

trabalho 6

ESTUDA++

Chloe Anne Scaramal

Gabriel Augusto Paiva

João Pedro Zanetti

Marcelo Gabriel Milani Santos

Marcos Antônio da Silva Junior

Paulo Daniel Forti da Fonseca

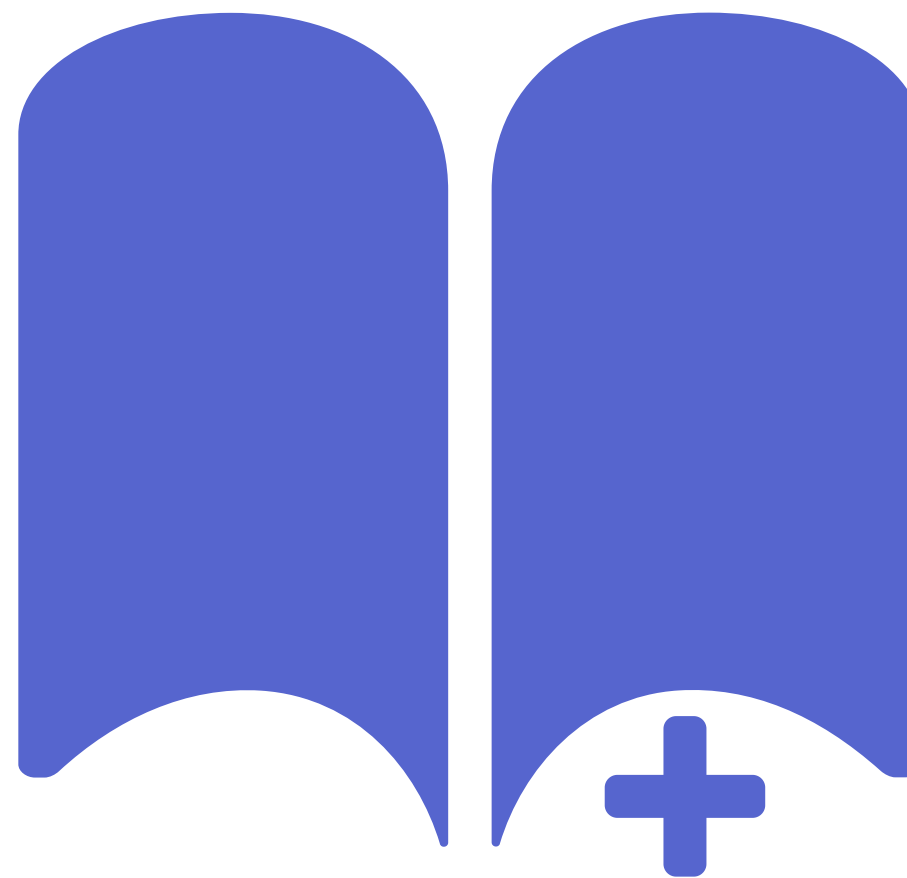
Pedro Henrique Lopes Duarte

Rayane Reis Mota

Vinicius Resende Garcia

OBJETIVO

O objetivo central no desenvolvimento do Estuda++ foi a aplicação prática de tecnologias de ponta e metodologias modernas de engenharia de software. O projeto serviu como um campo de provas para a construção de uma aplicação Android nativa utilizando um stack 100% Kotlin, desde o frontend com Jetpack Compose até ao backend com Ktor. Buscámos implementar uma arquitetura limpa e escalável (MVVM), gerir um ciclo de vida de desenvolvimento automatizado com CI/CD no Google Cloud, e integrar funcionalidades complexas como Inteligência Artificial e Geolocalização.

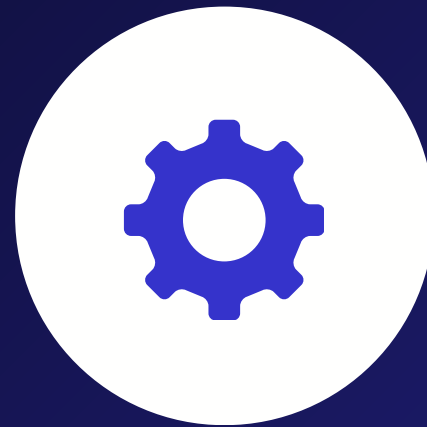


PONTOS PRINCIPAIS



UI Moderna e Declarativa

Interface declarativa e moderna, que acelera o desenvolvimento e melhora a manutenção dos ecrãs.



