



# BIBLIOTECH



APRESENTAÇÃO FINAL – ENGENHARIA DE SOFTWARE

**Universidade Federal do Tocantins**

**Curso: Ciência da Computação**

**Professor: Edeilson Milhomem**

**Grupo 1: Guilherme Thomaz, Ítalo Henrik, Jhennifer da Silva,  
Luiz Felipe da Paz e Marcos Freire**

**BIBLIOO.TECH**

# Parte 1: Experiência na disciplina

**Universidade Federal do Tocantins**

**Curso: Ciência da Computação**

**Professor: Edeilson Milhomem**

**Grupo 1: Guilherme Thomaz, Ítalo Henrik, Jhennifer da Silva,  
Luiz Felipe da Paz e Marcos Freire**



# Importância das atividades Apg-1 e Apg-2

## Apg-1

---

- Permitiu aprender o fluxo de trabalho (**feature -> develop -> release -> main**) em um ambiente controlado, com a ideia do **gitflow**.
- Forçou a criação de um **README.md** de qualidade, essencial para a documentação de qualquer projeto.
- Primeiro contato prático com a padronização de código em equipe.

## Apg-2

---

- A prototipagem nos ajudou a visualizar o produto e a alinhar as expectativas da equipe.
- A escrita de **User Stories** nos força a pensar na perspectiva do usuário final, agregando valor real ao produto.
- O planejamento das **Sprints** (Iterações 1 e 2) foi nosso primeiro exercício de estimativa e quebra de tarefas complexas em partes gerenciáveis.

# APRENDIZAGEM BASEADA EM PROJETOS E METODOLOGIAS ÁGEIS

*Desenvolvemos habilidades além do código: negociação, planejamento e comunicação.*

**Entregas de Valor:** Foco em entregar features funcionais ao final de cada iteração, em vez de um "tudo ou nada" no final.

**Adaptação a Mudanças:** Adotamos um planejamento adaptativo, onde cada nova etapa do projeto foi diretamente informada pela experiência e pelo feedback da etapa anterior.



# Lições Aprendidas

A Importância de se ter uma comunicação clara na equipe.

Como planejar o desenvolvimento de um produto real.

Aprendemos a usar o Git de forma profissional e também a importância de seguir um padrão de código para manter a organização.

# Desafios Lidados

## Gerenciamento de tempo e prazos

Como conciliar as tarefas do projeto com as demais disciplinas. Houve momentos em que o ritmo desacelerou por falta de tempo, e isso exigiu reorganização e redistribuição das tarefas.

## Validação constante do sistema

Testar o sistema a cada nova feature foi desafiador, pois mudanças em uma parte do código às vezes afetavam outras funcionalidades. A validação manual exigia atenção constante.



# Pontos Positivos da Disciplina

## Experiência de Mercado

A disciplina simulou um ambiente real de desenvolvimento ágil, uma vivência que vale mais que muitas aulas teóricas.

## Atuação em Times

Vivência real em trabalho em equipe com divisão de tarefas e responsabilidades

## Processo Ágil

A aplicação prática de Sprints, User Stories e planejamento iterativo nos deu uma base sólida.

# Pontos Negativos da Disciplina

## Dificuldade em aplicar padrões desde o início

---

Um ponto negativo que identificamos foi o fato de que, por conta da estrutura da disciplina, não conseguimos adotar desde o início padrões fundamentais como o MVC e a arquitetura baseada em APIs. Como esses elementos estavam previstos apenas para iterações futuras isso dificultou o planejamento técnico inicial.

A ausência desses padrões exigiu reestruturações posteriores para adequar o projeto às boas práticas. Acreditamos que, se esses conceitos fossem incentivados desde o início, teríamos uma base mais sólida para crescer o sistema de forma mais fluida e consistente.

