

Projekt kryptokollen



Genererad av Dalle-3

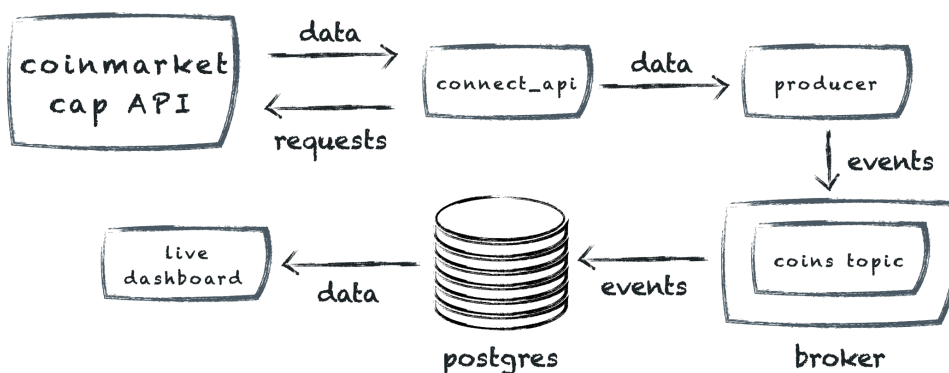
Syfte

Projektet syftar till att implementera en data platformslösning för strömmande data. Förutom att använda ett visst antal tekniker finns ett fokus kring att jobba tillsammans i ett data engineering team. Detta innebär både agil utveckling och att kunna använda git och github i ett team.

Bakgrund

Företaget Kryptokollen AB vill bygga en användarvänlig dashboard för att visualisera livedata och statistik för olika kryptovalutor. En ide med deras produkt kryptokoll är att man ska kunna som kund kunna luta sig tillbaka och se sina favoritkryptovalutors priser och data ändras i realtid.

Kryptokollen har haft en BI-utvecklare (BI) och en Data engineer (DE) som tillsammans med product owner (PO) kommit fram till en övergripande arkitektur. DE och BI skulle egentligen velat göra färdigt produkten med komplett dokumentation, men företagets VD ville snabbt se en prototyp för att kunna söka investering.



Dessvärre blev DE headhuntad till ett annat bolag och valde att säga upp sig. Innan han slutade implementerade han modellen till viss del och BI-utvecklaren lyckades bygga en initial dashboard. När DE slutade skickade företaget BI-utvecklaren på konsultuppdrag hos en kund.

VDn visade detta för sina investerare och fick in riskkapital från bland annat Yohanna. Yohanna som tidigare investerat i dataintensiva projekt förstår vikten av noggrannhet, då det oftast straffar sig av att ta genvägar.

Därför har hon gett PO mer befogenheter att göra det här projektet noggrant. Dessutom är det viktigt med fler funktionaliteter i dashboarden som ger bättre kundupplevelse.

Roller i organisationen

Person	Roll	Beskrivning
Yohanna	riskkapitalist	investerat i dataintensiva projekt, begränsad teknisk kompetens, bra businesskunskaper
Roager	VD	tuff förhandlare, duktig på business
Börje	product owner	begränsad teknisk kompetens, bra kontakt med både Yohanna och Jacobi
Feliz	frontendutvecklare	sitter för närvarande med annat projekt, men kan assistera med git och github
Yulia	BI-utvecklare	sitter ute hos kund

Scenario

Yohanna har varit med i tidigare projekt som letts av studerande på yrkeshögskolan inom data engineering och varit mycket nöjd över resultaten. Därför bestämmer sig Yohanna tillsammans med Roager att ta in LIA till detta projekt också.

Teamet i Kryptokollen skissar hastigt en onboardingplan och en kravställning.

Onboarding

Roager berättar bakgrund

Feliz går igenom hur ni ska jobba med git och github i det här projektet

- bjuda in teamet
- git branches
- pull requests
- github projects
- tilldela personer till tasks
- issues

Börje går igenom agil projektmetodik i kryptokollen

- jobba enligt kanban
- backlog refinement
- definition of done
- 1 task åt gången

Uppgift 0 - uppvärmning

Det som beskrevs i bakgrunden av det som DE och Yulia satte upp hittar ni i föreläsningarna 10 och 13 i kursrepot. Föreläsning a1 och a2 behandlar git och github i team. Se till så att varje person självständigt går

igenom dessa föreläsningar och lär sig innehållet.

Detta kan göras asynkront, dvs att andra uppgifter kan göras utan att ha gått igenom alla lektionerna.

Uppgift 1 - setup

Githubrepo

En person sätter upp ett publikt githubrepo och bjuder in övriga. Sätt även upp github projects för att ha en kanbanboard där ni kan se vilka tasks som behöver göras och vilka som jobbar med vilken task.

