

# Análise e Otimização de Lead Time em Processos Ágeis: Estudo de Caso em Empresa de Marketplace

**Victor Hugo Oliveira de Amorim**

Pablo Freire Matos - Orientador



# Resumo da Apresentação



Introdução



Objetivos



Metodologia



Trabalhos Correlatos



Proposta do Trabalho



Resultados



Considerações Finais



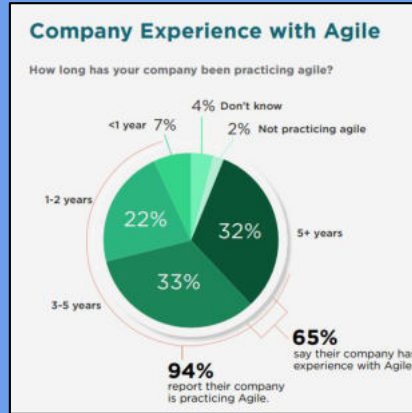
# Metodologia Ágil

- Eficiência em performance
- Feedback constante do cliente
- Comunicação frequente entre times
- Cumprimento de prazos





## Metodologia Ágil



Fonte: Digital.ai (2022)

- 94% das empresas de software possuem alguma experiência com desenvolvimento ágil. (Digital.ai, 2022)
- O desenvolvimento ágil apresenta um aumento de 232% entre 2021 e 2022. (Digital.ai, 2022)





## Lead Time / Duração de atividades:

The screenshot shows a Jira issue page for 'Add epic / MAR-6789'. The 'Activity' tab is selected and highlighted with a green box. Below the tab, there are filters for 'Show: All', 'Comments', 'History', 'Time Between Statuses', 'Time in Status', and 'Worklogs'. The 'History' filter is also highlighted with a green box. The activity list shows four updates by Yulia Borivets: 'updated the Rank', 'changed the Status', 'updated the Resolution', and 'updated the Description'. The 'Status' update shows a transition from 'IN PROGRESS' to 'DONE'. A 'Details' sidebar on the right lists fields like Activity, Goal, Channel, Cost center, Sprint, Assignee, Priority, and Story point estimate.

Fonte: Próprio autor

- Prazo de entrega
- Updates de atividades
- Histórico de changelog





### Principais Obstáculos

- Tecnologias e Frameworks
- Custo operacional
- Extrações dos dados
- Tratamento e armazenamento
- Disponibilização





## Objetivos

1

Extrair os dados diretamente da fonte

2

Tratar e armazenar os dados em um banco relacional

3

Garantir a alimentação automática das tabelas



4

Exibir análises dos dados



## Metodologia do Trabalho



Fonte: Próprio autor

- Extração e armazenamento em cloud AWS S3
- Tratamentos dos dados com PySpark
- Criação do DW para análises
- Criação de gráficos no Tableau