Arquitetura Robusta MVVM

Código limpo e organizado, com uma clara separação entre a lógica de negócio e a interface do utilizador.



Backend e Firebase

Plataforma robusta para autenticação de utilizadores e base de dados em tempo real, sem gerir servidores.

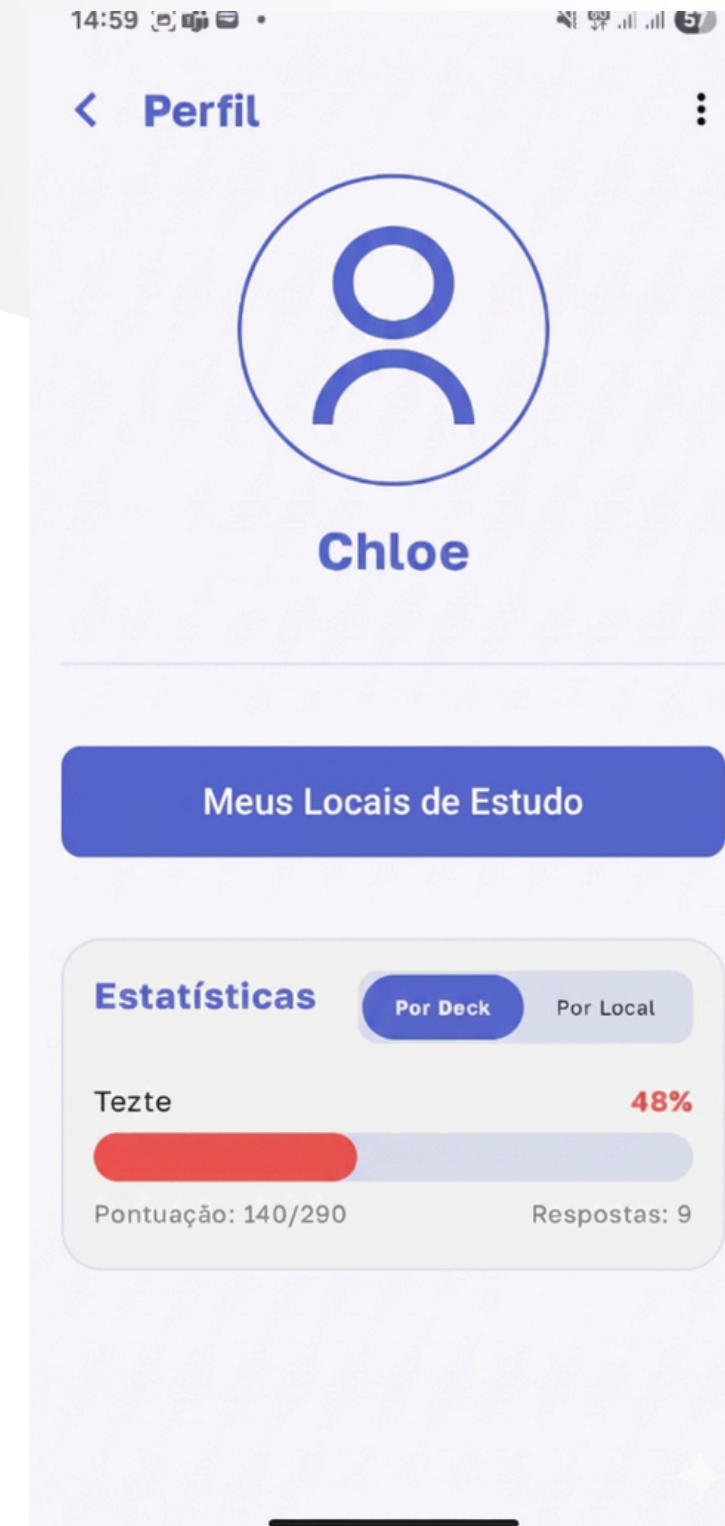


Serviços Inteligentes e API Própria

Uma API própria em Kotlin para orquestrar funcionalidades complexas como a integração com IA e Geolocalização.

