Requisitos para o Portal 2a. geração

Maria Luiza Sanchez

Sumário

- Requisitos do Science Portal - 2a. geração
  - Tipos de Usuário
  - Requisitos de cada Tipo de Usuário
  - Estrutura do Science Portal - 2a. geração
    - Organização em Componentes
- Opções de Implementação
  - Estrutura Proposta de Software do LINEA
  - Onde Estamos ...
  - Depois ...

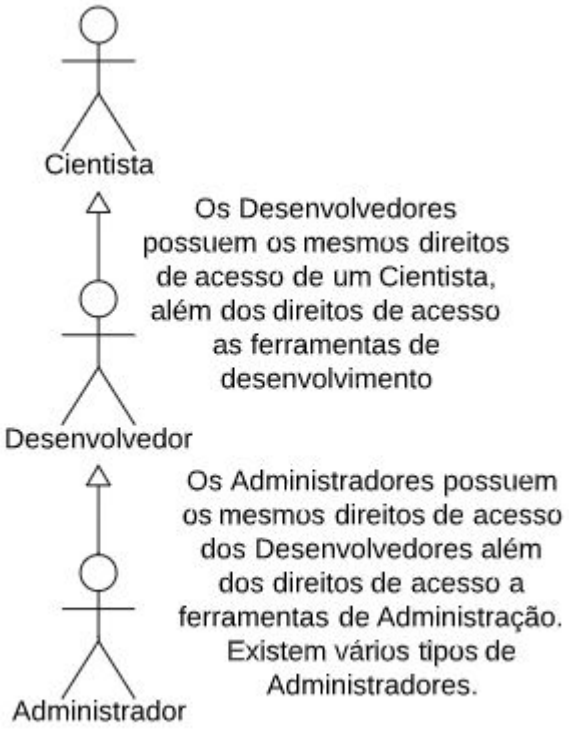
Novos Requisitos LSST

DES → LSST (40 a 100 x)

- Maior poder de processamento
- Melhora no armazenamento e recuperação de dados
- Reprocessamento parcial em caso de falhas
- Melhora na gestão de elementos do Portal

Tipos de Usuários

- Cientista
- Desenvolvedor
  - Cientistas desenvolvem novos pipelines e disponibilizam no Portal
- Administrador
  - Usuários especiais responsáveis por procedimentos especiais como criação de classes



Cientista

- Papel: Usar o Portal
  - Executar Pipelines
  - Consultar Processos em execução/executados
  - Resubmeter um Processo a partir de um **Checkpoint** ***(new)***
  - Consultar Processos já Executados e seus resultados
  - Consultar resultados - produtos - disponibilizados pela execução de pipelines
  - Exportar produtos gerados no Portal
  - Importar produtos para o Portal

Desenvolvedor

- Papel: Criar novos pipelines para o Portal
  - Conjunto de ferramentas facilitadoras
  - Ambiente amigável edição e testes
  - Design patterns que facilitem a implementação
    - Esqueletos de código com soluções para problemas comuns
  - Especificação configuração de pipelines
    - parâmetros de configuração, defaults, entradas e saídas
  - Ambiente e ferramentas para documentação de funções, componentes e pipelines;
  - Apoio de testes (testes de unidades, testes automático para pipelines, histórico de testes).

Administrador

- Papel: Controlar processos internos ao Portal
  - Exemplos de Administradores
    - Administrador de Classe
    - Administrador de Produtos
    - Administrador de Pipelines
  - O Portal deve prover Ferramentas de gestão
    - Exemplo: Para Classes
      - Cadastrar novas classes com toda sua descrição
      - Consultar e Editar Classes existentes
      - …..

Vantagens do SciencePortal

- Dados:
  - disponibilização de acesso a dados (**produtos**) já importados de diversos sites
- Desenvolvimento:
  - empregar modelos predefinidos que facilitem o processo em especial para resolver problemas de paralelização e acesso a dados
  - manter um arcabouço de software científico e compartilhar experiências
- Execução em um ambiente controlado que:
  - monitora a execução;
  - permite reexecução parcial;
  - mantém histórico de execuções anteriores
  - disponibiliza resultados (**produtos**) para consulta e validação

