

Airport

Simuleret Analyse- og Rapportløsning i Power BI

Abrahim Borgi

Projektformål.....	2
Process & Workflow	3
ETL.....	4
Data Model.....	5
Power BI Rapport	6
Side 1: Overblik.....	7
Side 2: Passagerflow & Tidsmønstre	8
Side 3: Kapacitet & Udnyttelse	9
Opsumming.....	10
Relevante filer.....	10

Projektformål

Dette projekt er udviklet med tre centrale formål:

- At levere et realistisk end-to-end projekt under tidspres og uden adgang til interne systemer eller faktisk data - **i stedet gennem syntetiske data og eksterne kilder.**

Resultatet er en fuld data- og rapporteringspipeline tilpasset Airport.

Den omfatter simulering af passagerer og billetter, integration af faktiske flyafgange, kobling til flymodeller og kapaciteter samt **opbygning af et Power BI-dashboard** med KPI'er for passagerflow, check-in og sikkerhedstider samt belægningsgrader.

⚠ Bemærk: Billet- og passagerdata er fuldt syntetiske. De er baseret på antagelser og statistiske fordelinger, som gerne skulle afspejle plausible mønstre for passagere.

Process & Workflow

Dette projekt er bygget i faser med en iterativ tilgang, så hver del kunne testes, valideres og forfines undervejs:

Step 1: Scoping

- Definerede centrale forretningsspørgsmål inspireret af jobopslaget samt generelt kendskab til lufthavne (fx passagerflow, kapacitetsudnyttelse, rettidighed).
- Designede en fleksibel datamodel: passagerer, pas, fly, billetter og flymodeller.

Step 2: Data Simulation & Kodning (Python / Docker)

- Simulerede realistiske data for passagerer (pasnumre, lande), check-in typer samt tidsstempler for check-in og security.
- Integrerede eksterne data via historiske flyafgange fra API (AeroDataBox) og Lufthavne (opendatasoft).
- Kombinerede syntetiske og virkelige data for at skabe et konsistent datasæt med relationer mellem fly og billetter.
- Brug docker til at køre de forskellige services

Step 3: Database / DBT

- Strukturerede rene tabeller med primær-/fremmednøgler og datatyper.
- Udvikler Raw > Staging > Mart, struktur ved brug af dbt.
- Oprettede views (planlagt vs faktisk runway-tid).

Step 4: Power BI

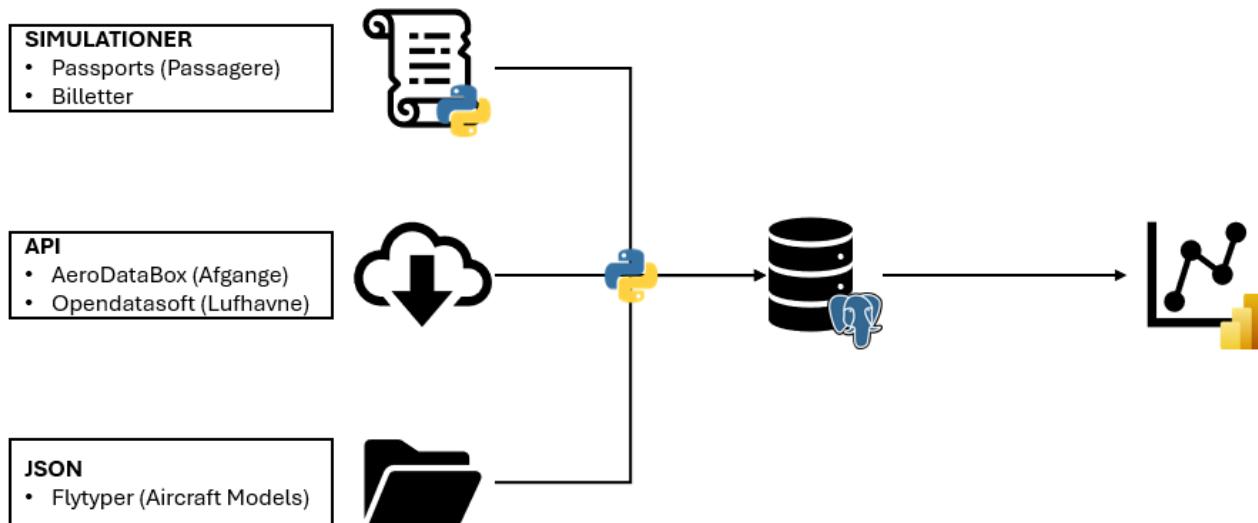
- Udvikle semantisk model ved brug af tabular editor (TE2).
- Designede interaktive rapportsider (Overblik, Passagerflow & Tidsmønstre, Capacity & Coverage).
- Implementerede KPI'er, visualiseringer og feltslicere, der giver overblik over passagermængder, tidsmønstre og belægningsgrader.