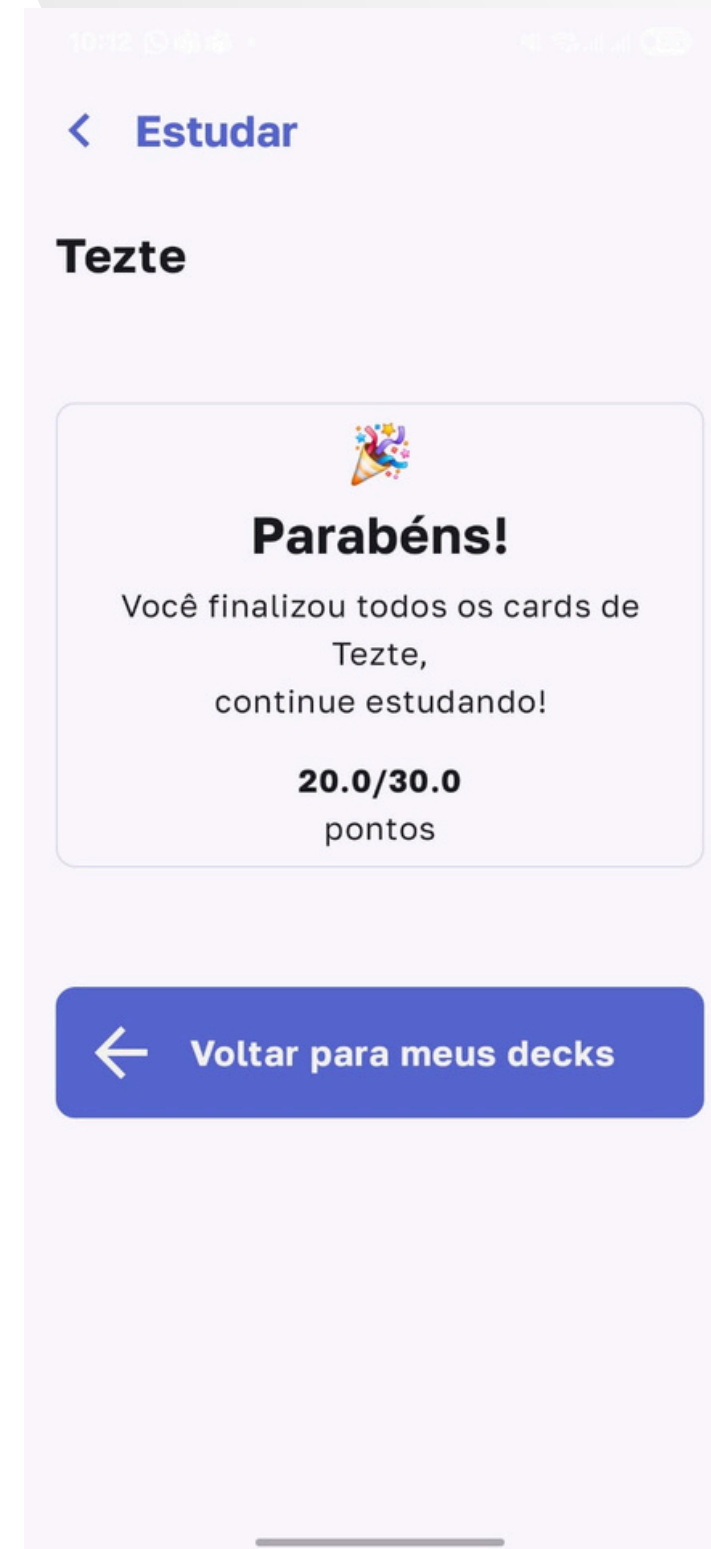
UI E A EXPERIÊNCIA DO USUÁRIO

- Ao estudar, a interface precisa de reagir instantaneamente ao toque do usuário para virar um cartão ou mostrar a próxima.
- Criação de flashcards adapta-se ao tipo que o utilizador escolhe
- O novo componente de estatísticas no perfil, com as suas barras de progresso e a alternância entre visões