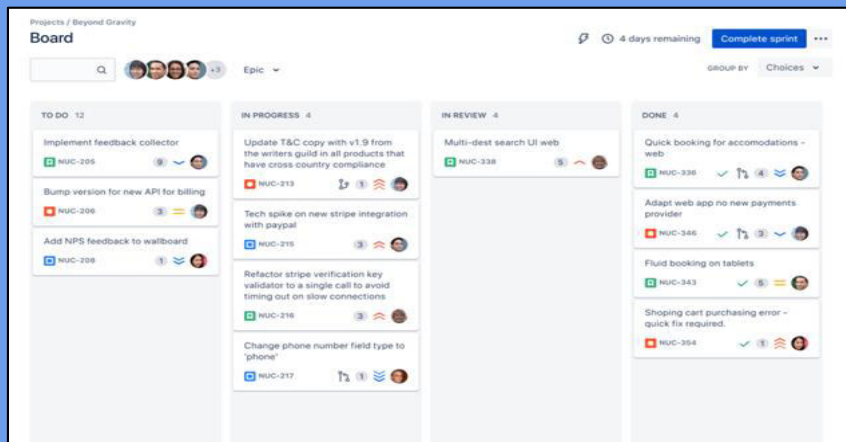


## Trabalhos Correlatos

Autor(es)	Instituição	Foco	Linguagem	Armazenagem	Banco de dados	BI
Tratamento Proposto (2023)	IFBA	Processos de coleta, transferência e consolidação de dados em ferramentas de metodologias ágeis, construindo métricas com ênfase em <b>Lead Time</b> .	Python	AWS	PostgreSQL	Tableau
Alves (2023)	UFSC	Desenvolver uma arquitetura de DW que <b>automatize o processo de coleta, transferência e consolidação</b> de dados		Servidor Próprio		Power BI
Maximiano (2023)	UNESP	Desenvolver um pipeline de dados responsável por realizar a coleta e o tratamento do conjunto de informações, possibilitando por meio de dashboards, a <b>análise de dados da área da saúde</b> para o estado de São Paulo		AWS		Redash
Haefner (2022)	UFRS	Uma <b>arquitetura de processamento</b> de dados na nuvem que seja facilmente adaptável a futuras implementações de dados de <b>estações meteorológicas</b>		Servidor Próprio	MariaDB	N/A
Leal (2021)	IFSP	Desenvolver e avaliar diferentes ferramentas open source para a criação de <b>pipeline de Dados para big data</b> , via processo de ELT, com enfoque em <b>ciências humanas</b> .		AWS	MongoDB	Tableau
Amaral (2020)	USP	Desenvolvimento de um algoritmo para <b>transformação de dados industriais</b> com linguagem de programação Python integrado no ambiente de computação em nuvem da Google				



## Fonte de dados



Fonte: Jira Software (2023)

- Jira Software permite o gerenciamento de projetos
- Organiza e armazena histórico de tarefas
- Exibe tarefas pendentes e define tipos de atividades



## Conexão a API

```
1 # This code sample uses the 'requests' library:
2 # http://docs.python-requests.org
3 import requests
4 from requests.auth import HTTPBasicAuth
5 import json
6
7 url = "https://your-domain.atlassian.net/rest/api/3/issue/picker"
8
9 auth = HTTPBasicAuth("email@example.com", "<api_token>")
10
11 headers = {
12     "Accept": "application/json"
13 }
14
15 query = {
16     'query': 'query'
17 }
18
19 response = requests.request(
20     "GET",
21     url,
22     headers=headers,
23     params=query,
24     auth=auth
25 )
26
27 print(json.dumps(json.loads(response.text), sort_keys=True, indent=4, separators=(",", ": ")))
```

Fonte: Jira Software (2023)

- Requests
- Token de segurança
- Extração



## Orquestração e Agendamento

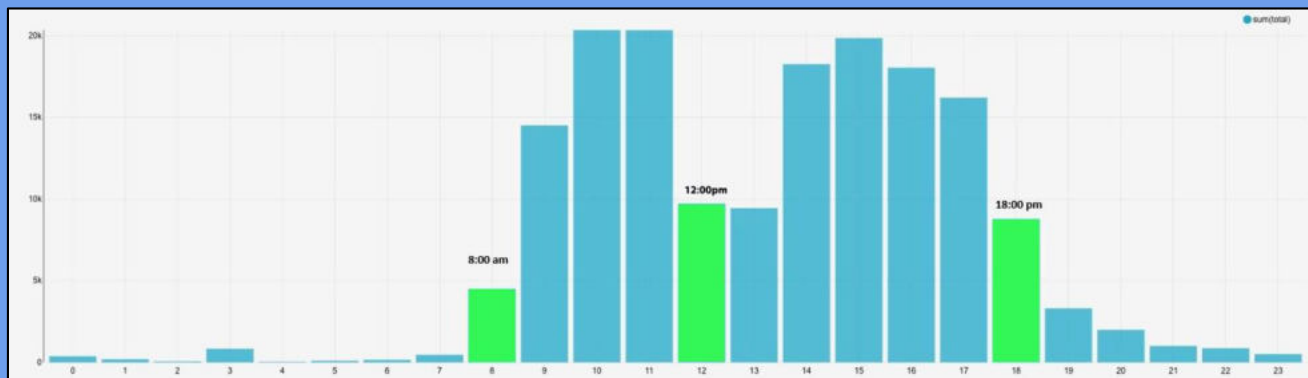
DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
jira_capitalization	datasciences	17	0 18 ***	2023-09-26, 15:00:00	2023-09-26, 18:00:00	...	...	...
jira_employees_extract	datasciences	13	0 8 ***	2023-09-25, 00:00:00	2023-09-26, 00:00:00	...	...	...
jira_extractor	datasciences	17	0 8 ***	2023-09-25, 00:00:00	2023-09-26, 00:00:00	...	...	...
jira_extractor_raw	datasciences	1	0 22 /14 * *	2023-09-16, 22:00:00	2023-09-23, 22:00:00	...	...	...