# APRESENTAÇÃO DO PRODUTO

## BIBLIOTECH



The screenshot shows the homepage of the BiblioTech website. At the top, there is a navigation bar with the BiblioTech logo, followed by links for "Funcionalidades", "Sobre", "Contato", and a green "Começar Agora" button. The main heading "Sua Biblioteca Digital Completa" is displayed prominently. Below it, a descriptive text reads: "Descubra milhares de livros digitais e físicos em uma plataforma moderna e intuitiva. Leia, compre e organize sua biblioteca pessoal de forma simples e acessível." Two buttons are present: a green "Criar Conta Grátis" button and a white "Explorar Funcionalidades" button. The bottom section features a heading "Por que escolher o BiblioTech?" followed by three icons: a book icon labeled "Catálogo Extenso", a smartphone icon labeled "Leitor Integrado", and a shopping cart icon labeled "Compra Simplificada".

**BiblioTech**

Funcionalidades Sobre Contato Começar Agora

## Sua Biblioteca Digital Completa

Descubra milhares de livros digitais e físicos em uma plataforma moderna e intuitiva. Leia, compre e organize sua biblioteca pessoal de forma simples e acessível.

Criar Conta Grátis Explorar Funcionalidades

### Por que escolher o BiblioTech?

**Catálogo Extenso**

**Leitor Integrado**

**Compra Simplificada**

# APRESENTAÇÃO DO PRODUTO

# BIBLIOTECH

## Nossas Principais Funcionalidades



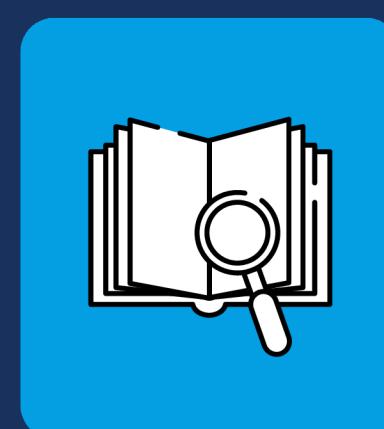
### Jornada de Compra Completa

- Navegação e busca inteligente no catálogo.
- Carrinho de compras e lista de desejos.
- Processo de checkout simples



### Leitura Integrada

- Leitor de eBooks nativo, sem precisar sair da plataforma.



### Experiência Pós-Venda

- Biblioteca pessoal com todos os livros adquiridos.
- Histórico detalhado de pedidos.



### Gerenciamento Completo

- Painel de controle para o perfil do usuário.
- Área administrativa para gestão total do catálogo.

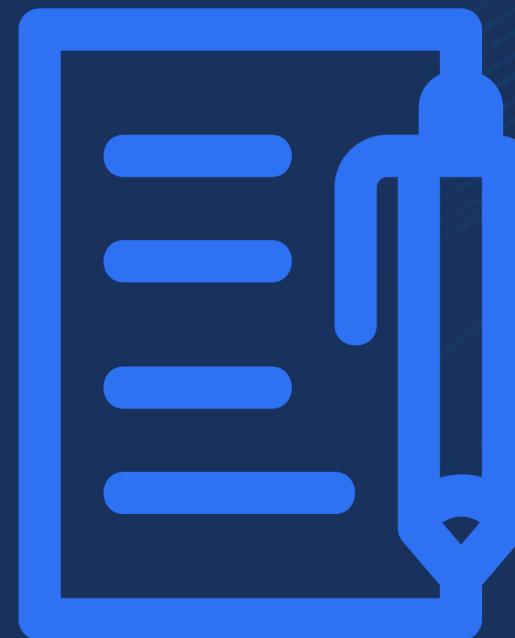
# Parte 2: Teste de conhecimento

**Universidade Federal do Tocantins**

**Curso: Ciência da Computação**

**Professor: Edeilson Milhomem**

**Grupo 1: Guilherme Thomaz, Ítalo Henrik, Jhennifer da Silva,  
Luiz Felipe da Paz e Marcos Freire**



## ESTRATÉGIAS PARA ENTENDER O PROBLEMA E OS REQUISITOS

Para entender o **problema** e seus **requisitos**, conversaríamos com o cliente, faríamos **entrevistas, reuniões** e criariamos representações visuais como **fluxogramas**. Isso nos ajudaria a captar todas as necessidades, até mesmo as não explícitas.