VIKTIGT

Kom ihåg att jobba i egna branches och kör pull från main branchen innan ni gör pull request till main. På så sätt löses eventuella merge conflicts i sin egna branch.

Gör många committs, vänta inte på att göra få stora committs. På så sätt versionshanteras koden bättre, ni får backup och kodbasen utvecklas över tid.

Virtual environment

En person installerar de paket ni behöver och tar fram en requirementsfil som pushas till github. Övriga personer installerar de olika dependencies som finns i requirementsfilen.

Producer

Ändra så att producern plockar in data med 30-60 sekunders intervall för att både spara på antalet API-anrop och att datan inte uppdateras så frekvent.

Uppgift 2 - EDA av API

Börje berättar att DE tidigare sagt att coinmarketcap API har mycket att bjuda på som DE inte haft tid att undersöka. Gör explorativ undersökning av följande endpoints:

- `v1/cryptocurrency/quotes/latest`
- `v1/cryptocurrency/listings/latest`

Här är det viktigt att varje person gör en egen EDA så att varje person sätter in sig i rådatan. Ni skulle kunna ha en folder som heter explorations där ni lägger in jupyter notebook fil för varje person ex `eda_person1.ipynb`, `eda_person2.ipynb` ...

Uppgift 3 - connect api

DE har hastigt byggt in api-anropen direkt i `coin_producer.py`, men ser vi på arkitekturen hade de tänkt att ha en separat `conenct_api` modul. På detta sätt kan man återanvända koden för API-anrop till eventuellt flera producers i framtiden. Refaktorera och lyft ut denna kod i egen modul.

Uppgift 4 - annan kryptovaluta

DE och Yulia har byggt mycket enkel visualisering av bitcoindata och en mycket enkel dataframe som visar datan. Nu vill Börje att ni ska testa en annan kryptovaluta som inte är bitcoin.

Uppgift 5 - välja valuta

Som ni ser är priset i USD, men denna produkt är avsedd till Sverige och de nordiska länderna. Se till att ha en filtrering på de nordiska valutorna så att prisgrafen uppdateras till rätt valuta.

Här finns två sätt att lösa det på

1. hårdkoda exchange rate för valutorna
2. plocka från ett API ex [exchangerates](#) (BONUS)

Uppgift 6 - bygga ut dashboarden

Visa mer statistik i dashboarden som volym, volymändring, prisändring mm. Tänk på att använda relevanta prefix ex istället för att skriva 94032945, skriv 94.0M så blir det lättare att läsa.

Uppgift 7 - välja kryptovalutor (BONUS)

Ta in två kryptovalutor genom data pipelinen. Här får ni fundera över arkitekturen, men 2 producers och 2 consumers kan vara bra att ha. Fundera över hur ni kan modularisera och återanvända kod. Rita upp er arkitektur.

Presentera för stakeholders

Ni har 10 minuter per grupp att presentera ert projekt. Tänk på att er publik består av Börje, Feliz, Yohanna, Roager, Yulia och klassen så

- presentera dashboarden
- presentera kort hur ni jobbat agilt
- kort presentera er implementation
- hur ni jobbat med git och github i teamet

VIKTIGT

ALLA ska presentera

Individuell inlämningsuppgift

Skriv en rapport på 1-2 sidor där ni beskriver tekniskt de olika stegen i projektet. Skriv på engelska och använd lämplig teknisk terminologi. Här är exempel på frågor att förhålla sig till

- hur har datan flödat från upstreams till downstreams?
- beskriv hur ni har transformerat datan
- vad har ni gjort för avvägningar i dashboarden?
- vad är syftet bakom de olika delarna i er arkitektur?
- vilka teknologier har ni använt och till vilka syften i det här projektet?
- om ni hade mer tid, hur skulle ni bygga vidare ert projekt?
- kort reflektion kring hur grupparbetet fungerat

beskriv gärna fler punkter, men håll dig till max 2 sidor för rapporten

Bedömning

Projektet bedöms individuellt, ni kommer skicka in er gemensamma kodbas i github och kanbanboardet. Där går det att följa vad respektive person gjort under projektet.

Godkänt

- gjort uppgifterna korrekt i grupp
- varit aktiv i projektet, utfört relevanta tasks
- gjort flera relevanta commits med pull requests mot main
- individuella uppgiften är gjord på grundläggande nivå

Väl godkänt

- projektet är utfört på tillräcklig hög nivå
- koden är strukturerad, lätt att följa och kod har återanvänts på ett effektivt sätt
- bonusuppgifterna är lösta korrekt