

Relatório

4

Projeto Flask: Integração de Microsserviços.

Grupo 4: Alan Araújo da Silveira, Ana Cláudia Monteiro Misquita, Fernanda Aparecida Figueiredo da Silva, João Pedro Silva Antunes, Matheus Barros Ferreira

Data da atividade: 29/05/2025

Objetivo

Desenvolvimento de uma API (Policación Programming Interface) com utilização do framework Flask, para um sistema escolar, com funcionalidades para o gerenciamento de informações de alunos, turmas e professores. O sistema tem como objetivo fornecer uma interface eficiente para a criação, leitura, atualização e exclusão de dados (CRUD), permitindo uma administração mais ágil e precisa das informações escolares. Além disso, a API foi construída com a prática de Test-Driven Development (TDD), garantindo a qualidade e confiabilidade do código desde as primeiras fases de desenvolvimento. Na segunda entrega, foi estabelecido e

implementado o padrão MVC (Model - View - Controller), que organiza o código em camadas, facilitando a manutenção, a reutilização e o crescimento do projeto. Ele separa responsabilidades, deixando o desenvolvimento mais claro, eficiente e colaborativo.

2. Introdução

A API será desenvolvida com foco na integração de sistemas no contexto educacional, para organizar e acessar informações fundamentais a fim de melhorar a gestão e a comunicação entre alunos, professores e administradores. No quarto momento, o grupo realizou a integração de Microserviços. Para a criação das camadas, foi necessário clonar duas APIs do professor no nosso repositório do GitHub, tivemos que adaptar as apis em nosso código para poder deixá-las prontas para serem utilizadas.

3. Descrição e Análise do Caso

Inicialmente, quando nos foi apresentado detalhes de como seriam construídas as camadas, imaginamos que não haveria tantas dificuldades, visto que, a construção não exigia muitas alterações no ato de refatorar as APIs, no entanto, houveram alguns obstáculos enfrentados que tornaram o processo um tanto desafiador, mas que superados após muita análise, estudo, e algumas tentativas.

4. Implementação ou Procedimento

Os integrantes Alan e João Pedro iniciaram o processo de criação dos bancos de dados de atividades e reserva, que resumidamente, somente foi necessário criar o banco com as informações utilizadas no código. O processo foi bem simples ao criar os arquivos, o integrante Alan explicou que a parte mais dificultosa, foi fazer a conexão do banco de dados com os Models. O integrante Alan e João Pedro no desenvolvimento e procedimento de envio de arquivos para o Github, ao ajudar no

desenvolvimento. Após isso, partimos para a criação da segunda fase: MVC de reserva e atividades. Nesse momento todos os integrantes do grupo tiveram problemas sérios de comunicação (que após algumas conversas, foi declarado ter sido a maior adversidade durante o projeto como um todo), e por isso, a integrante Ana fez a criação da camada MVC de atividades e após finalizado e enviado, foi informada que a integrante Fernanda, também havia realizado a criação de reserva, assim somente restando fazer a conexão entre as APIs e implementação do Docker. No dia seguinte o integrante João Pedro ficou responsável ao implementar o Docker com as novas alterações e o integrante Matheus ficou responsável por fazer a conexão das APIs. Ao comentar sobre suas realizações e dificuldades no entregável 4, Alan, explica que corrigiu problemas que estavam acontecendo no banco de dados, corrigiu também os problemas do Models de atividades com o método POST, para que todo o ecossistema funcionasse da maneira correta.

5. Resultados

Apesar dos conflitos e dos obstáculos enfrentados, os objetivos finais do grupo foram alcançados, e a integração de Microsserviços no projeto, conseguimos superar as adversidades que vieram ao decorrer desse projeto, nos aprimoramos em relação de vários aspectos, principalmente na parte de codificação e refatoração dos códigos.

6. Conclusão

A implementação não foi tão simples quanto esperávamos, por conta dos problemas que tivemos quando implementamos as APIs de microsserviços, entretanto, tivemos êxito em nossos objetivos, conseguimos melhorar na parte de comunicação do grupo mas precisa melhorar em outros aspectos, com tudo podemos tirar muito conhecimento desse projeto.

7. Impacto e Conexão com o Mundo Real

Microserviços oferecem impacto e conexão com o mundo real ao permitir o desenvolvimento e implantação de aplicações de forma mais ágil, escalável e resiliente. Essa arquitetura facilita a entrega contínua de valor aos clientes, pois cada serviço pode ser atualizado e implantado de forma independente, sem interromper o funcionamento de outros. Microserviços também permitem que equipes trabalhem em paralelo em diferentes serviços, acelerando o ciclo de desenvolvimento e lançamento de novas funcionalidades.

8. Desafios Futuros e Melhorias

Acredita-se que o código é uma estrutura viva, e assim como tudo que é vivo, precisa ser constantemente transformada e melhorada. Por isso, recorrentemente, são feitas buscas para a otimização do desempenho dos testes, funções, Models, Controllers, mais para frente também será implementado a camada de Views, e assim por diante. Mas além disso, é de extrema importância a comunicação entre os integrantes do grupo e é de suma importância a evolução de cada um neste projeto ou em projetos futuros em outras áreas da TI.