**Laporan Tugas Besar IF4054**
**Pengoperasian Sistem Perangkat Lunak**
**Semester I Tahun 2024/2025**

# Customer Churn Prediction Ops Pipeline


**Disusun oleh:**

13521128 Muhammad Abdul Aziz Ghazali

13521146 Muhammad Zaki Amanullah

13521109 Rizky Abdillah Rasyid

**PROGRAM STUDI TEKNIK INFORMATIKA**
**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**
**INSTITUT TEKNOLOGI BANDUNG**

**2025**

# Bab I
# Pendahuluan

## A. Latar Belakang

Dalam industri telekomunikasi, mempertahankan pelanggan menjadi tantangan yang semakin signifikan seiring dengan meningkatnya persaingan pasar dan beragamnya pilihan layanan yang tersedia. Salah satu tantangan utama yang dihadapi perusahaan telekomunikasi adalah customer churn, yaitu situasi di mana pelanggan memutuskan untuk berhenti menggunakan layanan suatu perusahaan dan beralih ke penyedia layanan lain. Fenomena ini tidak hanya mengurangi pendapatan perusahaan, tetapi juga meningkatkan biaya operasional karena kebutuhan untuk menarik pelanggan baru sebagai pengganti pelanggan yang hilang.

Pendekatan tradisional untuk menangani churn, seperti survei dan analisis manual, sering kali tidak memadai dalam mengidentifikasi pola perilaku pelanggan yang kompleks. Dengan kemajuan teknologi, khususnya dalam bidang big data dan machine learning, analisis data yang lebih mendalam dan prediktif menjadi memungkinkan. Pemanfaatan machine learning dapat membantu perusahaan dalam mengidentifikasi pelanggan yang berpotensi churn lebih awal dan menyediakan solusi yang lebih terarah untuk mempertahankan mereka. Namun, salah satu tantangan utama dalam penerapan model prediktif adalah fenomena data drifting, yaitu perubahan distribusi data yang terjadi seiring waktu. Data drifting dapat menyebabkan model kehilangan akurasi dan relevansi jika tidak dikelola dengan baik.

Untuk memastikan implementasi model machine learning yang efektif, diperlukan sebuah sistem yang dapat mengintegrasikan pengembangan, pelatihan, pengujian, dan deployment model secara otomatis dan berulang. Sistem ini juga harus dilengkapi dengan mekanisme untuk mendeteksi dan mengelola data drifting, seperti menggunakan Population Stability Index (PSI) untuk mendeteksi perubahan distribusi data. Konsep ini dikenal sebagai Machine Learning Operations (MLOps). Dengan menerapkan MLOps, perusahaan dapat memastikan pipeline pengembangan model berjalan secara efisien, scalable, dan mudah dikelola, sekaligus mampu beradaptasi terhadap perubahan data dan kebutuhan bisnis yang dinamis.

Proyek ini bertujuan untuk merancang dan mengimplementasikan sistem MLOps untuk prediksi customer churn pada industri telekomunikasi. Sistem ini akan menggabungkan teknologi seperti Apache Spark untuk pemrosesan data, Apache Airflow untuk orkestrasi workflow, dan MLflow untuk manajemen model. Dengan pendekatan ini, perusahaan dapat mengoptimalkan strategi retensi pelanggan secara proaktif, mengurangi churn, dan meningkatkan keuntungan, sambil memastikan bahwa model tetap relevan dalam menghadapi tantangan data drifting.

