# Training dashboard

## Målsättning

Vi vill skapa en dashboard där man visar data konditionsdata från Garmin connect appen (Löpning/simmning/utomhuscykling/inomhuscykling) och data från Strengthlogappen. Detta för att det inte finns någon dashboard som kombinerar data från Garmin och Strengthlog.

## Genomförare av projektet

Erik Unevik och Alexander Hrachovina.

## Affärsidén

Garmin är ett av världens ledande företag när det kommer till träningsklockor med mer avancerade funktioner och metrics. Strengthlog appen är svenskutvecklad och är inriktad mot styrketräning. Oss veterligen finns det ingen app som genererar data från bägge dessa applikationer vilket vi tänker ändra på.

## Presenterat innehåll från Garmin appen

Vi tänker oss metrics man kan sortera på enskilda pass, vecka, månad, år för val av sport (Löpning/simmning/utomhuscykling/inomhuscykling:

* Distans
* Medelpuls
* Antal löpta/cyklade höjdmeter
* Karta över cyklade/löpta sträckan
* Tid
* Förbrända kalorier
* Medelhastighet
* Träningseffekt

Så här ser en screenshot ut från Garmins hemsida idag för inspiration:

En bild som visar text, skärmbild, diagram, programvara

AI-genererat innehåll kan vara felaktigt.

## Presenterat innehåll från Strengthlog appen

* Tid
* Lyft vikt i kilon
* Eventuellt personbästa

## Data och insamling

Data inhämtning kommer ske i två separata delar från Garmin respektive Strengthlog.

### Garmin

Vi planerar att hämta data från Garmins API dit vi kommer göra en ansökan om tillgång:

<https://www.garmin.com/en-US/forms/GarminConnectDeveloperAccess/>

### Strengthlog

Då man kan exportera datan till en CSV fil från appen så har vi prövat det, men fick förvarning av deras team ”att den går det inte att göra mycket vettigt med, vilket verkade stämma, se screenshot nedan.

En bild som visar text, skärmbild, Teckensnitt, nummer

AI-genererat innehåll kan vara felaktigt.

Vi kommer därför i avgränsningssyfte bruka en egen träningsdagbok som sparas i en CSV fil där vi sparar variablerna:

* Datum
* Tid
* Lyft vikt i kilon

En målsättning om möjligt är att man i ett dokument likt google drive eller annan app ska kunna föra in träningsstatistiken enkelt.

## Tekniker och metodologi

Vi använder en modern data warehouse‑inspirerad arkitektur, där vi kommer använda oss av AWS infrastruktur och dess Warehouse Redshift med flera lager:

1. **Ingestion (staging‑lager):**
   1. Garmin‑data hämtas med hjälp av *dlt* (data loading tool) och laddas till **Redshift**.
   2. StrengthLog‑CSV (historik) och manuella input via Taipy förs också in till samma staging.
2. **Refined‑lager (dbt‑modeller):**
   1. Datan harmoniseras och modelleras i ett **stjärnschema** bestående av en gemensam faktatabell (fact\_workouts) samt dimensionstabeller (dim\_date, dim\_athlete, dim\_sport, dim\_exercise).
   2. Detta säkerställer konsekventa nycklar och gemensam granularitet för både cardio och styrka.
3. **Mart‑lager:**
   1. Vi bygger BI‑vänliga vyer utifrån fact‑tabellen, t.ex. cardio\_summary och strength\_summary.
   2. Dessa tabeller används för att snabbt kunna bygga visualiseringar.
4. **Visualisering (Taipy):**
   1. Dashboards byggs i *Taipy* och kopplas direkt till mart‑lagret i Redshift.
   2. Här kan användaren filtrera per atlet, sport eller tidsperiod samt se nyckel‑KPIs, kartor, trender och PB‑markeringar.

En bild som visar text, skärmbild, diagram, Teckensnitt

AI-genererat innehåll kan vara felaktigt.

## Eventuella avgränsningar

Valet av AWS/Redshift sker för att det skulle kunna gagna Eriks möjligheter att få en attraktiv LIA plats. Men om det sen skulle visa sig för komplicerat att genomföra projektet likväl via AWS/redshift så får man planera att göra det i Azure i stället som vi lär oss att hantera i Big data cloud kursen.

Om tidsbrist skulle uppstå så är ett alternativ att slopa antingen Garmin eller Strengthlog datan och bara fokusera på en datakälla.

## Sammanfattning

Slutresultatet blir en träningsdashboard som automatiskt hämtar och konsoliderar träningsdata från både Garmin Connect och StrengthLog, lagrar den i en modern data pipeline, och presenterar en samlad bild i ett interaktivt och användarvänligt gränssnitt (Taipy).