Step 5: Dokumentation

- Skrev struktureret dokumentation for scripts, datamodel og opbygning.
- Udarbejdede screenshots og eksporterbare rapportfiler til brug i ansøgningen.

ETL

Projektet er udviklet som en komplet ETL-pipeline, hvor forskellige datakilder bringes sammen i en PostgreSQL-database og visualiseres i Power BI.



Kilder og indlæsning:

- **Simulationer (Python):** Genererede syntetiske data for *Passports (passagerer)* og *Billetter*, inkl. check-in og security-tider.
- **API'er:**
 - *AeroDataBox* – hentede historiske afgange fra Airport.
 - *Opendatasoft* – metadata om lufthavne.
- **JSON-filer:** Importerede flytyper og deres karakteristika (kapacitet, rækkevidde, motorer).

Transformering og lagring:

- Python-scripts håndterede sammenkobling, datarensning og -kontrol.
- Python scripts bliver kørt af de 3 docker containere
 - service_api_to_client_airport
 - service_create_table_views_from_sql_airport
 - service_json_to_client_airport
 - service_simulations_airport
- Upsert-logik blev anvendt til at sikre, at data i PostgreSQL altid var ajourført.
- Alt blev gemt i relationelle tabeller med primær- og fremmednøgler.

- Transformering fra ingested data til semantisk model er udviklet i dbt.

Visualisering:

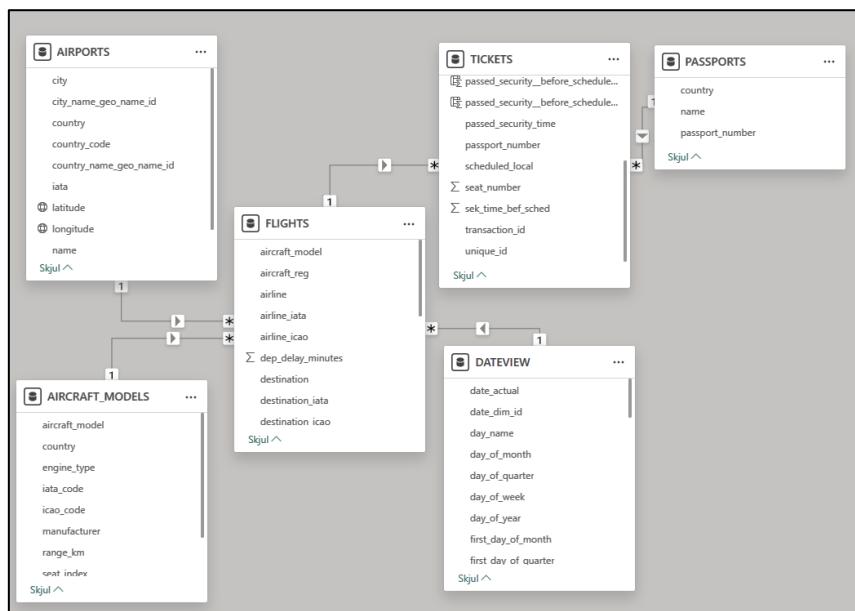
- Semantisk Model opbygget i Tabular Editor (TE2)
- Power BI koblet direkte til PostgreSQL.
- Tre rapportsider blev udviklet: *Overview, Passenger Flow & Timing og Capacity & Coverage.*

