PROJETO FINAL
TECNOLOGIAS DE BIG DATA E INTELIGÊNCIA ARTIFICIAL

GRUPO 8 | INDIGESTION

Nome dos Alunos:
Alan Batista
Manuella Paez
Mario José C M Prado
Wislom Diogo Almeida

Coordenador:
Prof. Fabio Jardim

LABDATA | FUNDAÇÃO INSTITUTO DE ADMINISTRAÇÃO BUSINESS SCHOOL

Agenda

1. Contextualização do trabalho
2. Visão e objetivo do projeto
3. Documentação da solução
    i. Diagrama da arquitetura e descrição dos serviços
    ii. Detalhamento e configurações técnicas
4. Demonstração da solução e entregáveis

1. Contextualização do trabalho

O sistema de transporte público de São Paulo atende **milhões de pessoas** diariamente, e a **eficácia na gestão da frota de ônibus** é crucial para **assegurar uma boa qualidade do serviço público** prestado e, consequentemente, a satisfação da população com o tema.

O trabalho possui como objetivo a construção de uma **aplicação** que possibilite o **monitoramento em tempo quase real dos ônibus** em circulação no estado de São Paulo e que **ofereça métricas e KPIs importantes para tomada de decisão**.

2. Visão e objetivo do Projeto

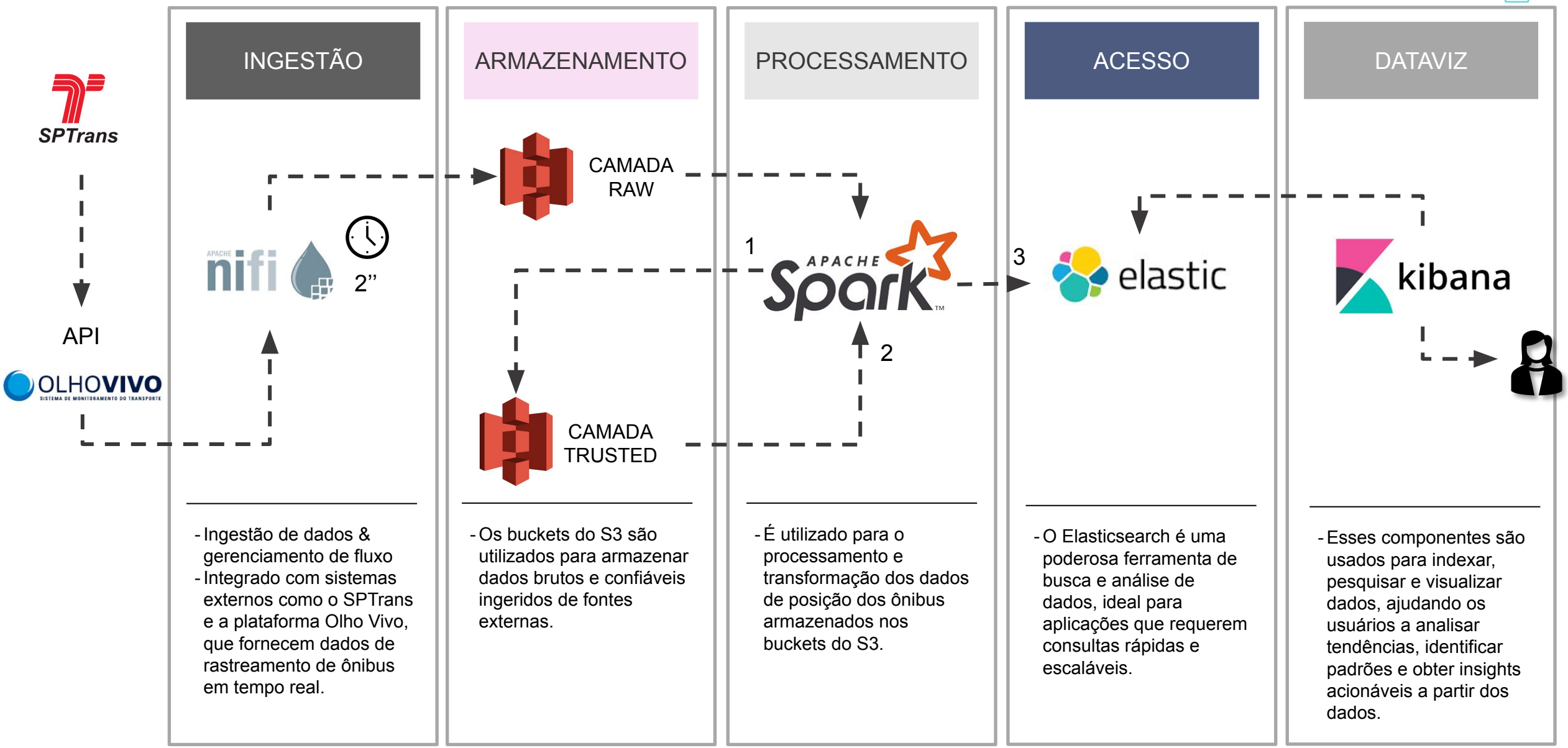
Solução de monitoramento em tempo real dos ônibus em circulação no cidade de São Paulo