Produtos

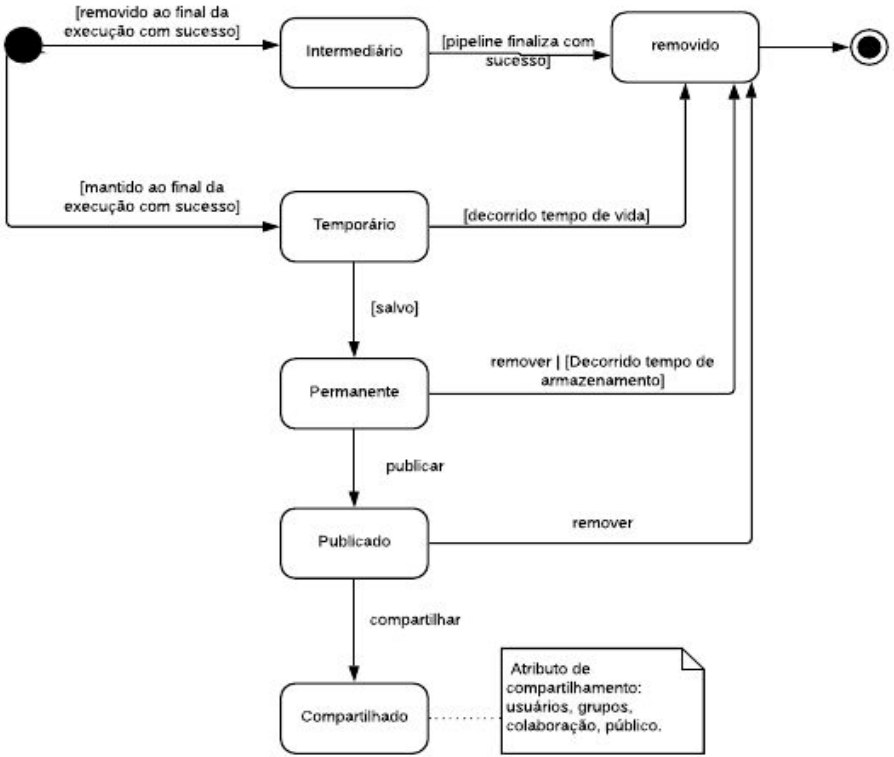
- **Produtos Externos** e **Produtos Primários**
  - Catálogos, Máscaras, Mapas, Imagens
- **ProdutosLinea**
  - Gerados por colaboradores no Linea internamente ao Linea ou em outros ambientes
- **ProdutosPortal**
  - ProdutosLinea gerados no portal - Resultados de execução de pipeline

- **Proveniência**: requisito fundamental a manutenção da proveniência de todos produtos disponibilizados no Linea.