Fonte: Próprio autor

- Orquestração de scripts
- Agendamento personalizado das DAGS (Directed Acyclic Graph)



## Orquestração e Agendamento



Fonte: Próprio autor

- Extração em modelo batch
- Análise predefinida para menor custo de operação
- Maior atividade na plataforma em horários comerciais



## Tratamento e Armazenamento

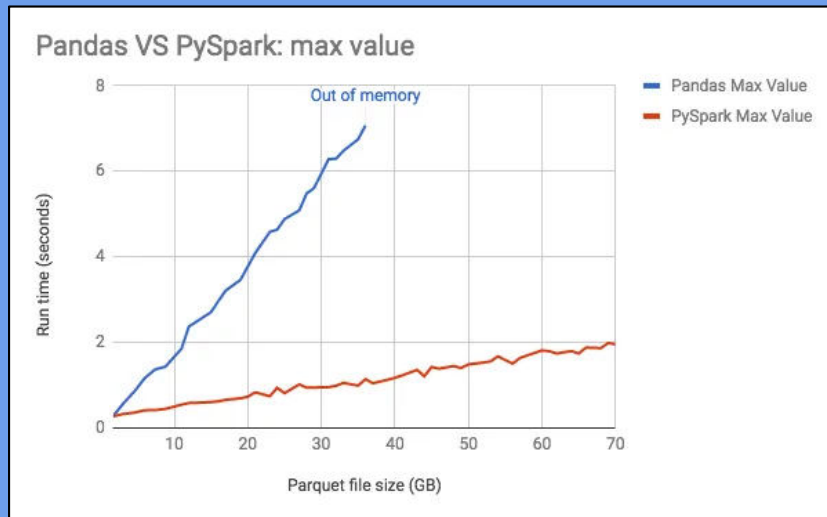
```
jira_transform.py x requirements.txt
jira_transform.py > jiratssueslakehouse > transform_dataframe
1 import argparse
2 from pyspark.sql.functions import *
3 from pyspark.sql.types import *
4 from pyspark.sql import SparkSession
5 from datetime import datetime, timedelta
6
7 class JiraIssuesLakehouse:
8     def __init__(self, spark_session, execution_date= datetime.today().date()):
9         self.execution_date = execution_date
10        self.spark_session = spark_session
11
12    def transform_dataframe(self):
13
14        issues_metrics_update = self.spark_session.sql(F'''
15            select
16                key                                as issue_key,
17                get_json_object(fields__project, '$.vertical')    as vertical,
18                get_json_object(fields__project, '$.league')      as league,
19                get_json_object(fields__issuetype, '$.issue_type') as issue_type,
20                get_json_object(fields__project, '$.priority')    as priority,
21                get_json_object(fields__issuetype, '$.issue_status') as issue_status,
22                get_json_object(fields__issuetype, '$.status_category') as status_category,
23                from_utc_timestamp(to_timestamp(fields__created), 'America/Sao_Paulo') as created_date,
24                from_utc_timestamp(to_timestamp(progress_date), 'America/Sao_Paulo') as statuscategory_progress_date,
25                from_utc_timestamp(to_timestamp(done_date), 'America/Sao_Paulo') as statuscategory_done_date,
26                to_date('self.execution_date', "yyyy-MM-dd") as execution_date
27            from cross_jira.issues_raw
28            where dt=date('self.execution_date')
29        ''')
30        return issues_metrics_update
```

Fonte: Próprio autor

- As transformações e tipagens dos dados serão realizadas com PySpark
- O script tem a data de execução como parâmetro para um maior controle de versionamento