Uso dos dados da API OLHO VIVO da SPTrans e dados GTFS

Integração com um data lake para armazenamento e enriquecimento de dados

Business intelligence para monitoramento em tempo quase real e análises para otimização do transporte público de São Paulo, Brasil.

3.i Solução | Arquitetura da solução e descrição dos serviços



- Ingestão de dados & gerenciamento de fluxo
- Integrado com sistemas externos como o SPTrans e a plataforma Olho Vivo, que fornecem dados de rastreamento de ônibus em tempo real.

- Os buckets do S3 são utilizados para armazenar dados brutos e confiáveis ingeridos de fontes externas.

- É utilizado para o processamento e transformação dos dados de posição dos ônibus armazenados nos buckets do S3.

- O Elasticsearch é uma poderosa ferramenta de busca e análise de dados, ideal para aplicações que requerem consultas rápidas e escaláveis.

- Esses componentes são usados para indexar, pesquisar e visualizar dados, ajudando os usuários a analisar tendências, identificar padrões e obter insights acionáveis a partir dos dados.

3.ii Solução | Detalhamento e configurações técnicas

INGESTÃO E ARMAZENAMENTO | PROCESSAMENTO | ACESSO E DATAVIZ

Credenciais de acesso:

NiFi:
- **URL**: http://localhost:49090
- **Porta padrão para acesso à interface web do NiFi :** 9090

MinIO:
Console: http://localhost:49001
Serviço de Armazenamento: http://localhost:49000
Portas expostas:
- 9000: Porta padrão para acesso ao serviço MinIO
- 9001: Porta do console de gerenciamento

Credenciais de Acesso:
Usuário: admin / **Senha**: minioadmin

GITHUB

1 nifi

Detalhamentos adicionais

Configuração:
Imagem Utilizada: apache/nifi:${NIFI_VERSION}
Container Name: nifi-otmzsp
Hostname: nifi-otmzsp

Ambiente:
NIFI_WEB_HTTP_PORT: 9090
NIFI_WEB_HTTPS_HOST: nifi
Timezone (TZ): America/Sao_Paulo

Volumes:
- *./volumes/nifi/util:/útil* | Diretório para utilitários.
- *./volumes/nifi/util/jar:/util/jar* | Diretório para arquivos JAR.
- *./volumes/nifi/conf:/opt/nifi/nifi-current/conf* | Diretório de configuração do NiFi.

Comando:
sh -c "ln -snf /usr/share/zoneinfo/$$TZ /etc/localtime && echo $$TZ > /etc/timezone" | Configuração do timezone.

Recursos:
Limite de Memória: 2 GB para garantir desempenho adequado.

2

Configuração:
Imagem Utilizada: minio/minio:${MINIO_VERSION}
Container Name: minio-otmzsp

Estrutura de Pastas:
- Raw: Diretório para armazenamento de dados brutos.
- Trusted: Diretório para armazenamento de dados confiáveis.

Volumes montados para persistência de dados: /minio_data/raw e /minio_data/trusted .

Health Check:
- **Comando**: ["CMD", "mc", "ready", "local"]
- **Intervalo**: 5 segundos
- **Retries**: 5 tentativas
- **Timeout**: 5 segundos

3.ii Solução | Detalhamento e configurações técnicas

INGESTÃO E ARMAZENAMENTO | **PROCESSAMENTO** | ACESSO E DATAVIZ

Credenciais de acesso:

Apache SPARK:

- **URL**: http://localhost:8080 (UI do Spark Master)

Portas expostas - Master:

- 7077: Porta do master para gerenciar os workers.
- **URL**: http://localhost:8081 (UI do Spark Worker)
- **Portas expostas - Worker:** 8081: Interface web do worker.