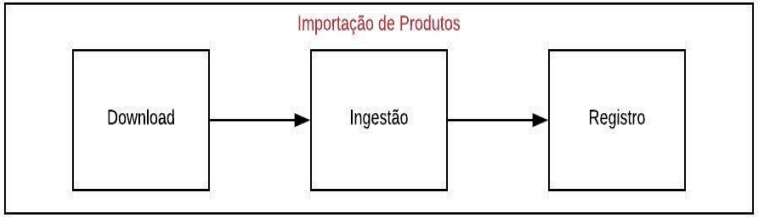
Todo Produto tem uma Classe

Produtos

Tempo de Vida ProdutosPortal



Produtos Externos / ProdutosLinea



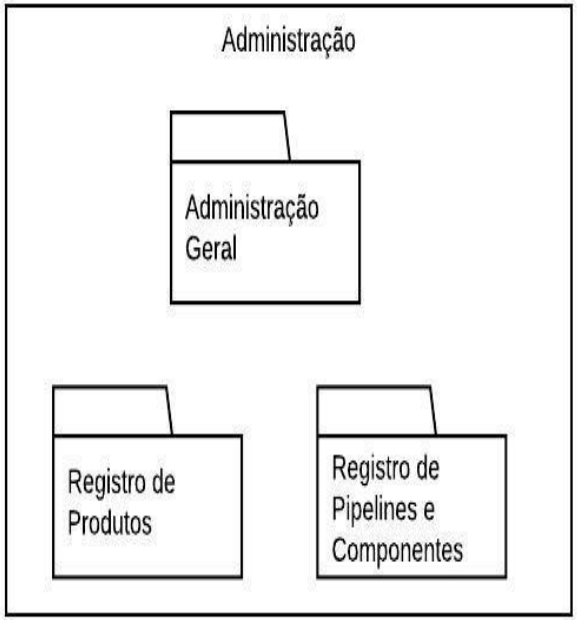
Classe

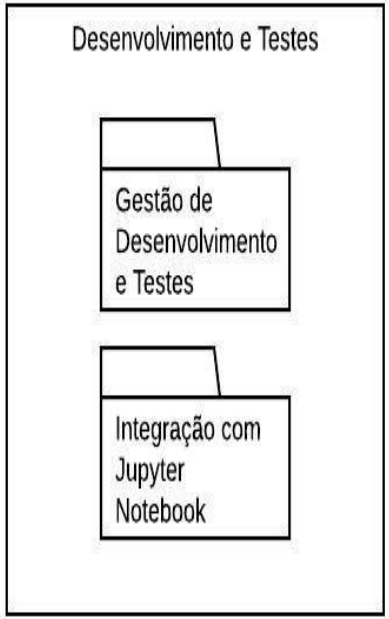
- Especificada e cadastrada previamente Responsabilidade: ***Administrador de Classes***.
  - Definição dos tipos de entradas de um pipeline
  - Definir as características dos produtos resultantes da execução do pipeline

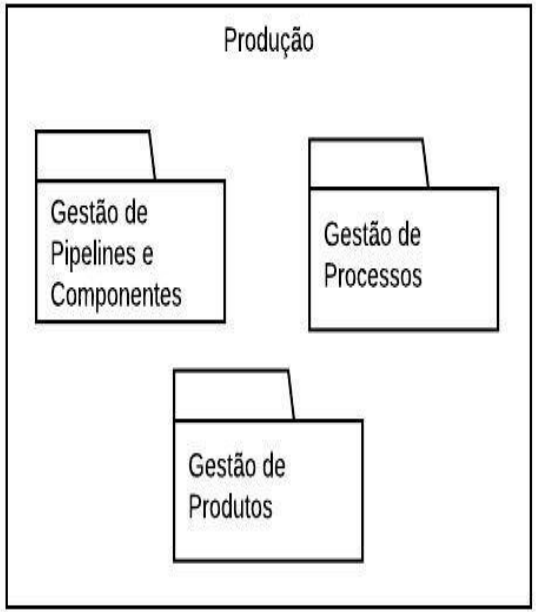
Uma classe é definida por:

- um nome;
- um tipo que especifica se o produto desta classe é uma tabela, uma imagem, etc;
- releases aos quais a classe se aplica;
- forma de armazenamento: banco de dados, flat file ou ambos;
- versão da definição da classe.

