

1. Introdução

Reescrever

2. Referencial teórico

2.1. Conceitos relacionados à gestão do conhecimento

2.1.1. Dado, informação e conhecimento

2.1.2. Gestão do conhecimento

2.2. Modelos de maturidade

2.2.1. Níveis de maturidade propostos por Philip Crosby

2.2.2. Padrões culturais de software propostos por Gerald Weinberg

2.2.3. Níveis de capacidade propostos pela norma ISO/IEC 15504

2.2.4. Os níveis de maturidade do CMMI

2.2.5. Modelo de maturidade DevOps

2.2.6. O modelo de representação do avanço do conhecimento SHU-HA-RI

2.3. Cynefin Framework

2.4. Teoria de Grafos

2.5. Trabalhos relacionados

~~3. Abordagem proposta~~

~~3.1. Cronograma~~

~~4. Potenciais aplicações~~

~~4.1. Grade Curricular~~

~~4.2. Sustentabilidade e gestão ambiental~~

~~4.3. Empreendedorismo~~

~~4.4. Música~~

3. O "produto" que foi criado

- *O que foi implementado.*
- *Como foi implementado.*

3.1. A aplicação

- *Descrição funcional da aplicação*
- Como acessar a aplicação
- O que ela é capaz de fazer
- Imagens das telas

3.2. Tecnologias envolvidas

- *Descrição técnica da aplicação*
- Repositório
 - GitHub
 - Licença MIT
- Hospedagem
 - Google Firebase
- Integração Contínua
 - Travis CI
- Código fonte
 - React
 - Estrutura geral do código
- Qualidade de código
 - Sonar
- Testes automatizados
 - Índice de cobertura

4. Resultados e perspectivas

- *Algoritmos de análise que foram implementados*
- *Novos algoritmos possíveis*

4.1. Implementado

- Grade do curso de Engenharia de Sistemas
 - Seleção de "disciplinas" a partir de restrições locais
- Modelo de maturidade DevOps
 - Seleção de "atividades" a partir de objetivos futuros

4.2. Perspectivas

- Algoritmos
 - Cálculo de vértices críticos
 - Cálculo de caminho crítico

- Aplicações em outras áreas
 - Educação infantil
 - Definições de habilidades de um profissional

5. Considerações finais

Reescrever