



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE ENSINO SUPERIOR DO SERIDÓ
BACHARELADO EM SISTEMAS DE INFORMAÇÃO.

LUCAS S.

RELATÓRIO DO TRABALHO DA UNIDADE III: NEUROQUIZ

CAICÓ - RN
2025

RESUMO

Este trabalho documenta o desenvolvimento do NeuroQuiz, um aplicativo móvel multiplataforma criado com React Native e Expo. O objetivo do projeto é gerar quizzes interativos e personalizados a partir de textos fornecidos pelo usuário, utilizando uma API de Inteligência Artificial. O desenvolvimento incluiu a modelagem de dados, a implementação de uma interface de usuário coesa e a configuração da navegação entre as telas. Para garantir a persistência dos dados do histórico, foi utilizada a solução de armazenamento local AsyncStorage. A funcionalidade central de geração de quizzes foi implementada através de uma chamada de acesso remoto à API do Google Gemini. O resultado é um protótipo funcional que valida o conceito do projeto e cumpre todos os requisitos de interface e armazenamento propostos.

Palavras-chave: React Native, Expo, Desenvolvimento Mobile, Inteligência Artificial, Gemini API, AsyncStorage.

ABSTRACT

This paper documents the development of NeuroQuiz, a cross-platform mobile application built with React Native and Expo. The project's objective is to generate interactive and personalized quizzes from user-provided text by leveraging an Artificial Intelligence API. The development process included data modeling, the implementation of a cohesive user interface, and the configuration of screen navigation. To ensure data persistence for the user's history, the AsyncStorage local storage solution was used. The core quiz generation feature was implemented through a remote access call to the Google Gemini API. The result is a functional prototype that validates the project's concept and fulfills all proposed interface and storage requirements.

Keywords: React Native, Expo, Mobile Development, Artificial Intelligence, Gemini API, AsyncStorage.

SUMÁRIO

1	Introdução	4
2	Modelo de Dados e Entidades	5
3	Arquitetura do Sistema	5
4	Design e Apresentação das Telas	6
5	Acesso Remoto: Integração com a API de Inteligência Artificial .	7
6	Implementação do Armazenamento Local	8
7	Conclusão	9

1 Introdução

O presente relatório detalha o processo de concepção, desenvolvimento e implementação do NeuroQuiz, um aplicativo móvel multiplataforma criado no âmbito da disciplina de Programação para Dispositivos Móveis. O projeto nasceu com o objetivo de otimizar a rotina de estudos, propondo uma solução inovadora que utiliza Inteligência Artificial para converter textos em questionários interativos de forma automática.

Utilizando o framework React Native e a plataforma Expo, foi construído um protótipo funcional completo. Este documento descreve as etapas cruciais do projeto, desde a modelagem inicial das entidades e casos de uso, passando pelo design da interface de usuário e a implementação da navegação entre telas, até a integração de funcionalidades essenciais como o armazenamento local de dados com AsyncStorage e o acesso remoto à API do Google Gemini para a geração dinâmica dos quizzes.

O trabalho a seguir não só apresenta o resultado técnico, mas também documenta os desafios encontrados e as soluções de arquitetura adotadas para construir uma aplicação estável.

2 Modelo de Dados e Entidades

As funcionalidades do NeuroQuiz são suportadas por um conjunto de entidades de dados que representam os conceitos centrais do sistema. As principais entidades definidas são:

- **User (Usuário):** Representa o usuário do aplicativo.
- **Quiz:** Modela um questionário completo.
- **Question (Pergunta):** Representa uma única pergunta dentro de um Quiz.
- **Answer (Resposta):** Modela uma única opção de resposta para uma pergunta.
- **QuizSession (Sessão de Quiz):** Armazena os dados de uma tentativa de um usuário ao responder um Quiz, formando o histórico de desempenho.

3 Arquitetura do Sistema

O diagrama a seguir ilustra visualmente a modelagem das entidades de dados e das classes de serviço que estruturam o sistema.