JUPYTER:

- **URL**: http://localhost:8888

Portas expostas:

- 8888: Porta padrão para acessar a interface web do Jupyter Notebook.
- 4040 a 4043: Portas usadas para monitoramento do Spark.

GITHUB

Detalhamentos adicionais

3 Apache Spark

Configuração:
Imagem Utilizada: apache/spark:${SPARK_VERSION}
Container Name:

- Master: *spark-master-otmzsp*
- Worker: *spark-worker-otmzsp*

Modos de Operação:

2. **Master**:

Variáveis de Ambiente:
SPARK_MODE: master
SPARK_MASTER_HOST: spark-master-otmzsp
TZ: America/Sao_Paulo
Limite de Memória: 2 GB.
Comando: /opt/spark/bin/spark-class org.apache.spark.deploy.master.Master

3. **Worker**:

Variáveis de Ambiente:
SPARK_MODE: worker
SPARK_MASTER_URL: spark://spark-master-otmzsp:7077
SPARK_WORKER_MEMORY: 1g
TZ: America/Sao_Paulo
Limite de Memória: 1 GB.
Comando: /opt/spark/bin/spark-class org.apache.spark.deploy.worker.Worker spark://spark-master-otmzsp:7077

4 jupyter

Configuração:
Imagem Utilizada: *jupyter/pyspark-notebook:latest*
Container Name: *jupyter-otmzsp*
Ambiente:
JUPYTER_TOKEN: "" | Desabilita o token de segurança para acesso.

Volumes:
../notebooks:/home/jovyan/work | Diretório para armazenar notebooks, permitindo persistência de dados e fácil acesso.

Comando:
start-notebook.sh --NotebookApp.token='' --NotebookApp.password='' | Inicia o servidor Jupyter sem token ou senha, facilitando o acesso.

3.ii Solução | Detalhamento e configurações técnicas

INGESTÃO E ARMAZENAMENTO | PROCESSAMENTO | ACESSO E DATAVIZ

Credenciais de acesso:

Elastic Search:
- URL: http://localhost:49200 (API)

Kibana:
- **URL: http://localhost:45601**

GITHUB

5 elastic

Detalhamentos adicionais

Configuração:
Imagem Utilizada: elasticsearch:7.17.20
Container Name: elasticsearch-otmzsp
Hostname: elasticsearch-otmzsp

Ambiente:
discovery.type: single-node | configuração para executar em modo single node
.ES_JAVA_OPTS: "-Xms2g -Xmx2g" | configurações de memória do Java
.xpack.security.enabled: "false" | desabilita a segurança para simplificar a configuração

Volumes:
./volumes/elasticsearch/esdata:/usr/share/elasticsearch/data | Mapeamento para persistência de dados.

Health Check:
Comando: curl -sS --fail http://elasticsearch-otmzsp:9200/_cluster/health?wait_for_status=yellow&timeout=0s
Intervalo: 1 segundo
Retries: 3 tentativas
Start Period: 20 segundos
Timeout: 5 segundos

6 kibana

Configuração:
Imagem Utilizada: kibana:7.17.20
Container Name: kibana-otmzsp
Hostname: kibana-otmzsp

Ambiente:
ELASTICSEARCH_HOSTS: "http://elasticsearch-otmzsp:9200" | configuração para conectar ao Elasticsearch.

Volumes:
./volumes/kibana/data:/usr/share/kibana/data | Mapeamento para persistência de dados, garantindo que as configurações e dashboards sejam mantidos.