## Tratamento e Armazenamento

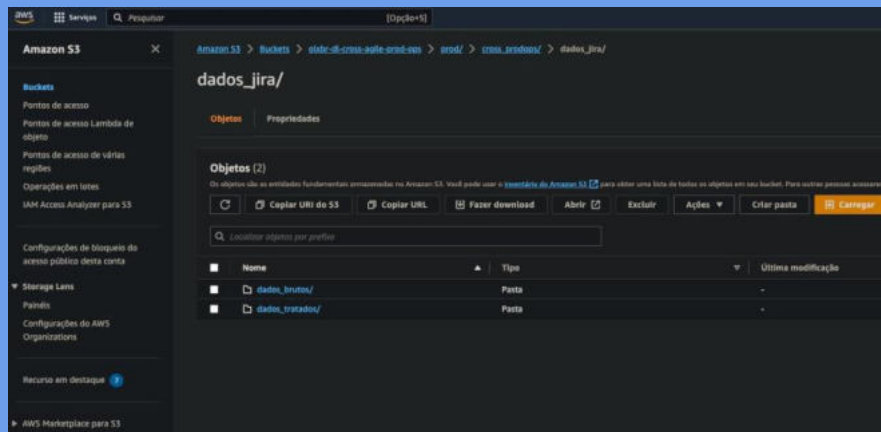


Fonte: Juhari (2021)

- O PySpark se adequa muito bem no cenário de big data
- Comparado ao Pandas, vemos a importância da clusterização



## Tratamento e Armazenamento



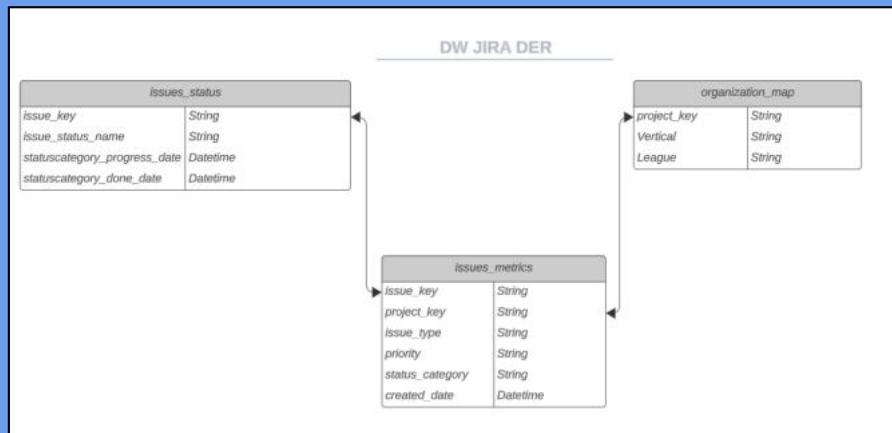
Fonte: Próprio autor

- As extrações orquestradas pela apache airflow terão como destino um diretório no S3
- A coleta é dividida em duas partes, dados brutos e tratados
- O versionamento permite maior consistência de histórico





## Data Warehouse



Fonte: Próprio autor

- Uso do SGBD PostgreSQL
- Tabela Status
- Tabela Organization Map
- Tabela Metrics



## Tableau



Fonte: Próprio autor

- Os gráficos foram disponibilizados no Tableau, democratizando o acesso às informações
- É possível verificar a taxa de acesso ao dashboard, confirmando sua importância para a rotina das equipes
- O Tableau permite um fluxo otimizado e leituras intuitivas das informações



## Resultados



Fonte: Próprio autor

- Liberdade para análises internas
- Aumento de performance
- Melhora no tempo médio
- Extinção de trabalhos duplicados

Acesso à amostra de dados:

[https://docs.google.com/spreadsheets/d/1Woic0LSnlaf\\_Hn99SHeLJFqZadl\\_i8K6-\\_pPvc-CBE/edit#gid=0](https://docs.google.com/spreadsheets/d/1Woic0LSnlaf_Hn99SHeLJFqZadl_i8K6-_pPvc-CBE/edit#gid=0)

Acesso ao Dashboard:

[https://public.tableau.com/app/profile/victor.amorim/viz/Amostra\\_artigo\\_16997967633140/MtricasdeLeadTime](https://public.tableau.com/app/profile/victor.amorim/viz/Amostra_artigo_16997967633140/MtricasdeLeadTime)





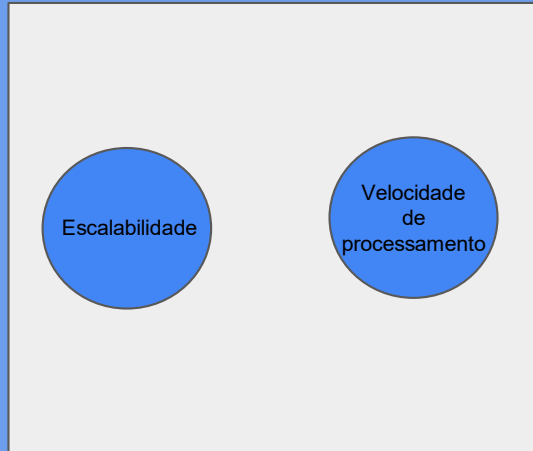
## FLUXO DO PROJETO





## Comparação com Trabalhos Correlatos

### Benefícios



### Dificuldades





### Considerações Finais

- A implementação deste projeto possibilitou uma ampla gama de análises gerenciais.
- Ferramentas *open source* foram eficazes e importantes em termos orçamentais.
- Além disso, o controle, o armazenamento e a propriedade dos dados tornam-se vantagens adicionais para a empresa.