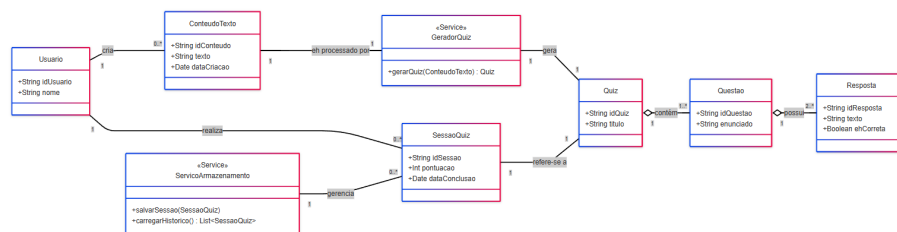


Figura 1 – Diagrama de Classes do Sistema NeuroQuiz.

4 Design e Apresentação das Telas

O projeto NeuroQuiz foi desenvolvido com foco em um design moderno e uma interface intuitiva, buscando oferecer ao usuário uma experiência de uso simples e direta. Para criar uma identidade visual consistente, todas as telas do aplicativo compartilham o mesmo plano de fundo, resultando em uma aparência coesa e profissional.

A fim de demonstrar a funcionalidade completa e a navegação da aplicação, foi produzido um vídeo que ilustra o fluxo de uso na prática. A demonstração em vídeo exhibe a interação do usuário desde a geração de um quiz com Inteligência Artificial até a consulta e gerenciamento do histórico de resultados.

O vídeo completo da aplicação em funcionamento está disponível no link a seguir: [Link para o Vídeo de Demonstração](#)

5 Acesso Remoto: Integração com a API de Inteligência Artificial

O projeto exigia a utilização de um serviço online. Esta funcionalidade foi implementada conectando o aplicativo a uma API de Inteligência Artificial, que é a responsável por criar os quizzes automaticamente.

A primeira tecnologia escolhida foi a API da OpenAI. A ideia era enviar um texto para a IA e receber um quiz pronto. No entanto, durante os testes, o acesso foi bloqueado por falta de créditos na conta, o que impediu o uso da ferramenta.

Para resolver o problema, o projeto foi adaptado para usar a API do Google Gemini, que oferecia um acesso gratuito mais adequado para o desenvolvimento.

Com essas alterações, a funcionalidade principal do aplicativo foi concluída. O NeuroQuiz agora se conecta com sucesso à API do Google Gemini para criar quizzes personalizados a partir do texto que o usuário fornece.

6 Implementação do Armazenamento Local

Para que o histórico dos quizzes ficasse salvo no celular, foi necessário implementar uma solução de armazenamento local. A abordagem para isso foi ajustada durante o desenvolvimento pois estava ocorrendo muitos erros.

A primeira escolha foi usar o SQLite, um banco de dados poderoso. O plano era criar tabelas para guardar os resultados de forma bem estruturada. No entanto, a instalação dessa ferramenta no projeto causou erros persistentes que impediam o aplicativo de funcionar, indicando um problema de compatibilidade com o ambiente de desenvolvimento.

Para resolver esse problema e garantir que o aplicativo funcionasse corretamente, a estratégia foi alterada.

A solução final foi usar o AsyncStorage, uma ferramenta de armazenamento mais simples e totalmente compatível com o projeto. Com ela, os dados do histórico são convertidos para um formato de texto (JSON) e salvos diretamente na memória do dispositivo. Essa abordagem funcionou perfeitamente, eliminou os erros de instalação e cumpriu o objetivo de salvar os dados do usuário de forma permanente.

7 Conclusão

Este projeto resultou na criação de uma versão inicial e funcional do aplicativo NeuroQuiz. Todas as metas propostas para a disciplina foram alcançadas, incluindo a construção de uma interface de usuário clara e uma navegação completa entre as diferentes telas.

Durante o desenvolvimento, surgiram desafios técnicos, como a necessidade de trocar a API de Inteligência Artificial e o método de armazenamento de dados. Esses problemas foram resolvidos com sucesso, trocando a API da OpenAI pela do Google Gemini e substituindo o banco de dados SQLite pelo AsyncStorage, o que garantiu o funcionamento do aplicativo.

Ao final, o aplicativo cumpre seus principais requisitos: salva o histórico de quizzes no aparelho e se conecta a um serviço online para criar os testes. O projeto agora serve como uma base sólida para futuras melhorias, como a criação de contas de usuário e novas funcionalidades.