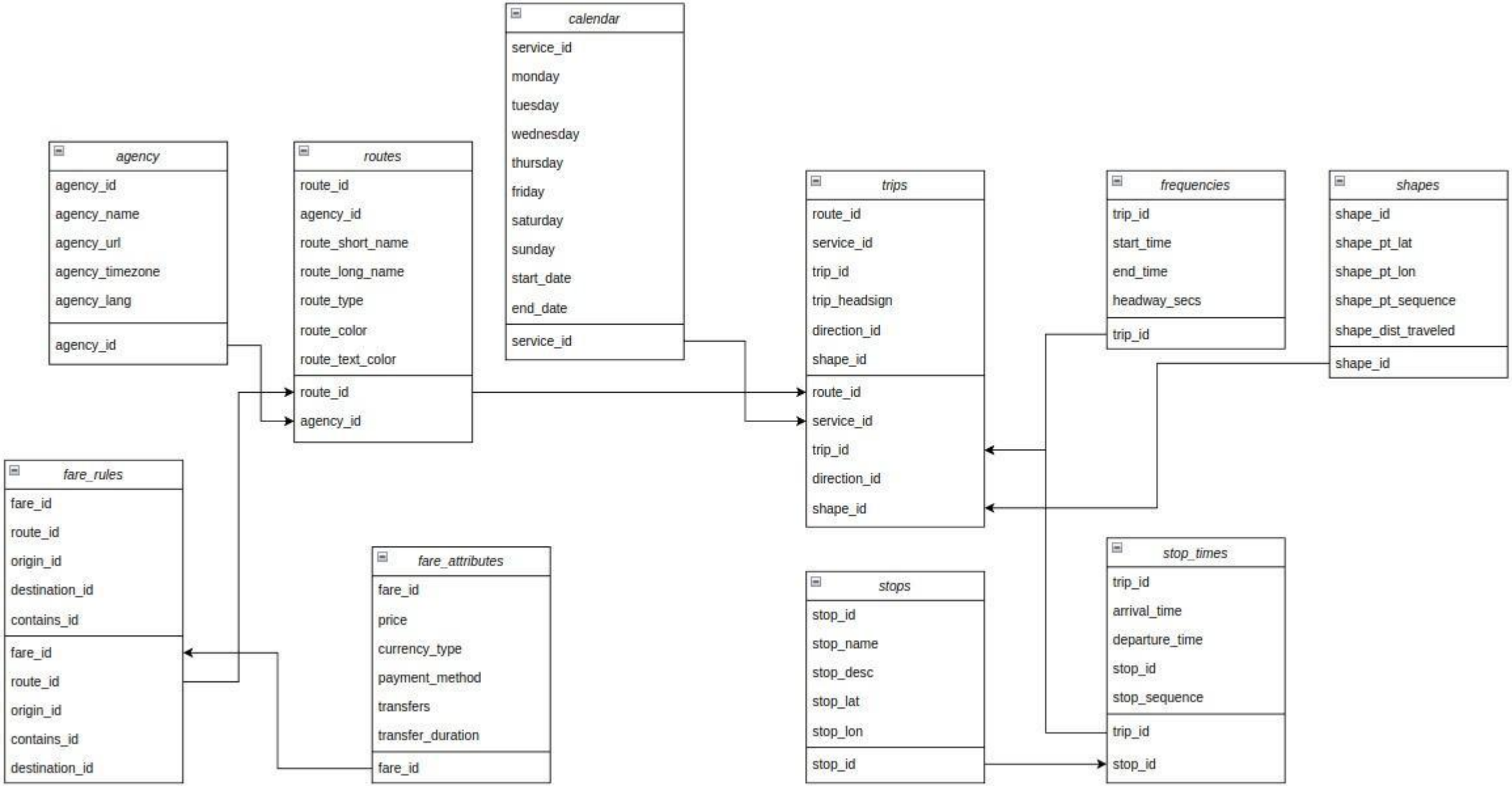
Dependências:
Elasticsearch: O Kibana depende do Elasticsearch, sendo necessário garantir que o serviço do Elasticsearch esteja saudável antes do início do Kibana.