Componentes do Portal

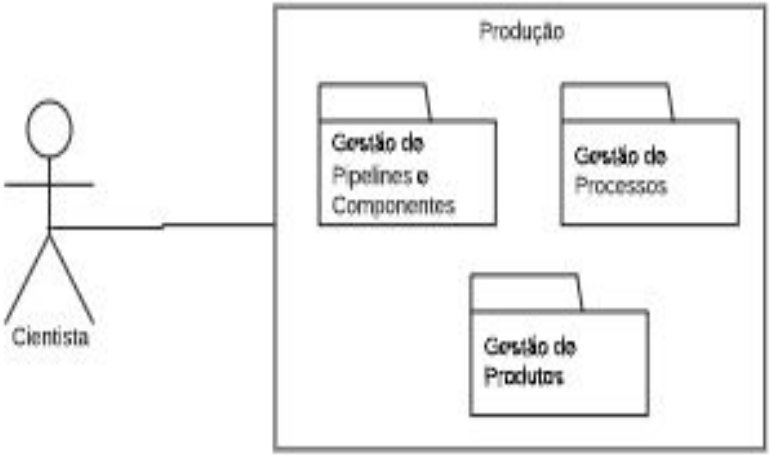




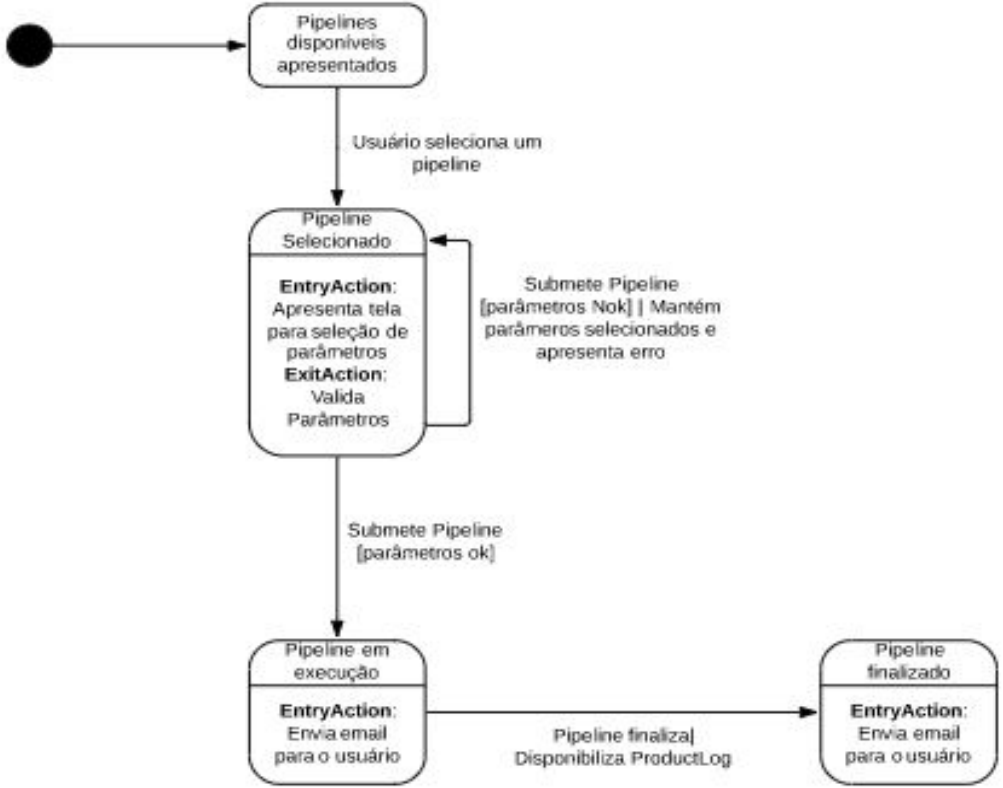


Componente

Produção



Produção - Gestão de Pipelines



Considera-se ***processo*** a execução de um pipeline.

- pipeline selecionado para execução;
- valores dos parâmetros de configuração empregados na execução;
- produtos selecionados como entrada;
- produtos resultantes da execução do pipeline;
- productLog - com resultados da execução e jupyter notebook;
- controlam checkpoints para reexecução parcial

Produção - Gestão de Processos

Funcionalidades:

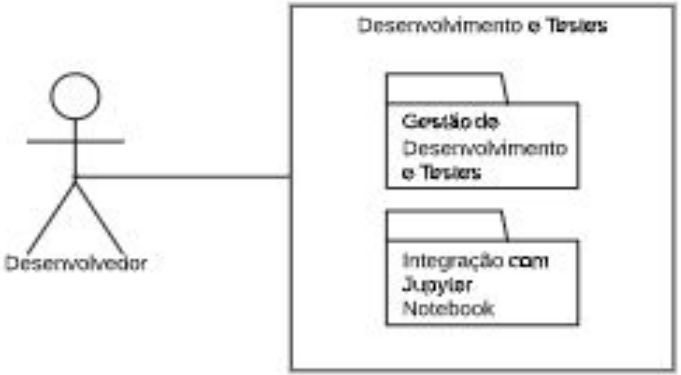
- Manter o registro de todos os processos executados e seus resultados.
  - pipeline e versão executado;
  - productLog (que contém parâmetros e configuração de execução)
  - resultados (produtosPortal criados)
  - data de execução do pipeline
  - usuário que executou os pipeline
  - remoção: só dos resultados - informações de execução seu productLog nunca poderão ser removidos.
- **Dashboard**, que resume todas as informações de processos visíveis ao usuário
  - toda a cadeia de processos que geraram produtosPortal empregados como parâmetros de entrada

Produção - Gestão de Produtos

Funcionalidades:

- Produtos existentes (a partir da seleção de critérios de filtragem);
- Possibilita inspecionar dados de um um Produto;
- Ingestão produtos externos de pequeno porte
    - O produto é registrado no portal.
    - Registrar todos os detalhes da transferência, tipo de armazenamento selecionado, etc.

