



Universidade Federal de Pernambuco
Centro de informática

Graduação em Ciência da computação
Proposta de Trabalho de Graduação

Consistência em banco de dados noSQL Firestore a partir de linguagem tipada Typescript

Aluno

Thomas Anderson Feitosa Monteiro (tafm@cin.ufpe.br)

Orientador

Robson do Nascimento Fidalgo (rdnf@cin.ufpe.br)

Recife, junho de 2021

SUMÁRIO

1 Contexto	3
2 Objetivos	3
3 Cronograma	3
4 Possíveis avaliadores	3

1. Contexto

Um dos grandes problemas quando se trata de banco de dados noSQL diz respeito a consistência dos dados, dada a natureza em que os dados são armazenados de forma não estruturada. Um exemplo clássico é o MongoDB, que é comumente utilizado por programadores Node.js em conjunto com a biblioteca Mongoose, que permite definir o formato dos dados que serão armazenados nas coleções e dessa forma garante um mínimo de consistência nos vários trechos de código que precisarem armazenar dados na mesma coleção. Esse problema de consistência é amplamente discutido principalmente no contexto de microsserviços no qual várias aplicações diferentes precisam ler e escrever no mesmo banco noSQL e é ainda mais difícil manter essa padronização, entre serviços e até equipes diferentes de desenvolvimento. No Firestore - banco de dados noSQL do Google - assim como no MongoDB existe o mesmo desafio. O banco do Google é famoso quando se trata de construção de provas de conceito ou aplicações simples, pois a facilidade de se criar uma aplicação nova é enorme dado que o banco Firestore é 100% gerenciado e não é necessário lidar com problemas infraestrutura, escalabilidade, entre outros. Porém, à medida que a aplicação cresce, mais serviços são construídos e a complexidade aumenta, começam a aparecer rapidamente os problemas de consistência.

2. Objetivos

O objetivo geral deste trabalho é a construção de uma ferramenta que venha a auxiliar o desenvolvimento em Node.js utilizando Firestore a partir de geração de código para inserções e updates no banco, bem como mocks de estado do banco para auxílio na escrita de testes automatizados, demonstrando os benefícios em uma aplicação real. Os objetivos específicos são: (i) desenvolver ferramenta de auxílio para desenvolvimento de testes automatizados (ii) desenvolver ferramenta de geração de código para inserção e updates consistentes de dados; (iii) documentar e disponibilizar publicamente para uso geral no repositório de pacotes Node.js NPM

3. Cronograma

Atividade	Prazo
Desenvolver ferramenta de auxílio para desenvolvimento de testes automatizados	Junho.2, Junho.3
desenvolver ferramenta de geração de código para inserção e updates consistentes de dados	Junho.4, Julho.1
Documentar e disponibilizar publicamente para uso geral no repositório de pacotes Node.js NPM	Julho.2

Escrita da monografia	Julho.2, Julho.3, Julho.4, Agosto.1
-----------------------	--

4. Possíveis avaliadores

- Henrique Rebelo (hemr@cin.ufpe.br)
- Fernando Castor (fjclf@cin.ufpe.br)