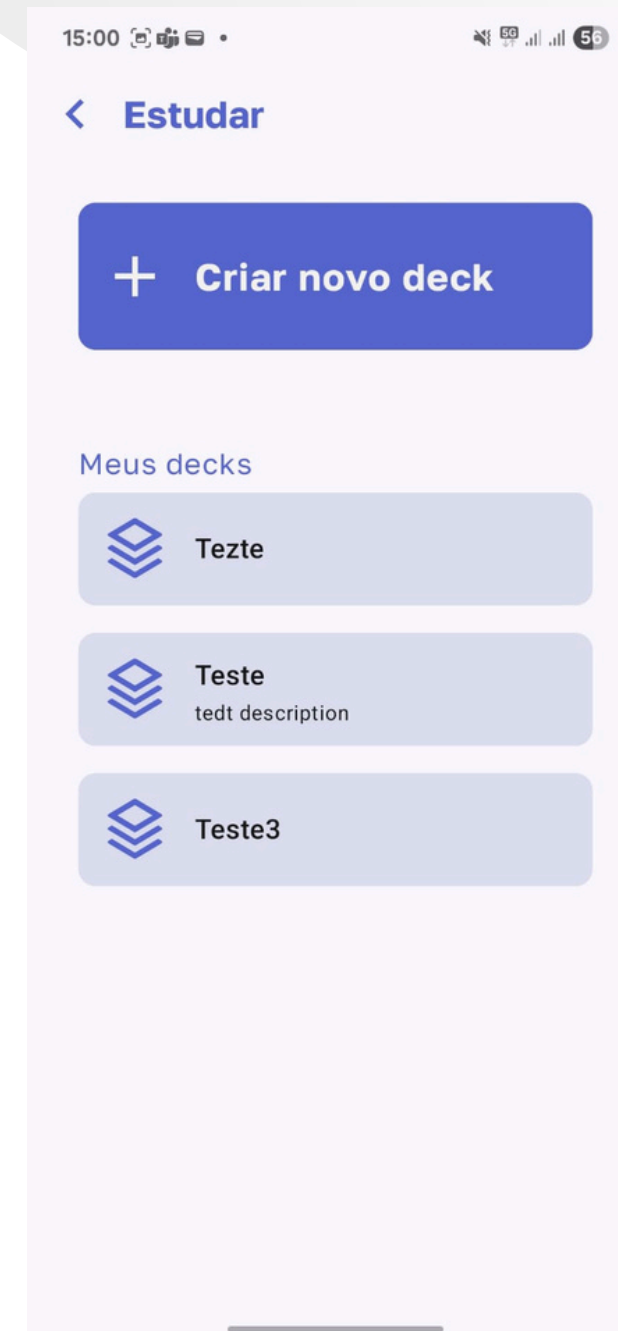
ARQUITETURA E A LÓGICA DO PROGRAMA

- A lógica de autenticação fica totalmente separada da interface.
- Quando você cria um novo deck ou flashcard, o DeckViewModel encarrega-se de o guardar no Firebase e de avisar a interface para que a lista seja atualizada automaticamente



FIREBASE E A SINCRONIZAÇÃO DOS DADOS

- Todos os baralhos e flashcards que o utilizador cria são guardados instantaneamente no Realtime Database. Ele pode começar a estudar no seu telemóvel e continuar mais tarde noutro dispositivo, com todos os seus dados perfeitamente sincronizados.
- As conversas com a MonitorIA são guardadas no Firebase. Assim, o utilizador pode sempre rever as perguntas que fez e as respostas



LLMS E ENGENHARIA DE PROMPT

-
- Gemini
 - Claude
 - magicPatterns (Gerar prova de conceito)

Foi essencial fornecer todo o contexto do projeto (arquivos, regras de negócio) para as LLMs.

Os prompts mais efetivos forneciam:

1. Um papel clara para a IA - "você é um desenvolvedor Kotlin" ou "você é um engenheiro de software"
2. Uma função objetiva e especificando quais tecnologias utilizar - "Utilize Jib para containilizar o app e implanta-lo em Cloud Run..."

MUITO
OBRIGADO!