PROJETO FINAL
DATA ENGINEERING

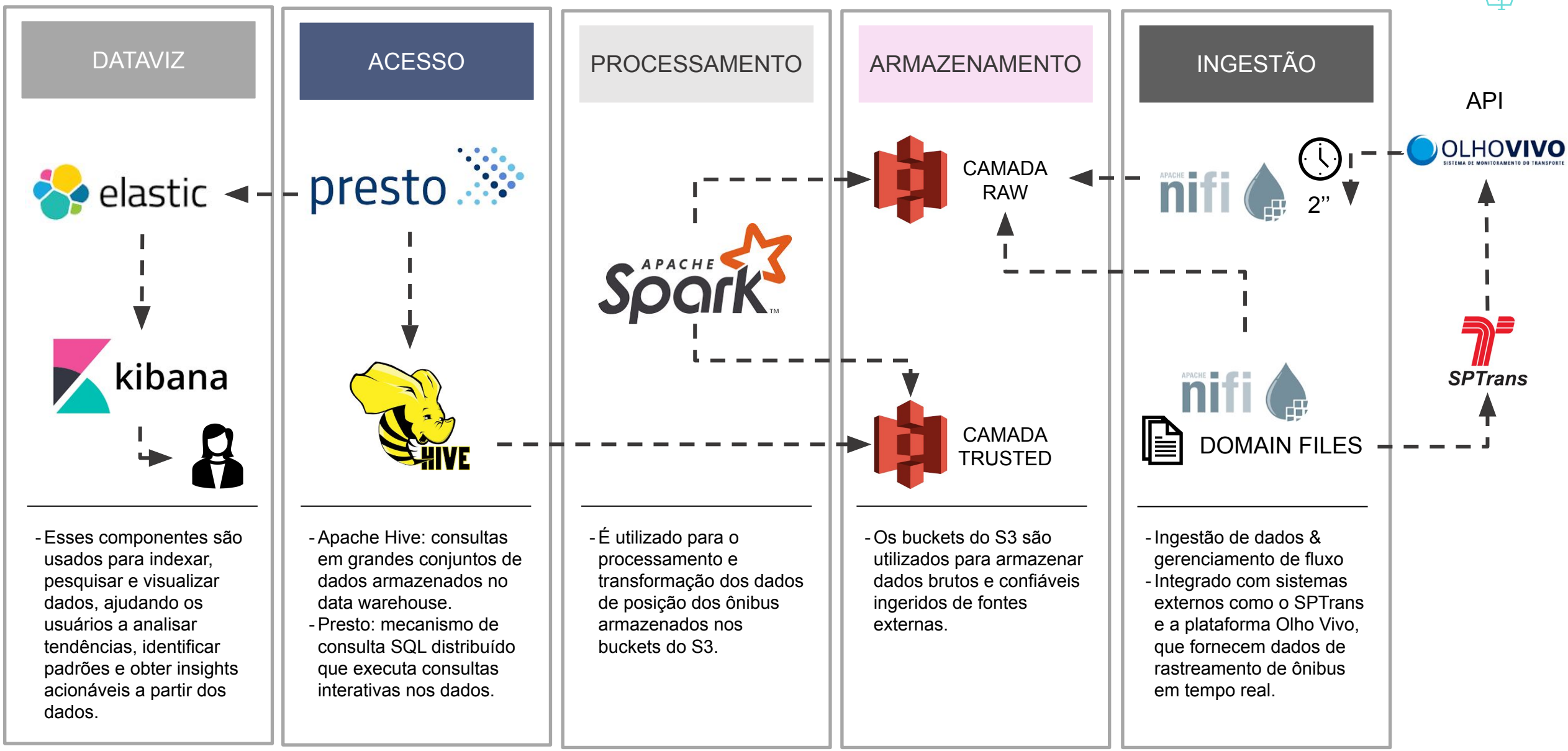
GRUPO 8 | INDIGESTION

Nome dos Alunos:
Alan Batista
Manuella Paez
Mario José C M Prado
Wislom Diogo Almeida

Coordenador:
Prof. Fabio Jardim


calendar
service_id
monday
tuesday
wednesday
thursday
friday
saturday
sunday
start_date
end_date
service_id
agency
agency_id
agency_name
agency_url
agency_timezone
agency_lang
agency_id
routes
route_id
agency_id
route_short_name
route_long_name
route_type
route_color
route_text_color
route_id
agency_id
trips
route_id
service_id
trip_id
trip_headsign
direction_id
shape_id
route_id
service_id
trip_id
direction_id
shape_id
frequencies
trip_id
start_time
end_time
headway_secs
trip_id
shapes
shape_id
shape_pt_lat
shape_pt_lon
shape_pt_sequence
shape_dist_traveled
shape_id
fare_rules
fare_id
route_id
origin_id
destination_id
contains_id
fare_id
route_id
origin_id
contains_id
destination_id
fare_attributes
fare_id
price
currency_type
payment_method
transfers
transfer_duration
fare_id
stops
stop_id
stop_name
stop_desc
stop_lat
stop_lon
stop_id
stop_times
trip_id
arrival_time
departure_time
stop_id
stop_sequence
trip_id
stop_id

3.i Solução | Arquitetura da solução



3.ii Solução | Documentação da aplicação

INGESTÃO E ARMAZENAMENTO | PROCESSAMENTO E ACESSO | DATAVIZ

Credenciais de acesso:

NiFi:
- **URL**: http://localhost:49090
- **Porta**: 9090 (mapeada para a porta 49090 no host)

