**Architectuur**

**1/ Hoe zou je security in deze applicatie implementeren? Illustreer met een tekening.**

Security is een ‘crosscutting concern’ doorheen het gehele systeem. Dit betekent dat er rekening zou moeten gehouden worden op elk vlak. Bij de OGRS application architecture is er niets vermeld over de security van het systeem, terwijl dit heel belangrijk is. Security zorgt ervoor dat de de applicatie en het systeem bestendig zijn tegen aanvallen van buitenaf.

Ten eerste zouden de database streng beveiligd moeten worden. Deze bevatten gevoelige gegevens van klanten die niet verkrijgbaar zouden mogen zijn. Ook de encryptie van data is belangrijk. Wanneer gegevens doorheen de architectuur van het systeem lopen zouden deze geëncrypteerd moeten worden.

Ten slotte moeten de accounts van de gebruikers zelf beveiligd worden. Doormiddel van sessions en wachtwoord encryptie kan men dit doen.

**2/ Hoe zou je logging in deze applicatie implementeren? Illustreer met een tekening.**

Logging van bepaalde dingen is ook een ‘crosscutting concern’ waarmee rekening moet gehouden worden. Bij bepaalde zaken is logging handig.

De database zou van logging moeten voorzien worden, als dit niet ingebouwd is. Dit zorgt ervoor dat alle communicatie dat met de database is gebeurd op te vragen is.

Event logging is ook handig bij de applicatie. Het zorgt ervoor dat alle ‘events’ gelogd worden. Dat betekent dus dat alles dat gebeurd word bijgehouden.

**3/ Waar zit het grootste risico in deze architectuur?**

Het grootste risico in deze architectuur bevindt zich bij de ‘business layer’. Deze is verbonden met de 2 verschillende front-end gedeeltes (‘online shop’ en (catalogue management’) en de bath file service. De front-end gedeeltes zijn dus gescheiden van de database door de business layer en hangen dus hiervan af voor de nodige data. De business layer is ook verbonden met het ‘accounting systeem’. De business layer moet dus altijd correct werken (geen bugs bijvoorbeeld), anders werkt de rest van de applicatie niet meer. Ook moet de business layer de nodige stress aankunnen om te blijven werken.

**4/ Beschrijf in je eigen woorden wat software architectuur is.**

Software architectuur definieert de high level structuren van een software systeem. De verschillende structuren en componenten van het systeem en hun relaties worden duidelijk gemaakt. Door middel van documentatie wordt er duidelijk gemaakt aan de stakeholders wat de architectuur inhoudt. De documenatie bestaat uit een ‘Blueprint’ die verduidelijking geeft over het huidige plan (van de architectuur) en de plannen voor de toekomst. Het geeft een gedetailleerde visie over bijvoorbeeld de nodige kosten en infrastructuur dat nodig is voor het systeem/het project. Het geeft ook de nodige optie voor toekomstige veranderingen aan het systeem dat tegenwoordig heel belangrijk is. Het is noodzakelijk dat software makkelijk verandert kan worden.

Die blueprint van de architectuur bestaat uit 5 delen (views) die elk deel van het systeem grondig uitleggen en uitplannen:

* Business architecture views
* Functional architecture views
* Information architecture views
* Application or technical architecture views
* Infrastructure architecture views