. Essa abordagem viabiliza maior organização, reaproveitamento e facilidade na hora de dar manutenção ao código. Segundo Pessôa Filho (2024, p. 12), “a programação orientada a objetos busca aproximar o desenvolvimento de software ao mundo real, representando entidades através de classes e objetos, permitindo reutilização e maior clareza no código”.

Dentre as linguagens que seguem essa abordagem, o C# (C Sharp) sobressai, uma criação da **Microsoft**, bastante empregada na elaboração de programas para Windows, internet e até mesmo jogos. Alba (2015, p. 27) explica que “C# é uma linguagem moderna, orientada a objetos e fortemente tipada, desenhada para oferecer simplicidade e produtividade no desenvolvimento de aplicações de software”.

**Desenvolvimento de aplicações desktop**

No universo da criação de softwares para computadores, a linguagem C# se destaca como uma das ferramentas mais abrangentes para o desenvolvimento de soluções corporativas. Sua vasta utilização na construção de sistemas desktop é notável, especialmente através das plataformas **.NET Framework**. Essa preferência é justificada pela solidez da linguagem, sua tipagem rigorosa e a vasta gama de bibliotecas que simplificam a interação com bancos de dados, a geração de relatórios e o tratamento de informações em tempo real. Segundo Troelsen e Japikse (2017, p. 34), “C# fornece uma plataforma estável e moderna para construção de aplicações que variam desde programas simples de linha de comando até soluções empresariais complexas”. Essa versatilidade torna a linguagem especialmente adequada para softwares de agendamento, controle de clientes e sistemas administrativos em geral.

Além disso, criar aplicativos desktop para Windows usando C# com **Windows Forms e** WPF permite desenvolver visuais fáceis de usar. Isso significa que o usuário consegue usar o programa de forma simples, e o programador consegue criar tudo de forma ágil. De acordo com Liberty (2008, p. 56), “o Windows Forms simplifica o processo de criação de aplicações desktop, fornecendo componentes prontos que permitem ao programador se concentrar na lógica do negócio”. Assim, no contexto de um sistema de gerenciamento de agendamentos, o **C#** aliado ao Windows Forms representa uma solução eficiente, equilibrando facilidade de desenvolvimento e experiência satisfatória para o usuário.

Para criar o sistema que organiza os agendamentos, Utilizei **Windows Forms**. Essa decisão foi motivada por três pontos cruciais: a facilidade em colocar tudo em prática, a conexão direta com o banco de dados e a agilidade em criar as telas do programa. Segundo Alba (2015, p. 102), “o Windows Forms se destaca por sua facilidade de uso e pela ampla quantidade de componentes visuais prontos, o que permite ao desenvolvedor criar aplicações completas de forma rápida”. Assim, essa abordagem possibilita que o sistema seja entregue em menor tempo, sem abrir mão da qualidade e da estabilidade.

**APIs e ASP.NET**

O **ASP.NET** é um dos principais frameworks da Microsoft para o desenvolvimento de aplicações **web** e **serviços de API**, sendo totalmente integrado à plataforma .NET. Sua utilização permite criar sistemas robustos, seguros e escaláveis, facilitando a comunicação entre diferentes aplicações e dispositivos.

amplia as funcionalidades de sistemas desktop, permitindo que serviços externos sejam consumidos ou disponibilizados. No caso do sistema de gerenciamento de agendamentos, uma API foi implementada com o objetivo de **enviar e-mails automáticos aos pacientes** sempre que um agendamento é confirmado ou cancelado. Essa funcionalidade garante maior agilidade na comunicação e melhora a experiência do usuário final, reduzindo falhas humanas no processo de notificação, assim, ao integrar a API ao sistema desktop desenvolvido em Windows Forms, é possível unir a praticidade de uma aplicação local com a eficiência de serviços conectados à internet, garantindo que os pacientes recebam informações em tempo real sobre seus compromissos.

**Model-View-Controller (MVC)**

A arquitetura **Model-View-Controller (MVC)** é amplamente utilizada no desenvolvimento de sistemas web por proporcionar uma clara **separação de responsabilidades** entre dados (Modelo), interface (Visão) e lógica de controle (Controlador), o que favorece a organização, manutenção e testes da aplicação. De acordo com estudo publicado na **Revista Eletrônica Científica de Ciência da Computação (RE3C)**, “a arquitetura MVC se mostra cada vez mais presente no ambiente de desenvolvimento para Web. Devido à sua facilidade de criar testes unitários, facilidade de manutenção e maior controle do código gerado, a arquitetura MVC se torna uma das principais arquiteturas de hoje em dia” este padrão, quando utilizado no **framework ASP.NET MVC**, torna possível distribuir funcionalidades como entrada, processamento e saída de forma organizada e modular

**Referências**

ALBA, V. *C# e .NET: desenvolvimento de aplicações*. São Paulo: Novatec, 2015.  
PESSÔA FILHO, Joaquim. *Programação orientada a objetos com C#*. São Paulo: Senac São Paulo, 2024

LIBERTY, Jesse. *Aprendendo C# 3.0*. Rio de Janeiro: Alta Books, 2008.

MACORATTI, José Carlos. *C# e Orientação a Objetos*. São Paulo: Clube dos Autores, 2012.

SOUSA, Roberto; COSTA, Mariana. *Ambientes de Programação e Ferramentas de Desenvolvimento*. Rio de Janeiro: Ciência Moderna, 2020.

SILVA, André Luiz da. *Introdução ao Desenvolvimento de Software com Visual Studio*. São Paulo: Érica, 2018.

TROELSEN, Andrew; JAPIKSE, Philip. *Pro C# 7: With .NET and .NET Core*. New York: Apress, 2017.

BALENA, Francesco. *Programando em C#: guia do desenvolvedor*. São Paulo: Pearson Education, 2006.

SANTOS, Isaías et al. Possibilidades e limitações da arquitetura MVC (Model – View – Controller) com ferramenta IDE (Integrated Development Environment). *Revista Eletrônica Científica de Ciência da Computação (RE3C)*, Unifenas, 2010.

OLIVEIRA, Marcelo. *Ferramentas de Desenvolvimento e Integração Contínua*. Porto Alegre: Bookman, 2019.

MIRO. *Miro: Collaborative whiteboard platform for distributed teams.* Disponível em: https://miro.com/. Acesso em: 11 set. 2025.

TRELLO. *Trello: Organize anything, together.* Disponível em: https://trello.com/. Acesso em: 11 set. 2025.

GRAFANA. *Grafana: Open platform for analytics and monitoring.* Disponível em: https://grafana.com/. Acesso em: 11 set. 2025.

SLACK. *Slack: Where work happens.* Disponível em: https://slack.com/. Acesso em: 11 set. 2025.

DE MARCO, Tom. Slack: Getting Past Burnout, Busywork, and the Myth of Total Efficiency. 1. ed. Nova York: Broadway Books, 2001.