# Bab II

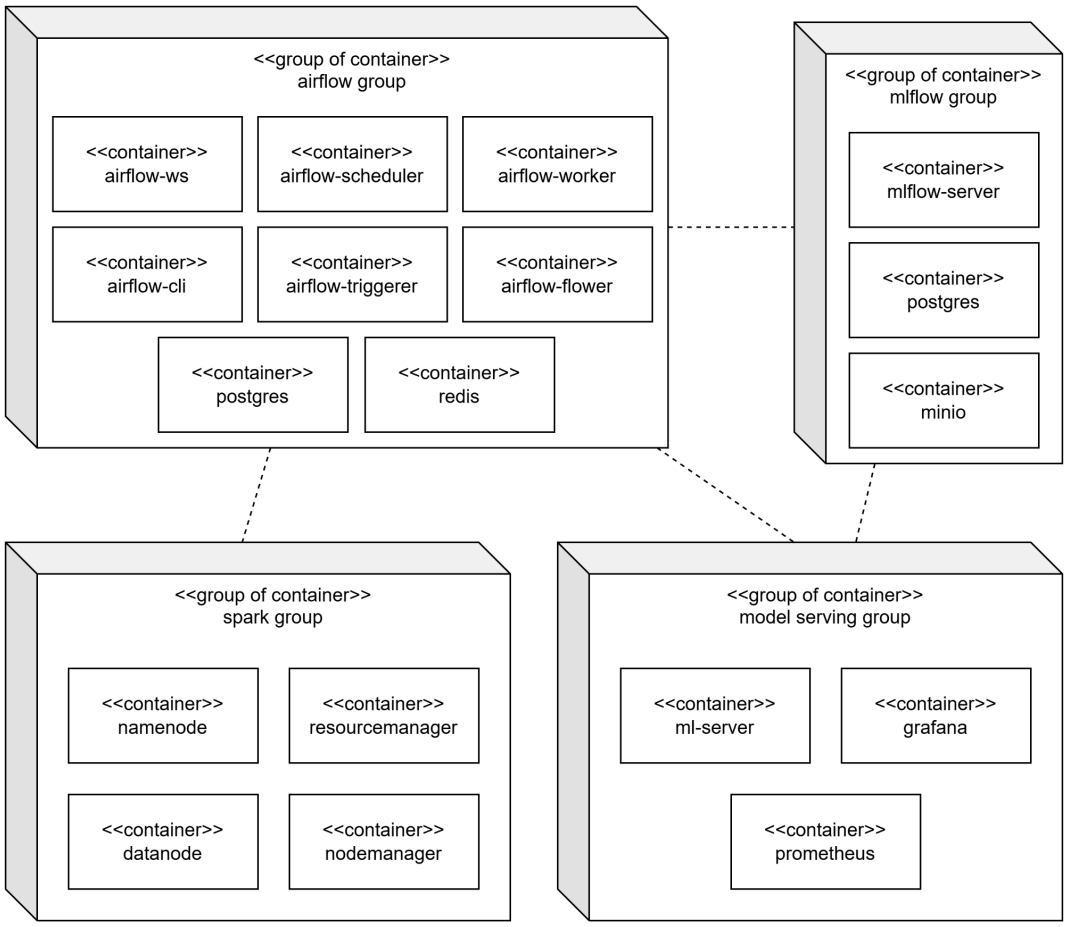
# Implementasi

## A. Tools

Pada tugas ini, tools yang digunakan adalah sebagai berikut berserta perannya.

1. Apache Spark
   Apache Spark digunakan untuk melakukan preprocessing data serta melakukan pelatihan model. Apache Spark sering digunakan dalam data preprocessing dan pelatihan model karena kemampuannya memproses data dalam skala besar secara terdistribusi dan efisien, baik dalam memori
2. Apache Airflow
   Apache Airflow berperan sebagai orkestrator untuk mengotomasi, menjadwalkan, dan memantau pipeline data dan machine learning secara terstruktur dan terkelola. Dengan mendefinisikan workflow sebagai kode (DAG sehingga memungkinkan integrasi berbagai pipeline data maupun machine learning.
3. Mlflow
   Mlflow berperan untuk mengelola siklus hidup dari machine learning mulai dari manajemen model dan tracking eksperimen. Pada sistem, mlflow berperan dalam menyimpan model dan mencatat setiap eksperimen dan metadata pada model yang dihasilkan.
4. Docker
   Docker berperan sebagai container dalam meluncurkan perangkat lunak yang dibutuhkan seperti, spark, airflow, mlflow dan model inference.
5. FastAPI
   FastAPI digunakan untuk membangun API yang cepat dan efisien dalam serving model pada lingkungan produksi.
6. Gitlab
   Pada sistem, gitlab digunakan untuk melakukan CI CD pada pipeline preprocessing dan training model. Selain itu, digunakan sebagai repository penyimpanan kode sumber.

