**1. Exploração inicial**

* Reunião com os donos/gestores da academia para entender necessidades:
* Cadastro de alunos e planos.
* Controle de pagamentos.
* Agenda de treinos e aulas coletivas.
* Relatórios (presença, inadimplência, etc.).
* Criação de **histórias de usuário** (ex.: “Como aluno, quero ver meu treino pelo app para seguir corretamente o plano do professor”).

**2. Planejamento**

Estimativa rápida de esforço para cada funcionalidade.

Priorização com o cliente:

* 1ª iteração → funcionalidades básicas de cadastro e check-in.
* 2ª iteração → controle de pagamentos.
* 3ª iteração → treinos personalizados e agenda.

**3. Práticas-chave do XP aplicadas**

* **Programação em pares**: dois devs trabalham juntos no mesmo código para aumentar qualidade.
* **Propriedade coletiva do código**: todos podem alterar qualquer parte do sistema.
* **Integração contínua**: a cada funcionalidade feita, o código é integrado e testado.
* **Refatoração constante**: melhorar o código sem mudar sua função, garantindo manutenibilidade.
* **Testes automatizados**: cada história de usuário gera testes que devem passar antes da entrega.
* **Feedback rápido**: ao final de cada iteração (1 a 2 semanas), o cliente da academia avalia o que foi entregue.

**4. Execução em ciclos curtos**

* Iterações de **1 a 2 semanas**.
* No fim de cada ciclo → entrega de uma versão funcional:
  + - **Exemplo 1**: versão inicial com cadastro de alunos e check-in na recepção.
    - **Exemplo 2**: controle de mensalidades e relatórios de pagamentos.
    - **Exemplo 3**: app do aluno com treinos e agendamento.

**5. Benefícios no projeto da academia**

* Rápida adaptação se os donos pedirem mudanças (ex.: mudar regras de pagamento ou incluir integração com Pix).
* Entregas frequentes permitem uso real cedo, sem esperar o sistema “todo pronto”.
* Código sempre testado e de qualidade, reduzindo riscos de falhas no dia a dia da academia.

**Lens:**

**1. Identificar o Valor**

No projeto da academia, o **valor** é definido em conjunto com os donos/gestores na "Exploração inicial". O valor são as funcionalidades que eles estão dispostos a pagar e que resolvem os problemas deles: cadastro, controle de pagamentos, agenda de treinos e relatórios. A **criação de histórias de usuário** é a forma como o time de desenvolvimento concretiza esse valor.

**2. Mapear o Fluxo de Valor**

O fluxo de valor é o caminho que uma funcionalidade percorre, desde a ideia inicial até a entrega. No seu exemplo, o fluxo é bem claro:

1. **Ideia:** "Preciso de um sistema para gerenciar alunos."
2. **Fluxo de Trabalho:** A ideia é quebrada em histórias de usuário.
3. **Processamento:** A equipe trabalha nessas histórias em ciclos curtos (iterações).
4. **Entrega:** Uma versão funcional é entregue ao cliente ao final de cada iteração.
5. **Feedback:** O cliente da academia avalia o que foi entregue.

**3. Criar Fluxo Contínuo**

Aqui é onde as práticas do XP se encaixam perfeitamente nos princípios do Lean. A meta é garantir que o trabalho flua sem interrupções (desperdícios). No seu exemplo, isso é feito através de:

* **Integração Contínua:** Evita que o trabalho de diferentes desenvolvedores se torne um gargalo. Ao integrar e testar o código frequentemente, o fluxo não é interrompido por grandes conflitos de código.
* **Refatoração Constante:** Mantém o código "limpo" e fácil de dar manutenção, evitando futuros desperdícios de tempo com correções complexas.
* **Programação em Pares:** Duas pessoas trabalhando juntas podem resolver problemas mais rapidamente, evitando que o trabalho pare por causa de um obstáculo.

**4. Estabelecer Produção Puxada**

A "Produção Puxada" é evidente no seu item de "Planejamento" e "Execução em ciclos curtos". A equipe não começa a trabalhar em todas as funcionalidades ao mesmo tempo. Em vez disso, ela **"puxa"** as funcionalidades do backlog (lista de tarefas) para a primeira iteração apenas quando está pronta para trabalhar nelas. Isso garante que o trabalho só comece quando há capacidade, evitando o excesso de produção e o acúmulo de tarefas incompletas.

**5. Buscar a Perfeição**

O Lean vê o processo como algo a ser melhorado continuamente. No seu projeto, isso é feito com:

* **Testes Automatizados:** Não apenas garantem a qualidade, mas também permitem que a equipe faça mudanças e refatore com confiança, sem medo de introduzir erros.
* **Feedback Rápido:** Ao final de cada iteração, o cliente da academia fornece feedback. Isso permite que a equipe aprenda e ajuste o curso do projeto, buscando um produto final que seja o mais próximo possível da perfeição.