📘 MODUL PRAKTIKUM 1

Setup Environment & Git Workflow

Track: Spark + Cloud (Industry Ready)

Language : **Python**
Editor : **VS Code**
Terminal : **PowerShell**
Durasi : **150 menit**

🎯 Tujuan Praktikum

Setelah praktikum ini mahasiswa mampu:

1. Menggunakan VS Code sebagai environment kerja
2. Menggunakan PowerShell untuk menjalankan perintah
3. Menginstal dan menjalankan PySpark
4. Membuat cluster gratis MongoDB Atlas
5. Membuat struktur project profesional
6. Menggunakan Git dan GitHub
7. Menjalankan Spark job sederhana

🧠 Gambaran Besar Workflow

Praktikum 1 ini membangun fondasi lingkungan kerja Data Engineer.

Alur yang akan disiapkan:

VS Code
→ PowerShell
→ PySpark
→ MongoDB Atlas (Cloud)
→ Struktur Project
→ Git Versioning
→ Spark Job

🚀 Technology Stack Praktikum 1

"Fondasi Data Engineer Modern"

Pada praktikum pertama ini, kita tidak sekadar install software.
Kita sedang membangun **stack teknologi dasar yang digunakan di industri Big Data modern.**

---

1 VS Code – Development Environment

VS Code adalah editor kode modern yang ringan dan fleksibel.

Kenapa dipilih?

- Digunakan luas di industri
- Mendukung Python & Git
- Bisa integrasi terminal
- Cocok untuk workflow data engineer

---

2 PowerShell – Command Line Interface

PowerShell digunakan sebagai terminal utama.

Kenapa penting?

Karena data engineer tidak hanya coding, tapi juga:

- Install dependency
- Menjalankan job
- Mengatur environment
- Mengelola Git

Terminal adalah alat kerja utama di dunia cloud.

---

3 Python 3.10 – Bahasa Pemrograman Utama

Python adalah bahasa paling populer di Big Data.

Alasan dipilih:

- Ekosistem besar
- Banyak library data
- Standar industri
- Stabil untuk PySpark

Kita menggunakan versi 3.10 karena paling kompatibel dengan Spark saat ini.

---

4 PySpark – Distributed Data Processing Engine

PySpark adalah interface Python untuk Apache Spark.

Fungsinya:

- Memproses data besar
- Melakukan agregasi
- Transformasi data
- Analisis terdistribusi

Konsep penting:

- Membuat SparkSession
- Membuat DataFrame
- Melakukan groupBy
- Menjalankan distributed job

Walaupun datanya kecil, arsitekturnya sudah distributed-ready.

---

5 MongoDB Atlas – Cloud Database

MongoDB Atlas adalah database NoSQL berbasis cloud.

Kenapa dipakai?

- Free tier tersedia
- Simulasi cloud production
- Digunakan di banyak startup & enterprise

Mahasiswa belajar:

- Membuat cluster cloud
- Mengatur database access
- Mengatur network access
- Mengambil connection string

Ini sudah menyentuh konsep:
Cloud-native data architecture

---

6 Git & GitHub – Version Control & Collaboration

Git digunakan untuk:

- Mengelola versi kode
- Melacak perubahan
- Kolaborasi
- Portfolio profesional

Mahasiswa belajar:

- git init
- git add
- git commit
- git push
- .gitignore
- License

Ini bukan sekadar tugas.
Ini adalah skill profesional.

🧠 Gambaran Arsitektur Sederhana

Stack ini sebenarnya sudah mencerminkan arsitektur industri:
Local Development
↓
Spark Processing Engine
↓
Cloud Database (MongoDB Atlas)
↓
Version Control (GitHub)
Kita sedang melatih mahasiswa dengan pendekatan:
Local development + Cloud integration + Distributed processing

---

🎯 Filosofi Stack Ini

Praktikum 1 bukan tentang analitik.

Praktikum 1 adalah tentang:

- Environment reproducibility
- Cloud readiness
- Distributed computing mindset
- Professional workflow

Ini fondasi sebelum masuk:

- Data ingestion
- ETL pipeline
- Data warehouse
- Streaming
- Machine learning

🔥 Kenapa Ini Disebut "Industry-Ready Stack"?