Pipeline'en sikrer dermed et samlet flow fra datagenerering og eksterne kilder til en brugervenlig rapport, der kan understøtte analyser og beslutninger

Data Model

Datamodellen er designet til at afspejle et forenklet, men realistisk billede af Airport's operationelle dataunivers.

Den kombinerer både det simulerede og eksterne data som danner grundlag for rapportering i Power BI.



Tabeller:

- **PASSPORTS** – Simulerede passagerer med unikke pasnumre og tilhørende nationalitet (navn, land).
- **FLIGHTS** – Faktiske afgange fra Airport hentet via API (AeroDataBox), inkl. planlagt og faktisk afgangstid, status, terminal, gate, rutenummer og rute-koder (IATA/ICAO).
- **TICKETS** – Simulerede billetter koblet til fly og passagerer. Indeholder bl.a. sædenummer, check-in type, check-in tid og tidspunkt for passage af security.

- **AIRCRAFT_MODELS** – Metadata om flytyper (fra JSON), herunder sædekapacitet, rækkevidde, motortype samt IATA/ICAO-koder og producent.
- **AIRPORTS** – Referenceoplysninger om lufthavne (IATA, navn, by, land, koordinater) anvendt til rute- og destinationsvisning.
- **DATEVIEW** – Datodimension.

Power BI Rapport

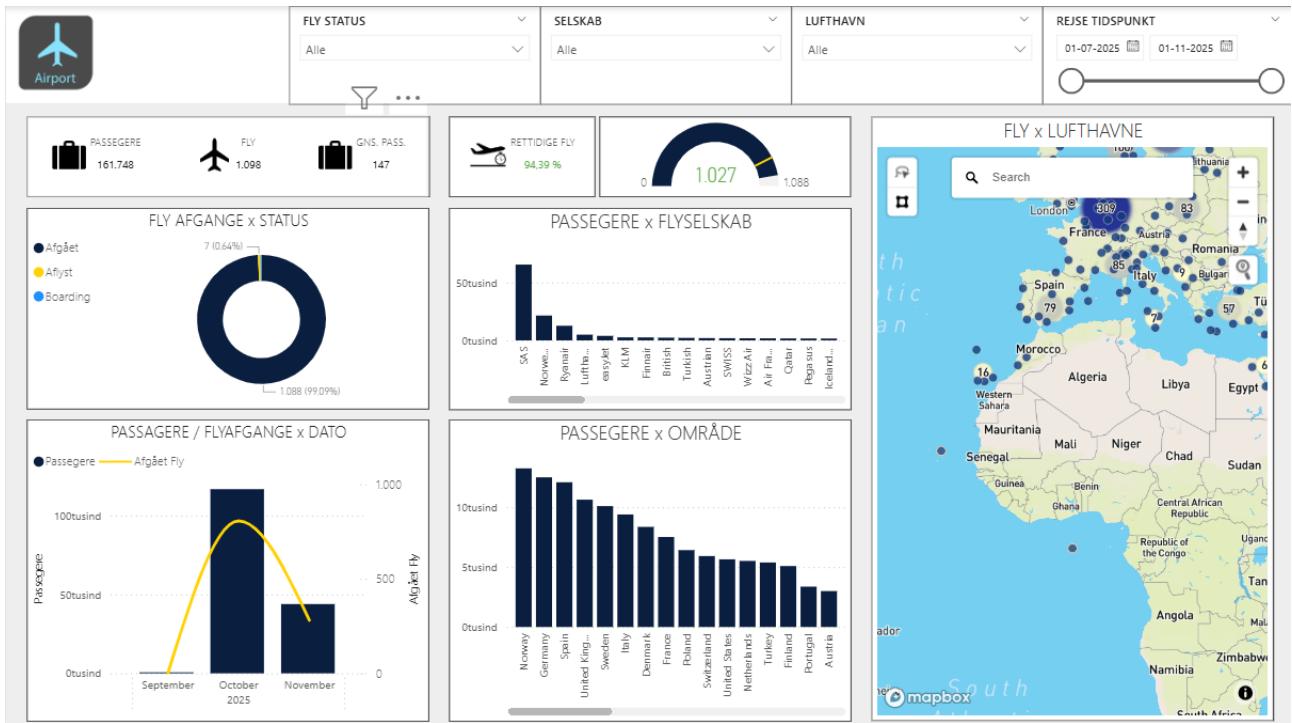
Rapporten er struktureret i tre sider, hver med fokus på centrale forretningssspørgsmål for en lufthavn. Visualiseringerne er bygget med et ensartet layout og et tilpasset Airport-tema, så rapporten både fremstår professionel og let at navigere i.