## DEFINIÇÃO DOS REQUISITOS PRIORITÁRIOS

Definir prioridades, classificando os requisitos com base no valor que trazem para o cliente e a viabilidade técnica. Assim, saberíamos o que entregar primeiro.

## VALIDAÇÃO DOS REQUISITOS

Validaríamos os requisitos por meio de protótipos e critérios de aceitação. Essas validações seriam feitas com o cliente antes do desenvolvimento, garantindo alinhamento total.

## DEFINIÇÃO E VALIDAÇÃO DAS TECNOLOGIAS

A escolha das tecnologias seria baseada em testes rápidos, experiência da equipe, suporte e escalabilidade. Validaríamos a escolha com provas de conceito quando necessário.

## IMPORTÂNCIA DO PLANEJAMENTO DAS ITERAÇÕES (SPRINTS).

O planejamento por iterações, como sprints, é essencial para organizar o trabalho, permitir entregas frequentes e facilitar ajustes com base no feedback do cliente.

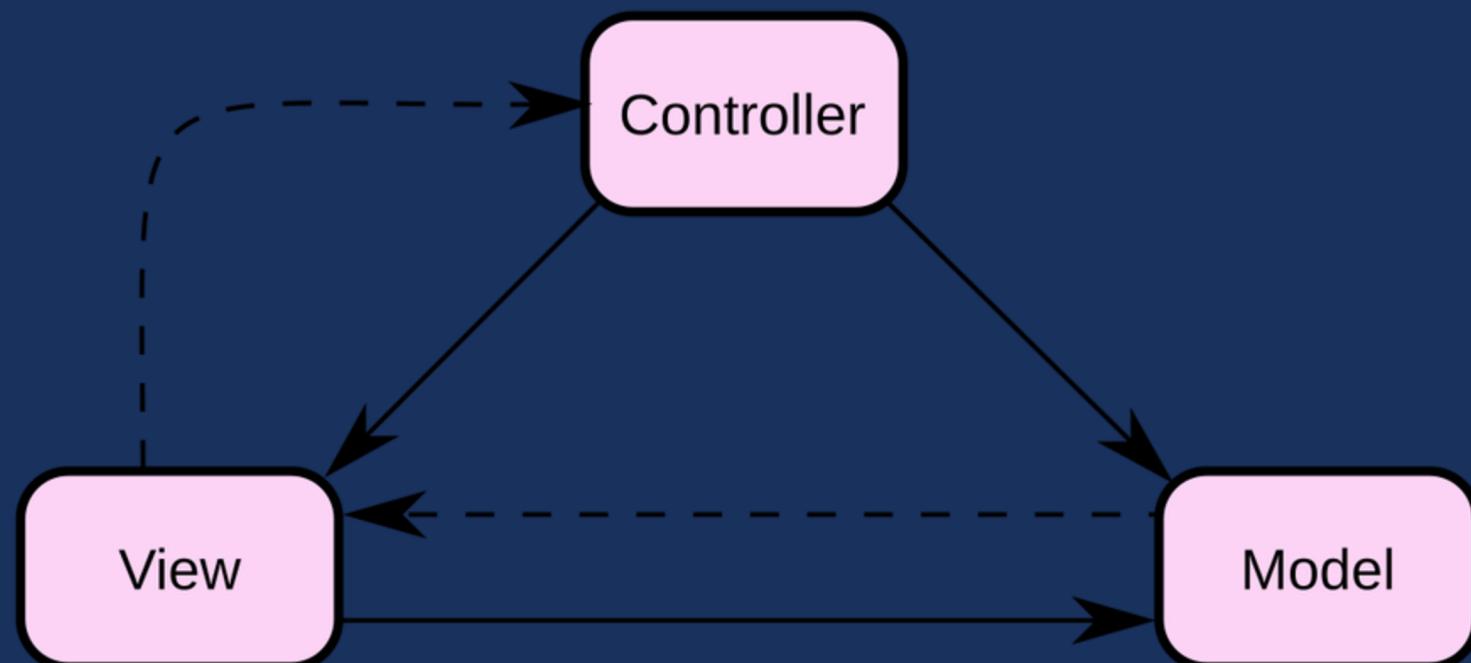


## IMPORTÂNCIA DA DEFINIÇÃO DE VALOR EM UMA ITERAÇÃO

Definir o valor em cada iteração garante foco nas funcionalidades mais relevantes e permite que o cliente veja progresso real ao longo do desenvolvimento.