Karena kombinasi ini:

✓ Digunakan startup
✓ Digunakan perusahaan teknologi
✓ Digunakan data engineer junior
✓ Cloud-compatible
✓ Scalable

Hari ini mungkin hanya membuat:

groupBy().sum()

Tapi di balik itu, kalian sudah:

- Menjalankan distributed engine
- Terhubung ke cloud database
- Mengelola project dengan Git
- Bekerja dengan workflow profesional

🟢 BAGIAN 1 – Persiapan Awal (WAJIB)

1️⃣ Install Software

Pastikan sudah terpasang:

- Python 3.10+
- Git
- VS Code

2️⃣ Buka VS Code

- Klik "Open Folder"
- Buat folder baru bernama:

bigdata-project

Klik **Select Folder**

3️⃣ Set Terminal ke PowerShell

Tekan:

Ctrl + Shift + P

Ketik:

Terminal: Select Default Profile

Pilih:

PowerShell

Buka terminal:

Ctrl + J

BAGIAN 2 – Install & Test PySpark

Step 1 – Cek Python

Di terminal VS Code (PowerShell):

```
python --version
```

Jika muncul versi → lanjut.

Step 2 – Install PySpark

```
pip install pyspark
```

Step 3 – Test PySpark

Ketik:

```
python
```

Lalu masukkan:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder \
    .appName("TestSpark") \
    .getOrCreate()

print("Spark berhasil dijalankan!")
```

Jika tidak error → sukses.

Keluar:

```
exit()
```

BAGIAN 3 – Setup MongoDB Atlas (Cloud)

Step 1 – Login

Buka:
https://www.mongodb.com/atlas

Login / Daftar.

Step 2 – Buat Cluster Gratis

Di Dashboard:

1. Klik **Build a Database**
2. Pilih **M0 Free Tier**
3. Provider: **AWS**
4. Region: Singapore (atau default)
5. Klik **Create**

Tunggu hingga status **Active**.

Step 3 – Buat Database User

Klik:

Database Access → Add New Database User

Isi:

- Username
- Password sederhana (hindari simbol # @)

Klik Save.

Step 4 – Tambahkan IP

Klik:

Network Access → Add IP Address

Pilih:

Allow Access from Anywhere (0.0.0.0/0)

Klik Confirm.

Step 5 – Ambil Connection String

Klik:

Database → Connect → Connect your application

Driver:

Python

Copy connection string.

Step 6 – Test Koneksi

Install pymongo:

```
pip install pymongo
```

Buat folder:

```
scripts
```

Buat file:

```
scripts/test_mongo.py
```

Isi:

```
from pymongo import MongoClient

uri = "PASTE_CONNECTION_STRING_DI_SINI"

try:
    client = MongoClient(uri)
    print("Koneksi berhasil!")
    print(client.list_database_names())
except Exception as e:
    print("Koneksi gagal:", e)
```

Jalankan:

```
python scripts/test_mongo.py
```

Jika berhasil → lanjut.

BAGIAN 4 – Setup Cloud Storage (Simulasi)

Di folder project, buat:

```
cloud_storage
data
notebooks
reports
```

Struktur akhir:

```
bigdata-project/
│
├── data/
├── cloud_storage/
├── scripts/
├── notebooks/
├── reports/
├── requirements.txt
└── README.md
```

Cloud storage hari ini masih simulasi lokal.

BAGIAN 5 – Git Workflow

Step 1 – Inisialisasi Git

Di terminal:

```
git init
```

Step 2 – Tambahkan File

```
git add .
git commit -m "initial project setup"
```

Step 3 – Push ke GitHub

1. Buat repo baru di GitHub
   a. Buka https://github.com
   b. Login ke akun anda
   c. Klik tombol "+" di pojok kanan atas
   d. Pilih "New repository"

Isi:
Repository name → contoh: bigdata-technology-2026
Pilih Public
⚠ Jangan centang "Initialize with README" kalau project sudah ada di lokal

e. Klik Create repository

2. Copy URL repo

Setelah repo berhasil dibuat, kamu akan diarahkan ke halaman repo baru.
Di halaman itu:

a. Klik tombol hijau "**Code**"
b. Akan muncul popup
c. Pastikan tab HTTPS dipilih
d. Klik ikon 📋 (Copy) di sebelah URL

Contoh URL yang dicopy:
https://github.com/muhayat-lab/bigdata-technology-2026.git

Lalu jalankan:

```
git remote add origin https://github.com/muhayat-lab/bigdata-technology-2026.git
git branch -M main
git push -u origin main
```

Jika berhasil → project online.

🟢 BAGIAN 6 – Spark Job Sederhana

Buat file:

scripts/simple_job.py

Isi:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder \
    .appName("SimpleJob") \
    .getOrCreate()

data = [("A", 10), ("B", 20), ("A", 30)]
columns = ["category", "value"]

df = spark.createDataFrame(data, columns)

df.groupBy("category").sum("value").show()

spark.stop()
```

Jalankan:

python scripts/simple_job.py

Jika muncul hasil agregasi → sukses.

```
+--------+----------+
|category|sum(value)|
+--------+----------+
|A       |40        |
|B       |20        |
+--------+----------+
```

📦 OUTPUT WAJIB

Mahasiswa wajib mengumpulkan:

1. Screenshot Spark berjalan
2. Screenshot MongoDB Atlas cluster Active
3. Link GitHub repository
4. File simple_job.py
5. Screenshot hasil eksekusi Spark

✅ 1️⃣ Screenshot Spark Berjalan

📌 Yang harus terlihat:

- Terminal VS Code
- Tidak ada error merah
- SparkSession berhasil dibuat

Contoh tampilan yang benar:

```
PS I:\bigdata-project> python
>>> from pyspark.sql import SparkSession
>>> spark = SparkSession.builder.appName("TestSpark").getOrCreate()
>>> print("Spark berhasil dijalankan!")
Spark berhasil dijalankan!
```

✅ 2️⃣ Screenshot MongoDB Atlas Cluster Active

📌 Yang harus terlihat:

Masuk ke MongoDB Atlas → Database

Harus terlihat:

Cluster0
Status: ACTIVE
Tier: M0 (Free)
Region: Singapore (atau sesuai)
Contoh tampilan:
Project: BigData

Cluster0
M0 FREE
Status: ACTIVE

📸 Screenshot harus menampilkan:

- Nama cluster
- Status ACTIVE (warna hijau)
- Tier M0

✅ 3️⃣ Screenshot Link GitHub Repository

📌 Yang harus terlihat:

- URL repository di browser
- File project terlihat
- README.md ada
- Folder scripts terlihat

Contoh URL:
https://github.com/username/bigdata-spark-praktikum

Tampilan:

bigdata-spark-praktikum
├── data/
├── scripts/
├── cloud_storage/
├── README.md

📸 Screenshot harus menampilkan:

- URL bar browser
- Isi repository

✅ 4️⃣ Screenshot File simple_job.py

📌 Yang harus terlihat:

File terbuka di VS Code dengan isi seperti ini:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder \
   .appName("SimpleJob") \
   .getOrCreate()

data = [("A", 10), ("B", 20), ("A", 30)]
columns = ["category", "value"]

df = spark.createDataFrame(data, columns)
df.groupBy("category").sum("value").show()

spark.stop()
```

📸 Screenshot harus menampilkan:

- VS Code
- File simple_job.py
- Isi kode lengkap

✅ 5️⃣ Screenshot Hasil Eksekusi Spark

📌 Yang harus terlihat:

```
PS I:\bigdata-project> python scripts/simple_job.py
```

```
+--------+----------+
|category|sum(value)|
+--------+----------+
|A       |40        |
|B       |20        |
+--------+----------+
```

📸 Screenshot harus menampilkan:

- Perintah python scripts/simple_job.py
- Output tabel agregasi
- Tidak ada error

🎯 Checklist Validasi

Item	Valid Jika
Spark Screenshot	Tidak ada error merah
MongoDB	Status ACTIVE
GitHub	Repo public & bisa diakses
simple_job.py	Kode lengkap
Output Spark	Tabel agregasi muncul

📊 Rubrik Penilaian (Skor 1–4)

Aspek	4	3	2	1
PySpark setup	Berjalan sempurna	Minor error	Banyak error	Tidak berhasil
MongoDB setup	Stabil	Terkoneksi	Sering gagal	Tidak berhasil
Struktur project	Profesional	Cukup rapi	Kurang jelas	Tidak sesuai
Git workflow	Commit & push benar	Commit saja	Ada error	Tidak pakai Git
Spark job	Berjalan benar	Sebagian	Banyak bug	Tidak jalan

🎓 Insight Akhir

Hari ini kita belum belajar analitik.

Tapi kita sudah membangun:

- Lingkungan kerja *Data Engineer*
- *Cloud database*
- *Version control*
- *Distributed processing tool*

Ini adalah fondasi karier Big Data.