## B. Arsitektur



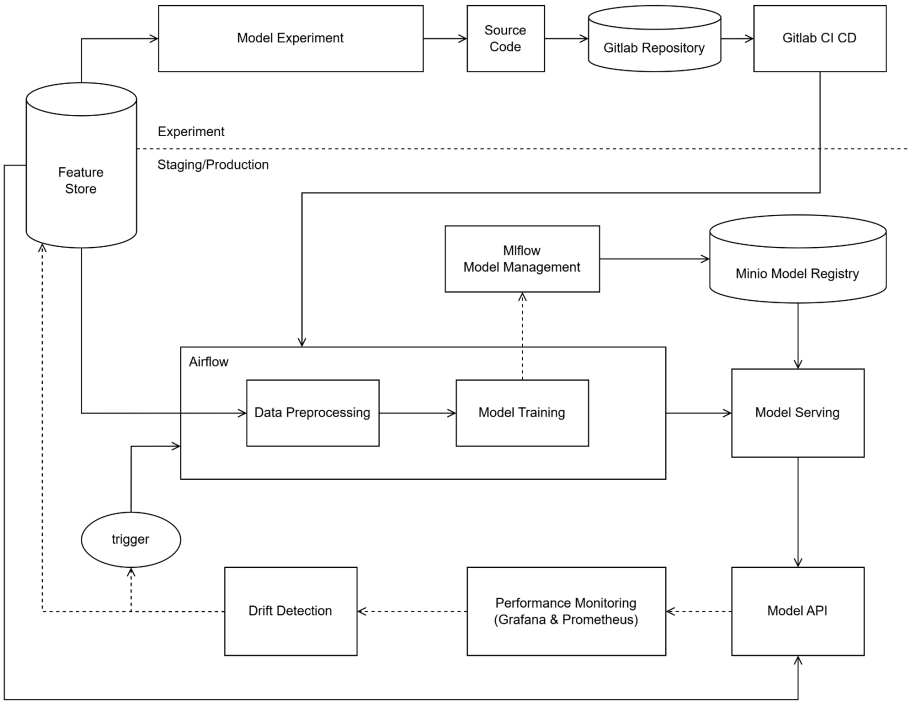

Gambar Deployment Diagram MLOps dan DataOps

Pada diagram diatas, grup container dibagi menjadi empat, yaitu spark, airflow, mlflow dan model-serving. Pada grup container airflow berisi container yang menunjang menjadwalkan dan otomasi pipeline pemrosesan data dan pelatihan model. Pada grup spark berperan sebagai spark yang menunjang pemrosesan data. Grup mlflow berperan dalam menyimpan model pada minio object storage dan menyimpan metadata-nya pada postgres. Terakhir adalah grup model serving yang digunakan untuk tempat aplikasi yang digunakan sebagai inference model dan melakukan monitoring dengan grafana dan prometheus.

## C. Alur Kerja Sistem

Alur kerja sistem yang dibangun mengacu pada alur proses pada MLOps Level 2 Menurut google. Berikut merupakan diagram alur pada sistem yang dikembangkan. erdasarkan diagram alur kerja yang ditampilkan, sistem ini terdiri dari beberapa

komponen utama yang saling terhubung untuk menghasilkan pipeline yang efisien. Data mentah dan fitur-fitur yang relevan disimpan dalam Feature Store, yang menjadi sumber utama untuk pemrosesan lebih lanjut. Apache Airflow digunakan untuk mengorkestrasi tugas-tugas pembersihan data dan transformasi fitur, memastikan data bebas dari null, string kosong, atau nilai tidak konsisten sebelum digunakan dalam pelatihan model.




Eksperimen model dilakukan dengan menggunakan data dari Feature Store untuk mengembangkan dan menguji berbagai algoritma machine learning. Semua kode sumber untuk eksperimen disimpan di GitLab Repository dan dikelola dengan GitLab CI/CD untuk memastikan integrasi dan pengiriman kode berjalan mulus. Setelah itu, data yang telah diproses digunakan untuk melatih model dengan memanfaatkan pipeline otomatis yang diatur oleh Airflow, sementara MLflow digunakan untuk mencatat, mengelola, dan melacak model yang dihasilkan selama proses pelatihan.

Model yang telah divalidasi dan memenuhi kriteria performa disimpan dalam Minio Model Registry untuk dikelola lebih lanjut, dan model yang telah disetujui disajikan melalui API model untuk digunakan dalam aplikasi produksi. Sistem menggunakan alat seperti Grafana dan Prometheus untuk memantau performa model secara real-time, termasuk metrik seperti akurasi prediksi dan waktu respons. Population Stability Index (PSI) digunakan untuk mendeteksi pergeseran data atau model drift yang dapat memengaruhi performa model di lingkungan produksi. Jika drift terdeteksi, pemrosesan ulang atau pelatihan ulang akan dipicu secara otomatis melalui Airflow.

Workflow dilengkapi dengan pemicu otomatis untuk memastikan bahwa proses retraining dimulai saat terjadi perubahan data yang signifikan. Model yang baru dilatih diintegrasikan kembali ke dalam pipeline menggunakan GitLab CI/CD untuk memastikan

konsistensi. Alur kerja ini dirancang untuk memastikan bahwa sistem prediksi churn dapat berjalan secara otomatis dan berulang dengan tingkat keandalan tinggi. Diagram alur kerja ini mencerminkan integrasi mendalam antara pengembangan, monitoring, dan deployment, yang memungkinkan penyesuaian cepat terhadap perubahan data dan kebutuhan bisnis.

# Bab III

# Hasil

## A. Repositori

https://gitlab.informatika.org/snoopidog/tubes-mlops-customerchurn.git

## B. Hasil Pekerjaan

Tugas Besar Pengoperasian Perangkat Lunak

## C. Pembagian Kerja

| NIM | Kontribusi |
|---|---|
| 13521128 | Apache Airflow Deployment, Container environment and orchestration. |
| 13521109 | MLFlow, FastAPI, Data drift simulation. |
| 13521146 | Eksperimentasi model *machine learning*, *CI/CD* pipeline. |

## D. Referensi

Referensi yang digunakan dalam pengerjaan tugas besar ini meliputi:


1. Dataset: "Telco Customer Churn" yang diakses dari Kaggle (https://www.kaggle.com/datasets/blastchar/telco-customer-churn).
2. Dokumentasi resmi Apache Spark (https://spark.apache.org/docs/latest/).
3. Dokumentasi resmi Apache Airflow (https://airflow.apache.org/docs/).
4. Dokumentasi resmi MLflow (https://mlflow.org/docs/latest/).
5. Dokumentasi resmi GitLab untuk pipeline CI/CD (https://docs.gitlab.com/ee/ci/).