## IMPORTÂNCIA DE SEGUIR PADRÕES (EX: MVC).

Seguir padrões como o MVC traz organização, facilita a manutenção do código e melhora o trabalho em equipe, já que separa bem as responsabilidades.



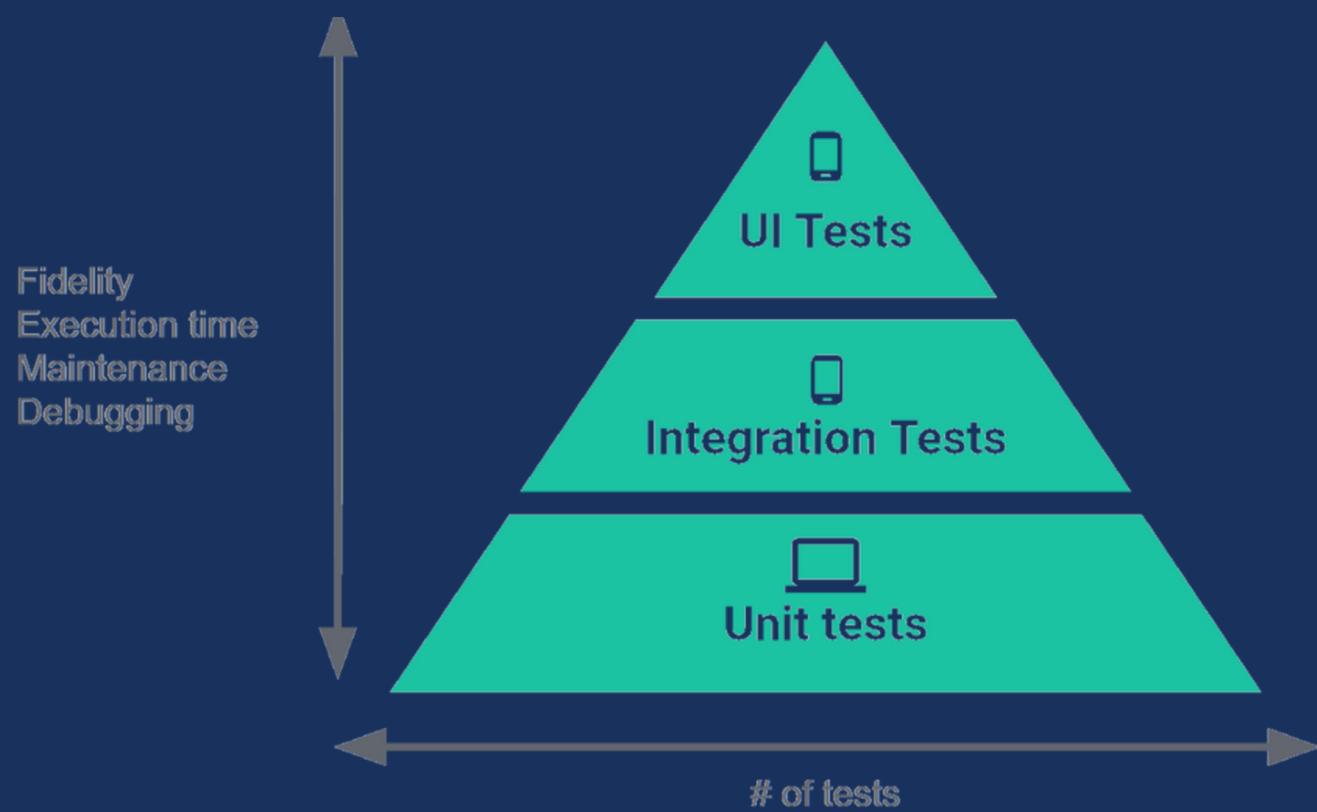
## IMPORTÂNCIA DA ARQUITETURA BASEADA EM SERVIÇOS (API REST).

