

Vraag 1: Welke belangrijke tradeoffs zijn er gemaakt?

Een belangrijke tradeoff is dat ze een aparte applicatie gebuiken, die ook data haalt uit de operational database, om de verkoop te monitoren en hierdoor zal de hoofdapplicatie trager werken terwijl de monitor applicatie aan het werken is.

Ook is er gekozen voor een gelayerde architectuur waardoor ondermeer de belangrijke logica extra beveiligd kan worden, dit ten koste van performantie aangezien de layers elkaar steeds moeten aanspreken.

Vraag 2: Waar zit het grootste risico in deze architectuur?

De accounting app is het grootste risico aangezien deze niet in eigen handen is zal het moeilijk zijn om fouten in deze applicatie op te lossen. Ook bij verdere uitbreidingen in de toekomst kan de accounting app voor problemen zorgen.

Vraag 3: Met welke crosscutting concerns zijn er rekening gehouden?

Er wordt rekening gehouden met exception handling. Doordat er static validation rules worden toegepast in de front end zullen validation exceptions opgevangen worden. Ook wordt er rekening gehouden met communication, de hoeveelheid communicatie tussen de verschillende layers wordt beperkt en aangeroepen via een facade.

Vraag 4: Hoe zou je logging in deze applicatie implementeren?

Elke layer zal moeten loggen naar een zelfde store of naar een eigen store op zo een manier dat de resultaten nadien kunnen samenhangen. Er moet wel opgelet worden dat er hierbij geen gevoelige informatie gelogd wordt.