



Development Dashboard Safety Stock (Slow Moving) s.d. September 2022

Jakarta, November 2022



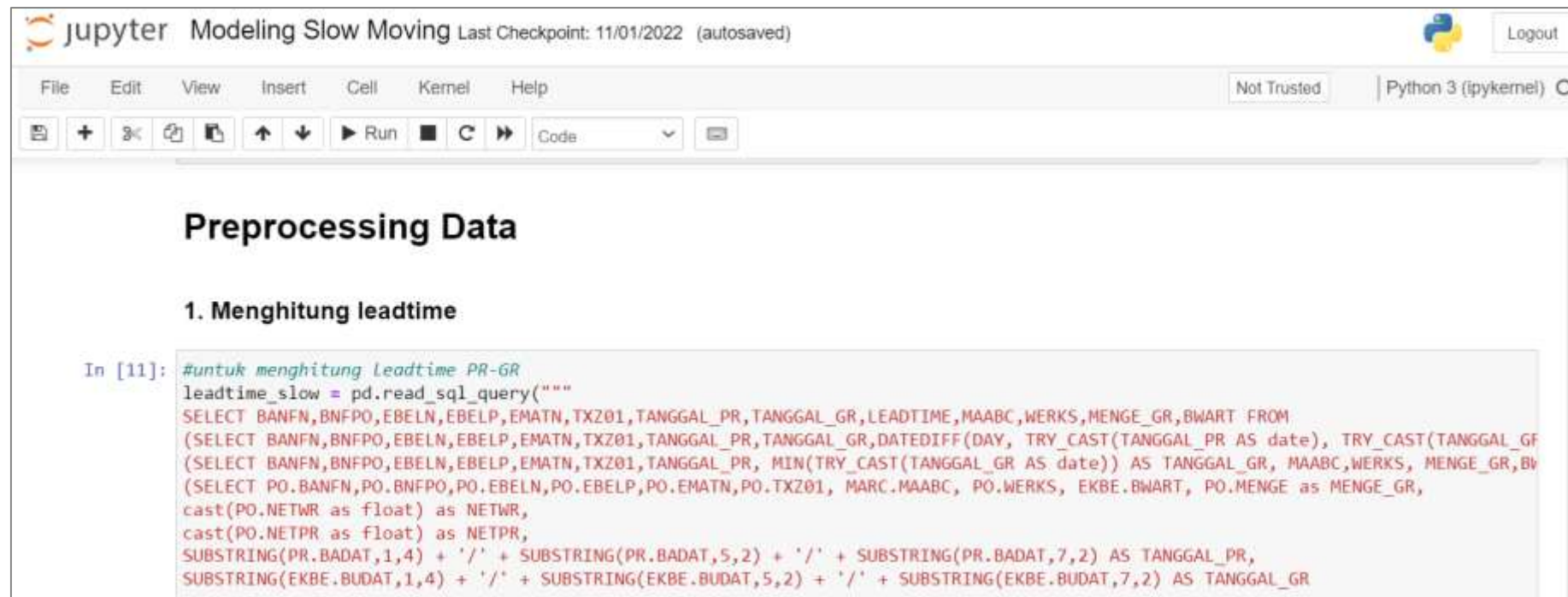
No.	Program Kerja	Timeline	% Progress	Status	Realisasi s.d. September 2022	Tindak Lanjut
		Sep '22 s.d Des '22	25%	Deployment Bidang Pengadaan	<ul style="list-style-type: none"> - Business Understanding: memahami proses bisnis management inventory khususnya sparepart kategori slow moving. Berkoordinasi dengan user dalam menentukan pengolahan data yang sesuai - Data Understanding: menentukan parameter yang dibutuhkan dalam pengolahan data, yaitu kode anak perusahaan, equipment, material, leadtime, dll - Data Preparation: masuk ke program python, aggregasi data, missing values checking 	Modelling dan deployment

Lampiran Pendukung

Beberapa referensi yang digunakan:

1. <https://media.neliti.com/media/publications/226929-perencanaan-pengadaan-suku-cadang-berdas-14d2a39b.pdf>
2. https://drive.google.com/drive/folders/1aK9WC0yqv2JvpDtIXW4T_1Hu4XWYIV_3

Script Penarikan Data di Program Python:



The screenshot shows a Jupyter Notebook interface with the title 'Modeling Slow Moving' and a last checkpoint of '11/01/2022 (autosaved)'. The notebook is running Python 3 (ipykernel). The main heading is 'Preprocessing Data', followed by the sub-heading '1. Menghitung leadtime'. The code cell shows a SQL query to calculate leadtime from a database.

```
In [11]: #untuk menghitung Leadtime PR-GR
leadtime_slow = pd.read_sql_query("""
SELECT BANFN,BNFPO,EBELN,EBELP,EMATN, TXZ01,TANGGAL_PR,TANGGAL_GR,LEADTIME,MAABC,WERKS,MENGE_GR,BWART FROM
(SELECT BANFN,BNFPO,EBELN,EBELP,EMATN, TXZ01,TANGGAL_PR,TANGGAL_GR,DATEDIFF(DAY, TRY_CAST(TANGGAL_PR AS date), TRY_CAST(TANGGAL_GR AS date)) AS TANGGAL_GR, MAABC,WERKS, MENGE_GR,BWART FROM
(SELECT BANFN,BNFPO,EBELN,EBELP,EMATN, TXZ01,TANGGAL_PR, MIN(TRY_CAST(TANGGAL_GR AS date)) AS TANGGAL_GR, MAABC,WERKS, MENGE_GR,BWART FROM
(SELECT PO.BANFN,PO.BNFPO,PO.EBELN,PO.EBELP,PO.EMATN,PO.TXZ01, MARC.MAABC, PO.WERKS, EKBE.BWART, PO.MENGE as MENGE_GR,
cast(PO.NETWR as float) as NETWR,
cast(PO.NETPR as float) as NETPR,
SUBSTRING(PR.BADAT,1,4) + '/' + SUBSTRING(PR.BADAT,5,2) + '/' + SUBSTRING(PR.BADAT,7,2) AS TANGGAL_PR,
SUBSTRING(EKBE.BUDAT,1,4) + '/' + SUBSTRING(EKBE.BUDAT,5,2) + '/' + SUBSTRING(EKBE.BUDAT,7,2) AS TANGGAL_GR
```



The screenshot shows the output of a Jupyter Notebook cell. It displays the command to save a CSV file and the resulting table of leadtime data.

```
In [6]: #leadtime_avg.to_csv("Leadtime_avg.csv", index = False)
leadtime_avg
```

Out[6]:

WERKS	EMATN	MAABC	LEADTIME
6000089	B	62	000000
6000973	B	91	000000
B001	6001290	B	55 000000



The screenshot shows a Jupyter Notebook cell with code to merge the leadtime data into a main dataset.

```
In [12]: dataset = pd.merge(GI_tabel, leadtime_avg, on = ["EMATN","WERKS"], how = 'inner')
dataset["cut_date"] = '2021-12-31'
dataset["cut_date"] = dataset["cut_date"].apply(pd.to_datetime, errors='coerce')
dataset["operate_day"] = dataset["cut_date"] - dataset["DATAB"]
dataset["operate_day"] = pd.to_numeric(dataset["operate_day"].dt.days, downcast='integer')
```