


### JUSTIFICATIVAS

Passado

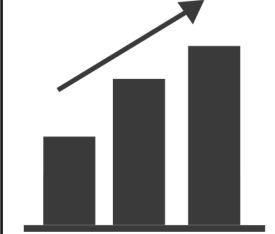
Desenvolvimento de software tradicional é complexo, demorado e repetitivo.

Falta de ferramentas que facilitem a parte inicial de um projeto, ou seja, sua estruturação, criação de entidades e elaboração de CRUDS



### OBJ SMART

Lançar em 3 meses uma solução baseada em Langium que automatize a geração de código e documentação para projetos Backend, reduzindo em 10% o tempo de desenvolvimento em equipes de software.



### BENEFÍCIOS

Futuro

Geração automática de código Backend (Python, Java, C#).


Documentação integrada (README, Swagger, Diagramas UML).

Configuração simplificada via linguagem própria.

Suporte a múltiplas tecnologias (Django, GraphQL, Docker, etc.).

Facilidade para decisão da estrutura do projeto.

Automatização de tarefas repetitivas (Criação de CRUDS)



### PRODUTO

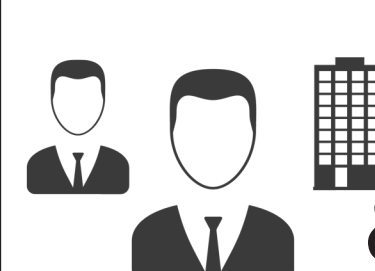
Extensão para VS Code que permite criar arquivos .Spark e gerar:

- Código Backend (Python, Java, C#).
- Documentação (README, Swagger).
- Diagramas UML (classes, pacotes).
- Configurações de infraestrutura (Docker, GitLab CI/CD).



### REQUISITOS

O spark deve conseguir gerar CRUDs em uma linguagem específica (dita no arquivo .spark que serve como entrada da ferramenta) de forma já estruturada.



### STAKEHOLDERS EXTERNOS


& Fatores externos

Stakeholders:

- Equipe de desenvolvimento backend do LEDS

Fatores Externos:

- Concorrência (ex.: outras ferramentas de low-code).
- Evolução de tecnologias Backend.




### EQUIPE

Desenvolvedores de software:  
Arthur Valentim Caldeira, Bruno Alves de Moraes, Caio Chiabai de Oliveira, Diego Bourguignon Rangel, Lucas Gabriel Araújo Codeco, Murilo dias de Oliveira.

Gerentes de projeto (Scrum Master):  
Alex Rossoni Junior, Davi Breda, João Pedro Pagotto da Costa, João Victor Vieira Rangel, João Victor Lopes Lopes, Thalison Vinícius Euzebio Ribeiro.


Product Owner: LEDS



### RESTRIÇÕES

Dependência do ecossistema VS Code.

Tempo curto para desenvolvimento.



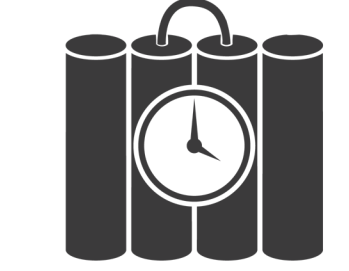
### PREMISSAS

Os usuários têm conhecimento básico de VS Code e linguagens Backend.

O mercado demanda ferramentas de automação para desenvolvimento.



### GRUPO DE ENTREGAS



### RISCOS

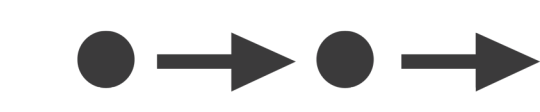
Baixa adoção devido à curva de aprendizado.  
- Criar tutoriais em vídeo e documentação interativa.

Limitações na cobertura de casos de uso complexos.  
-Modularizar a ferramenta para permitir extensões personalizadas.  
-Coletar feedback contínuo dos usuários.

Concorrência com ferramentas estabelecidas.  
- Diferenciação por foco em integração (código + doc+ diagramas)  
-Parcerias com comunidades de devs (GitHub, StackOverflow)

Curto período para finalizar o desenvolvimento do projeto.  
-Priorizar MVP com funcionalidades essenciais (gerador de código + swagger)  
- Adotar metodologias ágeis.

Dependência do marketplace de extensões do VS Code  
-Planejar versão standalone (CLI) futuramente  
- Publicar em múltiplas plataformas (GitHub, OpenVSX).



### LINHA DO TEMPO

Fase 1 (1-2 semanas): Imersão/On-boarding

Fase 2 (3-4 semanas):


Fase 3 (5-6 semanas):

Fase 4 (7-8 semanas):

Fase 5 (9-10 semanas):

Fase 6 (11-12 semanas):

Fase 7 (13-14 semanas):



### CUSTOS

Tempo de desenvolvimento da extensão e linguagem.

Tempo de produção de documentação e materiais de treinamento.

R\$ 100.000,00