APIs REST permitem modularidade, integração com outros sistemas e crescimento do projeto de forma organizada e reutilizável.



## IMPORTÂNCIA DE TESTES UNITÁRIOS

Testes unitários garantem que cada parte do sistema funcione corretamente, ajudam a detectar erros cedo e trazem segurança durante mudanças no código.



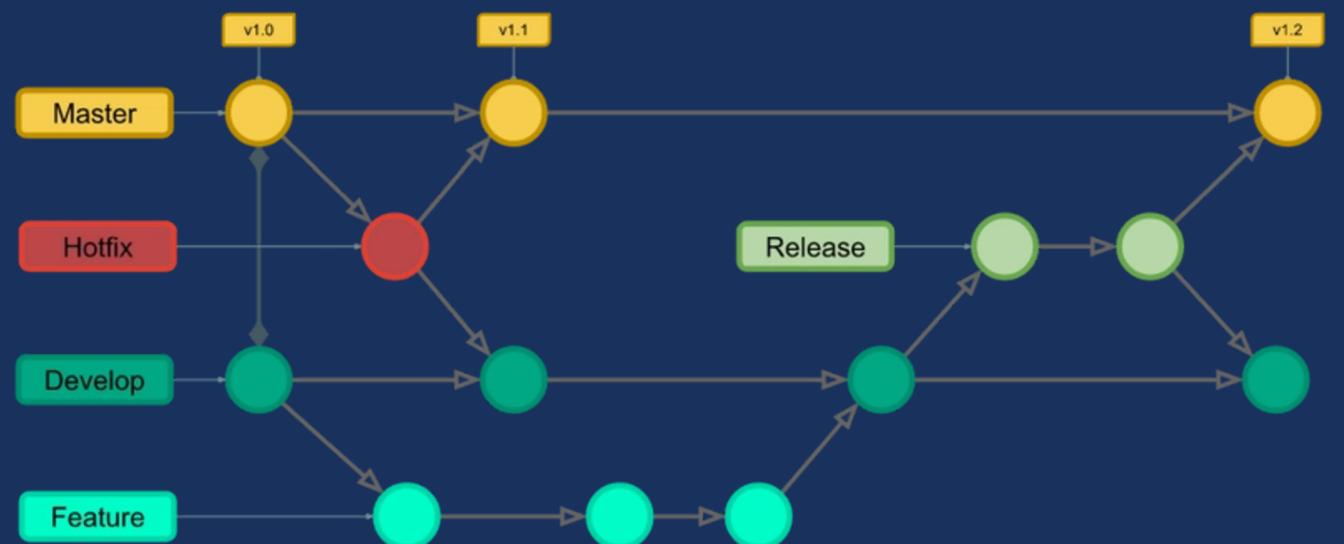
## IMPORTÂNCIA DO CONTROLE DE VERSÃO E COMO É REALIZADO

É vital para gerenciar o histórico de alterações e permitir o trabalho em equipe. É realizado com sistemas como o Git, que controla as versões localmente, e plataformas como o GitHub, que centraliza o código e facilita a colaboração.



## VISÃO SOBRE O FLUXO DE TRABALHO DO GITFLOW

O GitFlow organiza o trabalho em branches bem definidos, como **develop**, **feature** e **main**, facilitando o fluxo do projeto e evitando erros.



## ENTENDIMENTO POR RELEASE DO PRODUTO

Uma release é uma versão estável do produto, pronta para ser entregue ao cliente. Ela representa uma entrega concreta, validada e funcional.



# BIBLIOTECH



OBRIGADO PELA ATENÇÃO

BIBLIOO.TECH