

# Pesquisa de Mercado e Viabilidade do Produto PROJETO: ReelView API – Plataforma de Gestão de Mídia

**Contexto e Problema Identificado** No cenário atual, o consumo de filmes e séries está fragmentado entre múltiplos serviços de streaming (Netflix, Prime Video, etc.). Essa descentralização gera duas dores principais nos usuários: a dificuldade de rastreamento do que já foi assistido e a falta de centralização de críticas confiáveis. O mercado carece de soluções leves, sem poluição visual e que ofereçam dados centralizados via API.

## Público-Alvo

- **Usuários Finais:** Cinéfilos que necessitam de uma ferramenta "Watchlist" organizada para catalogar filmes, atribuir notas e descobrir novos conteúdos baseados em rankings.
- **Desenvolvedores:** Devido à natureza do projeto (API RESTful), o ReelView serve como um *Backend-as-a-Service* para devs que desejam criar interfaces Web ou Mobile sem se preocupar com a complexidade de servidores e banco de dados.

## Análise da Concorrência

- **IMDb:** Extremamente completo, mas possui interface visualmente poluída, lenta e excesso de publicidade.
- **Letterboxd:** Muito popular, mas focado estritamente em interação social.
- **Trakt.tv:** Funcional, mas complexo para usuários casuais.

**O Diferencial:** O ReelView foca em performance e simplicidade, entregando dados essenciais de forma direta e segura, preenchendo a lacuna de uma solução leve e rápida.

**Solução Proposta e Diferenciais Técnicos** O projeto se destaca não apenas pela utilidade, mas pela engenharia de software aplicada:

- **Integridade de Dados (Integração TMDb):** O sistema consome a API pública do *The Movie Database*, garantindo que o catálogo nasça povoado com sinopses e imagens oficiais, sem depender de cadastro manual.
- **Segurança (Identity):** Uso de **BCrypt** para hash de senhas e **JWT** para autenticação *stateless*, garantindo segurança de nível industrial.
- **Arquitetura Escalável (Clean Architecture):** O desacoplamento entre camadas permite manutenção facilitada. O uso de **Programação Assíncrona (Async/Await)** garante que a API suporte múltiplos acessos simultâneos sem bloqueio de recursos.

**Conclusão** O ReelView é viável tecnicamente e mercadologicamente. Ele atende à demanda por organização de mídia pessoal utilizando uma stack tecnológica moderna (.NET, EF Core), oferecendo uma base sólida, segura e documentada para o consumo de informações cinematográficas.