microservices (calculation services):

we will recive order object. From “pub/sub”

recivce from sub: order entitie: id, restaurantid, totalPrice, orderDatetime , DeliveringDatetime, customerId, deliveringadresse,

Table: entity agent bonus: agentid, amount, DeliveringDatetime, order id

Table entity mtogo earning: ? amount, datetime, order id

Table entity restaurant earning ?: amount, datetime, orderid

we have 3 main logic:

1. Mtogo earning: metogo fee: based on order.totalrpice

First 100 Kr 16%

Amount to 500 Kr 15%

Amount to 1.000 Kr 14%

Anything more 13%

1. restaurant earning:

order.total price – mtogo earning.

1. Agent bonus:

Employees will get 5 % as bonus from Mtogo.amount value before VAT (moms) only between 18:00-06:00

Kode:

1. Kode – logic – entity – value object – aggregate – error handling ? (\*)
2. Test
3. Design pattern (Filter – Port - Adapter) (\*)
4. Pub-sub (Hvad forventer du at modtag ? sub) (hvad forventer du at sende afsted ? pub) (\*)
5. Rest api (\*)
6. Dockerfile (image) (\*)
7. Database connection (hvordan er bedst at gøre med docker compose) ? (\*)
8. Api gateway (kræver at opsætte først google cloud – vent indtil vider) (\*)

Opsætning - github:

1. Projekt – user stories osv.
2. Git-action – til main
   1. CI dvs. Test alt
   2. CD derefter deploy det.

Opsætning – google cloud:

1. Install Cgloud – sæt vores user/project osv. (\*)
2. Opret login(rolebased) (\*)
3. Api gateway (\*)
4. Deploy vores project – dvs. flere server, kubernetes, pubsub (\*)