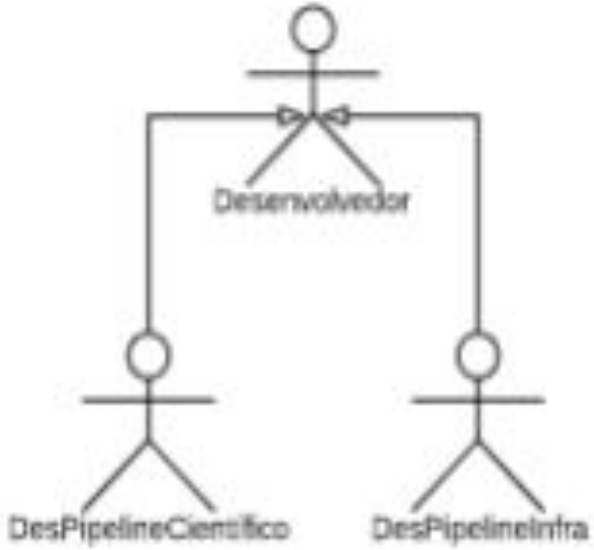
- Permitir a exportação de produtosPortal, diretamente para a máquina do usuário ou para sites associados como o NCSA por exemplo.



Componente

Desenvolvimento e Testes

Desenvolvimento e Testes



Objetivo: Reunir facilidades que visam melhorar a produtividade do processo de desenvolvimento de software científico.

DesPipelineCientífico:

- trata o problema com jupyter notebook
- testa seu algoritmo

DesPipelineInfra:

- Pipelines de uso geral - tratamento inicial dos dados
- Cientistas especialistas em desenvolvimento

Processo de Desenvolvimento

- Integração do processo de desenvolvimento com o Jupyter Notebook

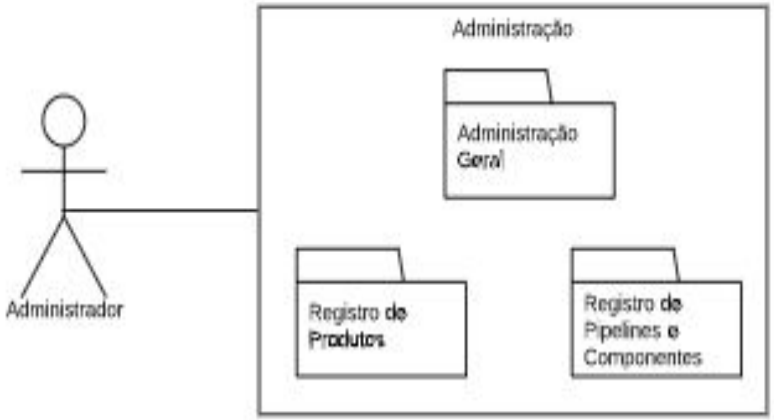
- Controle do processo de disponibilização para produção
  - Controle de qualidade
  - Controle de documentação
  - Avaliação de Impacto

- Manutenção de uma base de programas desenvolvidos pelos cientistas

Desenvolvimento e Testes

Jupyter Notebook ⇒ Ambiente Paralelo do Portal
Cientista ⇒ Equipe de Desenvolvimento do Portal

- ferramentas facilitadoras para codificação e teste do pipeline
  - Salvar testes e conjunto de testes
- Classe de Problema ⇒ Classe de Solução
  - padrões de projeto ("design patterns") para o modelo de programação de componentes;
  - padrões de projeto ("design patterns") para o modelo de programação de pipelines (incluindo modelos de paralelismo);
- Preparar pipeline para outros ambientes
  - Por exemplo - criar container para um dado pipeline, para uma dada versão;
- Processo de Desenvolvimento
  - Documentação de funções, componentes e pipelines;
  - Análise de eficiência e de qualidade do código de um pipeline;
  - Grafo de dependência de componentes, entradas e saídas