- **Page 1: Overblik** giver et overordnet billede af passagerer, fly og om flyene afgår rettidigt.
- **Page 2: Passagerflow & Tidsmønstre** går i dybden med hvor tidligt passagerer checker ind og passerer security i forhold til afgangstid.
- **Page 3: Capacity & Coverage** analyserer hvor godt flyenes sædekapacitet udnyttes.

Alle sider understøtter interaktive slicere for flystatus, selskab, lufthavn og tidsperiode.

Side 1: Overblik

Første side giver et overblik over trafikken i lufthavnen for den valgte periode.



Den er designet til at svare på spørgsmål som:

- Hvor mange passagerer og fly er håndteret i perioden?
- Hvor stor en andel af flyene afgik til tiden?
- Hvilke flyselskaber og destinationer fylder mest?
- Hvornår er belastningen størst fordelt over tid?

Denne side fungerer som en indgang, hvor **ledelse og interesserter hurtigt kan få indsigt i trafiktal, rettidighed og de vigtigste mønstre på tværs af selskaber, destinationer og tid.**

Side 2: Passagerflow & Tidsmønstre

Anden side fokuserer på passagerernes rejse gennem lufthavnen – fra check-in til security og frem mod afgang.



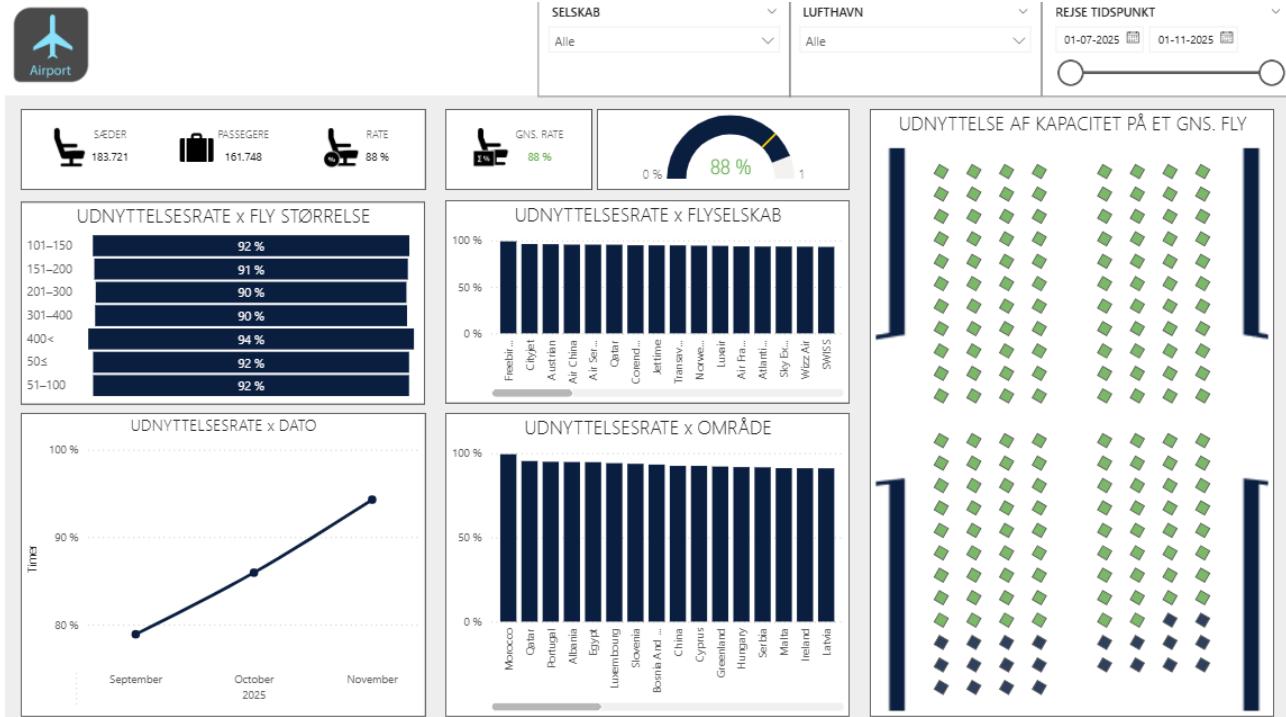
Den er designet til at svare på spørgsmål som:

- Hvor tidligt checker passagerer ind og går gennem security i forhold til afgangstid?
- Hvilke forskelle er der mellem online check-in og onsite check-in?
- Hvor stor en andel af passagererne kommer igennem security i ”god tid”, og hvor mange er sent ude?
- Hvordan varierer check-in og security-mønstre over dagen og på tværs af datoer?