MinIO:
Console: http://localhost:49001
Serviço de Armazenamento: http://localhost:49000
Portas:
- 9000 (serviço de armazenamento) mapeada para a porta 49000 no host
- 9001 (console) mapeada para a porta 49001 no host

Credenciais de Acesso:
Usuário: admin / **Senha**: minioadmin

README/GITHUB

	1 nifi	2 SPTrans OLHOVIVO	3 nifi
Macro-etapas	Definir serviços principais (NiFi e MinIO) e criar rede personalizada otmzsp-network	Gerar certificado da API Olho Vivo	Configurar o StandardSSLContextService no NiFi
Detalhamento	1. Arquivo docker-compose.yml 2. Comando iniciar docker-compose	1. Obter certificado 2. Importar o certificado para um keystore Java	1. Acessar a interface web do NiFi 2. Adicionar e configurar o StandardSSLContextService 3. Aplicar a config. Reiniciar o serviço se necessário
Pré-configurações	Volume montado para persistência de dados e logs (ex.: repositórios de banco de dados, arquivos de fluxo, conteúdo, etc.). Definido fuso horário de SP. Versões corretas do NiFi e MinIO configuradas nas variáveis de ambiente ${NIFI_VERSION} e ${MINIO_VERSION} no arquivo .env. Diretório de dados: /mini_data/raw (brutos) /mini_data/trusted (confiáveis)	Configuração de diretório util/jks exista antes de executar o comando. ----- Documentação detalhada: Aqui Dicionário de dados: Aqui	NiFi configurado para utilizar o StandardSSLContextService em seus processadores que se comunicam via HTTPS.

3.ii Solução | Documentação da aplicação

INGESTÃO E ARMAZENAMENTO | **PROCESSAMENTO E ACESSO** | DATAVIZ

Credenciais de acesso:

Apache SPARK:
- **URL**: http://localhost:49090

Portas Spark Master:
- ~~7077~~
- ~~8080~~

Portas Spark Worker: 8081

JUPYTER:
Console: http://localhost:49001
Serviço de Armazenamento: http://localhost:49000
Portas:
- 4040
- 4041
- 4042
- 4043

README/GITHUB

	1 Apache Spark	2 jupyter	3 Apache Spark
Macro-etapas	Gerenciar os recursos e distribuir as tarefas entre os nós do cluster e Executar as tarefas distribuídas	Tratamento dos dados	Tratamento dos dados
Detalhamento	1. Executar o Spark Master com o comando deploy.master 2. Executar o Spark Worker com o comando deploy.worker	1. Executar o comando start notebook	1. Executar o comando start notebook
Pré-configurações			

3.ii Solução | Documentação da aplicação

INGESTÃO E ARMAZENAMENTO | PROCESSAMENTO E ACESSO | DATAVIZ

Credenciais de acesso:

Elastic Search:
- **URL**: http://localhost:49090

Portas Spark Master:
- ~~7077~~
- ~~8080~~

Password: “12345”

Portas Spark Worker: 8081

JUPYTER:

Console: http://localhost:49001

Serviço de Armazenamento: http://localhost:49000

Portas:
- 4040
- 4041
- 4042
- 4043

README/GITHUB

	1 elastic	2 kibana	3
Macro-etapas	XXXXXX	XXXXXX	XXXXXX
Detalhamento	1. XXXXXX	1. XXXXXX	1. XXXXXX
Pré-configurações			

3.ii Solução | Análise de dados

Camada Raw: dados brutos, sem transformações

Camada Trusted: Dados tratados

Camada Bussiness: Dados enriquecidos

Dados extraídos diretamente das camadas de origem (API Olho Vivo e do GTFS);

Dados referentes à solução de monitoramento near real time dos ônibus em circulação.

Ex.: Latitude e longitude dos veículos em tempo real, horários de chegada previstos,..

Integração dos dados a partir das múltiplas fontes (API e GTFS) e análise de possíveis cruzamentos de dados;

Tratamento dos dados e aplicação de regras de limpeza.

Dados enriquecidos;

Manipulação dos dados para consolidação em métricas e KPIs, definidos para atender às necessidades de negócio;

Base definida para consumo por sistemas de visualização e relatórios analíticos.

Exemplos de KPIs

- Velocidade média do ônibus
- Tempo total das viagens
- Tempo médio em paradas
- Horário estimado de chegada

Melhorias: avaliaão upgrade plano pago do mapa para visualização linhas tracejados

Limitação contronada pelo campo keyword