# Necessidades

* representação visual dos metadados (eg. tamanho icones, estilo icones PK&FK, disposição espacial)
* metadados table schema (eg. metadados do schema para cada recurso e busca/filtro/ocultar colunas, mostrar detalhes, amostra dos dados)
* diagramas estáticos
* diagramas interativos (eg. tooltip descrição, visibilidade colunas, foco)
* metadados de validação
* metadados recursos vs schema (eg. particionamento)
* metadados estatísticas descritivas (eg. amostra dos dados, médias, % valores ausentes, etc)
* diagramas interativos e responsivos ao schema (eg. [crossfilter](https://emilyriederer.github.io/demo-crosstalk/analysis/demo.html))
* exportar para PDF
* exportar diagramas para SVG

# Premissas estimativas

* ANALISTA DE REQUISITOS, DE TESTES, DE DADOS, serão tratados como DESENVOLVEDOR para fins de estimativa
* LÍDER TÉCNICO E ARQUITETO DE SOFTWARE serão tratados como LIDER TECNICO
* Fazer estimativas separadas para:
  + gerador de site estático
  + customização do CKAN
* Na estimativa, considerar que todas as histórias do produto “gerador de site estático” será executado em primeiro lugar, e depois as do produto “customização CKAN”

# Links Material de Referência

* [Edital Registro de Preço](http://www.compras.mg.gov.br/images/stories/ArquivosLicitacoes/SEPLAG/consultaspublicas/sei-govmg-29234207-padrao-termo-de-referencia-rp-de-servico-desenvolvimento-e-sustentacao-de-sistemas-com-metodos-ageis.pdf)
* [Especifição - Em elaboração](https://github.com/dados-mg/projeto-contrato/blob/main/dados-abertos-espec.md)
* [Datahub](https://datahub.io/core/gdp): Catalogo de dados que também é baseado nas especificações frictionless
* [dbdiagram](https://dbdiagram.io/home): A free, simple tool to draw ER diagrams by just writing code. Designed for developers and data analysts.
* [crosstalk](https://emilyriederer.github.io/demo-crosstalk/analysis/demo.html): Interatividade entre “gráfico” e tabela
* [ckanext-validation](https://github.com/frictionlessdata/ckanext-validation): CKAN extension for validating Data Packages using Table Schema.
* [components](https://github.com/frictionlessdata/components): Visual components for the Frictionless Data project in TypeScript/React.