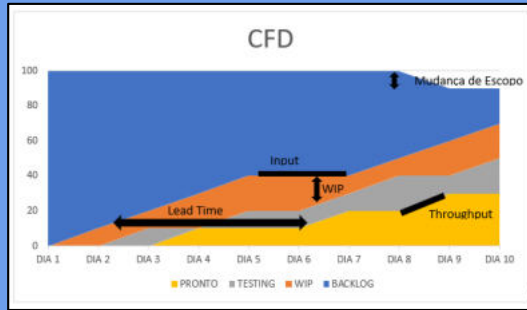




## Considerações Finais

### Trabalhos Futuros

- Cálculo de Diagrama de fluxo cumulativo



- Cálculo de custo por atividade



# Referência Citadas na Apresentação

- ALVES, Y. O. (2023). "Design e implementação de uma arquitetura de data Warehouse.". Universidade Federal de Santa Maria, RS, <https://repositorio.ufsm.br/handle/1/28249>
- AMARAL, Daniel Rodrigo Ximenes (2020) " Desenvolvimento de algoritmos com Apache Spark para tratamento de dados industriais para Business Analytics com Tableau". Universidade de São Paulo, 2020. <https://sistemas.eel.usp.br/bibliotecas/monografias/2020/MEF20006.pdf>
- DIGITAL.AI, (2022), "15th State of Agile Report. Agile adoption accelerates across the enterprise". [https://info.digital.ai/rs/981-LQX-968/images/SOA15.pdf?\\_ga=2.2071153.1396698264.1663000131-2013811604.1663000131](https://info.digital.ai/rs/981-LQX-968/images/SOA15.pdf?_ga=2.2071153.1396698264.1663000131-2013811604.1663000131)
- JUHARI T., (2021), "Benchmarking Pandas VS Spark". <https://hirazone.medium.com/benchmarking-pandas-vs-spark-7f7166984de2>
- LEAL, D. A. (2023). "Pentaho, airflow e python: avaliação de ferramentas para a criação de pipeline de dados". Instituto Federal de Educação Ciência e Tecnologia de São Paulo, <http://pergamum.ifsp.edu.br/pergamumweb/vinculos/000075/0000754b.pdf>.
- L3 Software. (2023). "Jira: o software que se tornou o braço direito dos desenvolvedores". <https://l3software.com.br/blog/software/jira-o-software-que-se-tornou-o-braco-direito-dos-desenvolvedores/>.
- MAXIMIANO, C. F. C. (2023). "Uso de tecnologias de big data para processamento e análise de dados da área da saúde do estado de São Paulo".Universidade Estadual Paulista, Instituto de Ciência e Tecnologia, SP, <https://repositorio.unesp.br/handle/11449/243963>.
- SANTOS, C. (2023). "Metodologia Ágil Empresas: Adoção da metodologia ágil por empresas" Awari.[https://awari.com.br/metodologia-agil-empresas-adocao-da-metodologia-agil-por-empresas/?utm\\_source=blog&utm\\_campaign=projeto+blog&utm\\_medium=Metodologia%20%C3%81gil%20Empresas:%20Ado%C3%A7%C3%A3o%20da%20metodologia%20%C3%A1gil%20por%20empresas](https://awari.com.br/metodologia-agil-empresas-adocao-da-metodologia-agil-por-empresas/?utm_source=blog&utm_campaign=projeto+blog&utm_medium=Metodologia%20%C3%81gil%20Empresas:%20Ado%C3%A7%C3%A3o%20da%20metodologia%20%C3%A1gil%20por%20empresas)



# Análise e Otimização de Lead Time em Processos Ágeis: Estudo de Caso em Empresa de Marketplace

Obrigado!!!

**Victor Hugo Oliveira de Amorim**

Pablo Freire Matos - Orientador