Componente

Administração

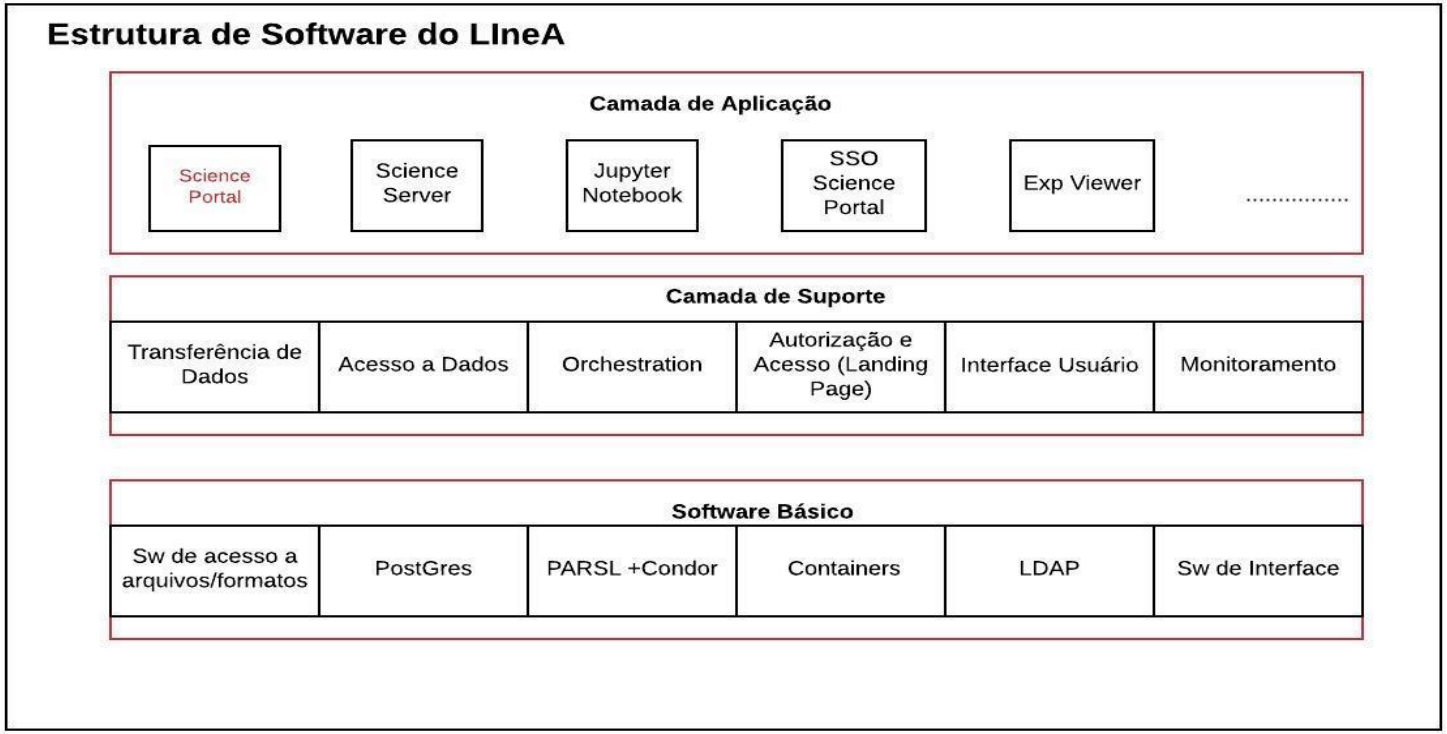
Administração

- **Registro de Produtos**
  - Administrador de Produtos
    - Produtos Externos de grande porte
    - Produtos "deprecated"
  - Manter o cadastro de todos os produtos
  - Manutenção do ambiente - **GarbageCollector**
- **Administração Geral**
  - Gestão de bibliotecas de desenvolvimento
  - Gestão de classes;
- **Registro de Pipelines e Componentes**
  - Pipeline aprovado ⇒ requisição de "ingestão"
    - Requisitar - Aprovar - Ingerir
    - Workflow para controle pelo CCB (infra estrutura)
    - Após o registro o pipeline é disponibilizado para produção.

Estrutura de Software do Linea

Opções de Implementação

Estrutura Proposta de Software do LINEA



Onde estamos….

- Refinando a especificação de Requisitos Funcionais
  - Base DES Science Portal
  - Introduzindo melhorias e lições aprendidas
- Requisitos Não Funcionais
  - Analisando alternativas de implementação e ferramentas
  - Alternativas para implementação das melhorias pretendidas
  - Proximidade as tecnologias empregadas no LSST
- Detalhamento da Camada de Suporte

E Depois….

- Implementar toda a infra estrutura proposta nesses documentos - TI
- Definir o conjunto de pipelines de infra estrutura para o LSST (cientistas)
- Implementar o conjunto de pipelines e colocá-los em produção (cientistas/TI)