Denne side giver indblik i passagerflowet, og kan dermed bruges til at vurdere **kapacitetsbehov i terminal og security**, samt **identificere perioder med risiko for flaskehalse**.

Side 3: Kapacitet & Udnyttelse

Tredje side undersøger hvor effektivt flyenes sædekapacitet bliver udnyttet.



Den er designet til at svare på spørgsmål som:

- Hvor høj er den gennemsnitlige udnyttelsesrate (belægning) på tværs af alle fly?
- Hvordan varierer udnyttelsen efter flystørrelse, flyselskab og område?
- Hvilke dage viser høj eller lav udnyttelse, og kan der ses mønstre over tid?
- Hvordan ser et gennemsnitligt fly ud, når udnyttelsesgraden visualiseres på sædeniveau.

Denne side giver indblik i, hvordan efterspørgslen matcher den tilgængelige kapacitet. **Den kan bruges til at vurdere hvorvidt ressourcer (flytyper og ruter) udnyttes optimalt.**

Opsummering

Projektet demonstrerer min evne til at udvikle en end-to-end data- og rapporteringsløsning for Airport.

Jeg har kombineret simulerede og eksterne data til en konsistent datamodel og **bygget en Power BI-rapport, der giver indsigt i passagerflow, rettidighed og kapacitetsudnyttelse.**

Resultatet viser, hvordan jeg i praksis kan omsætte komplekse datakilder til brugbare analyser og visualiseringer, der understøtter beslutningstagning, samt visulisere dette på en fornuftig måde.

Relevante filer

- **Github-projekt** (<https://github.com/AbrahimBorgiPrivat/Copenhagen-Airport-Case>) med hele casen, herunder:
 - Alle Python-scripts til datagenerering, upserts og orkestrering
 - PostgreSQL-skema og eksempel på datastruktur
 - Ekstern dataindlæsning (API'er og JSON)
 - PBIP-fil til Power BI