PDF-Downloader – Præsentation

Slide 1 – Titel

PDF-Downloader (Specialisterne – virkelig kunde-case)

- Udviklet i .NET 9
- Fokus: Fejlhåndtering, multithreading og fil-IO

Slide 2 – Case & krav

- Kunden har et gammelt Python-script der fejler.
- Behov for et pålideligt .NET-program som:
- Downloader PDF-rapporter fra Excel/CSV.
- Bruger fallback URL hvis primær fejler.
- Navngiver filer efter *BRNummer*.
- Gemmer status for hver rapport.

Slide 3 – Løsnings-overblik

- .NET 9 Console App
- Pipeline: **Metadata → Download → Rapport**
- Konfigurerbar via CLI
- Fokus på robusthed og skalerbarhed

Slide 4 – Arkitektur

- **Program.cs** – CLI & Ctrl+C
- **ApplicationRunner** – Orkestrerer hele kørselen
- **MetadataLoader** – Læser Excel/CSV
- **DownloadManager** – HTTP, concurrency, fallback, skip
- **StatusReportWriter** – Genererer CSV-rapport

Slide 5 – Metadata indlæsning

- Understøtter Excel (ClosedXML) & CSV (CsvHelper)
- Kolonner: `BRnum`, `Pdf_URL`, `Pdf_URL_Alt`
- Validerer manglende/ugyldige URLs
- Konfigurerbar via CLI

Slide 6 – CLI

Eksempel:

```
dotnet run --
  --input "..\samples\Metadata2006_2016.xlsx"
  --output "./Downloads"
  --status "./Downloads/status.csv"
  --id-column "BRnum"
  --url-column "Pdf_URL"
  --fallback-url-column "Pdf_URL_Alt"
  --limit 0
  --max-concurrency 50
```

- `--limit` styrer antal rækker
- `--max-concurrency` styrer parallelle downloads
- `--no-skip-existing` overskriver gamle filer

Slide 7 – Concurrency (OH-SHIT MOMENT)

- Første kørsel forsøgte at hente **26.923 PDF'er samtidig** 😱
- Maskinen gik i knæ – læring: total tasks ≠ samtidige tasks
- Løsning: semafor med `MaxConcurrency`
- Typisk: 10 på netværk / 50 hjemme
- Resultat: Stabil performance og kontrolleret belastning

Slide 8 – Fejlhåndtering

- Udfald:
- `Downloaded`
- `SkippedExisting`
- `Failed`
- `NoUrl`
- Meningsfulde beskeder: HTTP, Timeout, Content-Type
- **CancellationToken** sikrer clean shutdown

Slide 9 – Filhåndtering

- Filnavn = renset `Id`

- Skip eksisterende filer
- Outputmappe oprettes automatisk
- Idempotent genkørsel

Slide 10 – Statusrapport

- CSV output:
- `Id`, `Outcome`, `Message`, `SourceUrl`, `SavedFile`
- Brugbar i Excel eller audit
- Mulighed for fortsatte kørsler baseret på rapport

Slide 11 – Demo

1. `--limit 10 --max-concurrency 5`
2. `--max-concurrency 50`
3. Vis `status.csv`
4. Re-run → "SkippedExisting" vokser

Slide 12 – Læring & næste skridt

Styrker: - Klar arkitektur og SoC - Stabil multithreading - Tydelig CLI og statusrapport

Svagheder: - Ingen DB eller resume-funktion - Simpel retry-strategi

Forbedringer: - Eksponentiel backoff - Structured logging og metrics - Unit tests og CI/CD - GUI til batchovervågning